Order this document
by AN1744/D

Motorola Semiconductor Application Note

AN1744

Resetting Microcontrollers During Power Transitions

By Greg Racino
Consumer Systems Group
Austin, Texas

Introduction

A simple function such as resetting an MCU during the application or removal of power can cause many problems if not handled properly. Symptoms of an improperly handled reset during power transitions can range from a slight delay in MCU response after power-up to very erratic, inconsistent behavior to total system failure.

This document covers the main issues relating to this problem and aims to lead the user of HC05, HC08, and HC11 devices to a safe and reliable approach to transitioning power in their application.

Although the information in this application note applies to most HC05, HC08, and HC11 devices, some variations in the sources of reset and implementation of the internal circuits do exist.

Consult the pertinent device specification in addition to this document to make sure your system will operate properly during power transitions.

Document Outline

Reset in its most basic function ensures that the MCU starts or restarts executing software code in a controlled manner. This document covers those situations that are related to resetting the MCU when power is applied and removed from the MCU. It does not cover general system protection features such as COP (computer operating properly), illegal instruction reset, and illegal address reset except where it pertains directly to guaranteeing a safe and reliable power transition. The general use of these types of protection features is covered in other application notes.

These factors determine whether the MCU will receive the proper reset signals, both internally and externally generated, during power transitions:

1. Influences on reset during power-up:
    a. The power-on reset function (POR)
    b. System power-up sequence
    c. Oscillator startup time effects on POR delay
    d. External reset pin
    e. Low-voltage reset (LVR)
    f. Effects of MOR (mask option register) on LVR in OTP (one-time programmable) devices
2. Influences on reset during power down:
    a. System protection mechanisms
    b. Low-voltage reset (LVR)
    c. Protecting non-volatile memory

In most applications, several of the reset sources can occur at the same time. The combination of these must be considered for the safe operation of the MCU application.

Influences During Power-Up

For the MCU to power up and begin executing software in a controlled manor, a certain sequence of events must occur beginning with the power-on reset circuit.

Referring to **Figure 1** and **Figure 2** will help explain the sequence of events that occurs inside and outside the MCU during the power-up sequence.

Power-On Reset Function (POR)

The power-on reset function is accomplished through the use of several circuits inside the MCU. The three main components of this function are:

1. POR circuit
2. POR counter chain logic
3. Internal reset logic

Refer to **Figure 1** for a block diagram of the components within the MCU.

All HC05s, HC11s, and HC08s contain an internal POR circuit. The purpose of the POR circuit is to pre-condition certain logical circuits within the MCU as V_{DD} begins to rise. Two of the circuits that are pre-conditioned by the POR circuit are the internal reset generation logic and the POR counter chain logic.

NOTE: *The POR circuit itself does not hold the MCU in reset for any pre-determined amount of time, but the combination of the POR circuit, the POR counter chain logic, and the MCU's reset logic will create the power-on delay (t_{POR} shown in **Figure 2**).*

As seen in **Figure 2**, the internal POR signal operates only during the early stages of power-up. In fact, at just above the voltage level required for the CMOS logic to begin recognizing logical states, this signal will no longer be asserted. The mechanism that negates the internal POR signal is feedback from all the circuits that use it, indicating that initialization has completed. This all takes place in a short period of time early in the power-up sequence.

Once the POR circuit initializes the reset logic, which in turn asserts the internal reset signal, the POR counter delay logic holds the MCU in a reset state for 4064 (HC05 and HC11) or 4096 (HC08) oscillator clock

cycles during the power-up sequence. This is referred to in most databooks as the oscillator stabilization time. The time, t_{POR}, will vary depending on the crystal frequency used, the voltage at which oscillations start, and the V_{DD} rise time.

