BR101/D
REV 30

# Technical and Applications Literature

## Selector Guide and Cross References

Effective Date 2nd Quarter 2000

Digital DNA™ from Motorola

# Contents

# Introduction

To complement the industry's broadest line of semiconductor products, Motorola offers a complete library of User's Manuals describing the capabilities of its products in circuit and system design. These documents are supplemented by Selector Guides and Application Notes, and by Data Books which detail the electrical characteristics of the products.

All literature items can be obtained by mail from the Literature Distribution Centers, and can also be ordered on-line via Motorola's website.

This document provides abstracts, listings and a selector guide for all of this literature, together with a Device Cross Reference for the Application Notes.

## Motorola SPS Internet Server

Motorola has set up a website to deliver its technical data to the global Internet community. Technical data such as the complete Master Selection Guide, along with the North American price book, are available on the site with full search capabilities. Other data include abstracts of data books, application notes and selector guides. All have easy text search capability. Ordering literature from the Literature Distribution Center is available on line. Other features of Motorola's Internet server include the availability of a searchable press release database, technical training information with on-line registration capabilities, an on-line technical support form to send technical questions and receive answers through email, information on product groups, full search capabilities of device models, a listing of the Domestic and International sales offices, and links directly to other Motorola websites.

**How to reach us:**

After accessing the Internet, use the following URL: **http://motorola.com/semiconductors/**

# Ordering Information

An Order Form is no longer included in this document. Telephone or fax your nearest Distribution Center to place an order. Also, remember that orders can be placed on the **World Wide Web** – for further details see the information on the previous page.

## Literature Centers

| | **How to reach us:** |
|---|---|
| USA/EUROPE/locations not listed: | Motorola Literature Distribution<br>P.O. Box 5405<br>Denver, Colorado 80217<br>*Phone: 1-303-675-2140 or 1-800-441-2447*<br>*Fax: 1-303-675-2150* |
| JAPAN: | Motorola Japan Ltd.<br>SPS, Technical Information Center<br>3-20-1, Minami-Azabu, Minato-ku<br>Tokyo 106-8573 Japan.<br>*Phone: 81-3-3440-3569* |
| ASIA/PACIFIC: | Motorola Semiconductors H.K. Ltd.<br>Silicon Harbour Centre, 2 Dai King Street<br>Tai Po Industrial Estate<br>Tai Po, N.T., Hong Kong.<br>*Phone: 852-26668334* |

# Applications Documents

## Introduction

Motorola's Applications Literature provides guidance to the effective use of its semiconductor families across a broad range of practical applications. Many different topics are discussed – in a way that is not possible in a device data sheet – from detailed circuit designs complete with PCB layouts, through the matters to consider when embarking on a design, to complete overviews of a microprocessor family and its design philosophy.

Information is presented in the form of Application Notes and Article Reprints (originally published[1] in the electronics press), plus detailed Engineering Bulletins, Benchbriefs[2], Design Concepts and APRs[3]. This section provides a guide to these items; it includes an abstract of each document, a Selector Guide listing documents under subject or device-type headings, and a Device Cross Reference listing them by featured devices. Documents new to this issue are highlighted.

The Application Notes, Article Reprints, Engineering Bulletins, and Design Concepts are included to enhance the user's knowledge and understanding of Motorola's products. However, before attempting to design-in a device referenced in these documents, contact the local Motorola supplier for product availability and available application support.

Each section of the Applications Literature Selector Guide also includes cross references to a selection from Motorola's extensive range of Data Books, Brochures, Technical Bulletins and Selector Guides which may provide further relevant information.

Information in this document is given in good faith and no liability is accepted for errors or omissions. Includes literature available as of April 1, 2000.

1 Article Reprints are reproduced with the permission of the original publisher.

2 A Benchbrief is an Engineering Bulletin produced by Motorola Asia-Pacific Group.

3 APRs are applications documents relating specifically to Digital Signal Processing.

™ All trademarks are recognized.

# Applications Documents

## Index

# Applications Documents

## Device Cross Reference

*This quick-reference list indicates where specific components are featured in Application Notes, Article Reprints, Engineering Bulletins and Design Concepts.*

| | | |
|---|---|---|
| ASB200 | AN1651/D | ASB201 – Uncompensated Series Sensor Module |
| | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| | AN1654/D | ASB210 – 10" H2O Sensor Module |
| | AN1655/D | ASB200 – Motorola Sensor Development Controller Board |
| ASB201 | AN1651/D | ASB201 – Uncompensated Series Sensor Module |
| ASB202 | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| ASB205 | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| ASB210 | AN1654/D | ASB210 – 10" H2O Sensor Module |
| CPU16 | AN476/D | CPU16 and the Configurable Timer Module (CTM) in Engine Control |
| | AN1283/D | Transporting M68HC11 Code to M68HC16 Devices |
| | *EB251/D | How to Calculate Instruction Times on the MC68HC16 |
| | EB269/D | Using the SCI on Modular MCUs: An Example |
| | *EB276/D | Using the ITC Function on the Time Processor Unit A |
| | EB281/D | Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers |
| CPU32 | EB253/D | How to Use the Lookup and Interpolate Instruction on the CPU32 |
| | EB269/D | Using the SCI on Modular MCUs: An Example |
| | *EB276/D | Using the ITC Function on the Time Processor Unit A |
| | EB281/D | Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers |
| | *EB310/D | Using Bus Error Stack Frames to Diagnose CPU32 Released Write Faults |
| CS4218 | *AN1790/D | Programming the CS4218 CODEC for Use with DSP56300 Devices |
| DEVB103 | AN1249/D | Brushed DC Motor Control Using the MC68HC16Z1 |
| DEVB114 | AN1305/D | An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a… |
| DEVB129 | AN1304/D | Integrated Sensor Simplifies Bar Graph Pressure Gauge |
| DEVB147 | AN1309/D | Compensated Sensor Bar Graph Pressure Gauge |
| DEVB158 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| DEVB160 | AN1316/D | Frequency Output Conversion for MPX2000 Series Pressure Sensors |
| DEVB173 | AN1324/D | A Simple Sensor Interface Amplifier |
| DMA08 | AN1711/D | DMA08 Systems Compatibilities |
| DS1307 | AN1759/D | Add a Non-Volatile Clock to the MC68HC705J1A |
| DS1620 | AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| DS2401 | AN1757/D | Add a Unique Silicon Serial Number to the HC05 |
| DS2502 | AN1757/D | Add a Unique Silicon Serial Number to the HC05 |
| DSP56ADC16 | APR8/D | Principles of Sigma-Delta Modulation for Analog-to-Digital Converters |
| | APR10/D | DSP96002 Interface Techniques and Examples |
| DSP56L811 | APR21/D | Software UART on the DSP56L811 Using GPIO Port B |
| DSP56000 | ANE415/D | MC68HC11 Implementation of IEEE-488 Interface for DSP56000 Monitor |
| | APR3/D | Fractional and Integer Arithmetic Using the DSP56000 Family of General-Purpose Digital… |
| | APR4/D | Implementation of Fast Fourier Transforms on Motorola's DSP56000/DSP56001 and… |
| | APR5/D | Implementation of PID Controllers on the Motorola DSP56000/DSP56001 |
| | APR14/D | Conference Bridging in the Digital Telecomms Environment Using the Motorola DSP56000 |
| | APR15/D | Implementation of Adaptive Controllers on the Motorola DSP56000/DSP56001 |

| | | |
|---|---|---|
| DSP56000 (continued) | APR36/D | Interfacing the DSP560xx/DSP563xx Families to the Crystal CS4226 Multichannel Codec |
| DSP56001 | APR1/D | Digital Sine-Wave Synthesis Using the DSP56001/DSP56002 |
| | APR2/D | Digital Stereo 10-Band Graphic Equalizer Using the DSP56001 |
| | APR4/D | Implementation of Fast Fourier Transforms on Motorola's DSP56000/DSP56001 and… |
| | APR5/D | Implementation of PID Controllers on the Motorola DSP56000/DSP56001 |
| | APR6/D | Convolutional Encoding and Viterbi Decoding Using the DSP56001 with a V.32 Modem… |
| | APR7/D | Implementing IIR/FIR Filters with Motorola's DSP56000/DSP56001 |
| | APR9/D | Full-Duplex 32 kbit/s CCITT ADPCM Speech Coding on the Motorola DSP56001 |
| | APR11/D | DSP56001 Interface Techniques and Examples |
| | APR14/D | Conference Bridging in the Digital Telecomms Environment Using the Motorola DSP56000 |
| | APR15/D | Implementation of Adaptive Controllers on the Motorola DSP56000/DSP56001 |
| | DCE406/D | Interface for MC68000 to DSP56001 Host Port |
| | EB420/D | Converting DSP56001-Based Designs to the DSP56002 |
| DSP56002 | *AN1829/D | Software Differences Between the DSP56002 and the DSP56303 |
| | *AN1830/D | Hardware Differences Between the DSP56002 and the DSP56303 |
| | APR16/D | Calculating Timing Requirements of External SRAM for the 24-bit DSP56000 Family |
| | EB420/D | Converting DSP56001-Based Designs to the DSP56002 |
| DSP56004 | APR31/D | Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode |
| DSP56007 | APR31/D | Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode |
| DSP56009 | APR31/D | Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode |
| DSP56156 | APR404/D | G.722 Audio Processing on the DSP56100 Microprocessor Family |
| | APR405/D | Minimal Logic DRAM Interface for the DSP56156 |
| DSP56300 | AN1289/D | DSP5630x FSRAM Module Interfacing |
| | AN1751/D | DSP563xx Port A Programming |
| | AN1764/D | DSP56300 Enhanced Synchronous Serial Interface (ESSI) Programming |
| | AN1772/D | Efficient Compilation of Bit-Exact Applications for DSP563xx |
| | AN1781/D | Booting DSP563xx Devices Through the Serial Communication Interface (SCI) |
| | *AN1788/D | DSP563xx HI32 PCI Functions |
| | *AN1790/D | Programming the CS4218 CODEC for Use with DSP56300 Devices |
| | *AN1808/D | DSP56300 HI08 Host Port Programming |
| | APR20/D | Application Optimization for the DSP56300/DSP56600 Digital Signal Processors |
| | APR22/D | Application Conversion from the DSP56100 Family to the DSP56300/600 Families |
| | *APR26/D | Interfacing Flash Memory with the Motorola DSP56300 Family of Digital Signal Processors |
| | APR30/D | DSP56300 Assembly Code Development Using the Motorola Toolsets |
| | APR36/D | Interfacing the DSP560xx/DSP563xx Families to the Crystal CS4226 Multichannel Codec |
| | APR37/D | Implementing AC-link with ESAI |
| | APR38/D | Interfacing Serial EEPROM to DSP563xx |
| | APR40/D | Implementing Viterbi Decoder Using the VSL Instruction on DSP Families DSP56300… |
| DSP56301 | AN1780/D | DSP563xx HI32 as a PCI Agent |
| DSP56303 | AN1782/D | Converting DSP56303 Designs to DSP56307 Designs |
| | *AN1804/D | DSP56307 Port A-to-HI08 Interface |
| | *AN1829/D | Software Differences Between the DSP56002 and the DSP56303 |
| | *AN1830/D | Hardware Differences Between the DSP56002 and the DSP56303 |
| | *APR25/D | Interfacing Fast SRAM to Motorola's DSP56300 Family of Digital Signal Processors |
| | *APR26/D | Interfacing Flash Memory with the Motorola DSP56300 Family of Digital Signal Processors |
| | *APR27/D | Interfacing EPROM and EEPROM Memory with Motorola's DSP56300 Family of Digital Signal… |
| DSP56304 | APR33/D | ROM Software Patching on the Motorola DSP56304 |
| DSP56307 | AN1782/D | Converting DSP56303 Designs to DSP56307 Designs |
| | *AN1804/D | DSP56307 Port A-to-HI08 Interface |
| | APR39/D | Programming the DSP56307 Enhanced Filter Coprocessor (EFCOP) |
| DSP56309 | *AN1804/D | DSP56307 Port A-to-HI08 Interface |
| DSP56362 | APR37/D | Implementing AC-link with ESAI |
| DSP56362EVM | *AN1810/D | Developing DSP56364 Software Using the DSP56362 EVM |
| DSP56364 | *AN1810/D | Developing DSP56364 Software Using the DSP56362 EVM |
| DSP56600 | APR20/D | Application Optimization for the DSP56300/DSP56600 Digital Signal Processors |
| | APR22/D | Application Conversion from the DSP56100 Family to the DSP56300/600 Families |
| | APR40/D | Implementing Viterbi Decoder Using the VSL Instruction on DSP Families DSP56300… |
| DSP56690 | *AR366/D | Motorola Cellular DSP Does It All |
| DSP56800 | DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |

| | | |
|---|---|---|
| DSP96002 | APR4/D | Implementation of Fast Fourier Transforms on Motorola's DSP56000/DSP56001 and… |
| | APR10/D | DSP96002 Interface Techniques and Examples |
| G4 | *AN1794/D | PowerPC Backside L2 Timing Analysis for the PCB Design Engineer |
| | *AN1795/D | Designing G4 Systems |
| HI08 | *AN1804/D | DSP56307 Port A-to-HI08 Interface |
| | *AN1808/D | DSP56300 HI08 Host Port Programming |
| ITC122 | AN1702/D | Brushless DC Motor Control Using the MC68HC705MC4 |
| | *AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| ITC127 | AN1702/D | Brushless DC Motor Control Using the MC68HC705MC4 |
| | *AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| ITC132 | AN1624/D | ITC137 68HC708MP16 Motion Control Development Board |
| | *AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| ITC137 | AN1624/D | ITC137 68HC708MP16 Motion Control Development Board |
| LM311 | AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| | AN1518/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| | AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| LM339 | AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| | AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| LM358 | AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| | AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| LM3914 | AN1309/D | Compensated Sensor Bar Graph Pressure Gauge |
| | AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| M68F375 | *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| M68FDDIADS | EB406/D | Getting Started with the FDDI ADS Board |
| M68HC05 | AN442/D | Driving LCDs with M6805 Microprocessors |
| | AN477/D | Simple A/D for MCUs without Built-In A/D Converters |
| | AN1219/D | M68HC08 Integer Math Routines |
| | AN1222/D | Arithmetic Waveform Synthesis with the HC05/08 MCUs |
| | AN1227/D | Using 9346 Series Serial EEPROMs with 6805 Series Microcontrollers |
| | AN1262/D | Simple Real-Time Kernels for M68HC05 Microcontrollers |
| | AN1723/D | Interfacing MC68HC05 Microcontrollers to the IBM AT Keyboard Interface |
| | AN1744/D | Resetting Microcontrollers During Power Transitions |
| | AN1752/D | Data Structures for 8-bit Microcontrollers |
| | AN1757/D | Add a Unique Silicon Serial Number to the HC05 |
| | AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| | AN1783/D | Determining MCU Oscillator Start-up Parameters |
| | EB181/D | Frequently Asked Questions and Answers: M68HC05 Family MCAN Module |
| | EB410/D | PASM05 to INTROL M68HC05 Assembler Conversion |
| | EB413/D | Resetting MCUs |
| | EB416/D | Modular Target Cables for Motorola Development Systems |
| M68HC05Px | AN1736/D | Variations in the Motorola MC68HC05Px Family |
| M68HC08 | AN1218/D | HC05 to HC08 Optimization |
| | AN1219/D | M68HC08 Integer Math Routines |
| | AN1222/D | Arithmetic Waveform Synthesis with the HC05/08 MCUs |
| | AN1744/D | Resetting Microcontrollers During Power Transitions |
| | AN1752/D | Data Structures for 8-bit Microcontrollers |
| | AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| | AN1783/D | Determining MCU Oscillator Start-up Parameters |
| | EB416/D | Modular Target Cables for Motorola Development Systems |
| M68HC11 | AN427/D | MC68HC11 EEPROM Error Correction Algorithms in C |
| | AN432/D | 128K byte Addressing with the M68HC11 |
| | AN974/D | MC68HC11 Floating-Point Package |
| | AN997/D | CONFIG Register Issues Concerning the M68HC11 Family |
| | AN1010/D | MC68HC11 EEPROM Programming from a Personal Computer |
| | AN1058/D | Reducing A/D Errors in Microcontroller Applications |
| | AN1060/D | MC68HC11 Bootstrap Mode |
| | AN1064/D | Use of Stack Simplifies M68HC11 Programming |
| | *AN1237/D | Using M68HC11 Microcontrollers with WSI Programmable Peripheral Devices |
| | AN1326/D | Barometric Pressure Measurement Using Semiconductor Pressure Sensors |
| | AN1744/D | Resetting Microcontrollers During Power Transitions |

| | | |
|---|---|---|
| M68HC11 (continued) | AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| | AN1783/D | Determining MCU Oscillator Start-up Parameters |
| | ANE405/D | Bi-Directional Data Transfer Between MC68HC11 and MC6805L3 Using SPI |
| | ANE415/D | MC68HC11 Implementation of IEEE-488 Interface for DSP56000 Monitor |
| | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| | EB294/D | How to Write the 64-Cycle Time-Protected Registers on M68HC11 Development Tools |
| | *EB299/D | Why M68HC711D3PGMR Software Does Not Run on 486 33MHz Computers |
| | *EB303/D | Handling Considerations for Avoiding Intermittent Programming and Execution Failures… |
| | EB413/D | Resetting MCUs |
| | EB416/D | Modular Target Cables for Motorola Development Systems |
| M68HC11EVB | *EB187/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVB |
| M68HC11EVBU | *AN4003/D | Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal… |
| | *EB291/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC11EVBU |
| M68HC11EVM | EB191/D | Programming EPROM and EEPROM on the M68HC11EVM |
| M68HC12 | AN1280A/D | Using the Callable Routines in D-Bug12 |
| | AN1284/D | Transporting M68HC11 Code to M68HC12 Devices |
| | AN1295/D | Demonstration Model of fuzzyTECH Implementation on M68HC12 |
| | AN1716/D | Using M68HC12 Indexed Indirect Addressing |
| | AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| | AN1783/D | Determining MCU Oscillator Start-up Parameters |
| M68HC16 | AN461/D | An Introduction to the HC16 for HC11 Users |
| | AN1230/D | A Background Debugging Mode Driver Package for Modular Microcontrollers |
| | AN1283/D | Transporting M68HC11 Code to M68HC16 Devices |
| | *EB251/D | How to Calculate Instruction Times on the MC68HC16 |
| | EB264/D | Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the… |
| | EB265/D | Generating Edge-Sensitive Interrupts on the MC683xx and MC68HC16 Families of MCUs |
| | *EB266/D | Unexplained Three-Stating of the Address Bus on M68300 and M68HC16 Devices |
| | *EB274/D | Generating Interrupts on the Time Processor Unit |
| | EB277/D | Coherency in the Time Processor Unit (TPU) |
| | EB279/D | Low Output Levels on Output Pins |
| | EB305/D | Startup Problems When Using a Software Background Mode Debugger and Booting from… |
| | TPUPN00/D | Using the TPU Function Library and TPU Emulation Mode |
| M68HC16Z1EVB | EB252/D | MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers |
| | EB306/D | Using Exercise 7 on the M68HC16Z1EVB and the Necessity of Word Alignment |
| M68HC711D3 | *EB299/D | Why M68HC711D3PGMR Software Does Not Run on 486 33MHz Computers |
| M68HC711E9PGMR | *EB184/D | Enabling the Security Feature on the MC68HC711E9 Devices with PCbug11 on the… |
| | *EB188/D | Enabling the Security Feature on M68HC811E2 Devices with PCbug11 on the… |
| | *EB189/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC711E9PGMR |
| M68HC711EPGMR | *EB185/D | Simplify MC68HC711E9 EPROM Programming with PCbug11 and the M68HC711EPGMR… |
| M68ICD16 | EB252/D | MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers |
| M200 | *AN1817/D | MMC20xx M•CORE OnCE Port Communication and Control Sequences |
| M6805 | AN442/D | Driving LCDs with M6805 Microprocessors |
| M68300 | AN1200/D | Configuring the M68300 Family Time Processing Unit (TPU) |
| | AN1230/D | A Background Debugging Mode Driver Package for Modular Microcontrollers |
| | EB264/D | Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the… |
| | EB265/D | Generating Edge-Sensitive Interrupts on the MC683xx and MC68HC16 Families of MCUs |
| | *EB266/D | Unexplained Three-Stating of the Address Bus on M68300 and M68HC16 Devices |
| | EB268/D | Starting and Stopping the Time Processor Clock Using the Background Debug Mode |
| | EB277/D | Coherency in the Time Processor Unit (TPU) |
| | EB279/D | Low Output Levels on Output Pins |
| | EB305/D | Startup Problems When Using a Software Background Mode Debugger and Booting from… |
| | *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |
| | EB414/D | Low Power Write Enable Generation for M68300 Family Microprocessors |
| | TPUPN00/D | Using the TPU Function Library and TPU Emulation Mode |
| | TPUPN01/D | Queued Output Match TPU Function (QOM) |
| MAC4DC | *AN1663/D | Low Cost Universal Motor Sensorless Phase Angle Drive System |
| MC68705B16 | AN1612/D | Shock and Mute Pager Applications Using Accelerometer |
| MC68EC040 | DC414/D | An 8-bit EPROM Interface for an MC68EC040/MC68360 System |
| MC68F333 | AN1255/D | MC68F333 Flash EEPROM Programming Utilities |
| | AN1724/D | Implementing SCI Receive and Transmit Buffers in C |

| | | |
|---|---|---|
| MC68F333 (continued) | *EB254/D | Setting the Programming Voltage of Modular Microcontrollers with FLASH EEPROM |
| MC68HC(7)05J1A | AN1292/D | Adding a Voice User Interface to M68HC05 Applications |
| MC68HC(8)05K3 | AN1288/D | Programming the MC68HC(8)05K3's Personality EEPROM on the MMDS and MMEVS |
| MC68HC05 | AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| MC68HC05B16 | AN1571/D | Digital Blood Pressure Meter |
| | AN1611/D | Impact and Tilt Measurement Using Accelerometer |
| MC68HC05B4 | ANE416/D | MC68HC05B4 Radio Synthesizer |
| MC68HC05B5 | AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| MC68HC05B6 | AN1097/D | Calibration-Free Pressure Sensor System |
| MC68HC05BD3 | *AN-HK-24/H | Software Emulation of DDC1 Hardware using HC05BD3 |
| MC68HC05C0 | AN1286/D | MC68HC05C0 Bus Structure Design |
| MC68HC05C4 | AN991/D | Using the Serial Peripheral Interface to Communicate Between Multiple Microcomputers |
| | AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| MC68HC05C5 | AN1066/D | Interfacing the MC68HC05C5 SIOP to an I2C Peripheral |
| MC68HC05Cx | AN1298/D | Variations in the Motorola MC68HC(7)05Cx Family |
| MC68HC05F2 | AN-HK-17/H | MC68HC05F2 DTMF Output Low Voltage Active Filter |
| MC68HC05F6 | AN-HK-12/H | MC68HC05F6 Tone Pulse Dialer |
| MC68HC05J1 | AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| MC68HC05JB2 | AN1732/D | A Universal Serial Bus Gamepad Device using the MC68HC05JB2 |
| MC68HC05JJ | AN1738/D | Instruction Cycle Timing of MC68HC05JJ/JP Series Microcontrollers |
| | AN1740/D | Applications Using the Analog Subsystem of MC68HC05JJ/JP Series Microcontrollers |
| | AN1741/D | In-Circuit and Emulation Considerations for MC68HC05JJ/JP Series Microcontrollers |
| MC68HC05JJ6 | *AN1663/D | Low Cost Universal Motor Sensorless Phase Angle Drive System |
| MC68HC05JP | AN1738/D | Instruction Cycle Timing of MC68HC05JJ/JP Series Microcontrollers |
| | AN1740/D | Applications Using the Analog Subsystem of MC68HC05JJ/JP Series Microcontrollers |
| | AN1741/D | In-Circuit and Emulation Considerations for MC68HC05JJ/JP Series Microcontrollers |
| MC68HC05K1 | AN465/D | Secure Remote Control using the 68HC05K1 and the 68HC05P3 |
| MC68HC05L10 | AN-HK-13A/H | MC68HC05L10, an Enhanced Version of L9 for Handheld Equipment Applications |
| MC68HC05L11 | AN-HK-15/H | MC68HC05L11 Hand-Writing Applications |
| MC68HC05L6 | AN442/D | Driving LCDs with M6805 Microprocessors |
| MC68HC05L9 | AN-HK-10/H | MC68HC05L9 Microcomputer Applications Demo Board |
| MC68HC05MC4 | *AN1627/D | Low Cost High Efficiency Sensorless Drive for Brushless DC Motor Using MC68HC(7)05MC4 |
| MC68HC05P3 | AN465/D | Secure Remote Control using the 68HC05K1 and the 68HC05P3 |
| MC68HC05P9 | AN1586/D | Designing a Homemade Digital Output for Analog Voltage Output Sensors |
| MC68HC05SR3 | *AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| | *AN-HK-23/H | MC6805R3, MC68HC05SR3 Technical Comparison |
| MC68HC05V7 | AN1257/D | Using the M68HC05 Family On-Chip Voltage Regulator |
| MC68HC05X16 | EB421/D | The Motorola MCAN Module |
| MC68HC05X32 | EB421/D | The Motorola MCAN Module |
| MC68HC05X4 | AN464/D | Software Driver Routines for the Motorola MC68HC05 CAN Module |
| | EB421/D | The Motorola MCAN Module |
| MC68HC08 | AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| MC68HC08KH12 | AN1748/D | Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12 |
| MC68HC11 | AN495/D | RDS Decoding for an HC11-Controlled Radio |
| | AN1597/D | Longwave Radio Data Decoding Using an HC11 and an MC3371 |
| | EB195/D | How to Configure the Reset Pin on the MC68HC11 |
| | *EB198/D | Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11 |
| | *EB292/D | Initialization Considerations when Moving from the BUFFALO Monitor to a Standalone… |
| MC68HC11A0 | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB286/D | C Macro Definitions for the MC68HC11A8/A7/A1/A0 |
| MC68HC11A1 | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB286/D | C Macro Definitions for the MC68HC11A8/A7/A1/A0 |
| MC68HC11A7 | EB286/D | C Macro Definitions for the MC68HC11A8/A7/A1/A0 |
| MC68HC11A8 | AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB286/D | C Macro Definitions for the MC68HC11A8/A7/A1/A0 |

| | | |
|---|---|---|
| MC68HC11C0 | EB283/D | C Macro Definitions for the MC68HC11C0 |
| MC68HC11D0 | EB284/D | C Macro Definitions for the MC68HC(7)11D3/D0 |
| MC68HC11D3 | EB284/D | C Macro Definitions for the MC68HC(7)11D3/D0 |
| MC68HC11E0 | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC11E1 | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC11E20 | EB285/D | C Macro Definitions for the MC68HC(7)11E20 |
| MC68HC11E32 | EB419/D | ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker |
| | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC11E8 | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC11E9 | AN1285/D | Stepper Motor Control with an MC68HC11E9 Microcontroller |
| | *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC11ED0 | EB288/D | C Macro Definitions for the MC68HC11ED0 |
| | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC11EVBU | EB197/D | Using Pseudo-Interrupt Vectors on the M68HC11EVBU |
| MC68HC11F1 | EB289/D | C Macro Definitions for the MC68HC11F1 |
| MC68HC11G5 | AN432/D | 128K byte Addressing with the M68HC11 |
| MC68HC11K1 | *AN1242/D | High Performance M68HC11 System Design Using the WSI PSD4xx and PSD5xx Families |
| MC68HC11K4 | AN1215/D | PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers |
| MC68HC11KA2 | *EB312/D | Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs |
| MC68HC11KA4 | *EB312/D | Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs |
| MC68HC11KAx | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| MC68HC11KS2 | *EB312/D | Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs |
| MC68HC11KS8 | *EB312/D | Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs |
| MC68HC11KW1 | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| MC68HC11Kx | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| MC68HC11N4 | AN1215/D | PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers |
| MC68HC11P2 | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| MC68HC11PH8 | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| | EB419/D | ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker |
| | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC16 | EB259/D | Why MC68300 and MC68HC16 MCUs May Halt after Release of Reset |
| | EB260/D | Why an MC68300 or MC68HC16 Microcontroller May Fail to Release Reset |
| | EB261/D | Autovector Generation Using Chip Select Logic on MC68300 and MC68HC16 Devices |
| | *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |
| MC68HC16W1 | AN476/D | CPU16 and the Configurable Timer Module (CTM) in Engine Control |
| MC68HC16Y1 | AN461/D | An Introduction to the HC16 for HC11 Users |
| MC68HC16Z1 | AN461/D | An Introduction to the HC16 for HC11 Users |
| | AN1213/D | 16-bit DSP Servo Control with the MC68HC16Z1 |
| | AN1233/D | Using M68HC16 Digital Signal Processing to Build an Audio Frequency Analyzer |
| | AN1249/D | Brushed DC Motor Control Using the MC68HC16Z1 |
| | AN1254/D | Using the MC68HC16Z1 for Audio Tone Generation |
| | EB273/D | Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors |
| MC68HC16Z1EVB | EB309/D | Using Exercise 8 on the MC68HC16Z1EVB |
| MC68HC55 | *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| MC68HC68T1 | ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |
| MC68HC705 | AN499/D | Let the MC68HC705 Program Itself |
| MC68HC705B16 | AN1638/D | Offset Calibration of Gauge Pressure Sensor Using Parallel I/O Ports |
| | EB166/D | System Design Considerations: Converting from the MC68HC805B6 to the MC68HC705B16… |
| MC68HC705B16N | EB180/D | Differences between the MC68HC705B16 and the MC68HC705B16N |
| MC68HC705B5 | AN1305/D | An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a… |
| MC68HC705C8 | AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| | AN1226/D | Use of the 68HC705C8A in Place of a 68HC705C8 |
| | AN1228/D | Interfacing the HC05 MCU to the MC145051 A/D Converter |

| | | |
|---|---|---|
| MC68HC705C8A | AN1226/D | Use of the 68HC705C8A in Place of a 68HC705C8 |
| | AN1256/D | Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the… |
| | AN1734/D | Pulse Width Modulation Using the 16-Bit Timer |
| | AN1745/D | Interfacing the HC705C8A to an LCD Module |
| | AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| | AN1761/D | Interfacing the MC68HC705C8A to the X76F041 PASS SecureFlash |
| MC68HC705Cx | AN1298/D | Variations in the Motorola MC68HC(7)05Cx Family |
| MC68HC705J1A | AN1238/D | HC05 MCU LED Drive Techniques Using the MC68HC705J1A |
| | AN1239/D | HC05 MCU Keypad Decoding Techniques Using the MC68HC705J1A |
| | AN1240/D | HC05 MCU Software-Driven Asynchronous Serial Communication Techniques Using the… |
| | AN1241/D | Interfacing the MC68HC705J1A to 9356/9366 EEPROMs |
| | AN1256/D | Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the… |
| | AN1730/D | Digital Amplification of an Analog Signal Using the MC68HC705J1A |
| | AN1742/D | Programming the 68HC705J1A In-Circuit |
| | AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| | AN1758/D | Add Addressable Switches to the HC05 |
| | AN1759/D | Add a Non-Volatile Clock to the MC68HC705J1A |
| | AN1760/D | Interfacing the AD8402 Digital Potentiometer to the MC68HC705J1A |
| | *AN1820/D | Software I2C Communications |
| MC68HC705J2 | AN477/D | Simple A/D for MCUs without Built-In A/D Converters |
| | AN1737/D | Migrating from the MC68HC705J2 to the MC68HC705JJ7 |
| MC68HC705JJ7 | *AN1663/D | Low Cost Universal Motor Sensorless Phase Angle Drive System |
| | AN1737/D | Migrating from the MC68HC705J2 to the MC68HC705JJ7 |
| MC68HC705JP7 | AN1655/D | ASB200 – Motorola Sensor Development Controller Board |
| MC68HC705K1 | AN1228/D | Interfacing the HC05 MCU to the MC145051 A/D Converter |
| | *AN1746/D | Migrating from the MC68HC705K1 to the MC58HC705KJ1 |
| | AN1747/D | Migrating from the MC68HC705K1 to the MC68HC805K3 |
| MC68HC705KJ1 | *AN1746/D | Migrating from the MC68HC705K1 to the MC58HC705KJ1 |
| | AN1747/D | Migrating from the MC68HC705K1 to the MC68HC805K3 |
| MC68HC705L16 | AN1743/D | Scrolling Message Software |
| | AN1763/D | Driving LCD Displays Using the MC68HC705L16 Microcontroller |
| MC68HC705MC4 | AN1702/D | Brushless DC Motor Control Using the MC68HC705MC4 |
| | *AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| MC68HC705P6A | AN1775/D | Expanding Digital Input with an A/D Converter |
| MC68HC705P9 | AN1551/D | Low-Pressure Sensing with the MPX2010 Pressure Sensor |
| | AN1584/D | “Very Low Pressure” Smart Sensing Solution with Serial Communications Interface |
| | AN1585/D | High-Performance, Dynamically-Compensated Smart Sensor System |
| | AN1733/D | Implementing Caller ID Functionality in MC68HC(7)05 Applications |
| MC68HC705SR3 | *AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| MC68HC705V8 | AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| | AN1257/D | Using the M68HC05 Family On-Chip Voltage Regulator |
| MC68HC708LN56 | AN1287/D | MC68HC708LN56 LCD Utilities |
| | AN1762/D | Automatic Contrast Control of LCD Displays Using the MC68HC708LN56 Microcontroller |
| MC68HC708MP16 | AN1624/D | ITC137 68HC708MP16 Motion Control Development Board |
| | AN1712/D | “Get Your Motor Running” with the MC68HC708MP16 |
| | AN1728/D | Making Low-Distortion Waveforms with the MC68HC708MP16 |
| | *AN1792/D | Using an MC68HC908MR24 in Place of an MC68HC708MP16 |
| MC68HC711 | AN499/D | Let the MC68HC705 Program Itself |
| | *AN1237/D | Using M68HC11 Microcontrollers with WSI Programmable Peripheral Devices |
| | AN1597/D | Longwave Radio Data Decoding Using an HC11 and an MC3371 |
| | *EB303/D | Handling Considerations for Avoiding Intermittent Programming and Execution Failures… |
| MC68HC711D0 | EB284/D | C Macro Definitions for the MC68HC(7)11D3/D0 |
| MC68HC711D3 | EB284/D | C Macro Definitions for the MC68HC(7)11D3/D0 |
| MC68HC711E0 | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC711E1 | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC711E20 | *AN4003/D | Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal… |
| | *EB182/D | How the ROMON Bit Behaves on the E Series HC11 MCUs |
| | EB285/D | C Macro Definitions for the MC68HC(7)11E20 |
| | *EB293/D | Simplify MC68HC711E20 EPROM Programming with PCbug11 |

| | | |
|---|---|---|
| MC68HC711E20 (contd.) | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC711E32 | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC711E8 | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| MC68HC711E9 | AN1536/D | Digital Boat Speedometers |
| | AN1753/D | Implementing a FLASH Memory System in an MC68HC711E9 Design |
| | APR31/D | Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode |
| | *EB182/D | How the ROMON Bit Behaves on the E Series HC11 MCUs |
| | *EB184/D | Enabling the Security Feature on the MC68HC711E9 Devices with PCbug11 on the… |
| | *EB185/D | Simplify MC68HC711E9 EPROM Programming with PCbug11 and the M68HC711EPGMR… |
| | *EB187/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVB |
| | EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| | *EB296/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVBU |
| | EB298/D | Programming the BUFFALO Monitor into an MC68HC711E9 |
| MC68HC711EA9 | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC711K4 | *AN1242/D | High Performance M68HC11 System Design Using the WSI PSD4xx and PSD5xx Families |
| MC68HC711P2 | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| MC68HC711PH8 | EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| | EB422/D | Enhanced M68HC11 Bootstrap Mode |
| MC68HC805B6 | EB166/D | System Design Considerations: Converting from the MC68HC805B6 to the MC68HC705B16… |
| MC68HC805C4 | ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |
| MC68HC805K3 | AN1747/D | Migrating from the MC68HC705K1 to the MC68HC805K3 |
| MC68HC805L6 | ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |
| MC68HC811A2 | ANE415/D | MC68HC11 Implementation of IEEE-488 Interface for DSP56000 Monitor |
| MC68HC811E2 | *EB188/D | Enabling the Security Feature on M68HC811E2 Devices with PCbug11 on the… |
| | *EB189/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC711E9PGMR |
| | *EB291/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC11EVBU |
| | *EB295/D | Programming the EEPROM on the MC68HC811E2 with the M68HC11EVM Board |
| | EB301/D | Programming EEPROM on the MC68HC811E2 During Program Execution |
| MC68HC908AS60 | *AN1827/D | Programming and Erasing FLASH Memory on the MC68HC908AS60 |
| MC68HC908GP20 | AN1770/D | In-Circuit Programming of FLASH Memory in the MC68HC908GP20 |
| | *AN-HK-31/H | Using the MC68HC908GP32 in Place of MC68HC908GP20 |
| MC68HC908GP32 | *AN-HK-31/H | Using the MC68HC908GP32 in Place of MC68HC908GP20 |
| | *AN-HK-32/H | In-Circuit Programming of FLASH Memory in the MC68HC908GP32 |
| MC68HC908GR8 | *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| MC68HC908JB8 | *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| MC68HC908JK3 | *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| MC68HC908JL3 | *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| | *AN-HK-33/H | In-Circuit Programming of FLASH Memory in the MC68HC908JL3 |
| MC68HC908KH12 | AN1748/D | Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12 |
| MC68HC908KX8 | *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| MC68HC908MR24 | *AN1664/D | Low Cost 3-Phase AC Motor Control System Based on MC68HC908MR24 |
| | *AN1792/D | Using an MC68HC908MR24 in Place of an MC68HC708MP16 |
| MC68HC912B32 | AN1718/D | A Serial Bootloader for Reprogramming the MC68HC912B32 Flash EEPROM |
| | AN1774/D | Interfacing the MC68HC912B32 to an LCD Module |
| | *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| | EB183/D | Erasing and Programming the FLASH EEPROM on the MC68HC912B32 |
| MC68HSC705C8A | AN1734/D | Pulse Width Modulation Using the 16-Bit Timer |
| MC6805L3 | ANE405/D | Bi-Directional Data Transfer Between MC68HC11 and MC6805L3 Using SPI |
| MC6805R3 | *AN-HK-23/H | MC6805R3, MC68HC05SR3 Technical Comparison |
| MC13077 | AN492/D | A Video Display Board for CD-i Development |
| MC13145 | *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| | *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| MC13146 | *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| | *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| MC13156 | AN1539/D | An IF Communication Circuit Tutorial |
| MC14600 | *AN1690/D | Alarm IC General Applications Overview |
| MC33033 | AN1307/D | A Simple Pressure Regulator Using Semiconductor Pressure Transducers |

| | | |
|---|---|---|
| MC33073 | AN1536/D | Digital Boat Speedometers |
| MC33079 | AN1100/D | Analog to Digital Converter Resolution Extension Using a Motorola Pressure Sensor |
| MC33120 | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MC33161 | AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| MC33169 | AN1599/D | Power Control with the MRFIC0913 GaAs Integrated Power Amplifier and MC33169 Support… |
| | AN1602/D | 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard… |
| MC33179 | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| MC33272 | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1325/D | Amplifiers for Semiconductor Pressure Sensors |
| MC33274 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| | AN1325/D | Amplifiers for Semiconductor Pressure Sensors |
| MC33411 | *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| | *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| MC33790 | *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| MC34064 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| MC34160 | AN1597/D | Longwave Radio Data Decoding Using an HC11 and an MC3371 |
| MC44200 | AN492/D | A Video Display Board for CD-i Development |
| MC68000 | DCE406/D | Interface for MC68000 to DSP56001 Host Port |
| MC68030 | ANE426/D | An MC68030 32-bit High Performance Minimum System |
| MC68175 | APR34/D | MC68328 Microprocessor Application: FLEX Alphanumeric Chip MC68175 Interface for… |
| MC68230 | ANE426/D | An MC68030 32-bit High Performance Minimum System |
| MC68300 | EB259/D | Why MC68300 and MC68HC16 MCUs May Halt after Release of Reset |
| | EB260/D | Why an MC68300 or MC68HC16 Microcontroller May Fail to Release Reset |
| | EB261/D | Autovector Generation Using Chip Select Logic on MC68300 and MC68HC16 Devices |
| MC68302 | AR350/D | Adapt Non-ISDN Terminals to ISDN Data Rates |
| MC68306 | AN1264/D | JTAG Flash Memory Programmer |
| MC68307 | AN1264/D | JTAG Flash Memory Programmer |
| MC68328 | APR34/D | MC68328 Microprocessor Application: FLEX Alphanumeric Chip MC68175 Interface for… |
| MC68331 | AN473/D | A Minimum Evaluation System for the MC68331 and MC68332 |
| | AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| | EB273/D | Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors |
| MC68332 | AN473/D | A Minimum Evaluation System for the MC68331 and MC68332 |
| | AN1062/D | Using the QSPI for Analog Data Acquisition |
| | AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| | *EB267/D | The Double Bus Fault Monitor |
| | EB270/D | Problems with the PPWA Function on Revision P MC68332 Devices |
| | EB273/D | Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors |
| | *EB274/D | Generating Interrupts on the Time Processor Unit |
| | *EB276/D | Using the ITC Function on the Time Processor Unit A |
| | EB279/D | Low Output Levels on Output Pins |
| MC68336 | AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| MC68360 | DC414/D | An 8-bit EPROM Interface for an MC68EC040/MC68360 System |
| MC68376 | *AN1776/D | Stereo Audio Transmission over the CAN Bus Using the Motorola MC68376 with TouCAN… |
| MC68681 | ANE426/D | An MC68030 32-bit High Performance Minimum System |
| MC88100 | AN1125/D | DRAM Interface to the MC88200 M Bus |
| MC88110 | EB163/D | Running the MC88110 in Lockstep |
| | EB164/D | Interrupt Latency in the MC88110 |
| | EB165/D | Hardware Implications of xmem as a st followed by a ld |
| MC88200 | AN1125/D | DRAM Interface to the MC88200 M Bus |
| MC88914 | AN1125/D | DRAM Interface to the MC88200 M Bus |
| MC144115P | AN442/D | Driving LCDs with M6805 Microprocessors |
| MC144143 | AN1235/D | A Set Top Closed-Caption Decoder |
| MC145000 | AN442/D | Driving LCDs with M6805 Microprocessors |
| MC145003 | AN442/D | Driving LCDs with M6805 Microprocessors |
| MC145004 | AN442/D | Driving LCDs with M6805 Microprocessors |
| MC145040 | AN1062/D | Using the QSPI for Analog Data Acquisition |

| | | |
|---|---|---|
| MC145041 | AN1062/D | Using the QSPI for Analog Data Acquisition |
| MC145050 | AN1062/D | Using the QSPI for Analog Data Acquisition |
| MC145051 | AN1062/D | Using the QSPI for Analog Data Acquisition |
| | AN1228/D | Interfacing the HC05 MCU to the MC145051 A/D Converter |
| MC145157 | ANE416/D | MC68HC05B4 Radio Synthesizer |
| MC145170 | AN1671/D | MC145170 PSpice Modeling Kit |
| MC145407 | AN1240/D | HC05 MCU Software-Driven Asynchronous Serial Communication Techniques Using the… |
| | EB419/D | ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker |
| MC145422 | AN948/D | Data Multiplexing Using the Universal Digital Loop Transceiver and the Data Set… |
| MC145426 | AN948/D | Data Multiplexing Using the Universal Digital Loop Transceiver and the Data Set… |
| MC145428 | AN948/D | Data Multiplexing Using the Universal Digital Loop Transceiver and the Data Set… |
| MC145429 | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MC145453 | AN1326/D | Barometric Pressure Measurement Using Semiconductor Pressure Sensors |
| | AN1536/D | Digital Boat Speedometers |
| MC145474 | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MC145488 | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MC145490EVK | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MC145554 | AN1054/D | ISDN System Development Using MC145490EVK/MC145491EVK Development Kits |
| MCD210 | AN492/D | A Video Display Board for CD-i Development |
| MCD1460 | AN492/D | A Video Display Board for CD-i Development |
| MCF5307 | *AN1802/D | Connecting the MCF5307 to 168-pin Unbuffered SDRAM DIMMs |
| MCM63V736 | *AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm… |
| MCM63Z736 | AN1729/D | BurstRAM to ZBT RAM |
| | AN1773/D | ZBT Primer |
| MCM63Z737 | AN1773/D | ZBT Primer |
| MCM63Z818 | AN1773/D | ZBT Primer |
| MCM63Z819 | AN1773/D | ZBT Primer |
| MCM67B518 | AN1223/D | A Zero Wait State Secondary Cache for Intel's Pentium |
| MCM67B618 | AN1223/D | A Zero Wait State Secondary Cache for Intel's Pentium |
| MCM69C232 | AN1296/D | Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card |
| | AN1726/D | Using Motorola's Fast Static RAM CAMs on a Media Independent Interface |
| MCM69C432 | AN1296/D | Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card |
| | AN1726/D | Using Motorola's Fast Static RAM CAMs on a Media Independent Interface |
| MCM69D536 | AN1704/D | Switch Fabric Implementation Using Shared Memory |
| | AN1779/D | Using the MCM69D536/MCM69D618 NetRAMs with Different Speed Computing Elements |
| MCM69D618 | AN1704/D | Switch Fabric Implementation Using Shared Memory |
| | AN1707/D | Dual Port Memory for Multiprocessor Applications |
| | AN1779/D | Using the MCM69D536/MCM69D618 NetRAMs with Different Speed Computing Elements |
| MCM69F536 | AN1261/D | Use of 32K x 36 FSRAM in Non-Parity Applications |
| MCM69F536C | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MCM69F618C | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MCM69P536 | AN1261/D | Use of 32K x 36 FSRAM in Non-Parity Applications |
| MCM69P737 | AN1729/D | BurstRAM to ZBT RAM |
| MCM2814 | *AN4003/D | Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal… |
| MCM6164 | ANE426/D | An MC68030 32-bit High Performance Minimum System |
| MCM6287 | AR241/D | Building Fast SRAMs with no Process 'Tricks' |
| MCM6288 | AR241/D | Building Fast SRAMs with no Process 'Tricks' |
| MCM6292 | AR256/D | Motorola's Radical SRAM Design Speeds Systems 40% |
| | AR258/D | High Frequency System Operation Using Synchronous SRAMs |
| | AR260/D | Enhancing System Performance Using Synchronous SRAMs |
| MCM6293 | AR256/D | Motorola's Radical SRAM Design Speeds Systems 40% |
| | AR258/D | High Frequency System Operation Using Synchronous SRAMs |
| | AR260/D | Enhancing System Performance Using Synchronous SRAMs |
| MCM6294 | AR256/D | Motorola's Radical SRAM Design Speeds Systems 40% |
| | AR258/D | High Frequency System Operation Using Synchronous SRAMs |
| | AR260/D | Enhancing System Performance Using Synchronous SRAMs |

| | | |
|---|---|---|
| MCM6295 | AR256/D | Motorola's Radical SRAM Design Speeds Systems 40% |
| | AR258/D | High Frequency System Operation Using Synchronous SRAMs |
| | AR260/D | Enhancing System Performance Using Synchronous SRAMs |
| MCM54400 | APR405/D | Minimal Logic DRAM Interface for the DSP56156 |
| MCM62486 | AN1209/D | The Motorola BurstRAM |
| MCM91000 | AN1125/D | DRAM Interface to the MC88200 M Bus |
| MCM514256 | APR11/D | DSP56001 Interface Techniques and Examples |
| MEVB16 | EB252/D | MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers |
| MGP4N60ED | *AN1627/D | Low Cost High Efficiency Sensorless Drive for Brushless DC Motor Using MC68HC(7)05MC4 |
| MGP8N60ED | *AN1664/D | Low Cost 3-Phase AC Motor Control System Based on MC68HC908MR24 |
| MHW612 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW613 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW709 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW710 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW720 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW808 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW820 | EB107/D | Mounting Considerations for Motorola RF Power Modules |
| MHW910 | *AN1643/D | RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit |
| MHW1810 | *AN1643/D | RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit |
| MHW1910 | *AN1643/D | RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit |
| MHW10000 | AR333/D | RF Modems Simplified |
| MMA1000P | AN1632/D | MMA1000P Product Overview and Interface Considerations |
| | AN1640/D | Reducing Accelerometer Susceptibility to BCI |
| MMAS40G | AN1559/D | Application Considerations for a Switched Capacitor Accelerometer |
| MMAS40G10D | AN1612/D | Shock and Mute Pager Applications Using Accelerometer |
| | AN4004/D | ±2g Acceleration Sensing Module Based on a ±40g Integrated Accelerometer |
| MMAS40G10S | AN1611/D | Impact and Tilt Measurement Using Accelerometer |
| MMBT3904L | AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| MOC2A60 | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| MPBF375BGA | *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| MPC105 | AN1269/D | PowerPC Microprocessor Clock Modes |
| MPC106 | AN1265/D | Configuring the MPC2604GA Integrated L2 Cache with the MPC106 |
| | AN1269/D | PowerPC Microprocessor Clock Modes |
| | AN1722/D | SDRAM System Design Using the MPC106 |
| | AN1725/D | Initializing SDRAM Parameters for Motorola MPC106-Based Systems |
| | AN1727/D | Designing PCI 2.1-Compliant MPC106 Systems |
| | AN1768/D | Using Registered SDRAM DIMMs with the MPC106 |
| MPC505 | AN1281/D | MPC505 Interrupts |
| | AN1282/D | Board Strategies for Ensuring Optimum Frequency Synthesizer Performance |
| MPC509 | AN1282/D | Board Strategies for Ensuring Optimum Frequency Synthesizer Performance |
| MPC555 | *AN1778/D | Using the MIOS on the MPC555 Evaluation Board |
| MPC601 | AN486/D | Low Cost MPC601 EVM |
| | AN1271/D | PowerPC 60x Microprocessor to AD1848 CODEC Interface |
| | AN1272/D | Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption |
| | AN4000/D | Visual Debug for MPC60x |
| MPC602 | AN1269/D | PowerPC Microprocessor Clock Modes |
| MPC603 | AN1269/D | PowerPC Microprocessor Clock Modes |
| | AN1271/D | PowerPC 60x Microprocessor to AD1848 CODEC Interface |
| | AN1272/D | Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption |
| | AN4000/D | Visual Debug for MPC60x |
| | AR359/D | The Making of the PowerPC |
| MPC603e | AN1269/D | PowerPC Microprocessor Clock Modes |
| | AN1294/D | Multiprocessor Systems and the PowerPC 603e Microprocessor |
| | AN1769/D | Designing a Minimal PowerPC System |
| MPC603ev | AN1769/D | Designing a Minimal PowerPC System |
| MPC604 | AN1269/D | PowerPC Microprocessor Clock Modes |
| | AN1271/D | PowerPC 60x Microprocessor to AD1848 CODEC Interface |

| | | |
|---|---|---|
| MPC604 (continued) | AN1272/D | Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption |
| | AN1291/D | Avoiding Multiprocessing Paradoxes with the PowerPC 604 Microprocessor |
| | AN1769/D | Designing a Minimal PowerPC System |
| | AN4000/D | Visual Debug for MPC60x |
| MPC604e | AN1269/D | PowerPC Microprocessor Clock Modes |
| MPC620 | AR360/D | PowerPC 620 Soars |
| MPC740 | AN1769/D | Designing a Minimal PowerPC System |
| MPC750 | AN1769/D | Designing a Minimal PowerPC System |
| | *AN1794/D | PowerPC Backside L2 Timing Analysis for the PCB Design Engineer |
| | *AN1800/D | Programming the Thermal Assist Unit in the MPC750 Microprocessor |
| MPC755 | *AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm… |
| MPC801 | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MPC823 | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MPC850 | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MPC860 | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MPC2604GA | AN1265/D | Configuring the MPC2604GA Integrated L2 Cache with the MPC106 |
| MPC8260 | AN1777/D | MPC8xx to BurstRAM Interfacing |
| MPE603e | AN1769/D | Designing a Minimal PowerPC System |
| MPE603ev | AN1769/D | Designing a Minimal PowerPC System |
| MPE604 | AN1769/D | Designing a Minimal PowerPC System |
| MPFB1632 | *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| MPX10 | AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| | AN1556/D | Designing Sensor Performance Specifications for MCU-based Systems |
| | AN1557/D | A Cookbook Approach to Designing a Differential-Signal Amplifier for Sensor Applications |
| | AN1585/D | High-Performance, Dynamically-Compensated Smart Sensor System |
| | AN1651/D | ASB201 – Uncompensated Series Sensor Module |
| | AN1668/D | Washing Appliance Sensor Selection |
| MPX11 | AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| MPX12 | AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| | AN1668/D | Washing Appliance Sensor Selection |
| MPX50 | AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| MPX100 | AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| | AN1651/D | ASB201 – Uncompensated Series Sensor Module |
| MPX700 | AN1105/D | A Digital Pressure Gauge Using the Motorola MPX700 Series Differential Pressure… |
| MPX2000 | AN1097/D | Calibration-Free Pressure Sensor System |
| | AN1309/D | Compensated Sensor Bar Graph Pressure Gauge |
| | AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| | AN1325/D | Amplifiers for Semiconductor Pressure Sensors |
| | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| | AN1586/D | Designing a Homemade Digital Output for Analog Voltage Output Sensors |
| | AN1660/D | Compound Coefficient Pressure Sensor PSPICE Models |
| MPX2010 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1325/D | Amplifiers for Semiconductor Pressure Sensors |
| | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| | AN1551/D | Low-Pressure Sensing with the MPX2010 Pressure Sensor |
| | AN1556/D | Designing Sensor Performance Specifications for MCU-based Systems |
| | AN1557/D | A Cookbook Approach to Designing a Differential-Signal Amplifier for Sensor Applications |
| | AN1584/D | "Very Low Pressure" Smart Sensing Solution with Serial Communications Interface |
| | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| | AN1654/D | ASB210 – 10" H2O Sensor Module |
| | AN1668/D | Washing Appliance Sensor Selection |
| MPX2050 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| MPX2100 | AN1082/D | Simple Design for a 4-20mA Transmitter Interface Using a Motorola Pressure Sensor |
| | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |

| | | |
|---|---|---|
| MPX2100 (continued) | AN1316/D | Frequency Output Conversion for MPX2000 Series Pressure Sensors |
| | AN1318/D | Interfacing Semiconductor Pressure Sensors to Microcomputers |
| | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| | AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| MPX2100A | AN1326/D | Barometric Pressure Measurement Using Semiconductor Pressure Sensors |
| MPX2100DP | AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| MPX2200 | AN1100/D | Analog to Digital Converter Resolution Extension Using a Motorola Pressure Sensor |
| | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| | AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| | AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| MPX2700 | AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| | AN1324/D | A Simple Sensor Interface Amplifier |
| | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| MPX5000 | AN1660/D | Compound Coefficient Pressure Sensor PSPICE Models |
| | *AR601/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| MPX5006 | AN1646/D | Noise Considerations for Integrated Pressure Sensors |
| | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| | AN1668/D | Washing Appliance Sensor Selection |
| MPX5010 | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| MPX5050 | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| MPX5050GP | AN1571/D | Digital Blood Pressure Meter |
| MPX5100 | AN1304/D | Integrated Sensor Simplifies Bar Graph Pressure Gauge |
| | AN1305/D | An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a… |
| | AN1307/D | A Simple Pressure Regulator Using Semiconductor Pressure Transducers |
| | AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| | AN1518/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| MPX5700 | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| MPX5999 | AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| MPX7100 | AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| MRF28x | *AR624/D | Aluminum-Based Metallization Enhances Device Reliability |
| MRF154 | AR347/D | A Compact 1kW 2-50MHz Solid-State Linear Amplifier |
| MRF260 | EB90/D | Low-Cost VHF Amplifier Has Broadband Performance |
| MRF262 | EB90/D | Low-Cost VHF Amplifier Has Broadband Performance |
| MRF264 | EB93/D | 60 Watt VHF Amplifier Uses Splitting/Combining Techniques |
| MRF286 | AN1673/D | Solder Reflow Mounting Method for the MRF286 and Similar Packages |
| | *EB211/D | Thermal Management and Solder Mounting Method for the MRF286, 60 Watt Power Device… |
| MRF873 | AN1526/D | RF Power Device Impedances: Practical Considerations |
| MRF1027T1 | AN1675/D | A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low… |
| MRF1047T1 | AN1675/D | A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low… |
| | AN1676/D | A Cascade 2 Stage Low Noise Amplifier Using the MRF1047T1 Low Noise Transistor |
| MRF6522-10 | AN1670/D | 60 watts, GSM 900MHz, LDMOS Two-Stage Amplifier |
| MRFIC917 | AN1602/D | 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard… |
| MRFIC1502 | AN1610/D | Using Motorola's MRFIC1502 in Global Positioning System Receivers |
| MRFIC1806 | *AN1532/D | Using the Motorola MRFIC1806/1807 Dual Demonstration Board |
| MRFIC1807 | *AN1532/D | Using the Motorola MRFIC1806/1807 Dual Demonstration Board |
| MRFIC1817 | AN1602/D | 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard… |
| PAL16R6 | APR405/D | Minimal Logic DRAM Interface for the DSP56156 |
| PBGA | AN1231/D | Plastic Ball Grid Array (PBGA) |
| | AN1232/D | Thermal Performance of Plastic Ball Grid Array (PBGA) Packages for Next Generation… |
| PCF8573 | AN1066/D | Interfacing the MC68HC05C5 SIOP to an I2C Peripheral |

| | | |
|---|---|---|
| SCM68000 | *AN1703/D | Low Power on the SCM68000 (EC000 Core) |
| SPGMR08 | *AN-HK-32/H | In-Circuit Programming of FLASH Memory in the MC68HC908GP32 |
| TDA3048 | AN465/D | Secure Remote Control using the 68HC05K1 and the 68HC05P3 |
| TPV375 | AN1028/D | 35/50 Watt Broadband (160-240MHz) Push-Pull TV Amplifier Band III |
| TZA120 | AN1082/D | Simple Design for a 4-20mA Transmitter Interface Using a Motorola Pressure Sensor |
| X76F041 | AN1761/D | Interfacing the MC68HC705C8A to the X76F041 PASS SecureFlash |

# Applications Documents

## Literature Selector Guide

*This selector guide lists applications documents under subject and device-type headings. It also includes cross references to some of Motorola's other literature which may provide further relevant information.*

### A/D and D/A Conversion

| | |
|---|---|
| AN477/D | Simple A/D for MCUs without Built-In A/D Converters |
| AN1058/D | Reducing A/D Errors in Microcontroller Applications |
| AN1062/D | Using the QSPI for Analog Data Acquisition |
| AN1222/D | Arithmetic Waveform Synthesis with the HC05/08 MCUs |
| AN1228/D | Interfacing the HC05 MCU to the MC145051 A/D Converter |
| AN1256/D | Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the MC68HC705C8A and the MC68HC705J1A |
| AN1740/D | Applications Using the Analog Subsystem of MC68HC05JJ/JP Series Microcontrollers |
| AN1775/D | Expanding Digital Input with an A/D Converter |
| *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| *EB292/D | Initialization Considerations when Moving from the BUFFALO Monitor to a Standalone MC68HC11 |

***Additional information relevant to A/D and D/A Conversion may be found in the following Motorola documents:***

| | |
|---|---|
| ADCRM/AD | Analog-to-Digital Converter Reference Manual |
| QADCRM/AD | Queued Analog-to Digital Converter Reference Manual |

