1 Introduction

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

This document covers the main issues relating to this problem and leads users of MC9RS08KA devices to a safe, reliable approach to transitioning power in an application.

Reset in its most basic function ensures the MCU starts or restarts executing software code in a controlled manner. This document covers situations related to resetting the MCU when power is applied and removed from the MCU such as power-on reset (POR) and low-voltage detect (LVD) reset. It also covers general system protection features such as computer operating properly (COP), illegal opcode detect (ILOP), and illegal address detect (ILAD). This application note
Summary of Resets

summarizes all sources of reset (synchronous, asynchronous, internal and external) available within the MC9RS08KA MCU generated during power-up or power-down transitions. Furthermore, system issues in the event of power-down are discussed.

On the MC9RS08KA, the reset pin is input only. Also, there is limited visibility of the internal reset and clocking. Therefore, it is difficult to observe the exact clocking and timing of the various reset sequences. The purpose of the application note is to provide information on the range of timing for the various reset conditions.

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.

2 Summary of Resets

The MC9RS08KA MCU has a number of reset sources described as synchronous and asynchronous. A synchronous reset comes from an MCU module configured to use the bus clock. An asynchronous reset comes from a different clock domain such as the COP that uses a completely asynchronous 1 kHz clock source, or from the MCU input RESET. An asynchronous reset generally takes longer to exit the reset sequence as some re-timing of the signal is required to avoid glitches.

![Figure 1. Sources of Reset within MC9RS08KA](image)

Resetting MC9RS08KA During Power Transitions, Rev. 0
Asynchronous reset sources:

- Low-voltage detect (LVD) reset
- Power-on reset (POR)
- External RESET pin
- Computer operating properly (COP) time-out

Synchronous reset sources:

- Illegal address detect (ILAD)
- Illegal opcode detect (ILOP) due to:
  - Detect of attempted illegal instruction execution from core module
  - Detect of attempted background debug (BGND) instruction execution when background debug module (BDM) disabled
  - Detect of STOP instruction execution when STOP instruction disabled
- Background debug controller (BDC) RESET command

Figure 1 illustrates how the synchronous and asynchronous resets are partitioned within the MCU.

All synchronous resets within the MCU behave similarly. Only the timing of these events is discussed. The asynchronous resets have slight differences in their behavior so a brief description of their operation is discussed as well as their timing.

## 2.1 Asynchronous Resets

There are a number of sources of asynchronous reset. This section details the operation and timing of POR, LVD, COP time-out, and RESET.

### 2.1.1 POR – Power on Reset

The POR function is implemented through the use of several circuits inside the MCU. The three main components of this function are:

- POR circuit
- POR counter chain logic
- Internal reset logic

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 pre-conditioned by the POR circuit are the internal reset generation logic and the POR counter chain logic.

The combination of the POR circuit, the POR counter chain logic, and the MCU’s reset logic creates a power-on delay. As seen in Figure 2, the internal POR signal only operates during the early stages of power-up. The mechanism that negates the internal POR signal is the feedback from all the circuits that use it, indicating that initialization has complete. This all occurs during the POR sequence and takes a period of 16 system clock cycles.
Summary of Resets

The POR circuit initializes the reset logic which in turn enables the bus clock after a further period of two system clock cycles. The POR counter delay logic then holds the MCU in a reset state, referred to as the reset exit sequence for a period of 70 bus clock cycles.

The total period for POR is 79 bus clock cycles (equivalent to 158 system clock cycles).

POR Period:

\[
\text{POR Period} = \text{POR Sequence} + \text{Reset Exit Sequence}
\]

\[
\text{POR Period} = (16 + 2) \text{ system clocks} + 70 \text{ bus clocks}
\]

\[
\text{POR Period} = 79 \text{ bus clocks}
\]

\[
\text{POR Period} = 158 \text{ system clocks}
\]

The only purpose of the POR circuit is to precondition internal 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, typically 0.9 V, and remain there for several microseconds to re-arm itself to detect the next rise of $V_{DD}$. A temporary loss of $V_{DD}$, called brown-out, can cause internal logic storage elements to change state and potentially disrupt proper MCU operation. In this case, covered in the next section of the application note, a LVR circuit is required to protect the MCU and give it a clean reset.

If the External $\text{RESET}$ pin is driven low during handling of a POR, the MCU has the reset operation detailed in Section 2.1.3, “External Reset”.

![Figure 2. POR and LVR Reset Timing Diagram](image)

2.1.2 LVR – Low Voltage Reset

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 the trip falling threshold voltage ($V_{TF}$). The LVR circuit continues to hold the MCU in reset until the $V_{DD}$ voltage reaches the trip rising threshold voltage ($V_{TR} = 1.8$ V).

Reseting MC9RS08KA During Power Transitions, Rev. 0

Freescale Semiconductor
$V_{TF} +$ hysteresis). The hysteresis, typically 80mV, is to ensure the MCU does not bounce in and out of reset with small amounts of $V_{DD}$ noise that exist in the system.

The reset sequence and timing after a low voltage event ($V_{DD} > V_{TR}$) is identical to that of POR. See Figure 2. The low-voltage detection circuit and the low-voltage detect reset are enabled by default after reset. Low voltage detect enable (LVDE) and low voltage detect reset enable (LVDRE) are controlled via the system power management status and control 1 register (SPMSC1).

Although Figure 2 shows POR and LVR share timing, the voltages at which they function are different. The point at which an LVR occurs is closer to $V_{DD}$ than a POR (see MC9RS08KA data sheet for voltage figures).

Figure 3 illustrates the falling threshold voltage ($V_{TF}$) and rising threshold ($V_{RF}$) voltage for an LVR event and the POR voltage ($V_{POR}$) and POR rearm voltage ($V_{REARM}$) for a POR event. $V_{DD}$ is not plotted on a linear scale. The difference between POR and LVD voltages are exaggerated for illustrative purpose.