Because the purpose of the POR circuit is only to precondition internal circuits/logic, it will not detect a loss of power to the MCU. In fact, the V_{DD} voltage must fall to a level much below the logic operating level, usually about 0.2 V, and remain there for several tens of milliseconds to re-arm itself to detect the next rise of V_{DD}. A temporary loss of V_{DD}, sometimes called brown-out, can cause internal logic storage elements to change state and thus potentially disrupt proper MCU operation. In this case, which is covered in this application note's section on **Influences on Reset During Power Down on page 13**, an LVR circuit is required to protect the MCU and give it a clean reset.

System Power-Up Sequence

As the MCU begins to experience a rising voltage on its V_{DD} pin, several things begin to happen:

- Once V_{DD} reaches a level sufficient for the CMOS logic to begin operating (~1-V range, typically), the internal oscillator circuits will begin to oscillate regardless of the type of oscillator used (crystal, ceramic resonator, resistor capacitor (RC), etc.). This voltage is given the term V_{OS}.
- Once the oscillator (and hence the internal systems clocks) begin to run, all internal logic that must be initialized during power-up must be in the correct logical state.
- At this point, the POR counter chain and reset logic will continue to assert the internal reset signal for a period of 4064 (HC05 and HC11) or 4096 (HC08) POR clock cycles.
- Once this period of time (which depends on the oscillator frequency) has expired, the POR circuit will negate its input into the reset logic.
- If no other sources of reset occur, the MCU will come out of reset and the CPU will begin executing code.

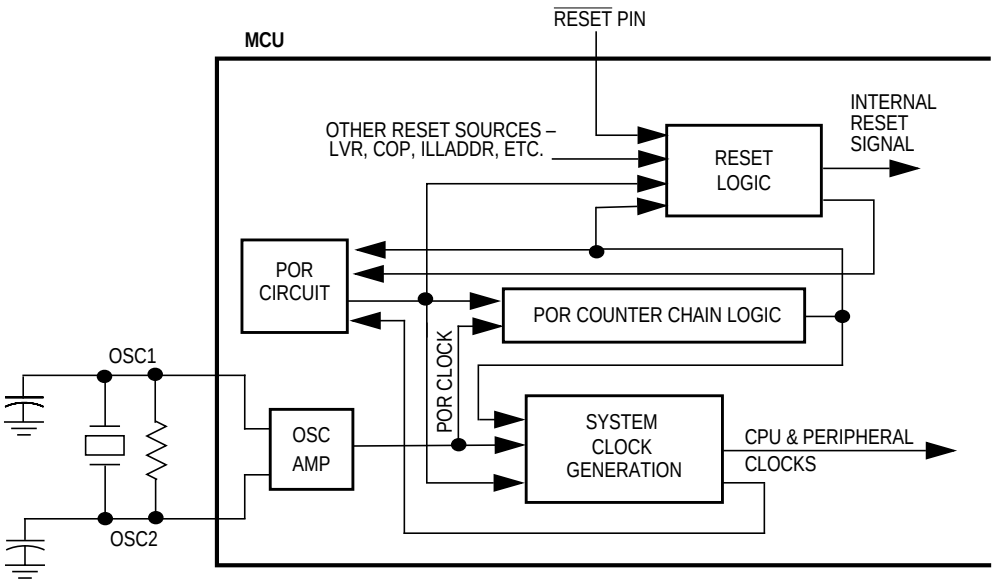


Figure 1. Internal Power-On Reset and Functional Blocks

When relying only on POR to provide MCU initialization during power-up, the rate at which V_{DD} rises (the V_{DD} slew rate) becomes very important. It is important to make sure V_{Run} reaches or exceeds the MCU's minimum operating voltage, V_{DDMIN}, for the chosen oscillator frequency and ambient operating temperature before the time $t_{OS} + t_{POR}$ as shown in **Figure 2**. This simple equation can help calculate if the V_{DD} slew rate is sufficient.