### Audio Amplifiers and Systems

| | |
|---|---|
| AN1292/D | Adding a Voice User Interface to M68HC05 Applications |
| AN1730/D | Digital Amplification of an Analog Signal Using the MC68HC705J1A |
| AN1764/D | DSP56300 Enhanced Synchronous Serial Interface (ESSI) Programming |
| APR36/D | Interfacing the DSP560xx/DSP563xx Families to the Crystal CS4226 Multichannel Codec |

***Additional information relevant to Audio Amplifiers and Systems may be found in the following Motorola documents:***

| | |
|---|---|
| * BR1763/D | 24-bit Digital Audio Solutions |
| * BR1771/D | Streamaster – Digital Imagination: Let Your Mind Flow |
| * BR1803/D | Digital Audio/DTV Solutions |
| * DSP56007UM/AD | DSP56007 24-bit Digital Signal Processor User's Manual |
| DSP56009UM/AD | DSP56009 User's Manual |
| * DSP56011UM/AD | DSP56011 24-bit Digital Signal Processor User's Manual |
| * DSP56309UM/D | DSP56309 24-bit Digital Signal Processor User's Manual |
| * DSP56364UM/D | DSP56364 User's Manual |
| SG185/D | Digital Audio Solutions |

### Automotive Applications

| | |
|---|---|
| AN464/D | Software Driver Routines for the Motorola MC68HC05 CAN Module |
| AN465/D | Secure Remote Control using the 68HC05K1 and the 68HC05P3 |
| AN476/D | CPU16 and the Configurable Timer Module (CTM) in Engine Control |
| AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| AN1257/D | Using the M68HC05 Family On-Chip Voltage Regulator |
| AN1259/D | System Design and Layout Techniques for Noise Reduction in MCU-Based Systems |
| AN1622/D | EMC Considerations for Automotive Sensors |
| AN1632/D | MMA1000P Product Overview and Interface Considerations |
| AN1640/D | Reducing Accelerometer Susceptibility to BCI |
| AN1645/D | Micromachined Electromechanical Sensors for Automotive Applications |
| AN1731/D | VPW J1850 Multiplexing and Motorola's Byte Data Link Controller (BDLC) Module |
| AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |

**Automotive Applications** continued

| | |
|---|---|
| *AN1776/D | Stereo Audio Transmission over the CAN Bus Using the Motorola MC68376 with TouCAN Module |
| *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| *AN1827/D | Programming and Erasing FLASH Memory on the MC68HC908AS60 |
| AN4004/D | ±2g Acceleration Sensing Module Based on a ±40g Integrated Accelerometer |
| *AR320/D | Electronics: a Driving Force in Automotive Technology |
| EB181/D | Frequently Asked Questions and Answers: M68HC05 Family MCAN Module |
| EB421/D | The Motorola MCAN Module |
| TPUPN14/D | Position-Synchronised Pulse Generator (PSP) |
| TPUPN15A/D | Period Measurement with Additional Transition Detection TPU Function (PMA) |
| TPUPN15B/D | Period Measurement with Missing Transition Detection TPU Function (PMM) |

***Additional information relevant to Automotive Applications may be found in the following Motorola documents:***

| | |
|---|---|
| BDLCRM/AD | Byte Data Link Controller Reference Manual |
| BR477/D | Smart Mover – Stepper Motors with Integrated Serial Bus Controller |
| BR484/D | 68302 |
| BR934/D | Sensing Solutions from Motorola – Sensors for the Automotive Industry |
| BR1424/D | Sensing the Needs of the Future – Automotive Sensor Solutions |
| BR1465/D | 8-bit Microcontrollers for Multiplex Wiring |
| * BR1517/D | Acceleration Sensors from Motorola |
| * BR1523/D | Motorola's New Integrated Automotive MAP/BAP Sensors Offer Top Performance at Half the Package Size |
| BR1704/D | 68HC08: High Performance, 8-bit Microcontrollers with CAN, J1850 and Flash Memory Options |
| * BR1772/D | Instrument Cluster Systems |
| * BR1773/D | Automotive Door Systems |
| * BR1775/D | Occupant Safety Solutions |
| * BR1776/D | Mechatronics for Automotive |
| BR1781/D | Occupant Safety Systems Solutions |
| * BR1792/D | Access and Remote Control Systems |
| BR3005/D | Intelligent Sensor Solutions |
| HC908AT32GRS/D | MC68HC908AT32 General Release Specification |
| * SG187/D | Automotive Selector Guide |

## Computer Systems

| | |
|---|---|
| AN1050/D | Designing for Electromagnetic Compatibility (EMC) with HCMOS Microcontrollers |
| AN1051/D | Transmission Line Effects in PCB Applications |
| AN1209/D | The Motorola BurstRAM |
| AN1210/D | A Protocol Specific Memory for Burstable Fast Cache Memory Applications |
| AN1223/D | A Zero Wait State Secondary Cache for Intel's Pentium |
| AN1265/D | Configuring the MPC2604GA Integrated L2 Cache with the MPC106 |
| AN1272/D | Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption |
| AN1282/D | Board Strategies for Ensuring Optimum Frequency Synthesizer Performance |
| AN1288/D | Programming the MC68HC(8)05K3's Personality EEPROM on the MMDS and MMEVS |
| AN1579/D | Understanding the Multivibrator Based Crystal Oscillator Circuit Used on the BiCMOS MPC Family of Clock Drivers |
| *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| AN1707/D | Dual Port Memory for Multiprocessor Applications |
| AN1723/D | Interfacing MC68HC05 Microcontrollers to the IBM AT Keyboard Interface |
| AN1748/D | Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12 |
| AN1752/D | Data Structures for 8-bit Microcontrollers |
| AN1757/D | Add a Unique Silicon Serial Number to the HC05 |
| *AN1765/D | Ethernet Configuration for a ColdFire Evaluation Board to Download Files from a Personal Computer |
| AN1769/D | Designing a Minimal PowerPC System |
| *AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm Application |
| *AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| APR10/D | DSP96002 Interface Techniques and Examples |
| *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |

***Additional information relevant to Computer Systems may be found in the following Motorola documents:***

| | |
|---|---|
| BR488/D | 68306 68307 68322 |
| BR1333/D | Timing Solutions |
| BR1427/D | PC Brochure |
| * BR1466/D | The New Clock Driver Generation |
| * BR1517/D | Acceleration Sensors from Motorola |
| BR1756/D | PCI Controller-less Modem Chip Set and Software |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| DL156/D | Fast Static RAM – Component and Module Data |
| EMDVPOC/D | Embedded Developer Pocket Guide |
| HC705JB2GRS/H | 68HC705JB2 General Release Specification |
| MPC821UM/AD | MPC821 PowerPC Portable Systems Microprocessor User's Manual |
| SG171/D | Fast Static RAM Division Product Update |

## Digital Signal Processing

| | |
|---|---|
| AN1051/D | Transmission Line Effects in PCB Applications |

| | |
|---|---|
| AN1213/D | 16-bit DSP Servo Control with the MC68HC16Z1 |
| AN1233/D | Using M68HC16 Digital Signal Processing to Build an Audio Frequency Analyzer |
| AN1289/D | DSP5630x FSRAM Module Interfacing |
| AN1751/D | DSP563xx Port A Programming |
| AN1764/D | DSP56300 Enhanced Synchronous Serial Interface (ESSI) Programming |
| AN1772/D | Efficient Compilation of Bit-Exact Applications for DSP563xx |
| AN1779/D | Using the MCM69D536/MCM69D618 NetRAMs with Different Speed Computing Elements |
| AN1780/D | DSP563xx HI32 as a PCI Agent |
| AN1781/D | Booting DSP563xx Devices Through the Serial Communication Interface (SCI) |
| AN1782/D | Converting DSP56303 Designs to DSP56307 Designs |
| *AN1788/D | DSP563xx HI32 PCI Functions |
| *AN1790/D | Programming the CS4218 CODEC for Use with DSP56300 Devices |
| *AN1804/D | DSP56307 Port A-to-HI08 Interface |
| *AN1808/D | DSP56300 HI08 Host Port Programming |
| *AN1810/D | Developing DSP56364 Software Using the DSP56362 EVM |
| *AN1829/D | Software Differences Between the DSP56002 and the DSP56303 |
| *AN1830/D | Hardware Differences Between the DSP56002 and the DSP56303 |
| APR1/D | Digital Sine-Wave Synthesis Using the DSP56001/DSP56002 |
| APR2/D | Digital Stereo 10-Band Graphic Equalizer Using the DSP56001 |
| APR3/D | Fractional and Integer Arithmetic Using the DSP56000 Family of General-Purpose Digital Signal Processors |
| APR4/D | Implementation of Fast Fourier Transforms on Motorola's DSP56000/DSP56001 and DSP96002 Digital Signal Processors |
| APR5/D | Implementation of PID Controllers on the Motorola DSP56000/DSP56001 |
| APR6/D | Convolutional Encoding and Viterbi Decoding Using the DSP56001 with a V.32 Modem Trellis Example |
| APR7/D | Implementing IIR/FIR Filters with Motorola's DSP56000/DSP56001 |
| APR8/D | Principles of Sigma-Delta Modulation for Analog-to-Digital Converters |
| APR9/D | Full-Duplex 32 kbit/s CCITT ADPCM Speech Coding on the Motorola DSP56001 |
| APR10/D | DSP96002 Interface Techniques and Examples |
| APR11/D | DSP56001 Interface Techniques and Examples |
| APR12/D | Twin CODEC Expansion Board for the DSP56000 Application Development System |
| APR14/D | Conference Bridging in the Digital Telecomms Environment Using the Motorola DSP56000 |
| APR15/D | Implementation of Adaptive Controllers on the Motorola DSP56000/DSP56001 |
| APR16/D | Calculating Timing Requirements of External SRAM for the 24-bit DSP56000 Family |
| APR20/D | Application Optimization for the DSP56300/ DSP56600 Digital Signal Processors |
| APR21/D | Software UART on the DSP56L811 Using GPIO Port B |
| APR22/D | Application Conversion from the DSP56100 Family to the DSP56300/600 Families |
| *APR25/D | Interfacing Fast SRAM to Motorola's DSP56300 Family of Digital Signal Processors |
| *APR26/D | Interfacing Flash Memory with the Motorola DSP56300 Family of Digital Signal Processors |
| *APR27/D | Interfacing EPROM and EEPROM Memory with Motorola's DSP56300 Family of Digital Signal Processors |
| APR30/D | DSP56300 Assembly Code Development Using the Motorola Toolsets |
| APR31/D | Booting and Simple Usage of the DSP56004/007/ 009 SHI Port in SPI Mode |
| APR33/D | ROM Software Patching on the Motorola DSP56304 |
| APR34/D | MC68328 Microprocessor Application: FLEX Alphanumeric Chip MC68175 Interface for One-Way Pager |
| APR35/D | Designing Motorola DSP56xxx Software for Nonrealtime Tests File I/O Using SIM56xxx and ADS56xxx |
| APR36/D | Interfacing the DSP560xx/DSP563xx Families to the Crystal CS4226 Multichannel Codec |
| APR37/D | Implementing AC-link with ESAI |
| APR38/D | Interfacing Serial EEPROM to DSP563xx |
| APR39/D | Programming the DSP56307 Enhanced Filter Coprocessor (EFCOP) |
| APR40/D | Implementing Viterbi Decoder Using the VSL Instruction on DSP Families DSP56300 and DSP56600 |
| APR404/D | G.722 Audio Processing on the DSP56100 Microprocessor Family |
| APR405/D | Minimal Logic DRAM Interface for the DSP56156 |
| *AR366/D | Motorola Cellular DSP Does It All |
| DCE406/D | Interface for MC68000 to DSP56001 Host Port |
| EB420/D | Converting DSP56001-Based Designs to the DSP56002 |

***Additional information relevant to Digital Signal Processing may be found in the following Motorola documents:***

| | |
|---|---|
| BR1133/D | 68K and ColdFire Family Product Portfolio Overview |
| * BR1475/D | Datacomms |
| * BR1502/D | Wireless Infrastructure Solutions |
| * BR1763/D | 24-bit Digital Audio Solutions |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| * BR1789/D | Got DSP? |
| * BR1803/D | Digital Audio/DTV Solutions |
| BR3006/D | Wireless Communications Resource Guide |
| DL156/D | Fast Static RAM – Component and Module Data |
| DSP002EVMSG/D | DSP56002EVM – Test Drive the Future |
| DSP56KFAMUM/AD | DSP56000 Digital Signal Processor Family Manual |
| DSP56L811EMUM/AD | DSP56L811 Evaluation Module User's Manual |

**Digital Signal Processing** continued

| | |
|---|---|
| DSP56L811UM/AD | DSP56L811 User's Manual |
| * DSP56LF812UM/AD | DSP56LF812 16-bit Digital Signal Processor User's Manual |
| DSP56002PIX/D | Motorola's DSP56002 24-bit General Purpose Digital Signal Processor |
| DSP56002UM/AD | DSP56002 Digital Signal Processor User's Manual |
| DSP56004PIX/D | Motorola's DSP56004 24-bit Digital Signal Processor |
| DSP56004UM/AD | DSP56004 Digital Signal Processor User's Manual |
| DSP56007PIX/D | Motorola's DSP56007 24-bit Digital Processor |
| * DSP56007UM/AD | DSP56007 24-bit Digital Signal Processor User's Manual |
| DSP56009UM/AD | DSP56009 User's Manual |
| * DSP56011UM/AD | DSP56011 24-bit Digital Signal Processor User's Manual |
| DSP56100FM/AD | DSP56100 Digital Signal Processor Family Manual |
| DSP56300FM/AD | DSP56300 24-Bit Digital Signal Processor Family Manual |
| DSP56301UM/AD | DSP56301 24-Bit Digital Signal Processor User's Manual |
| DSP56302EMUM/AD | DSP56302 Evaluation Module User's Manual |
| DSP56302UM/AD | DSP56302 User's Manual |
| DSP56303EMUM/AD | DSP56303 Evaluation Module User's Manual |
| DSP56303UM/AD | DSP56303 User's Manual |
| DSP56304UM/AD | DSP56304 User's Manual |
| * DSP56307UM/D | DSP56307 24-bit Digital Signal Processor User's Manual |
| * DSP56309UM/D | DSP56309 24-bit Digital Signal Processor User's Manual |
| * DSP56311UM/D | DSP56311 24-bit Digital Signal Processor User's Manual |
| * DSP56364UM/D | DSP56364 User's Manual |
| * DSP56600FM/AD | DSP56600 16-bit Digital Signal Processor Family Manual |
| * DSP56602UM/AD | DSP56602 16-bit Digital Signal Processor User's Manual |
| DSP56603EMUM/AD | DSP56603 Evaluation Module User's Manual |
| * DSP56603UM/AD | DSP56603 16-bit Digital Signal Processor User's Manual |
| DSP56800FM/AD | DSP56800 Family Manual |
| DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |
| * DSP56824UM/AD | DSP56824 16-bit Digital Signal Processor User's Manual |
| MC68356UM/AD | MC68356 Signal Processing Communications Engine User's Manual |
| SG146/D | Digital Signal Processors Update |
| SG171/D | Fast Static RAM Division Product Update |
| SG185/D | Digital Audio Solutions |

## Instrumentation and Control

| | |
|---|---|
| AN477/D | Simple A/D for MCUs without Built-In A/D Converters |
| AN1050/D | Designing for Electromagnetic Compatibility (EMC) with HCMOS Microcontrollers |
| AN1058/D | Reducing A/D Errors in Microcontroller Applications |
| AN1067/D | Pulse Generation and Detection with Microcontroller Units |
| AN1215/D | PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers |
| AN1239/D | HC05 MCU Keypad Decoding Techniques Using the MC68HC705J1A |
| AN1241/D | Interfacing the MC68HC705J1A to 9356/9366 EEPROMs |
| AN1259/D | System Design and Layout Techniques for Noise Reduction in MCU-Based Systems |
| AN1292/D | Adding a Voice User Interface to M68HC05 Applications |
| AN1304/D | Integrated Sensor Simplifies Bar Graph Pressure Gauge |
| AN1305/D | An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a Microprocessor |
| AN1316/D | Frequency Output Conversion for MPX2000 Series Pressure Sensors |
| AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| AN1518/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| AN1525/D | The A-B-Cs of Signal-Conditioning Amplifier Design for Sensor Applications |
| * AN1639/D | Phase Noise Measurement Using the Phase Lock Technique |
| * AN1690/D | Alarm IC General Applications Overview |
| AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| AN1759/D | Add a Non-Volatile Clock to the MC68HC705J1A |
| AN1760/D | Interfacing the AD8402 Digital Potentiometer to the MC68HC705J1A |
| APR15/D | Implementation of Adaptive Controllers on the Motorola DSP56000/DSP56001 |
| AR511/D | Biasing Solid State Amplifiers to Linear Operation |
| AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| * AR601/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| * AR602/D | Home-Brewed Circuits Tailor Sensor Outputs to Specialized Needs |

***Additional information relevant to Instrumentation and Control may be found in the following Motorola documents:***

| | |
|---|---|
| BR484/D | 68302 |
| BR489/D | 68360 Quad Integrated Communications Controller (QUICC) |
| BR1188/D | LonWorks Networks for Industrial and Process Control |
| * BR1519/D | Medical Sensing from Motorola |
| BR1704/D | 68HC08: High Performance, 8-bit Microcontrollers with CAN, J1850 and Flash Memory Options |
| BR3005/D | Intelligent Sensor Solutions |

## Interfacing

**see also Telecommunications**

| | |
|---|---|
| AN442/D | Driving LCDs with M6805 Microprocessors |
| AN991/D | Using the Serial Peripheral Interface to Communicate Between Multiple Microcomputers |
| AN1066/D | Interfacing the MC68HC05C5 SIOP to an $I^2C$ Peripheral |
| AN1082/D | Simple Design for a 4-20mA Transmitter Interface Using a Motorola Pressure Sensor |
| AN1239/D | HC05 MCU Keypad Decoding Techniques Using the MC68HC705J1A |
| AN1240/D | HC05 MCU Software-Driven Asynchronous Serial Communication Techniques Using the MC68HC705J1A |
| AN1256/D | Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the MC68HC705C8A and the MC68HC705J1A |
| AN1292/D | Adding a Voice User Interface to M68HC05 Applications |
| AN1316/D | Frequency Output Conversion for MPX2000 Series Pressure Sensors |
| AN1667/D | Software SCI Implementation to the MISC Communication Protocol |
| *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| AN1723/D | Interfacing MC68HC05 Microcontrollers to the IBM AT Keyboard Interface |
| AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| AN1725/D | Initializing SDRAM Parameters for Motorola MPC106-Based Systems |
| AN1727/D | Designing PCI 2.1-Compliant MPC106 Systems |
| AN1732/D | A Universal Serial Bus Gamepad Device using the MC68HC05JB2 |
| AN1748/D | Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12 |
| AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| AN1758/D | Add Addressable Switches to the HC05 |
| AN1760/D | Interfacing the AD8402 Digital Potentiometer to the MC68HC705J1A |
| AN1774/D | Interfacing the MC68HC912B32 to an LCD Module |
| AN1775/D | Expanding Digital Input with an A/D Converter |
| *AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm Application |
| *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| *AN1818/D | Software SCI Routines with the 16-bit Timer Module |
| *AN1820/D | Software $I^2C$ Communications |
| *AN-HK-24/H | Software Emulation of DDC1 Hardware using HC05BD3 |
| ANE405/D | Bi-Directional Data Transfer Between MC68HC11 and MC6805L3 Using SPI |
| ANE415/D | MC68HC11 Implementation of IEEE-488 Interface for DSP56000 Monitor |
| APR21/D | Software UART on the DSP56L811 Using GPIO Port B |
| *AR601/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| EB281/D | Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers |
| EB406/D | Getting Started with the FDDI ADS Board |
| EB421/D | The Motorola MCAN Module |
| TPUPN07/D | Asynchronous Serial Interface TPU Function (UART) |

***Additional information relevant to Interfacing may be found in the following Motorola documents:***

| | |
|---|---|
| BR477/D | Smart Mover – Stepper Motors with Integrated Serial Bus Controller |
| BR1133/D | 68K and ColdFire Family Product Portfolio Overview |
| BR3020/D | Remote Access: ISDN Solutions Kit |
| MC68EN302RM/AD | MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual (Supplement to MC68302UM/AD) |
| *MC68HC681UM/AD | MC68HC681 Dual Asynchronous Receiver/ Transmitter (DUART) |
| MC68360UM/AD | MC68360 Quad Integrated Communications Controller User's Manual |
| QMCSUPPLEMENT/AD | MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals |
| *SG187/D | Automotive Selector Guide |

## Low Power/Battery Applications

| | |
|---|---|
| *AN1690/D | Alarm IC General Applications Overview |
| AN1762/D | Automatic Contrast Control of LCD Displays Using the MC68HC708LN56 Microcontroller |
| AN-HK-13A/H | MC68HC05L10, an Enhanced Version of L9 for Handheld Equipment Applications |
| *EB198/D | Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11 |
| EB414/D | Low Power Write Enable Generation for M68300 Family Microprocessors |

***Additional information relevant to Low Power/Battery Applications may be found in the following Motorola documents:***

| | |
|---|---|
| *BR1516/D | Alarm Integrated Circuits from Motorola |
| *BR1520/D | Smoke Detector ICs from Motorola |
| MCORERM/AD | M•CORE Reference Manual |
| MCORESALES/D | M•CORE Architecture |
| MC68LC302RM/AD | MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual |
| MC68PM302RM/AD | Integrated Multiprotocol Processor with PCMCIA Interface Reference Manual |
| MMC2001RM/D | M•CORE MMC2001 Reference Manual |
| *MPC801UM/AD | MPC801 Integrated Microprocessor for Embedded Systems User's Manual |

**Low Power/Battery Applications** continued

| | |
|---|---|
| MPC821UM/AD | MPC821 PowerPC Portable Systems Microprocessor User's Manual |
| MPC823UM/D | PowerPC MPC823 User's Manual |

## Memory

| | |
|---|---|
| AN432/D | 128K byte Addressing with the M68HC11 |
| AN1051/D | Transmission Line Effects in PCB Applications |
| AN1125/D | DRAM Interface to the MC88200 M Bus |
| AN1209/D | The Motorola BurstRAM |
| AN1210/D | A Protocol Specific Memory for Burstable Fast Cache Memory Applications |
| AN1223/D | A Zero Wait State Secondary Cache for Intel's Pentium |
| AN1227/D | Using 9346 Series Serial EEPROMs with 6805 Series Microcontrollers |
| AN1231/D | Plastic Ball Grid Array (PBGA) |
| AN1232/D | Thermal Performance of Plastic Ball Grid Array (PBGA) Packages for Next Generation FSRAM Devices |
| AN1241/D | Interfacing the MC68HC705J1A to 9356/9366 EEPROMs |
| *AN1242/D | High Performance M68HC11 System Design Using the WSI PSD4xx and PSD5xx Families |
| AN1243/D | Output Loading Effects on Fast Static RAMS |
| AN1255/D | MC68F333 Flash EEPROM Programming Utilities |
| AN1261/D | Use of 32K x 36 FSRAM in Non-Parity Applications |
| AN1265/D | Configuring the MPC2604GA Integrated L2 Cache with the MPC106 |
| AN1289/D | DSP5630x FSRAM Module Interfacing |
| AN1296/D | Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card |
| AN1299/D | ATM Switch with Shared Memory – A Simple Model |
| AN1704/D | Switch Fabric Implementation Using Shared Memory |
| AN1707/D | Dual Port Memory for Multiprocessor Applications |
| AN1718/D | A Serial Bootloader for Reprogramming the MC68HC912B32 Flash EEPROM |
| AN1722/D | SDRAM System Design Using the MPC106 |
| AN1726/D | Using Motorola's Fast Static RAM CAMs on a Media Independent Interface |
| AN1729/D | BurstRAM to ZBT RAM |
| AN1751/D | DSP563xx Port A Programming |
| AN1753/D | Implementing a FLASH Memory System in an MC68HC711E9 Design |
| AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| AN1761/D | Interfacing the MC68HC705C8A to the X76F041 PASS SecureFlash |
| AN1768/D | Using Registered SDRAM DIMMs with the MPC106 |
| AN1770/D | In-Circuit Programming of FLASH Memory in the MC68HC908GP20 |
| AN1773/D | ZBT Primer |
| AN1777/D | MPC8xx to BurstRAM Interfacing |
| AN1779/D | Using the MCM69D536/MCM69D618 NetRAMs with Different Speed Computing Elements |
| *AN1802/D | Connecting the MCF5307 to 168-pin Unbuffered SDRAM DIMMs |
| *AN1827/D | Programming and Erasing FLASH Memory on the MC68HC908AS60 |
| *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| *AN4003/D | Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal Evaluation Board |
| *AN-HK-33/H | In-Circuit Programming of FLASH Memory in the MC68HC908JL3 |
| APR11/D | DSP56001 Interface Techniques and Examples |
| *APR25/D | Interfacing Fast SRAM to Motorola's DSP56300 Family of Digital Signal Processors |
| *APR26/D | Interfacing Flash Memory with the Motorola DSP56300 Family of Digital Signal Processors |
| *APR27/D | Interfacing EPROM and EEPROM Memory with Motorola's DSP56300 Family of Digital Signal Processors |
| APR38/D | Interfacing Serial EEPROM to DSP563xx |
| APR405/D | Minimal Logic DRAM Interface for the DSP56156 |
| AR241/D | Building Fast SRAMs with no Process 'Tricks' |
| AR256/D | Motorola's Radical SRAM Design Speeds Systems 40% |
| AR258/D | High Frequency System Operation Using Synchronous SRAMs |
| AR260/D | Enhancing System Performance Using Synchronous SRAMs |
| *EB182/D | How the ROMON Bit Behaves on the E Series HC11 MCUs |
| *EB185/D | Simplify MC68HC711E9 EPROM Programming with PCbug11 and the M68HC711EPGMR Board |
| *EB254/D | Setting the Programming Voltage of Modular Microcontrollers with FLASH EEPROM |

***Additional information relevant to Memory may be found in the following Motorola documents:***

| | |
|---|---|
| BR1150/D | 7 x 17 PBGA Sample Preview |
| *BR1502/D | Wireless Infrastructure Solutions |
| BR1702/D | Fast Static RAMS and The Communications Market |
| BR1716/D | Motorola Fast SRAMs: World Class Solutions |
| *BR1782/D | Networking and Computing Solutions from Motorola |
| DL156/D | Fast Static RAM – Component and Module Data |
| MC88200UM/AD | MC88200 Cache/Memory Management Unit User's Manual |
| SG171/D | Fast Static RAM Division Product Update |

## Microprocessors

### 8-bit MPU/MCU

AN427/D MC68HC11 EEPROM Error Correction Algorithms in C

AN432/D 128K byte Addressing with the M68HC11

AN442/D Driving LCDs with M6805 Microprocessors

AN464/D Software Driver Routines for the Motorola MC68HC05 CAN Module

AN465/D Secure Remote Control using the 68HC05K1 and the 68HC05P3

AN477/D Simple A/D for MCUs without Built-In A/D Converters

AN495/D RDS Decoding for an HC11-Controlled Radio

AN499/D Let the MC68HC705 Program Itself

AN906A/D Self-Programming the MC68701 and the MC68701U4

AN974/D MC68HC11 Floating-Point Package

AN991/D Using the Serial Peripheral Interface to Communicate Between Multiple Microcomputers

AN997/D CONFIG Register Issues Concerning the M68HC11 Family

AN1010/D MC68HC11 EEPROM Programming from a Personal Computer

AN1050/D Designing for Electromagnetic Compatibility (EMC) with HCMOS Microcontrollers

AN1057/D Selecting the Right Microcontroller Unit

AN1058/D Reducing A/D Errors in Microcontroller Applications

AN1060/D MC68HC11 Bootstrap Mode

AN1064/D Use of Stack Simplifies M68HC11 Programming

AN1066/D Interfacing the MC68HC05C5 SIOP to an $I^2C$ Peripheral

AN1067/D Pulse Generation and Detection with Microcontroller Units

AN1097/D Calibration-Free Pressure Sensor System

AN1215/D PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers

AN1218/D HC05 to HC08 Optimization

AN1219/D M68HC08 Integer Math Routines

AN1222/D Arithmetic Waveform Synthesis with the HC05/08 MCUs

AN1224/D Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8

AN1226/D Use of the 68HC705C8A in Place of a 68HC705C8

AN1227/D Using 9346 Series Serial EEPROMs with 6805 Series Microcontrollers

AN1228/D Interfacing the HC05 MCU to the MC145051 A/D Converter

*AN1237/D Using M68HC11 Microcontrollers with WSI Programmable Peripheral Devices

AN1238/D HC05 MCU LED Drive Techniques Using the MC68HC705J1A

AN1239/D HC05 MCU Keypad Decoding Techniques Using the MC68HC705J1A

AN1240/D HC05 MCU Software-Driven Asynchronous Serial Communication Techniques Using the MC68HC705J1A

AN1241/D Interfacing the MC68HC705J1A to 9356/9366 EEPROMs

*AN1242/D High Performance M68HC11 System Design Using the WSI PSD4xx and PSD5xx Families

AN1256/D Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the MC68HC705C8A and the MC68HC705J1A

AN1257/D Using the M68HC05 Family On-Chip Voltage Regulator

AN1259/D System Design and Layout Techniques for Noise Reduction in MCU-Based Systems

AN1262/D Simple Real-Time Kernels for M68HC05 Microcontrollers

AN1263/D Designing for Electromagnetic Compatibility with Single-Chip Microcontrollers

AN1274/D HC08 SCI Operation with Various Input Clocks

AN1283/D Transporting M68HC11 Code to M68HC16 Devices

AN1284/D Transporting M68HC11 Code to M68HC12 Devices

AN1285/D Stepper Motor Control with an MC68HC11E9 Microcontroller

AN1286/D MC68HC05C0 Bus Structure Design

AN1287/D MC68HC708LN56 LCD Utilities

AN1288/D Programming the MC68HC(8)05K3's Personality EEPROM on the MMDS and MMEVS

AN1292/D Adding a Voice User Interface to M68HC05 Applications

AN1298/D Variations in the Motorola MC68HC(7)05Cx Family

AN1305/D An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a Microprocessor

AN1315/D An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor

AN1316/D Frequency Output Conversion for MPX2000 Series Pressure Sensors

AN1322/D Applying Semiconductor Sensors to Bar Graph Pressure Gauges

AN1326/D Barometric Pressure Measurement Using Semiconductor Pressure Sensors

AN1518/D Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors

AN1536/D Digital Boat Speedometers

AN1551/D Low-Pressure Sensing with the MPX2010 Pressure Sensor

AN1571/D Digital Blood Pressure Meter

AN1584/D "Very Low Pressure" Smart Sensing Solution with Serial Communications Interface

AN1585/D High-Performance, Dynamically-Compensated Smart Sensor System

AN1586/D Designing a Homemade Digital Output for Analog Voltage Output Sensors

AN1597/D Longwave Radio Data Decoding Using an HC11 and an MC3371

**Microprocessors: 8-bit MPU/MCU** continued

| | |
|---|---|
| AN1611/D | Impact and Tilt Measurement Using Accelerometer |
| AN1612/D | Shock and Mute Pager Applications Using Accelerometer |
| *AN1627/D | Low Cost High Efficiency Sensorless Drive for Brushless DC Motor Using MC68HC(7)05MC4 |
| AN1638/D | Offset Calibration of Gauge Pressure Sensor Using Parallel I/O Ports |
| AN1655/D | ASB200 – Motorola Sensor Development Controller Board |
| *AN1663/D | Low Cost Universal Motor Sensorless Phase Angle Drive System |
| *AN1664/D | Low Cost 3-Phase AC Motor Control System Based on MC68HC908MR24 |
| AN1667/D | Software SCI Implementation to the MISC Communication Protocol |
| *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| AN1702/D | Brushless DC Motor Control Using the MC68HC705MC4 |
| AN1705/D | Noise Reduction Techniques for Microcontroller-Based Systems |
| AN1706/D | Microcontroller Oscillator Circuit Design Considerations |
| AN1711/D | DMA08 Systems Compatibilities |
| AN1712/D | “Get Your Motor Running” with the MC68HC708MP16 |
| AN1716/D | Using M68HC12 Indexed Indirect Addressing |
| *AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| AN1723/D | Interfacing MC68HC05 Microcontrollers to the IBM AT Keyboard Interface |
| AN1728/D | Making Low-Distortion Waveforms with the MC68HC708MP16 |
| AN1730/D | Digital Amplification of an Analog Signal Using the MC68HC705J1A |
| AN1731/D | VPW J1850 Multiplexing and Motorola’s Byte Data Link Controller (BDLC) Module |
| AN1732/D | A Universal Serial Bus Gamepad Device using the MC68HC05JB2 |
| AN1733/D | Implementing Caller ID Functionality in MC68HC(7)05 Applications |
| AN1734/D | Pulse Width Modulation Using the 16-Bit Timer |
| AN1736/D | Variations in the Motorola MC68HC05Px Family |
| AN1737/D | Migrating from the MC68HC705J2 to the MC68HC705JJ7 |
| AN1738/D | Instruction Cycle Timing of MC68HC05JJ/JP Series Microcontrollers |
| AN1740/D | Applications Using the Analog Subsystem of MC68HC05JJ/JP Series Microcontrollers |
| AN1741/D | In-Circuit and Emulation Considerations for MC68HC05JJ/JP Series Microcontrollers |
| AN1742/D | Programming the 68HC705J1A In-Circuit |
| AN1743/D | Scrolling Message Software |
| AN1744/D | Resetting Microcontrollers During Power Transitions |
| AN1745/D | Interfacing the HC705C8A to an LCD Module |
| *AN1746/D | Migrating from the MC68HC705K1 to the MC58HC705KJ1 |
| AN1747/D | Migrating from the MC68HC705K1 to the MC68HC805K3 |
| AN1748/D | Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12 |
| AN1752/D | Data Structures for 8-bit Microcontrollers |
| AN1753/D | Implementing a FLASH Memory System in an MC68HC711E9 Design |
| AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| AN1757/D | Add a Unique Silicon Serial Number to the HC05 |
| AN1758/D | Add Addressable Switches to the HC05 |
| AN1759/D | Add a Non-Volatile Clock to the MC68HC705J1A |
| AN1760/D | Interfacing the AD8402 Digital Potentiometer to the MC68HC705J1A |
| AN1761/D | Interfacing the MC68HC705C8A to the X76F041 PASS SecureFlash |
| AN1762/D | Automatic Contrast Control of LCD Displays Using the MC68HC708LN56 Microcontroller |
| AN1763/D | Driving LCD Displays Using the MC68HC705L16 Microcontroller |
| AN1770/D | In-Circuit Programming of FLASH Memory in the MC68HC908GP20 |
| AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| AN1775/D | Expanding Digital Input with an A/D Converter |
| AN1783/D | Determining MCU Oscillator Start-up Parameters |
| *AN1792/D | Using an MC68HC908MR24 in Place of an MC68HC708MP16 |
| *AN1818/D | Software SCI Routines with the 16-bit Timer Module |
| *AN1820/D | Software $I^2C$ Communications |
| *AN1827/D | Programming and Erasing FLASH Memory on the MC68HC908AS60 |
| *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| *AN4003/D | Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal Evaluation Board |
| AN-HK-10/H | MC68HC05L9 Microcomputer Applications Demo Board |
| AN-HK-12/H | MC68HC05F6 Tone Pulse Dialer |
| AN-HK-13A/H | MC68HC05L10, an Enhanced Version of L9 for Handheld Equipment Applications |
| AN-HK-15/H | MC68HC05L11 Hand-Writing Applications |
| AN-HK-17/H | MC68HC05F2 DTMF Output Low Voltage Active Filter |
| *AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| *AN-HK-23/H | MC6805R3, MC68HC05SR3 Technical Comparison |

| | |
|---|---|
| *AN-HK-24/H | Software Emulation of DDC1 Hardware using HC05BD3 |
| *AN-HK-31/H | Using the MC68HC908GP32 in Place of MC68HC908GP20 |
| *AN-HK-32/H | In-Circuit Programming of FLASH Memory in the MC68HC908GP32 |
| *AN-HK-33/H | In-Circuit Programming of FLASH Memory in the MC68HC908JL3 |
| ANE405/D | Bi-Directional Data Transfer Between MC68HC11 and MC6805L3 Using SPI |
| ANE416/D | MC68HC05B4 Radio Synthesizer |
| ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |
| APR31/D | Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode |
| EB166/D | System Design Considerations: Converting from the MC68HC805B6 to the MC68HC705B16 Microcontroller |
| EB180/D | Differences between the MC68HC705B16 and the MC68HC705B16N |
| EB181/D | Frequently Asked Questions and Answers: M68HC05 Family MCAN Module |
| *EB182/D | How the ROMON Bit Behaves on the E Series HC11 MCUs |
| *EB184/D | Enabling the Security Feature on the MC68HC711E9 Devices with PCbug11 on the MC68HC711E9PGMR |
| *EB185/D | Simplify MC68HC711E9 EPROM Programming with PCbug11 and the M68HC711EPGMR Board |
| *EB187/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVB |
| *EB188/D | Enabling the Security Feature on M68HC811E2 Devices with PCbug11 on the M68HC711E9PGMR |
| *EB189/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC711E9PGMR |
| EB191/D | Programming EPROM and EEPROM on the M68HC11EVM |
| EB192/D | A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers |
| *EB193/D | Replacing 68HC11A Series MCUs with 68HC11E Series MCUs |
| EB195/D | How to Configure the Reset Pin on the MC68HC11 |
| EB197/D | Using Pseudo-Interrupt Vectors on the M68HC11EVBU |
| *EB198/D | Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11 |
| EB262/D | DSACK Generation on the System Integration and Single-Chip Integration |
| EB283/D | C Macro Definitions for the MC68HC11C0 |
| EB284/D | C Macro Definitions for the MC68HC(7)11D3/D0 |
| EB285/D | C Macro Definitions for the MC68HC(7)11E20 |
| EB286/D | C Macro Definitions for the MC68HC11A8/A7/A1/A0 |
| EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| EB288/D | C Macro Definitions for the MC68HC11ED0 |
| EB289/D | C Macro Definitions for the MC68HC11F1 |
| *EB291/D | Programming MC68HC811E2 Devices with PCbug11 and the M68HC11EVBU |
| *EB292/D | Initialization Considerations when Moving from the BUFFALO Monitor to a Standalone MC68HC11 |
| *EB293/D | Simplify MC68HC711E20 EPROM Programming with PCbug11 |
| EB294/D | How to Write the 64-Cycle Time-Protected Registers on M68HC11 Development Tools |
| *EB295/D | Programming the EEPROM on the MC68HC811E2 with the M68HC11EVM Board |
| *EB296/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVBU |
| EB298/D | Programming the BUFFALO Monitor into an MC68HC711E9 |
| *EB299/D | Why M68HC711D3PGMR Software Does Not Run on 486 33MHz Computers |
| EB301/D | Programming EEPROM on the MC68HC811E2 During Program Execution |
| *EB303/D | Handling Considerations for Avoiding Intermittent Programming and Execution Failures with MC68HC11-Windowed Devices |
| *EB312/D | Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs |
| EB410/D | PASM05 to INTROL M68HC05 Assembler Conversion |
| EB413/D | Resetting MCUs |
| EB416/D | Modular Target Cables for Motorola Development Systems |
| EB419/D | ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker |
| EB421/D | The Motorola MCAN Module |
| EB422/D | Enhanced M68HC11 Bootstrap Mode |

***Additional information relevant to 8-bit MPU/MCU may be found in the following Motorola documents:***

| | |
|---|---|
| ADCRM/AD | Analog-to-Digital Converter Reference Manual |
| BDLCRM/AD | Byte Data Link Controller Reference Manual |
| BR266/D | M68HC11EVM Evaluation Module |
| BR278/D | M68HC11EVB Evaluation Board |
| BR291/D | M68705EVM Evaluation Module |
| BR479/D | M68HC11 Microcontroller – EEPROM |
| BR736/D | M68HC11EVBU Universal Evaluation Board |
| BR748/D | M68HC711D3PGMR Programmer Board |
| BR1116/D | Advanced Microcontroller Division Literature Guide |
| *BR1136/D | Motorola Fuzzy Logic |
| BR1161/D | Infinite Solutions – Motorola's CSIC Family of Microcontrollers: The 68HC05 and 68HC08 |
| BR1170/D | Hardware Development Tools |
| BR1179/D | Motorola CSIC Microcontrollers – Extraordinary Flexibility |
| BR1182/D | Motorola Modular Evaluation Systems (MMEVS) |
| BR1183/D | Motorola Modular Development Systems (MMDS) |
| BR1184/D | Emulation Modules (EM) |
| BR1185/D | Target Cable Accessories |
| BR1186/D | 68HC705 Parallel Programmers (PGMR) and 68HC708 Universal Serial Programmer (SPGMR08) |
| BR1187/D | Motorola CAN – The Total Solution for CAN Microcontrollers |

**Microprocessors: 8-bit MPU/MCU** continued

| | |
|---|---|
| BR1190/D | In-Circuit Simulators (ICS) |
| BR1465/D | 8-bit Microcontrollers for Multiplex Wiring |
| BR1480/D | Silicon Solutions for Off Line Motor Drives |
| BR1704/D | 68HC08: High Performance, 8-bit Microcontrollers with CAN, J1850 and Flash Memory Options |
| * BR1772/D | Instrument Cluster Systems |
| * BR1791/D | Internet and Networking Solutions for Motorola 68HC08 Microcontrollers |
| BR3006/D | Wireless Communications Resource Guide |
| * BR3034/D | Providing the Appliance Electronic Solutions You Need Today. And Tomorrow |
| CPU08RM/AD | M68HC08 Central Processor Unit Reference Manual |
| CPU12RG/D | CPU12 Reference Guide |
| DMA08RM/AD | DMA08 Direct Memory Access Reference Manual |
| FLYR14/D | Computer-Controlled DC Motor Drives: System Development Tool Set |
| HC05C0GRS/D | 68HC05C0 General Release Specification |
| HC05C12AGRS/D | MC68HC05C12A, MC68HCL05C12A, MC68HSC05C12A General Release Specification |
| HC05C4AGRS/D | MC68HC05C4A, MC68HCL05C4A, MC68HSC05C4A General Release Specification |
| HC05C8AGRS/D | MC68HC05C8A, MC68HCL05C8A, MC68HSC05C8A General Release Specification |
| HC05C9AGRS/D | MC68HC05C9A, MC68HCL05C9A, MC68HSC05C9A General Release Specification |
| HC05CT4GRS/D | MC68HC05CT4 General Release Specification |
| HC05H12GRS/D | MC68HC(7)05H12 General Release Specification |
| HC05J5AGRS/H | 68HC05J5A/68HRC05J5A/68HC705J5A/ 68HRC705J5A General Release Specification |
| HC05L16GRS/D | MC68HC05L16/MC68HC705L16 General Release Specification |
| HC05L5GRS/D | 68HC05L5/68HC705L5 General Release Specification |
| HC05PL4GRS/H | MC68HC05PL4A, MC68HC05PL4B, MC68HC705PL4B General Release Specification |
| HC05RC18GRS/D | MC68HC05RC9/MC68HC05RC18 General Release Specification |
| HC08KL8GRS/D | MC68HC08KL8 General Release Specification |
| HC68VBIGRS/D | MC68HC68VBI General Release Specification |
| HC705CT4GRS/D | MC68HC705CT4 General Release Specification |
| HC705JB2GRS/H | 68HC705JB2 General Release Specification |
| HC705MC4GRS/D | MC68HC705MC4 General Release Specification |
| HC705RC17GRS/D | 68HC705RC17 General Release Specification |
| HC708KL8GRS/D | 68HC708KL8 General Release Specification |
| HC708MP16GRS/D | MC68HC708MP16 General Release Specification |
| HC908AT32GRS/D | MC68HC908AT32 General Release Specification |
| HC908MR24GRS/D | 68HC908MR24 General Release Specification |
| M68EM05C0UM/D | M68EM05C0 Emulation Module User's Module |
| * M68EM05E5UM/D | M68EM05E5 Emulation Module User's Manual |
| * M68EM05JP7UM/D | M68EM05JP7 Emulation Module User's Manual |
| * M68EM05P18UM/D | M68EM05P18 Emulation Module User's Manual |
| * M68EM05V8UM/D | M68EM05V8 Emulation Module User's Manual |
| * M68EM08AX48UM/D | M68EM08AX48 Emulation Module User's Manual |
| * M68EM08MP16UM/D | M68EM08MP16 Emulation Module User's Manual |
| * M68EML05P6AUM/D | M68EML05P6A Emulation Module User's Manual |
| M68HC05AG/AD | M68HC05 Applications Guide |
| M68HC08RG/AD | HC08 Family Reference Guide |
| M68HC11RM/AD | M68HC11 Reference Manual |
| M68PRM/D | M6800 Programming Reference Manual |
| M6805UM/AD3 | M6805 HMOS / M146805 CMOS Family User's Manual |
| M6809PM/AD | MC6809-MC6809E Microprocessor Programming Manual (1981) |
| MC68HC05CxRG/AD | MC68HC05Cx HCMOS Single-Chip Microcontrollers Programming Reference Guide |
| MC68HC11A8RG/AD | MC68HC11A8 Programming Reference Guide |
| MC68HC11C0RG/AD | MC68HC11C0 Programming Reference Guide |
| MC68HC11D3RG/AD | MC68HC11D3/MC68HC711D3 Programming Reference Guide |
| MC68HC11ERG/AD | MC68HC11E Programming Reference Guide |
| MC68HC11F1RG/AD | MC68HC11F1 Programming Reference Guide |
| MC68HC11K4RG/AD | MC68HC11K4/MC68HC711K4 Programming Reference Guide |
| MC68HC11L6RG/AD | MC68HCL6/MC68HC711L6 Programming Reference Guide |
| MC68HC11MRG/AD | M68HC11 M Series Programming Reference Guide |
| MC68HC11NRG/AD | MC68HC11N Series Programming Reference Guide |
| MCCIRM/AD | Multichannel Communication Interface Reference Manual |
| MCUASM/D | MCUasm Assembly Language Development Toolset |
| MCUDEVTLDIR/D | Motorola Microcontroller Development Tools Directory |
| * SG187/D | Automotive Selector Guide |
| TIM08RM/AD | TIM08 Timer Interface Module Reference Manual |

## 16-bit MPU/MCU

| | |
|---|---|
| AN461/D | An Introduction to the HC16 for HC11 Users |
| AN476/D | CPU16 and the Configurable Timer Module (CTM) in Engine Control |
| AN1050/D | Designing for Electromagnetic Compatibility (EMC) with HCMOS Microcontrollers |
| AN1213/D | 16-bit DSP Servo Control with the MC68HC16Z1 |
| AN1230/D | A Background Debugging Mode Driver Package for Modular Microcontrollers |
| AN1233/D | Using M68HC16 Digital Signal Processing to Build an Audio Frequency Analyzer |
| AN1249/D | Brushed DC Motor Control Using the MC68HC16Z1 |
| AN1254/D | Using the MC68HC16Z1 for Audio Tone Generation |
| AN1259/D | System Design and Layout Techniques for Noise Reduction in MCU-Based Systems |
| AN1263/D | Designing for Electromagnetic Compatibility with Single-Chip Microcontrollers |
| AN1280/D | Using and Extending D–Bug 12 Routines |
| AN1280A/D | Using the Callable Routines in D-Bug12 |
| AN1283/D | Transporting M68HC11 Code to M68HC16 Devices |
| AN1284/D | Transporting M68HC11 Code to M68HC12 Devices |
| AN1295/D | Demonstration Model of fuzzyTECH Implementation on M68HC12 |
| AN1705/D | Noise Reduction Techniques for Microcontroller-Based Systems |
| AN1706/D | Microcontroller Oscillator Circuit Design Considerations |

| | |
|---|---|
| AN1718/D | A Serial Bootloader for Reprogramming the MC68HC912B32 Flash EEPROM |
| AN1774/D | Interfacing the MC68HC912B32 to an LCD Module |
| *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| *AN1816/D | Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol |
| AR362/D | Whipping Up Real-Time Designs – Programming Motorola's TPU |
| DCE406/D | Interface for MC68000 to DSP56001 Host Port |
| EB183/D | Erasing and Programming the FLASH EEPROM on the MC68HC912B32 |
| *EB251/D | How to Calculate Instruction Times on the MC68HC16 |
| EB252/D | MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers |
| EB259/D | Why MC68300 and MC68HC16 MCUs May Halt after Release of Reset |
| EB260/D | Why an MC68300 or MC68HC16 Microcontroller May Fail to Release Reset |
| EB261/D | Autovector Generation Using Chip Select Logic on MC68300 and MC68HC16 Devices |
| EB262/D | DSACK Generation on the System Integration and Single-Chip Integration |
| EB263/D | How to Program Chip Selects on Modular Microcontrollers with a System Integration Module or a Single-Chip Integration Module |
| EB264/D | Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the Single-Chip Integration Module or System Integration Module |
| EB265/D | Generating Edge-Sensitive Interrupts on the MC683xx and MC68HC16 Families of MCUs |
| *EB266/D | Unexplained Three-Stating of the Address Bus on M68300 and M68HC16 Devices |
| EB269/D | Using the SCI on Modular MCUs: An Example |
| EB273/D | Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors |
| *EB274/D | Generating Interrupts on the Time Processor Unit |
| EB275/D | Example Using the Queued Serial Peripheral Interface on Modular MCUs |
| *EB276/D | Using the ITC Function on the Time Processor Unit A |
| EB277/D | Coherency in the Time Processor Unit (TPU) |
| EB279/D | Low Output Levels on Output Pins |
| EB280/D | Programming the Channel Control Registers on the Time Processor Unit |
| EB281/D | Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers |
| *EB282/D | Using the Output Compare Function on the Time Processor Unit and an Example that Includes PPWA |
| EB305/D | Startup Problems When Using a Software Background Mode Debugger and Booting from RAM or an Empty ROM Socket |
| EB306/D | Using Exercise 7 on the M68HC16Z1EVB and the Necessity of Word Alignment |
| EB309/D | Using Exercise 8 on the MC68HC16Z1EVB |
| *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |
| TPUPN04/D | Table Stepper Motor TPU Function (TSM) |
| TPUPN05/D | Multichannel PWM TPU Function (MCPWM) |
| TPUPN06/D | Programmable Time Accumulator TPU Function (PTA) |
| TPUPN07/D | Asynchronous Serial Interface TPU Function (UART) |
| TPUPN08/D | New Input Capture/Input Transition Counter TPU Function (NITC) |
| TPUPN09/D | Multiphase Motor Commutation TPU Function (COMM) |
| TPUPN10/D | Hall Effect Decode TPU Function (HALLD) |
| TPUPN11/D | Period/Pulse Width Accumulator TPU Function (PPWA) |
| TPUPN12/D | Output Compare TPU Function (OC) |
| TPUPN13/D | Stepper Motor TPU Function (SM) |
| TPUPN14/D | Position-Synchronised Pulse Generator (PSP) |
| TPUPN15A/D | Period Measurement with Additional Transition Detection TPU Function (PMA) |
| TPUPN15B/D | Period Measurement with Missing Transition Detection TPU Function (PMM) |
| TPUPN17/D | Pulse Width Modulation TPU Function (PWM) |
| TPUPN18/D | Discrete Input/Output TPU Function (DIO) |
| TPUPN19/D | Synchronized Pulse-Width Modulation (SPWM) |
| TPUPN20/D | Quadrature Decode TPU Function (QDEC) |

***Additional information relevant to 16-bit MPU/MCU may be found in the following Motorola documents:***

| | |
|---|---|
| BDLCRM/AD | Byte Data Link Controller Reference Manual |
| BR1116/D | Advanced Microcontroller Division Literature Guide |
| BR1133/D | 68K and ColdFire Family Product Portfolio Overview |
| BR1169/D | The M68HC16 and M68300 Families of Modular Microcontrollers |
| BR1170/D | Hardware Development Tools |
| BR1187/D | Motorola CAN – The Total Solution for CAN Microcontrollers |
| *BR1199/D | The M68HC12 Family of 16-bit Microcontrollers |
| BR1480/D | Silicon Solutions for Off Line Motor Drives |
| *BR1772/D | Instrument Cluster Systems |
| BR3006/D | Wireless Communications Resource Guide |
| CPU12RM/AD | CPU12 Reference Manual |
| CPU16RM/AD | M68HC16 Family Reference Manual |
| M6809PM/AD | MC6809-MC6809E Microprocessor Programming Manual (1981) |
| M68000UM/AD | M68000 8-/16-/32-bit Microprocessors User's Manual, Ninth Edition |
| MC68HC16Y1UM/AD | MC68HC16Y1 User's Manual |
| MC68HC16ZUM/AD | M68HC16 Z Series User's Manual |
| MCUASM/D | MCUasm Assembly Language Development Toolset |
| MCUDEVTLDIR/D | Motorola Microcontroller Development Tools Directory |
| SCIMRM/AD | Single-Chip Integration Module Reference Manual |
| *SG187/D | Automotive Selector Guide |
| SIMRM/AD | System Integration Module Reference Manual |

## 32-bit MPU/MCU

| | |
|---|---|
| AN473/D | A Minimum Evaluation System for the MC68331 and MC68332 |
| AN1051/D | Transmission Line Effects in PCB Applications |
| AN1062/D | Using the QSPI for Analog Data Acquisition |
| AN1125/D | DRAM Interface to the MC88200 M Bus |
| AN1200/D | Configuring the M68300 Family Time Processing Unit (TPU) |
| AN1230/D | A Background Debugging Mode Driver Package for Modular Microcontrollers |
| AN1236/D | Timing Performance of TPU I/O Hardware |
| AN1255/D | MC68F333 Flash EEPROM Programming Utilities |
| AN1259/D | System Design and Layout Techniques for Noise Reduction in MCU-Based Systems |
| AN1263/D | Designing for Electromagnetic Compatibility with Single-Chip Microcontrollers |
| AN1264/D | JTAG Flash Memory Programmer |
| *AN1703/D | Low Power on the SCM68000 (EC000 Core) |
| AN1705/D | Noise Reduction Techniques for Microcontroller-Based Systems |
| AN1706/D | Microcontroller Oscillator Circuit Design Considerations |
| AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| *AN1765/D | Ethernet Configuration for a ColdFire Evaluation Board to Download Files from a Personal Computer |
| *AN1776/D | Stereo Audio Transmission over the CAN Bus Using the Motorola MC68376 with TouCAN Module |
| *AN1791/D | Using the QAD64 Module on the M68F375 Microcontroller |
| *AN1802/D | Connecting the MCF5307 to 168-pin Unbuffered SDRAM DIMMs |
| ANE426/D | An MC68030 32-bit High Performance Minimum System |
| AR350/D | Adapt Non-ISDN Terminals to ISDN Data Rates |
| AR362/D | Whipping Up Real-Time Designs – Programming Motorola's TPU |
| DC414/D | An 8-bit EPROM Interface for an MC68EC040/ MC68360 System |
| EB163/D | Running the MC88110 in Lockstep |
| EB164/D | Interrupt Latency in the MC88110 |
| EB165/D | Hardware Implications of xmem as a st followed by a ld |
| EB253/D | How to Use the Lookup and Interpolate Instruction on the CPU32 |
| *EB254/D | Setting the Programming Voltage of Modular Microcontrollers with FLASH EEPROM |
| EB256/D | Use of the Lock Bit on Modular Microcontrollers with FLASH EEPROM |
| EB257/D | Detecting Loss of Clock on Modular Microcontrollers |
| EB258/D | Sources of Reset on Modular Microcontrollers |
| EB259/D | Why MC68300 and MC68HC16 MCUs May Halt after Release of Reset |
| EB260/D | Why an MC68300 or MC68HC16 Microcontroller May Fail to Release Reset |
| EB261/D | Autovector Generation Using Chip Select Logic on MC68300 and MC68HC16 Devices |
| EB263/D | How to Program Chip Selects on Modular Microcontrollers with a System Integration Module or a Single-Chip Integration Module |
| EB264/D | Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the Single-Chip Integration Module or System Integration Module |
| EB265/D | Generating Edge-Sensitive Interrupts on the MC683xx and MC68HC16 Families of MCUs |
| *EB266/D | Unexplained Three-Stating of the Address Bus on M68300 and M68HC16 Devices |
| *EB267/D | The Double Bus Fault Monitor |
| EB268/D | Starting and Stopping the Time Processor Clock Using the Background Debug Mode |
| EB269/D | Using the SCI on Modular MCUs: An Example |
| EB270/D | Problems with the PPWA Function on Revision P MC68332 Devices |
| EB273/D | Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors |
| *EB274/D | Generating Interrupts on the Time Processor Unit |
| EB275/D | Example Using the Queued Serial Peripheral Interface on Modular MCUs |
| *EB276/D | Using the ITC Function on the Time Processor Unit A |
| EB277/D | Coherency in the Time Processor Unit (TPU) |
| EB278/D | Latency on the Time Processor Unit |
| EB279/D | Low Output Levels on Output Pins |
| EB280/D | Programming the Channel Control Registers on the Time Processor Unit |
| EB281/D | Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers |
| *EB282/D | Using the Output Compare Function on the Time Processor Unit and an Example that Includes PPWA |
| EB305/D | Startup Problems When Using a Software Background Mode Debugger and Booting from RAM or an Empty ROM Socket |
| *EB310/D | Using Bus Error Stack Frames to Diagnose CPU32 Released Write Faults |
| *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |
| EB414/D | Low Power Write Enable Generation for M68300 Family Microprocessors |
| TPUPN00/D | Using the TPU Function Library and TPU Emulation Mode |
| TPUPN01/D | Queued Output Match TPU Function (QOM) |
| TPUPN02/D | Fast Quadrature Decode TPU Function (FQD) |
| TPUPN03/D | Frequency Measurement TPU Function (FQM) |
| TPUPN04/D | Table Stepper Motor TPU Function (TSM) |
| TPUPN05/D | Multichannel PWM TPU Function (MCPWM) |

| | |
|---|---|
| TPUPN06/D | Programmable Time Accumulator TPU Function (PTA) |
| TPUPN07/D | Asynchronous Serial Interface TPU Function (UART) |
| TPUPN08/D | New Input Capture/Input Transition Counter TPU Function (NITC) |
| TPUPN09/D | Multiphase Motor Commutation TPU Function (COMM) |
| TPUPN10/D | Hall Effect Decode TPU Function (HALLD) |
| TPUPN11/D | Period/Pulse Width Accumulator TPU Function (PPWA) |
| TPUPN12/D | Output Compare TPU Function (OC) |
| TPUPN13/D | Stepper Motor TPU Function (SM) |
| TPUPN14/D | Position-Synchronised Pulse Generator (PSP) |
| TPUPN15A/D | Period Measurement with Additional Transition Detection TPU Function (PMA) |
| TPUPN15B/D | Period Measurement with Missing Transition Detection TPU Function (PMM) |
| TPUPN17/D | Pulse Width Modulation TPU Function (PWM) |
| TPUPN18/D | Discrete Input/Output TPU Function (DIO) |
| TPUPN19/D | Synchronized Pulse-Width Modulation (SPWM) |
| TPUPN20/D | Quadrature Decode TPU Function (QDEC) |

