Motorola Semiconductor Engineering Bulletin

EB269

Using the SCI on Modular MCUs: An Example

By Sharon Darley Austin, Texas

Introduction

The serial communication interface (SCI) is part of the queued serial module (QSM) and multi-channel communication interface (MCCI) on modular microcontrollers. It is used to communicate with external devices and other MCUs via an asynchronous serial bus.

The example program published here was assembled with the assembler available from P&E Microsystems. The CPU32 code was assembled with IASM32, and the CPU16 code was assembled with IASM16. The code was run on P&E's debugger (ICD32 for the CPU32 code and ICD16 for the CPU16 code).

The program can be used for debugging purposes. For example, if the system is not working properly, this program could be used to print error messages to the computer screen.



Example Routine

This example routine will print a 5-character message to the computer screen using the SCI. If using the ICD32 or ICD16 debugger, complete these steps before running this program:

- 1. Connect an RS-232 cable from a serial port on the personal computer to the serial connector on the development board.
- 2. Once in the debugger, set the serial communications protocol to the correct COM port, 9600 baud, no parity, eight data bits, and one stop bit. For example, if using COM 2, type in the command: serial 2 9600 n 8 1.
- 3. Now, enable the serial communications by typing: serialon
- 4. Finally, enter this program and run it by typing:

g 400 (CPU32 code) or g 200 (CPU16 code)

CPU32 COde

SCCR0	EQU	\$FFFC08	
SCCR1	EQU	\$FFFC0A	
SCSR	EQU	\$FFFC0C	
SCDR	EQU	\$FFFC0E	
SYNCR	EQU	\$FFFA04	
SYPCR	EQU	\$FFFA21	
	ORG	\$400	;begin program at \$400, immediately after ;the exception table
INIT_SI	М		
	MOVE.B	#\$7F,(SYNCR).L	;increase clock speed
	CLR.B (SYPCR).L	;disable software watchdog
INIT_SC	'I		
	MOVE.W :	#\$0037,(SCCR0).L	;set the SCI baud rate to 9600
	MOVE.W	#\$000C,(SCCR1).L	;enable the receiver and transmitter
PRINT			
	LEA (ME	SSAGE).L,AO	;load the effective address of the
			imessage
			;to be printed into address register A0.
* The n	ext two	commands load the effective a	address of the last character
* of th	e messag	e into address register A1.	

MOVE.L A0,A1 ADDA.L #\$5,A1

* The next three commands check to see if the transmit data register is empty * by looking at the TDRE bit in the SCI status register (SCSR). If the TDRE bit * is zero, then there is data in register TDR that has not yet been sent to the * transmit serial shifter. If the TDRE bit is one, then the transfer has * occurred, and a new character may be written to register TDR. Thus, this * sequence of code loops until the TDRE bit is one. LOOP

MOVE.W (SCSR).L,DU	
ANDI.W #\$0100,D0	
BEQ LOOP	
MOVE.B (A0)+,D0	;move the current letter of the message
	; into D0. Then, increment A0 to point to
	;the next letter
MOVE.W D0,(SCDR).L	;transfer the current letter to SCDR
CMPA.L A1,A0	;check to see if at the end of the
	;message
BNE LOOP	; if not, print another character
FINISH	
BRA FINISH	;stay here when done
MESSAGE	
FCB '12345'	;"12345" will be printed

CPU16 CODE

SCCR0	EQU	\$FC08	
SCCR1	EQU	\$FC0A	
SCSR	EQU	\$FC0C	
SCDR	EQU	\$FC0E	
SYNCR	EQU	\$FA04	
SYPCR	EQU	\$FA21	
	ORG	\$200	;begin program at \$200,immediately after
			;the exception table
INIT_SI	ГM		
	LDAB #	\$0F	
	TBEK		
CLRB TBXK			
	LDAA #	\$7F	
	STAA S	YNCR	;increase clock speed
CLR SYPCR			;disable software watchdog

CLRA

STD

AIX

SUBE BNE

BRA

FCB

FINISH

MESSAGE

SCDR

#1

#\$01

LOOP

FINISH

'12345'

INIT_S	CI					
	LDD #\$0037					
	STD SCCR0	;set the SCI baud rate to 9600				
	LDD #\$000C					
	STD SCCR1	;enable the receiver and transmitter				
PRINT						
	LDX #MESSAGE	;load the address of the message ;to be printed into address register X.				
	LDE #\$0005	; counter that will count to end of				
		;message				
* The r	next three commands check to see if t	he transmit data register is empty				
* then	there is data in register TDP that h	LLUS register (SCSR). II IDREL IS ZERO,				
* tran	chiefe is data in register ink that is	is one then the transfer has				
* 00011	rred and a new character may be writ	ten to register TDP Thus this				
* acquerge of gode loops until the TDPE bit is one						
sequence of code toops where the thre bit is one.						
LOOP	LDAB 0,X	;1st char in accumulator B				
CHAR	LDAA SCSR	;see if TDRE bit in SCI Status Register				
	ANDA #\$01	;is cleared				
	BEQ CHAR	;wait until it is				

;store char to be printed in data

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights or the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and (\widehat{A}) are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

;register

;point to next char

;stay here when done

;loop to print next char

;"12345" will be printed

How to reach us:

USA/EUROPE/Locations Not Listed: Motorola Literature Distribution, P.O. Box 5405, Denver, Colorado 80217, 1-800-441-2447 or 1-303-675-2140. Customer Focus Center, 1-800-521-6274

JAPAN: Motorola Japan Ltd.: SPD, Strategic Planning Office, 141, 4-32-1 Nishi-Gotanda, Shinagawa-ku, Tokyo, Japan, 03-5487-8488 ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd., Silicon Harbour Centre, 2 Dai King Street, Tai Po Industrial Estate, Tai Po, New Territories, Hong Kong, 852-26629298

Mfax™, Motorola Fax Back System: RMFAX0@email.sps.mot.com; http://sps.motorola.com/mfax/; TOUCHTONE, 1-602-244-6609; US and Canada ONLY, 1-800-774-1848

HOME PAGE: http://motorola.com/sps/



Mfax is a trademark of Motorola, Inc.

© Motorola, Inc., 1999