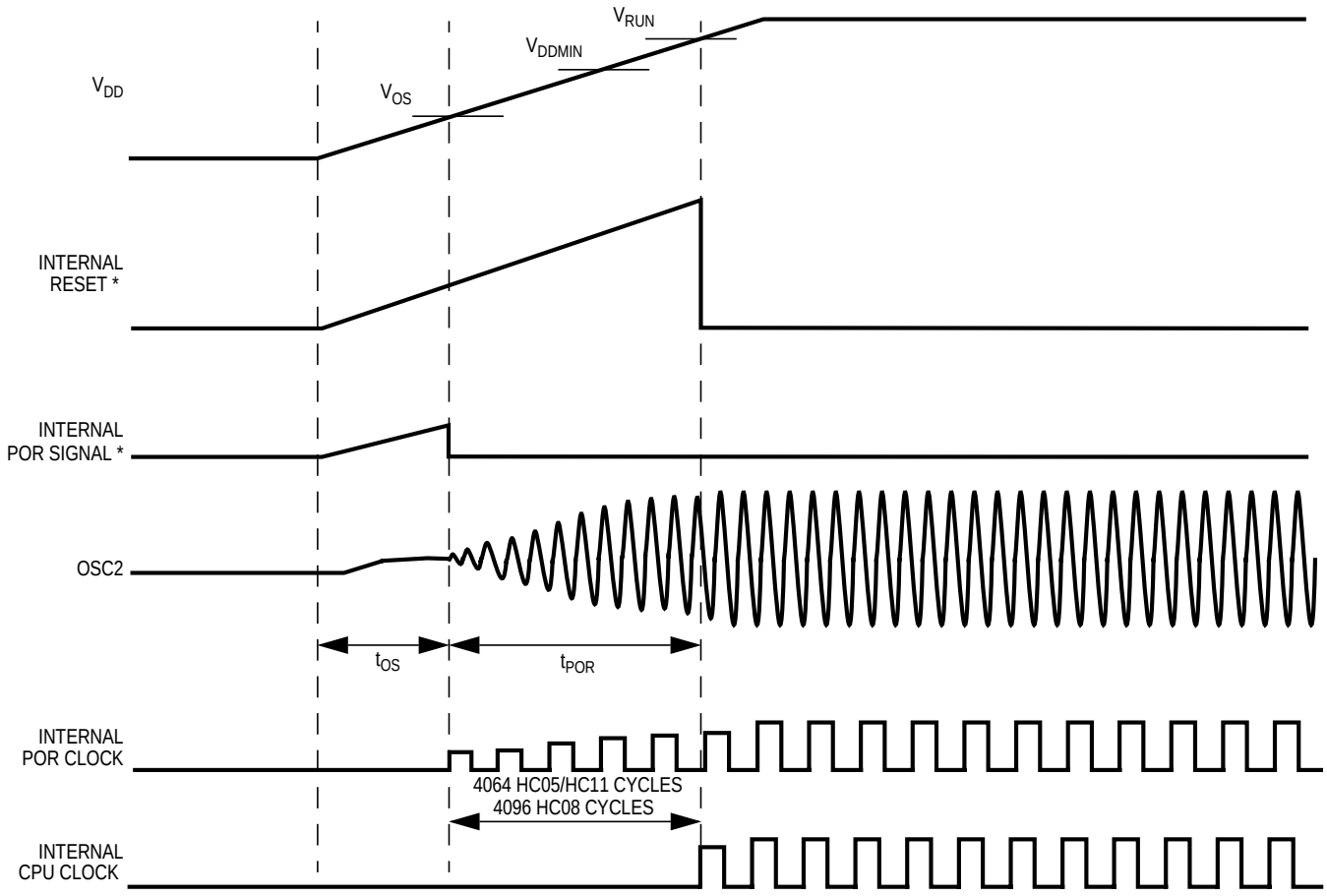
For HC05s and HC11s:

$$V_{Run} = V_{OS} + \left(\frac{4064}{f_{OSC}}\right) \times V_{DD}\text{slewrate}$$

For HC08s:

$$V_{Run} = V_{OS} + \left(\frac{4096}{f_{OSC}}\right) \times V_{DD}\text{slewrate}$$

In these equations, V_{Run} can never exceed the level of V_{DD}.



The oscillator clock rate and the number of POR clock cycles are not drawn to scale.