***Additional information relevant to 32-bit MPU/MCU may be found in the following Motorola documents:***

| | |
|---|---|
| BR484/D | 68302 |
| BR1114/D | The 68300 Family Integrated Microprocessors and Microcontrollers |
| BR1116/D | Advanced Microcontroller Division Literature Guide |
| BR1118/D | Motorola's 68LC040 Microprocessor |
| BR1119/D | Motorola's 68EC040 Microprocessor |
| BR1133/D | 68K and ColdFire Family Product Portfolio Overview |
| BR1153/D | The 68060 Family |
| BR1169/D | The M68HC16 and M68300 Families of Modular Microcontrollers |
| BR1170/D | Hardware Development Tools |
| BR1187/D | Motorola CAN – The Total Solution for CAN Microcontrollers |
| BR1427/D | PC Brochure |
| * BR1475/D | Datacomms |
| BR1480/D | Silicon Solutions for Off Line Motor Drives |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| BR3006/D | Wireless Communications Resource Guide |
| BR3020/D | Remote Access: ISDN Solutions Kit |
| COLDFIREFAM/D | ColdFire: Variable-Length RISC Processors |
| CPU32RM/AD | CPU32 Central Processor Unit Reference Manual |
| CTMRM/D | Configurable Timer Module Reference Manual |
| EMDVPOC/D | Embedded Developer Pocket Guide |
| GPTRM/AD | Modular Microcontroller Family General Purpose Timer Reference Manual |
| M68000PM/AD | M68000 Family Programmer's Reference Manual |
| M68020UM/AD | MC68020/MC68EC020 Microprocessors User's Manual |
| M68040UM/AD | MC68040, MC68040V, MC68LC040, MC68EC040, MC68EC040V Microprocessors User's Manual |
| M68060UM/AD | MC68060, MC68LC060, MC68EC060 Microprocessors User's Manual |
| MC68EN302RM/AD | MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual (Supplement to MC68302UM/AD) |
| MC68EZ328UM/D | MC68EZ328 DragonBall-EZ Integrated Processor User's Manual |
| MC68F333UM/AD | MC68F333 User's Manual |
| MC68LC302RM/AD | MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual |
| MC68PM302RM/AD | Integrated Multiprotocol Processor with PCMCIA Interface Reference Manual |
| MC68SC302UM/AD | MC68SC302 Passive ISDN Protocol Engine User's Manual |
| MC68030UM/AD | MC68030 Enhanced 32-bit MPU User's Manual, third edition |
| MC68302UM/AD | MC68302 Integrated Multiprotocol Processor User's Manual |
| MC68306UM/AD | MC68306 Integrated EC000 Processor User's Manual |
| MC68322UM/AD | Bandit: MC68322 Integrated Printer Processor User's Manual |
| MC68328UM/AD | MC68328 (DragonBall) Integrated Processor User's Manual |
| MC68330UM/AD | MC68330 Integrated CPU32 Processor Users Manual |
| MC68332UM/AD | MC68332 User's Manual |
| MC68340UM/AD | MC68340 Integrated Processor User's Manual |
| MC68356UM/AD | MC68356 Signal Processing Communications Engine User's Manual |
| MC68360UM/AD | MC68360 Quad Integrated Communications Controller User's Manual |
| MC68840UM/AD | MC68840 Integrated Fiber Distributed Data Interface User's Manual |
| MC88410UM/AD | MC88410 Secondary Cache Controller User's Manual |
| MCF5102UM/AD | MCF5102 ColdFire User's Manual |
| MCF5200PRM/AD | MCF5200 ColdFire Microprocessor Family Programmer's Reference Manual |
| MCF5202UM/AD | ColdFire MCF5202 User's Manual |
| * MCF5204UM/AD | MCF5204 ColdFire Integrated Microprocessor User's Manual |
| * MCF5206UM/AD | MCF5206 ColdFire Integrated Microprocessor User's Manual |
| MCF5307UM/AD | MCF5307 ColdFire Integrated Microprocessor User's Manual |
| MCUDEVTLDIR/D | Motorola Microcontroller Development Tools Directory |
| QADCRM/AD | Queued Analog-to Digital Converter Reference Manual |
| QMCSUPPLEMENT/AD | MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals |
| QSMRM/AD | Queued Serial Module Reference Manual |
| SCIMRM/AD | Single-Chip Integration Module Reference Manual |
| SG171/D | Fast Static RAM Division Product Update |
| SG175/D | Networking Systems Division and Personal Computing Division: Product Information |
| * SG187/D | Automotive Selector Guide |
| SIMRM/AD | System Integration Module Reference Manual |
| TPURM/AD | M68300 Family Time Processor Unit Reference Manual |

## 8-bit Peripherals

| | |
|---|---|
| AN1759/D | Add a Non-Volatile Clock to the MC68HC705J1A |
| ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |

***Additional information relevant to 8-bit Peripherals may be found in the following Motorola document:***

| | |
|---|---|
| BR1116/D | Advanced Microcontroller Division Literature Guide |

## 16/32-bit Peripherals

| | |
|---|---|
| ANE426/D | An MC68030 32-bit High Performance Minimum System |
| DC414/D | An 8-bit EPROM Interface for an MC68EC040/ MC68360 System |

***Additional information relevant to 16/32-bit Peripherals may be found in the following Motorola documents:***

| | |
|---|---|
| BR488/D | 68306 68307 68322 |
| BR489/D | 68360 Quad Integrated Communications Controller (QUICC) |
| BR1104/D | Motorola's FDDI Chip Set |
| * MC68HC681UM/AD | MC68HC681 Dual Asynchronous Receiver/ Transmitter (DUART) |
| MC68HC901UM/AD | MC68HC901 Multi-Function Peripheral User's Manual |
| MC68605UM/AD | MC68605 X.25 Protocol Controller User's Manual |
| MC68824UM/AD | MC68824 Token Bus Products User's Manual |
| MC68836UM/AD | MC68836 FDDI User's Manual |
| MC68837UM/AD | MC68837 FDDI User's Manual |
| MC68838UM/AD | MC68838 FDDI User's Manual |
| MC68839UM/AD | MC68839 FDDI System Interface User's Manual |
| MC68847UM/AD | MC68847 Quad ELM FDDI User's Manual |
| MC88200UM/AD | MC88200 Cache/Memory Management Unit User's Manual |
| SG175/D | Networking Systems Division and Personal Computing Division: Product Information |

## PowerPC

| | |
|---|---|
| AN486/D | Low Cost MPC601 EVM |
| AN1265/D | Configuring the MPC2604GA Integrated L2 Cache with the MPC106 |
| AN1267/D | PowerPC 603 Hardware Interrupt Latency in Embedded Applications |
| AN1269/D | PowerPC Microprocessor Clock Modes |
| AN1271/D | PowerPC 60x Microprocessor to AD1848 CODEC Interface |
| AN1272/D | Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption |
| AN1281/D | MPC505 Interrupts |
| AN1282/D | Board Strategies for Ensuring Optimum Frequency Synthesizer Performance |
| AN1291/D | Avoiding Multiprocessing Paradoxes with the PowerPC 604 Microprocessor |
| AN1294/D | Multiprocessor Systems and the PowerPC 603e Microprocessor |
| AN1579/D | Understanding the Multivibrator Based Crystal Oscillator Circuit Used on the BiCMOS MPC Family of Clock Drivers |
| AN1722/D | SDRAM System Design Using the MPC106 |
| AN1725/D | Initializing SDRAM Parameters for Motorola MPC106-Based Systems |
| AN1727/D | Designing PCI 2.1-Compliant MPC106 Systems |
| AN1768/D | Using Registered SDRAM DIMMs with the MPC106 |
| AN1769/D | Designing a Minimal PowerPC System |
| AN1777/D | MPC8xx to BurstRAM Interfacing |
| * AN1778/D | Using the MIOS on the MPC555 Evaluation Board |
| * AN1794/D | PowerPC Backside L2 Timing Analysis for the PCB Design Engineer |
| * AN1795/D | Designing G4 Systems |
| * AN1800/D | Programming the Thermal Assist Unit in the MPC750 Microprocessor |
| * AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm Application |
| AN4000/D | Visual Debug for MPC60x |
| AR359/D | The Making of the PowerPC |
| AR360/D | PowerPC 620 Soars |
| * AR363/D | Programming PowerPC Embedded Applications |

***Additional information relevant to PowerPC may be found in the following Motorola documents:***

| | |
|---|---|
| BR1154/D | MPC500 Family: RISC PowerPC Microcontrollers |
| BR1155/D | MPC500 Family: Software Development Tools |
| BR1165/D | MPC500 Family: RTEK Real-Time Embedded Kernel |
| BR1166/D | MPC500 Family: Evaluation Board |
| BR1427/D | PC Brochure |
| * BR1466/D | The New Clock Driver Generation |
| * BR1475/D | Datacomms |
| * BR1502/D | Wireless Infrastructure Solutions |
| BR1724/D | PowerPC Resource Guide |
| * BR1751/D | Empower Your Ideas |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| * BR1783/D | Touchstone™ Manufacturers Kit for Internet Access Devices |
| * BR1798/D | PowerPC Resource Guide |
| DL156/D | Fast Static RAM – Component and Module Data |
| EMDVPOC/D | Embedded Developer Pocket Guide |
| MCUDEVTLDIR/D | Motorola Microcontroller Development Tools Directory |
| MPCBUSIF/AD | PowerPC Microprocessor Family: The Bus Interface for 32-bit Microprocessors |
| MPCPRG/D | PowerPC Microprocessor Family: The Programmer's Reference Guide |
| MPC105UM/AD | PowerPC PCI Bridge/Memory Controller User's Manual |
| * MPC509UM/AD | MPC509 RISC Microprocessor User's Manual |
| MPC603eUM/AD | PowerPC 603e RISC Microprocessor User's Manual |
| * MPC604EUM/AD | PowerPC 604e RISC Microprocessor User's Manual |
| * MPC620UM/AD | PowerPC 620 RISC Microprocessor User's Manual |
| MPC750UM/AD | MPC750 RISC Microprocessor User's Manual |
| * MPC801UM/AD | MPC801 Integrated Microprocessor for Embedded Systems User's Manual |

| | |
|---|---|
| MPC821UM/AD | MPC821 PowerPC Portable Systems Microprocessor User's Manual |
| MPC823RG/D | PowerPC MPC823 Pocket Guide |
| MPC823UM/D | PowerPC MPC823 User's Manual |
| * MPC850UM/D | MPC850 Integrated Communications Microprocessor User's Manual |
| MPC860UM/AD | MPC860 PowerQUICC User's Manual |
| * MPC8240UM/D | MPC8240 Integrated Processor User's Manual |
| * MPC8260UM/D | MPC8260 PowerQUICC II User's Manual |
| PPC620/D | PowerPC 620 Microprocessors |
| PPC620FACT/D | PowerPC 620 Microprocessor Fact Sheet |
| QMCSUPPLEMENT/AD | MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals |
| RCPURM/AD | MPC500 Family: RCPU Reference Manual |
| SG171/D | Fast Static RAM Division Product Update |
| SG175/D | Networking Systems Division and Personal Computing Division: Product Information |
| * SG187/D | Automotive Selector Guide |
| SIURM/AD | MPC500 Family: System Integration Unit Reference Manual |

## M•CORE

| | |
|---|---|
| * AN1817/D | MMC20xx M•CORE OnCE Port Communication and Control Sequences |

***Additional information relevant to M•CORE may be found in the following Motorola documents:***

| | |
|---|---|
| * BR1796/D | Internet and Networking Solutions for Motorola's M•CORE Microcontrollers |
| MCORERM/AD | M•CORE Reference Manual |
| MCORESALES/D | M•CORE Architecture |
| MMC2001RM/D | M•CORE MMC2001 Reference Manual |

## Motor & Lighting Control

| | |
|---|---|
| AN1249/D | Brushed DC Motor Control Using the MC68HC16Z1 |
| AN1285/D | Stepper Motor Control with an MC68HC11E9 Microcontroller |
| AN1307/D | A Simple Pressure Regulator Using Semiconductor Pressure Transducers |
| AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| AN1624/D | ITC137 68HC708MP16 Motion Control Development Board |
| * AN1627/D | Low Cost High Efficiency Sensorless Drive for Brushless DC Motor Using MC68HC(7)05MC4 |
| * AN1663/D | Low Cost Universal Motor Sensorless Phase Angle Drive System |
| * AN1664/D | Low Cost 3-Phase AC Motor Control System Based on MC68HC908MR24 |
| AN1702/D | Brushless DC Motor Control Using the MC68HC705MC4 |
| AN1712/D | "Get Your Motor Running" with the MC68HC708MP16 |
| * AN1717/D | ITC127 MC68HC705MC4 Motion Control Development Board |
| AN1728/D | Making Low-Distortion Waveforms with the MC68HC708MP16 |
| AN1734/D | Pulse Width Modulation Using the 16-Bit Timer |
| * AN1792/D | Using an MC68HC908MR24 in Place of an MC68HC708MP16 |
| * AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| TPUPN04/D | Table Stepper Motor TPU Function (TSM) |
| TPUPN09/D | Multiphase Motor Commutation TPU Function (COMM) |
| TPUPN10/D | Hall Effect Decode TPU Function (HALLD) |
| TPUPN13/D | Stepper Motor TPU Function (SM) |
| TPUPN17/D | Pulse Width Modulation TPU Function (PWM) |
| TPUPN19/D | Synchronized Pulse-Width Modulation (SPWM) |
| TPUPN20/D | Quadrature Decode TPU Function (QDEC) |

***Additional information relevant to Motor & Lighting Control may be found in the following Motorola documents:***

| | |
|---|---|
| BR477/D | Smart Mover – Stepper Motors with Integrated Serial Bus Controller |
| BR1480/D | Silicon Solutions for Off Line Motor Drives |
| * BR3034/D | Providing the Appliance Electronic Solutions You Need Today. And Tomorrow |
| * DSP56LF812UM/AD | DSP56LF812 16-bit Digital Signal Processor User's Manual |
| DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |
| * DSP56824UM/AD | DSP56824 16-bit Digital Signal Processor User's Manual |
| FLYR14/D | Computer-Controlled DC Motor Drives: System Development Tool Set |
| HC05H12GRS/D | MC68HC(7)05H12 General Release Specification |
| HC705MC4GRS/D | MC68HC705MC4 General Release Specification |
| * SG187/D | Automotive Selector Guide |

## Mounting Techniques & Surface Mount

| | |
|---|---|
| AN936/D | Mounting Techniques, Lead Forming and Testing of Motorola's MPX Series Pressure Transducers |
| AN1051/D | Transmission Line Effects in PCB Applications |
| AN1231/D | Plastic Ball Grid Array (PBGA) |
| AN1232/D | Thermal Performance of Plastic Ball Grid Array (PBGA) Packages for Next Generation FSRAM Devices |
| AN1580/D | Mounting and Soldering Recommendations for the Motorola Power Flat Pack Package |
| AN1617/D | Mounting Recommendations for Copper Tungsten Flanged Transistors |
| AN1673/D | Solder Reflow Mounting Method for the MRF286 and Similar Packages |
| AN1705/D | Noise Reduction Techniques for Microcontroller-Based Systems |
| AN4005/D | Thermal Management and Mounting Method for the PLD 1.5 RF Power Surface Mount Package |
| APR42/D | 15 x 15mm PBGA Daisy-Chain Application Report |
| * AR561/D | Next Generation Packaging Solutions |
| EB107/D | Mounting Considerations for Motorola RF Power Modules |

**Mounting Techniques & Surface Mount** continued

| | |
|---|---|
| EB109/D | Low Cost UHF Device Gives Broadband Performance at 3.0 Watts Output |
| EB209/D | Mounting Method for RF Power Leadless Surface Mount Transistors |
| *EB211/D | Thermal Management and Solder Mounting Method for the MRF286, 60 Watt Power Device in a CuW (Copper Tungsten) Base Package |
| *EB303/D | Handling Considerations for Avoiding Intermittent Programming and Execution Failures with MC68HC11-Windowed Devices |

***Additional information relevant to Mounting Techniques & Surface Mount may be found in the following Motorola documents:***

| | |
|---|---|
| BR1150/D | 7 x 17 PBGA Sample Preview |
| BR1176/D | Motorola & Ball Grid Array Technology |
| * BR1518/D | Motorola's Small Outline Package for Pressure Sensors |

## Multimedia/Internet

| | |
|---|---|
| AN492/D | A Video Display Board for CD-i Development |
| AN1254/D | Using the MC68HC16Z1 for Audio Tone Generation |
| AN1271/D | PowerPC 60x Microprocessor to AD1848 CODEC Interface |

***Additional information relevant to Multimedia/Internet may be found in the following Motorola documents:***

| | |
|---|---|
| BR1171/D | Motorola Multimedia Communications |
| * BR1178/D | ADSL |
| * BR1757/D | Multichannel Infrastructure Modem and Evaluation Kit |
| * BR1771/D | Streamaster – Digital Imagination: Let Your Mind Flow |
| * BR1783/D | Touchstone™ Manufacturers Kit for Internet Access Devices |
| * BR1791/D | Internet and Networking Solutions for Motorola 68HC08 Microcontrollers |
| * BR1796/D | Internet and Networking Solutions for Motorola's M•CORE Microcontrollers |
| * BR1803/D | Digital Audio/DTV Solutions |
| * DSP56011UM/AD | DSP56011 24-bit Digital Signal Processor User's Manual |
| DSP56302UM/AD | DSP56302 User's Manual |
| DSP56303UM/AD | DSP56303 User's Manual |
| DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |
| MPC823UM/D | PowerPC MPC823 User's Manual |

## Networking

| | |
|---|---|
| AN464/D | Software Driver Routines for the Motorola MC68HC05 CAN Module |
| AN1054/D | ISDN System Development Using MC145490EVK/ MC145491EVK Development Kits |
| AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| AN1296/D | Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card |
| AN1299/D | ATM Switch with Shared Memory – A Simple Model |
| AN1704/D | Switch Fabric Implementation Using Shared Memory |
| AN1726/D | Using Motorola's Fast Static RAM CAMs on a Media Independent Interface |
| AN1732/D | A Universal Serial Bus Gamepad Device using the MC68HC05JB2 |
| AN1758/D | Add Addressable Switches to the HC05 |
| *AN1765/D | Ethernet Configuration for a ColdFire Evaluation Board to Download Files from a Personal Computer |
| *AN1776/D | Stereo Audio Transmission over the CAN Bus Using the Motorola MC68376 with TouCAN Module |
| *AN1807/D | Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm Application |
| AR333/D | RF Modems Simplified |
| AR350/D | Adapt Non-ISDN Terminals to ISDN Data Rates |
| EB406/D | Getting Started with the FDDI ADS Board |

***Additional information relevant to Networking may be found in the following Motorola documents:***

| | |
|---|---|
| BDLCRM/AD | Byte Data Link Controller Reference Manual |
| BR1104/D | Motorola's FDDI Chip Set |
| BR1134/D | LonWorks Technology: the Smart Choice for Intelligent Distributed Control! |
| BR1187/D | Motorola CAN – The Total Solution for CAN Microcontrollers |
| BR1188/D | LonWorks Networks for Industrial and Process Control |
| * BR1475/D | Datacomms |
| * BR1502/D | Wireless Infrastructure Solutions |
| BR1729/D | MC92500 Asynchronous Transfer Mode Cell Processors |
| BR1731/D | Integrated Solutions for ATM Systems |
| * BR1764/D | Thin Clients. Smart Technology for a Fast Paced World |
| * BR1765/D | MC92501EVB ATMC Evaluation Board |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| * BR1791/D | Internet and Networking Solutions for Motorola 68HC08 Microcontrollers |
| * BR1796/D | Internet and Networking Solutions for Motorola's M•CORE Microcontrollers |
| BR3020/D | Remote Access: ISDN Solutions Kit |
| DL159/D | LonWorks Technology Device Data |
| LONUG/AD | LonBuilder User's Guide |
| MC68EN302RM/AD | MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual (Supplement to MC68302UM/AD) |
| MC68LC302RM/AD | MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual |
| MC68PM302RM/AD | Integrated Multiprotocol Processor with PCMCIA Interface Reference Manual |
| MC68302UM/AD | MC68302 Integrated Multiprotocol Processor User's Manual |

| | |
|---|---|
| MC68360UM/AD | MC68360 Quad Integrated Communications Controller User's Manual |
| MC68824UM/AD | MC68824 Token Bus Products User's Manual |
| MC68836UM/AD | MC68836 FDDI User's Manual |
| MC68837UM/AD | MC68837 FDDI User's Manual |
| MC68838UM/AD | MC68838 FDDI User's Manual |
| MC68839UM/AD | MC68839 FDDI System Interface User's Manual |
| MC68840UM/AD | MC68840 Integrated Fiber Distributed Data Interface User's Manual |
| MC68847UM/AD | MC68847 Quad ELM FDDI User's Manual |
| MC92500UM/D | ATM Cell Processor Design Reference Manual |
| MC92501UM/D | MC92501 ATM Cell Processor User's Manual |
| * MPC850UM/D | MPC850 Integrated Communications Microprocessor User's Manual |
| MPC860UM/AD | MPC860 PowerQUICC User's Manual |
| * MPC8240UM/D | MPC8240 Integrated Processor User's Manual |
| * MPC8260UM/D | MPC8260 PowerQUICC II User's Manual |
| QMCSUPPLEMENT/AD | MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals |
| SG175/D | Networking Systems Division and Personal Computing Division: Product Information |

## Optoelectronics and Displays

| | |
|---|---|
| AN1238/D | HC05 MCU LED Drive Techniques Using the MC68HC705J1A |
| AN1743/D | Scrolling Message Software |
| AN1745/D | Interfacing the HC705C8A to an LCD Module |
| AN1762/D | Automatic Contrast Control of LCD Displays Using the MC68HC708LN56 Microcontroller |
| AN1763/D | Driving LCD Displays Using the MC68HC705L16 Microcontroller |
| AN1774/D | Interfacing the MC68HC912B32 to an LCD Module |
| EB406/D | Getting Started with the FDDI ADS Board |

***Additional information relevant to Optoelectronics and Displays may be found in the following Motorola documents:***

| | |
|---|---|
| * BRMVVD01/D | VirtuoVue®Q Display: The World in a Single Vue |
| BR1480/D | Silicon Solutions for Off Line Motor Drives |
| MC68836UM/AD | MC68836 FDDI User's Manual |
| MC68837UM/AD | MC68837 FDDI User's Manual |
| MC68838UM/AD | MC68838 FDDI User's Manual |
| MC68839UM/AD | MC68839 FDDI System Interface User's Manual |
| MC68840UM/AD | MC68840 Integrated Fiber Distributed Data Interface User's Manual |
| MC68847UM/AD | MC68847 Quad ELM FDDI User's Manual |

## Phase-Locked Loop

| | |
|---|---|
| AN535/D | Phase-Locked Loop Design Fundamentals |
| AN1282/D | Board Strategies for Ensuring Optimum Frequency Synthesizer Performance |
| AN1579/D | Understanding the Multivibrator Based Crystal Oscillator Circuit Used on the BiCMOS MPC Family of Clock Drivers |
| * AN1639/D | Phase Noise Measurement Using the Phase Lock Technique |
| AN1671/D | MC145170 PSpice Modeling Kit |

***Additional information relevant to Phase-Locked Loop may be found in the following Motorola document:***

| | |
|---|---|
| BR3006/D | Wireless Communications Resource Guide |

## Pressure, Gas & Acceleration Sensors

| | |
|---|---|
| AN935/D | Compensating for Nonlinearity in the MPX10 Series Pressure Transducer |
| AN936/D | Mounting Techniques, Lead Forming and Testing of Motorola's MPX Series Pressure Transducers |
| AN1082/D | Simple Design for a 4-20mA Transmitter Interface Using a Motorola Pressure Sensor |
| AN1097/D | Calibration-Free Pressure Sensor System |
| AN1100/D | Analog to Digital Converter Resolution Extension Using a Motorola Pressure Sensor |
| AN1105/D | A Digital Pressure Gauge Using the Motorola MPX700 Series Differential Pressure Sensor |
| AN1304/D | Integrated Sensor Simplifies Bar Graph Pressure Gauge |
| AN1305/D | An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a Microprocessor |
| AN1307/D | A Simple Pressure Regulator Using Semiconductor Pressure Transducers |
| AN1309/D | Compensated Sensor Bar Graph Pressure Gauge |
| AN1315/D | An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor |
| AN1316/D | Frequency Output Conversion for MPX2000 Series Pressure Sensors |
| AN1318/D | Interfacing Semiconductor Pressure Sensors to Microcomputers |
| AN1322/D | Applying Semiconductor Sensors to Bar Graph Pressure Gauges |
| AN1324/D | A Simple Sensor Interface Amplifier |
| AN1325/D | Amplifiers for Semiconductor Pressure Sensors |
| AN1326/D | Barometric Pressure Measurement Using Semiconductor Pressure Sensors |
| AN1513/D | Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors |
| AN1516/D | Liquid Level Control Using a Motorola Pressure Sensor |
| AN1517/D | Pressure Switch Design with Semiconductor Pressure Sensors |
| AN1518/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| AN1525/D | The A-B-Cs of Signal-Conditioning Amplifier Design for Sensor Applications |
| AN1536/D | Digital Boat Speedometers |
| AN1551/D | Low-Pressure Sensing with the MPX2010 Pressure Sensor |
| AN1556/D | Designing Sensor Performance Specifications for MCU-based Systems |

**Pressure, Gas & Acceleration Sensors** continued

| | |
|---|---|
| AN1557/D | A Cookbook Approach to Designing a Differential-Signal Amplifier for Sensor Applications |
| AN1559/D | Application Considerations for a Switched Capacitor Accelerometer |
| AN1571/D | Digital Blood Pressure Meter |
| AN1573/D | Understanding Pressure and Pressure Measurement |
| AN1584/D | "Very Low Pressure" Smart Sensing Solution with Serial Communications Interface |
| AN1585/D | High-Performance, Dynamically-Compensated Smart Sensor System |
| AN1586/D | Designing a Homemade Digital Output for Analog Voltage Output Sensors |
| AN1611/D | Impact and Tilt Measurement Using Accelerometer |
| AN1612/D | Shock and Mute Pager Applications Using Accelerometer |
| AN1622/D | EMC Considerations for Automotive Sensors |
| AN1632/D | MMA1000P Product Overview and Interface Considerations |
| AN1636/D | Implementing Auto Zero for Integrated Pressure Sensors |
| AN1638/D | Offset Calibration of Gauge Pressure Sensor Using Parallel I/O Ports |
| AN1640/D | Reducing Accelerometer Susceptibility to BCI |
| AN1645/D | Micromachined Electromechanical Sensors for Automotive Applications |
| AN1646/D | Noise Considerations for Integrated Pressure Sensors |
| AN1651/D | ASB201 – Uncompensated Series Sensor Module |
| AN1652/D | ASB202 – MPX2000 Series Sensor Module |
| AN1653/D | ASB205 – MPX5000 Series Sensor Module |
| AN1654/D | ASB210 – 10" $H_2O$ Sensor Module |
| AN1655/D | ASB200 – Motorola Sensor Development Controller Board |
| AN1660/D | Compound Coefficient Pressure Sensor PSPICE Models |
| AN1668/D | Washing Appliance Sensor Selection |
| *AN1690/D | Alarm IC General Applications Overview |
| AN1754/D | Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer |
| AN4004/D | ±2g Acceleration Sensing Module Based on a ±40g Integrated Accelerometer |
| AR560/D | Simple Pressure Switches Comprise Transducers, Comparators and Op Amps |
| *AR601/D | Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors |
| *AR602/D | Home-Brewed Circuits Tailor Sensor Outputs to Specialized Needs |

***Additional information relevant to Pressure, Gas & Acceleration Sensors may be found in the following Motorola documents:***

| | |
|---|---|
| BR1477/D | Sensor Products Division: Competitive Product Cross Reference |
| *BR1497/D | Motorola's Signal Conditioned Surface Mount Pressure Sensor |
| *BR1498/D | Motorola's Stress Isolated Top Piston Fit Pressure Sensor |
| *BR1499/D | Motorola's Pressure Sensors: Heart of a Water Speed Measurement System |
| BR1512/D | Sensor Device Information Matrix – Quarter 1, 1999 |
| *BR1516/D | Alarm Integrated Circuits from Motorola |
| *BR1517/D | Acceleration Sensors from Motorola |
| *BR1518/D | Motorola's Small Outline Package for Pressure Sensors |
| *BR1519/D | Medical Sensing from Motorola |
| *BR1520/D | Smoke Detector ICs from Motorola |
| *BR1523/D | Motorola's New Integrated Automotive MAP/BAP Sensors Offer Top Performance at Half the Package Size |
| BR3005/D | Intelligent Sensor Solutions |
| BR3009/D | Senseon Intelligent Sensor Solutions |
| BR3012/D | Next Generation Packaging for SENSEON Pressure Sensors |
| BR3015/D | The SENSEON Family of Advanced Acceleration Sensors |
| BR3019/D | The SENSEON Chemical Sensor Family |
| *BR3034/D | Providing the Appliance Electronic Solutions You Need Today. And Tomorrow |
| DL200/D | Pressure Sensor Device Data |
| HB218/D | Senseon: Pressure Sensor Distributor Handbook |
| SG162/D | Sensor Products Division |
| *SG187/D | Automotive Selector Guide |

## Quality and Reliability

***Information relevant to Quality and Reliability may be found in the following Motorola documents:***

| | |
|---|---|
| BR1202/D | Motorola Quality System Review Guidelines |
| BR1427/D | PC Brochure |

## Radio Applications

| | |
|---|---|
| AN495/D | RDS Decoding for an HC11-Controlled Radio |
| AN1231/D | Plastic Ball Grid Array (PBGA) |
| AN1539/D | An IF Communication Circuit Tutorial |
| AN1597/D | Longwave Radio Data Decoding Using an HC11 and an MC3371 |
| AN1599/D | Power Control with the MRFIC0913 GaAs Integrated Power Amplifier and MC33169 Support IC |
| AN1610/D | Using Motorola's MRFIC1502 in Global Positioning System Receivers |
| AN1670/D | 60 watts, GSM 900MHz, LDMOS Two-Stage Amplifier |
| AN1675/D | A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low Noise Transistor |
| AN1676/D | A Cascade 2 Stage Low Noise Amplifier Using the MRF1047T1 Low Noise Transistor |
| *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |

| | |
|---|---|
| ANE416/D | MC68HC05B4 Radio Synthesizer |
| AR511/D | Biasing Solid State Amplifiers to Linear Operation |
| *AR629/D | Digital Predistortion Techniques for RF Power Amplifiers with CDMA Applications |
| *EB198/D | Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11 |

***Additional information relevant to Radio Applications may be found in the following Motorola documents:***

| | |
|---|---|
| BR1467/D | Extend Your Scope in Wireless Systems – The New Hipercomm Generation |
| * BR1502/D | Wireless Infrastructure Solutions |
| BR3006/D | Wireless Communications Resource Guide |
| * DSP56600FM/AD | DSP56600 16-bit Digital Signal Processor Family Manual |
| HB219/D | Introduction to the Oncore ChipSet |
| SG46/D | RF Products Selector Guide |
| SG417/D | Semiconductor Products for Wireless Communications |

## RF

| | |
|---|---|
| AN535/D | Phase-Locked Loop Design Fundamentals |
| AN779/D | Low-Distortion 1.6 to 30MHz SSB Driver Designs |
| AN1028/D | 35/50 Watt Broadband (160-240MHz) Push-Pull TV Amplifier Band III |
| AN1526/D | RF Power Device Impedances: Practical Considerations |
| *AN1532/D | Using the Motorola MRFIC1806/1807 Dual Demonstration Board |
| AN1539/D | An IF Communication Circuit Tutorial |
| AN1580/D | Mounting and Soldering Recommendations for the Motorola Power Flat Pack Package |
| AN1599/D | Power Control with the MRFIC0913 GaAs Integrated Power Amplifier and MC33169 Support IC |
| AN1602/D | 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard Motorola RFICs |
| AN1610/D | Using Motorola's MRFIC1502 in Global Positioning System Receivers |
| AN1617/D | Mounting Recommendations for Copper Tungsten Flanged Transistors |
| *AN1639/D | Phase Noise Measurement Using the Phase Lock Technique |
| *AN1643/D | RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit |
| AN1670/D | 60 watts, GSM 900MHz, LDMOS Two-Stage Amplifier |
| AN1673/D | Solder Reflow Mounting Method for the MRF286 and Similar Packages |
| AN1675/D | A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low Noise Transistor |
| AN1676/D | A Cascade 2 Stage Low Noise Amplifier Using the MRF1047T1 Low Noise Transistor |
| *AN1687/D | A Full-Featured Wireless Interface for RS-232 Communications |
| *AN1691/D | Practical Solutions for Medium Data Rate Wireless Communications |
| *AN1696/D | Broadband Intermodulation Performance Development Using Rohde & Schwarz Vector Network Analyzer ZVR |
| AR179/D | RF Power Transistors Catapult into High-Power Systems |
| AR333/D | RF Modems Simplified |
| AR347/D | A Compact 1kW 2-50MHz Solid-State Linear Amplifier |
| *AR366/D | Motorola Cellular DSP Does It All |
| AR510/D | VSWR Protection of Solid State RF Power Transistors |
| AR597/D | GaAs RF ICs Target 2.4GHz Frequency Band |
| AR606/D | PCS and RF Components |
| AR612/D | Plastic Packages Hold Power RF MOSFETs |
| AR614/D | Advantages of LDMOS in High Power Linear Amplification |
| *AR624/D | Aluminum-Based Metallization Enhances Device Reliability |
| *AR625/D | SiGe HBTs Drive biCMOS Deeper into RF's Realm |
| *AR626/D | SiGe Gets Real |
| *AR627/D | Motorola Pushes for the Handset Market |
| *AR628/D | Impedance Measurements for High Power RF Transistors Using the TRL Method |
| *AR629/D | Digital Predistortion Techniques for RF Power Amplifiers with CDMA Applications |
| EB90/D | Low-Cost VHF Amplifier Has Broadband Performance |
| EB93/D | 60 Watt VHF Amplifier Uses Splitting/Combining Techniques |
| EB107/D | Mounting Considerations for Motorola RF Power Modules |
| EB109/D | Low Cost UHF Device Gives Broadband Performance at 3.0 Watts Output |
| EB209/D | Mounting Method for RF Power Leadless Surface Mount Transistors |
| *EB211/D | Thermal Management and Solder Mounting Method for the MRF286, 60 Watt Power Device in a CuW (Copper Tungsten) Base Package |

***Additional information relevant to RF may be found in the following Motorola documents:***

| | |
|---|---|
| BR1443/D | Communications – State-of-the-Art is Never Stationary |
| BR1444/D | Communications – 1994 Motorola Resource Guide |
| * BR1466/D | The New Clock Driver Generation |
| BR1467/D | Extend Your Scope in Wireless Systems – The New Hipercomm Generation |
| * BR1502/D | Wireless Infrastructure Solutions |
| * BR1504/D | RF Power |
| * BR1782/D | Networking and Computing Solutions from Motorola |
| BR3006/D | Wireless Communications Resource Guide |
| BR3023/D | In Step With Your Success (RF Semiconductor Division) |

**RF** continued

| | |
|---|---|
| DL110/D | RF Device Data |
| HB215/D | RF Application Reports |
| HB219/D | Introduction to the Oncore ChipSet |
| SG46/D | RF Products Selector Guide |
| SG382/D | Motorola RF CATV Distribution Amplifiers |
| SG384/D | Motorola RF LDMOS Product Family |
| SG417/D | Semiconductor Products for Wireless Communications |

## Smart Card/Conditional Access

**see also Microprocessors: 8-bit MPU/MCU**

***Information relevant to Smart Card/Conditional Access may be found in the following Motorola documents:***

| | |
|---|---|
| BR1734/D | Smart Chip: The Smartcard brain at your Fingertips |

## Software & Programming

| | |
|---|---|
| AN427/D | MC68HC11 EEPROM Error Correction Algorithms in C |
| AN499/D | Let the MC68HC705 Program Itself |
| AN974/D | MC68HC11 Floating-Point Package |
| AN1010/D | MC68HC11 EEPROM Programming from a Personal Computer |
| AN1060/D | MC68HC11 Bootstrap Mode |
| AN1064/D | Use of Stack Simplifies M68HC11 Programming |
| AN1200/D | Configuring the M68300 Family Time Processing Unit (TPU) |
| AN1215/D | PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers |
| AN1218/D | HC05 to HC08 Optimization |
| AN1219/D | M68HC08 Integer Math Routines |
| AN1224/D | Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8 |
| AN1230/D | A Background Debugging Mode Driver Package for Modular Microcontrollers |
| AN1255/D | MC68F333 Flash EEPROM Programming Utilities |
| AN1262/D | Simple Real-Time Kernels for M68HC05 Microcontrollers |
| AN1263/D | Designing for Electromagnetic Compatibility with Single-Chip Microcontrollers |
| AN1264/D | JTAG Flash Memory Programmer |
| AN1280/D | Using and Extending D–Bug 12 Routines |
| AN1280A/D | Using the Callable Routines in D-Bug12 |
| AN1283/D | Transporting M68HC11 Code to M68HC16 Devices |
| AN1284/D | Transporting M68HC11 Code to M68HC12 Devices |
| AN1287/D | MC68HC708LN56 LCD Utilities |
| AN1291/D | Avoiding Multiprocessing Paradoxes with the PowerPC 604 Microprocessor |
| AN1667/D | Software SCI Implementation to the MISC Communication Protocol |
| AN1711/D | DMA08 Systems Compatibilities |
| AN1716/D | Using M68HC12 Indexed Indirect Addressing |
| AN1718/D | A Serial Bootloader for Reprogramming the MC68HC912B32 Flash EEPROM |
| AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| AN1732/D | A Universal Serial Bus Gamepad Device using the MC68HC05JB2 |
| AN1738/D | Instruction Cycle Timing of MC68HC05JJ/JP Series Microcontrollers |
| AN1741/D | In-Circuit and Emulation Considerations for MC68HC05JJ/JP Series Microcontrollers |
| AN1742/D | Programming the 68HC705J1A In-Circuit |
| AN1752/D | Data Structures for 8-bit Microcontrollers |
| *AN1765/D | Ethernet Configuration for a ColdFire Evaluation Board to Download Files from a Personal Computer |
| AN1772/D | Efficient Compilation of Bit-Exact Applications for DSP563xx |
| *AN1800/D | Programming the Thermal Assist Unit in the MPC750 Microprocessor |
| *AN1810/D | Developing DSP56364 Software Using the DSP56362 EVM |
| *AN1831/D | Using MC68HC908 On-Chip FLASH Programming Routines |
| *AN-HK-23/H | MC6805R3, MC68HC05SR3 Technical Comparison |
| ANE425/D | Use of the MC68HC68T1 RTC with M6805 Microprocessors |
| APR30/D | DSP56300 Assembly Code Development Using the Motorola Toolsets |
| APR33/D | ROM Software Patching on the Motorola DSP56304 |
| APR35/D | Designing Motorola DSP56xxx Software for Nonrealtime Tests File I/O Using SIM56xxx and ADS56xxx |
| AR362/D | Whipping Up Real-Time Designs – Programming Motorola's TPU |
| *AR363/D | Programming PowerPC Embedded Applications |
| EB166/D | System Design Considerations: Converting from the MC68HC805B6 to the MC68HC705B16 Microcontroller |
| EB183/D | Erasing and Programming the FLASH EEPROM on the MC68HC912B32 |
| EB191/D | Programming EPROM and EEPROM on the M68HC11EVM |
| EB252/D | MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers |
| EB253/D | How to Use the Lookup and Interpolate Instruction on the CPU32 |
| EB263/D | How to Program Chip Selects on Modular Microcontrollers with a System Integration Module or a Single-Chip Integration Module |

| | |
|---|---|
| EB264/D | Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the Single-Chip Integration Module or System Integration Module |
| *EB267/D | The Double Bus Fault Monitor |
| EB268/D | Starting and Stopping the Time Processor Clock Using the Background Debug Mode |
| EB277/D | Coherency in the Time Processor Unit (TPU) |
| EB287/D | C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0 |
| *EB292/D | Initialization Considerations when Moving from the BUFFALO Monitor to a Standalone MC68HC11 |
| *EB293/D | Simplify MC68HC711E20 EPROM Programming with PCbug11 |
| *EB295/D | Programming the EEPROM on the MC68HC811E2 with the M68HC11EVM Board |
| *EB296/D | Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVBU |
| EB301/D | Programming EEPROM on the MC68HC811E2 During Program Execution |
| EB306/D | Using Exercise 7 on the M68HC16Z1EVB and the Necessity of Word Alignment |
| EB309/D | Using Exercise 8 on the MC68HC16Z1EVB |
| *EB310/D | Using Bus Error Stack Frames to Diagnose CPU32 Released Write Faults |
| *EB314/D | Updating the Software Watchdog on M683xx and MC68HC16 Products |
| EB410/D | PASM05 to INTROL M68HC05 Assembler Conversion |
| EB419/D | ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker |
| EB422/D | Enhanced M68HC11 Bootstrap Mode |
| TPUPN00/D | Using the TPU Function Library and TPU Emulation Mode |
| TPUPN02/D | Fast Quadrature Decode TPU Function (FQD) |
| TPUPN03/D | Frequency Measurement TPU Function (FQM) |
| TPUPN04/D | Table Stepper Motor TPU Function (TSM) |
| TPUPN05/D | Multichannel PWM TPU Function (MCPWM) |
| TPUPN06/D | Programmable Time Accumulator TPU Function (PTA) |
| TPUPN07/D | Asynchronous Serial Interface TPU Function (UART) |
| TPUPN08/D | New Input Capture/Input Transition Counter TPU Function (NITC) |
| TPUPN09/D | Multiphase Motor Commutation TPU Function (COMM) |
| TPUPN10/D | Hall Effect Decode TPU Function (HALLD) |
| TPUPN11/D | Period/Pulse Width Accumulator TPU Function (PPWA) |
| TPUPN12/D | Output Compare TPU Function (OC) |
| TPUPN13/D | Stepper Motor TPU Function (SM) |
| TPUPN14/D | Position-Synchronised Pulse Generator (PSP) |
| TPUPN15A/D | Period Measurement with Additional Transition Detection TPU Function (PMA) |
| TPUPN15B/D | Period Measurement with Missing Transition Detection TPU Function (PMM) |
| TPUPN17/D | Pulse Width Modulation TPU Function (PWM) |
| TPUPN18/D | Discrete Input/Output TPU Function (DIO) |
| TPUPN19/D | Synchronized Pulse-Width Modulation (SPWM) |
| TPUPN20/D | Quadrature Decode TPU Function (QDEC) |

***Additional information relevant to Software & Programming may be found in the following Motorola documents:***

| | |
|---|---|
| BR748/D | M68HC711D3PGMR Programmer Board |
| BR1116/D | Advanced Microcontroller Division Literature Guide |
| BR1155/D | MPC500 Family: Software Development Tools |
| BR1165/D | MPC500 Family: RTEK Real-Time Embedded Kernel |
| CPU12RG/D | CPU12 Reference Guide |
| EMDVPOC/D | Embedded Developer Pocket Guide |
| M68HC08RG/AD | HC08 Family Reference Guide |
| M6809PM/AD | MC6809-MC6809E Microprocessor Programming Manual (1981) |
| M68000PM/AD | M68000 Family Programmer's Reference Manual |
| M68000UM/AD | M68000 8-/16-/32-bit Microprocessors User's Manual, Ninth Edition |
| MC68HC11A8RG/AD | MC68HC11A8 Programming Reference Guide |
| MC68HC11C0RG/AD | MC68HC11C0 Programming Reference Guide |
| MC68HC11D3RG/AD | MC68HC11D3/MC68HC711D3 Programming Reference Guide |
| MC68HC11ERG/AD | MC68HC11E Programming Reference Guide |
| MC68HC11F1RG/AD | MC68HC11F1 Programming Reference Guide |
| MC68HC11K4RG/AD | MC68HC11K4/MC68HC711K4 Programming Reference Guide |
| MC68HC11L6RG/AD | MC68HCL6/MC68HC711L6 Programming Reference Guide |
| MC68HC11MRG/AD | M68HC11 M Series Programming Reference Guide |
| MC68HC11NRG/AD | MC68HC11N Series Programming Reference Guide |
| MCUASM/D | MCUasm Assembly Language Development Toolset |
| MCUDEVTLDIR/D | Motorola Microcontroller Development Tools Directory |
| MPCPRG/D | PowerPC Microprocessor Family: The Programmer's Reference Guide |
| RCPURM/AD | MPC500 Family: RCPU Reference Manual |
| SG146/D | Digital Signal Processors Update |
| SIURM/AD | MPC500 Family: System Integration Unit Reference Manual |
| TPURM/AD | M68300 Family Time Processor Unit Reference Manual |

## Telecommunications

### see also Interfacing

| | |
|---|---|
| AN948/D | Data Multiplexing Using the Universal Digital Loop Transceiver and the Data Set Interface |
| AN1054/D | ISDN System Development Using MC145490EVK/ MC145491EVK Development Kits |
| AN1231/D | Plastic Ball Grid Array (PBGA) |
| AN1241/D | Interfacing the MC68HC705J1A to 9356/9366 EEPROMs |
| AN1254/D | Using the MC68HC16Z1 for Audio Tone Generation |
| AN1274/D | HC08 SCI Operation with Various Input Clocks |
| AN1296/D | Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card |

**Telecommunications** continued

| | |
|---|---|
| AN1299/D | ATM Switch with Shared Memory – A Simple Model |
| *AN1532/D | Using the Motorola MRFIC1806/1807 Dual Demonstration Board |
| AN1599/D | Power Control with the MRFIC0913 GaAs Integrated Power Amplifier and MC33169 Support IC |
| AN1602/D | 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard Motorola RFICs |
| AN1612/D | Shock and Mute Pager Applications Using Accelerometer |
| *AN1643/D | RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit |
| AN1670/D | 60 watts, GSM 900MHz, LDMOS Two-Stage Amplifier |
| AN1675/D | A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low Noise Transistor |
| AN1704/D | Switch Fabric Implementation Using Shared Memory |
| AN1724/D | Implementing SCI Receive and Transmit Buffers in C |
| AN1733/D | Implementing Caller ID Functionality in MC68HC(7)05 Applications |
| AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| AN1771/D | Precision Sine-Wave Tone Synthesis Using 8-bit MCUs |
| AN1772/D | Efficient Compilation of Bit-Exact Applications for DSP563xx |
| *AN1818/D | Software SCI Routines with the 16-bit Timer Module |
| AN-HK-08/H | A Medium Scale PABX |
| AN-HK-12/H | MC68HC05F6 Tone Pulse Dialer |
| AN-HK-17/H | MC68HC05F2 DTMF Output Low Voltage Active Filter |
| *AN-HK-22/H | MC68HC05SR3, MC68HC705SR3 Design Notes |
| APR1/D | Digital Sine-Wave Synthesis Using the DSP56001/DSP56002 |
| APR9/D | Full-Duplex 32 kbit/s CCITT ADPCM Speech Coding on the Motorola DSP56001 |
| APR12/D | Twin CODEC Expansion Board for the DSP56000 Application Development System |
| APR14/D | Conference Bridging in the Digital Telecomms Environment Using the Motorola DSP56000 |
| APR34/D | MC68328 Microprocessor Application: FLEX Alphanumeric Chip MC68175 Interface for One-Way Pager |
| APR37/D | Implementing AC-link with ESAI |
| APR39/D | Programming the DSP56307 Enhanced Filter Coprocessor (EFCOP) |
| APR40/D | Implementing Viterbi Decoder Using the VSL Instruction on DSP Families DSP56300 and DSP56600 |
| *AR366/D | Motorola Cellular DSP Does It All |
| AR606/D | PCS and RF Components |
| *AR627/D | Motorola Pushes for the Handset Market |
| *EB198/D | Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11 |
| TPUPN07/D | Asynchronous Serial Interface TPU Function (UART) |

***Additional information relevant to Telecommunications may be found in the following Motorola documents:***

| | |
|---|---|
| BR484/D | 68302 |
| BR488/D | 68306 68307 68322 |
| BR489/D | 68360 Quad Integrated Communications Controller (QUICC) |
| BR1116/D | Advanced Microcontroller Division Literature Guide |
| BR1133/D | 68K and ColdFire Family Product Portfolio Overview |
| *BR1178/D | ADSL |
| BR1195/D | VeComp: Vector Communications Processors – Technology Overview |
| BR1196/D | CODEC. Plug In. WorldWide. |
| BR1443/D | Communications – State-of-the-Art is Never Stationary |
| BR1444/D | Communications – 1994 Motorola Resource Guide |
| BR1467/D | Extend Your Scope in Wireless Systems – The New Hipercomm Generation |
| *BR1475/D | Datacomms |
| *BR1502/D | Wireless Infrastructure Solutions |
| BR1702/D | Fast Static RAMS and The Communications Market |
| *BR1712S/D | The EL1501C ADSL Line Driver from Elantec Semiconductor |
| BR1729/D | MC92500 Asynchronous Transfer Mode Cell Processors |
| BR1730/D | Qorus Video Conferencing: Where the Quality and Value are Clear to See |
| BR1731/D | Integrated Solutions for ATM Systems |
| BR1752/D | Qorus Development Kit: Get a Clear Picture of What Qorus Video Conferencing Technology Can Do |
| BR1753/D | Motorola Analog Modem Systems |
| BR1754/D | External/Embedded Modem Chip Set and Software |
| BR1755/D | ISA Controller-less Modem Chip Set and Software |
| BR1756/D | PCI Controller-less Modem Chip Set and Software |
| *BR1757/D | Multichannel Infrastructure Modem and Evaluation Kit |
| *BR1782/D | Networking and Computing Solutions from Motorola |
| *BR1788/D | MC145660 CopperGold ADSL Transceiver |
| BR3006/D | Wireless Communications Resource Guide |
| BR3020/D | Remote Access: ISDN Solutions Kit |
| *DSP56LF812UM/AD | DSP56LF812 16-bit Digital Signal Processor User's Manual |
| DSP56302UM/AD | DSP56302 User's Manual |
| DSP56303UM/AD | DSP56303 User's Manual |
| *DSP56307UM/D | DSP56307 24-bit Digital Signal Processor User's Manual |
| *DSP56309UM/D | DSP56309 24-bit Digital Signal Processor User's Manual |
| *DSP56311UM/D | DSP56311 24-bit Digital Signal Processor User's Manual |
| *DSP56600FM/AD | DSP56600 16-bit Digital Signal Processor Family Manual |
| DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |

| | |
|---|---|
| * DSP56824UM/AD | DSP56824 16-bit Digital Signal Processor User's Manual |
| HC05CT4GRS/D | MC68HC05CT4 General Release Specification |
| HC05PL4GRS/H | MC68HC05PL4A, MC68HC05PL4B, MC68HC705PL4B General Release Specification |
| HC705CT4GRS/D | MC68HC705CT4 General Release Specification |
| MC68EN302RM/AD | MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual (Supplement to MC68302UM/AD) |
| MC68LC302RM/AD | MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual |
| MC68PM302RM/AD | Integrated Multiprotocol Processor with PCMCIA Interface Reference Manual |
| MC68SC302UM/AD | MC68SC302 Passive ISDN Protocol Engine User's Manual |
| MC68302UM/AD | MC68302 Integrated Multiprotocol Processor User's Manual |
| MC68356UM/AD | MC68356 Signal Processing Communications Engine User's Manual |
| MC68360UM/AD | MC68360 Quad Integrated Communications Controller User's Manual |
| MC68605UM/AD | MC68605 X.25 Protocol Controller User's Manual |
| MC92501UM/D | MC92501 ATM Cell Processor User's Manual |
| * MPC801UM/AD | MPC801 Integrated Microprocessor for Embedded Systems User's Manual |
| MPC821UM/AD | MPC821 PowerPC Portable Systems Microprocessor User's Manual |
| * MPC850UM/D | MPC850 Integrated Communications Microprocessor User's Manual |
| MPC860UM/AD | MPC860 PowerQUICC User's Manual |
| * MPC8240UM/D | MPC8240 Integrated Processor User's Manual |
| * MPC8260UM/D | MPC8260 PowerQUICC II User's Manual |
| QMCSUPPLEMENT/AD | MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals |
| SG46/D | RF Products Selector Guide |
| SG171/D | Fast Static RAM Division Product Update |
| SG175/D | Networking Systems Division and Personal Computing Division: Product Information |
| SG417/D | Semiconductor Products for Wireless Communications |

## TV and Video

| | |
|---|---|
| AN492/D | A Video Display Board for CD-i Development |
| AN1028/D | 35/50 Watt Broadband (160-240MHz) Push-Pull TV Amplifier Band III |
| AN1235/D | A Set Top Closed-Caption Decoder |
| AN1241/D | Interfacing the MC68HC705J1A to 9356/9366 EEPROMs |
| AN1730/D | Digital Amplification of an Analog Signal Using the MC68HC705J1A |
| AN1755/D | Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM |
| AR333/D | RF Modems Simplified |

***Additional information relevant to TV and Video may be found in the following Motorola documents:***

| | |
|---|---|
| BR1730/D | Qorus Video Conferencing: Where the Quality and Value are Clear to See |
| BR1752/D | Qorus Development Kit: Get a Clear Picture of What Qorus Video Conferencing Technology Can Do |
| * BR1771/D | Streamaster – Digital Imagination: Let Your Mind Flow |
| * BR1803/D | Digital Audio/DTV Solutions |
| * DSP56011UM/AD | DSP56011 24-bit Digital Signal Processor User's Manual |
| * DSP56307UM/D | DSP56307 24-bit Digital Signal Processor User's Manual |
| * DSP56309UM/D | DSP56309 24-bit Digital Signal Processor User's Manual |
| * DSP56311UM/D | DSP56311 24-bit Digital Signal Processor User's Manual |
| DSP56800WP1/D | Novel Digital Signal Processing Architecture with Microcontroller Features |
| HC05RC18GRS/D | MC68HC05RC9/MC68HC05RC18 General Release Specification |
| HC68VBIGRS/D | MC68HC68VBI General Release Specification |
| SG46/D | RF Products Selector Guide |
| SG382/D | Motorola RF CATV Distribution Amplifiers |

## All Products and Application Areas

| | |
|---|---|
| BR101/D | Technical Literature and Information Catalog |
| * BR319/D | Motorola Information Management Services |
| BR380/D | SPS Bar Code Label Specifications |
| BR474/D | European Bar Code Specifications |
| BR481/D | Setting New Standards for Quality and Technical Excellence in Everything We Do |
| BR1202/D | Motorola Quality System Review Guidelines |
| BR1410/D | MAP – Metric Awareness Program |
| BR1460/D | Combinational Technologies |
| BR1494/D | Semiconductor Sales and Product Training Solutions Self Study Guide |
| * BR1514/D | Devices Not Recommended for New Design or On Lifetime Buy |
| * BR3026S/D | Motorola Diversity |
| SG73/D | Master Selection Guide |
| SG379/D | North America Sales Price List |
| * SHIPTRKUM/D | Shipment Tracking System User Manual |

# Applications Documents

## Abstracts

### MC68HC11 EEPROM Error Correction Algorithms in C

A modified Hamming code is used to correct one-bit errors and detect two-bit errors in data blocks of up to 11 bits – avoiding the problem of erroneous correction of two-bit errors. The technique is implemented entirely in 'C', and additional functions are provided to program and read MC68HC11 EEPROM using the encoding/decoding algorithms.

**Order by: AN427/D**

---

### 128K byte Addressing with the M68HC11

The 64K byte direct addressing capability of the M68HC11 family is insufficient for some applications. This note describes two methods of memory paging – one software only, the other hardware plus software – that allow the MCU to address a 1Mbit EPROM (128K bytes) by manipulation of the address lines. The two methods illustrate the concept of paging and the inherent compromises; the technique may be expanded to other memory combinations. Includes full software listings.

**Order by: AN432/D**

---

### Driving LCDs with M6805 Microprocessors

The MC68HC05L series of MCUs include circuitry for direct LCD drive. Other MCUs in the M6805 and M68HC05 families have a variety of I/O and display drive capabilities. This comprehensive note describes alternative LCD drive arrangements for applications with different numbers of backplanes and display drive capabilities, including software-based and display driver chip solutions. Circuits and software listings are provided. The techniques apply equally to other MCU families such as the M6801 and M68HC11.

**Order by: AN442/D**

---

### An Introduction to the HC16 for HC11 Users

The HC16 provides a software upgrade path for HC11 users while providing full hardware compatibility with the asynchronous address and data buses used in Motorola's 32-bit MPUs. It is a highly modular family based on the CPU16 core, a true 16-bit design with an architecture that will be very familiar to HC11 users. However many of its features are new, and this document explains the differences that will be encountered by an experiences HC11 user moving to the HC16. It covers CPU architecture, software compatibility and HC16 hardware.

**Order by: AN461/D**

---

### Software Driver Routines for the Motorola MC68HC05 CAN Module

The Controller Area Network (CAN) protocol describes a serial communications protocol for interrupt-driven, real-time control applications, primarily in the automotive sector. This note describes driver routines which provide an interface between application software in the MCU ROM and the CAN module. The routines allow for the initialisation of the module, the transmission of messages previously stored in RAM, and the automatic handling of received messages. They have been written to run on the MC68HC05X4 but can easily be adapted to run on any M68HC05 MCU supporting the CAN protocol.

**Order by: AN464/D**

---

### Secure Remote Control using the 68HC05K1 and the 68HC05P3

This application note shows how the 68HC05K1 and the 68HC05P3 can be used together to form a multi-user secure remote-control system. Every time the single key on the transmitter is pressed it sends an infrared signal to the receiver; in order to make the system secure against attempts to capture the signal and retransmit it later for unauthorised purposes, the transmitter transmits a different signal every transmission. The system is especially suitable for enabling or disabling car alarms and for operating car central locking. Includes software code for both MCUs.

**Order by: AN465/D**

---

### A Minimum Evaluation System for the MC68331 and MC68332

The MC68331 and MC68332 are based on the 68000-compatible, CPU32 microprocessor core coupled with highly functional on-chip peripheral modules. This note describes the design of a minimum evaluation system based on either device. The system is intended

to be a low cost method of evaluation and also to be a starting point for engineers wishing to implement a development interface for their own designs. The design takes advantage of Background Debug Mode (BDM), a new development feature implemented on CPU32-based microprocessors.

**Order by: AN473/D**

## CPU16 and the Configurable Timer Module (CTM) in Engine Control

Special timer modules have been developed to simplify the task of controlling ever more complex engines within the constraints of ever tightening emissions regulations. The configurable timer module (CTM) was developed for automotive applications which require flexibility and high performance from an M68HC16 family MCU, plus the ability to be designed very quickly for a customer's specific requirements. This note describes the use of the CTM in a hypothetical engine management system, and provides demonstration code for the MC68HC16W1.

**Order by: AN476/D**

## Simple A/D for MCUs without Built-In A/D Converters

Non-critical resistance measurement is often needed, for example in temperature, light, pressure and position measurements using devices where the sensor is a variable resistance. Such measurements can be made at minimal cost using existing MCUs if a simple analog-to-digital converter is added. This application note describes a method of measuring an unknown resistance with an M68HC05 family MCU that does not have a built-in A/D converter. Both the theoretical and the practical aspects of the method are covered. The MCU used in the example is the MC68HC705J2. Includes a software listing for a practical application.

**Order by: AN477/D**

## Low Cost MPC601 EVM

Contains a detailed design for a low cost Evaluation Module (EVM) for the MPC601 PowerPC microprocessor. It includes a summary of the EVM features, a detailed description of the design with block diagrams and state diagrams, hardware schematics, listings for the PLDs, timing diagrams, memory map and PCB component layout.

**Order by: AN486/D**

## A Video Display Board for CD-i Development

Aimed mainly at the consumer market, CD-i allows users to interact with high quality audio-visual presentations for entertainment, education or business. The worldwide CD-i standard – known as the Green Book – defines the system hardware, software and encoding methods. There are two CD-i video standards: MPEG ISO-11172, and the Green Book 'Base Case' standard which defines the minimum functionality required of a player. This note provides a description, circuit diagrams and software listings for a Base Case Video board forming part of Motorola's CD-i Development System (MCDS). MCDS is a multi-board system based on the VME double-extended Eurocard specification.