* These internal signals are shown to be active high; for instance, they cause an internal reset condition when high.

t_{OS} = The time between the point when V_{DD} begins to rise and the point at which the internal POR clock starts.This includes the time for the oscillator to begin oscillating, but does not include stabilization time.

t_{POR} = 4064 (HC05 and HC11) or 4096 (HC08) internal POR clock cycles

V_{OS} = V_{DD} voltage at which internal logic begins to operate. For instance, the oscillator will begin to start and the POR clock may begin to run.

V_{DDMIN} = Minimum operating V_{DD} voltage specification for the particular oscillator frequency and temperature specification

V_{Run} = V_{DD} voltage at which the MCU's CPU clocks start to run and the CPU will begin to execute code

Figure 2. Proper Power-Up Sequence

f_{OSC} is the frequency of the crystal, resonator, or external clock source connected to the MCU. Recall that V_{Run} is the voltage at which the MCU's CPU will attempt to execute code and perform its intended functions.

These equations assume that the POR clock will start running when V_{DD} reaches V_{OS}. This is the worse case condition and will be the case when an external clock is already present as the MCU is powering up.

Example:

An HC05 with 0- to 5-V rise on V_{DD} in 100 ms, V_{OS}= 1.0 V, f_{OSC}= 2 MHz

$$V_{Run} = 1.0 + \left(\frac{4064}{2\ \text{MHz}}\right) \times 50 = 1.1\ \text{V}$$

If the V_{DDMIN} specification at 2.0 MHz is greater than 1.1 V, then this slew rate is too low. The solution is to either speed up the slew rate or hold the MCU in reset by some other means until V_{DD} reaches V_{DDMIN}.

Effects of Oscillator Startup Time

In the previous example, if the V_{DD} rise time is fast enough, for example 1 ms, then V_{Run} would easily exceed V_{DDMIN}. In this case, the MCU will begin to execute code 2 ms after the oscillator starts when using a 2-MHz crystal (4064/2 MHz). When using the MCU's internal crystal oscillator circuit and a crystal frequency in the 2- to 4-MHz range, the oscillator startup time is typically 5 to 10 ms and will vary depending on the components used, board layout, V_{DD} levels, and V_{DD} slew rate. When using lower frequency crystals, such as 32.768 kHz, the startup time can be much longer – in the 300- to 500-ms range – and very low-power oscillator circuits can be as long as 2 to 8 seconds.

The main concern of crystal startup time comes in applications with rapid power-up, and the MCU is required to begin executing code very quickly. In these cases, reliable MCU startup is usually not a problem, since V_{DD} often will rise to safe levels ($V_{Run} > V_{DDMIN}$) before the oscillator starts and before $t_{OS} + t_{POR}$. If V_{DD} rises fast enough to make t_{OS} a significant contributor to the startup time, then oscillator startup time should not be assumed to be 0.

In the cases when V_{DD} is rising at a moderate rate and $t_{OS} + t_{POR}$ is about the same as the oscillator startup time, it is best to assume that the oscillator will start immediately when V_{DD} reaches V_{OS}. Take precautions to make sure the MCU is held in the reset state until V_{DD} safely reaches V_{DDMIN}.

External Reset

If the POR counter chain times out and the MCU's V_{DD} level is not in a safe operating range, it is the job of some other mechanism (either internal or external) to continue to hold the MCU in a reset state until V_{DD} has reached V_{DDMIN}. As mentioned earlier, the V_{DDMIN} level will depend on the MCU's voltage, temperature, and frequency specifications.

In the case of the external reset pin, a simple and low-cost way to ensure that the device stays in reset long enough is to put an RC delay on the reset pin as shown in **Figure 3**. The component values must be chosen to create enough delay to keep the $\overline{\text{RESET}}$ input below the V_{IL} specification (typically 0.2 x V_{DD}) until V_{DD} reaches V_{DDMIN}.