**Order by: AN492/D**

## RDS Decoding for an HC11-Controlled Radio

This note describes the RDS (Radio Data System) aspects of the HC11 radio controller described in AN494/D. RDS adds a digital data capability to the FM VHF transmissions on band II (87.5 to 108MHz), as defined in CENELEC EN 50067. This application can use EON (Enhanced Other Networks) to retune the radio when a traffic announcement is taking place on another frequency. Provides an overview of the RDS features, a description of the RDS software and the way that it handles incoming data, and a full listing of the RDS and date calculation software modules (the main radio control software is listed in AN494/D).

**Order by: AN495/D**

## Let the MC68HC705 Program Itself

Programming the EPROM versions of Motorola's MCUs is normally achieved using an on-chip program stored in ROM; this program is executed in 'bootloader mode'. There are, however, occasions when it would be useful to custom program all or part of the EPROM in the normal user mode – to add program routines; to store serial numbers, calibration values or information about external equipment; or to remove test programs. This note describes the normal method of programming, presents hardware and software that allows the MCU to program itself, and discusses some ideas for enhancing the software. Includes full source code for the software described.

**Order by: AN499/D**

## Phase-Locked Loop Design Fundamentals

The fundamental design concepts for phase-locked loops implemented with integrated circuits are outlined. The necessary equations required to evaluate the basic loop performance are given in conjunction with a brief design example.

**Order by: AN535/D**

## Low-Distortion 1.6 to 30MHz SSB Driver Designs

A general discussion for broadband drivers and their requirements for linear operation. Design examples are given using Motorola plastic transistors and high-gain hybrid modules designed for operation in the 1.0 to 250MHz range. The amplifiers range in power gain from 25 to 55dB and are capable of driving power amplifiers to levels up to several hundred watts.

**Order by: AN779/D**

## Self-Programming the MC68701 and the MC68701U4

The MC68701 and MC68701U4 are EPROM versions of the M6801 microcomputer family. The MC68701 on-chip resources include a 2 kByte EPROM, a three-function timer, a serial communication interface (SCI), up to 29 parallel lines, 128 bytes of RAM and an oscillator. These resources give it extensive power and flexibility for ease of design. The enhanced features of the MC68701U4 include a 4 kByte EPROM, two input capture functions, three output compare functions, a counter alternate address and 192 bytes of RAM.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AN906A/D**

---

## Compensating for Nonlinearity in the MPX10 Series Pressure Transducer

This application note describes a technique to improve the linearity of Motorola's MPX10 series pressure transducers when they are interfaced to a microprocessor system. The linearization technique allows the user to obtain both high sensitivity and good linearity in a cost-effective system.

**Order by: AN935/D**

---

## Mounting Techniques, Lead Forming and Testing of Motorola's MPX Series Pressure Transducers

This document discusses assembly and testing techniques for Motorola MPX series pressure sensors in the chip-carrier package. Several design ideas are offered for pressure sensing applications.

**Order by: AN936/D**

---

## Data Multiplexing Using the Universal Digital Loop Transceiver and the Data Set Interface

This application note will describe the design of a short-haul multiplexer for asynchronous data at rates up to 9600 baud. The mux combines eight full-duplex data channels along with eight end-to-end RS–232 control signals onto a single pair of telephone wires for distances up to 2km.

**Order by: AN948/D**

---

## MC68HC11 Floating-Point Package

While most MC68HC11 applications can be implemented using 16-bit integer precision, certain algorithms may be difficult or impossible without floating-point. This application note details an efficient floating-point package that includes basic trig functions and square root in addition to add, subtract, multiply and divide. It requires just over 2k bytes of memory, with only 10 bytes of page zero RAM in addition to stack RAM.

**Order by: AN974/D**

## Using the Serial Peripheral Interface to Communicate Between Multiple Microcomputers

Communication between multiple processors can be difficult when different types are used. One solution is the SPI, an interface intended for communication between ICs on the same board. It can be implemented in software, allowing communication between two MCUs where one has SPI hardware and the other does not. Costly expansion buses and UARTs are eliminated. The scheme is illustrated with a temperature/time display circuit using an MC68HC05C4 and an MC68705R3.

**Order by: AN991/D**

---

## CONFIG Register Issues Concerning the M68HC11 Family

The M68HC11 is the first MCU family to offer semi-permanent configuration options, by means of the CONFIG register. Users experienced some initial difficulties, due partly to problems with early mask sets, partly to a lack of understanding about this new function. The mask problems are now resolved; this note explains the early fault mechanisms, and presents important application guidelines to ensure proper operation.

**Order by: AN997/D**

---

## MC68HC11 EEPROM Programming from a Personal Computer

Describes a simple and reliable method of programming the MC68HC11's internal EEPROM (or EEPROM connected to its external bus) by downloading data in Motorola S-record format from a standard personal computer (PC) fitted with a serial communications port. Includes BASIC program for the PC (to Program External EEPROM/RAM, Program Internal EEPROM, or Verify internal or External EEPROM/RAM) and the source listing of MC68HC11 code for downloading to RAM to receive S records.

**Order by: AN1010/D**

---

## 35/50 Watt Broadband (160-240MHz) Push-Pull TV Amplifier Band III

The main design aim for this broadband ultra-linear push-pull amplifier was to keep the design as simple as possible, in order to obtain the best performance from the two TPV375 transistors and to minimise the cost. A further target was to obtain the maximum gain by reducing input matching circuit losses. Includes circuit, background description, Smith charts and PCB layout.

**Order by: AN1028/D**

---

## Designing for Electromagnetic Compatibility (EMC) with HCMOS Microcontrollers

As the operating speeds of the latest HCMOS devices increase, the MCU system designer must take more account of the electro-magnetic compatibility (EMC) of the finished product. This discussion relates mainly to emission control, but most of the techniques also reduce electromagnetic susceptibility. Subjects include Legal

Requirements, RFI Problems, types of radiation, Supply Decoupling, Grounding Techniques and PCB Layouts. Incorporates an article reprint from EMC Technology describing an EMI/RFI diagnostic probe.

**Order by: AN1050/D**

## Transmission Line Effects in PCB Applications

As rise and fall times become faster in order to achieve high operating speeds, transmission line effects on PCBs can be very significant, with the possibility of unpredictable behavior. This note presents a guideline as to when to analyse, discusses the characteristics of different types of PCB trace, describes Lattice Diagram and Bergeron Plot analysis, and summarises termination methods. Includes 10 worked examples.

**Order by: AN1051/D**

## ISDN System Development Using MC145490EVK/ MC145491EVK Development Kits

An introduction to the design of an ISDN (Integrated Services Digital Network) Terminal Adapter based on the MC145490/91 Evaluation Kit. Overviews the board and the software required to implement its features. Provides a detailed description of each element of the system, plus example software written in a Pascal-like pseudo-code language. Includes an introduction to the ISDN concept and protocols.

**Order by: AN1054/D**

## Selecting the Right Microcontroller Unit

Selecting the proper MCU for an application is one of the critical decisions which can control the success or failure of the project. There are numerous criteria to consider; many of them are presented here along with the thought processes guiding their selection. The reader should attach an appropriate grading scale before evaluating the total and making the correct decision.

**Order by: AN1057/D**

## Reducing A/D Errors in Microcontroller Applications

The MCU with integrated Analogue to Digital Converter provides a highly cost-effective solution for many mixed analogue/digital applications. However, combining a wide bandwidth ADC system on the same die as a high-speed CPU can lead to noise problems in the analogue measurements. This comprehensive note lays down basic system guidelines for the design phase of an MCU-based product, to avoid ADC problems. Includes an examination of a real-world system.

**Order by: AN1058/D**

## MC68HC11 Bootstrap Mode

The M68HC11 Bootstrap Mode allows a user program to be loaded into internal RAM through the Serial Communications Interface (SCI). In addition to operating normally, this program can do anything a factory test program can do since the protected control bits become accessible; Expanded Mode resources are available because the control bits can be changed by the bootstrap program. Although the basic concepts are simple, some subtle implications of this mode need careful consideration, both to avoid problems and to find useful applications. Includes commented listings for selected M68HC11 bootstrap ROMs.

**Order by: AN1060/D**

## Using the QSPI for Analog Data Acquisition

Analogue to digital conversion is required in many MCU applications – it must be fast, accurate and inexpensive. While the MC68332 32-bit Integrated Microcontroller lacks direct A/D capability, an inexpensive solution is achieved using an external A/D Converter interfaced through the QSPI. The hardware and software examples described here use the MC145040 and MC145050 8 and 10-channel A/D converters. The discussion includes design methodology for maximum A/D throughput, simultaneous use of other peripherals with the QSPI and determination of overall system performance.

**Order by: AN1062/D**

## Use of Stack Simplifies M68HC11 Programming

Architectural extensions to the M6800 family built in to the MC68HC11 allow easy manipulation of data on the stack. The CPU uses the stack for subroutine and interrupt return addresses. This note discusses two additional uses – the storage of local variables and subroutine parameter passing – that can simplify programming and debugging. It describes the basic operation of the MC68HC11 stack, the concept of local and global variables, subroutine parameter passing, and the use of the instruction set to achieve the additional uses. Includes example listings illustrating the techniques.

**Order by: AN1064/D**

## Interfacing the MC68HC05C5 SIOP to an $I^2C$ Peripheral

A standard MCU may not have all the peripherals required in a system on chip. The problem can be solved by interfacing the MCU to off-chip peripherals, ideally using a synchronous serial communication port. Unfortunately these peripherals may not have an interface that is compatible with Motorola's simple synchronous Serial I/O Port (SIOP). This note describes how the SIOP on the MC68HC05C5 can be interfaced to an $I^2C$ peripheral, in this case the PCF8573 Clock/Timer. Includes circuit and software listings for a timer/calendar application that can interface with a terminal.

**Order by: AN1066/D**

## Pulse Generation and Detection with Microcontroller Units

MCUs are often required to generate timed output pulses, and to detect and measure input pulses. Output pulses might strobe a display latch, transmit a code or meter an action in a process control system. Input pulses can range from microseconds to hours, and include detecting pushbutton closures, receiving codes or measuring engine rotation. This note describes various methods of generation and detection using several families of Motorola MCUs with differing timer structures. Includes program listings.

**Order by: AN1067/D**

## Simple Design for a 4-20mA Transmitter Interface Using a Motorola Pressure Sensor

Pressure is an important parameter in many industrial applications such as air conditioning, liquid level sensing and flow control. The sensor is often located in a noisy environment and may be several hundred metres from its associated electronic system. If the signal is transmitted as a voltage it is susceptible to electromagnetic interference. If it is transmitted as a current, it is easier to recover at the receiver and is less affected by the length of the transmission line; this note describes a simple but high performance circuit.

**Order by: AN1082/D**

## Calibration-Free Pressure Sensor System

The MPX2000 Series of pressure transducers give an output signal proportional to applied pressure. They are available as both ported and unported assemblies for pressure, vacuum and differential measurement. By using the on-chip A/D converter of the MC68HC05B6 MCU, an accurate, reliable and versatile pressure measurement system can be designed which needs no external calibration.

**Order by: AN1097/D**

## Analog to Digital Converter Resolution Extension Using a Motorola Pressure Sensor

Describes a method of obtaining more than 8 bits of resolution with an 8-bit A/D, when interfacing a Motorola pressure sensor to a microprocessor. The electronic design is relatively simple and uses standard components.

**Order by: AN1100/D**

## A Digital Pressure Gauge Using the Motorola MPX700 Series Differential Pressure Sensor

This solid state digital pressure gauge is built with a Motorola MPX series pressure transducer, instrumentation amplifier and LCD display. Differential, gauge and vacuum readings from 0 to 100 p.s.i with a resolution of 1 p.s.i. can be made with the MPX700 sensor. Full scale readings down to 1 p.s.i. are possible with alternative MPX sensors and displays. Includes circuit diagram, parts list and calibration details.

**Order by: AN1105/D**

## DRAM Interface to the MC88200 M Bus

Describes the design and operation of a 25 to 33.3MHz Dynamic RAM interface to the M Bus of the MC88200 Cache/Memory Management (CMMU). The memory interface is divided into two sub-systems; the CPU sub-system includes one MC88100 CPU and two MC88200 CMMUs, while the memory sub-system contains two non-interleaved 32-bit wide banks of SIMMs with byte parity and a DRAM controller. The DRAM controller is implemented with Programmable Array Logic (PAL), a gate array and discrete hardware. Assumes some knowledge of the M88000 family and DRAMs.

**Order by: AN1125/D**

## Configuring the M68300 Family Time Processing Unit (TPU)

The TPU is a sophisticated embedded peripheral in the M68300 family of 32-bit MCUs handling the time-intensive tasks associated with embedded controllers with minimum processor intervention. Advanced functions incorporated in the TPU microcode may be executed through any of its 16 channels. Communication with the TPU is through dual-port RAM; a number of memory mapped registers must be configured initially, together with some channel parameters. This note discusses basic TPU operation, registers and parameters, with an example application and sample program.

**Order by: AN1200/D**

## The Motorola BurstRAM

Describes the operation of the MCM62486 32K x 9 Synchronous BurstRAM, designed to provide a high performance secondary cache for the Intel i486 microprocessor, and for future processors with burst protocol. Four of these devices can supply a 128 Kbyte direct-mapped bursting cache with parity support.

**Order by: AN1209/D**

## A Protocol Specific Memory for Burstable Fast Cache Memory Applications

Cache memory design has evolved rapidly in recent years, taking full advantage of application-specific fast static RAMs that have become available. Faster processor clock rates, larger on-chip processor caches, larger and faster FSRAMs, more efficient processor bus protocols and more efficient DRAM interfaces have all contributed. This note presents an overview of developments, and describes the operation of a high density, very fast Synchronous SRAM with on-chip burst counter and interface logic for the i486 processor, which is currently under development.

**Order by: AN1210/D**

## 16-bit DSP Servo Control with the MC68HC16Z1

Microcontrollers have come a long way. Once reserved strictly for computer applications, they have steadily encroached on areas previously dominated by analogue technology; closed-loop control systems are among the most recent bastions to fall. This application note discusses digital filter implementation of Proportional, Integral,

Differential (PID) control algorithms. The implementation takes advantage of the control-orientated digital signal processing capabilities of the M68HC16 MCU family.

**Order by: AN1213/D**

## PID Routines for MC68HC11K4 and MC68HC11N4 Microcontrollers

PID (Proportional, Integral, Derivative) compensation is one of the most common forms of closed-loop control, and a growing application area for embedded microprocessors. This note provides two working examples of PID control-loop software. The first, written primarily in C, shows a PID algorithm in a straightforward way using floating-point maths. The second example implements a PID algorithm in assembly language. Both examples are complete and ready to run on a Motorola M68HC11EVS evaluation board.

**Order by: AN1215/D**

## HC05 to HC08 Optimization

**Rev 2**

The HC08 Family is a performance extension to the HC05 Family of low cost MCUs. This application note describes the differences and advantages of the HC08 Family CPU (CPU08), including the new addressing modes, the many new instructions, and the performance improvements to existing instructions that result from the introduction of pipelining. Many examples are given to illustrate the use of the new instructions. Written for the engineering manager and design engineer, and assumes the reader has a background in MCU software and hardware design and is familiar with the HC05.

**Order by: AN1218/D**

## M68HC08 Integer Math Routines

**Rev 1**

The M68HC08 MCU is a fully upward-compatible performance extension of the M68HC05 Family, and users familiar with the M68HC05 should find little difficulty implementing the M68HC08's architectural enhancements. The six integer math subroutines in this application note take advantage of one of the main CPU enhancements – stack relative addressing. Storage space for local variables needed by a subroutine can now be allocated on the stack when a routine is entered and released on exit. These integer math routines are implemented using only 10 bytes of global RAM space.

**Order by: AN1219/D**

## Hamming Error Control Coding Techniques with the HC08 MCU

The MC68HC08 MCU is used here to illustrate the code development precess for error control coding (ECC) in a digital transmission system. A message frame consisting of a 4-bit data field with three parity bits will be encoded to allow the original four bits to be recovered, even if any single bit is corrupted during the transmission and reception processes. This process is based upon the class of linear error-correcting codes called Hamming codes. The process of time diversity is also discussed as a means of controlling burst errors in a transmission system.

**Order by: AN1221/D**

## Arithmetic Waveform Synthesis with the HC05/08 MCUs

Arithmetic Synthesis (AS) produces waveforms using an accumulated value that points to the next output time sample in a table (in contrast to direct digital synthesis where the "distance" between each sample in the table is constant). Given an accumulation constant predefined in memory a very precise waveform can be produced. This application note demonstrates the use of AS to create sinusoidal waveforms using an MCU. It is written for the HC08; although cycle execution time will be different, the program listing for the HC08 is also applicable to the HC05.

**Order by: AN1222/D**

## A Zero Wait State Secondary Cache for Intel's Pentium

In the next generation of desktop computers, first level (L1) on-chip cache memory hit rates will suffer as a result of users' migration from DOS to Windows to Windows NT™. To keep chip size down, the CPU designer can only afford relatively small increases in L1 cache size. Second level (L2) cache must help to avoid time consuming DRAM accesses. This note explains some of the system level, electrical, and timing issues associated with the design of a zero wait state secondary cache for Intel's Pentium, based on Motorola's new families of 64Kx18 and 32Kx18 Fast SRAMs.

**Order by: AN1223/D**

## Example Software Routines for the Message Data Link Controller Module on the MC68HC705V8

The Message Data Link Controller (MDLC) is a communication module designed for use with an automotive serial multiplex bus. It handles all communication duties, including message buffering, bus access and arbitration, and error detection. It interrupts the CPU only when a complete message has been received error-free or following a successful transmission. This note describes a basic set of MDLC driver routines for communicating on an SAE J1850-Class B bus using the MC68HC705V8 MCU. The methods also apply to any Motorola MC68HC05 or MC68HC08 microcontroller that contains the MDLC module.

**Order by: AN1224/D**

## Use of the 68HC705C8A in Place of a 68HC705C8

**Rev 4**

The MC68HC705C8A is an enhanced version of the MC68HC705C8 and is designed as a drop-in replacement. This note describes the small differences between the two devices, including the new customer-requested features of the 'A' version. Includes an example of a 4 x 4 keypad implementation, with software listing.

**Order by: AN1226/D**

## Using 9346 Series Serial EEPROMs with 6805 Series Microcontrollers

Serial EEPROMs have become an inexpensive way of maintaining small amounts of non-volatile data in MCU systems during power off. This note describes how M68HC05 Family microcontrollers can be used with 93x6 series serial EEPROMs made by a number of manufacturers; the series includes base numbers 9346, 9347, 9357, 9366, 9367, 32C101 and 33C102. Includes simple schematics, a list of commonly encountered problems, flow charts for the various functions, and source code listings for three different EEPROM algorithms.

**Order by: AN1227/D**

---

## Interfacing the HC05 MCU to the MC145051 A/D Converter

**Rev 1.1**

The MC145051 is a ratiometric 10-bit A/D converter providing 11 analogue conversion channels with an internal sample and hold. External communication for the channel address and converted data is through a serial interface. This note describes the interface between the A/D converter and an M68HC05 family MCU – in this case the MC68HC705C8 – using the SPI; it includes a schematic, flow chart and software listing. It also describes how to interface the ADC to MCUs which do not have an SPI module, using a software driver to 'bit bang' a port of the MCU; an HC705K1, one of the smallest M68HC05 MCUs, is used as an example.

**Order by: AN1228/D**

---

## A Background Debugging Mode Driver Package for Modular Microcontrollers

Motorola's 16 and 32-bit modular microcontrollers include an operating mode called Background Debugging Mode (BDM). When enabled, BDM allows an external host processor to control a target MCU and to access both memory and I/O devices through a simple serial interface. BDM is a useful feature for initial debugging of both hardware and software, and can also simplify production line testing and configuration. This note shows how to enable and control BDM using an IBM-compatible PC. Drivers and demonstration programs in C are included to allow the rapid implementation of a custom test fixture or debugging facility.

**Order by: AN1230/D**

---

## Plastic Ball Grid Array (PBGA)

**Rev 2**

The Plastic Ball Grid Array (PBGA) package is the industry description of the package sometimes referred to as the Overmolded Pad Array Carrier (OMPAC). Developed by Motorola in the late 1980s for use in space-limited Motorola products such as radios, pagers and cellular telephones, it has grown in popularity and is now adopted by JEDEC and soon by EIAJ. This note provides general information about the package including mechanical data, and discusses its use with surface mount processes. Topics include Package Construction, Motherboard Layout, Surface Mount Assembly, Solder Joint Inspection, Rework and Repair, Solder Joint Reliability and Reliability Stress Tests.

**Order by: AN1231/D**

---

## Thermal Performance of Plastic Ball Grid Array (PBGA) Packages for Next Generation FSRAM Devices

The use of the traditional Theta JA expression to describe the thermal performance of a Plastic Ball Grid Array (PBGA) package obscures its actual performance characteristics. Because the package is thermally closely coupled to its printed circuit board, its thermal performance is dominated by the temperature of the board. The performance is modelled here as the junction to board and junction to case thermal resistances. Practical measurements were taken on the 119 lead PBGA package on a variety of board types in natural and forced convection environments.

**Order by: AN1232/D**

---

## Using M68HC16 Digital Signal Processing to Build an Audio Frequency Analyzer

The MC68HC16Z1 is a high performance 16-bit MCU whose CPU16 instruction set simplifies the use of digital signal processing (DSP) algorithms and makes it easy to implement low-bandwidth filter and control-oriented applications. This tutorial-style application note provides concrete hardware/software applications that are used in the design of an MCU-based Audio Frequency Analyzer using DSP algorithms. It assumes a basic knowledge of the MC68HC16Z1 hardware and the CPU16 instruction set.

**Order by: AN1233/D**

---

## A Set Top Closed-Caption Decoder

Closed captioning is designed principally for the hearing impaired, but also has other applications such as enabling the TV service to be available in noisy environments and assisting the young and illiterate to learn to read. The inclusion of closed caption decoding circuitry is now compulsory in certain US TVs. The note describes the design of a set top decoder based on the MC144143 Closed Caption Decoder IC, including schematic, PCB artwork, component layout, parts list and a full description of its operation.

**Order by: AN1235/D**

---

## Timing Performance of TPU I/O Hardware

Describes the timing relationships between the Time Processor Unit (TPU) I/O pins and the system clock used to drive the TPU. These relationships, rather than event scheduling software latencies, define actual hardware performance. A working example shows how an output pulse from one TPU pin can be triggered by an input edge applied to another pin with no software overhead.

**Order by: AN1236/D**

## Using M68HC11 Microcontrollers with WSI Programmable Peripheral Devices

After system development using M68HC711 family MCUs with EPROM or One Time Programmable ROM, a final design is usually implemented using an equivalent mask-programmed M68HC11 device. However a quick and low-cost alternative method is to use a ROM-less M68HC11 derivative in conjunction with a WSI programmable PSD peripheral. The PSD devices combine EPROM, SRAM, programmable logic for memory map decoding, programmable I/O ports, power management and other capabilities on a single chip. This note describes the process of converting from a prototype design based on an MC68HC711 device to a production design using a low-cost M68HC11 derivative and a WSI PSD.

**Order by: AN1237/D**

---

## HC05 MCU LED Drive Techniques Using the MC68HC705J1A

The MC68HC705J1A has four I/O pins rated to sink 10mA, compared to the 1.6mA of the normal HCMOS I/O pins. This note describes how to use these pins to drive an LED directly, without the need for an external amplifying transistor. The calculations used to determine the value of the current limiting resistor is valid for any HC05 MCU port pin regardless of the pin's output low current rating – only the current specification need be changed in the equations.

**Order by: AN1238/D**

---

## HC05 MCU Keypad Decoding Techniques Using the MC68HC705J1A

A matrix keypad allows a designer to implement a large number of inputs with a small number of port pins. The note demonstrates the use of a matrix keypad with M68HC05 J- and K-series MCUs, including wake-up from STOP mode. The code is divided into a main routine handling STOP mode and interrupt servicing, and two subroutines handling the keypad decode and debouncing. The MC68HC705J1A is used in the example – in common with several other Motorola MCUs it has built-in pulldown resistors required by the keypad.

**Order by: AN1239/D**

---

## HC05 MCU Software-Driven Asynchronous Serial Communication Techniques Using the MC68HC705J1A

The Serial Communication Interface (SCI) is available in many Motorola MCUs, and provides full-duplex, UART communication between the MCU and other devices. It handles all transmission and reception duties and can detect error conditions such as framing errors, noise and overrun. Low cost, low pin count MCUs such as the MC68HC705J1A do not include an SCI. To perform serial communication a software routine must drive I/O port pins to emulate an SCI. This note describes such a routine, which includes noise and framing error detection, in an interface application between the MCU and the MC145407 RS232-C Transmitter/Receiver.

**Order by: AN1240/D**

## Interfacing the MC68HC705J1A to 9356/9366 EEPROMs

The 9356 and 9366 EEPROMs are an industry standard widely used to store non-volatile information in applications such as security systems, telephones, consumer electronics and test equipment. This note describes an interface between the 9356/66 and an MC68HC705J1A MCU. Communication between the MCU and the EEPROM is via a serial interface – since the MC68HC705J1A does not have a serial interface a software driver is used to program an I/O port. With some modification this software will work with other configurations in the 93xx series. Includes schematic, flowchart and program listing.

**Order by: AN1241/D**

---

## High Performance M68HC11 System Design Using the WSI PSD4xx and PSD5xx Families

This application note covers the conversion of a single-chip MC68HC711K4 microcontroller system to a two-chip design using the MC68HC11K1 with a WSI PSD412A1. WSI PSD4xx and PSD5xx programmable system devices (PSDs) are peripherals with flexible bus interfaces that provide microcontroller system designers with an integrated memory solution including SRAM, EPROM and programmable logic. They can often provide a cost-effective solution to problems such as insufficient I/O pins or insufficient memory.

**Order by: AN1242/D**

---

## Output Loading Effects on Fast Static RAMS

As Fast SRAM access times decrease, so do output transition times. With faster rise and fall times come additional problems associated with output and signal path impedances. In any system running at frequencies where the propagation delay of a signal path is greater than one half the total transition time, transmission line effects will be seen on the signal. This results in overshoot and undershoot at the load end of a conductor which can cause problems both in testing and in actual use. This note discusses the factors contributing to these effects and the measures that can be used to predict or eliminate them.

**Order by: AN1243/D**

---

## Brushed DC Motor Control Using the MC68HC16Z1

The MC68HC16Z1 is a 16-bit high performance MCU incorporating a number of on-chip modules. One such module is the General Purpose Timer, which provides several timing functions including Pulse Width Modulation (PWM). This note describes a DC motor control system based on the PWM function. The design monitors the motor speed as a function of shaft rotation period and changes the PWM output duty cycle to maintain constant speed. The interface between the MC68HC16Z1 and the motor is through the DEVB103 board described in AN1300/D (Interfacing Microcomputer to Fractional Horsepower Motors). Includes schematic, flow charts and program listing.

**Order by: AN1249/D**

## Using the MC68HC16Z1 for Audio Tone Generation

In many applications a microcontroller is required to generate audio frequency tones. These may be simple square waves or complex waveforms for computer-generated music. The advantages of using a microcontroller are that hardware costs can be reduced elsewhere as the MCU already exists in the product, and that software offers a very flexible solution. This note examines software techniques for generating an arbitrary waveform, and explains how to generate DTMF tones for use on the telephone network.

**Order by: AN1254/D**

---

## MC68F333 Flash EEPROM Programming Utilities

**Rev 1**

The MC68F333 MCU is a member of the M68300 modular microcontroller family. Two of its modules are flash EEPROM modules (FLASH), one of 16 kbytes, the other 48 kbytes. This note describes software utilities to program and erase the FLASH modules in the MC68F333 – they may be modified and used with other members of the M68300 family containing flash EEPROM. The utilities are drivers for the CPU32 background debugger program BD32 – this allows a simple PC interface to be supported without excessive increase in code size, and enables the MCU to be programmed using only an external programming voltage source.

**Order by: AN1255/D**

---

## Interfacing the HC05 MCU to a Multichannel Digital-to-Analog Converter using the MC68HC705C8A and the MC68HC705J1A

The MAX528/MAX529 (529) digital-to-analog converter by Maxim is an 8-bit, 8-channel DAC with programmable output buffers and a serial interface. This note describes an interface between the 529 and the MC68HC705C8A MCU that uses the serial peripheral interface (SPI). It also demonstrates a software I/O driver that can be used by HC05 family MCUs that do not have an SPI (in this case the MC68HC705J1A). Includes circuits, flowcharts and assembly code listings.

**Order by: AN1256/D**

---

## Using the M68HC05 Family On-Chip Voltage Regulator

The MC68HC705V8 and MC68HC05V7 MCUs are manufactured with a combination of Ultra High Voltage CMOS and bipolar analog technologies, and include an on-chip 5V voltage regulator. This is optimised for operation over the nominal 8-16V automotive range, but will also cope with jump-start and severe transient situations. This note provides an overview of the voltage regulator architecture and describes the external components and software required for correct operation. Includes a circuit example, PCB layout considerations to minimize EMI problems, and software flowcharts for handling initialization and standby mode.

**Order by: AN1257/D**

## System Design and Layout Techniques for Noise Reduction in MCU-Based Systems

Electromagnetic interference (EMI) issues are becoming more of a problem for system designers as semiconductors generally become faster, more integrated and often noisier. However most EMI problems can be avoided in advance by using an appropriate system design approach coupled with proper PCB layout. This note focuses on proven practical layout techniques to control EMI on MCU-based mixed-signal systems. Topics include a brief overview of EMI, general lout guidelines, and a noise reduction checklist.

**Order by: AN1259/D**

---

## Use of 32K x 36 FSRAM in Non-Parity Applications

The MCM69F536 and MCM69P536 are synchronous Fast Static BurstRAMs organized as 32K words of 36 bits. The JEDEC standard pin assignment for synchronous SRAMs defines the corner I/O pins of the 100-pin QFP/TQFP package as either parity I/O or no-connect. The MCM69F536, MCM69P536 and future x36 BurstRAMs can be used in non-parity applications by making one of two design choices, which are described here.

**Order by: AN1261/D**

---

## Simple Real-Time Kernels for M68HC05 Microcontrollers

A kernel is similar to an operating system in that it offers very fast software development and allows new modules to be added without interfering with existing modules. A real-time kernel is easy to debug, and encourages the user to develop software in an organized manner. This note demonstrates the operation of two different types of simple real-time kernel for the M68HC05 family of MCUs – a priority-based kernel and a time-based kernel. Assembly source code is provided for each.

**Order by: AN1262/D**

---

## Designing for Electromagnetic Compatibility with Single-Chip Microcontrollers

Almost every consumer, automotive and industrial application today contains an MCU. MCU producers are under constant pressure to reduce manufacturing costs, and one way of doing this is to reduce the geometries of the on-chip transistors and gates. As the gate size is reduced the transition time decreases and fast edges on signals produce harmonics which can cause emission problems if amplified. Also, devices with faster transition times can react to fast incoming signals. This note discusses the design of single-chip microcontroller applications taking Electromagnetic Compatibility (EMC) into account, including ‘defensive programming’.

**Order by: AN1263/D**

## JTAG Flash Memory Programmer

During manufacture of embedded processor systems there is no firmware in the on-board memory. Masking software into ROM during device fabrication is generally expensive and restricts new software releases. A more flexible approach is for the system manufacturer to upload the software during production, typically using the built-in serial debug interface in the case of the MC683xx, MPC5xx and MCF52xx families. An alternative which is often overlooked is the JTAG interface, considered by many manufacturers as mandatory, and therefore more generally available. This note describes a JTAG Flash EPROM programmer designed to run on an IBM compatible PC.

**Order by: AN1264/D**

---

## Configuring the MPC2604GA Integrated L2 Cache with the MPC106

**Rev 9**

Adding L2 cache to a system is one of the easiest ways to significantly increase the performance. It is fast becoming a requirement for all computers, especially for RISC architectures such as the PowerPC. The optimum cache is not really possible at reasonable cost, but good design can achieve a close approximation. The MPC2604GA is an integrated secondary cache for PowerPC-based designs and is the fastest L2 cache available for the PowerPC 60x bus. Integration of logic, tag and data on the same silicon allows it respond to a read hit with a 2-1-1-1 burst at 66MHz, with subsequent bursts as fast as 1-1-1-1. This short note explains how to configure it.

**Order by: AN1265/D**

---

## PowerPC 603 Hardware Interrupt Latency in Embedded Applications

The PowerPC 603 is a RISC design achieving a high level of performance using instruction pipelining and a superscalar architecture. In addition to branch folding, two instructions may complete in a single cycle, and as many as five instructions may execute simultaneously. This parallelism complicates how quickly the processor can service external interrupts. This note examines the 603's instruction flow, interrupt recognition method and latency factors. It demonstrates that instruction-caused exceptions do not affect the latency response of most embedded applications, suggests ways that designers can minimize latency, and describes how to use the decrementer exception to measure it.

**Order by: AN1267/D**

---

## PowerPC Microprocessor Clock Modes

The PowerPC microprocessors offer numerous clocking options to allow the designer to use low speed, low cost memory with low cost processors; low speed memory systems isolated from high speed processor cores and internal cache; high performance memory systems with maximum processor and cache frequencies; or embedded systems with unusual bus speeds. This note describes operation of the internal phase-locked loop and discusses how to set up particular frequencies.

**Order by: AN1269/D**

## PowerPC 60x Microprocessor to AD1848 CODEC Interface

This note describes how to interface the Analog Devices AD1848 SoundPort Stereo CODEC to the PowerPC 60x local bus. The AD1848 integrates key audio data conversion and control functions onto a single integrated circuit, and is intended to provide a complete single-chip audio solution for audio and multimedia applications. It provides a direct and near-glueless interface to the Industry Standard Architecture (ISA) AT bus – however there are a number of market areas in which non-ISA based systems are required, and this PowerPC design is a possible solution in these areas.

**Order by: AN1271/D**

---

## Spreadsheet Estimation of CPU-DRAM Subsystem Power Consumption

Energy efficiency of computers is becoming increasingly important to consumers, and it is therefore vital to obtain an accurate estimate of system power consumption early in the design phase. This note describes the use of a Microsoft® Excel 4.0 spreadsheet – referred to as DRAMP – which is available via the World Wide Web. DRAMP supports the 601, 603 and 604 microprocessors, and calculates both the total energy consumed and the average power dissipation for a wide range of operating conditions. Useful system information is generated as a by-product of the calculation.

**Order by: AN1272/D**

---

## HC08 SCI Operation with Various Input Clocks

Describes the operation of the Serial Communication Interface if the MC68HC708XL36 with Clock Generation Module A (CGMA). Specifically the information provides an analysis of the effects of the input clock on the SCI baud rate. SCI communication in various hardware applications is also examined (equal and unequal input clock frequencies), as well as code segments and listings.

**Order by: AN1274/D**

---

## Using and Extending D–Bug 12 Routines

One of the simplest and most economical environments for developing and debugging microcontroller software is a monitor/debugger program residing in ROM and executing in the target environment. This note provides information to allow software developers to utilize internal D-Bug12 routines and shows how to substitute user interrupt service routines for default D-Bug12 exception handlers. It provides six example listings, and source code is available on-line.

**Order by: AN1280/D**

---

## Using the Callable Routines in D-Bug12

All microcontrollers need some form of operating environment for the development and debugging of user software. One of the least expensive is a monitor/debugger program executing in the target environment. A ROM monitor can provide access to many internal utility routines and exception handlers that would otherwise have to be written by the developer. This note provides complete descriptions

of the D-Bug12 user-callable utility functions, and details of how to use them. In addition it shows how to substitute user interrupt service routines for D-Bug12's default exception handlers.

**Order by: AN1280A/D**

## MPC505 Interrupts

The MPC505 interrupt controller receives interrupt requests from multiple interrupt sources and generates a single interrupt signal to the RCPU. This application note describes the function of the interrupt controller and related interrupt registers, and also provides example initialization and handler routines.

**Order by: AN1281/D**

## Board Strategies for Ensuring Optimum Frequency Synthesizer Performance

Microcontroller-based applications can be delayed or jeopardized by poor phase locked loop (PLL) performance. This may be due to the design of the circuit board. This note describes common problems and suggests key practices to avoid PLL problems and performance degradation. Factors considered include board leakage, capacitor characteristics, phase noise from the reference signal, reference spurs and board noise. Discussion centers on the MPC505/MPC509 PLL.

**Order by: AN1282/D**

## Transporting M68HC11 Code to M68HC16 Devices

Devices in the M68HC16 MCU family are built from standard modules that interface via a common internal bus – modularity allows the rapid development of devices tailored to specific applications. The standard CPU in the M68HC16 Family is the 16-bit CPU16 module, and both its programming model and instruction set are designed to be compatible with the M68HC11 CPU. M68HC11 applications can be ported to the CPU16 with only moderate effort, however because the CPU16 has additional capabilities the functions of certain instructions have been modified or replaced. This note compares the capabilities of the two processors, provides information on differences in their instructions sets, and discusses cases that need special attention.

**Order by: AN1283/D**

## Transporting M68HC11 Code to M68HC12 Devices

The CPU12 is generally a proper superset of the M68HC11 CPU. Significant changes have been made to improve the efficiency and capabilities of the CPU without sacrificing compatibility with the popular M68HC11 Family. Every M68HC11 instruction mnemonic and source code statement can be assembled directly with a CPU12 assembler. However it is inevitable that some primary objectives could not be achieved without some differences in the object code. This note provides information that will allow the large number of programmers familiar with the M68HC11 to evaluate moving from an M68HC11 system to an MC68HC12 system.

**Order by: AN1284/D**

## Stepper Motor Control with an MC68HC11E9 Microcontroller

Provides basic design and implementation information for the construction of a stepper motor system – the controller used here is the MC68HC11E9. A general description of the system is given, together with a step-by-step hardware assembly section which is included to simplify practical construction. Includes a listing of basic software, suitable for modification to support a variety of control applications.

**Order by: AN1285/D**

## MC68HC05C0 Bus Structure Design

Explains the basics of designing a system with the MC68HC05C0 – a ROM-less, expanded bus MCU. The document begins by providing answers to frequently asked questions from designers accustomed to working with single-chip MCUs, such as "What can the Chip Selects do?" and "How do I minimize Stop Mode current?". The rest of the application note presents two example schematics illustrating the use of multiplexed and non-multiplexed modes, plus the source code for an MC68HC05C0 to 27C256 interface test program.

**Order by: AN1286/D**

## MC68HC708LN56 LCD Utilities

Describes a set of software utilities to bring functionality to the LCD module of the MC68HC708LN56 MCU. Includes information about some LCD software subroutines that – with minimal effort – can be called to write text to the display. All of this information can be used as a basis for the development of more complex graphics subroutines. The note provides a description of each subroutine, together with code listings and flow charts.

**Order by: AN1287/D**

## Programming the MC68HC(8)05K3's Personality EEPROM on the MMDS and MMEVS

Motorola's MMDS and MMEVS development systems enable designers to emulate members of the M68HC05 Family of MCUs. Host software provides access to the on-chip registers and peripherals of the emulated MCU. However the design of some HC05 peripherals does not allow them to be accessed directly; an example is the Personality EEPROM (PEEPROM) of the MC68HC(8)05K3 MCUs, whose data can only be accessed serially. Currently the only means of programming the PEEPROM – apart from user application software – is to use a standalone programmer. This note discusses the design and implementation of a DOS software utility that allows the PEEPROM to be programmed via an MMDS or MMEVS.

**Order by: AN1288/D**

## DSP5630x FSRAM Module Interfacing

As the complexity of executable code increases, some DSP-based applications demand higher and higher execution speed. To accommodate these requirements Motorola has developed the 24-bit DSP56300 family and a number of support chips to provide cost-effective, high-performance solutions. The DSP56300 core

incorporates a versatile memory interface providing glueless connection to a variety of memory types such as DRAMs, SRAMs and SSRAMs. This note describes several options for interfacing different asynchronous Fast Static RAM modules to the DSP5630x family.

**Order by: AN1289/D**

## Avoiding Multiprocessing Paradoxes with the PowerPC 604 Microprocessor

This document describes three paradoxes that may occur infrequently in a multiprocessing implementation using the PowerPC 604 microprocessor, and how they can be avoided: the lwarx/stwcx. instructions may allow a kill bus operation without modifying the cache block; an lwarx reservation set bus operation may be broadcast without a valid cache entry; and a write-with-kill bus operation may cause a loss of memory coherency.

**Order by: AN1291/D**

## Adding a Voice User Interface to M68HC05 Applications

As embedded MCU-based products become more sophisticated, more emphasis is being placed on their user interfaces. Visual interfaces can be controlled directly by the MCU without additional components. Voice-based user interfaces, on the other hand, are often implemented with speech synthesizers, speech processors, sound generators and DSPs operating in conjunction with the main processor. This note discusses the addition of a voice-based interface to an application based on the MC68HC(7)05J1A MCU, and highlights the use of members of the Information Storage Devices (ISD) 1000 and 2500 family of voice record/playback devices. It presents the design for an audible thermometer.

**Order by: AN1292/D**

## Multiprocessor Systems and the PowerPC 603e Microprocessor

This note describes some of the issues that need to be taken into accout by the systems designer when implementing a multiprocessor system using the 603e or PowerPC 603 processors. Although these processors do not generally provide the hardware support for multiprocessor systems that is available on the PowerPC 604, many of the hardware mechanisms of the 604 that allow efficient multiprocessor operation can in fact be provided by operating system software routines. This document discusses the 603e attributes that require operating system support in multiprocessor systems.

**Order by: AN1294/D**

## Demonstration Model of fuzzyTECH Implementation on M68HC12

The MC68HC12 MCU was introduced in mid-1996 as an upgrade to the M68HC11, one of the most widely used MCUs in the world. It is the first standard MCU to include a complete fuzzy logic instruction set, and this note presents a demonstration model illustrating its use. Inform Software Corp. and Motorola have created the fuzzyTECH MCU-68HC12 Edition, which supports both the M68HC12's fuzzy logic instruction set and its background debug mode. The demonstration model here is an autonomously guided tank, and this note discusses its fuzzy logic controller design and the fuzzyTECH implementation on the M68HC12 MCU.

**Order by: AN1295/D**

## Using the MCM69C232/MCM69C432 Content-Addressable Memory on an ATM Line Card

**Rev 1**

Because of their connection-based protocol, Asynchronous Transfer Mode (ATM) switches must translate each cell's address at every point along the routing path. The speed at which the address information can be translated is a function of several variables including line speed, number of lines connected and the speed of other circuitry on the card. Different approaches are used, of which Content-Addressable Memory (CAM) is the best. Motorola's introduction of the high-capacity, lower-cost MCM69C232 and MCM69C432 CAMs now makes the CAM method a cost-effective option.

**Order by: AN1296/D**

## Variations in the Motorola MC68HC(7)05Cx Family

There are many variations in the devices in Motorola's MC68HC05 C Family of 8-bit microcontrollers, and this note clarifies the important differences. It is particularly useful for designers familiar with one member of the Family who wish to move to another. Topics include Similarities, Comparisons, MC68HC705C9A and MC68HC705C12A, Changing from non-A to A Versions, Changing from OTP to ROM Versions, Changing from the 705C8A to the 705C9A, Voltage Frequency and Temperature Tables, and Development Tools.

**Order by: AN1298/D**

## ATM Switch with Shared Memory – A Simple Model

Asynchronous Transfer Mode (ATM) telecommunications are the mainstay of communication systems today, transmitting data with high throughput across large networks by the high speed transfer of small data payloads. This note uses an airport analogy as a model to explain the features and operation of an ATM switch based on Motorola's NetRAM memory.

**Order by: AN1299/D**

## Integrated Sensor Simplifies Bar Graph Pressure Gauge

**Rev 1**

Integrated semiconductor sensors such as the MPX5100 greatly simplify electronic measurement of pressure. Their linear 0.5V to 4.5V outputs are designed to interface directly with MCU A/D inputs. They can also be used with devices such as the LM3914 Bar Graph IC to create a simple Bar Graph Pressure Gauge. This note describes

the operation and calibration of the DEVB129 evaluation board, which has an on-board LED bar graph display and is designed to operate on an unregulated +12V supply.

**Order by: AN1304/D**

## An Evaluation System for Direct Interface of the MPX5100 Pressure Sensor with a Microprocessor

**Rev 1**

Recent design advances in pressure sensor technology now allow the direct interface of a pressure sensor to a microprocessor with on-chip A/D converter. This has been made possible by integrating a temperature compensated pressure sensor element and active linear circuitry on the same die. The DEVB114 evaluation board described here shows how simple the interface can be. Includes, circuit, parts list and source/assembly code for an MC68HC705B5.

**Order by: AN1305/D**

## A Simple Pressure Regulator Using Semiconductor Pressure Transducers

**Rev 1**

Semiconductor pressure transducers offer an economical means of achieving high reliability and performance in pressure sensing applications. The completely integrated MPX5100 (0-15p.s.i.) series provides a temperature compensated, high-level linear output suitable for interfacing directly with many linear control systems. This circuit illustrates how the MPX5100 can be used with a simple pressure feedback system based on the MC33033 Brushless Motor Controller to establish pressure regulation. Includes circuit diagram and PCB artwork.

**Order by: AN1307/D**

## Compensated Sensor Bar Graph Pressure Gauge

**Rev 1**

Compensated semiconductor-based pressure sensors such as Motorola's MPX2000 family are relatively easy to interface with digital systems. Using the circuit described here, pressure is translated into a 0.5 to 4.5 volt output range that is directly compatible with MCU A/D inputs. This range is also suitable for the LM3914 Bar Graph Display Driver. This note provides information on the use and operation of the DEVB147 evaluation board.

**Order by: AN1309/D**

## An Evaluation System Interfacing the MPX2000 Series Pressure Sensors to a Microprocessor

Outputs from compensated and calibrated pressure sensors such as Motorola's MPX2000 series are easily amplified and interfaced to a microprocessor. This application is based on the DEVB158 evaluation board which implements a simple analogue interface. The sensor output is amplified using a quad operational amplifier. No potentiometers are used to adjust the span and offset – these adjustments are made in software, the offset voltage being computed empirically each time power is applied to the system, and the result stored in RAM.

**Order by: AN1315/D**

## Frequency Output Conversion for MPX2000 Series Pressure Sensors

**Rev 1**

Sensing remotely and/or in noisy environments is particularly challenging for low-level (mV) voltage output sensors such as the MPX2000 Series pressure sensors. But converting the MPX2000 output to frequency is relatively easy to accomplish, and provides the noise immunity required for accurate sensing. The DEVB160 evaluation board presented here is an excellent tool either for "stand-alone" evaluation of the MPX2000 Series or as a building block for system prototyping. The output of the DEVB160 circuit is ideally conditioned for interfacing to MCU timer inputs that can measure the sensor frequency.

**Order by: AN1316/D**

## Interfacing Semiconductor Pressure Sensors to Microcomputers

**Rev 1**

The output voltage for Motorola's piezoresistive pressure sensors is generally 25 to 50mV full scale. Interfacing to a microcomputer, therefore, involves amplifying the relatively small output voltage, performing a differential to single-ended conversion, and scaling the signal into an appropriate range for A/D conversion. Alternately, it can be converted to a frequency modulated 5V waveform or 4-20mA current loop, either of which is relatively immune to noise on long interconnect lines. Sensing amplifiers, analogue to digital conversion, frequency modulation and 4-20mA current loops are considered.

**Order by: AN1318/D**

## Applying Semiconductor Sensors to Bar Graph Pressure Gauges

Bar graph displays are particularly useful in process monitoring applications where quick communication of a relative value may be more important than providing specific data – they quickly convey a sense of how much of something is present. Designing bar graph pressure gauges based on semiconductor sensors is relatively straightforward, and can make use of bar graph display drive ICs, microcomputers and MC33161 voltage monitors. Design examples of all three types are given here, with consideration of the trade-offs.

**Order by: AN1322/D**

## A Simple Sensor Interface Amplifier

**Rev 1**

Compensated semiconductor pressure sensors such as the MPX2000 family are relatively easy to interface to digital systems. With these sensors and the circuitry described here, pressure is translated into

a 0.5 to 4.5V signal that is directly compatible with MCU A/D inputs. This simple circuit is implemented in the DEVB173 development board. It consists of one dual op amp and a few resistors, and can accept MPX2010, MPX2050, MPX2100, MPX2200 and MPX2700 sensors to cover different pressure ranges.

**Order by: AN1324/D**

## Amplifiers for Semiconductor Pressure Sensors

**Rev 2**

Amplifiers interfacing semiconductor pressure sensors to electronic systems have typically been based upon classic instrumentation amplifier designs – well understood standard building blocks that also work reasonably well. But other circuits can do a better job of interfacing pressure sensors to today's mostly digital systems. This note presents an evolution of amplifier designs, beginning with a classic instrumentation amplifier and ending with a simpler circuit that is better suited to sensor interfacing.

**Order by: AN1325/D**

## Barometric Pressure Measurement Using Semiconductor Pressure Sensors

The digital barometer system described here is an excellent example of a sensing system using solid state components and software to measure barometric pressure accurately. The system serves as a foundation from which more complex systems can be developed. It is based on an MPX2100A series device, and also uses an M68HC11 MCU to convert the output of the signal conditioning circuit to a digital value, convert this measurement to inches of mercury, and output the data serially to an MC145453 LCD interface. The software is listed and is also available from the Motorola bulletin board.

**Order by: AN1326/D**

## Mounting Techniques and Plumbing Options of Motorola's MPX Series Pressure Sensors

**Rev 1**

Motorola offers a wide variety of ported pressure sensing devices incorporating a hose barb and mounting tabs. This application note provides some recommendations on types of fasteners for mounting and how to use them with Motorola sensors. It also recommends a variety of hoses and hose clamps, and includes a review of recommended mounting hardware, mounting torque specifications, hose applications and hose clamps. Includes a list of (US) suppliers of these parts.

**Order by: AN1513/D**

## Liquid Level Control Using a Motorola Pressure Sensor

**Rev 1**

This circuit provides a complete, low cost solution for the direct control of liquid level using a pump or solenoid valve. It is based on an MPX2000 series temperature compensated pressure sensor, and an MOC2A60 power opto isolator. Both devices are described, and a practical example is given which includes the theory and a full schematic. As well as being a practical circuit this application may be used to evaluate the two principal devices.

**Order by: AN1516/D**

## Pressure Switch Design with Semiconductor Pressure Sensors

**Rev 1**

This pressure switch design uses a comparator to provide a logic level output by comparing the output of a Motorola pressure sensor with a reference voltage. After an introduction to the sensor and amplifier sections of the circuit, the application note is concerned mainly with a discussion of the important comparator section, presenting circuits based on different op amp types and evaluating their performance in terms of switching speed and voltage levels. Includes a window comparator design.

**Order by: AN1517/D**

## Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors

A Frequency Modulated (FM) or Pulse Width Modulated (PWM) output is better than an analogue voltage for remote sensing applications in noisy environments. This note discusses a simple PWM circuit for use with the MPX5100 pressure sensor that generates a signal with a duty cycle proportional to applied pressure. It is intended for use with a microcontroller, which generates the pulse train to drive the circuit's ramp generator – the use of the same timebase to both generate and measure the PWM signal gives a stable and accurate result. Since the PWM output calibration is controlled by software any component tolerances can be compensated.

**Order by: AN1518/D**

## The A-B-Cs of Signal-Conditioning Amplifier Design for Sensor Applications

There are many pressure sensor applications where the flexibility of a custom designed signal conditioning circuit is beneficial, despite the availability of fully conditioned, calibrated and temperature compensated sensor ICs. The signal conditioning circuits described here are applicable to low-level, differential voltage output sensors in general, but the emphasis is on interfacing pressure sensors to amplifier circuits. Includes a general description of the two operational amplifier circuit, plus theoretical analysis.

**Order by: AN1525/D**

## RF Power Device Impedances: Practical Considerations

Many first-time RF power designers assume that small-signal techniques are also applicable to bipolar Class C and Class AB power amplifier design. In fact, higher power gain and better efficiency are achieved if the output is purposely mismatched. The note defines large-signal series equivalent input and output device impedances for RF power transistors, together with the techniques

for measuring them. It examines how these parameters change under varying load and bias conditions, and demonstrates the impact of the variations in a practical broadband test fixture design.

**Order by: AN1526/D**

## Using the Motorola MRFIC1806/1807 Dual Demonstration Board

The MRFIC1806/1807 Dual Demonstration Board combines Motorola's MRFIC Driver/Ramp IC and the MRFIC1807 PA/Switch IC on a common evaluation board. The two ICs form a system solution for the 1.9GHz DECT transmit power amplifier and transmit/receive antenna switch. The board enables an evaluation of system level transmitter performance specifications, and of transmit/receive switch performance, to be made. This note provides information on the use of the board, including circuits and performance details, and describes how it can be modified for PHS applications.

**Order by: AN1532/D**

## Digital Boat Speedometers

This boat speedometer design is based on an MPX2200GP silicon pressure sensor, analog signal-conditioning circuitry, an M69HC11-family MCU and a liquid crystal display. The sensing system converts water head pressure to boat speed, and has a range of 5 to 45 m.p.h. Motorola's silicon pressure sensors use a single piezoresistive element coupled to on-chip temperature compensation circuitry, ensuring simpler designs and improved performance and reliability. Includes a full description, circuit diagrams and program listing for the MC68HC711E9.

**Order by: AN1536/D**

## An IF Communication Circuit Tutorial

A tutorial on the use of IF communication integrated circuits, based on the ISM band channel and the Motorola MC13156. Examines the device's topology and discusses the classical parameters critical to the proper operation of a typical RF device – impedance matching the mixer, selecting the quad tank and filters, plus bit error rate testing for digital applications. The reader should end up with a better understanding of IF communications basics, and be able to specify the support components required.

**Order by: AN1539/D**

## Low-Pressure Sensing with the MPX2010 Pressure Sensor

Low-cost semiconductor pressure sensors are typically designed to measure full-scale pressures down to 10kPa (1.5 psi or around 1000mm of water) with reasonable accuracy. But some applications, such as heating and ventilating (HVAC) and washing machines, may need to measure full-scale pressure as low as 40mm of water. This 'smart sensing' system demonstrates a method of sensing full-scale pressures below 250mm of water with 1% full-scale resolution and 2% full-scale accuracy. An MCU is used for signal averaging, software calibration and software-based power supply rejection.

**Order by: AN1551/D**

## Designing Sensor Performance Specifications for MCU-based Systems

Using fixed-value components in the design of a sensor signal conditioning circuit makes the system easier and cheaper to produce in high volume. However in attempting to achieve the largest possible output voltage range for subsequent processing, there is a danger that device-to-device variations coupled with circuit variations and temperature effects can saturate the amplifier or exceed the limits of the following process (an A/D converter for example). This note discusses a methodology that optimizes a sensor system's performance while ensuring that the amplified output will always remain within the limits.

**Order by: AN1556/D**

## A Cookbook Approach to Designing a Differential-Signal Amplifier for Sensor Applications

Sensors with millivolt outputs need signal conditioning amplification to customize the output for the intended application, and to compensate for device-to-device variations in offset and span. This practical application note focuses on the mechanics of the necessary simple calculations, resistor selection and calibration of the final circuit in a step-by-step manner. It is based on sound engineering design principles which are explained in a separate appendix. The examples presented are concerned with pressure sensors in two situations: applications where the variations are taken from a data sheet, and those where the device characteristics can be measured before assembly.

**Order by: AN1557/D**

## Application Considerations for a Switched Capacitor Accelerometer

**Rev 1**

Low cost accelerometers are highly integrated devices with features such as signal conditioning, filtering, offset compensation and self test. Combining these features with economical plastic packaging requires that the signal conditioning circuitry should be as small as possible. One approach is to implement sampled data system and switched capacitor techniques as used in the MMAS40G accelerometer. As in all sampled data system, precautions should be taken to avoid signal aliasing errors. This note describes the MMAS40G, explaining how aliasing might occur and how to minimise it.

**Order by: AN1559/D**

## Digital Blood Pressure Meter

This note describes the concept of a digital Blood Pressure Meter which combines an integrated pressure sensor, analog signal conditioning circuitry, microcontroller hardware and software, and a liquid crystal display. The sensing system reads the cuff pressure and extracts the pulses for analysis and determination of systolic and diastolic pressure. The design is based on the Motorola

MPX5050GP 50kPa integrated pressure sensor which has a pressure range of zero to 300mm Hg. Includes circuit schematic and description, plus an MC68HC05B16 software description and flowchart.

**Order by: AN1571/D**

## Understanding Pressure and Pressure Measurement

Fluid systems, pressure and pressure measurement are complex subjects. This application note defines and explains the basic concepts of fluid mechanics in terms that are easy to understand while retaining the necessary technical accuracy and level of detail. It opens by defining Fluid Pressure and the various types of pressure measurement and units and goes on to discuss the concepts of static and dynamic pressure systems, including both steady-state and transient situations.

**Order by: AN1573/D**

## Understanding the Multivibrator Based Crystal Oscillator Circuit Used on the BiCMOS MPC Family of Clock Drivers

When a Phase Locked Loop (PLL) is used in a clock generator it is desirable to use a crystal controlled source as the reference clock – crystals provide accurate frequencies at reasonable cost. To minimise implementation costs many PLL clock generators integrate the crystal oscillator circuitry onto the chip, so that the crystal itself is the only external component. The standard Pierce oscillator, based on an inverter gate, is most commonly used, but many of Motorola's MPC clock drivers use an alternative multivibrator-based design. This note outlines the important differences between the two, and presents guidelines for applications which require very accurate clock frequencies.

**Order by: AN1579/D**

## Mounting and Soldering Recommendations for the Motorola Power Flat Pack Package

**Rev 3**

Motorola's Power Flat Pack-16 (PFP-16) is a superior package for high-power surface mount applications. It is a thin, space-efficient package offering a variety of soldering options, and can be assembled into PC boards using standard equipment. Unlike most surface mount packages it has very high thermal conductivity, allowing die to dissipate up to 5 watts without needing excessive board space. This note discusses handling and soldering considerations that will allow users to take full advantage of the PFP-16.

**Order by: AN1580/D**

## "Very Low Pressure" Smart Sensing Solution with Serial Communications Interface

This note is an update on recent progress in using local intelligence to improve functionality and performance for low-pressure smart sensing applications. The enhancements build on work documented in an earlier paper presented at Sensors Expo Boston '95. The original MPX2010-based system had been developed to measure 0-2.5 kPa with 1-2% accuracy. While this provides an accurate solution for a range spanning several kPa, it cannot maintain this performance for sub-kPa pressure ranges. It was therefore decided to develop a solution for full-scale pressure ranges as low as 0.375 kPa with 1-2% overall accuracy. Typical applications include liquid-level and gas-flow sensing.

**Order by: AN1584/D**

## High-Performance, Dynamically-Compensated Smart Sensor System

The sensor itself is at the heart of any measuring system that requires a physical condition to be converted to an electrical variable. The system presented in this application note converts a physical pressure to a voltage, and subsequently to a digital value, but the techniques are relevant to all types of sensor. Accuracy and resolution are the critical performance criteria. This system eliminates device-to-device process variations; corrects for temperature dependencies of the sensor output; and optimizes the available resolution by means of a closed-loop, MCU-based, dynamic compensation system.

**Order by: AN1585/D**

## Designing a Homemade Digital Output for Analog Voltage Output Sensors

A digital sensor output is generally preferred to an analog output in noisy environments and in remote sensing applications, because of its inherently better noise immunity. In addition, MCU-based systems with no built-in A/D converter have no option but to use a digital signal. The design example in this application note, which is based on an MPX2000-series pressure sensor, demonstrates how to easily convert an analog voltage output sensor to a digital output sensor. Includes sample calculations and example software for the MC68HC05P9 MCU.

**Order by: AN1586/D**

## Longwave Radio Data Decoding Using an HC11 and an MC3371

In the UK, the BBC's Radio 4 198kHz Longwave transmitter carries data as well as the audio signal; this has some similarities with the RDS data included in VHF radio signals in many European countries but has a much lower data rate and serves a different purpose. This application is based on an MC68HC(7)11 and an MC3371 superheterodyne receiver, and allows time and date to be permanently displayed while all incoming data can be displayed in hexadecimal form. It incorporates an alarm clock which can be used to switch on the radio at the required alarm time.

**Order by: AN1597/D**

## Power Control with the MRFIC0913 GaAs Integrated Power Amplifier and MC33169 Support IC

The transmitted RF power of typical multiple access radio systems is programmable within a given range. The advantages are that the interference level for close receivers is restricted, and that the

transmitter power consumption is reduced. In addition to this output power control, on/off switching of the RF power must be tightly controlled to avoid splattering the signal into adjacent channels; this is done by controlling the rise and fall times of the transmitter keying. This note discusses the details of waveform shaping and power control as applied to GSM TDMA systems using the MRFIC0913 GaAs Integrated Power Amplifier (IPA) and the MC33169 support IC.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AN1599/D**

---

## 3.6V and 4.8V GSM/DCS1800 Dual Band PA Application with DECT Capability Using Standard Motorola RFICs

The GSM communications standard in Europe is used in both the 900MHz and 1800MHz bands. With the prospect of system interoperability there is growing interest in portable phones capable of being used on both bands. This note describes the design, implementation and performance of a dual band GSM power amplifier for 900MHz and 1800MHz, using currently available standard RF ICs – with some modification the design can also be used for GSM/DCS1800/DECT applications. Includes circuits, parts lists and component layouts.

**Order by: AN1602/D**

---

## Using Motorola's MRFIC1502 in Global Positioning System Receivers

The Global Positioning System is a US Department of Defense operated facility consisting of 24 satellites in orbit at an altitude of 20,183km, which continuously broadcast a navigation message on two L-band frequencies. The coarse acquisition code (C/A) and the precise code (P) are broadcast on Link 1 at 1574.42MHz. Motorola's MRFIC1502 downconverter is targeted for the reception of the C/A code, although it is potentially capable of receiving the P-code also. This note describes its use as the downconverter in a GPS receiver.

**Order by: AN1610/D**

---

## Impact and Tilt Measurement Using Accelerometer

This note describes a system for the measurement of both tilt and impact using a Motorola MMAS40G10S accelerometer, supported by microcontroller hardware and software, and a liquid crystal display. Due to the wide, DC to 400Hz frequency response of the accelerometer, the system can measure both the static acceleration of the Earth's gravity and shock or vibration from an impact. Includes circuit schematic, flow chart and program listings for an MC68HC05B16 MCU.

**Order by: AN1611/D**

## Shock and Mute Pager Applications Using Accelerometer

**Rev 2**

In typical pager designs, whenever there is an incoming call the pager will 'beep' until one of the buttons is physically pressed. This can be inconvenient if the controls are not within easy reach. This note describes the use of a low-cost accelerometer to allow the beep to be muted by lightly tapping the pager, which could be inside a pocket or handbag. The design is based on a 40g MMAS40G10D accelerometer. Includes circuit schematic, parts list, PCB artwork, flow chart and source code for an MC68HC705B16 MCU.

**Order by: AN1612/D**

---

## Mounting Recommendations for Copper Tungsten Flanged Transistors

Because of mechanical constraints caused by the hardness of the flange material, RF power transistors with a Copper Tungsten (CuW) flange require special care in mounting. This note describes the correct mounting procedures, with emphasis on the surface flatness and the torque required. Packages affected include Case 360B, Case 375A, Case 375B, Case 395B, Case 395C, Case 398, and others.

**Order by: AN1617/D**

---

## EMC Considerations for Automotive Sensors

**Rev 1**

Electro Magnetic Compatibility (EMC) is a qualification requirement for automotive components, which need to work in an environment that is more and more contaminated with electromagnetic (EM) energy. This note considers the susceptibility to EM signals of plastic-packaged pressure sensors with integrated signal conditioning. Method and results of practical tests (GTEM and Direct RF Injection) are given, and show that the device has good immunity to the surrounding EM field. Discusses a filter technique which further improves performance.

**Order by: AN1622/D**

---

## ITC137 68HC708MP16 Motion Control Development Board

The ITC137 Motion Control Development Board described here complements the software development tools available for the 68HC708MP16 MCU. It provides motor control functions on a board that is easy to interface both to power stages and emulators, and its configuration is applicable to AC Induction, Brush DC and Brushless DC motors. This note includes a description, schematic and parts list for the board, plus an application example in which it is coupled to an ITC132 power stage and an induction motor. It also discusses connection to a host PC.

**Order by: AN1624/D**

## Low Cost High Efficiency Sensorless Drive for Brushless DC Motor Using MC68HC(7)05MC4

Variable speed drives are increasingly used in appliances to increase whole system efficiency and product performance. The low dynamic drive – where the load or speed is changed relatively slowly – is a solution in many cases, and can be controlled by simple algorithms. Three phase brushless DC motors are efficient and easy to drive, but commutation of motor phases relies on detecting rotor position. This note presents a sensorless speed control system which is especially suitable where high efficiency and low price are important. It gives a fundamental mathematical method for modelling, torque calculation and control concept. Simulation results show the drive behavior under different working conditions.

**Order by: AN1627/D**

---

## MMA1000P Product Overview and Interface Considerations

Motorola's accelerometer architecture combines a sensing element and a control ASIC in a single package to meet its stringent performance requirements at low cost. This note describes the MMA1000P accelerometer which uses a new control ASIC architecture. It explains important new features that have been incorporated, and presents an overview of the key performance characteristics of the new accelerometer. It also details the minimum supporting circuitry needed to interface the device to an MCU. Finally it discusses the power supply rejection ratio characteristics and presents an aliasing gain model.

**Order by: AN1632/D**

---

## Implementing Auto Zero for Integrated Pressure Sensors

**Rev 1**

Auto Zero for pressure sensors is a compensation technique based on sampling the offset of the sensor at a reference pressure in order to correct the sensor output for long-term drift or variation. It can be implemented easily when an integrated pressure sensor is interface to a microcontroller with A to D converter. The main requirement is that a zero pressure reference condition must exist at some point in the operating cycle of the equipment, for example at start up or during idle conditions. Typical systems that can benefit include washing machines, bottle filling and HVAC applications.

**Order by: AN1636/D**

---

## Offset Calibration of Gauge Pressure Sensor Using Parallel I/O Ports

**Rev 1**

External stresses an mounting position can affect the 'zero pressure' output reading of a gauge pressure sensor, especially when using low pressure devices. This note describes a method of calibrating the offset of a sensor using the parallel I/O ports of a microcontroller; the demonstration board contains an MPXT5006D sensor, MC68HC705B16 MCU and an LCD display. The offset value can be stored in ROM, EEPROM or RAM, depending on the application.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AN1638/D**

---

## Phase Noise Measurement Using the Phase Lock Technique

Phase noise causes the noise skirts that surround the carrier of any signal source; it may be caused by either amplitude or frequency modulation. This note explains the Phase Locked Loop (PLL) method for measuring phase noise. It uses a diode ring mixer, op-amp based amplifiers and an integrator to phase-lock two signal sources. In the open loop state there is a frequency difference between the sources which produces a beat at the mixer IF port, and this is the reference level for the measurement. Includes block diagram, analysis and discussion of the procedure for valid measurements.

**Order by: AN1639/D**

---

## Reducing Accelerometer Susceptibility to BCI

**Rev 1**

Automobile electronic systems – and airbag systems in particular – must pass stringent electromagnetic compatibility (EMC) tests. One of the toughest tests for the tolerance of the system to high frequency conducted emissions is the Bulk Current Injection (BCI). The entire airbag system must continue to function normally throughout this test. This note discusses how to reduce the susceptibility of the Motorola MMA1000P accelerometer to BCI; the information can also be applied to other electronic components.

**Order by: AN1640/D**

---

## RF LDMOS Power Modules for GSM Base Station Application: Optimum Biasing Circuit

The performance of RF power amplifiers for base station transceivers results in a trade-off between linearity, efficiency and gain, and this leads to an optimum quiescent current. But temperature range, supply and bias voltage variations, and manufacturing spread will modify this bias point. This application note presents a new biasing circuit, suitable for LDMOS RF power transistors, which minimises quiescent current variations.

**Order by: AN1643/D**

---

## Micromachined Electromechanical Sensors for Automotive Applications

**Rev 1**

Typical automotive applications for pressure sensors include MAP, BAP, lumbar seat and air bag; acceleration sensor applications include airbag, yaw rate, active suspension and ABS. This note discusses the function and applications of the two types, their

different micromachining techniques and different signal conditioning. It explains electrical characteristics and package styles, and presents the requirements of the MAP/BAP application in some detail.

**Order by: AN1645/D**

## Noise Considerations for Integrated Pressure Sensors

**Rev 1**

Motorola Integrated Pressure Sensors have trimmed outputs, built-in temperature compensation and an amplified single-ended output, making them compatible with the A/D converters of low cost microcontrollers. Although 8-bit ADCs are most common, higher resolutions are increasingly becoming available. In the higher resolution ADCs the noise that is inherent to piezoresistive bridges becomes a design consideration. This note presents simple techniques for mitigating the effects of shot noise, flicker noise and external noise to achieve excellent results.

**Order by: AN1646/D**

## ASB201 – Uncompensated Series Sensor Module

**Rev 1**

Describes the ASB201 plug-in module that forms part of a systems development tool set for pressure sensors. The module provides an analog signal from an Uncompensated series sensor to a Motorola ASB200 Sensor Development Controller, or it can be used on its own to provide power and signal connections to the sensor. Includes a schematic, parts list, pin by pin description and summary of design considerations.

**Order by: AN1651/D**

## ASB202 – MPX2000 Series Sensor Module

**Rev 1**

Describes the ASB202 plug-in module that forms part of a systems development tool set for pressure sensors. The module provides an analog signal from an MPX2000 series sensor to a Motorola ASB200 Sensor Development Controller, or it can be used on its own to provide power and signal connections to the sensor. Includes a schematic, parts list, pin by pin description and a summary of design considerations.

**Order by: AN1652/D**

## ASB205 – MPX5000 Series Sensor Module

**Rev 1**

Describes the ASB205 plug-in module that forms part of a systems development tool set for pressure sensors. The module provides an analog signal from an MPX5000 series sensor to a Motorola ASB200 Sensor Development Controller, or it can be used on its own to provide power and signal connections to the sensor. Includes a schematic, parts list and pin by pin description.

**Order by: AN1653/D**

## ASB210 – 10" $H_2O$ Sensor Module

**Rev 1**

Describes the ASB210 plug-in module that forms part of a systems development tool set for pressure sensors. The module provides a pulsed analog signal from an MPX2010 sensor to a Motorola ASB200 Sensor Development Controller – the pulsed architecture improves the sensitivity of MPX2010 sensors to allow a 0 to 10" measurement range. Includes schematic, parts list and a summary of design considerations.

**Order by: AN1654/D**

## ASB200 – Motorola Sensor Development Controller Board

**Rev 1**

Presents an MC68HC705JP7-based development board that is part of a systems development tool set for pressure sensors. When used with a series of companion plug-in modules the board provides a complete systems solution for measuring pressure and developing code. It receives signal inputs from a series of pressure sensor modules, receives command inputs from a DIP switch or terminal keyboard and sends results to a terminal or LCD. Includes schematics, parts list, full description and software overview.

**Order by: AN1655/D**

## Compound Coefficient Pressure Sensor PSPICE Models

**Rev 1**

Presents PSPICE models for Uncompensated, MPX2000 series and MPX5000 series pressure sensors. The models use compound coefficients to improve modeling of temperature dependent behavior. The discussion begins with an overview of the structure of the models, and is followed by an explanation of compound coefficients. The emphasis is on how to use the models to estimate sensor performance.

**Order by: AN1660/D**

## Low Cost Universal Motor Sensorless Phase Angle Drive System

This design for a phase angle motor control drive system is based on the MC68HC05JJ6 or MC68HC705JJ7 microcontrollers and the MAC4DC snubberless triac. The low-cost single-phase power board is intended for universal brushed motors operating from 1,000rpm to 10,000rpm, typically found in home appliances such as vacuum cleaners, washers, hand tools and food processors. The speed of the motor is regulated, but the design operates without a tachometer to reduce system cost.

**Order by: AN1663/D**

## Low Cost 3-Phase AC Motor Control System Based on MC68HC908MR24

This design for a 3-phase AC induction motor drive is based on the MC68HC908MR24 microcontroller, which is optimized for motor control applications. The system is designed as a low-cost, high-volume solution for medium power motors used in both industrial and appliance applications, such as washing machines, compressors, air conditioning units and simple industrial drives. The application note includes a review of trends and general requirements for variable speed 3-phase AC drives, and a description of both hardware and software with parts list and software listing.

**Order by: AN1664/D**

---

## Software SCI Implementation to the MISC Communication Protocol

Describes a software implementation of asynchronous serial communication for microcontrollers which do not have a hardware SCI. Use of such MCUs is important in low cost Niche Area Networks. Previous software SCI solutions have only provided very basic communications – in the system described here a complete data link protocol stack is implemented and executed. The MISC communication protocol is used here as an example.

**Order by: AN1667/D**

---

## Washing Appliance Sensor Selection

North American washing machines currently in production generally use mechanical sensors for water level measurement; either pressure switches with discrete trip points or electromechanical pressure sensors with frequency output. Tests indicate that the accuracy, linearity and repeatability required of high efficiency machines is difficult to achieve by these methods. Manufacturers in Europe, and more recently in North America, are looking at electronic solutions. This note is a guide to the selection of sensors from Motorola's MPX series, especially for high accuracy with few components in high volume applications.

**Order by: AN1668/D**

---

## 60 watts, GSM 900MHz, LDMOS Two-Stage Amplifier

Demonstrates the feasibility of a complete RF amplifier for GSM 900MHz base stations using LDMOS transistors in Class AB. The complete design requires only standard parts and components, yet shows superior performance in terms of gain efficiency, power and ruggedness. The target is an output power of 60 Watts continuous wave, with an efficiency as high as possible and gain in region of 30dB.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AN1670/D**

## MC145170 PSpice Modeling Kit

Device models and simulators such as PSpice can reduce development time; the key is in developing the proper model. IC design models could be used for simulating a circuit design, but it is a cumbersome, inefficient and often inaccurate method. Far better to devise a model which performs to the specification for the IC and trust the IC designer to produce a device that conforms. This note describes the application of this technique to the MC145170 PLL Frequency Synthesizer. PSpice Analog Behavioral Modeling is used to develop phase detector and VCO models, which are used to simulate open loop and closed loop, and to test purity of the VCO output.

**Order by: AN1671/D**

---

## Solder Reflow Mounting Method for the MRF286 and Similar Packages

Describes a solder mounting method for the MRF286 60 Watt power device. The methodology is recommended for any ceramic/metal flange device with similar materials and construction (copper tungsten flange with Alloy 42 leads) and a power range from 20 to 60 Watts. The method was developed after comprehensive simulation which included thermal management and mechanical stress modeling.

**Order by: AN1673/D**

---

## A Low Noise Amplifier with High IP3 for the 900MHz Band Using the MRF1057T1 Low Noise Transistor

The MRF1057T1 is a low noise bipolar junction transistor, part of a family of sub-micron geometry devices which also includes the MRF1027T1 and the MRF1047T1. The main difference between these devices is their current carrying capability. This note describes the performance of the MRF1057T1 in a Low Noise Amplifier circuit whose requirements are typical of the most recent cellular communication technologies such as CDMA and TDMA.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AN1675/D**

---

## A Cascade 2 Stage Low Noise Amplifier Using the MRF1047T1 Low Noise Transistor

This circuit design demonstrates the performance of the MRF1047T1 low noise bipolar transistor in a cascade LNA for a pager application. It provides a good compromise between low NF (1.6dB), high IP3 and high return losses with unconditional stability. Gain is typically 24dB. Includes print layout, components list, circuit schematic, simulated and measured data, and general information on the LNA circuit.

**Order by: AN1676/D**

## A Full-Featured Wireless Interface for RS-232 Communications

This full-duplex, wireless data communication link is targeted at RS-232 applications. An encoding technique has been designed which addresses many of the problems incurred when attempting to implement the RS-232 standard, including hardware flow control, the DC component of the transmitted signal, automatic synchronization from host to slave, and error detection. The design is implemented using Motorola's ISM Band RF chipset – consisting of the MC13145 RF Receiver, the MC13146 RF Transmitter and the MC33411 Baseband – plus an HC05 Family MCU and some CMOS logic.

**Order by: AN1687/D**

## Alarm IC General Applications Overview

The MC14600 is a versatile IC that can be configured with minimum external components to serve a wide range of alarm applications and circuit configurations. For example, it can be used in systems that detect pressure and temperature change, liquid levels, motion or intrusion. Essentially the device is a comparator with horn driver, LED driver and low battery detector. This note presents considerations in interfacing external components to the MC14600, and an approach for configuring it with a latch.

**Order by: AN1690/D**

## Practical Solutions for Medium Data Rate Wireless Communications

The availability of low cost FM transceiver ICs targeted for use in unlicensed FCC bands makes the implementation of medium rate data radios for the consumer market a reality. This note examines various practical solutions for an inexpensive, wireless point-to-point link. The 902-928MHz Industrial, Scientific and Medical (ISM) band was chosen as an ideal radio frequency medium because of its low implementation cost and its unlicensed (FCC Part 15) operation. Problems that are normally unique to the wireless solution are discussed, and the note examines solutions for both digital and RF portions of the link.

**Order by: AN1691/D**

## Broadband Intermodulation Performance Development Using Rohde & Schwarz Vector Network Analyzer ZVR

Historically, broadband linear amplifiers were developed using a technique of tuning and testing intermodulation distortion (IMD) via a two-tone signal at several frequencies in the band. After several iterations a linear circuit was achieved. This note presents an innovative system that provides simultaneous broadband swept measurements of gain, input return loss, and two-tone IMD, so facilitating real-time measurement of key high power RF amplifier design parameters. System theory and configuration are discussed, and procedures for system setup and calibration are detailed in appendices.

**Order by: AN1696/D**

## Brushless DC Motor Control Using the MC68HC705MC4

Details the design and analysis of a brushless DC motor control system using the MC68HC705MC4 MCU with two Motorola evaluation boards. Brushed DC motors have long been popular, partly because of their minimal need for electronic control. Now, however, the use of reasonably priced, electrically commutated, brushless DC motors is rising, together with the need for greater control. Such motors are found in disk drives, household appliances and automotive applications, where variable speed control, external connection and flexibility are required at little or no extra cost. The MC68HC705MC4 provides a flexible and low cost motor control platform.

**Order by: AN1702/D**

## Low Power on the SCM68000 (EC000 Core)

The SCM68000 (EC000 Core) has been redesigned to provide low-power, fully static operation. This document describes the recommended method for placing the SCM68000 into low-power mode to reduce power consumption to its quiescent value, while maintaining the internal state of the processor.

**Order by: AN1703/D**

## Switch Fabric Implementation Using Shared Memory

**Rev 1**

Computers and networks are inextricably linked in modern business, and just as computers continue to increase their performance, users demand ever increasing bandwidth in Mbits or Gbits per second from their networks. Many high speed technologies have emerged, but only Asynchronous Transfer Mode (ATM) can integrate voice, video and data. The switch fabric buffering scheme is of major importance to the flexibility and adaptability of the network. This note discusses two switch fabric implementations using Motorola's NetRAM, a dual-port SRAM designed specifically for the communications market, and compares the improved performance over the Burst SRAM that is sometimes used.

**Order by: AN1704/D**

## Noise Reduction Techniques for Microcontroller-Based Systems

The push towards faster MCUs and peripherals means that new product designs face an increasing threat from electromagnetic interference (EMI), now discussed more positively under the heading of 'electromagnetic compatibility' (EMC). EMI can, and often does, cause delays in product development, but early and continuous attention to EMC issues will give the product the best chance for minimum development costs and delays. This note focuses mainly on reducing emission, but many of the guidelines presented here also affect a system's susceptibility to interference.

**Order by: AN1705/D**

## Microcontroller Oscillator Circuit Design Considerations

The heartbeat of every microcontroller design is an oscillator, and most designs needing precise timing over a wide temperature range use a crystal. PCB designers have the task of integrating crystal and microcontroller functions without the help of mating specifications. This note promotes a systematic approach to good oscillator design, and points out some common pitfalls. It discusses oscillator theory, amplifier gain and crystal drive, potential problem areas and troubleshooting.

**Order by: AN1706/D**

## Dual Port Memory for Multiprocessor Applications

The most common implementation of a multiprocessor system is one where the processors share a common system bus. Because the bus is the sole avenue to access main memory and the system's I/O devices, it becomes a bottleneck which causes performance degradation. The use of external cache memory can help alleviate this problem, and this note discusses the various implementations of external caches and the pros and cons of each. In particular, it illustrates the advantages of using Motorola's dual port SRAM, the MCM69D618, for both the tag and data RAM of an in-line cache.

**Order by: AN1707/D**

## DMA08 Systems Compatibilities

The DMA08 direct memory access module for the HC08 Family provides many system functions. Some of these functions are directly related to DMA, such as the ability to perform efficient block transfers. Others are not so obvious, such as the ability to service module interrupts without having to exit the CPU from low power mode. This note demonstrates the advantages of using the DMA08 by illustrating many of its system capabilities in a single code example in which the DMA simultaneously services three separate module interrupts while the CPU is either doing other work or is in low-power mode.

**Order by: AN1711/D**

## "Get Your Motor Running" with the MC68HC708MP16

Electric motors affect almost every aspect of out lives today. With the focus on environmental issues companies are looking for ways to make motors more energy efficient, and electronic control is a key player. Microcontrollers enable control techniques that would have been difficult or impossible with analog circuitry, but all too often the choice of MCU is a compromise. The MC68HC08MP16 has been designed specifically to meet the requirements of low-cost DC servo and AV open loop systems, with particular emphasis on flexible PWM capability. This note discusses its ability to fit painlessly into a variety of different motor control applications.

**Order by: AN1712/D**

## Using M68HC12 Indexed Indirect Addressing

**Rev 1.0**

Indexed Indirect Addressing (IIA) adds an additional level of indirection to standard indexed addressing modes, but is not often found in CPU instruction sets. Its inclusion in the M68HC12, with other features, allows the M68HC12 to compete effectively with RISC processors having faster cycle times. Since IIA mode allows a programmer to include more function in a single instruction, the assembly code is efficient. Fewer instructions mean smaller programs and fewer memory accesses. The result is faster execution times and less program in memory.

**Order by: AN1716/D**

## ITC127 MC68HC705MC4 Motion Control Development Board

In motion control systems, microprocessors are used to generate pulse-width modulated (PWM) signals. However it is highly wasteful of processor time for PWM at higher frequencies to be generated in software. A dedicated hardware module, such as the one found in the MC68HC705MC4, is a far more efficient method. A systems development board for motion control using the dedicated module in the MC68HC705MC4 is presented here. It is designed to provide control signals for driving brush or three-phase brushless DC motors, and to interface directly with ITC122 and ITC132 power output stages.

**Order by: AN1717/D**

## A Serial Bootloader for Reprogramming the MC68HC912B32 Flash EEPROM

The MC68HC912B32 contains 32k bytes of bulk-erasable, byte or word programmable Flash EEPROM memory. Flash EEPROM has significant advantages over EPROM or ROM for both the OEM and the end user, but unlike devices in the M68HC11 family the MC68HC912B32 does not have firmware in bootstrap ROM to allow initial programming of the EEPROM via the SCI port. However it does contain a 2k byte erase-protected bootblock, and this can be used for a bootloader program allowing erasure and programming of the remaining 30k bytes. This note discusses the requirements of a serial bootloader and the implementation of the programming algorithm for the MC68HC912B32.

**Order by: AN1718/D**

## SDRAM System Design Using the MPC106

**Rev 1**

Discusses the implementation of an SDRAM-based memory system using the MPC106. Topics include System Analysis, the MPC106 Memory Controller, SDRAM Component Selection, Board Technology, 'Time-of-Flight', Termination, Clocks, Timing Analysis, The Data Path and Physical Layout. It ends with an overview of an example system.

**Order by: AN1722/D**

## Interfacing MC68HC05 Microcontrollers to the IBM AT Keyboard Interface

Since the inception of the IBM PC platform the keyboard has been its primary input device, and its interface now serves as part of the PC architecture standard. However, in recent years PC hardware engineers have designed other peripheral devices that can be used in place of or in conjunction with the keyboard. This note discusses the hardware and software issues involved in designing applications based on the M68HC05 Family of microcontrollers that can interact with an IBM AT computer via its keyboard interface.

**Order by: AN1723/D**

---

## Implementing SCI Receive and Transmit Buffers in C

In CPU32 devices with a Queued Serial Module (QSM), synchronous communication is provided by the Serial Communications Interface (SCI) part of the QSM. However the SCI buffers a single transmission or reception of 8 or 9 bits, while some applications need data to be transmitted and received as multiple bytes – text strings for example. If the main CPU software passes data directly to and from the SCI it may have to wait before it can write the next byte for transmission or read a received byte, which is inefficient use of the CPU. One solution, described here, is to implement software buffers for the SCI transmitter and receiver, servicing the SCI module via interrupt.

**Order by: AN1724/D**

---

## Initializing SDRAM Parameters for Motorola MPC106-Based Systems

Motorola's MPC106 PCI Bridge/Memory Controller provides a CHRP-compliant bridge between a PowerPC microprocessor family and the PCI (Peripheral Component Interconnect) bus. This document describes the correlation of the programmable SDRAM interface parameters of the MPC106 with typical SDRAM parameters found in manufacturers' data sheets. Parameters for Rev. 4.0 of the MPC106 are described, but the information is applicable with minor adjustment to earlier revisions.

**Order by: AN1725/D**

---

## Using Motorola's Fast Static RAM CAMs on a Media Independent Interface

The transition from 10Base to 100Base Ethernet presents some implementation differences: address filtering in 10Base bridge applications was often performed in software by the host CPU, whereas the higher data rate of 100Base Ethernet does not allow enough time for the host CPU in the bridge to accept or reject frames. One solution is the addition of a Content Addressable Memory (CAM) to the Ethernet Media Access Controller (MAC). CAMs have previously been too expensive to justify their common use, but Motorola's MCM69C232 adopts a different approach to reduce costs. This note illustrates the connection of a CAM between the Physical Interface Device (PHY) and the MAC.

**Order by: AN1726/D**

## Designing PCI 2.1-Compliant MPC106 Systems

Some PCI target devices are not compliant with specifications found in the PCI Local Bus Specification (Revision 2.1). This note describes how best to design PCI-based systems using the MPC106. Specifically, it is concerned with the hold time of the PCI clock. Knowledge of the MPC106, the PCI Local Bus Specification and board layout and routing concepts is assumed.

**Order by: AN1727/D**

---

## Making Low-Distortion Waveforms with the MC68HC708MP16

**Rev 1**

In designing a drive for AC induction motors the aim is to generate the cleanest possible sine waves. Unfortunately the 6-transistor topology commonly used in voltage sourced inverters requires that a "dead time" must be inserted between the turn-off of one transistor in a half-bridge and the turn-on of its complementary device. As a result a distortion is introduced. Now Motorola has developed a sensorless technique to generate correction waveforms, and has integrated this feature into the MC68HC708MP16 microcontroller. For the first time, the benefits of distortion correction will be available to low-cost motor control applications.

**Order by: AN1728/D**

---

## BurstRAM to ZBT RAM

The ability to perform back-to-back read/writes without any intermediate deselect cycles, using new products in the ZBT family of synchronous memories, offers a substantial performance improvement for a variety of platforms which currently use standard BurstRAMs. This note describes some of the footprint changes required to adapt a current BurstRAM socket to a ZBT device. Some of the operational differences are also discussed.

**Order by: AN1729/D**

---

## Digital Amplification of an Analog Signal Using the MC68HC705J1A

This design interfaces an MC68HC705J1A microcontroller to a multiplying digital-to-analog converter (MDAC) to digitally control the gain of an operational amplifier, allowing a mechanical potentiometer to be replaced by a more robust and reliable solution. The MDAC used here is the Analog Devices DAC8043 – a 12-bit, 8-pin serial device. The interface between the MCU and the MDAC is serial; an MCU with Serial Peripheral Interface (SPI) is ideal, but not all M68HC05 MCUs have SPI and a software I/O driver must be used. The MC68HC705J1A is used here to demonstrate the software driver routine.

**Order by: AN1730/D**

## VPW J1850 Multiplexing and Motorola's Byte Data Link Controller (BDLC) Module

With the dramatic increase in the amount of electronics in automobiles, the traditional wiring harness has been replaced by communication buses that allow multiple electronic devices to communicate via shared wiring. The Society of Automotive Engineers has standardized the allowable multiplexing networks within automobiles in three classes, each designed with specific systems in mind. The J1850 architecture is intended for medium speed nodes, and this note discusses its Variable Pulse Width (VPW) multiplexing and Motorola's Byte Data Link Controller (BDLC) module.

**Order by: AN1731/D**

---

## A Universal Serial Bus Gamepad Device using the MC68HC05JB2

The Universal Serial Bus (USB) is a user-friendly interconnection method designed to support consumer, telephony and productivity peripherals for the PC. The standard is implemented in an open software architecture through a base class and a series of horizontal device classes. Supporting all of the classes can be a tedious process for an embedded developer. This application note describes how the Motorola USB Device Firmware Library can be used to develop a USB application belonging to the Human Interface Device (HID) class, including how to set up the USB device information, and integrate the library with custom external hardware controlling firmware.

**Order by: AN1732/D**

---

## Implementing Caller ID Functionality in MC68HC(7)05 Applications

Caller ID is a service that transmits information about a telephone caller, such as a telephone number and name, to a called subscriber. The widespread acceptance of this service in both residential and commercial markets has led to the development of several different types of Caller ID devices such as adjunct boxes, computer peripherals and telephones. This note examines hardware and software issues involved in implementing Caller ID functionality in applications based on Motorola's M68HC(7)05 family of MCUs. Includes a design example of a computer peripheral that can capture Caller ID data and display it on an IBM PC or compatible.

**Order by: AN1733/D**

---

## Pulse Width Modulation Using the 16-Bit Timer

PWM is a technique used to control devices or to provide a variable DC voltage; common applications include motor, lighting and climate controls. In many cases, the added cost and complexity of dedicated PWM hardware cannot be justified, and the software implementation described here may be a viable alternative. The method uses the Output Compare function of the 16-bit free-running timer counter found in a wide variety of Motorola MCUs. Includes flowcharts and program listings.

**Order by: AN1734/D**

## Variations in the Motorola MC68HC05Px Family

**Rev 1**

Motorola's M68HC05 P Family of 8-bit microcontrollers is one of the largest and most widely used. This note clarifies the important differences between the various HC05P devices. It is particularly useful for designers who are familiar with one family member but wish to move to another. The discussion includes Similarities and Comparisons; Pinouts; The A Strategy; Changing from OTP/FLASH to ROM; Changing from Non-A to A Versions; Voltage, Frequency and Temperature Tables; and Development Tools.

**Order by: AN1736/D**

---

## Migrating from the MC68HC705J2 to the MC68HC705JJ7

The MC68HC705JJ7 is less expensive than the MC68HC705J2, yet provides a large number of additional features. This note describes the hardware and software changes required to migrate a design from the MC68HC705JJ7 to the MC68HC705J2.

**Order by: AN1737/D**

---

## Instruction Cycle Timing of MC68HC05JJ/JP Series Microcontrollers

The MC68HC05JJ and MC68HC05JP series microcontrollers have an asynchronous analog interface, and events can occur which are not specifically synchronized to software operations. For example, when sampling the outputs of the two voltage comparators the actual time when the CMP1 and CMP2 bits are read is dependent on bus speed and the instruction being executed. The timing within an instruction is not normally published; this note describes the hardware timing of the JJ/JP series and provides a method that allows the user to make individual timing measurements – the method can also be applied to other M68HC05 MCUs.

**Order by: AN1738/D**

---

## Applications Using the Analog Subsystem of MC68HC05JJ/JP Series Microcontrollers

The MC68HC05JJ/JP series of MCUs presents a unique combination of traditional digital peripherals and simple analog components which can be used to implement a variety of special functions. Features include a pair of analog comparators; input channel multiplexer; a current source; and a temperature sensing diode. Their simple nature requires very little die area yet they provide capability normally found in more expensive MCUs. This note presents a range of applications. Topics include an analog subsystem overview, voltage comparators, current source/discharge, analog multiplexers, analog power-up considerations, A/D conversion, and a design check list.

**Order by: AN1740/D**

## In-Circuit and Emulation Considerations for MC68HC05JJ/JP Series Microcontrollers

The MC68HC05JJ and MC68HC05JP series of MCUs presents a unique combination of traditional digital peripherals and simple analog components which can be used to implement a variety of special functions. However adding the capability of low-level analog signals to a digital IC creates issues not normally considered in MCU designs, for example circuit board design and emulation considerations connected with the level of analog accuracy. This note discusses issues which must be considered both in the end application and while developing software using an emulator such as the MMDS05.

**Order by: AN1741/D**

---

## Programming the 68HC705J1A In-Circuit

The low-cost MC68HC705J1A microcontroller does not have a built-in function to allow in-circuit programming, which may be necessary when sections of code such as lookup tables or calibration values need to be entered after assembly. This note describes how in-circuit programming can be achieved using previously programmed 'bootloader' code.

**Order by: AN1742/D**

---

## Scrolling Message Software

Many MCU applications use displays such as LCD or LED panels to present data, and modern displays are an efficient and affordable way for MCUs to communicate to the outside world. However one limitation is the amount of information that can be presented at any one time. To output a message that is longer than its display, MCU software needs a method of scrolling information across the screen. The method should be divided into independent tasks, allowing for normal paced-loop program execution. This note describes such a technique using a segmented display; it can easily be adapted to other types with a few changes in software.

**Order by: AN1743/D**

---

## Resetting Microcontrollers During Power Transitions

A simple function such as resetting an MCU during the application or removal of power can cause many problems if not handled properly. Symptoms can range from a slight delay in MCU response, to very erratic and inconsistent behavior, to total system failure. This document discusses the main issues in respect of HC05, HC08 and HC11 devices, and leads to a safe and reliable approach to transitioning power.

**Order by: AN1744/D**

---

## Interfacing the HC705C8A to an LCD Module

More and more applications use Liquid Crystal Displays (LCDs) to display data. This note describes the hardware and software interface required to display information from the MC68HC705C8A. It uses an Optrex DMC16207 LCD module, incorporating a Hitachi HD44780 LCD driver which provides the LCD segment waveforms and a simple parallel port interface. Circuitry and example code are also given to demonstrate a means of providing pre-defined messages from EPROM memory; the code can be modified easily to take SPI and SCI data and display it on the LCD module.

**Order by: AN1745/D**

---

## Migrating from the MC68HC705K1 to the MC58HC705KJ1

Motorola provides two different devices to which MC68HC705K1 (K1) applications can migrate: the MC68HC705KJ1 (KJ1) and the MC68HC805K3 (K3). The main differences between the two options are pin compatibility and cost. The KJ1 is not pin for pin compatible with the K1 but is about 70% of its cost, while the K3 is pin compatible but is roughly 90% of the cost. This application note describes the technical differences between the K1 and the KJ1.

**Order by: AN1746/D**

---

## Migrating from the MC68HC705K1 to the MC68HC805K3

Motorola offers two devices that allow easy migration of MC68HC705K1 applications. Depending on the specific design, system enhancements and cost considerations, two different migration paths are open. The MC68HC805K3 is pin for pin compatible with the MC68HC705K1 and is roughly 90% of the cost; the MC68HC705KJ1 is not pin for pin compatible, but is roughly 70% of the cost. This note discusses the differences between the two devices, plus some additional features of the MC68HC805K3.

**Order by: AN1747/D**

---

## Building a Universal Serial Bus Keyboard Hub Using the Motorola MC68HC(9)08KH12

The 'hot swap' capability of the Universal Serial Bus (USB) allows PC users to plug in peripherals such as keyboards and have them available immediately without having to reboot the computer; the tangle of wires is reduced and there is no need to configure DIP switches or load software drivers. This note reviews the design of a legacy type keyboard, discusses the MC68HC(9)08KH12 and the USB module operation, describes high-speed and low-speed USB connections, provides in-depth instruction on the programming of the 12 Kbytes of FLASH memory, and outlines the steps required to construct a keyboard hub.

**Order by: AN1748/D**

---

## DSP563xx Port A Programming

The DSP56300 expansion port – Port A – allows the memory space accessible to the DSP core, or the memory-mapped I/O, to be expanded. The interface is straightforward, and external memory is easily and quickly retrieved using DMA or simple MOVE commands. This note describes the hardware and software configurations required to connect the DSP core to external SRAM and DRAM, examples of moves to and from external memory, and examples of DMA accesses.

**Order by: AN1751/D**

## Data Structures for 8-bit Microcontrollers

Data structures describe how information is organized and stored in computer systems. Although they are usually presented in the context of large computers, the same principle can be applied to embedded 8-bit processors, where their efficient use can improve both the dynamic (time based) and static (storage based) performance of microcontroller software. The data structures presented here will be useful in the development of MCU software, and may be applied to an application in many different ways.

**Order by: AN1752/D**

---

## Implementing a FLASH Memory System in an MC68HC711E9 Design

FLASH technology offers several advantages for an M68HC11 microcontroller design, including field updates, lower power consumption and increased memory density. However there are some significant obstacles in implementation, arising mainly because FLASH requires a programming algorithm. This note describes a single board computer design which uses a FLASH device as its main program and data storage medium. The emphasis is on the hardware and firmware FLASH programming techniques. An example of a retrofit design is also included to show how to convert an existing EPROM-based design.

**Order by: AN1753/D**

---

## Interfacing the MC68HC705J1A to the DS1620 Digital Thermometer

Most temperature sensors transduce their reading to an electrical signal, providing a voltage level related to the measured temperature; this voltage is typically converted to a digital number by an A/D converter, and processed by an MCU. The Dallas Semiconductor DS1620 is a single-chip solution that reads temperature and converts it to a 9-bit digital value readable via a serial interface. It also provides three thermal alarm outputs for thermostatic control. This note describes the interface between the DS1620 and Motorola's MC68HC705J1A MCU, in an application measuring temperature in the range -55°C to +125°C. A software driver is created to provide the appropriate serial bus signals.

**Order by: AN1754/D**

---

## Interfacing the MC68HC705C8A to the DS2430A 256-bit 1-Wire EEPROM

Embedded applications increasingly demand non-volatile memory storage for data such as reprogrammable calibration constants, power down information in consumer electronics, ID number storage and telephone number memories. This note describes the interface between an MC68HC705C8A microcontroller and the DS2430 1-Wire™ 256-bit EEPROM from Dallas Semiconductor Corporation; the 1-Wire interface reduces the overhead of control, data, address and power pins. Includes circuitry and example code.

**Order by: AN1755/D**

## Add a Unique Silicon Serial Number to the HC05

Many embedded systems require serial numbers to help track printed circuit boards, identify nodes on a network or provide security access. The Dallas Semiconductor DS2401 Silicon Serial Number provides a unique, factory-lasered, 64-bit ROM number. Its address bus structure uses a 1-Wire™ interface to reduce the pin overhead. The DS2502 is a similar device, but with the addition of 1024 bits of user-programmable EPROM for storage of calibration constants, access codes and so on. This note describes the interface between an HC05 MCU and the DS2401, with circuitry and example code. Application-specific functions can easily be added for use with the DS2502.

**Order by: AN1757/D**

---

## Add Addressable Switches to the HC05

Describes the interface between an HC05 microcontroller and the DS2405 addressable switch from Dallas Semiconductor Corporation. The address bus structure uses a 1-Wire™ interface to reduce the overhead of control, data, address and power pins. The DS2405 allows an identification to be assigned to a node, with the additional control capability of an open-drain N-channel MOSFET which can be turned on or off via the 1-Wire bus. Includes circuit and example code, based on the MC68HC705J1A.

**Order by: AN1758/D**

---

## Add a Non-Volatile Clock to the MC68HC705J1A

Many embedded systems need measurement of time. This can be achieved internally by some MCUs that have an on-chip real-time clock; even so, for date, month and leap year measurement, substantial amounts of bandwidth and code space are required. The Dallas Semiconductor DS1307 64x8 real-time clock provides calendar and timekeeping functions, along with 56 bytes of non-volatile RAM. With its 2-wire interface timekeeping can be managed easily. This note describes the interface between the DS1307 and the MC68HC705J1A. Includes circuit and example code.

**Order by: AN1759/D**

---

## Interfacing the AD8402 Digital Potentiometer to the MC68HC705J1A

The digital potentiometer allows many applications of mechanical trimming potentiometers to be replaced by a solid-state solution. It provides several benefits over the mechanical device, including compact size, freedom from the effects of shock and vibration, and the ability to withstand oil, dust, temperature extremes and moisture. This note describes the interface between the MC68HC705J1A MCU and the AD8402 from Analog Devices, Inc., to create both the rheostat (2 terminal) and potentiometer divider (3 terminal) configurations for various analog circuits. Includes circuits and example code.

**Order by: AN1760/D**

## Interfacing the MC68HC705C8A to the X76F041 PASS SecureFlash

The increased frequency of code pirating and data tampering makes secure access to system code and data a requirement for embedded systems. The X76F041 Password Access Security Supervisor (PASS) from Xicor, Inc., provides the ability to password protect sensitive memory. It also contains non-volatile memory which can be used for system calibration constants, user information such as telephone numbers, and code patches. This note describes the interface between the X76F041 and the MC68HC705C8A, and includes circuitry and example code.

**Order by: AN1761/D**

---

## Automatic Contrast Control of LCD Displays Using the MC68HC708LN56 Microcontroller

In applications such as battery powered systems, where the power supply voltage can vary, the contrast of a Liquid Crystal Display (LCD) can change over time. This note describes how to implement automatic contrast control of an LCD in order to maintain constant contrast. It is achieved in software on the MC68HC08LN56 microcontroller by using the A/D converter in conjunction with the LCD controller. Source code for implementing the system is included, and the discussion covers factors such as how the amount of multiplexing, type of bias and voltage levels can affect the contrast.

**Order by: AN1762/D**

---

## Driving LCD Displays Using the MC68HC705L16 Microcontroller

This note describes how to use the MC68HC705L16 microcontroller as a Liquid Crystal Display (LCD) controller/driver. All LCD control and drive functions are performed by a single chip, which also retains all of the normal functionality of an MCU. Includes a description of the voltages and waveforms needed to control the LCD panel, and source code for controlling a multiplexed display.

**Order by: AN1763/D**

---

## DSP56300 Enhanced Synchronous Serial Interface (ESSI) Programming

**Rev 1**

The Enhanced Synchronous Serial Interface (ESSI) provides a full-duplex serial port. It consists of independent transmitter and receiver sections and a common ESSI clock generator. Three transmit shift registers enable it to transmit from three different pins simultaneously. Each DSP56300 family device includes two ESSIs, and so can accommodate a total of six ESSI transmitters for 6-channel surround sound applications. This application note describes the pins and registers that control ESSI operation, and describes its operation using small sections of code to illustrate practical programming guidelines.

**Order by: AN1764/D**

## Ethernet Configuration for a ColdFire Evaluation Board to Download Files from a Personal Computer

This note explains how to use a personal computer to transfer code to a target ColdFire evaluation board. Compared to serial transfers, downloading code through Ethernet reduces time and speeds up the evaluation process. Configuring Ethernet downloads is simple and fast, however it is important to follow all instructions carefully to ensure correct configuration.

**Order by: AN1765/D**

---

## Using Registered SDRAM DIMMs with the MPC106

The use of registered SDRAM modules (registered DIMMs) can eliminate many of the design problems associated with the increased capacitive loads of larger memory arrays. Registered SDRAM modules are a slight variation of JEDEC-standard, unbuffered, 168-pin memory modules, in which a registered driver has been inserted between the DIMM pins and certain control signals. The MPC106 PCI Bridge/ Memory Controller was not designed to directly control registered DIMMs, but with some software and hardware modifications it can support these modules with minimum cost. This note explains how.

**Order by: AN1768/D**

---

## Designing a Minimal PowerPC System

Describes how to design a small, high-speed Motorola PowerPC-based system, using any member of the MPC60x or MPC7xx family. To keep the design as simple as possible, only the most basic features necessary to run a debugger program are included. The design also uses a programmable ASIC to provide the necessary controls for a block of RAM, ROM and access to I/O, in place of the traditional MPC106 memory/PCI/cache controller.

**Order by: AN1769/D**

---

## In-Circuit Programming of FLASH Memory in the MC68HC908GP20

Describes two methods of programming the 20 Kbytes of on-board FLASH memory in the MC68HC908GP20 microcontroller. It explains how the FLASH is programmed and erased in-circuit in both user and monitor modes; how the control and protection registers are programmed; and additional considerations for this type of memory. A sample program is included which executes programming routines from RAM. The necessary functionality of a host program is also described – the program used here is available as a free download from the Motorola web site.

**Order by: AN1770/D**

---

## Precision Sine-Wave Tone Synthesis Using 8-bit MCUs

Many products containing microcomputers – including cell phones, base stations, repeaters, SLICs and cordless telephones – also need precision tone generators for functions such as DTMF signalling,

call progress tones, CTCSS and user interface chimes. While off-the-shelf components are available for these functions the cost can be greatly reduced by using the MCU to synthesize the tones. This note presents basic tone synthesis techniques and illustrates their implementation using the HC05, HC08, HC11 and HC12 Families of MCUs.

**Order by: AN1771/D**

## Efficient Compilation of Bit-Exact Applications for DSP563xx

Many of the standard algorithms in wireless and wireline communications – such as GSM speech coders and the G.723.1 and G.729a coders – use 16-bit, bit-exact C code and corresponding test vectors. They employ ANSI C integer data types and implement 16-bit fractional arithmetic operations. To specify the fractional arithmetic model the ANSI C code uses a set of subroutines that implement basic fractional operations. Theoretically it requires little effort to compile an algorithm for any DSP for which an ANSI C compiler is available; in reality, an efficient implementation requires some modifications to the C code before the compiler can compile it effectively.

**Order by: AN1772/D**

## ZBT Primer

The emergence of higher bandwidth networking systems and infrastructures has encouraged the development of SRAM solutions with faster data throughput. The Zero Bus Turnaround (ZBT) architecture has been standardized by Motorola, IDT and Micron, and eliminates bus latency by providing a more efficient use of the system bus. This note introduces the architecture and potential applications of ZBT, including issues associated with frequency, bandwidth, bus contention and temperature effects.

**Order by: AN1773/D**

## Interfacing the MC68HC912B32 to an LCD Module

**Rev 1.0**

More and more applications need liquid crystal displays (LCDs) to communicate effectively to the outside world. Some LCD suppliers provide only the LCD glass, so that the waveforms needed to directly drive the LCD segments must be generated by the microcontroller. Others provide an LCD module which has all glass and segment drives packaged in one circuit board. This note describes the hardware and software of an interface between the MC68HC912B32 MCU and an LCD module from Optrex Corporation, using a simple parallel port. Circuitry and example code are also given to demonstrate a facility to send pre-defined messages from memory to the display.

**Order by: AN1774/D**

## Expanding Digital Input with an A/D Converter

Many microcontroller applications require digital input and arbitration. For example, determining which key of a keypad has been pressed is commonly achieved by connecting a switch matrix to a series of digital inputs, and reading a digital input data register. Whilst this method is easily implemented, it requires the use of the MCU's parallel port pins, which may also be needed for other purposes. By using the analog to digital converter (ADC), connected to a resistor ladder, user input can be processed more efficiently. This note includes an example based on the MC68HC705P6A.

**Order by: AN1775/D**

## Stereo Audio Transmission over the CAN Bus Using the Motorola MC68376 with TouCAN Module

**Rev 1.0**

Stereo audio transmission is an interesting but unusual application for a Controller Area Network (CAN) system, and is described in this application note as an example to illustrate the use of the TouCAN module in the MC68376 microcontroller; use of the QADC and QSPI modules is also demonstrated. C source code for the TouCAN module and all audio transfer is provided. Stereo audio data is sampled by the QADC of one MC68376 and transmitted on the CAN bus by the TouCAN module to a second MC68376, which buffers the data and outputs to an external DAC for reproduction of the analog signal.

**Order by: AN1776/D**

## MPC8xx to BurstRAM Interfacing

In many applications, DRAM provides sufficient performance for MPC8xx PowerPC systems, including the MPC860, MPC823, MPC850, MPC801 and MPC8260. However in cases where performance must be optimized, or where cache performance is poor, it may be desirable to manipulate data in fast external memory. Current MPC8xx parts are available with external bus frequencies up to 50MHz, but future generations will include 66MHz and 100MHz and will require fast static memory to achieve optimum performance. This note describes how to interface an MPC8xx processor to the MCM69F536C (32K x 36) and the MCM69F618C (64K x 18) synchronous Fast Static RAMS.

**Order by: AN1777/D**

## Using the MIOS on the MPC555 Evaluation Board

This information is intended to help the microcontroller developer quickly set up the MPC555 Evaluation Board, which was developed jointly by Motorola and ETAS, Inc. The board allows the user to attach logic analyzers and debugging equipment to the MPC555, provides additional external memory and FLASH, and replicates ports and buses. The application note includes overviews both of the MPC555 microcontroller and of the evaluation board, provides setup and software installation instructions, and discusses two simple applications in detail as a way of introducing the board.

**Order by: AN1778/D**

## Using the MCM69D536/MCM69D618 NetRAMs with Different Speed Computing Elements

A dual port RAM is often used to allow two computing elements to communicate, even if they are running at different speeds. Motorola's MCM69D536 and MCM69D618 synchronous dual port NetRAMs

are well suited to this purpose, if the clock driving the slower computing element is derived from the clock driving the faster device. This note explains the interface.

**Order by: AN1779/D**

## DSP563xx HI32 as a PCI Agent

The Host Interface (HI32) is a fast 32-bit wide parallel host port that can connect directly to the host bus. It is a standard peripheral on DSP563xx family derivatives, and supports a variety of standard buses to provide a glueless connection to a number of industry-standard microcomputers, microprocessors, DSPs and DMA Controllers. HI32 runs in three different modes – this note discusses the PCI mode, and includes a Data Scatter and Gather application using a DSP56301 running on a DSP56301ADM board.

**Order by: AN1780/D**

## Booting DSP563xx Devices Through the Serial Communication Interface (SCI)

The DSP563xx bootloader code allows the DSP to load an application program and data through the Serial Communication Interface (SCI) to X, Y and P memory, and to begin executing the program on reset of the DSP. It also allows various DSP control registers to be programed before executing the downloaded program. This note describes how to use the bootloader code, including resetting the DSP, and downloading and running the bootloader. It contains overviews of the SCI, the DSP mode pins and operation modes, and the internal bootstrap ROM code.

**Order by: AN1781/D**

## Converting DSP56303 Designs to DSP56307 Designs

This application note details the device differences that must be considered when redesigning a system based on the DSP56303 for use on the DSP56307. Required changes are discussed under the headings of Hardware and Layout, Precessor Type ID, Control Registers, Memory Switches, and Bootstrap and Operating Modes. In addition, the new Enhanced Filter Coprocessor of the DSP56307 allows some optional enhancements, which are summarized here.

**Order by: AN1782/D**

## Determining MCU Oscillator Start-up Parameters

**Rev 2.0**

Many microcontrollers incorporate an inverting amplifier for use with an external crystal or ceramic resonator in a Pierce oscillator configuration. This application note describes how to calculate the minimum gain (transconductance) of the amplifier that is required to ensure oscillation with specific external components, and how to measure the transconductance to establish whether the minimum gain requirement has been met.

**Order by: AN1783/D**

## DSP563xx HI32 PCI Functions

This document describes the DSP563xx_HI32_PCI framework, a set of functions in C that enable the user to operate any DSP56300 family device with an HI32 interface through the PCI bus of a host running Microsoft Windows 95/98. The framework also includes a virtual device driver to support these functions.

**Order by: AN1788/D**

## Programming the CS4218 CODEC for Use with DSP56300 Devices

This application note presents an overview of the CS4218 16-bit Audio CODEC, and discusses how to program it for use with members of Motorola's DSP56300 Family. The CODEC performs analog-to-digital and digital-to-analog conversion, plus filtering and level setting, and is integrated with current DSP563xx evaluation modules. A sample program is included to show in detail the use of the Enhanced Synchronous Serial Interface ports (ESSI), and how the DSP's ESSI ports interface, initialize and transport data between the DSP and the CODEC.

**Order by: AN1790/D**

## Using the QAD64 Module on the M68F375 Microcontroller

This application note provides material for using the MPFB1632 Modular Platform Board and the MPBF375BGA evaluation board, highlighting the effective use of the QADC64 Queued Analog-to-Digital Converter module, typically used in automotive applications which require analog-to-digital conversion.

**Order by: AN1791/D**

## Using an MC68HC908MR24 in Place of an MC68HC708MP16

The MC68HC908MR24 is an improved version of the MC68HC708MP16 microcontroller, and was introduced to create a new family of products for motor control applications. It is designed to be a drop-in replacement for the earlier device, with the exception of the function of one pin. From a software point of view there are a number of address changes. This note documents the differences between the two devices to help users wishing to migrate from the MP16 to the MR24.

**Order by: AN1792/D**

## PowerPC Backside L2 Timing Analysis for the PCB Design Engineer

The backside Level-2 (L2) interfaces on the MPC750 and the G4 processors dramatically increase their performance. Since the L2 design basically connects the processor's L2 controller to the memory SRAMs, the main task for the board designer is to determine what board propagation delays will provide sufficient setup and hold margins for a given target frequency. This note discusses how to

determine the propagation delay restrictions for the backside L2 interface of PowerPC processors, and a method for optimizing the setup and hold margins by using clock offsets.

**Order by: AN1794/D**

## Designing G4 Systems

**Rev 1**

This application note describes the differences between the MPC60x bus and the native bus mode of the G4 processor – a new bus interface that is derived from the MPC60x bus. It also looks briefly at the 360-pin G4 processor, which is pin compatible with the MPC750 microprocessor. The document assumes a working knowledge of the MPC750 microprocessor and the MPC60x bus protocol.

**Order by: AN1795/D**

## Programming the Thermal Assist Unit in the MPC750 Microprocessor

The Thermal Assist Unit (TAU) on the MPC750 Family of microprocessors is an on-chip sensor that measures the instantaneous junction temperature of the device. This application note describes example software for programming it. It includes an overview of the TAU, a description of its operation, and programming examples to access it, determine the junction temperature, and calibrate the TAU to a known temperature.

**Order by: AN1800/D**

## Connecting the MCF5307 to 168-pin Unbuffered SDRAM DIMMs

**Rev 0.1**

By using inexpensive, fast synchronous DRAM (SDRAM) devices, many embedded system manufacturers are realizing lower board costs. Due to their large volume use in standard PCs, SDRAM devices are currently the least expensive memory available, as well as being one of the fastest memory types. This document shows how to interchange various standard SDRAM dual-in-line memory modules in a ColdFire MCF5307 design without rerouting the board. The MCF5307 integrates a Version 3 core with an 8 Kbyte unified cache, 4 Kbyte SRAM, an asynchronous/synchronous DRAM controller and various other popular embedded peripherals.

**Order by: AN1802/D**

## DSP56307 Port A-to-HI08 Interface

In multiple DSP applications, such as DSP resource boards and voice transcoder cards, a central microcontroller is typically used to schedule and assign the DSP workload across multiple devices. However the DSPs themselves can also perform scheduling; in this case the DSPs are organized in a tier, with a group of master DSPs each controlling a subset of slave DSPs. In the DSP56303, DSP56307 and DSP56309, the Port A bus on a master DSP can control the HI08 host interface on the slave DSP. This note describes a straightforward interface between master and slave DSP56307s.

**Order by: AN1804/D**

## Using Motorola's Dual Port NetRAMs for Interprocessor Communication in a Datacomm Application

Dual port static RAMs are commonly used for communications between computing elements in a multiprocessor system. This application note describes the use of Motorola's dual port NetRAM products to provide a bidirectional pathway between an MPU controlling a datacomm switch and an ASIC that performs transformations on the data packets passing through the switch. In this particular example, the MPU is an MPC755 that uses its backside Level-2 (L2) cache port to access the NetRAM.

**Order by: AN1807/D**

## DSP56300 HI08 Host Port Programming

The HI08 Host Port is a DSP56300 Family peripheral that provides a byte wide, full duplex, double buffered parallel port for communication with a host processor. Various programmable options enable a glueless connection between the DSP and several industry-standard processors and microcontrollers that are commonly used as system hosts. This note contains information about programming the HI08 Host Port, as a supplement to the user's manuals. Each section refers to the relevant signals and registers.

**Order by: AN1808/D**

## Developing DSP56364 Software Using the DSP56362 EVM

This application note is intended for programmers who wish to develop DSP56364 software on a DSP56362EVM, and addresses the many functional, I/O and memory map differences. The evaluation module can be used to develop both DSP56364 RAM-based solutions and custom mask ROM codes.

**Order by: AN1810/D**

## Using the HC912B32 to Implement the Distributed Systems Interface (DSI) Protocol

System design requirements are continually changing as systems become increasingly complex. In conventional systems, where sensors and actuators are connected directly to a microcontroller (MCU), the number of pins on the MCU limits system expansion. Extra cost may be incurred if a different or additional MCU is required. An alternative approach is to use a distributed bus architecture. The Distributed Systems Interface (DSI) is a master/slave system which allows both sensors and actuators to be connected as slaves to the same bus. This note provides an overview of the DSI and describes the hardware and software of a demonstration system.

**Order by: AN1816/D**

## MMC20xx M•CORE OnCE Port Communication and Control Sequences

The on-chip emulation (OnCE) port in Motorola's M•CORE M200 core is a JTAG-like serial interface. An external Command Controller can use the port to communicate with and control an M•CORE

M200xx core. In addition to other tasks, the Controller can cause the core to stop executing at a predefined instruction or data fetch, or even to program a non-volatile memory device connected to the core. This note describes the specific serial command sequences a command controller should present to a OnCE port to control an M200 core.

**Order by: AN1817/D**

## Software SCI Routines with the 16-bit Timer Module

Many applications that communicate to off-board devices require an asynchronous serial link. An MCU with a Serial Communications Interface (SCI) can provide this communications functionality. However in many applications, an MCU must be used that does not have an SCI, and it must be provided through software control of existing modules. A “bit-banged” approach (as documented in AN1240/D) is convenient, but requires dedicated software overhead while transmitting and receiving data. Through the use of the 16-bit free-running counter, the HC05 and other MCU families can provide an interrupt driven software SCI with minimal software overhead.

**Order by: AN1818/D**

## Software I²C Communications

The I2C protocol is a 2-wire serial communications interface that is implemented on numerous microcontrollers and peripheral devices. However many MCUs do not have an I²C module, yet are required to communicate with devices that do. This note describes a method of communicating on an I²C bus by controlling I/O pins; this “bit-banged” technique can be implemented on any Motorola MCU. Includes an overview of the interface and sample code for an MC68HC705J1A.