One consideration when choosing the values of R1 and C1 is to ensure that R1's value does not get too high and interfere with its ability to drive the $\overline{\text{RESET}}$ pin high given the worst case input leakage current of the pin. For example, if R1 is 100 Kohms, V_{DD} is 4.8 V, and the input leakage specification is 1 μA, then the input voltage on the $\overline{\text{RESET}}$ pin will be a minimum of 4.8 – (100 K x 1 μA) = 4.7 V.

This would be acceptable, but R1 values of 500 K or 1 M could possibly create an input level on the $\overline{\text{RESET}}$ pin that would be below the V_{IH} specification.

In addition, the capacitor C1 also should not get too large when using MCUs that have an active internal $\overline{\text{RESET}}$ pin pull down device. In this case, the surge current from the cap into the $\overline{\text{RESET}}$ pin when the pull down device turns on may be too large and may damage the pin circuitry. In general, limiting this current to less than 25 mA will be sufficient.

In this example circuit, diode D1 is used to quickly discharge the capacitor C1 when V_{DD} falls. This helps ensure a proper $\overline{\text{RESET}}$ pin input on the subsequent power-up, should it occur soon after the power down.

In general, the external RC network approach is only practical if the V_{DD} rise time is fairly fast. If it is too slow, the values of R1 and C1 must be too large, costly, and may interfere with normal operation.

When designing the $\overline{\text{RESET}}$ pin circuitry, consider how this signal is shared with other components within the system. In many MCUs, the $\overline{\text{RESET}}$ pin is actively driven low when an internal MCU reset condition exists.

Knowing what the active low drive characteristics are should also be considered when calculating the value of R1.