**Order by: AN1820/D**

## Programming and Erasing FLASH Memory on the MC68HC908AS60

An innovative type of FLASH non-volatile memory in Motorola’s 8-bit M68HC08 Family of microcontrollers allows in-circuit reprogramming over the entire automotive specification range. Advantages include in-system code revision, EPROM replacement as a reusable code development platform, quick time to market, and no obsolete inventory. This application note explains how to use the FLASH on the MC68HC908AS60, and provides example software for program and erase operations. The reprogramming algorithms are written in both M68HC08 assembly code and in C.

**Order by: AN1827/D**

## Software Differences Between the DSP56002 and the DSP56303

The software differences between the DSP56002 and the DSP656303 must be considered when a system based on the DSP56002 is redesigned to use the DSP56303. This note discusses the differences in detail. It does not describe a specific application, but summarizes the differences between two DSP products as a convenient reference for designers and programmers who are migrating from one to the other.

**Order by: AN1829/D**

## Hardware Differences Between the DSP56002 and the DSP56303

The hardware differences between the DSP56002 and the DSP56303 must be considered when a system based on the DSP56002 is redesigned to use the DSP56303. This note discusses the differences in detail. It does not describe a specific application, but summarizes the differences between two DSP products as a convenient reference for designers and programmers who are migrating from one to the other. The DSP56300 is a rich product family, and this document also addresses some changes that enable DSP56303 designs to migrate easily to other DSP56300 devices.

**Order by: AN1830/D**

## Using MC68HC908 On-Chip FLASH Programming Routines

Describes how to use the routines that are stored in ROM in the MC68HC908GR8, MC68HC908KX8, MC68HC908JL3/JK3, and the MC68HC908JB8 microcontrollers. These routines are used to program, erase and verify FLASH memory, and in most cases may be accessed either in user mode or in monitor mode. There are additional routines in the MC68HC908KX8 to trim the internal clock generator, which are also described here. The document describes the method of calling each routine, and specifies its action and returned data. A program is included which can be configured for use in any of the devices.

**Order by: AN1831/D**

## Visual Debug for MPC60x

When a new MPC60x system is in the initial debug phase, there is often no simple means of providing feedback to the designer. The circuit described here was implemented to overcome this. It enables two hex digits to be displayed by writing or reading memory locations; no initialisation or setup is required. The design uses only the address phase of the MPC60x bus and requires the target system to terminate the cycle. It consists of three 22v10 PLDs, for which full logic equations are provided, plus two 7-segment displays.

**Order by: AN4000/D**

## Testing and Programming MCM2814 Devices Using and M68HC11EVBU Universal Evaluation Board

The M68HC11EVBU Universal Evaluation Board is a low cost tool for debugging and evaluation of MC68HC11 A-series and E-series microcontrollers. This note describes its use with the PCbug11 software package to provide a user friendly system for testing a single MCM2814 EEPROM device, and to program multiple devices.

**Order by: AN4003/D**

## ±2g Acceleration Sensing Module Based on a ±40g Integrated Accelerometer

**Rev 4**

Micromachined accelerometers with built-in signal conditioning and calibration, such as Motorola's MMAS40G10D, are widely used in automotive systems such as airbag modules. Other automotive applications for accelerometers include active suspension, but for such applications a ±2g device is required and these are generally not available in large quantities and at low cost. This simple and inexpensive circuit demonstrates the use of the ±40g MMAS40G10D for sensing acceleration over a ±2g range. The design is based on the '40G-2G' evaluation board.

**Order by: AN4004/D**

---

## Thermal Management and Mounting Method for the PLD 1.5 RF Power Surface Mount Package

At power levels less than 2W, thermal management of surface mount components can be achieved through the contact pads on the printed circuit board; dissipation of the device is a function of the pad size. However at power levels in excess of 2W alternative techniques are necessary to remove the heat dissipated in the device, in order to maintain device junction temperature within the range specified for reliable operation. This note discusses the use of solder-filled thermal vias in the PCB, the recommended method of thermal management for the PLD 1.5 package (case 466-02).

**Order by: AN4005/D**

---

## A Medium Scale PABX

**Order by: AN-HK-08/H**

---

## MC68HC05L9 Microcomputer Applications Demo Board

**Order by: AN-HK-10/H**

---

## MC68HC05F6 Tone Pulse Dialer

**Order by: AN-HK-12/H**

---

## MC68HC05L10, an Enhanced Version of L9 for Handheld Equipment Applications

The MC68HC05L10 is an enhanced version of the MC68HC05L9, and a new member of the DRAGONKAT™ Family of MCUs. It is designed for use in handheld equipment such as organizers, meter readers, inventory checkers and personal communication products, where low power consumption and system optimization are critical. This note highlights the similarities and differences between the two MCUs, and describes software routines on new features such as keyscan, enhanced LCD drive and memory management unit (MMU).

**Order by: AN-HK-13A/H**

## MC68HC05L11 Hand-Writing Applications

**Order by: AN-HK-15/H**

---

## MC68HC05F2 DTMF Output Low Voltage Active Filter

**Order by: AN-HK-17/H**

---

## MC68HC05SR3, MC68HC705SR3 Design Notes

Provides design information as an aid to using the MC68HC05SR3 and MC68HC705SR3 microcontrollers, including overviews on typical applications such as PC keyboards, air conditioners and cordless phones.

**Order by: AN-HK-22/H**

---

## MC6805R3, MC68HC05SR3 Technical Comparison

This technical note documents the differences between the HCMOS MC68HC05SR3 and the NMOS MC6805R3. Although the devices are pin compatible, minor modification to existing R3 software is required because of differences in memory offset and instruction cycles.

**Order by: AN-HK-23/H**

---

## Software Emulation of DDC1 Hardware using HC05BD3

Data Display Channel (DDC) is a standard for monitor devices defined by VESA in 1994. There are two levels: DDC1 is a unidirectional communications protocol in which the monitor continuously sends out Extended Display Identification Data (EDID) to the host computer, while DDC2 is a bidirectional protocol between host and monitor, based on the IIC protocol. This application uses the low cost MC68HC05BD3 microcontroller to implement DDC1 features in software.

**Order by: AN-HK-24/H**

---

## Using the MC68HC908GP32 in Place of MC68HC908GP20

Describes the differences between the MC68HC908GP32 and the MC68HC908GP20 microcontrollers, highlighting the areas where the user must consider whether the GP32 should in fact be used in existing applications that use the GP20. Includes lists of relevant technical literature and development tools for the two devices.

**Order by: AN-HK-31/H**

## In-Circuit Programming of FLASH Memory in the MC68HC908GP32

The MC68HC908GP32 FLASH memory can be programmed or erased using routines running either in User mode or Monitor mode. This note describes a simple method of in-circuit programming, using Motorola's SPGMR08 Serial Programmer. The necessary high voltage is generated by an internal charge pump.

**Order by: AN-HK-32/H**

---

## In-Circuit Programming of FLASH Memory in the MC68HC908JL3

The 4k bytes of FLASH memory in the MC68HC908JL3 can be programmed or erased using routines running in either User mode or Monitor mode. A high voltage input is not required for either operation as it is generated by the internal charge-pump. In circuit programming allows the device to be programmed or erased with the device installed in the target system. This application note describes a simple method of in-circuit programming the FLASH memory of the MC68HC908JL3 and its variants.

**Order by: AN-HK-33/H**

---

## Bi-Directional Data Transfer Between MC68HC11 and MC6805L3 Using SPI

The powerful Serial Peripheral Interface available on many Motorola MCUs is implemented in 2 forms (the HCMOS families support only Level 1, Level 2 is implemented only on HMOS processors). Both levels communicate easily with each other, but Level 2 has additional capabilities including asynchronous communication. This note describes a method of achieving synchronous communication between levels 1 and 2, and explains the on-chip differences in SPI implementation.

**Order by: ANE405/D**

---

## MC68HC11 Implementation of IEEE-488 Interface for DSP56000 Monitor

The original purpose of this IEEE-488 (GPIB) interface to the Motorola DSP56000 Digital Signal Processor was to allow development of DSP56000 software and hardware on an HP9836 engineering workstation. The design is based on the MC68HC811A2; component count is minimized, and the MCU's 2K bytes of on-chip EEPROM are sufficient to avoid the need for external memory. The MCU runs in expanded mode, using the external bus to communicate with the DSP host port which is configured to appear as a group of MC68HC11 external memory locations.

**Order by: ANE415/D**

---

## MC68HC05B4 Radio Synthesizer

Synthesis of the local oscillator in a superheterodyne radio provides many advantages over mechanical tuning, including accuracy, stability and storing often-used frequencies. In this application, an MC145157 CMOS Synthesizer is controlled by an MC68HC05B4 MCU – the software is mask programmed in parts marked 'MC68HC05B4 DEMO', but could alternatively be programmed into an MC68HC805B6. A 6-digit LCD driver is controlled through the Serial Communications Interface, while the standby mode is used to eliminate interference with the radio.

**Order by: ANE416/D**

---

## Use of the MC68HC68T1 RTC with M6805 Microprocessors

The MC68HC68T1 Real Time Clock with RAM (20 bytes) communicates through a serial port, making it ideal for use with single-chip MCUs; additional features include Watchdog and Power Fail Detection. This example software and MC68HC805C4-based circuit demonstrate the creation of an LCD alarm clock. The code includes routines to use either the MCU's SPI port or lines of a parallel port – it could be used in any 6805 microprocessor, with a small change for HMOS devices.

**Order by: ANE425/D**

---

## An MC68030 32-bit High Performance Minimum System

Circuit and description of a high-performance 32-bit system using the fast synchronous bus interface of the MC68030 to access RAM with a two clock-cycle read and write bus-cycle. Uses commercially available memory devices, and standard FAST TTL interface logic for address decode. An MC68681 DUART provides two RS-232 serial ports; an MC68230 PI/T provides parallel I/O and 24-bit timer. Designed for 20MHz operation, with future 25MHz and 30MHz upgrades possible with faster memory devices.

**Order by: ANE426/D**

---

## Digital Sine-Wave Synthesis Using the DSP56001/ DSP56002

**Rev 2**

With the introduction of high-speed, high-precision digital signal processors, stable and low-distortion sine waves of any frequency can be produced digitally in communication and control applications. This document describes three look-up table methods for sine wave generation using the DSP56001. Total Harmonic Distortion (THD) performance and Maximum Synthesizable Frequency (MSF) are given in each case.

**Order by: APR1/D**

---

## Digital Stereo 10-Band Graphic Equalizer Using the DSP56001

The theory of the Infinite Impulse Response (IIR) algorithm – used for the bandpass filtering – and its relationship to the analogue passive filter are presented. Exact algebraic expressions are derived relating centre frequency (f0), quality factor (Q), gain (G) and phase angle (f) to the IIR coefficients. A hardware interface to a compact disc player is described. It demonstrates the use of the DSP's SSI port for receiving and transmitting data, the implementation of a set

of parallel second-order IIR filters, and the design of a low-cost memory-port bootstrap EPROM/DSP56001 system. This system is all-digital – A/D and D/A converters are not needed.

**Order by: APR2/D**

## Fractional and Integer Arithmetic Using the DSP56000 Family of General-Purpose Digital Signal Processors

**Rev 1**

The on-chip multiplier of the DSP56000 Family of general-purpose Digital Signal Processors directly supports fractional data formats and indirectly supports integer formats, with hardware and software benefits. This note discusses the use of the processors to perform arithmetic operations on data represented as integers, fractions and combinations of the two (mixed numbers, real numbers, floating-point numbers).

**Order by: APR3/D**

## Implementation of Fast Fourier Transforms on Motorola's DSP56000/DSP56001 and DSP96002 Digital Signal Processors

**Rev 3**

Frequency domain applications (as opposed to time domain applications) are becoming increasingly important as inexpensive processors become more readily available. The Fourier transform can be used as a mathematical tool for fast-filtering applications provided that sufficiently powerful 'engines' are available to implement the real-time filtering operation. Motorola's DSP56000/1 and DSP96002 digital signal processors provide particularly effective solutions. This report examines the mathematical basis of the FFT and demonstrates how DSP56000/1 features such as hardware DO-loop capability can simplify practical implementation. It examines the effects of round-off errors and the significance of the IEEE Floating-Point Specification.

**Order by: APR4/D**

## Implementation of PID Controllers on the Motorola DSP56000/DSP56001

**Rev 1**

Demonstrates how the DSP56000/1 may be used to solve real-time digital control problems, concentrating on implementing some general control algorithms which include Proportional-Integral-Derivative (PID) controllers and notch filters. Points out the advantages of Digital Signal Processing over traditional analogue electronics in real-time applications.

**Order by: APR5/D**

## Convolutional Encoding and Viterbi Decoding Using the DSP56001 with a V.32 Modem Trellis Example

**Rev 1**

Coding techniques – such as Hamming, BCH and Reed-Solomon – have long been used to correct errors in data transmission systems by adding redundant data, and in some cases scrambling the original data. This paper considers the use of Convolutional Encoding, a good method for correcting burst errors occurring during data transmission. Viterbi decoding is a maximum-likelihood method which is fast enough to allow real-time decoding for short constraint length codes, when using high speed processors. The DSP56001 is particularly efficient here.

**Order by: APR6/D**

## Implementing IIR/FIR Filters with Motorola's DSP56000/DSP56001

**Rev 2**

Considers the design of frequency-selective filters – both Infinite Impulse Response (IIR) and Finite Impulse Response (FIR) – which modify the frequency content and phase of input signals according to some specification. Provides some intuitive insight into digital filters, especially the calculation of coefficients in the digital domain to obtain the desired frequency response, and shows how to implement both classes of filter (IIR and FIR) on the DSP56001.

**Order by: APR7/D**

## Principles of Sigma-Delta Modulation for Analog-to-Digital Converters

The performance of digital signal processing and communication systems is limited by the precision of the digital input signal at the analogue/digital interface. Sigma-Delta ($\Sigma$-$\Delta$) modulation-based A/D conversion is now a cost-effective technology for high resolution (>12 bits) converters integrated on DSP ICs, though the $\Sigma$-$\Delta$ modulator has only become important since developments in VLSI technology have allowed practical implementations. This note explains the $\Sigma$-$\Delta$ technology implemented in the DSP56ADC16, and shows the superior performance compared to conventional converters. Specifically, a third-order noise-shaping oversampling structure is discussed.

**Order by: APR8/D**

## Full-Duplex 32 kbit/s CCITT ADPCM Speech Coding on the Motorola DSP56001

Describes the implementation of an Adaptive Differential Pulse Code Modulation (ADPCM) speech coder on the DSP56001. The algorithm used has been standardized by the CCITT in Recommendation G.721[1] for digital speech coding in a telecommunications environment; the standard specifies the translation of $\mu$-law or A-law PCM encoded speech at 64K bit/s to ADPCM encoded speech at 32K bit/s to provide 2:1 compression with very little loss of quality. Two implementations are described; one adheres

completely with the CCITT Recommendation, the other implements the same algorithm in a more efficient manner. Both provide full duplex operation on a single DSP56001.

**Order by: APR9/D**

## DSP96002 Interface Techniques and Examples

Describes DSP96002 interfacing in four situations: Three high performance interconnection techniques for two or more DSP96002s; Connecting the DSP96002 as an Attached Processor on the IBM PC/AT™ bus (ISA bus) to provide an IEEE floating-point numeric accelerator; Interfacing the DSP96002 to the VMEbus by making an ADS96002 board a VMEbus slave; and Interfacing the DSP96002 to two DSP56ADC16 Sigma-Delta A/D converters. Timing diagrams and program listings are provided where necessary. A final section describes a non-intrusive hardware cycle counter for the DSP96002 Application Development System.

**Order by: APR10/D**

## DSP56001 Interface Techniques and Examples

**Rev 1**

The cost of using SRAM to create a large memory for a DSP system can be prohibitive. Pseudo Static RAM (PSRAM) – which combines a dynamic RAM array with a simple interface and on-chip refresh logic – provides a compromise between high density, low cost, high speed and interface simplicity. This note presents a simple implementation of a PSRAM interface to the DSP56001. It also describes an interface with standard dynamic RAM for systems needing large amounts of memory, such as audio special effects. The final section shows an interface with an ISA Bus host processor that uses only two additional parts.

**Order by: APR11/D**

## Twin CODEC Expansion Board for the DSP56000 Application Development System

This twin CODEC board is designed to simplify the development of telecom applications based on the DSP56000 Family processors. It uses the standard telecom sampling frequency of 8KHz, and the filter has a 300Hz to 3.4KHz band-pass characteristic. The board interfaces directly with the DSP using the SSI interface and is intended for any situation where a DSP module is required to link two analogue lines. Includes PCB artwork and DSP software listings for conversion between data formats.

**Order by: APR12/D**

## Conference Bridging in the Digital Telecomms Environment Using the Motorola DSP56000

Conference Bridging allows telephone calls of three or more subscribers to be set up, and provides arbitration so that conversation can take place in a controlled manner. Digital bridges usually use a ‘single speaker’ algorithm to preserve good signal to noise ratio, with the loudest speaking subscriber selected as the current speaker. This scheme is a software implementation written for the DSP56000/1.

**Order by: APR14/D**

## Implementation of Adaptive Controllers on the Motorola DSP56000/DSP56001

An adaptive control system measures a certain performance rating. Based on the difference between desired and measured performance, the adjustment system modifies the parameters of the adaptive controller and the control law in order to maintain performance close to the desired value. This note shows how the DSP56000/DSP56001 digital signal processors can be used to solve real-time digital control problems. After reviewing the basic theory of adaptive control, it describes a number of implementations using serial-parallel reference models.

**Order by: APR15/D**

## Calculating Timing Requirements of External SRAM for the 24-bit DSP56000 Family

When interfacing the DSP56000 family Digital Signal Processors to external SRAM, the behavior of the internal clock is affected by the external clock and by the configuration of the Phase Locked Loop. This behaviour in turn determines the speed requirement of the external SRAM. Timing parameters are also affected by the configuration of the signals that the DSP uses to access the SRAM. This note is a tutorial on calculating the timing requirements. The examples and discussion are based on the DSP56002, but may also be applied to other members of the DSP56000 family having an external bus.

**Order by: APR16/D**

## Application Optimization for the DSP56300/DSP56600 Digital Signal Processors

The DSP56300 and DSP56600 are high-performance 24-bit and 16-bit cores in Motorola’s family of digital signal processors. They are based on the same pipeline structure, which is capable of executing an instruction every clock cycle. At the same time the cores maintain a Harvard architecture and programming model, similar to the older 24-bit DSP56000 core. DSP56300/DSP56600 code may be based on earlier DSP56000 code, or may be written specifically for these processors. This document is a supplement to the detailed DSP56300 and DSP56600 Family Manuals, and describes the new features to enable software engineers to utilize resources fully and to develop optimized applications.

**Order by: APR20/D**

## Software UART on the DSP56L811 Using GPIO Port B

The UART port is a common interface the is used on a vast number of devices and products. However because the serial peripherals available on the DSP56L811 are synchronous devices they do not provide intrinsic UART capability. This application note describes a software module for the DSP56L811 which allows it to emulate a UART by using GPIO Port B in conjunction with timer interrupts. The interrupt-based design keeps core intervention and overhead to a minimum.

**Order by: APR21/D**

## Application Conversion from the DSP56100 Family to the DSP56300/600 Families

The Motorola DSP56100 family and the DSP56300/600 families are similar in many ways, but the parts are not fully compatible. Therefore, using a DSP56300/600 in place of a DSP56100 requires modification of both the hardware and the software. This document summarizes information needed by a user to estimate the effort required to convert an application and the details involved in translating the software. Discusses the differences in architecture (mainly the organization of data memory), examines unsupported DSP56100 instructions and provides a functionally equivalent DSP56300/600 solution.

**Order by: APR22/D**

---

## Interfacing Fast SRAM to Motorola's DSP56300 Family of Digital Signal Processors

This document is a supplement to the DSP56300 24-Bit Digital Signal Processor Family Manual, and to the user's manuals and technical data sheets. It describes how to interface Asynchronous Fast Static RAM to the Memory Expansion Port of devices in Motorola's DSP56300 family. Its aim is to assist the DSP hardware engineer to fully utilize the processor's resources and to generate an optimized memory system. The designs use a minimum of additional parts, and take advantage of the DSP's control lines to create a virtually glueless memory interface. They are based on the DSP56303.

**Order by: APR25/D**

---

## Interfacing Flash Memory with the Motorola DSP56300 Family of Digital Signal Processors

This document is a supplement to the DSP56300 24-bit Digital Signal Processor Family Manual, and describes how to interface FLASH memory to the DSP56300 family. It discusses various methods for interfacing different types of memory to the Memory Expansion Port and to create an optimized memory design using a minimum of additional parts. Asynchronous implementations of FLASH and Programmable Erasable ROM (PEROM) are described.

**Order by: APR26/D**

---

## Interfacing EPROM and EEPROM Memory with Motorola's DSP56300 Family of Digital Signal Processors

This document is a supplement to the DSP56300 24-Bit Digital Signal Processor Family Manual, and to the user's manuals and technical data sheets. It describes how to interface EEPROM and EPROM to the Memory Expansion Port of devices in Motorola's DSP56300 family. Its aim is to assist the DSP hardware engineer to fully utilize the processor's resources and to generate an optimized memory system. The designs use a minimum of additional parts, and take advantage of the DSP's control lines to create a virtually glueless memory interface. They are based on the DSP56303.

**Order by: APR27/D**

## DSP56300 Assembly Code Development Using the Motorola Toolsets

Provides integrated supplementary information for the Motorola assembly toolsets used for the DSP56300 family of Digital Signal Processors, beyond that in the user's manuals for the DSP56300 Assembler, Linker, Simulator and dubugger. A detailed example is provided for management of multifile assembly code projects in the UNIX environment. Includes an overview of the Simulator and the Application Development System (ADS), plus helpful tips – beneficial to users new to the toolset – for facilitating software development.

**Order by: APR30/D**

---

## Booting and Simple Usage of the DSP56004/007/009 SHI Port in SPI Mode

Discusses the procedure for booting the symphony series of digital signal processors through the Serial Host Interface (SHI) in Serial Peripheral Interface (SPI) mode. The target platform used here is the DSP56004/007/009 Evaluation Module, but the procedure is applicable to all Motorola DSPs with SHI ports. After booting, the simple DSP application runs as a slave to the MC68HC711E9 microcontroller on the Evaluation Module, responding to button selections on the MCU interface by toggling a GPIO pin.

**Order by: APR31/D**

---

## ROM Software Patching on the Motorola DSP56304

Although software is typically very robust before being introduced to ROM, modifications must often be incorporated later. On the DSP56304, incorporation is achieved by enabling the patch mechanism and using the cache to replace, skip or insert portions of code in ROM. This report discusses both simple and complex cases of patching ROM code. In addition, it presents methods that optimize the patching procedure to minimize the impact on cache performance.

**Order by: APR33/D**

---

## MC68328 Microprocessor Application: FLEX Alphanumeric Chip MC68175 Interface for One-Way Pager

The Motorola MC68328 (Dragonball) is a low-cost, low-power, highly integrated microprocessor designed for consumer portable devices such as PDAs, pagers and cellular phones. FLEX Protocol is Motorola's multispeed, high-performance paging protocol used by 70% of the world's paging service providers, which is becoming the de facto paging standard. This note describes the hardware and software interfaces between the MC68328 (Dragonball) MPU and the MC68175 FLEX Alphanumeric Chip signal decoder to create a powerful solution for personal portable communication devices.

**Order by: APR34/D**

## Designing Motorola DSP56xxx Software for Nonrealtime Tests File I/O Using SIM56xxx and ADS56xxx

Debugging real time digital signal processing systems is inherently challenging due to their complex nature and the high performance they demand. By executing portions of a DSP application in nonrealtime using the Motorola DSP Simulator or the Application Development System (ADS), many bugs can be eliminated before system integration and test. This report presents the methods for performing File I/O using the Simulator and ADS. Conditional assembly allows quick software reconfiguration for simulation, ADS or realtime execution, and is presented here as a solution to multi-execution environment needs.

**Order by: APR35/D**

---

## Interfacing the DSP560xx/DSP563xx Families to the Crystal CS4226 Multichannel Codec

Professional and consumer-level audio processing is a common application for Motorola's DSP560xx and DSP563xx digital signal processors, which are a popular choice in products ranging from recording studio effects processors to home surround sound decoders. This application note focuses mainly on the DSP56xxx as a surround sound decoder and on the accompanying hardware such an application requires. Includes interface considerations, communications protocol, and Crystal CS4226 interface examples.

**Order by: APR36/D**

---

## Implementing AC-link with ESAI

**Rev 1.1**

The Enhanced Serial Audio Interface (ESAI) of the DSP56300 Family provides full capabilities for interfacing with a general AC'97 CODEC through an AC'97 Digital Serial Interface (the AC-link). This report describes how to implement an AC-link using the ESAI. It includes details of the physical connection, the Data-Flow model, system concept, DSP56300 Family resources used, and the assembly code and equates of the application.

**Order by: APR37/D**

---

## Interfacing Serial EEPROM to DSP563xx

Describes how to interface DSP56300 Family devices to industry standard, SPI-compatible, Serial Electrically Erasable Programmable Memory (SEEPROM) such as SGS-Thomson's ST95010/020/040 or National's NM25C020. The interface is based on either the Enhanced Synchronous Serial Interface (ESSI) or the Synchronous Communication Interface (SCI), which are available in several derivatives of the DSP5600 Family.

**Order by: APR38/D**

## Programming the DSP56307 Enhanced Filter Coprocessor (EFCOP)

The Enhanced Filter Coprocessor (EFCOP) is a general purpose peripheral module of the DSP56307. It is a fully programmable complex filter whose operations are completed concurrently with the DSP56300 core operations with minimal CPU intervention. The EFCOP has dedicated modes of operation which are optimized for cellular basestation applications. This document describes its programming model, and presents two application examples: a complete IIR filter, and an LMS echo canceller.

**Order by: APR39/D**

---

## Implementing Viterbi Decoder Using the VSL Instruction on DSP Families DSP56300 and DSP56600

Today's communication systems typically make considerable use of signal processing to improve performance. Two common functions are channel equalization and error correction; for equalization, maximum likelihood sequence estimation is among the most popular schemes, while for error correction, convolutional coding with Viterbi decoding is a method of choice. This note describes how to generate the assembly code needed for implementation of a Viterbi decoder on the DSP56300 and DSP56600, from a set of convolutional code polynomials.

**Order by: APR40/D**

---

## 15 x 15mm PBGA Daisy-Chain Application Report

Glob-top Plastic Ball Grid Array (PBGA) mechanical sample packages are available from Motorola for use in surface mount assembly process development. They are physically similar to the active devices, being manufactured to the same general material and physical specifications. This report describes the 15 x 15mm PBGA daisy-chain package and its use in developing PGBA surface mount techniques.

**Order by: APR42/D**

---

## G.722 Audio Processing on the DSP56100 Microprocessor Family

The CCITT standardized G.722 specification details the characteristics of a system for 7kHz audio-coding within 64 kbits/s that may be used for a variety of higher quality speech applications. This note describes the software implementation of a speech codec conforming to G.722 that uses Motorola's DSP56156. The coding system uses Sub-Band Adaptive Differential Pulse Code Modulation (SB-ADPCM) to decimate a signal sampled at 224 kbit/s to digital data for transmission at 64 kbit/s. The note also gives a brief overview of the latest speech coding techniques and the relative position of the G.722 algorithm.

**Order by: APR404/D**

## Minimal Logic DRAM Interface for the DSP56156

Many DSP applications require large amounts of memory. Significant reductions in cost can be achieved by using Dynamic RAM in place of fast Static RAM, though this will always be at the expense of memory access speed. This note describes a minimum glue-logic DRAM interface – designed for maximum performance – for the DSP56156. The scheme differs from conventional DRAM designs in that no latches are used to hold the row and column addresses during DRAM accesses – the DSP address and data lines are connected directly to the DRAM, and a single PAL16R6 is used for memory decoding, read, write and refresh control.

**Order by: APR405/D**

---

## Versatile Test Fixture Verifies the Switching Capability of UFR Rectifiers Under High di/dt Conditions

With the increasing requirement for more efficient power conversion systems through higher switching frequencies, the emphasis falls on using faster semiconductors for both the transistor switches and the associated freewheeling diodes. Power MOSFETs and even some bipolars serve the switching functions very well, which sometimes leaves selection of the type of rectifier for the application up in the air. Intuitively the choice seems easy, but how fast do they need to be and at what cost/performance trade-off? (*Powerconversion & Intelligent Motion*, October 1986)

**Order by: AR178/D**

---

## RF Power Transistors Catapult into High-Power Systems

Back in 1962, RF power transistors were nonexistent, or at best in their infancy. Today the RF power transistor is a mature product that provides solid state sources in kilowatt FM broadcast transmitters, TV transmitters, 120W two-way mobile radios, cellular telephones, aircraft communications, radar and a variety of military applications. RF power has reached today's technology primarily because of advancements in two key areas – die design and packaging. (*Microwaves & RF Magazine*, March 1987)

**Order by: AR179/D**

---

## Building Fast SRAMs with no Process 'Tricks'

Motorola has devised a series of circuit innovations that create true asynchronous Static RAMs capable of 25ns access times, without relying on process improvements like scaled geometries, three levels of interconnection, resistance reduction or bipolar/CMOS combinations. The result is remarkably low power consumption, immunity from skews, small die area and improved reliability. This article has the details. (*Electronics*, 7 August 1986)

**Order by: AR241/D**

## Motorola's Radical SRAM Design Speeds Systems 40%

Engineers designing Motorola's new Static RAMs develop a synchronous architecture that improves system throughput 40% while reducing component count 50%. With the addition of critical I/O latches on-chip, 8 to 10ns of interconnection delay is eliminated, together with external circuits often required to make asynchronous devices appear synchronous in high-speed cache memory systems. (*Electronics*, 23 July 1987)

**Order by: AR256/D**

---

## High Frequency System Operation Using Synchronous SRAMs

Designers demand faster and faster Static RAMs to support shorter processor cycle times. Fast SRAMs need precise control to achieve their full performance, creating the need for additional logic. The Synchronous Static RAM has on-chip latches for all inputs and outputs, added drive capability and a self-timed write capability. Most external logic is eliminated and the memory runs at higher speeds than standard SRAMs with comparable access times. The paper describes the new architecture and an application example of an MC68030 cache subsystem. (MIDCON 1987)

**Order by: AR258/D**

---

## Enhancing System Performance Using Synchronous SRAMs

The speed of high-performance systems is frequently limited by the performance of available SRAMs. The demand for sub-25ns devices is growing. Fast Static RAMs are already the driving force behind semiconductor technology, but with the smallest circuit features and the need for special processes they are often expensive to produce. Using conventional technology Motorola has developed an alternative 15ns device aimed specifically at meeting the real purpose of Fast SRAMs. (*ECN*, October 1987)

**Order by: AR260/D**

---

## Electronics: a Driving Force in Automotive Technology

While electrical power is not jeopardizing the internal combustion engine in passenger vehicles, the power of electronics is continually being exercised in new vehicle systems. The first generation of electrical/electronic systems such as anti-lock brakes and multiplex wiring are installed in limited production luxury vehicles. In order for these and future systems to be implemented in standard production vehicles, significant cost reductions will be required. New power electronics technology, including 'smartpower', promises to be the key to making the transition from concept to volume production. (*Automotive Engineering*, August 1988)

**Order by: AR320/D**

## RF Modems Simplified

A few years ago, if a system required an RF modem the engineer had to design it from scratch. Today, if the system needs can be met using FSK modulation, RF modems are available. An example is the MHW10000 family from Motorola, described here. These RF modules provide all the necessary transmitter and receiver functions for full duplex VHF modems capable of interfacing to a single T/R port. (*RF Design*, January 1990)

**Order by: AR333/D**

## RF Power FETs: Their Characteristics and Applications

The first article in this two-article reprint places the various FET manufacturing technologies in historical perspective, as a background to a detailed discussion of power FET characteristics and a comparison between FETs and bipolars used in RF designs. The second article considers in detail the effects of FET characteristics on device performance in various typical applications, including noise performance and linearity. (*QEX*, January 1989)

**Order by: AR346/D**

## A Compact 1kW 2-50MHz Solid-State Linear Amplifier

Solid-state high-power linear amplifiers are becoming more and more popular in the field of ham radio as the prices of HF power transistors continue to fall; 250W devices are now available for almost half the price of a few years ago. RF power FETs are still more expensive, but eventually their prices will also fall. This 1kW amplifier is a compact state-of-the-art design using two 600W FETs. It would be difficult, if not impossible, to design an amplifier offering similar performance over four and a half octaves using inexpensive bipolar transistors. Includes PC artwork. (*QEX*, July 1990)

**Order by: AR347/D**

## Adapt Non-ISDN Terminals to ISDN Data Rates

The emergence of the Integrated Services Digital Network (ISDN) raises the question of how to use older, non-ISDN equipment with the new system. The CCITT has proposed the V.110 and V.120 interface standards to solve this problem in two different ways. The standards are known as the Rate Adaption methods because they are concerned mainly with adapting the data rates of terminal equipment to the 64 kbit/s basic rate of ISDN. This article shows how the Motorola MC68302 Integrated Multiprotocol Processor can be used to implement either of the standards. (*Electronic Design*, 25 April 1991)

**Order by: AR350/D**

## The Making of the PowerPC

A primary design goal of the PowerPC 603 microprocessor was to provide sophisticated power management without compromising performance. The system designer can control energy consumption through both hardware and software, and the 603 also includes automatic internal power management. This article reviews the internal Dynamic Power Management, the four power states and the transitions between them. (*Communications of the ACM*, June 1994)

**Order by: AR359/D**

## PowerPC 620 Soars

In October 1994, IBM and Motorola jointly announced first silicon on the PowerPC 620, the first 64-bit implementation of the PowerPC architecture in a processor. Its faster logic, shorter pipelines and high-speed interface give it processing power that raises it to workstation and server status. This article provides a technical overview of the 620, comparing it in particular to the 604. (*BYTE Magazine*, November 1994)

**Order by: AR360/D**

## Whipping Up Real-Time Designs – Programming Motorola's TPU

Motorola's Time Processor Unit can offer tremendous flexibility for designers of embedded systems. This overview of the steps to mastering the timing coprocessor includes a description of the TPU and several examples of custom programming. (*Embedded Systems Programming*, March 1995)

**Order by: AR362/D**

## Programming PowerPC Embedded Applications

An Application Binary Interface (ABI) is a set of software conventions designed to allow interoperability between different vendors's development tools. The PowerPC Embedded ABI was developed by industry members, and represents an optimization for embedded applications of the SVR4 desktop ABI. This article examines the basic principles. (*Embedded Systems Programming*, December 1995)

**Order by: AR363/D**

## Motorola Cellular DSP Does It All

DSP56690 integrates M•Core MPU, supports multiple wireless standards. Motorola's new DSP56690 is a highly integrated embedded processor that supports all of the most common wireless standards worldwide. For the first time, a single chip handles analog cellular as well as a plethora of digital standards. However despite initial excitement over the prospect of a universal wireless phone, Motorola sees the DSP56690 as a malleable platform for a multitude of future cell phones. (*Microprocessor Report*, 6 December 1999)

**Order by: AR366/D**

## VSWR Protection of Solid State RF Power Transistors

Most transistor failures in solid state amplifiers occur at load mismatch phase angles that present a high current mode of operation to the transistor, resulting in an increase in the power dissipated. Since

the temperature time constant of a typical RF power transistor die is 0.5 to 1ms, any protection system must react faster than this. (*RF Design*, February 1991)

**Order by: AR510/D**

## Biasing Solid State Amplifiers to Linear Operation

Solid state devices intended for linear operation need a certain amount of “forward bias” to place their operating points in the linear region of the transfer curve. Bipolar devices require a constant voltage source, whereas MOSFETs can be biased with a simple resistor divider network – both are more complex if temperature stability is required. Applications requiring amplifier linearity include amplitude modulated systems for communications and broadcasting, nuclear magnetic resonance, digital cellular telephones, and signal sources for instrumentation.

**Order by: AR511/D**

## Simple Pressure Switches Comprise Transducers, Comparators and Op Amps

With the addition of a few components, conditioning circuits for pressure transducer signals can provide a logic-level output that changes state when a pressure crosses a threshold. This article describes such a switch based on the MPX2100DP. It provides background calculations, circuit details and performance comparisons when using different comparator circuits. (*EDN*, 14 April 1994)

**Order by: AR560/D**

## Next Generation Packaging Solutions

Even as components and surface-mount packages become more complex due to higher functionality and pin counts, there is still a need for basic, low-tech devices (relative to high-end microprocessors). And so the demand for discrete surface-mount devices (SMDs) continues to grow, a growth made possible by new silicon packaging efforts. (*Circuits Assembly*, May 1994)

**Order by: AR561/D**

## GaAs RF ICs Target 2.4GHz Frequency Band

Motorola has introduced its first GaAs RF integrated circuits. Designed for applications in the 2.40GHz to 2.48GHz industrial/scientific/ medical band, they support a wide range of modulation formats and include a low noise amplifier and downconverter mixer, a buffer/ exciter amplifier, and a power amplifier. All three devices feature low current operation at low voltages and are housed in plastic SOIC packages. (*Microwaves & RF*, July 1994)

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: AR597/D**

## Using a Pulse Width Modulated Output with Semiconductor Pressure Sensors

For remote sensing and noisy environment applications, a frequency modulated (FM) or pulse width modulated (PWM) output is better than an analog voltage output. FM outputs are more widely accepted, but it is difficult to obtain a stable FM signal without complex and expensive circuitry. If a programmable-frequency, stable PWM output could be obtained with simple, inexpensive circuitry, a PWM output would be a cost-effective solution for noisy and remote sensing applications, while incorporating the advantages of frequency outputs. This article describes such a design. (*Sensors*, November 1994)

**Order by: AR601/D**

## Home-Brewed Circuits Tailor Sensor Outputs to Specialized Needs

Although fully signal-conditioned, calibrated and temperature compensated monolithic sensor ICs are readily available, a custom signal conditioning circuit is often a better solution. Typical situations are the need to rapidly modify the transfer function of a prototype subsystem, or the available conditioned device is trimmed over a different pressure range. In such cases, an untrimmed sensor customized with a signal conditioner based on two or three op amps can be a simple and cost effective solution. This article presents a basic two op amp circuit that exhibits instrumentation amplifier characteristics. (*EDN*, 5 January 1995)

**Order by: AR602/D**

## PCS and RF Components

Suddenly, almost everyone in the electronics business is making products for “wireless” applications. But today’s “wireless” is different to the decades of radio communications between fixed points – the market is now defined by two important words: “portable” and “consumer”. This article looks at Personal Communications Services (PCS), and its effect on the development of semiconductors and semiconductor technologies. (*Applied Microwave and Wireless*, Spring 1995)

**Order by: AR606/D**

## Plastic Packages Hold Power RF MOSFETs

In the past, packages for RF power applications have generally been fabricated using ceramic materials; typically beryllium oxide (BeO). These packages have performed well for power bipolar and vertical FET devices, but at considerable cost compared to the plastic housings used elsewhere. With the increasing use of GaAs and LDMOS devices, as well as the need for high-performance, low-cost products for high-volume commercial applications, Motorola has developed a family of plastic packages that can accommodate a wide range of power devices from predrivers to output stages.

**Order by: AR612/D**

## Advantages of LDMOS in High Power Linear Amplification

Discusses the advantages of LDMOS over Bipolar technology for linear power amplification, especially for high dynamic amplification or high power amplification in the 1.0GHz range. It presents results from a study on recently introduced Motorola LDMOS transistors compared to the same generation of bipolar transistors, both operating at 960MHz in the GSM band. The study focussed on gain, linearity and intermodulation. (*Microwave Engineering*, April 1996)

**Order by: AR614/D**

---

## Aluminum-Based Metallization Enhances Device Reliability

Wireless communications infrastructure equipment vendors demand RF power transistors with high performance – gain, efficiency, linearity – to enable the development of superior power amplifier designs. They must be designed to meet RF performance goals without compromising device reliability. As a result, a design-for-reliability methodology must be adopted early in the transistor design cycle. A modern aluminum alloy top-metal system developed for third and fourth-generation 2GHz laterally diffused metal oxide semiconductor (LDMOS) transistors from Motorola delivers unprecedented levels of electromigration performance – translating into MTTFs well in excess of 100 years at 200˚C. (*Microwaves & RF*, October 1998)

**Order by: AR624/D**

---

## SiGe HBTs Drive biCMOS Deeper into RF's Realm

The demand for faster and faster transistors is forcing RF IC suppliers toward silicon-germanium (SiGe) Heterojunction Bipolar Transistors (HBTs). They are taking advantage of the advances over the last ten years of silicon implanted with germanium, and narrowing the operating frequency gap between silicon and GaAs transistors. (*Analog Applications*, 28 June 1999)

**Order by: AR625/D**

---

## SiGe Gets Real

Long heralded as the next great process for RF ICs, innovative SiGe parts are now available. By combining silicon with germanium, the process in theory offers IC designers the best of both worlds – they get the well-established attributes and process flow of silicon, coupled with a performance edge provided by the added germanium. But SiGe will not displace GaAs, which still offers the performance edge in targeted applications, and has a manufacturing track record. (*EDN*, 24 June 1999)

**Order by: AR626/D**

---

## Motorola Pushes for the Handset Market

Motorola is a device company that says it is going to surprise the world in GaAs over the coming years. This interview is with Dr. Mark Wilson, Device Engineering Manager at Motorola's Compound Semiconductor-1 (CS-1) fab in Arizona, USA. The company has expanded CS-1 into one of the world's largest GaAs manufacturing sites, and has experienced such rapidly increasing demands for its RF products that it is expanding capacity once again. (*III-Vs Review*, Vol. 12 No. 4 1999)

**Order by: AR627/D**

---

## Impedance Measurements for High Power RF Transistors Using the TRL Method

Impedance measurements on high power RF transistors are always a problem because of the very low values and lead widths, and because the devices may be in a push-pull configuration. Calibration kits to measure wide or balanced microstrip lines are difficult to build and provide poor accuracy. This article describes a way to accurately characterize high power RF transistors even with wide leads and in a single-ended or push-pull configuration. Different calculation options are presented. (*Microwave Journal*, October 1999)

**Order by: AR628/D**

---

## Digital Predistortion Techniques for RF Power Amplifiers with CDMA Applications

Power amplifiers used in the next-generation wireless communication systems based on spread-spectrum techniques must exhibit exceptional linearity. This must be achieved without sacrificing efficiency to any great extent. Digital predistortion techniques have been shown to improve the linearity of power amplifiers for narrowband applications. In this article, digital predistortion is applied to both GaAs and laterally diffused metal oxide semiconductor (LDMOS) amplifiers with narrowband and wideband input signals. The efficacy of the digital predistortion under various input signal peak-to-average power ratios is also considered. (*Microwave Journal*, October 1999)

**Order by: AR629/D**

---

## An 8-bit EPROM Interface for an MC68EC040/ MC68360 System

The MC68360 Quad Integrated Communication Controller (QUICC) has an operating mode where the internal CPU32+ core may be disabled to allow an external processor to use its peripherals, and its MC68040 Companion Mode enables a glueless interface to an external M68040 family MPU. This document outlines a method of booting an MC68EC040/MC68360 combination from a single 8-bit EPROM. Familiarity with both devices is assumed.

**Order by: DC414/D**

---

## Interface for MC68000 to DSP56001 Host Port

Shows how to interface an MC68000 (up to 16.67MHz) to one or two DSP56001s (20MHz or greater) through their HOST PORT interface. The interface provides read and write handshaking between the devices, reset capability for either DSP56001, 7-level interrupt encoding, and interrupt acknowledge handshaking for both DSP56001s.

**Order by: DCE406/D**

## Low-Cost VHF Amplifier Has Broadband Performance

This bulletin presents two VHF amplifier designs intended for FM or CW service in the 136-174 MHz band. Both feature the Motorola MRF260 and MRF262 plastic encased VHF transistors which are rated at 5.0 W and 15 W power output respectively. The devices are packaged in a standard T0-220 silicone epoxy case with the emitter wired to the metal tab and centre lead of the device. This common emitter configuration results in good RF performance, improved thermal conductivity, and ease of mounting in an RF amplifier by connecting the transistor mounting flange to RF and DC ground.

**Order by: EB90/D**

---

## 60 Watt VHF Amplifier Uses Splitting/Combining Techniques

Proven combining techniques can be used to obtain higher output power and added reliability at VHF. Simple matching networks and power transistors with moderate gain can produce performance comparable to that of a single-stage amplifier with a larger, more expensive device. Though not the ultimate answer, the splitter/ combiner method has distinct advantages over designs that force transistors into a parallel configuration. This 60 W amplifier operates from 150 to 175 MHz and features two low-cost MRF264 transistors. The design uses a modified Wilkinson combiner technique to produce 60W output with a drive level of 15W.

**Order by: EB93/D**

---

## Mounting Considerations for Motorola RF Power Modules

The packaging used for Motorola RF Power Modules consists of a copper flange on which the ceramic substrates are soldered, and a non-conductive cover which is either a snap-on design or attached by epoxy. The substrates are either 96% Alumina, 95.5% Alumina, or 99% Beryllium Oxide, and are attached to the copper flange using lead-tin or indium based soft solders. This bulletin discusses the mechanical factors that should be considered when mounting these modules in equipment.

**Order by: EB107/D**

---

## Low Cost UHF Device Gives Broadband Performance at 3.0 Watts Output

The package is the major cost in low to medium power RF transistors. Motorola introduced the common emitter TO-39 some years ago to limit cost increases. Good design and construction techniques can extend its use to broadband UHF amplifiers, like this broadband application of the low-cost MRF630, a transistor capable of 3W output power with 10dB gain at 512MHz. Emphasis is placed on mounting techniques.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: EB109/D**

## Running the MC88110 in Lockstep

Systems that use two MC88110s running in lockstep must take precautions to guarantee predictable behavior. First, all inputs must meet the setup and hold times so that they are recognised on the same clock. Second, the system must ensure that the initial decoding of invalid instructions does not adversely affect the timing of the processors. This note addresses the second problem.

**Order by: EB163/D**

---

## Interrupt Latency in the MC88110

This bulletin addresses interrupt latency in the MC88110. It provides a brief description of how interrupts are handled and includes examples of short and long interrupt latency cases. It is intended for hardware system designers who are familiar with the MC88110 Second Generation RISC Microprocessor User's Manual.

**Order by: EB164/D**

---

## Hardware Implications of xmem as a st followed by a ld

The MC88110 supports an Exchange Memory (xmem) instruction that is a combination of a Load and Store instruction. The xmem instruction is normally a read access followed by a write access (as implemented originally in the MC88100). However, the xmem instruction can also function as a write access followed by a read access if the xmem bit is set in the data MMU/cache control register (DCTL); the write-before-read option can improve system performance. This note looks at the hardware implications.

**Order by: EB165/D**

---

## System Design Considerations: Converting from the MC68HC805B6 to the MC68HC705B16 Microcontroller

Production of the MC68HC805B6 has been discontinued as a result of Motorola's corporate decision to eliminate the use of CFCs in manufacturing processes (CFCs were used in the processing of the MC68HC805B6 MCU). The recommended replacement is the MC68HC705B16. Whilst this is pin compatible with the earlier MCU and supports all its resources, it is not a drop-in replacement. This bulletin presents a detailed comparison of the two devices, with emphasis on the differences.

**Order by: EB166/D**

---

## Differences between the MC68HC705B16 and the MC68HC705B16N

The MC68HC705B16N is a new version of the MC68HC705B16. Both devices are covered in revision 4 of the MC68HC05B6 Family Technical Data Manual, but this bulletin summarises the differences, which affect the bootloader, reset pulse width, the reset twice requirement of the MC68HC705B16, the supply current in Stop mode, and the shrink level.

**Order by: EB180/D**

## Frequently Asked Questions and Answers: M68HC05 Family MCAN Module

Provides straightforward answers to 14 frequently asked questions about the M68HC05 Family MCAN module, with illustrations and assembler listing where appropriate. Questions include “How does the Ping-Pong principle of the two receive buffers work?” and “When and how should the single-line mode be used?”

**Order by: EB181/D**

## How the ROMON Bit Behaves on the E Series HC11 MCUs

The ROMON bit is responsible for mapping the on-board ROM or EPROM into the standard 64K addressing range. This bulletin discusses the functionality and differences in the ROMON bit in the MC68HC711E9 and MC68HC711E20 microcontrollers.

**Order by: EB182/D**

## Erasing and Programming the FLASH EEPROM on the MC68HC912B32

**Rev 1.0**

The 32 Kbytes of embedded FLASH EEPROM in the MC68HC912B32 serves as electrically programmable and erasable, non-volatile ROM emulation memory. It allows the storage of program code which must be executed frequently, must execute at high speed, or which might need to be upgraded in the field. This document outlines basic routines which can be used to program the FLASH EEPROM through the background debug mode interface (BDM), using a Motorola serial debug interface (SDIL) and the SDBUG12 (version 2.15) software from P & E Microcomputer Systems, Inc.

**Order by: EB183/D**

## Enabling the Security Feature on the MC68HC711E9 Devices with PCbug11 on the MC68HC711E9PGMR

This note provides a step-by-step procedure describing how to enable the security feature on MC68HC711E9 devices using the M68HC711E9PGMR Programming Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB184/D**

## Simplify MC68HC711E9 EPROM Programming with PCbug11 and the M68HC711EPGMR Board

This note provides a step-by-step procedure describing how to program MC68HC711E9 devices using the M68HC711EPGMR Evaluation Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB185/D**

## Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVB

This note provides a step-by-step procedure describing how to program MC68HC711E9 devices using the M68HC11EVB Evaluation Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB187/D**

## Enabling the Security Feature on M68HC811E2 Devices with PCbug11 on the M68HC711E9PGMR

This note provides a step-by-step procedure describing how to enable the security feature on MC68HC811E2 devices using the M68HC711E9PGMR Programming Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB188/D**

## Programming MC68HC811E2 Devices with PCbug11 and the M68HC711E9PGMR

This note provides a step-by-step procedure describing how to program MC68HC811E2 devices using the M68HC711E9PGMR Programming Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB189/D**

## Programming EPROM and EEPROM on the M68HC11EVM

The M68HC11EVM is one of the most versatile development tools for the MC68HC11 A and E series microcontrollers. The bulletin applies to revision G boards using the EVMbug monitor version 3.0, and explains how to use the M68HC11EVM to program EPROM and EEPROM of the supported devices. It also has details of how to upgrade an earlier board to revision 3.0 status.

**Order by: EB191/D**

## A Quick Tutorial for MC68HC11 K, KA, KW, P and PH Series Microcontrollers

New Pulse Width Modulation (PWM) timer channels are available on certain microcontrollers in the M68HC11 family. Each of the affected MCUs has four 8-bit PWM timer channels; each pair of channels may use one of two different frequency references derived from the E clock. To produce PWM waveforms with longer periods, pairs of channels may be concatenated to form single 16-bit channels. This bulletin provides descriptions and programming examples for both 8-bit and 16-bit PWM generation.

**Order by: EB192/D**

## Replacing 68HC11A Series MCUs with 68HC11E Series MCUs

**Rev 1.0**

The 68HC11E Family was designed to provide more ROM and RAM for 68HC11A users. Both families have the same pinouts in the 52-pin PLCC and 64-pin QPF packages, but the 68HC11E MCUs have several additional enhancements that require special design consideration. This bulletin describes the differences and addresses the extent of code analysis required for various MC68HC11A Family applications.

**Order by: EB193/D**

---

## How to Configure the Reset Pin on the MC68HC11

The ideal way to control the Reset pin on the MC68HC11 is with a low voltage inhibit circuit. However many designers would like to use an RC circuit, and wonder why the MC68HC11 Reference Manual specifically forbids this. This bulletin explains why a capacitor should never be connected to the Reset pin and describes the correct Reset configuration.

**Order by: EB195/D**

---

## Using Pseudo-Interrupt Vectors on the M68HC11EVBU

The User's Manual for the MC68HC11EVBU Universal Evaluation Board contains a printing error in respect of the pseudo-interrupt vectors to be used when running the BUFFALO monitor. This bulletin corrects the error, and clarifies the need for the vectors when running the monitor.

**Order by: EB197/D**

---

## Turn Off Your E Clock to Reduce Noise Emission on the MC68HC11

Since the introduction of the MC68HC11A8 in 1984, many new functions and peripherals have been developed, adding functionality and variety to this versatile family of microcontrollers. However many of the products that now contain M68HC11 Family MCUs are also noise sensitive applications such as digital pagers and cellular phones. This application note describes a feature on the MC68HC11 which will reduce radiated noise, as well as reducing power consumption, by turning off the E Clock.

**Order by: EB198/D**

---

## Mounting Method for RF Power Leadless Surface Mount Transistors

The use of leadless surface mount RF devices in wireless systems has challenged previously well-accepted assembly techniques and thermal management methods. In mounting these devices, key attention must be given to their Coefficient of Thermal Expansion, terminals, heat sink and the board, as well as to the solder hierarchy within the system. This note addresses issues and solutions for mounting RF power transistors in the range 1-20 watts power dissipation; the 8W MRF1507 is used as an illustration.

*NOTE: This document references an older device that is not recommended for new designs.*

**Order by: EB209/D**

---

## Thermal Management and Solder Mounting Method for the MRF286, 60 Watt Power Device in a CuW (Copper Tungsten) Base Package

The critical aspects of any mounting method for power devices are how to transfer heat away from the semiconductor, and whether the mechanical construction is such that the package will not come apart or cause a performance failure during normal operation in expected environments. This note presents a process for mounting the MRF286 60W power device, together with the methodology used in the development of the procedure. It is also applicable to other packages with the same type of construction and materials for devices ranging from 20W to 60W dissipated power.

**Order by: EB211/D**

---

## How to Calculate Instruction Times on the MC68HC16

Chapter 8 of the CPU16 Reference Manual (CPU16RM/AD) explains how to calculate instruction timing. This engineering bulletin is a supplement which summarizes the most important points in the chapter and explains some of the examples in more detail.

**Order by: EB251/D**

---

## MOVB, MOVW, PSHM, and PULM Syntax Differences on MC68HC16 Assemblers

The architecture of the M68HC16 Family of microcontrollers contains several new instructions compared to the M68HC11 Family, which add more powerful addressing modes to speed execution of certain repetitive operations. Some assemblers differ in the syntax of the instructions, and this bulletin discusses the differences in the MASM16 and IASM16 assemblers. MASM is shipped with the M68HC16Z1EVB, while IASM is shipped with the M68ICD16 dubugger package and all MEVB16 boards.

**Order by: EB252/D**

---

## How to Use the Lookup and Interpolate Instruction on the CPU32

The Table Lookup and Interpolate instruction approximates a number that lies between two consecutive entries in a lookup table, as a function of one variable or of several variables. This bulletin explains the use of the instruction when only one variable is involved.

**Order by: EB253/D**

## Setting the Programming Voltage of Modular Microcontrollers with FLASH EEPROM

The MC68F333 has two FLASH electrically erasable programmable ROM (EEPROM) modules, which can be configured in various way. This bulletin provides information concerning the program/erase voltage on the VFPE pins.

**Order by: EB254/D**

## Use of the Lock Bit on Modular Microcontrollers with FLASH EEPROM

Explains the function of the LOCK bit in the FEEMCR register of modular microcontrollers with FLASH EEPROM, and the operation of the shadow registers.

**Order by: EB256/D**

## Detecting Loss of Clock on Modular Microcontrollers

If the voltage controlled oscillator (VCO) of a modular microcontroller is used to generate the system clock frequency, the clock logic can detect a clock failure. This bulletin explains the possible built-in actions if this happens. Loss of clock cannot be detected when an external oscillator is used.

**Order by: EB257/D**

## Sources of Reset on Modular Microcontrollers

Motorola modular microcontrollers have several sources of reset, which are listed and explained here. Synchronous resets are asserted at the end of the current bus cycle, while asynchronous reset signals cause an immediate reset of the system. The reset status register records which type of reset last occurred.

**Order by: EB258/D**

## Why MC68300 and MC68HC16 MCUs May Halt after Release of Reset

A problem can occur with MC68300 and MC68HC16 microcontrollers showing specific symptoms including: the Reset line periodically asserts due to the watchdog timer timing out; all bus activity stops after eight or four pulses on the CSBOOT line; HALT, AS and DS are not asserted; and the FREEZE pin is asserted with a logic 1. This bulletin explains the problem and its solution.

**Order by: EB259/D**

## Why an MC68300 or MC68HC16 Microcontroller May Fail to Release Reset

Under certain conditions a device from either the MC68300 or MC68HC16 families may fail to come out of Reset. This bulletin explains the possible causes and the steps that should be taken to avoid such problems.

**Order by: EB260/D**

## Autovector Generation Using Chip Select Logic on MC68300 and MC68HC16 Devices

In MC68300 and MC68HC16 devices, an AVEC signal can be used to respond to an interrupt acknowledge cycle for one of the external interrupts (IRQ[7:1]), and automatically to choose a particular interrupt vector. IRQ[7:1] use vector numbers 11-17 on the CPU16 and vector numbers 25-31 on the CPU32. It is possible to respond to the interrupt acknowledge cycle with a DSACK signal which requires an external device to supply its own interrupt vector. This bulletin explains how to use the internal chip select logic to generate the AVEC signal.

**Order by: EB261/D**

## DSACK Generation on the System Integration and Single-Chip Integration

The System Integration Module (SIM) and Single-Chip Integration Module (SCIM) provide programmable chip-select logic to help the designer interface a modular microcontroller to external peripherals. A frequently asked question is: What happens if a chip-select is programmed as either discrete output or alternate function in the chip-select pin assignment register, but the chip-select base address and option registers are also programmed? This bulletin provides the answer.

**Order by: EB262/D**

## How to Program Chip Selects on Modular Microcontrollers with a System Integration Module or a Single-Chip Integration Module

Modular MCUs with a System Integration Module (SIM) have 12 chip-select lines, and those with a Single-Chip Integration Module (SCIM) have nine chip-selects. Both types are programmed in the same way, and in addition to enabling a peripheral or memory chip they can be programmed to generate data transfer and size acknowledge (DSACK) signals. Two problems sometimes encountered are that the chip select generates the wrong number of wait states before the DSACK, or that the wrong chip-select is enabled. There are specific reasons for these problems, which are addressed in this bulletin.

**Order by: EB263/D**

## Disabling All Interrupts on Power-On for MC683xx and MC68HC16 Devices Using the Single-Chip Integration Module or System Integration Module

From a power-up condition in either the MC683xx or the MC68HC16 MCU families, program runaway is possible if the IRQ7 line is driven to an active low condition. This bulletin explains how to recover from this situation by properly configuring the Single-Chip Integration Module (SCIM) or System Integration Module (SIM).

**Order by: EB264/D**

## Generating Edge-Sensitive Interrupts on the MC683xx and MC68HC16 Families of MCUs

One way to generate edge-sensitive interrupts is to use an input to a peripheral module such as the General Purpose Timer (GPT) or the Time Processor Unit (TPU). This bulletin presents alternative techniques.

**Order by: EB265/D**

---

## Unexplained Three-Stating of the Address Bus on M68300 and M68HC16 Devices

Some users of devices in the M68300 and M68HC16 Families have observed that the devices sometimes appear to halt operation and enter a three-stated condition. This appears to occur either randomly or in a certain block of code, but not at a specific instruction in that code. This engineering bulletin explains the likely cause of the problem.

**Order by: EB266/D**

---

## The Double Bus Fault Monitor

A double bus fault is an exception upon an exception. In some older user's manuals the double bus fault monitor is not referred to because it was previously called the Halt Monitor. This bulletin provides some basic information about the monitor.

**Order by: EB267/D**

---

## Starting and Stopping the Time Processor Clock Using the Background Debug Mode

Developers of Time Processor Unit (TPU) microcode may want to halt microcode execution when debugging. One way to do this is to enter the Background Debug Mode (BDM), a special operating mode in which normal instruction execution is suspended. This bulletin details the procedure.

**Order by: EB268/D**

---

## Using the SCI on Modular MCUs: An Example

The Serial Communication Interface (SCI) is part of the Queued Serial Module (QSM) and the Multi-Channel Communication Interface (MCCI) of modular microcontrollers, and is used to communicate with external devices and other MCUs via an asynchronous serial bus. This example program was assembled with the assembler available from P & E Microsystems, and can be used for simple debugging purposes.

**Order by: EB269/D**

---

## Problems with the PPWA Function on Revision P MC68332 Devices

The Period/Pulse Width Accumulation (PPWA) function on the Time Processor Unit is an accurate way to accumulate and measure pulse periods and pulse high times – it can accumulate either 16 or 24-bit periods or pulse widths. However some users have experienced problems when using the 24-bit mode on Revision P MC68332 processors, and this bulletin explains the reasons and some work-arounds.

**Order by: EB270/D**

---

## Which Pins on the MC68331/332 and MC68HC16Z1 Need Pullup Resistors

Gives details of which pins on the MC68331, MC68332 and MC68HC16Z1 need pullup resistors, and what the pullup value should be.

**Order by: EB273/D**

---

## Generating Interrupts on the Time Processor Unit

This engineering bulletin describes the three steps required to generate interrupts on a TPU channel, with example code.

**Order by: EB274/D**

---

## Example Using the Queued Serial Peripheral Interface on Modular MCUs

The Queued Serial Peripheral Interface (QSPI) is compatible with the Serial Peripheral Interface (SPI) used in the M68HC11 and M68HC05 Families of microcontrollers. However it differs in having a queue, with programmable queue pointers, to allow up to 16 transfers without CPU intervention. It also has a wrap-around mode that allows continuous transfers to and from the queue, a feature that is useful in applications such as the control of an A/D converter. This bulletin uses an example program to explain how to initialize the QSPI in wrap-around mode, and how to enable interrupts.

**Order by: EB275/D**

---

## Using the ITC Function on the Time Processor Unit A

The ITC function of the TPU counts input transitions and timestamps the last two. This engineering bulletin explains how to use it, and includes program code for both CPU32 and CPU16-based microcontrollers.

**Order by: EB276/D**

---

## Coherency in the Time Processor Unit (TPU)

Sometimes two parameters in the Time Processor Unit (TPU) parameter RAM must be updated together – if one is updated the other must also be updated immediately, a condition called Coherency. Some of the current TPU functions have two parameters that must be read or written coherently; this bulletin identifies some examples and explains how to handle them.

**Order by: EB277/D**

## Latency on the Time Processor Unit

Each time function on the Time Processor Unit (TPU) is divided into several states that are executed by the microengine. The states consist of microinstructions that are stored in the TPU, or stored in RAM if the TPU is running in emulation mode. Each state requires a certain amount of time to execute, and the states are not necessarily executed back to back. Thus there is a certain amount of latency in the TPU. This bulletin discusses the factors involved in latency calculations, and some of the effects of TPU latency.

**Order by: EB278/D**

---

## Low Output Levels on Output Pins

A problem reported occasionally with the MC68332 and related devices is that a pin or group of pins has a logic high level of about 3 volts. This is usually due to poor power pin connections, and this bulletin highlights the areas to watch.

**Order by: EB279/D**

---

## Programming the Channel Control Registers on the Time Processor Unit

The Time Processor Unit (TPU) has several control registers that are shared by all 16 channels. Some are not always used, but four of them must always be initialized. This bulletin explains their functions.

**Order by: EB280/D**

---

## Halting and Re-Starting the Queued Serial Peripheral Interface on Modular Microcontrollers

Halting the Queued Serial Peripheral Interface (QSPI) on modular microcontrollers before the end queue pointer is reached requires use of a special sequence, to ensure that the current serial transfer completes and the QSPI halts at a known state on a boundary between two queue entries. This bulletin explains the correct procedure, involving three bits in three different registers, and includes example code for CPU32 and CPU16.

**Order by: EB281/D**

---

## Using the Output Compare Function on the Time Processor Unit and an Example that Includes PPWA

The Output Compare function (OC) on the time processor unit has three modes of operation: Read TCR1/TCR2, Host Initiated Pulse, and Continuous Pulse. This bulletin describes the use of OC, with examples in Host Initiated Pulse and Continuous Pulse modes.

**Order by: EB282/D**

---

## C Macro Definitions for the MC68HC11C0

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC11C0, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB283/D**

---

## C Macro Definitions for the MC68HC(7)11D3/D0

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC(7)11D3 and MC68HC(7)11D0, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB284/D**

---

## C Macro Definitions for the MC68HC(7)11E20

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC(7)11E20, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB285/D**

---

## C Macro Definitions for the MC68HC11A8/A7/A1/A0

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC11A8, MC68HC11A7, MC68HC11A1 and MC68HC11A0, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB286/D**

---

## C Macro Definitions for the MC68HC(7)11E9/E8/E1/E0

With more microcontroller users moving to high level languages such as C, macro definition files like the one outlined in this bulletin can speed software development. The file reproduced here is available on the web, and the download includes an ASCII text copy of this documentation as well as the file itself. Motorola's designated register and bit names are used, and any user already familiar with M68HC11 assembly language and architecture – a requirement even for those who think they will only program in C – will be able to use the file easily.

**Order by: EB287/D**

## C Macro Definitions for the MC68HC11ED0

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC11ED0, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB288/D**

---

## C Macro Definitions for the MC68HC11F1

With more microcontroller users moving to high level languages like C, macro definition files can speed software development efforts. The file for the MC68HC11F1, reproduced and described in this Bulletin, is available on Motorola's Freeware Data System. It uses Motorola's designated register and bit names for the device, and any user familiar with M68HC11 assembly language and architecture will readily be able to make use of it.

**Order by: EB289/D**

---

## Programming MC68HC811E2 Devices with PCbug11 and the M68HC11EVBU

This note provides a step-by-step procedure describing how to program MC68HC811E2 devices using the M68HC11EVBU Evaluation Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB291/D**

---

## Initialization Considerations when Moving from the BUFFALO Monitor to a Standalone MC68HC11

Analog to digital converter code written under the BUFFALO monitor will often not run on a standalone MC68HC11 because of a simple error. The A/D converter must be initialized by writing a 1 to the power-up bit in the option register. This engineering bulletin fils in the details.

**Order by: EB292/D**

---

## Simplify MC68HC711E20 EPROM Programming with PCbug11

This note provides a step-by-step procedure describing how to program MC68HC711E20 devices using the M68HC711E9PGMR Programming Board in conjunction with the PCbug11 software, since the GLOADE9 software shipped with the board cannot be use to program MC68HC711E20 devices. PCbug11 is available from the download area of the Microcontroller website.

**Order by: EB293/D**

## How to Write the 64-Cycle Time-Protected Registers on M68HC11 Development Tools

To achieve maximum flexibility, MC68HC11 software can control a number of hardware options which customize the operating environment. However it is necessary to take precautions against runaway software that could change the hardware configuration unintentionally. The result is a set of registers known as the time-protected registers. They are writable in normal operating modes (single-chip and expanded) only during the first 64 clock cycles after reset. This bulletin explains how to write to the registers when under the control of the BUFFALO monitor, which uses all of the first 64 cycles before control is passed to the user.

**Order by: EB294/D**

---

## Programming the EEPROM on the MC68HC811E2 with the M68HC11EVM Board

Special considerations must be taken into account when using the M68HC11EVM to program the on-chip EEPROM of the MC68HC811E2. The procedure is the same as that used to program the CONFIG resister of the device. This bulletin presents a step-by-step procedure.

**Order by: EB295/D**

---

## Programming MC68HC711E9 Devices with PCbug11 and the M68HC11EVBU

This note provides a step-by-step procedure describing how to program MC68HC711E9 devices using the M68HC11EVBU Evaluation Board in conjunction with the PCbug11 software, which is available from the download area of the Microcontroller website (and is also supplied with some evaluation boards).

**Order by: EB296/D**

---

## Programming the BUFFALO Monitor into an MC68HC711E9

If communication with an EVBU containing an MC68HC711E9 cannot be established using a terminal emulation program such as Procomm or Kermit, the most likely reason is that the EPROM in the MC68HC711E9 has not been programmed with the BUFFALO monitor. This bulletin explains how.

**Order by: EB298/D**

---

## Why M68HC711D3PGMR Software Does Not Run on 486 33MHz Computers

M68HC711D3PGMR software cannot communicate with an HC11 when it is run on a 33MHz or faster 486 IBM PC or clone. This bulletin explains why. Unfortunately the only solution is to run GLOADD3 on a slower machine.

**Order by: EB299/D**

## Programming EEPROM on the MC68HC811E2 During Program Execution

The 2K of EEPROM in the MC68HC811E2, currently the largest EEPROM array in the M68HC11 Family of microcontrollers, can be used for both program code and data. The device also has 256 bytes of RAM; some users may choose to use this RAM for program variables during execution, others may prefer to store data and variables in the non-volatile EEPROM so that data will still be valid after a power-off/power-on sequence. This bulletin explains how to do this.

**Order by: EB301/D**

## Handling Considerations for Avoiding Intermittent Programming and Execution Failures with MC68HC11-Windowed Devices

Problems when attempting to program MC68HC11-windowed EPROM devices, or to use them in target applications, may be attributable to lead alignment problems. The leads on these devices (with the FS package suffix) are more flexible than those on identical EPROM devices in PLCC packages (with the FN package suffix). This additional flexibility may cause problems in applications where they are handled extensively.

**Order by: EB303/D**

## Startup Problems When Using a Software Background Mode Debugger and Booting from RAM or an Empty ROM Socket

The method used by many in-circuit debuggers for M68300 and M68HC16 microcontrollers is to invoke the MCU's background mode. There are several methods for entering background mode, and this bulletin addresses some concerns that must be taken into account.

**Order by: EB305/D**

## Using Exercise 7 on the M68HC16Z1EVB and the Necessity of Word Alignment

The M68HC16Z1EVB is shipped with eight examples of working programs that can be run on the board. Exercise 7 uses an output compare to drive three input captures, in the general-purpose timer. The output compare is set to toggle each time the 16-bit counter reaches $1000; one input capture triggers on both edges, one on rising edges only, and the third on falling edges. Various external actions indicate correct operation of the exercise. This bulletin describes the exercise, and explains why executable code must start on an even address.

**Order by: EB306/D**

## Using Exercise 8 on the MC68HC16Z1EVB

The MC68HC16Z1EVB is shipped with eight exercise programs as examples of working programs which can be run on the board. One of these uses the Burr-Brown PCM56P D/A converter. The disk shipped with early packages of the board contains a line of code which will not assemble correctly. This bulletin explains the simple correction.

**Order by: EB309/D**

## Using Bus Error Stack Frames to Diagnose CPU32 Released Write Faults

Discusses a methodology that uses bus error (BERR) stack frames to diagnose problems in a memory transaction in the write unit of the CPU32. It is generally not possible to determine absolutely which instruction generated the memory transaction that caused the BERR resulting from a 'released write fault'. By studying BERR stack frames, the location and cause of programming or hardware problems can be localized and sometimes pinpointed.

**Order by: EB310/D**

## Replacing 68HC11KA4/KA2 MCUs with 68HC11KS2/KS8 MCUs

The 68HC11KSx Family was designed to provide more ROM and RAM for 68HC11KA4 users, and a software selectable Slow Mode feature has been added to reduce power consumption. Both families have the same pinouts in the 68-pin PLCC and 80-pin TQPF packages. Most users can run 68HC11KAx code on 68HC11KSx devices without changes; however the system memory configuration and the software use of registers and control bits must be reviewed.

**Order by: EB312/D**

## Updating the Software Watchdog on M683xx and MC68HC16 Products

The software watchdog on MC683xx and MC68HC16 devices is used to detect software that may not operate properly. It is enabled from the release of external reset, and if it is not reset at appropriate intervals it will time out and cause a software watchdog reset, equivalent to a hardware reset. Some hardware considerations limit the rate at which it can be reset. This bulletin explains the details and potential pitfalls.

**Order by: EB314/D**

## Getting Started with the FDDI ADS Board

An introductory tutorial explaining how to use the Motorola FDDI fibre optic Applications Development System M68FDDIADS (commonly known as FADS). It describes the general features of FADS and some of the specific commands. The M68FDDIADS is a demonstration system and basic development platform for the Motorola FDDI chip set; all four chips are present on the board, together with an MC68020 MPU with 1Mbyte of RAM and a ROM-resident debug monitor. A comprehensive system of software-driven menus allows access to all chips and Station Management (SMT) via the user interface.

**Order by: EB406/D**

## PASM05 to INTROL M68HC05 Assembler Conversion

This bulletin describes the most common differences between Motorola's PASM05 V0.05 Assembler and the third party INTROL Assembler V3.06, for MS-DOS Hosts used with M68HC05 development systems. Understanding the differences will help reduce the engineering time required to convert the syntax from PASM05 to INTROL; with the use of an intelligent Editor that supports MACRO EDITING functions, multiple conversions can be achieved with one keypress. The EDITOR used in the conversion described here was "BRIEF V3.0" by UnderWare Inc.

**Order by: EB410/D**

---

## Resetting MCUs

MCU Reset in its most basic form ensures that the MCU starts executing code in a controlled manner when power is applied. It may also be used to prevent the device running out of specification, and can cause a system reset at the board level if the MCU executes code in an unexpected way (watch-dog). But this simple function can cause problems since different applications impose different start-up and power down conditions. This document covers the main issues relating to Reset and aims to lead the user of HC05 and HC11 devices to a safe and reliable approach for their particular application.

**Order by: EB413/D**

---

## Low Power Write Enable Generation for M68300 Family Microprocessors

Dynamic bus sizing support circuitry is normally implemented using PALs. However in battery-powered applications PALs can draw unacceptable amounts of power. The circuit presented here implements byte write enable functions for a 16-bit processor/memory combination, using just two low power 74HCTTL devices.

**Order by: EB414/D**

---

## Modular Target Cables for Motorola Development Systems

Target cables connect emulation equipment such as an EVS or MMDS to the user's target system by providing an interface from the emulator to the MCU socket in the user's system. Traditionally each MCU package requires a different cable, adding time and development cost to the project. However a modular strategy allows the use of common parts of target cables, leading to more rapid cable development and a reduction in costs. This bulletin discusses the cable strategy the supports the M68HC05, M68HC08 and M68HC11 families.

**Order by: EB416/D**

---

## ROMed HC11E32 and HC11PH8 Including Buffalo Monitor and PCbug11 Talker

This bulletin describes the monitor software contained in ROMed versions of the MC68HC11PH8 and the MC68HC11E32, including the Buffalo Monitor and PCbug11 Talker. These ROM codes also include the multiband RDS radio application described in application notes AN494/D and AN495/D. Includes a description of a minimal circuit for running the monitors.

**Order by: EB419/D**

---

## Converting DSP56001-Based Designs to the DSP56002

**Rev 1**

This bulletin details the differences between the DSP56001 and the DSP56002 which need to be taken into account when redesigning a DSP56001-based system to use the DSP56002. They fall into two main categories – changes which must be taken into account by the designer, and changes which the designer may choose to implement when appropriate.

**Order by: EB420/D**

---

## The Motorola MCAN Module

The Control Area Network (CAN) was developed by Robert Bosch GmbH as a serial communications protocol for use in automotive applications. It has been optimised for operation in interrupt-driven, real-time environments, and in addition to its key role in automotive multiplexing applications, it is finding widespread use for industrial control. Motorola has now integrated the MCAN (Motorola CAN) module into several MC68HC05-based MCUs. This bulletin describes the module and its registers, the interface with the CAN bus, and examples of initialisation, transmit and receive software.

**Order by: EB421/D**

---

## Enhanced M68HC11 Bootstrap Mode

Motorola has enhanced the capability of the Special Bootstrap Mode of many of the M68HC11 family MCUs, following an increase in ROM sizes. This bulletin provides the bootstrap mode listings for the MC68HC11ED0, MC68HC711EA9, MC68HC11PH8, MC68-HC711PH8, secured MC68HC711E20, secured MC68HC711E32 and secured MC68HC11E32. Mode enhancements include the addition of autostart facilities for PLL systems, enhanced security options, and embedded PCbug11 talkers.

**Order by: EB422/D**

---

## Using the TPU Function Library and TPU Emulation Mode

The Time Processor Unit (TPU) is an on-chip sub-module used in Motorola's M68000 and M68HC16 families of modular microcontrollers. It is dedicated to performing complex timing and I/O tasks to relieve the CPU of most of the timing overhead. This note describes its function library and explains how to use it. It contains sufficient information for a user to select a new suite of functions from the library, assemble them to run in TPU emulation mode, and submit the debugged code as a new TPU micro-ROM code mask. The user should also refer to the TPU Reference Manual (TPURM/AD) for detailed information.

**Order by: TPUPN00/D**

## Queued Output Match TPU Function (QOM)

**Rev 2**

In the M68300 Family Time Processor Unit, the QOM function generates complex pulse trains without CPU intervention using a sequence of output matches. An output match causes a programmable pin response when a user-defined value is matched by the value of a free-running counter. QOM generates multiple output matches using a queue of offset times and pin responses in Parameter RAM. Various modes of queued operation are supported. This note provides a detailed description of the QOM function, including performance notes and several example pulse trains.

**Order by: TPUPN01/D**

---

## Fast Quadrature Decode TPU Function (FQD)

The Fast Quadrature Decode (FQD) is a TPU input function that uses two channels to decode a pair of out-of-phase signals to increment or decrement a position counter. It is generally used for decoding position and direction information from a slotted encoder in motion control systems. FQD differs from the QDEC function in operating in either normal or fast modes, switching dynamically between the two according to the encoder speed. In Fast mode, the TPU can decode reliably at more than four times the normal maximum count rate. This Programming Note describes the use of the FQD.

**Order by: TPUPN02/D**

---

## Frequency Measurement TPU Function (FQM)

Frequency Measurement (FQM) is a TPU function that counts the number of pulses presented to a channel pin within a user-specified time window. Either rising or falling edges can be used at the beginning of a pulse, and pulses can be accumulated for a single window time or in repetitive windows. The pulse count is available to the user as a 16-bit number. This Programming Note describes the parameters, registers, configuration and performance of FQM. It includes four examples demonstrating the use of the function.

**Order by: TPUPN03/D**

---

## Table Stepper Motor TPU Function (TSM)

**Rev 1**

Table Stepper Motor provides the TPU with the capability to drive two-phase stepper motors in full- or half-step modes. The TPU can accelerate the motors, run them at constant speed (or slew), and decelerate them independently of the CPU, which need only initialize the function once and then supply a desired position each time a move is required. The acceleration/deceleration profile is freely configurable by the user with a variable length table that offers up to 82 step rates. The TPU can control up to eight motors in full-step mode, four motors in half-step mode or a combination of both.

**Order by: TPUPN04/D**

---

## Multichannel PWM TPU Function (MCPWM)

This TPU output function uses externally-gated multiple channels to produce sophisticated pulse-width modulated (PWM) signals which can be used for a variety of applications including motor control. MCPWM allows a user to select edge-aligned or center-aligned timing relationships between multiple PWM waveforms. Center-aligned relationships include dead time and inversion options to support driving H-bridges and inverters. MCPWM can also generate a programmable periodic CPU interrupt request for high time updating.

**Order by: TPUPN05/D**

---

## Programmable Time Accumulator TPU Function (PTA)

The Programmable Time Accumulator function measures the high time, low time or period of an input signal over a user-defined number of periods, presenting the result to the host CPU in the form of a 32-bit accumulation. PTA does not link to other TPU channels at the end of accumulation – PPWA should be used if this function is required. This note provides a detailed description of parameters and parameter RAM assignment, the TPU host interface, performance and use of the function, and examples of typical applications.

**Order by: TPUPN06/D**

---

## Asynchronous Serial Interface TPU Function (UART)

**Rev 1**

The UART function uses two TPU channels to provide a 3-wire (RxD, TxD and GND) asynchronous serial interface. All the standard baud rates and parity settings can be selected. The CPU interface to UART consists of a command register which defines the operation (number of data bits, baud rate, parity); a status register which gives information about the data register and errors; and a data register which holds the data to be transmitted or data that has been received. These registers are implemented for the UART function using the TPU parameter RAM and host sequence bits.

**Order by: TPUPN07/D**

---

## New Input Capture/Input Transition Counter TPU Function (NITC)

The NITC function detects rising and/or falling input transitions. When a transition is detected the current TCR timer value or a parameter RAM value is captured. The channel continues to detect and count input transitions until it has counted the maximum programmable number stored in the parameter MAX_COUNT. NITC can count the programmed maximum number of transitions continually, or can count once and then stop. Once the programmed number of transitions is counted, it can send an interrupt request to the host CPU and generate a link to a sequential block of up to eight channels.

**Order by: TPUPN08/D**

## Multiphase Motor Commutation TPU Function (COMM)

**Rev 1**

COMM uses multiple TPU channels to produce drive-enable signals for commutating brushless motors. Motor types supported include three- and four-phase brushless DC and three-phase switched reluctance motors. It is used in conjunction with other TPU functions to provide a choice of sensored (Hall effect or optical, with HALLD) or sensorless (from an encoder, with QDEC, FQD or DUC) commutation. The signals produced by COMM are gated externally with a PWM, also generated by the TPU, to drive the motor. COMM is a flexible function and may also meet the requirements of other multisignal applications.

**Order by: TPUPN09/D**

## Hall Effect Decode TPU Function (HALLD)

The Hall effect decode function is a TPU input function that uses two or three channels to decode signals from Hall effect sensors into a state number. The function is designed primarily for use with the COMM function in brushless motor applications, in which case the state number represents the current angular position of the rotor, but can also be used in other applications requiring the decoding of multiple digital inputs.

**Order by: TPUPN10/D**

## Period/Pulse Width Accumulator TPU Function (PPWA)

The period/pulse-width accumulator (PPWA) function allows any channel to accumulate up to 255 periods or pulse widths, in TCR clock counts. The hexadecimal value that PPWA returns as the accumulated sum must be multiplied by the appropriate TCR resolution to obtain the actual length of time in seconds. PPWA has four operating modes, and can accumulate a 16-bit sum or a 24-bit sum. In the 16-bit modes it can generate a link to a sequential block of up to eight channels after each accumulation period. In all modes, PPWA can generate an interrupt after each accumulation period.

**Order by: TPUPN11/D**

## Output Compare TPU Function (OC)

**Rev 1**

The output compare (OC) function can generate a single output transition, a single pulse, or a continuous 50% duty cycle pulse train, on receiving a link from another channel. The first two actions require the CPU to initiate each output edge or pulse. The third action generates a continuous square wave without CPU intervention. OC can also be used to read the most recent TCR1 and TCR2 values.

**Order by: TPUPN12/D**

## Stepper Motor TPU Function (SM)

The stepper motor control function (SM) accelerates and decelerates a stepper motor linearly, using up to 14 step rates. The CPU provides the desired step position as a 16-bit parameter and the TPU steps the motor to this position using an acceleration/deceleration profile. The target position can be changed by the CPU while the TPU is stepping the motor. A 16-bit parameter initialized for each channel by the CPU defines the output state of the pin. The bit pattern defines the method of stepping, such as full or half step.

**Order by: TPUPN13/D**

## Position-Synchronised Pulse Generator (PSP)

**Rev 1**

PSP generates an output transition referenced to a time determined previously by another channel. Typically it is used with a PMM or PMA function on another channel. There are two operating modes: in angle-angle mode, the rising and falling edges of the output pulse are determined independently of each other; in angle-time mode, the falling edge of the output pulse is determined in reference to the rising edge. Up to 15 PSP function channels may operate with a single input reference channel executing a PMA or PMM function.

**Order by: TPUPN14/D**

## Period Measurement with Additional Transition Detection TPU Function (PMA)

PMA detects additional transitions embedded in a series of input pulses; an additional transition is when the current period is less than the previous period multiplied by a programmable ratio. It has two operating modes: in Count mode it counts the number of extra transitions compared with a programmable maximum value before resetting the TCR2 counter. In Bank mode, the TCR2 counter resets when an extra transition is detected and the flag BANK_SIGNAL is non-zero. PMA is typically used in automotive applications for detecting a reference point in the form of an extra tooth on a flywheel with regularly spaced teeth.

**Order by: TPUPN15A/D**

## Period Measurement with Missing Transition Detection TPU Function (PMM)

**Rev 1**

PMM detects missing transitions embedded in a series of input pulses; a missing transition is when the current period is greater than the previous period multiplied by a programmable ratio. It has two operating modes: in Count mode it counts the number of missing transitions compared with a programmable maximum value before resetting the TCR2 counter. In Bank mode, the TCR2 counter resets when a missing transition is detected and the flag BANK_SIGNAL is non-zero. PMM is typically used in automotive applications for detecting a reference point in the form of a missing tooth on a flywheel with regularly spaced teeth.

**Order by: TPUPN15B/D**

## Pulse Width Modulation TPU Function (PWM)

The PWM output function generates a PWM waveform in which the period and/or the high time can be changed at any time by the CPU. It has two modes of operation: in level mode a 0% or a 100% duty-cycle waveform can be generated; in normal mode, waveforms with duty-cycles between 0% and 100% can be generated. Generally

any change is used in subsequent waveform synthesis after a low-to-high transition. An immediate update is possible in either mode, after which the new period and/or high time is reflected in the output waveform during the immediate host-service state, instead of waiting for the transition.

**Order by: TPUPN17/D**

---

## Discrete Input/Output TPU Function (DIO)

DIO allows the user to configure a TPU channel as an input or output. As an input the channel can be read at any time or sampled at a periodic rate. As an output it can be driven high or low on command by the CPU. A parameter RAM location, PIN_LEVEL, is used to record the 16 most recent states of the channel. The parameter may be updated at one of four conditions: when a transition occurs; when the CPU requests to read the logical value driving the pin; when the CPU requests to drive the pin to a specified logical value; or at a periodic rate specified in the MATCH_RATE register.

**Order by: TPUPN18/D**

---

## Synchronized Pulse-Width Modulation (SPWM)

**Rev 1**

The Synchronized Pulse-Width Modulation (SPWM) function generates a pulse-width modulated (PWM) waveform in which the CPU can change the period or high time at any time. When synchronized to a time function on a second channel, SPWM low-to-high transitions have a time relationship to transitions on the second channel. Includes examples with diagrams of the initial parameter RAM contents, initial control bit settings and diagrams of the output waveforms.

**Order by: TPUPN19/D**

---

## Quadrature Decode TPU Function (QDEC)

The Quadrature Decode function (QDEC) is a TPU input function that uses two channels to decode a pair of out-of-phase signals in order to increment or decrement a counter. It is particularly useful for decoding position and direction information from a slotted encoder in motion control systems, so replacing expensive external solutions. This note provides detailed description of parameters and parameter RAM assignment, the TPU host interface, performance and use of the function, and examples of typical applications.

**Order by: TPUPN20/D**

# Applications Documents

## Alphanumeric Index

# Data Books

## Timing Solutions

**Rev 6**

With frequencies approaching 50MHz in today's RISC and CISC microprocessor systems, precise clock signals are required to maintain a synchronous system. This data book presents Motorola's range of low skew clock drivers, together with a discussion of design considerations to help achieve the best performance.

**Order by: BR1333/D**

---

## RF Device Data

**Rev 10**

Provides data sheet information on Motorola's extensive range of RF products. Products are categorized into three main sections – Discrete Transistors, Monolithic Integrated Circuits and Amplifiers – and a comprehensive Selector Guide lists the devices under a variety of application, frequency band and package classifications. Case dimensions and a competitor cross reference are included – the cross reference lists functionally similar products under a 'closest replacement' heading in order to accommodate the unique products that now exist as a result of new technologies and packaging concepts.

**Order by: DL110/D**

---

## Fast Static RAM – Component and Module Data

**Rev 5**

Motorola offers a broad range of Fast SRAMs for virtually any digital data processing application. This book contains complete specifications for individual FSRAM circuits in data sheet form, together with an introduction to Motorola's quality and reliability programme, and an applications section. The book is divided into BiCMOS, CMOS, Application Specific and Module products, and includes a comprehensive selector guide, an industry cross reference, and 126 pages of applications information.

**Order by: DL156/D**

## LonWorks Technology Device Data

**Rev 5**

Through the LONWORKS program, Motorola offers the MC143120 and MC143150 NEURON chips. These are sophisticated VLSI devices that make it possible to implement low-cost local operating network applications. This book combines specifications for these parts with a large selection of applications literature. Other sections include a Technology/Licensing Overview, a summary of the Neuron Chip Family hardware resources, Communications and Subsystems, I/O Interfaces, Programming Model, the LonTalk Protocol, and details of the data structures.

**Order by: DL159/D**

---

## Pressure Sensor Device Data

**Rev 4**

Provides basic information on Motorola's pressure sensors, with application ideas and data sheets on this broad product line. Includes an introduction to the principle of operation, a separate data sheet section for the recently introduced Signal Conditioned and High Impedance products, data sheets on all the other devices in the family, Quality and Reliability data, 178 pages of applications information, plus package outlines and handling recommendations.

**Order by: DL200/D**

---

## Senseon: Pressure Sensor Distributor Handbook

**Rev 1**

Intended to introduce Motorola's pressure sensors to product distributors, this handbook is a guide to the basic what, where, how and why of SENSEON Pressure Sensors. It is comprehensive yet lighthearted, and requires minimal technical background in order to grasp the basic concepts.

**Order by: HB218/D**

# Selector Guides and Application Literature

## Motorola Quality System Review Guidelines

**Rev 5**

Motorola's Quality System Review (QSR) is a means by which the company evaluates the continuing health of the Quality System in each of its major business units and suppliers. It defines a vision of how Motorola's business should be conducted, sets a common goal of perfection, and provides an awareness of Quality System requirements across the whole organisation. The QSR Guidelines are provided to train the reviewers, aid the understanding of each review question and assist in the scoring process. They may also be of interest to Motorola's quality conscious customers.

**Order by: BR1202/D**

---

## Sensor Device Information Matrix – Quarter 1, 1999

This comprehensive matrix is a selector guide to application notes and related device information on Motorola's pressure sensors and accelerometers. Documents are listed under Uncompensated Pressure Devices, On-Chip Temperature Compensated and Calibrated Devices, Integrated Pressure Sensors, and Accelerometers.

**Order by: BR1512/D**

---

## PowerPC Resource Guide

A guide to PowerPC hardware and software products from Motorola, IBM and third party developers. Sections include Microprocessors and Peripherals, Hardware and Software Development Tools, Board Level Products & Evaluation Boards, and Consulting Services. Lists worldwide sales and distribution offices.

**Order by: BR1724/D**

---

## Occupant Safety Systems Solutions

**Rev 1**

A quick reference selector guide to Motorola's microcontrollers, accelerometers, pressure sensors and SMARTMOS devices relevant to occupant safety systems.

**Order by: BR1781/D**

## Novel Digital Signal Processing Architecture with Microcontroller Features

Traditional digital signal processors are designed to execute signal processing algorithms as efficiently as possible. This has led to some serious compromises between developing a good DSP architecture and a good microprocessor architecture. This paper presents Motorola's new 16-bit architecture, used in the DSP56800 family, which is designed to maintain the performance of the DSP while adding microcontroller functionality. Target applications are those demanding low costs with moderate performance, such as wireline and wireless modems, digital wireless messaging, digital answering machines and featurephones, servo and AC motor control, and digital cameras.

**Order by: DSP56800WP1/D**

---

## Embedded Developer Pocket Guide

**Rev 4**

This Pocket Guide contains a listing of virtually all Third Party Embedded Developers supporting Motorola's 68K, ColdFire and PowerPC embedded processors through the High Performance Embedded System Division's (HPESD) Developer Program. This program comprises more than 50 third party developers, and makes available the broadest possible portfolio of development tools to enable Motorola's customers to deliver innovative, world-class products. Each page of this Guide provides an overview of the developer, with contact details and a listing of development tools and supported MCUs.

**Order by: EMDVPOC/D**

---

## RF Application Reports

A collection of 92 of Motorola's Application Notes, Article Reprints and Engineering Bulletins concerned with RF products. Topics include RF Power MOSFETs, RF Power Bipolar, RF Integrated Circuits and RF Linear Amplifiers.

**Order by: HB215/D**

## RF Products Selector Guide

**Rev 19**

This publication presents RF products of Motorola Phoenix, Motorola Toulouse (France), and Motorola Hong Kong. The RF products are categorized by Power FETs, Power Bipolar, Small Signal Bipolar, Integrated Circuits, and Low and High Power Amplifiers. Includes a list of relevant applications literature, case outlines, and an industry cross reference information with an indication of devices not recommended for new designs.

**Order by: SG46/D**

---

## Master Selection Guide

**Rev 22**

The Master Selection Guide lists all of Motorola's semiconductor products – the broadest product line in the industry. It provides the engineer with a means of first-order selection of devices for specific applications. Sections include ASICs; Microcomputer Components; TTL, ECL, CMOS and Special Logic; Linear/Interface Circuits; Discrete and Military Products; the presentation is appropriate to the product families, but generally follows the standard Selector Guide and Cross Reference format. In addition, a Device Index, Subject Cross Reference and comprehensive Contents section allow the efficient location of specific products.

**Order by: SG73/D**

---

## Digital Signal Processors Update

**Rev 20**

This selector guide describes Motorola's architecturally-compatible Digital Signal Processing Chips, including 16- and 24-bit fixed point and 32-bit floating point families, peripheral chips, and development tools.

**Order by: SG146/D**

---

## Sensor Products Division

**Rev 31**

This quarterly publication details the pressure and acceleration sensors and evaluation tools available from the Sensors Products Division.

**Order by: SG162/D**

---

## Fast Static RAM Division Product Update

**Rev 29, 4Q99**

This selector guide provides an overview of Motorola's fast-growing FSRAM product line. Included are synchronous, asynchronous and FSRAM modules.

**Order by: SG171/D**

## Networking Systems Division and Personal Computing Division: Product Information

**Rev 15, 1Q00**

This selector guide lists the devices in Motorola's PowerPC microprocessor family, including devices for embedded applications, plus 68K Networking and Communications devices and development tools. Includes package illustrations, part number breakdown, and a table of available documentation.

**Order by: SG175/D**

---

## Digital Audio Solutions

**Rev 3**

A quarterly selector guide listing Motorola's products for audio applications, including DSP hardware, microcontrollers and software solutions, plus Motorola and third party development tools.

**Order by: SG185/D**

---

## Automotive Selector Guide

**Rev 2, 1Q00**

A comprehensive selector guide to Motorola's semiconductor products for the automotive industry, including microcontrollers from the 8, 16 and 32-bit families, plus drivers, controllers, sensors and other devices designed specifically for automotive applications.

**Order by: SG187/D**

---

## North America Sales Price List

**Rev 14**

This guide lists North American suggested resale prices for Motorola commercial components and development systems. A Quick Reference lists new devices, deleted devices and lifetime buy products. Includes Motorola Sales Offices, standard policies and disclaimers, and software licenses.

**Order by: SG379/D**

---

## Motorola RF CATV Distribution Amplifiers

**Rev 3**

Motorola has excelled as a leading supplier of innovative technical products to the cable TV market since its inception. This selector guide lists outline specifications for Motorola's CATV forward amplifiers, reverse amplifiers and fiber optic receivers; many are state-of-the-art products using transistors with sub-micron geometries.

**Order by: SG382/D**

---

## Motorola RF LDMOS Product Family

**Rev 6**

Motorola's LDMOS (Laterally Diffused Metal Oxide Silicon) process is fast becoming the technology of choice in new communications products, making high power, high frequency RF amplifier designs simpler and more cost effective. This selector guide summarizes

the devices available in the areas of RF High Power Transistors, Discrete Transmitter Devices for battery applications, RF Amplifier Modules, and RF Monolithic ICs.

**Order by: SG384/D**

## Semiconductor Products for Wireless Communications

Motorola provides a number of unique, state-of-the-art silicon solutions for wireless communications, with particular emphasis on the new digital systems. This document lists a sample of devices from the vast portfolio of products for DECT, GSM, PCN, CT2 and Wireless LAN applications.

**Order by: SG417/D**

# User's Manuals

## Analog-to-Digital Converter Reference Manual

This manual describes the capabilities, operation and functions of the analogue-to-digital converter (ADC) module incorporated in many of the MCUs in Motorola's modular microcontroller family. The module is a unipolar, successive-approximation converter with eight modes of operation and selectable 8 or 10-bit resolution. Monotonicity is guaranteed for both 8 and 10-bit conversions. The manual includes a functional overview, an explanation of ADC control through the Intermodule Bus (IMB), and descriptions of the analogue and digital subsystems.

**Order by: ADCRM/AD**

---

## Byte Data Link Controller Reference Manual

The Byte Data Link Controller (BDLC) is a serial communication module which allows the user to send and receive messages across an SAE J1850 serial communication network. The user's software handles each transmitted or received message on a byte-by-byte basis, while the BDLC performs the network access, arbitration, message framing and error detection. This manual is intended as an aid to the development of software that uses the BDLC to perform SAE J1850 communication; some implementations of the module may provide enhanced capabilities and software designers should also refer to the MCU specification.

**Order by: BDLCRM/AD**

---

## M68HC08 Central Processor Unit Reference Manual

**Rev 1**

The CPU08 is the central processing unit of the M68HC08 Family of MCUs. It is fully object code compatible with the M68HC05, offering increased performance with no loss of software investment. It also appeals to users of other MCU architectures who need its speed, low power consumption and processing capabilities. This manual provides an overview of the CPU08 and its architecture, describes its interrupts, reset procedures and addressing modes, and gives detailed Instruction Set information in an instruction-per-page format.

**Order by: CPU08RM/AD**

## CPU12 Reference Guide

**Rev 1**

A convenient reference guide providing quick access to essential CPU12 information, including the Programming Model, Instruction Set Summary, Postbyte Encoding, Memory Expansion and Opcode Map.

**Order by: CPU12RG/D**

---

## CPU12 Reference Manual

**Rev 1**

The CPU12 is a high-speed, 16-bit processing unit that has a programming model identical to that of the industry standard M68HC11 CPU. Its instruction set is a proper superset of the M68HC11 instruction set, and HC11 source code is accepted by CPU12 assemblers without change. It offers an extensive set of indexed addressing capabilities in addition to the addressing modes found in other Motorola MCUs. The main goal of this manual is to provide professionals and students in electronic design and software development with the information necessary to implement control systems using M68HC12 devices.

**Order by: CPU12RM/AD**

---

## M68HC16 Family Reference Manual

**Rev 2**

The CPU16 is a high speed 16-bit processor module that allows modular microcontrollers to provide an upgrade path for M68HC11 users while maintaining compatibility with existing systems. Its architecture is a superset of the M68HC11 architecture. This manual describes register organisation, memory management, bus interfacing, addressing modes and instruction set. Instructions are also described on an instruction-per-page basis in alphanumeric order. Additional sections cover instruction timing, exception processing, on-chip development support and digital signal processing (DSP) capabilities.

**Order by: CPU16RM/AD**

## CPU32 Central Processor Unit Reference Manual

**Rev 1**

This Reference Manual describes the capabilities, operation and programming of the CPU32 processor module integrated in some members of the M68300 Family of embedded controllers. It is written for systems designers, and systems and applications programmers. The manual provides a full description of the instruction set, with clock cycle timing – instructions are based on the MC68000, with support for many MC68020 extensions plus new instructions for controller applications. It also describes the architecture, addressing modes, data organisation, exception processing and on-chip development support.

**Order by: CPU32RM/AD**

---

## Configurable Timer Module Reference Manual

The Configurable Timer Module (CTM) is one of the modules used in Motorola's microcontroller family. Modules are connected together by the InterModule Bus (IMB), but the CTM is unusual in that it is in itself modular. This manual introduces the CTM, and details the operation of its internal bus with the IMB, its interrupt functions, and the Counter Prescaler, Free-Running Counter, Modulus Counter, Single Action, Double Action and Pulse Width Modulation submodules. There is a section on electrical specifications and timing information, and appendices provide a register summary and an example of a typical implementation.

**Order by: CTMRM/D**

---

## DMA08 Direct Memory Access Reference Manual

Direct Memory Access (DMA) is usually associated with larger computer systems, where it allows blocks of data to be moved around the system with minimal processor intervention. DMA is the first example of co-processing associated with Motorola's modular HC08 family. This reference manual introduces version A of the DMA08, the DMA module of the HC08 family. Version B of the module has some differences, and is discussed in an appendix. Sections include an Overview, Module Description, Transfer Operation, Register Description and Application Examples.

**Order by: DMA08RM/AD**

---

## DSP56000 Digital Signal Processor Family Manual

**Rev 1**

Motorola's DSP56000 Family of 24-bit general purpose Digital Signal Processors features a modular chip layout based round a standard central processing module. This manual describes this module in detail and provides practical information for designers. After an introduction to digital signal processing, sections include DSP56000 Central Architecture Overview, Data Arithmetic Logic Unit, Address Generation Unit, Program Control Unit, Instruction Set Description, Processing States, External Memory Port, PLL Clock Oscillator and On Chip Emulator. A 338 page alphabetic appendix describes each instruction in detail. (Specific details of the DSP56000/1 devices are given in DSP56000UM/AD.)

**Order by: DSP56KFAMUM/AD**

## DSP56L811 Evaluation Module User's Manual

**Rev 1**

Describes the basic structure and operation of the DSP56L811 Evaluation Module (DSP56L811EVM), and details the additional equipment required to use it, the specifications of the key components, the software provided – including demonstration code, self-test code and software required to develop and debug sophisticated applications – plus detailed schematic diagrams and a parts list. Includes both a Quick Start guide and detailed information about key components.

**Order by: DSP56L811EMUM/AD**

---

## DSP56L811 User's Manual

**Rev 1**

Thee DSP56L811 is a member of the DSP56800 family of core-based DSPs. This general purpose DSP combines processing power with configuration flexibility, making it a cost-effective solution for both signal processing and control applications. It uses an MPU-style, general purpose 16-bit DSP core plus program and data memories. This manual describes the DSP56L811, its memory, operating modes and peripheral modules, and should be read in conjunction with DSP56800FAM/AD, the DSP56800 Family Manual, which describes the CPU, programming models and instruction set details.

**Order by: DSP56L811UM/AD**

---

## DSP56LF812 16-bit Digital Signal Processor User's Manual

The DSP56LF812 is a member of the DSP56800 family of programmable 16-bit digital signal processors. The core consists of three execution units operating in parallel, allowing as many as six operations during each instruction cycle. Its configuration flexibility, compact program code and low cost make it well suited to cost-sensitive applications such as digital wireless messaging, digital answering machines and feature phones, wireline and wireless modems, servo and AC motor control, and digital cameras. This user's manual describes the DSP56LF812, its memory, operating modes and peripherals; it is intended to be used with the DSP56800 Family Manual (DSP56800FM/AD) which describes the CPU, core programing models and instruction set.

**Order by: DSP56LF812UM/AD**

---

## DSP56002 Digital Signal Processor User's Manual

**Rev 1**

This manual describes the memory, operating modes and peripheral modules of the DSP56002 24-bit Digital Signal Processor (it should be read in conjunction with the DSP56K CPU Manual or Family Manual, which both provide detailed information about the CPU, programming models and instruction set). It includes signal descriptions, memory modules and operating modes, the external memory port, the Port B general purpose I/O and host port, and the multi-function Port C which is used mainly for serial communications.

Appendices contain programming sheets to simplify programming the DSP56002 registers, and a listing of the on-chip bootstrap program.

**Order by: DSP56002UM/AD**

## DSP56004 Digital Signal Processor User's Manual

**Rev 2**

This manual describes the memory, operating modes and peripheral modules of the DSP56004 24-bit Digital Signal Processor (it should be read in conjunction with the DSP56K CPU Manual or Family Manual, which both provide detailed information about the CPU, programming models and instruction set). It includes signal descriptions, the external memory interface, the serial host interface, serial audio interface, and general purpose I/O. Appendices contain a listing of the on-chip bootstrap program, application examples, and programming sheets to simplify programming the DSP56004 registers.

**Order by: DSP56004UM/AD**

## DSP56007 24-bit Digital Signal Processor User's Manual

The DSP56007 is a general purpose digital signal processor designed for audio and sound effects applications. It is based on the DSP56000 core architecture and implemented in the same scalable technology as other members of the 24-bit DSP56000 family. This architecture is especially suited to audio applications, for which 16-bit architectures have insufficient precision and 32-bit architectures are normally too expensive. This user's manual describes the DSP56007, its memory, operating modes and peripherals; it is intended to be used with the DSP56000 Family Manual (DSP56KFAMUM/AD) which describes the CPU, programing models and instruction set.

**Order by: DSP56007UM/AD**

## DSP56009 User's Manual

The DSP56009 is a high performance audio DSP based on the DSP56000 core architecture, and implemented in the same scalable architecture as the DSP56002 and other 24-bit DSP56000 family modular products. As a result of its processing power and large memory it supports a variety of digital audio decompression functions such as Dolby AC-3 Surround, MPEG1 Layer 2 and Digital Theater Systems (DTS). This manual describes the DSP56009 in detail, including its memory, operating modes, external memory and audio interfaces, and peripheral modules.

**Order by: DSP56009UM/AD**

## DSP56011 24-bit Digital Signal Processor User's Manual

The DSP56011 is a high performance programmable DSP developed specifically for Digital Versatile Disk (DVD), high-Definition TV (HDTV) and advanced set-top audio decoding. Its memory configuration and peripherals differentiate this DSP from other DSP56000 family members. The DSP56000 architecture is especially suited to audio applications, for which 16-bit architectures have insufficient precision and 32-bit architectures are normally too expensive. This user's manual describes the DSP56011, its memory, operating modes and peripherals; it is intended to be used with the DSP56000 Family Manual (DSP56KFAMUM/AD) which describes the CPU, programing models and instruction set.

**Order by: DSP56011UM/AD**

## DSP56100 Digital Signal Processor Family Manual

The DSP56100 Family Manual describes the components that are common to all the DSP56100 family members. After an overview of the CPU architecture it provides detailed information on the Data ALU, Address Generation Unit, Program Control Unit and on-chip PLL. There are descriptions of the five processing states, bus operation, OnCE on-chip emulation, application development tools and the Dr. Bub DSP Bulletin Board. The manual includes an overview of the instruction set plus detailed information on each instruction, arranged alphabetically as one instruction per page.

**Order by: DSP56100FM/AD**

## DSP56300 24-Bit Digital Signal Processor Family Manual

**Rev 2**

The new DSP56300 core in Motorola's family of programmable CMOS Digital Signal Processors is capable of executing an instruction every clock cycle, so yielding a twofold performance increase compared to the 56000 core while maintaining object code compatibility with it. It consists of an Expansion Port and DRAM Controller, Data ALU, Address Generation Unit, Instruction Cache Controller, Program Control Unit, DMA Controller, PLL Clock Oscillator, On-Chip Emulator and the Peripheral and Memory Expansion Bus. This manual provides full user information on all these items, plus an alphanumeric page-per-instruction description of the instruction set and timing information.

**Order by: DSP56300FM/AD**

## DSP56301 24-Bit Digital Signal Processor User's Manual

**Rev 2**

The DSP56301 is a member of Motorola's 56300 family of programmable CMOS Digital Signal Processors. Devices in this family are based on the DSP56300 core – capable of executing an instruction every clock cycle – with additional on-chip modules chosen from a library of pre-designed elements. The DSP56301 includes X and Y data RAM, an Instruction Cache and Program RAM, Triple Timer, Host Interface, ESSI Interface and SCI Interface modules. This manual describes these modules, and provides pin descriptions and memory maps.

**Order by: DSP56301UM/AD**

## DSP56302 Evaluation Module User's Manual

**Rev 1**

Describes the basic structure and operation of the DSP56302 Evaluation Module (DSP56302EVM), and details the additional equipment required to use it, the specifications of the key components, the software provided, detailed schematic diagrams and a parts list. Includes both a Quick Start guide and detailed information about key components.

**Order by: DSP56302EMUM/AD**

## DSP56302 User's Manual

The DSP56302 is a member of Motorola's DSP56300 family of programmable CMOS DSPs. It uses the DSP56300 core – a high performance, single-clock-cycle-per-instruction engine providing up to twice the performance of the popular DSP56000 family while retaining code compatibility. A rich instruction set and low power dissipation enables a new generation of wireless, telecoms and multimedia products. This manual describes its memory, operating modes and peripheral modules, including the General Purpose I/O capability, Host Interface (HI08), Enhanced Synchronous Serial Interface, Timer Module, On-Chip Emulation (OnCE) and JTAG Port.

**Order by: DSP56302UM/AD**

## DSP56303 Evaluation Module User's Manual

**Rev 2**

Describes the basic structure and operation of the DSP56303 Evaluation Module (DSP56303EVM), and details the additional equipment required to use it, the specifications of the key components, the software provided, detailed schematic diagrams and a parts list. Includes both a Quick Start guide and detailed information about key components.

**Order by: DSP56303EMUM/AD**

## DSP56303 User's Manual

The DSP56303 is a member of Motorola's DSP56300 family of programmable CMOS DSPs.It uses the DSP56300 core – a high performance, single-clock-cycle-per-instruction engine providing up to twice the performance of the popular DSP56000 family while retaining code compatibility. A rich instruction set and low power dissipation enables a new generation of wireless, telecoms and multimedia products. This manual describes its memory, operating modes and peripheral modules, including the General Purpose I/O capability, Host Interface (HI08), Enhanced Synchronous Serial Interface, Timer Module, On-Chip Emulation (OnCE) and JTAG Port.

**Order by: DSP56303UM/AD**

## DSP56304 User's Manual

The DSP 56304 is a member of the DSP56300 family of programmable CMOS DSPs. It retains code compatibility with Motorola's popular DSP56000 core family, but its rich instruction set offers up to twice the performance to open the door to a new generation of wireless, telecommunications and multimedia products. This manual describes the processor, its memory, operating modes and peripheral modules. Includes a reference section for programmers.

**Order by: DSP56304UM/AD**

## DSP56307 24-bit Digital Signal Processor User's Manual

The DSP56307 is a member of the DSP56300 family of programmable 24-bit digital signal processors, developed to support wireless infrastructure applications with general filtering operations. The on-chip EFCOP processes filter algorithms in parallel with core operation to increase overall DSP performance and efficiency, and the DSP is well suited for high-end multichannel telecomms applications including multi-line voice, data and fax processing, and video conferencing. This user's manual describes the DSP56307, its memory, operating modes and peripherals; it is intended to be used with the DSP56300 Family Manual (DSP56300FM/AD) which describes the CPU, core programing models and instruction set.

**Order by: DSP56307UM/D**

## DSP56309 24-bit Digital Signal Processor User's Manual

The DSP56307 is a member of the DSP56300 family of programmable 24-bit digital signal processors. The DSP56300 core is a high performance engine with a single clock cycle per instruction, and provides up to twice the performance of Motorola's popular DSP56000 core family, while retaining code compatibility. The DSP56309 targets telecomms applications such as multi-line voice, data and fax processing, video conferencing, audio applications and control. This user's manual describes the DSP56309, its memory, operating modes and peripherals; it is intended to be used with the DSP56300 Family Manual (DSP56300FM/AD) which describes the CPU, core programing models and instruction set.

**Order by: DSP56309UM/D**

## DSP56311 24-bit Digital Signal Processor User's Manual

**Rev 1**

The DSP56311 is a member of the DSP56300 family of programmable 24-bit digital signal processors, developed to support wireless infrastructure applications with general filtering operations. The on-chip EFCOP processes filter algorithms in parallel with core operation to increase overall DSP performance and efficiency, and the DSP is well suited for high-end multichannel telecomms applications including multi-line voice, data and fax processing, and video conferencing. This user's manual describes the DSP56311, its memory, operating modes and peripherals; it is intended to be used with the DSP56300 Family Manual (DSP56300FM/AD) which describes the CPU, core programing models and instruction set.

**Order by: DSP56311UM/D**

## DSP56364 User's Manual

The DSP56364 24-bit Digital Signal Processor, based on the DSP56300 architecture, is targeted at applications such as sound field processors and acoustic equalizers that require digital audio signal processing. It combines the high-performance DSP56300 core with the audio signal processing capability of the popular Motorola Symphony family, over which it gives a two-fold performance increase while retaining code compatibility. This user's manual provides a detailed description of the core configuration, memory and peripherals that are specific to the DSP56364.

**Order by: DSP56364UM/D**

## DSP56600 16-bit Digital Signal Processor Family Manual

The DSP56600 family of 16-bit high performance digital signal processors is designed specifically for low-power digital handset cellular applications, and is capable of performing a wide variety of fixed-point DSP algorithms. Its architecture features a central processing module that is common to all family members, plus a variety of highly integrated and cost-effective DSP peripherals. This family manual describes the central processor and instruction set, and is intended to be used with the appropriate user's manual which presents the specific details of the device including operating modes, memory and peripherals.

**Order by: DSP56600FM/AD**

## DSP56602 16-bit Digital Signal Processor User's Manual

The DSP56602 is a member of the DSP56000 family of programmable 16-bit CMOS digital signal processors. It is a general purpose DSP combining processing power with configuration flexibility, making it an excellent and cost-effective solution for signal processing and control functions. This user's manual describes the DSP56602, its memory and operating modes; it is intended to be used with the DSP56600 Family Manual (DSP56600FM/AD) which describes the CPU, programming modes and instruction set.

**Order by: DSP56602UM/AD**

## DSP56603 Evaluation Module User's Manual

Describes the basic structure and operation of the DSP56603 Evaluation Module (DSP56603EVM), and details the additional equipment required to use it, the specifications of the key components, the software provided – including demonstration code, self-test code and software required to develop and debug sophisticated applications – plus schematic diagrams and a parts list. A substantial appendix provides a detailed description of Assembler Directives and Structure Control Statements. Intended for users with experience of DSP development tools.

**Order by: DSP56603EMUM/AD**

## DSP56603 16-bit Digital Signal Processor User's Manual

The DSP56603 is a member of the DSP56000 family of programmable 16-bit CMOS digital signal processors. It is a general purpose DSP combining processing power with configuration flexibility, making it an excellent and cost-effective solution for signal processing and control functions. This user's manual describes the DSP56603, its memory and operating modes; it is intended to be used with the DSP56600 Family Manual (DSP56600FM/AD) which describes the CPU, programming modes and instruction set.

**Order by: DSP56603UM/AD**

## DSP56800 Family Manual

Thee DSP56800 Family is based on the DSP56800 16-bit DSP core, to which a range of standard peripherals can be added to create specific devices. This manual describes the core in detail, and will help the user to understand the operation and instruction set of the DSP56800 Family, and to write code for DSP algorithms, general control tasks, communication routines and data manipulation algorithms. It is intended to be used with the appropriate DSP56800 Family member's User's Manual which will explain the specific features of the device. Also includes instruction timing data and instruction-per-page details of each instruction, plus sources of additional technical support.

**Order by: DSP56800FM/AD**

## DSP56824 16-bit Digital Signal Processor User's Manual

The DSP56824 is a member of the DSP56800 family of programmable 16-bit digital signal processors. The core consists of three execution units operating in parallel, allowing as many as six operations during each instruction cycle. Its configuration flexibility, compact program code and low cost make it well suited to cost-sensitive applications such as digital wireless messaging, digital answering machines and feature phones, wireline and wireless modems, servo and AC motor control, and digital cameras. This user's manual describes the DSP56824, its memory, operating modes and peripherals; it is intended to be used with the DSP56800 Family Manual (DSP56800FM/AD) which describes the CPU, core programing models and instruction set.

**Order by: DSP56824UM/AD**

## Modular Microcontroller Family General Purpose Timer Reference Manual

The General Purpose Timer is one of the modules used within Motorola's family of modular microcontrollers. It is a simple but flexible 11-channel timer for use in systems where a moderate level of CPU control is required, and it communicates with other modules through the InterModule Bus. This manual describes the operation and use of all sections of the module, including Compare/Capture Unit, Pulse Accumulator, Prescaler, PWM Unit, Interrupts and General Purpose I/O. It includes a section of applications information, plus electrical, timing and direct signal descriptions.

**Order by: GPTRM/AD**

## Introduction to the Oncore ChipSet

**Rev 1**

The Oncore ChipSet has been developed to allow the GPS (Global Positioning by Satellite) function to be integrated into existing high-volume application platforms. It includes the same three integrated circuits that are found in Motorola's GT Oncore Receiver: the MRFIC1502 RF IC Downconverter, the MCS38140 Digital Correlator IC, and the MC68331 Microprocessor. This system provides high performance in foliage and urban canyon environments, with fast Time To First Fix (TTFF) and reacquisition. This document contains technical specifications, integration considerations and communications information.

**Order by: HB219/D**

---

## 68HC05C0 General Release Specification

**Rev 1.2**

The 8-bit MC68HC05C0 microcomputer is suitable for applications which require an external address and data bus. It provides a mode select for either a muxed or non-muxed bus, and a clock stretching capability for slower peripherals. On-chip modules include an oscillator, CPU, RAM, serial and parallel I/O, multi-function timer, 16-bit timer and a low-voltage reset. This specification presents the technical details.

**Order by: HC05C0GRS/D**

---

## MC68HC05C12A, MC68HCL05C12A, MC68HSC05C12A General Release Specification

**Rev 3.0**

The MC68HC05C12A is an enhanced version of the MC68HC05C8. It includes keyboard scanning logic, a high-current sink and source pin, a COP watchdog timer, and ROM security. This specification presents the technical details of the device. Appendices detail the differences in the low power MC68HCL05C12A and high speed MC68HSC05C12A versions.

**Order by: HC05C12AGRS/D**

---

## MC68HC05C4A, MC68HCL05C4A, MC68HSC05C4A General Release Specification

**Rev 4.0**

The MC68HC05C4A is an enhanced version of the MC68HC05C4. It includes keyboard scanning logic, a high-current sink and source pin, a COP watchdog timer, and ROM security. This specification presents the technical details of the device. Appendices detail the differences in the low power MC68HCL05C4A and high speed MC68HSC05C4A versions.

**Order by: HC05C4AGRS/D**

## MC68HC05C8A, MC68HCL05C8A, MC68HSC05C8A General Release Specification

**Rev 3.0**

The MC68HC05C8A is an enhanced version of the MC68HC05C8. It includes keyboard scanning logic, a high-current sink and source pin, a COP watchdog timer, and ROM security. This specification presents the technical details of the device. Appendices detail the differences in the low power MC68HCL05C8A and high speed MC68HSC05C8A versions.

**Order by: HC05C8AGRS/D**

---

## MC68HC05C9A, MC68HCL05C9A, MC68HSC05C9A General Release Specification

**Rev 4.0**

The MC68HC05C9A HCOMS microcomputer is a member of the M68HC05 family. It includes 15,936 bytes of user ROM, 352 bytes of RAM, a serial communications interface, a serial peripheral interface and a 16-bit capture compare times. Eight mask options are available to select external interrupt capability (including an internal pullup) on each of the port B pins. This specification presents the technical details of the device. Appendices detail the differences in the low power MC68HCL05C9A and high speed MC68HSC05C9A versions.