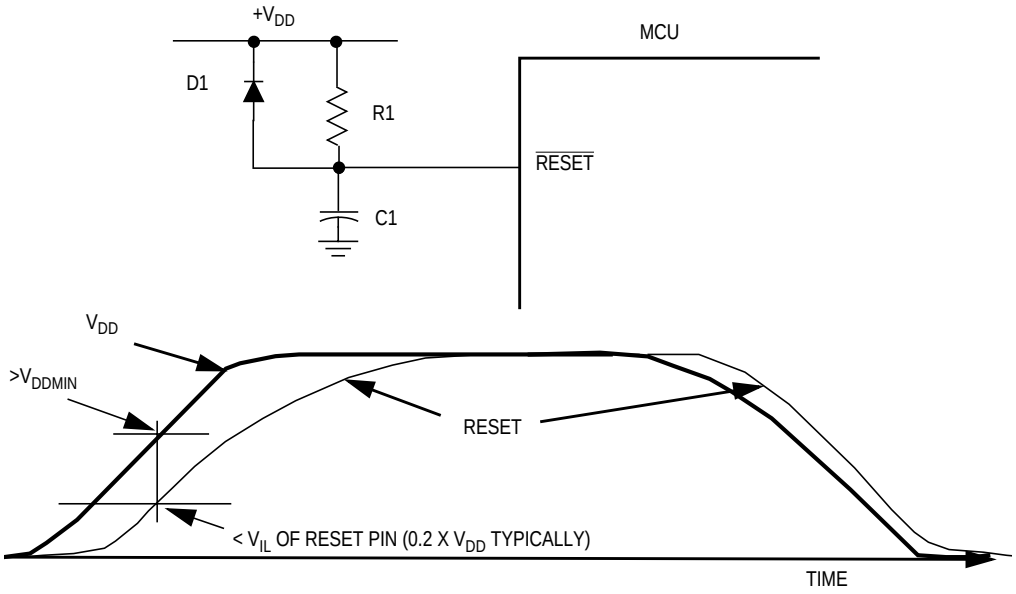


Figure 3. RC Circuit on External Reset Pin

Low-Voltage Reset

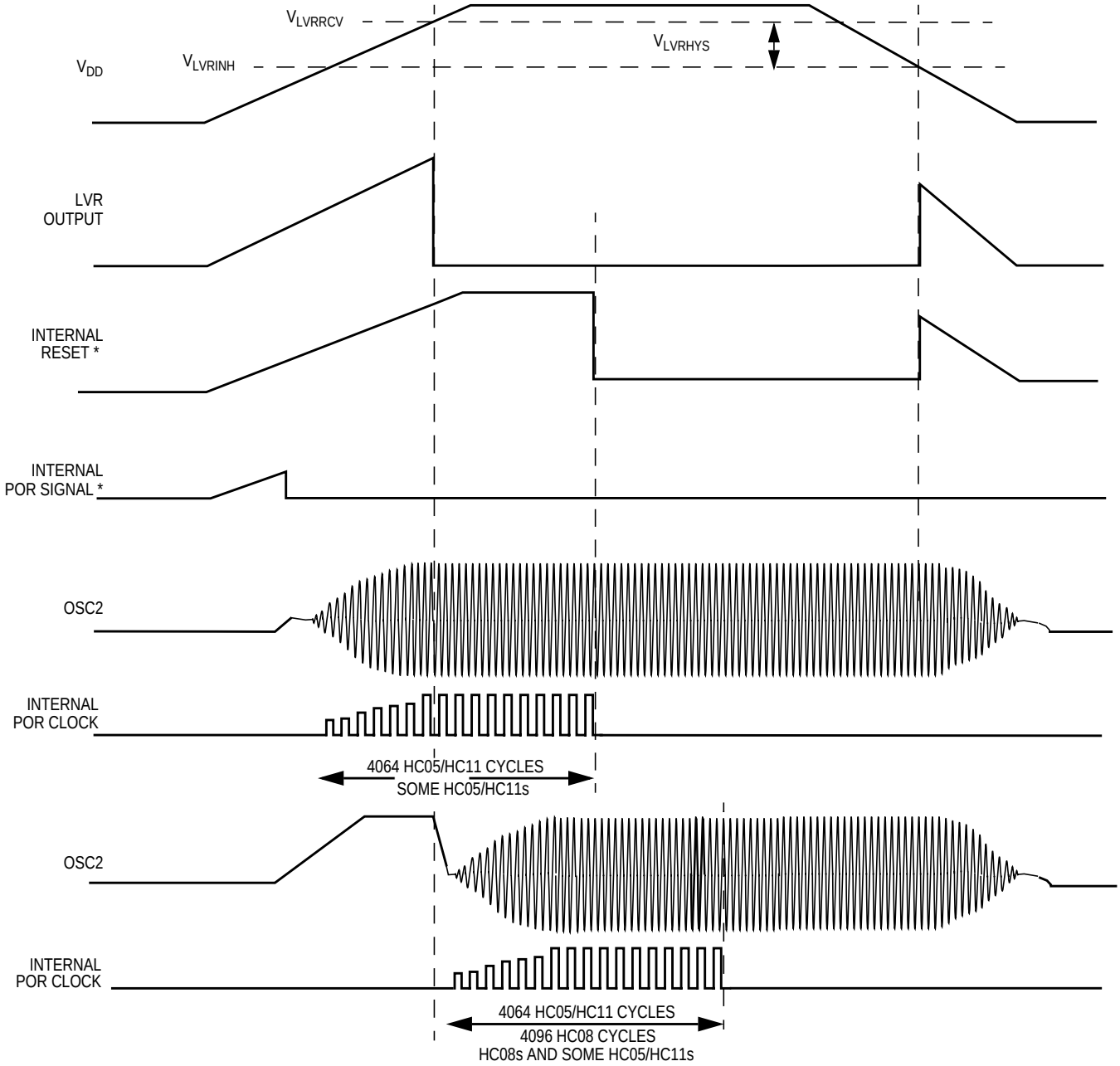
If an external RC circuit is not a suitable means of generating an external $\overline{\text{RESET}}$ signal during power-up or it is also desired to reset the MCU during power down or during a brown-out condition, then a low-voltage reset is the best choice.