**Order by: HC05C9AGRS/D**

---

## MC68HC05CT4 General Release Specification

**Rev 2.0**

The MC68HC05CT4 is a 44-pin member of the M68HC05 family of microcontrollers and is intended for cordless telephone applications. The memory map includes 5376 bytes of on-chip ROM and 256 bytes of RAM. The MCU has three 8-bit I/O ports, one with pullup options and keyscan capability, and one 7-bit I/O port. Other features include a bird core, bird timer, serial synchronous interface (SSI), 16-bit timer, dual 60MHz PLL, a pulse width modulator and an on-chip COP watchdog circuit. This specification presents the technical details.

**Order by: HC05CT4GRS/D**

---

## MC68HC05E5 General Release Specification

**Rev 1.0**

The 8-bit MC68HC05E5 is a low-cost addition to the M68HC05 Family. The HC05 CPU core has been enhanced with a 15-stage multifunction timer and a programmable PLL. The MCU includes has two 8-bit I/O ports and one 4-bit I/O port, and its 8kbyte of memory includes 384 bytes of RAM and 5120 bytes of user ROM. This specification presents the technical details.

**Order by: HC05E5GRS/D**

## MC68HC(7)05H12 General Release Specification

**Rev 1.0**

The MC68HC(7)05H12 microcomputer is a member of the 8-bit M68HC05 family. It contains an on-chip oscillator, 256 bytes of user RAM, monitor ROM, user ROM or EPROM, parallel I/O, one core timer, two 16-bit programmable timers, COP watchdog system, SCI and SPI, a 4-channel A/D converter and an 8-channel 8-bit PWM for control of H-bridge drivers, with on-chip power driver circuitry. This specification presents the technical details.

**Order by: HC05H12GRS/D**

---

## 68HC05J5A/68HRC05J5A/68HC705J5A/ 68HRC705J5A General Release Specification

**Rev 2.1**

The MC68HC05J5A is a member of the low-cost, high-performance M68HC05 Family of 8-bit MCUs. It is an enhanced version of the MC68HC05J5, with expanded RAM, new ROM sizes, and an additional 16-bit timer with TCAP. It is available in a variety of packages. This specification presents the technical details.

**Order by: HC05J5AGRS/H**

---

## MC68HC05L16/MC68HC705L16 General Release Specification

**Rev 3.0**

The MC68HC05L16 is an 80-pin Quad Flat Pack MCU in the M68HC05 Family, offering sophisticated on-chip peripheral functions. It has five parallel ports, a timebase circuit, 8 and 16-bit timers, COP watchdog timer, LCD drivers and a Simple Serial Peripheral Interface (SSPI). The memory map includes 16k bytes of user ROM and 512 bytes of RAM. This specification presents the technical details.

**Order by: HC05L16GRS/D**

---

## 68HC05L5/68HC705L5 General Release Specification

**Rev 2.0**

The MC68HC05L5 is an 80-pin microcontroller with highly sophisticated on-chip peripherals. It includes five parallel ports, 8 and 16-bit timers, a Computer Operating Properly (COP) watchdog timer, LCD drivers and a Simple Serial Peripheral Interface (SSPI). The memory map includes 8 Kbytes of user ROM and 256 bytes of static RAM. This specification presents the technical details.

**Order by: HC05L5GRS/D**

---

## MC68HC05PL4A, MC68HC05PL4B, MC68HC705PL4B General Release Specification

**Rev 1.1**

The MC68HC05PL4A and MC68HC05PL4B are part of the M68HC05 HCMOS MCU family, and are designed specifically for use as a pair in the handset and base set of cost-sensitive CTO and CT1 analog cordless phones. Features include an industry standard M68HC05 core, built-in low frequency RC oscillator, 256 bytes of user RAM, 4k bytes of user ROM, ROM security, 15 bidirectional I/O lines (23 in MC68HC05PL4B) with keyboard interrupts and high current sink pins, and a multiplexed DTMF output with 6-bit D/A converter. This specification presents the technical details.

**Order by: HC05PL4GRS/H**

---

## MC68HC05RC9/MC68HC05RC18 General Release Specification

**Rev 2.0**

The MC68HC05RC18 MCU is a low-cost, general purpose member of the M68HC05 family that is designed for remote control applications. On-chip peripherals include a Carrier Modulator Transmitter (CMT). There are 20 I/O lines (eight having keyscan logic and pullups) and a low-power reset pin. This specification provides full technical details.

**Order by: HC05RC18GRS/D**

---

## MC68HC08KL8 General Release Specification

**Rev 2.0**

The MC68HC08KL8 is a member of the low-cost M68HC08 family of high-performance 8-bit microcontrollers. It is fully compatible with the Universal Serial Bus (USB) Specification rev. 1.0, with an on-chip USB transceiver and 3.3V regulator, USB data control logic for packet decoding/generation, CRC checking and generation, and NRZI encoding and decoding. Features include 368 bytes of RAM, 8k bytes of on-chip ROM, ROM data security, 39 general purpose I/O pins, an 8-bit keyboard interrupt port and 8 LED direct drive pins. This specification presents the technical details.

**Order by: HC08KL8GRS/D**

---

## MC68HC68VBI General Release Specification

**Rev 3.0**

The Motorola MC68HC68VBI is a low cost HCMOS video peripheral capable of decoding user-definable vertical blanking interval (VBI) data formats from NTSC, PAL or SECAM video signals. A fully duplexed serial peripheral interface (SPI) or Motorola 68HC(7)11 multiplexed expansion bus allows interface with the host processor. Features include data extraction in most formats, specialized PDC mode, internal PLL frequency generator and quasi-horizontal sync detection. This specification presents the technical details.

**Order by: HC68VBIGRS/D**

---

## MC68HC705CT4 General Release Specification

**Rev 2.0**

The MC68HC705CT4 MCU is a 44-pin member of the M68HC05 Family that is intended for use in cordless telephone applications. Features include three 8-bit I/O ports, one with pullup options and keyscan capability, and one 7-bit I/O port; 5136 bytes of user

EPROM, 240 bytes of boot ROM and 256 bytes of RAM; Synchronous Serial I/O (SSI); and dual 60MHz clock. This specification provides the technical details.

**Order by: HC705CT4GRS/D**

## 68HC705JB2 General Release Specification

**Rev 1.1**

The MC68HC705JB2 is a member of the low-cost, high-performance M68HC05 Family of MCUs, and is specifically designed for use in applications requiring a Universal Serial Bus (USB). It features a fully compliant USB with one control endpoint and two interrupt endpoints, in addition to 2048 bytes of user EPROM, 128 bytes of RAM, a Multi-Function Timer, 16-bit Input Capture/Output Compare Timer, and 11 bi-directional I/O pins. This specification presents the technical details.

**Order by: HC705JB2GRS/H**

## MC68HC705MC4 General Release Specification

**Rev 2.0**

The MC68HC705MC4 MCU is a low cost member of the M68HC05 Family that is intended for use in industrial motor control and power supply applications. Features include a 2-channel, 8-bit, high speed PWM module, with a commutation multiplexer for brushless permanent magnet motor control; a 6-input, 8-bit A/D controller; and a serial communications interface. This specification provides the technical details.

**Order by: HC705MC4GRS/D**

## 68HC705RC17 General Release Specification

**Rev 2.0**

The 68HC705RC17 is a general purpose, low-cost addition to the M68HC05 family of MCUs, and is intended for remote control applications. Features include the HC05 CPU core with 14-stage core timer with real time interrupt, COP watchdog system and programmable PLL synthesizer. On-chip peripherals include a carrier modulator transmitter. This specification presents the technical details.

**Order by: HC705RC17GRS/D**

## 68HC708KL8 General Release Specification

**Rev 2.0**

The MC68HC708KL8 is a member of the low-cost, high-performance M68HC08 family of 8-bit MCUs. Features include 8k bytes of EPROM or OTPROM with data security, 39 general purpose I/O lines, a 2-channel 16-bit Timer, a Universal Serial Bus (USB) module and 8-bit Keyboard Interrupt port. This specification presents the technical details.

**Order by: HC708KL8GRS/D**

## MC68HC708MP16 General Release Specification

**Rev 2.0**

This specification presents the technical details of the MC68HC05PL4.

**Order by: HC708MP16GRS/D**

## MC68HC908AT32 General Release Specification

**Rev 2.0**

The MC68HC908AT32 is a member of the low-cost M68HC08 family of high-performance 8-bit microcontrollers. It is designed to emulate two separate automotive MCU families, the MC68HC08AZ32 and the MC68HC08AS20. Features include 32 Kbytes of FLASH ROM with data security, 512 bytes of EEPROM with security option, 1 Kbyte of RAM, SPI and SCI, and system protection features. The two versions include different additional timer and ADC modules, plus a Motorola Scalable CAN Controller or an SAE J1850 Byte Data Link Controller Digital module. This specification presents the technical details and demonstrates the unique qualities of both families.

**Order by: HC908AT32GRS/D**

## 68HC908MR24 General Release Specification

**Rev 1.0**

The 68HC908MR24 is a member of the low-cost, high-performance M68HC08 family of 8-bit MCUs. Features include 8MHz internal bus frequency, 24 Kbytes of FLASH Electrically Erasable ROM with security, on-chip programming firmware for use with host PC, 12-bit 6-channel center or edge-aligned pulse width modulator, Clock Generator Module, SCI, SPI, 16-bit 4-channel and 16-bit 2-channel timer interface modules, and 10-bit 10-channel ADC. This specification presents the technical details.

**Order by: HC908MR24GRS/D**

## LonBuilder User's Guide

This User's Guide teaches developers how to use the LonBuilder Developer's Workbench to develop and test LonWorks applications. It is intended for both hardware and software developers having some programming or basic digital hardware knowledge. It presents a comprehensive overview of the Developer's Workbench and the application development cycle, and explains the use of all the LonBuilder features. Chapters describe how to create, debug and install nodes, and how to monitor and test a development network. Appendices describe the menus, keyboard shortcuts, a sample memory map and the LonBuilder utility programs.

**Order by: LONUG/AD**

## M•CORE Reference Manual

The architecture of the 32-bit M•CORE microRISC engine has been designed for high performance and cost-sensitive embedded control applications, with particular emphasis on reduced system power consumption. M•CORE is a streamlined execution engine providing many of the same performance enhancements as mainstream RISC designs, while lowering the memory bandwidth needed to sustain a high rate of instruction execution. This manual provides an overview

of the processor, and full details of the registers and instruction set, for system software developers and application programmers developing products for M•CORE-based systems.

**Order by: MCORERM/AD**

## M68EM05C0 Emulation Module User's Module

The M68EM05C0 Emulation Module provides the MMDS05 and the MMEVS05/08 development systems with the capability to emulate target systems based on the M68EM05C0 MCUs. This hardware user's manual explains connection, configuration and operation information specific to the module.

**Order by: M68EM05C0UM/D**

## M68EM05E5 Emulation Module User's Manual

The M68EM05E5 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC(7)05JE5 and MC68HC(7)05E1 microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EM05E5UM/D**

## M68EM05JP7 Emulation Module User's Manual

The M68EM05JP7 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC705JP7, MC68HC705JJ7 and MC68HC05JJ6 microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EM05JP7UM/D**

## M68EM05P18 Emulation Module User's Manual

The M68EM05P18 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC805P18 and MC68HC05P18 microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in any one of three development systems – the MMEVS platform board, the MMDS05 modular development system, or the HC05EVS evaluation system.

**Order by: M68EM05P18UM/D**

## M68EM05V12 Emulation Module User's Manual

The M68EM05V12 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC705V12 and MC68HC05V12 microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EM05V12UM/D**

## M68EM05V8 Emulation Module User's Manual

The M68EM05V8 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC705V8 and MC68HC05V7 microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in any one of three development systems – the MMEVS05/MMEVS08 modular evaluation system, the MMDS05 modular development system, or the HC05EVS evaluation system.

**Order by: M68EM05V8UM/D**

## M68EM08AX48 Emulation Module User's Manual

**Rev 1.0**

The M68EM08AX48 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC08AX48 microcontroller. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EM08AX48UM/D**

## M68EM08MP16 Emulation Module User's Manual

**Rev 1.0**

The M68EM08MP16 Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC08MP16 microcontroller. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EM08MP16UM/D**

## M68EML05P6A Emulation Module User's Manual

**Rev 1.0**

The M68EML05P6A Emulation Module allows a Motorola modular development tool to be configured to emulate target systems based on the MC68HC705P6A, MC68HC05P1A, MC68HC05P4A and MC68HC05P9A microcontrollers. This user's manual explains its connection, configuration and operation information. The module can be installed in either of two development systems – the MMEVS platform board or the MMDS0508 modular development system.

**Order by: M68EML05P6AUM/D**

## M68HC05 Applications Guide

**Rev 3**

Assumes no knowledge of microcontrollers and no MCU applications experience. Provides a basic but thorough introduction to the features and operation of microcontrollers, followed by a chapter describing the architecture, addressing modes, instruction set, communications and timer of the MC68HC705C8. The final section traces the development of the hardware and software for a practical application (a home thermostat project) with circuit diagram and full software listing. Full M68HC05 instruction set details are given in an appendix, and the book ends with 50 review questions based on the guide.

**Order by: M68HC05AG/AD**

## HC08 Family Reference Guide

**Rev 1**

A convenient pocket-sized guide providing quick access to essential M68HC08 information such as the Instruction Set, full details of instructions that have been added to the M68HC05 set, Address Mode descriptions, Programming Model, Interrupt Stacking Order and an Opcode Map.

**Order by: M68HC08RG/AD**

## M68HC11 Reference Manual

**Rev 3**

A valuable aid in the development of M68HC11 applications. Detailed descriptions of all internal subsystems have been developed and checked against Motorola internal design documentation, making it perhaps the most comprehensive reference manual available for the M68HC11 family; it complements the data sheet but does not replace it. Practical applications demonstrate the operation of each subsystem; they are treated as complete systems, including hardware/software interactions and trade-offs. Discusses interfacing techniques to prevent component damage, and efficient use of the instruction set.

**Order by: M68HC11RM/AD**

## M6800 Programming Reference Manual

Motorola's M6800 development tools are designed to simplify the development of systems based on the M6800 family of MCUs and peripherals. This manual – first published in 1976 – provides descriptions of the M6800 Program-visible Registers, Interrupts and Stack Operations, Addressing Modes, and Instruction Set.

**Order by: M68PRM/D**

## M6805 HMOS / M146805 CMOS Family User's Manual

**Rev 3**

Provides users with concise information on Motorola's M6805 HMOS and M146805 CMOS microcomputer families. Thorough descriptions and instructions are given, beginning with a general description and introduction to the families, and including details of the hardware and software features illustrated with many 'standard' applications. More advanced applications are covered by reprinted application notes. The manual concludes with detailed definitions of each instruction, arranged in alphanumeric order, with a cycle-by-cycle operation summary.

**Order by: M6805UM/AD3**

## MC6809-MC6809E Microprocessor Programming Manual (1981)

The MC6809 and MC6809E are greatly enhanced, upward-compatible and faster extensions of the MC6800 MPU. This Programming Manual provides details of the additional features, the addressing modes and programming considerations, assuming some familiarity with the MC6800. Detailed information about each instruction is given in an instruction-per-page format, arranged in alphabetical order of mnemonic. The commands and code of the ASSIST09 Monitor Program are also included.

**Order by: M6809PM/AD**

## M68000 Family Programmer's Reference Manual

**Rev 1**

Contains detailed information, in an instruction-per-page format, on each of the instructions used by the MPUs and coprocessors in the M68000 family. Includes MPUs from the MC68000 to the MC68040, the MC68851 PMMU, the MC68881 and MC68882 Floating-Point Coprocessors, and the CPU32 processor core used in the M68300 family. The manual is divided into Integer Instructions, Floating-Point Instructions, Supervisor (Privileged) Instructions, and CPU32 Instructions and Addressing Modes. A Format Summary lists all the instructions in binary format, and a processor/instruction cross reference is included.

**Order by: M68000PM/AD**

## M68000 8-/16-/32-bit Microprocessors User's Manual, Ninth Edition

**Rev 8**

Provides hardware details and programming information for the MC68000, MC68008, MC68010 and MC68HC000 microprocessors. The MC68008 has an 8-bit data bus and smaller addressing range; the MC68010 introduced virtual memory to the family and has a few different instructions; the MC68HC000 uses about 10% of the power of the MC68000; otherwise the devices are very similar. The manual fully describes their electrical and operating characteristics, noting any differences. Includes detailed information about each instruction, arranged in alphabetical order of mnemonic.

**Order by: M68000UM/AD**

## MC68020/MC68EC020 Microprocessors User's Manual

**Rev 2**

The MC68020 was the first full 32-bit implementation of Motorola's M68000 family. It is joined by the MC68EC020, an economical version designed for embedded controller (EC) applications. This User's Manual describes the capabilities, operation and programming

of the two devices, highlighting differences where applicable. An introduction provides an overview of the devices and their instruction sets. Other sections include Processing States, Signal Description, On-Chip Cache, Bus Operation, Exception Processing, Coprocessor Interface, Instruction Timing, Applications Information, and electrical and mechanical data.

**Order by: M68020UM/AD**

## MC68040, MC68040V, MC68LC040, MC68EC040, MC68EC040V Microprocessors User's Manual

**Rev 1**

The MC68040, MC68040V, MC68LC040, MC68EC040 and MC68EC040V are third-generation, 32-bit MPUs in the M68000 family. They use multiple concurrent execution units and a highly integrated architecture to achieve very high performance. This manual describes the capabilities, operation and programming of the five devices. Sections include Integer Unit, Memory Management, On-Chip Caches, Signal Description, IEEE 1149.1 Test Access Port (JTAG), Bus Operation, Exception Processing, Floating-Point Unit and Instruction Timing.

**Order by: M68040UM/AD**

## MC68060, MC68LC060, MC68EC060 Microprocessors User's Manual

**Rev 1**

The MC68060, MC68LC060 and MC68EC060 are the first processors in the M68060 product line. All offer superscalar integer performance of more than 100 MIPS at 66MHz while maintaining compatibility with the rest of the M68000 Family. This manual describes their capabilities, operation and programming. Sections include a general introduction, Signal Description, Integer Unit, Memory Management, the Caches, Floating Point Unit, Bus Operation, Exception Processing, JTAG and Debug Pipe Control Modes, Instruction Timing, Applications, and Electrical and Thermal Characteristics.

**Order by: M68060UM/AD**

## MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual (Supplement to MC68302UM/AD)

The MC68EN302 is a multiprotocol integrated communications controller based on the MC68302. It adds an Ethernet controller which is independent of the three on-chip serial channels, plus a DRAM control and a JTAG interface. This manual describes aspects of the programming, capabilities, registers and operation of the MC68EN302 where they differ from the MC68302. Separate chapters describe the Module Bus Controller, DRAM Control Module (DCM), Ethernet Controller and JTAG Test Access Port.

**Order by: MC68EN302RM/AD**

## MC68EZ328 DragonBall-EZ Integrated Processor User's Manual

**Rev 1**

The MC68EZ328 (DragonBall-EZ) microprocessor – the second generation DragonBall – is designed to save time, power, cost, board space, pin count and programming steps when designing a product. Its functionality might require 20 separate components in another system. The MC68EZ328 combines an MC68EC000 processor with intelligent peripheral modules and typical system interface logic – all are optimally connected, timed with the same clock, fully tested and uniformly documented. This manual discusses the details of how to initialize, configure and program the MC68EZ328 microprocessor; it assumes basic knowledge of 68K architecture.

**Order by: MC68EZ328UM/D**

## MC68F333 User's Manual

The MC68F333 is a highly integrated 32-bit microcontroller which includes a Single Chip Integration Module, an 8-channel 10-bit ADC, a Time Processor Unit, a 512-byte Standby RAM, 3.5 Kbyte RAM with TPU emulation, and two flash EEPROM modules. This user's manual describes all the modules in detail, and includes electrical and timing information. Address maps and register diagrams are summarized in an appendix for convenience.

**Order by: MC68F333UM/AD**

## MC68HC05Cx HCMOS Single-Chip Microcontrollers Programming Reference Guide

**Rev 1**

A convenient pocket-sized guide providing quick access to essential MC68HC05C-series information such as Block Diagrams, Memory Maps, the Programming Model, Registers and Control Bits, Instructions, Addressing Modes, Execution Times and Pin Assignments.

**Order by: MC68HC05CxRG/AD**

## MC68HC11A8 Programming Reference Guide

**Rev 1**

A convenient pocket-sized guide providing quick access to essential MC68HC11A8 information such as the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignments.

**Order by: MC68HC11A8RG/AD**

## MC68HC11C0 Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential MC68HC11C0 information such as a Block Diagram, the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignment.

**Order by: MC68HC11C0RG/AD**

## MC68HC11D3/MC68HC711D3 Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential MC68HC11D3 and MC68HC711D3 information such as the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignment.

**Order by: MC68HC11D3RG/AD**

## MC68HC11E Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential information for the MC68HC11E series of MCUs, including the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignments. The guide covers the MC68HC11E0, 'E1, 'E8, 'E9 and 'E20, the MC68HC711E9 and 'E20, the MC68S711E9 and the MC68HC811E2.

**Order by: MC68HC11ERG/AD**

## MC68HC11F1 Programming Reference Guide

**Rev 2**

A convenient pocket-sized guide providing quick access to essential MC68HC11F1 information such as the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignments.

**Order by: MC68HC11F1RG/AD**

## MC68HC11K4/MC68HC711K4 Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential information on the MC68HC11K4 MCU, and on the MC68HC711K4 EPROM version. It includes the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignments.

**Order by: MC68HC11K4RG/AD**

## MC68HCL6/MC68HC711L6 Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential MC68HC11L6 and MC68HC711L6 information such as the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instructions, Addressing Modes, Execution Times, Registers and Control Bits, and Pin Assignment.

**Order by: MC68HC11L6RG/AD**

## M68HC11 M Series Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential programming information for Motorola's M68HC11 M-series MCUs, including the Programming Model, Crystal Dependent Timing, Interrupt Vectors, Memory Map, Opcode Maps, Instruction Set, Addressing Modes, Execution Times, Special Operations, Registers and Control Bits, and Pin Assignments.

**Order by: MC68HC11MRG/AD**

## MC68HC11N Series Programming Reference Guide

A convenient pocket-sized guide providing quick access to essential M68HC11 N-series information such as the Programming Model, Interrupt Data, Memory Map, Opcode Maps, Instruction Set, Addressing Modes and Timing Information, Registers and Control Data.

**Order by: MC68HC11NRG/AD**

## MC68HC16Y1 User's Manual

The MC68HC16Y1 is a high-speed 16-bit MCU in the M68HC16 family. It incorporates a true 16-bit CPU, single-chip integration module (SCIM), an 8/10-bit ADC, multi-channel communication interface (MCCI), general purpose timer (GPT), a 2 kByte standby RAM module with TPU emulation capability (TPURAM) and a 48K masked ROM. These modules are interconnected by an intermodule bus (IMB). This manual includes comprehensive information on all these modules, with timing diagrams and an instruction set summary. Appendices cover electrical and mechanical characteristics, a comprehensive register summary and development support.

**Order by: MC68HC16Y1UM/AD**

## M68HC16 Z Series User's Manual

The M68HC16 Z-series microcontrollers are high-speed 16-bit devices in the M68HC16 Family, and are upward compatible with M68HC11 devices. They are built from standard modules, interfacing via a common internal bus, to facilitate rapid development of devices for specific applications. All contain an Analog-to-Digital Converter (ADC) and General Purpose Timer, plus either a Queued Serial Module (QSM) or Multichannel Communications Interface (MCCI). This manual provides a detailed overview of all the devices in the Z-series with design information for each module. Includes a comprehensive Register Summary and Programming Examples.

**Order by: MC68HC16ZUM/AD**

## MC68HC681 Dual Asynchronous Receiver/ Transmitter (DUART)

The MC68HC681 Dual Asynchronous Receiver/Transmitter (DUART) is part of the M68000 Family of peripheral devices, and interfaces directly to the MC68000 processor via an asynchronous bus structure. This user's manual describes the operation, programming and registers

of the DUART, and provides electrical and timing data. An appendix details the differences between the MC68HC681 and the MC68HC2681, which is functionally equivalent.

**Order by: MC68HC681UM/AD**

## MC68HC901 Multi-Function Peripheral User's Manual

The MC68HC901 Multi-Function Peripheral (MFP) is a member of the M68000 Family, and interfaces directly to the MC68000 through the asynchronous bus structure. Both vectored and polled interrupt schemes are supported, with the MFP providing unique vector number generation for each of 16 interrupt sources. Handshake lines are provided to allow DMA Controller interfacing. This User's Manual describes the operation of the MFP, including signal description, bus operation, interrupt structure, I/O port, timers, USART, and electrical and mechanical characteristics.

**Order by: MC68HC901UM/AD**

## MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual

The MC68LC302 is a low power version of the MC68302 Integrated Multiprotocol Processor (IMP). In simple terms it is the same device, but minus the third Serial Communications Controller (SCC3), and with a new static 68000 core, new timer and low power modes. It is packaged in a low profile package for reduced board space and makes it suitable for use in applications such as PCMCIA. This manual describes all the differences between the MC68LC302 and the MC68302, full details of which are contained in the MC68302 User's Manual, reference MC68302UM/AD.

**Order by: MC68LC302RM/AD**

## Integrated Multiprotocol Processor with PCMCIA Interface Reference Manual

The MC68PM302 is a derivative of the MC68302 Integrated Multiprotocol Processor (IMP). It can operate in two modes – in one mode it functions as an enhanced MC68302 with a new static 68000 core, new timer and low power modes, and additional parallel I/O pins; in the second mode it offers the same enhanced capability, but with PCMCIA and 16550 UART functionality instead of the additional I/O pins. It is packaged in a low profile package suitable for use in Type II PCMCIA cards. This manual describes all the differences between the MC68PM302 and the MC68302, full details of which are contained in the MC68302 User's Manual, reference MC68302UM/AD.

**Order by: MC68PM302RM/AD**

## MC68SC302 Passive ISDN Protocol Engine User's Manual

The MC68SC302 Passive ISDN Protocol Engine (PIPE) is an ISA 'Plug and Play'/PC card ISDN communication controller optimized for ISDN passive cards. It has been developed from the popular MC68302 Integrated Multiprotocol Processor and features glueless connection to Motorola's MC145572 and MC145574 transceivers. The three serial communication channels have been optimized to support two 64kbit per second B-channels and one 16kbit per second D-channel. This manual describes the programming, capabilities, registers and operation of the MC68SC302, including the Interrupts and Timer, Communications Processor, 'Plug and Play' Interface and PCMCIA Interface.

**Order by: MC68SC302UM/AD**

## MC68030 Enhanced 32-bit MPU User's Manual, third edition

**Rev 2**

The MC68030 is a second-generation 32-bit MPU in Motorola's M68000 family. It combines a CPU core, instruction and data caches, bus controller and memory management unit in a single VLSI device. This manual describes its capabilities, operation and programming. Sections include Data Organisation and Addressing, Instruction Set, Processing States, Signal Description, On-Chip Caches, Bus Operation, Exception Processing, Memory Management Unit, Coprocessor Interface, Instruction Timing, Applications Information, Electrical Specifications and Mechanical Data.

*ISBN 0 13 566969 3*

**Order by: MC68030UM/AD**

## MC68302 Integrated Multiprotocol Processor User's Manual

**Rev 3**

The MC68302 IMP is a VLSI device incorporating the main building blocks needed to design a wide variety of powerful communications controllers. It may be configured to support 5 different protocols, any 3 operating simultaneously. This manual describes its architecture; the MC68000 processor core on which it is based; the System Integration Block which provides basic timing and interfacing functions required by virtually every application; the Communications Processor which includes 3 independent serial channels with 6 DMA controllers; plus Signal Descriptions and Electrical Characteristics.

**Order by: MC68302UM/AD**

## MC68306 Integrated EC000 Processor User's Manual

The MC68306 is an integrated processor containing an MC68EC000 processor and elements required in many MC68000 and MC68EC000-based systems, reducing design time especially in systems using serial interfaces and Dynamic RAM. This user's manual introduces the core and the on-chip peripherals, describes the signals and 68000 bus operation, provides detailed information about the core and Serial Module, and discusses the IEEE 1149.1 Test Access Port.

**Order by: MC68306UM/AD**

## Bandit: MC68322 Integrated Printer Processor User's Manual

**Rev 1**

The MC68322 is a high-performance integrated printer processor combining an MC68000 compatible core processor, a RISC graphics processor, a print engine video controller and system integration

features on a single chip. Specialised display list banding techniques performed by the graphics processor allow system memory requirements to be reduced significantly. This manual includes sections on the Core; Bus Operation; Interrupts; System Integration Module; DRAM Controller; DMA, Parallel Port and Print Engine Interfaces; RISC Graphics Processor; Graphic Operations and Orders; and electrical and mechanical information.

**Order by: MC68322UM/AD**

## MC68328 (DragonBall) Integrated Processor User's Manual

As the consumer market for portable devices expands, system requirements become more demanding. Fewer components, smaller board space, lower power consumption and lower system cost are major criteria. Motorola has introduced the MC68328 DragonBall integrated portable system processor to address these needs. It provides key features for portable systems, such as a real-time clock, LCD oscillator, pulse-width modulator, timers, SPI and the SIM28 system integration module. This User's Manual describes the capability, operation and programming of the MC68328.

**Order by: MC68328UM/AD**

## MC68330 Integrated CPU32 Processor Users Manual

The MC68330 is a 32-bit integrated processor linking high-performance data manipulation capability with circuits typically required in embedded controller applications. It combines the CPU32 core processor and the SIM40 system integration module. This User's Manual describes the programming, capabilities, registers and operation of the MC68330. Sections provide signal descriptions, full details of bus operation, and explain the use of the CPU32 and SIM40. The Guide also covers use of the IEEE 1149.1 Test Access Port, and gives applications guidelines.

**Order by: MC68330UM/AD**

## MC68332 User's Manual

**Rev 1**

The MC68332 is a 32-bit integrated microcontroller in the M68300 Family, combining high-performance data manipulation capabilities with powerful peripheral subsystems. This manual includes sections describing the input and output signals; timing, exception processing and arbitration for the external bus; the submodules of the System Integration Module (SIM); the Queued Serial Module; operation of the 2K Standby RAM; plus overviews of the MC68020-based CPU32 processor, the Time Processor Unit (TPU) and available emulation systems. It includes electrical and mechanical data.

**Order by: MC68332UM/AD**

## MC68340 Integrated Processor User's Manual

**Rev 1**

The MC68340 is a 32-bit integrated processor in the M68300 Family, combining high-performance data manipulation capabilities with powerful peripheral subsystems. This manual includes sections describing the input and output signals; timing, exceptions and arbitration for the external bus; the submodules of the System Integration Module (SIM); the MC68020-based CPU32 processor; the high-performance DMA Controller module; the serial communications module; the twin timer modules; and the IEEE 1149.1-standard test port. It includes applications guidelines and electrical and mechanical data.

**Order by: MC68340UM/AD**

## MC68356 Signal Processing Communications Engine User's Manual

The MC68356 is the first commercially available monolithic device to include a general purpose digital signal processor, a CISC microprocessor and a RISC microprocessor on a single chip. The features of its multiprotocol communications processor are a subset of the MC68302, the DSP is DSP56002-based, and its PCMCIA slave interface emulates the UART16550. This manual describes its architecture and external signals, and includes sections on Clock Generation and Low Power Control; the 68000 Core, Memory Map and SIB; the Communications Processor; the PCMCIA Controller; DSP Ports and Memory; and the IEEE 1149.1 Test Access Port.

**Order by: MC68356UM/AD**

## MC68360 Quad Integrated Communications Controller User's Manual

**Rev 1**

The MC68360 Quad Integrated Communication Controller (QUICC) is a development of the MC68302, but with higher performance, increased flexibility and major extensions to capability. It incorporates four Serial Communications Controllers (SCC), two serial Management Controllers (SMC) and a Serial Peripheral Interface (SPI). This manual provides full details concerning the use and operation of the QUICC, including signal descriptions, memory map, bus operation, an overview of the CPU32+, System Integration Module (SIM60), Communication Processor Module (CPM), Test Access Port and electrical characteristics. Includes a section discussing practical applications.

**Order by: MC68360UM/AD**

## MC68605 X.25 Protocol Controller User's Manual

The MC68605 X.25 Protocol Controller (XPC) is an intelligent HCMOS communications protocol controller that implements the 1984 CCITT X.25 Recommendation, data link access procedure (LAPB). This manual provides full user information including operating modes, a description of the internal registers and the shared memory structures that provide communication with the host processor, details of the command set and the external signals, and the operation of the bus. Timing and state diagrams are given on foldout sheets for ease of reference.

**Order by: MC68605UM/AD**

## MC68824 Token Bus Products User's Manual

**Rev 1**

The MC68824 Token Bus Controller (TBC) was the first single-chip device to implement the IEEE 802.4 Media Access Control (MAC) sublayer of the Manufacturing Automation Protocol (MAP). It operates as an intelligent peripheral that relieves its host microprocessor of the frame formatting and token management functions, using on-chip DMA to transfer data frames to and from memory. This manual is a detailed functional and electrical description of the device, including programming information and an overview of IEEE 802.4.

**Order by: MC68824UM/AD**

## MC68836 FDDI User's Manual

The Fiber Distributed Data Interface is a Local Area Network (LAN) under the auspices of ANSI. It supports a 100mBits/sec token ring with up to 1000 stations. The MC68836 FDDI Clock Generator implements the lower portion of the physical layer functions of the standard including Clock Recovery, Data Recovery, NRZI Conversions and 5-bit parallel-to-serial/serial-to-parallel conversions. This User's Manual describes its operation, signals, timing and applications.

**Order by: MC68836UM/AD**

## MC68837 FDDI User's Manual

The Fiber Distributed Data Interface is a Local Area Network (LAN) under the auspices of ANSI. It supports a 100mBits/sec token ring with up to 1000 stations. The MC68837 Elasticity Buffer and Link Management (ELM) chip implements the physical layer (PHY) functions of the standard including data framing, elasticity buffer, encoding, decoding, smoothing, line state detection and repeat filter. It also contains a number of station management functions. This User's Manual describes its operation, registers, signals and timing.

**Order by: MC68837UM/AD**

## MC68838 FDDI User's Manual

The Fiber Distributed Data Interface (FDDI) is a 125Mbit/sec, fibre-optic based token ring designed to accommodate rings up to 1000 stations, with 2km between stations and 200km total ring length. The ANSI standard for FDDI networks defines a number of protocols including the data link Media Access Control (MAC) layer. Motorola's MC68838 chip implements this protocol. This manual provides an overview and functional description of the device, with details of the 36 control/status registers, signal descriptions, bus and MAC-PHY operation, and transmit and receive data path operation.

**Order by: MC68838UM/AD**

## MC68839 FDDI System Interface User's Manual

The Fiber Distributed Data Interface is a Local Area Network (LAN) under the auspices of ANSI. It supports a 100mBits/sec token ring with up to 1000 stations. Motorola's FDDI chipset consists of an FDDI Clock Generator, an Elasticity and Link Management physical layer circuit, a Media Access Control circuit, and an FDDI System Interface (FSI). This manual describes the FSI. Sections include Functional Block Description; Registers; Signal Descriptions; Commands and Indications; Functional Operation; Initialisation, Programming and Examples; Port Operation; Boundary Scan Details; Electrical Specifications and Mechanical Data. System performance requirements are discussed in an appendix.

**Order by: MC68839UM/AD**

## MC68840 Integrated Fiber Distributed Data Interface User's Manual

**Rev 1**

FDDI is a fibre-optic-based, token ring local area network standard developed to accommodate rings of up to 1000 stations and a total ring length of 200km, operating at speeds up to 100Mbps. This ANSI standard specifies the Media Access Control (MAC) layer, the Physical (PHY) layer, the Physical Medium Dependent function and the Station Management function. The MC68840 implements the MAC and PHY layers. This manual provides an overview of the device, plus full descriptions of the functional blocks, registers, ports, external signals and test operations. Includes two practical examples to illustrate the design process.

**Order by: MC68840UM/AD**

## MC68847 Quad ELM FDDI User's Manual

The MC68847 Quad ELM implements four MC68837 ELM (Elasticity Buffer and Link Management) devices on a single chip, providing a low cost solution for concentrator applications. Each implements the physical layer (PHY) functions of the FDDI standard including data framing, elasticity buffer, encoding, decoding, smoothing, line state detection and repeat filter. This User's Manual describes its operation, registers, signals and timing.

**Order by: MC68847UM/AD**

## MC88200 Cache/Memory Management Unit User's Manual

**Rev 1**

The MC88200 CMMU is a high-performance, HCMOS VLSI device combining demand-paged virtual memory with 16K bytes of on-chip cache memory. It is specifically designed to operate with the MC88100 RISC processor. Separate chapters provide full details of the memory management functions and cache operation, and are followed by descriptions of the signals, bus operation, timing and registers. 48 pages of applications information discuss the use of multiple MC88200s, memory bus connections, and power and ground considerations. Contains electrical characteristics and mechanical data.

**Order by: MC88200UM/AD**

## MC88410 Secondary Cache Controller User's Manual

The MC88410 is a highly integrated secondary cache controller in the M88110 family that reduces both memory latency and system bus use, while extending multiprocessing capabilities to achieve a higher level of system performance. This User's Manual gives an overview of the MC88110/MC88410 system and the benefits of

using the secondary cache, describes the MC88410 operation and its signals in detail, and provides functional descriptions of the processor and system bus interfaces. It includes a chapter on diagnostics and JTAG.

**Order by: MC88410UM/AD**

## ATM Cell Processor Design Reference Manual

Each switching system in an Asynchronous Transfer Mode (ATM) network handles multiple physical links, and transfers each arriving ATM cell between its source and destination links using prearranged routing. ATM standards divide the tasks on either side of the switch into PHY-layer (physical layer) tasks and ATM-layer tasks. The MC92500 is a cell processing device which provides ATM-layer cell processing and routing functions between a PHY-layer device and an ATM switch fabric. This reference manual provides design information for the MC92500, including a Functional Description; Register, External Memory and Signal Descriptions, Data Structures, Ingress and Egress Data Path Operation, System Operation, Support for Operations and Maintenance, interface descriptions and more.

**Order by: MC92500UM/D**

## MC92501 ATM Cell Processor User's Manual

**Rev 1**

An ATM network is composed mainly of switching elements, each handling multiple physical links. A typical core switch consists of a switch matrix and some line cards, one card for each physical link or group of links. At the edges of the network, access multiplexers route a single link to multiple links. Motorola's MC92501 ATM Cell Processor can be used to provide ATM-layer cell processing and routing functions in both the line cards and in the access multiplexer. This Users' Manual provides detailed information on the operation and use of the MC92501.

**Order by: MC92501UM/D**

## Multichannel Communication Interface Reference Manual

**Rev 1**

This manual describes the capabilities, operation and functions of the Multichannel Communication Interface (MCCI), an integral module in Motorola's family of modular microcontrollers. The MCCI contains a Serial Peripheral Interface (SPI) and two Serial Communication Interfaces (SCI). Sections include an Overview of the module, Signal Descriptions, Configuration and Control Registers, and separate chapters describing the SCI and SPI submodules.

**Order by: MCCIRM/AD**

## MCF5102 ColdFire User's Manual

**Rev 1**

ColdFire is a microprocessor architecture optimized for embedded processing. It combines the architectural simplicity of 32-bit fixed length RISC with a memory-saving variable length instruction set – its higher code density requires less program memory than for fixed length systems and allows the use of lower cost memory for given performance. The MCF5102 if the first chip in the family, and includes the capability to execute existing 68000 code to provide an upgrade bridge. This User's Manual describes the capabilities, operation and programming of the MCF5102. Instruction timing is provided, but full details of the instruction set are given in the M68000 Family Programmer's Reference Manual, M68000PM/AD.

**Order by: MCF5102UM/AD**

## MCF5200 ColdFire Microprocessor Family Programmer's Reference Manual

**Rev 1.0**

This manual contains information about the software instructions used by the ColdFire 5200 microprocessors. It includes sections on the addressing capabilities, exception processing, timing, and on the instructions themselves in both summary and alphanumeric page-per-instruction format.

**Order by: MCF5200PRM/AD**

## ColdFire MCF5202 User's Manual

ColdFire is a revolutionary microprocessor architecture that is optimized for embedded processing applications, bringing new levels of price and performance to cost-sensitive high-volume products. Based on the concept of variable-length RISC technology, ColdFire combines the architectural simplicity of conventional 32-bit RISC with a memory-saving, variable length instruction set. This manual describes the programming, capabilities and operation of the MCF5202 processor. Topics include signal descriptions, details of the core and cache, bus operations, debug support, JTAG specification, and an overview of the issues involved in porting embedded development tools from M68000 architecture.

**Order by: MCF5202UM/AD**

## MCF5204 ColdFire Integrated Microprocessor User's Manual

The MCF5204 integrated microprocessor combines a ColdFire processor core with peripheral functions such as a timer, serial interface and System Integration Module (SIM) to provide low-cost, enhanced system performance for embedded control applications. ColdFire combines the architectural simplicity of conventional 32-bit RISC with a simplified version of the variable length M68000 instruction set. This user's manual describes the programming, capabilities and operation of the MCF5204; reference should also be made to the MCF5200 ColdFire Family Programmer's Reference Manual (MCF5200PRM/AD).

**Order by: MCF5204UM/AD**

## MCF5206e ColdFire Integrated Microprocessor User's Manual

The MCF5206e integrated microprocessor combines a Version 2 ColdFire processor core with peripheral functions such as a DRAM controller, timers, general purpose I/O and serial interfaces, debug module and system integration to provide low-cost, enhanced system performance for embedded control applications. It is an enhanced version of the MCF5206 processor, with the same peripheral set, DMA, MAC, Hardware Divide, a larger cache and larger SRAM.

This user's manual describes the programming, capabilities and operation of the MCF5206e; reference should also be made to the MCF5200 ColdFire Family Programmer's Reference Manual (MCF5200PRM/AD).

**Order by: MCF5206eUM/D**

## MCF5206 ColdFire Integrated Microprocessor User's Manual

The MCF5206 integrated microprocessor combines a ColdFire processor core with peripheral functions such as a DRAM controller, timers, general purpose I/O and serial interfaces, and system integration to provide low-cost, enhanced system performance for embedded control applications. ColdFire combines the architectural simplicity of conventional 32-bit RISC with a simplified version of the variable length M68000 instruction set. This user's manual describes the programming, capabilities and operation of the MCF5206; reference should also be made to the MCF5200 ColdFire Family Programmer's Reference Manual (MCF5200PRM/AD).

**Order by: MCF5206UM/AD**

## MCF5307 ColdFire Integrated Microprocessor User's Manual

The ColdFire processor core is designed for embedded control applications. Its architecture uses variable-length RISC instruction set technology to give new levels of price and performance to cost-sensitive, high-volume markets; denser binary code requires less memory for a given application. The MCF5307 integrated microprocessor combines a ColdFire core with a Multiply-Accumulate (MAC) unit, DRAM controller, timers, parallel and serial interfaces, and system integration. These on-chip functions greatly reduce the time required for typical system design and implementation. This User's Manual describes the programming, capabilities and operation of the MCF5307.

**Order by: MCF5307UM/AD**

## Motorola Microcontroller Development Tools Directory

**Rev 5**

A directory of hardware and software development tools – from Motorola and from third party vendors – for the M68HC05, M68HC08, M68HC11, M68HC16, M68300 and MPC500 microcontroller families. Includes a cross reference listing products under Adapters, Emulators, Evaluation Boards, Logic Analyzers, Programmers, Other Hardware Tools, Assemblers, Compilers, Debuggers, Integrated Development Environments, Real-Time Operating Systems, Simulators and Other Software Tools.

**Order by: MCUDEVTLDIR/D**

## M•CORE MMC2001 Reference Manual

The 32-bit M•CORE microRISC engine represents a new family of microprocessor core products. It provides many of the same performance enhancements as mainstream RISC designs, but the processor architecture has been designed for high-performance and cost-sensitive embedded control applications, with particular emphasis on reduced system power consumption. This reference manual describes the CPU, memory map, signals, ROM module, Static RAM module, External Interface Module, Clock Module and low power modes, Timer/Reset Module, Interrupt Controller, UART, SPI, Keypad Port, PWM and OnCE Debug Module. A Programming Reference is provided in an appendix.

**Order by: MMC2001RM/D**

## PowerPC Microprocessor Family: The Bus Interface for 32-bit Microprocessors

The main purpose of this manual is to provide a detailed functional description of the 60x bus interface, the communication channel for the first generation of PowerPC microprocessors, as implemented on the PowerPC 601, 603 and 604 microprocessors. It is intended to help system and chip set developers by being a central reference source for the interface presented by these processors, describing both the basic signals that are common to all the processors and the signals that are not common but which can maximize the performance of a system implementation.

**Order by: MPCBUSIF/AD**

## PowerPC Microprocessor Family: The Programmer's Reference Guide

The main purpose of this guide is to provide a concise method for system developers and application programmers to implement software that is compatible across the PowerPC family of processors and other devices. A Register Summary gives a brief overview of the PowerPC register set, including a programming model and quick reference guide for 32-bit and 64-bit registers. The Memory Control Model outlines the page table entry and segment table entry. Exception Vectors is a quick reference for exception types and the conditions that cause them. And PowerPC Instruction Set gives detailed information on the entire instruction set.

**Order by: MPCPRG/D**

## PowerPC PCI Bridge/Memory Controller User's Manual

The MPC105 PCI bridge/memory controller provides a PowerPC reference platform-compliant bridge between the PowerPC microprocessor family and the peripheral component interconnect (PCI) bus. PCI support allows system designers to rapidly design systems using peripherals already designed for PCI and the other standard interfaces available in the personal computer hardware environment.

**Order by: MPC105UM/AD**

## MPC509 RISC Microprocessor User's Manual

The MPC509 is 32-bit implementation of the PowerPC architecture, providing 32-bit effective addresses, integer data types of 8, 16 and 32 bits, and floating point data types of 32 and 64 bits. It has four execution units that can operate in parallel, and is capable of issuing one sequential instruction per clock. Branch instructions are evaluated ahead of time when possible, resulting in zero-cycle

execution time in many cases. This user's manual describes the functionality of the MPC509 for software and hardware developers who wish to develop products that use it.

**Order by: MPC509UM/AD**

## PowerPC 603e RISC Microprocessor User's Manual

**Rev 1**

The MPC603e is built on the low power, low cost and high performance attributes of the MPC603, while providing enhanced capabilities through higher clock speed, greater system clock flexibility, increases in cache size and set-associativity. Although this manual is concerned with the 603e, all the information applies to both devices except where noted in an appendix. It includes detailed chapters on the Programming Model, Instruction and Data Cache Operation, Exception Processing, Memory Management, Instruction Timing, Signal Descriptions, System Interface Operation, and Power Management. Appendices include an instruction set listing and details of 603 differences.

**Order by: MPC603eUM/AD**

## PowerPC 604e RISC Microprocessor User's Manual

The PowerPC 604e is an implementation of the PowerPC family of RISC microprocessors. It has seven execution units that can operate in parallel, and with uniform instructions allowing rapid execution times the design yields high efficiency and throughput. It has separate memory management units and on-chip caches for instructions and data, and a host of features to further enhance performance. This user's manual describes the functionality of the 604e for software and hardware developers who wish to develop products that use it.

**Order by: MPC604EUM/AD**

## PowerPC 620 RISC Microprocessor User's Manual

The PowerPC 620 is 64-bit implementation of the PowerPC architecture, providing 64-bit effective addresses, integer data types of 8, 16, 32 and 64 bits, and floating point data types of 32 and 64 bits. It has six execution units that can operate in parallel, and with uniform instructions allowing rapid execution times the design yields high efficiency and throughput. It has separate memory management units and separate 32 Kbyte on-chip caches for instructions and data, a 40-bit address bus, and a host of features to further enhance performance. This user's manual describes the functionality of the 620 for software and hardware developers who wish to develop products that use it.

**Order by: MPC620UM/AD**

## MPC750 RISC Microprocessor User's Manual

The MPC750 is an implementation of the 32-bit portion of the PowerPC microprocessor architecture, to provide 32-bit effective addresses, 8/16/32-bit integer data types and 32/64-bit floating-point data types. It is a superscalar processor capable of completing two instructions simultaneously. This manual defines the functionality of the MPC750 and MPC740 microprocessors. It is intended for use by system and application hardware and software developers with an understanding of operating systems, MPU system design, basic principles of RISC processing and details of the PowerPC architecture.

**Order by: MPC750UM/AD**

## MPC801 Integrated Microprocessor for Embedded Systems User's Manual

The MPC801 PowerPC Quad Integrated Communications Controller (PowerQUICC) is a versatile one-chip integrated microprocessor and peripheral combination that can be used in a variety of controller applications. It is a low-cost version of the MPC860 that provides an effective price/performance solution across a wide range of applications. Like the MPC860 it combines a high-performance PowerPC core with a multifaceted system integration package. This user's manual describes the operation of all of the MPC801 functionality, with emphasis on the I/O functions.

**Order by: MPC801UM/AD**

## MPC821 PowerPC Portable Systems Microprocessor User's Manual

The MPC821 PowerPC Portable Systems Microprocessor is a versatile one-chip integrated microprocessor and peripheral device that can be used in a variety of controller applications. It is a PowerPC derivative of the MC68360 QUICC, and is intended particularly for use in high performance and portable communications systems where lower power consumption is essential. This comprehensive manual describes the operation of the MPC821, with particular emphasis on the I/O functions and the Communication Processor Module.

**Order by: MPC821UM/AD**

## PowerPC MPC823 Pocket Guide

This convenient pocket guide contains design guidelines, the memory map, list of registers, instructions and list of external signals for the MPC823 microprocessor. The lists include cross references to the MPC823 User's Manual.

**Order by: MPC823RG/D**

## PowerPC MPC823 User's Manual

The MPC823 PowerPC microprocessor is a versatile, one-chip integrated microprocessor and peripheral combination that can be used in a variety of portable electronic products; it excels in low-power image capture and personal communication products. It is essentially a low cost version of the MPC821, enhanced with additional communication and display capabilities. These additional features are provided by a specialized RISC processor that can perform signal processing functions for image compression and decompression, and which supports six serial channels. This substantial User's Manual discusses the operation, possible configurations, and specifications of the MPC823.

**Order by: MPC823UM/D**

## MPC850 Integrated Communications Microprocessor User's Manual

The MPC850 is a versatile, one-chip integrated microprocessor and peripheral combination that can be used in a variety of controller applications, excelling particularly in communications and networking products. It includes support for Ethernet, and is specifically designed for cost-sensitive, remote access and telecommunications applications. The purpose of this user's manual is to help communications systems designers build systems using the MPC850, and to help software designers provide operating systems and user-level applications that take best advantage of its features.

**Order by: MPC850UM/D**

---

## MPC860 PowerQUICC User's Manual

**Rev 1**

The MPC860 PowerPC Quad Integrated Communications Controller (PowerQUICC) is a versatile, one-chip integrated microprocessor and peripheral device that can be used in a variety of controller applications. It is a PowerPC derivative of the MC68360, and is intended particularly for use in both communications and networking systems. This comprehensive manual describes the operation of the MPC860, with particular emphasis on the I/O functions and the Communication Processor Module. An appendix discusses the movement of applications from the MC68360 QUICC environment to the MPC860 PowerQUICC environment.

**Order by: MPC860UM/AD**

---

## MPC8240 Integrated Processor User's Manual

The MPC8240 is a cost-effective, general purpose integrated processor designed for applications using PCI in networking infrastructure, telecommunications and other embedded markets.It can also be used for control processing in applications such as network routers and switches, mass storage subsystems, network appliances, and print and imaging systems. It is based on a 32-bit PowerPC 603e processor core, and performs many peripheral functions on-chip. This user's manual describes the functionality of the MPC8240 for system software and hardware developers wishing to design products based on the device.

**Order by: MPC8240UM/D**

---

## MPC8260 PowerQUICC II User's Manual

The MPC8260 PowerQUICC II is a versatile communications processor that integrates on one chip a high-performance PowerPC microprocessor – an embedded variant of the MPC603e – plus a flexible system integration unit and many communications peripheral controllers that can be used in a variety of communications and networking applications. Three new high-performance channels (compared to the MPC860) support emerging protocols such as 155Mbps ATM and Fast Ethernet. This user's manual is intended for software and hardware developers and application programers who wish to develop products that use the MPC8260. It assumes a basic understanding of computer networking, OSI layers, RISC architecture and communications protocols.

**Order by: MPC8260UM/D**

## Queued Analog-to Digital Converter Reference Manual

The Queued Analog-to-Digital Converter (QADC) is a 10-bit, unipolar, successive approximation converter module. It supports 16 analog channels with internal multiplexing or 44 channels in the expanded, externally multiplexed mode. This manual provides information on the operation and use of the module, including Signal Descriptions, Configuration and Control, External Multiplexing, Pin Connection Considerations, Analog Subsystem, Digital Control, Interrupts, and examples of Queue Priority schemes.

**Order by: QADCRM/AD**

---

## MC68MH360, MPC860MH and MPC860DH: A Supplement to the MC68360 and MPC860 User's Manuals

The standard QUICC family members work in Time Division Multiplexed (TDM) applications but can only support one logical channel per Serial Communication Controller (SCC). The QMC (QUICC Multichannel Controller) protocol emulates up to 64 logical channels within one SCC using the same TDM physical interface. The QMC parts – MC68MH360, MPC860MH and MPC860DH – are pin-compatible with their respective family members and can be used in identical applications with minor adjustments. This manual provides an overview of the protocol and describes the use and operation of the devices.

**Order by: QMCSUPPLEMENT/AD**

---

## Queued Serial Module Reference Manual

The Queued Serial Module (QSM) is an integral module in Motorola's family of embedded microcontrollers. Its two sub-modules provide the MCU with two independent serial interfaces: the Queued Serial Peripheral Interface (QSPI) is a full-duplex, synchronous serial interface designed for communication with peripherals and other MCUs; the Serial Communications Interface (SCI) is a full-duplex UART. This Manual describes the capabilities, operation and functions of the QSM, including details of registers, operational flow diagrams and signal descriptions.

**Order by: QSMRM/AD**

---

## MPC500 Family: RCPU Reference Manual

The RCPU is a single-issue, 32-bit implementation of the PowerPC architecture, used in the MPC500 family of microcontrollers. This manual describes the RCPU for system software and hardware developers intending to develop products for RCPU-based systems. Topics include an overview of the architecture and features; Registers; Operand Conventions; Addressing Modes and Instruction Set Summary; Instruction Cache; Exceptions; Instruction Timing; Development Support; and full descriptions of individual instructions.

**Order by: RCPURM/AD**

## Single-Chip Integration Module Reference Manual

The Single-Chip Integration Module (SCIM) forms part of many of Motorola's 16 and 32-bit modular MCUs. It supplies a clock signal to the other modules, provides system protection features, manages the external bus, and provides on-chip chip-select signals and I/O ports. This manual describes all these functions and gives details of system reset and initialisation. Some MCUs necessarily contain a reduced pin-count version of the SCIM, and these variants are discussed. Separate appendices provide details of electrical and timing characteristics, and a summary of registers.

**Order by: SCIMRM/AD**

## Shipment Tracking System User Manual

**Rev 1**

This User's Manual provides information about how to use and retrieve product shipment information from the Shipment Tracking System (STS) software. It includes an overview of Motorola's supply chain management, and detailed information about the 11 screens in the STS application. Procedures for correct data entry, and error messages associated with improper or missing data, are described.

**Order by: SHIPTRKUM/D**

## System Integration Module Reference Manual

This manual describes the capabilities, operation and functions of the System Integration Module (SIM), an integral module in many of Motorola's 16 and 32-bit modular microcontrollers. The SIM supplies a clock to the rest of the MCU; provides system protection features, on-chip Chip Select signals and I/O ports; and manages the external bus. This manual highlights CPU differences that affect the SIM; describes the protection features, clock generation, external bus interface, interrupt system, chip selects and reset procedures; and provides electrical and timing characteristics and register descriptions.

**Order by: SIMRM/AD**

## MPC500 Family: System Integration Unit Reference Manual

The System Interface Unit (SIU) and Peripheral Control Unit (PCU) of the MPC500 Family processors are implemented as two separate on-chip units, working together to provide system support and interfaces between external and on-chip memory and peripherals. They handle system protection, clocks, interrupt support, reset control, test support, chip selects and interfaces to external and internal buses. This reference manual defines the functionality of the units, and is intended for software and hardware developers working with MPC500 family systems.

**Order by: SIURM/AD**

## TIM08 Timer Interface Module Reference Manual

The Timer Interface Module is one of the modules in Motorola's M68HC08 family of microcontrollers. This manual describes the 4-channel implementation – the module can also be implemented with 2, 6 or 8 channels. It provides an overview of the timer features, signal descriptions, and detailed information on the prescaler, 16-bit modulo counter, capture compare unit, interrupt generation, and the handling of the different HC08 operating modes. Includes a chapter of applications information, and an appendix containing electrical specifications, memory map and register descriptions.

**Order by: TIM08RM/AD**

## M68300 Family Time Processor Unit Reference Manual

**Rev 3**

The TPU is an integrated module within the 32-bit M68300 Family. It is a special-purpose MCU performing a variety of both simple and complex timing tasks – including input capture, output compare, PWM, stepper motor control, and many others – to minimise CPU overhead. This Manual gives a practical overview of the module's features; a description of the content and use of the three types of register that configure the TPU and its 16 channels; a detailed explanation of the operation of each time function; and a detailed guide to the TPU architecture. Appendices include algorithm state descriptions and microinstruction formats.

**Order by: TPURM/AD**

# Technical Data Services

## Scattering Parameter Plotting Utility

**Rev 1**

An IBM compatible computer disk (5.25" floppy) that permits the user to view S-Parameter files on a VGA monitor. Two port S-parameters are displayed on a Smith® Chart as a function of frequency. One can also view stability circles, $f_t$ vs frequency and $G_{MAX}$ vs frequency as well as convert S-Parameters to H-, Y- or Z-Parameters.

**Order by: DK106/D**

---

## Impedance Matching Program

This 5.25" IBM compatible disk contains a specialized form of CAD specifically developed for RF power amplifier circuit design. Its data base contains input and output impedances for most of Motorola's RF power transistors and allows the user to match these impedances manually by means of a variety of matching elements. The impedances and the results of the matching elements are displayed on a Smith® Chart plot that allows the user to see graphically what effects are created by his/her choice of matching components

**Order by: DK107/D**

---

## Master Selection Guide

**Rev 22**

For the design engineer, the Motorola Master Selection Guide is perhaps the most important single document for the identification and preliminary selection of components for circuit and system designs. Within its pages is a complete listing and description of Motorola semiconductor devices currently in general use, and those recommended for new designs. It serves two purposes:

1. It lists all standard products in the vast Motorola semiconductor inventory for rapid identification.
2. It divides this total product offering into a variety of major product categories, with sufficient technical information to permit an intelligent first-order evaluation as to the most suitable devices for a specific application.

**Order by: SG73/D**

# Index to Books, Guides and User's Manuals

## Data Books and Handbooks



## Selector Guides & Application Literature



## User's Manuals

# Technical Data Services

**How to reach us:**

**USA/EUROPE/Locations not listed:** Motorola Literature Distribution;
P.O. Box 5405; Denver, Colorado 80217. 1-303-675-2140 or 1-800-441-2447

**Customer Focus Center: 1-800-521-6274**

**HOME PAGE:** http://motorola.com/semiconductors/

**JAPAN:** Motorola Japan Ltd.; SPS, Technical Information Center; 3-20-1, Minami-Azabu. Minato-ku, Tokyo 106-8573 Japan. 81-3-3440-3569

**ASIA/PACIFIC:** Motorola Semiconductors H.K. Ltd.; Silicon Harbour Centre,
2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong. 852-26668334