The function of a low-voltage reset circuit is to continuously monitor the MCU's V_{DD} voltage level and generate a reset signal if V_{DD} ever falls below a certain trigger voltage, V_{LVRINH} (see **Figure 4**). The LVR circuit will continue to hold the MCU in reset until the V_{DD} voltage reaches another trigger voltage called the LVR recovery voltage, V_{LVRRCV}. The difference between V_{LVRRCV} and V_{LVRINH} is called the LVR hysteresis, $V_{LVRHYS.}$ V_{LVRHYS} exists to ensure the MCU does not bounce in and out of reset with small amounts of V_{DD} noise which typically exist in the system. V_{LVRHYS} is usually 100 to 200 mV.

A low-voltage reset can come from either an internal or external source on the $\overline{\text{RESET}}$ pin (see **Figure 1**). If the MCU does not have an internal LVR circuit, then an externally generated LVR can be performed with a device such as the Motorola MC34064, MC34164, MC33464, and MC33465 series. Sometimes the MCU's internal circuit that performs this function is called an LVI (low-voltage inhibit). Most of the time, the V_{LVRIHN} and V_{LVRRCV} trip points of the LVR circuit are fixed to a single set of voltages. If the trigger levels of an internal LVR are not suitable to prevent the device from operating outside of its voltage, frequency, and temperature specification, then an external LVR must be used.

Since the V_{DD} rise and fall times in many systems vary greatly, the internal LVR circuits are designed to balance quiescent operating current and response time. If V_{DD} rises or falls extremely fast, then the LVR circuit may not generate a reset input at precisely the specified V_{LVRIHN} and V_{LVRRCV} trip points. In the case of rising V_{DD}, the V_{LVRRCV} trip point will appear a little higher than that specified and the V_{LVRIHN} trip point will appear a little lower. As far as reliable power down and power-up are concerned, this is not a problem on V_{DD} rise, but it could pose a problem on V_{DD} fall if the V_{DD} slew rate exceeds about 50 V/ms.

On many HC05 and HC08 MCUs with an internal LVR circuit, the POR counter chain is cleared and prevented from counting while the LVR output is asserted. This provides extra stabilization time allowing V_{DD} to rise a little higher for an extra operating voltage margin. This is seen in **Figure 4** on the internal reset signal. In addition, in the HC08, the oscillator is also prevented from starting while the LVR reset is asserted.



V_{LVRRCV} = Low-voltage recovery voltage. The voltage at which the LVR circuit output is negated.
V_{LVRINH} = Low-voltage inhibit voltage. The voltage at which the LVR circuit output is asserted.
V_{LVRHYS} = Low-voltage reset hysteresis voltage. The difference between the recovery and inhibit voltage.
* These internal signals are shown to be active high; for example, they cause an internal reset condition when

Figure 4. Low-Voltage Reset Operation

Effects of Mask Option Register on the LVR in OTP Devices

Several Motorola MCUs with internal LVR circuits have a version that uses some sort of internal non-volatile memory as the main user memory:

- MC68HC705xx
- MC68HC805xx
- MC68HC908xx

These are commonly referred to as OTP (one-time programmable) devices. Because these devices do not have to go through a mask ROM cycle in the fabrication process to install application software and select mask options, the mask options in some MCUs are implemented in the same non-volatile memory bit cells that are used to build the main memory array. The non-volatile register that contains the mask option bits are called the mask option register (MOR). Many times, an LVR enable/disable bit will be present in the MOR.

During power-up and periodically thereafter, these non-volatile, MOR bits are read and periodically refreshed. The logical state of the non-volatile bit is copied into a static latch. The output of the static latch controls the circuitry to which the MOR bit is connected.

Since some HC05 MCUs with internal LVR circuits were not designed originally for excessively slow V_{DD} rise and fall times, this may pose a problem in these applications because the MOR signals that enable the LVR are not available until V_{DD} reaches the 2.0- to 2.5-V level and the internal clocks begin to run.

There are two reasons for this:

- First, many times the MOR bits are built into the main memory array and the MCU's internal clocks must be running and V_{DD} voltage must be at a sufficient level to read the memory array.
- Second, the V_{DD} level required to read a non-volatile memory bit is usually higher than that required to read a mask programmed option bit. Consequently, the operation of the MOR bits can only be guaranteed down to a specific V_{DD} level.

As a result, an MCU with an LVR enable/disable bit in the MOR will work better to emulate LVR functions during power down than during power-

up. However, it also has its limitations during power down. In those cases, the LVR will operate properly to a point. Then, due to the refresh operation, the MOR bit may turn off the LVR, which could release the internal reset signal.

If the V_{DD} rise and fall times are slow, then it is best to use an external RC circuit on the $\overline{\text{RESET}}$ pin or an external LVR circuit when dealing with some HC05 OPT devices.

In the mask ROM equivalent device, the LVR can be used safely for power-up as well as power down.

In the case of the MC68HC908xx Family of MCUs, the LVR MOR enable/disable bit, which is implemented with only a static latch and no non-volatile bit, is contained in the CONFIG register. This bit is initialized to enable the LVR during a POR reset and must be turned off by software if the LVR operation is not required. This approach is robust during power transitions but is not as immune to sudden and severe V_{DD} transients or brown-out conditions as the implementation, which uses the non-volatile bit with periodic refresh.

Influences on Reset During Power Down

In addition to encountering problems during power-up, the application designer also must consider the possibility of the MCU operating incorrectly as the system is powering down.

System Protection Mechanisms

Fortunately, there are robust means of protecting the system during power loss that are readily available. Among them are the use of the computer operating properly (COP) function, illegal address reset (IAR), or illegal instruction reset (IIR,) and the LVR circuit. The COP, IAR, and IIR are available on nearly all HC05, HC11, and HC08 devices and should be used to protect the system not only during power transitions but under normal operating conditions.

As V_{DD} begins to fall, the MCU may begin to operate outside of its specified operating range (V_{DD} falls below V_{DDMIN}). When this happens, the internal circuits may not perform as expected and could result in the MCU performing erratically and erroneously. If the COP, IAR, or IIR are in use, then the MCU will receive a reset before it operates under this

condition very long. It can take anywhere between one to several cycles, even several hundreds or thousands of cycles, before a reset from COP, IAR, or IIR will be asserted. In the case of a COP reset, it depends on when the COP was last serviced. In the case of IAR or IIR, it will depend on how long it takes for erroneous data to be read from memory or interpreted by the CPU. There is no way to predict how long this will take.

In many applications, especially those in which V_{DD} falls to V_{OS} rapidly, this is not an issue, since not many CPU clock cycles will elapse during the power down time.

LVR During Power Down

In applications which are not tolerant of even a single erroneous cycle, the use of either an internal or external LVR will protect the system even further. As shown in **Figure 4**, if the V_{LVRINH} voltage is at or above the minimum operating voltage of the MCU V_{DDMIN}, then the system will be forced into a reset once V_{DD} falls to an unsafe level.

Protecting Non-Volatile Memory

One of the MCU's common subsystems – on-chip erasable and programmable EEPROM or FLASH memory – can be particularly vulnerable to permanent data corruption during power loss. This is only a problem when the on-board memory is in the process of being altered (programmed or erased) during a power down. Since it usually takes 10 ms or more to perform a program or erase operation, there is a wide window of opportunity to lose power and interrupt the program or erase sequence. Even if the system makes use of an LVR to prevent erratic MCU operation during power down, there is no guarantee that a program or erase operation can complete successfully before the LVR reset occurs.

One way of preventing this problem is to use an LVR circuit not as a hardware source of reset but to serve as an indicator to the software. On many devices, the LVR status bit can be polled to predict when V_{DD} has started to fall, and the software can decide whether a program or erase operation has time to occur before V_{DD} is below V_{DDMIN}. Once the decision and action have been taken, the software can put the MCU into reset by using external circuitry.

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: Nippon Motorola Ltd.: SPD, Strategic Planning Office, 141, 4-32-1 Nishi-Gotanda, Shinigawa-Ku, Tokyo, Japan. 03-5487-8488

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd., 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., 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/

MOTOROLA