![Figure 3. Supply Voltage Variation Generating POR and LVR events](image)

If the external $\overline{\text{RESET}}$ pin is driven in during handling of a LVR, the MCU has the reset operation detailed in Section 2.1.3, “External Reset”.

### 2.1.3 External Reset

External reset is used to drive an asynchronous reset to the chip. The $\overline{\text{RESET}}$ pin enable bit, RSTPE, is located in the system options register (SOPT) and is used to enable this input-only pin in any functional mode, user or test.

An external reset is recognized when $\overline{\text{RESET}}$ is externally driven low. Upon detection of an external reset, the MCU executes the reset exit flow. External reset is an asynchronous event and causes the internal
Summary of Resets

System reset to be asserted immediately. Timing associated with external resets varies dependent on the duration the \texttt{RESET} pin is held low.

Figure 4 shows the timing in the case where the \texttt{RESET} pin is asserted for a duration of between 1.5 and 69 bus clock cycles. In this case, the pin is released asynchronously and the pullup associated with the pad pulls the pad high. System reset is exited after 74 bus clock cycles.

![Figure 4. External Reset Asserted Between 1.5 and 69 Bus Clock Cycles](image)

If the \texttt{RESET} pin is held low beyond the 69 bus clock cycles (where internally the system is ready to exit reset), the MCU remains held in system reset and the system clock source is disabled. See Figure 5. In this case, when the pin is released asynchronously and the pullup associated with the pad pulls the pad high the bus clock is re-enabled 18 system clock cycles later. Looking at External Reset Asserted Greater Than 69 Bus Clock Cycles the MCU exits reset by the fifth rising edge of bus clock after the \texttt{RESET} pin is de-asserted. After the bus clock is re-established, the MCU exits system reset by the fifth rising edge of the bus clock.

![Figure 5. External Reset Asserted Greater Than 69 Bus Clock Cycles](image)
2.1.4 Computer Operating Properly (COP) Timer

The computer operating properly (COP) timer is a free running counter that helps software recover from code runaway. The COP is enabled on exit from reset. The COP is only disabled by clearing the computer operating properly watchdog enable (COPE) bit in the SOPT register. When a COP timer overflow occurs, an internal reset sequence is started. See Figure 6 for COP reset handling. A COP reset is prevented by periodically clearing the COP counter. The COP counter is cleared by writing any value to the system reset status (SRS) register address before a COP time-out occurs. When a COP clear is done, the counter is re-initialized to zero.

![Figure 6. COP Reset Timing Diagram](image)

The COP counter is initialized to zero by any system reset. The COP counter is also initialized to zero by the first write to the SOPT register after any system reset. Subsequent writes to SOPT have no effect on COP operation i.e. the COP is locked. Locking the COP protects the MCU in the event of code runaway meaning the COP can not be accidentally cleared. There is an associated short and long time-out controlled by the COPT bit in the SOPT register. Table 1 summarizes the timeouts. The COP defaults to long time-out ($2^8$ cycles). The COP counter is suspended and does not increment during background debug mode. When the MCU enters STOP mode, the COP counter is re-initialized to zero. The COP counter begins from zero as soon as the MCU exits STOP mode.

### Table 1. COP Timing Options

<table>
<thead>
<tr>
<th>COPT</th>
<th>COP Overflow Count</th>
<th>COP Overflow Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$2^5$</td>
<td>32 mS</td>
</tr>
<tr>
<td>1</td>
<td>$2^8$</td>
<td>256 mS</td>
</tr>
</tbody>
</table>

If the External RESET pin is driven in during handling of a COP reset, the MCU has the reset operation detailed in Section 2.1.3, “External Reset”.

Resetting MC9RS08KA During Power Transitions, Rev. 0
2.2 Synchronous Resets

Upon detection of a synchronous internal reset such as illegal address detect, illegal opcode detect, or background debug controller reset the reset flow is executed. Because no re-timing of these signals is required, a synchronous reset takes fewer clock cycles to complete than an asynchronous reset. Synchronous reset sources are cleared when the system reset is asserted. System reset is exited after 71 bus clock cycles. Figure 7 illustrates synchronous reset timing.

![Figure 7. Synchronous Reset Timing Diagram](image)

If the External RESET pin is driven in during handling of a synchronous internal reset, the MCU has the reset operation detailed in Section 2.1.3, “External Reset”.

3 System Issues During Power Down

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

3.1 System Protection Mechanisms

Fortunately, there are readily available means to protect the system during power loss. Among them are the use of the computer operating properly (COP) function, illegal address detect (ILAD), or illegal opcode detect (ILOP), and the LVR circuit. The COP, ILAD, and ILOP should be used to protect the system not only during power transitions, but also 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, ILAD, or ILOP are in use, the MCU resets at some time. However, it can take anywhere between one to several cycles, even several hundreds or thousands of cycles, before a reset from COP, ILAD, or ILOP is asserted. In the case of a COP reset, it depends on the COP count value (COPT) and when the COP was last serviced. In the case of ILAD or ILOP, it depends 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 $V_{OS}$ (threshold voltage at which internal logic and oscillator operates) rapidly, this is not an issue because not many CPU clock cycles will elapse during the power down time.

### 3.2 Protecting Non-Volatile Memory

The MCU’s on-chip erasable and programmable 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. Because it usually takes at least 40 ms to program and 500 ms or more to erase the MCU, there is a large 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. The Low Voltage-Detect Flag (LVDF) status bit in SPMSC1 can be polled to predict when $V_{DD}$ has started to fall. The software can decide whether a program or erase operation has time to occur before $V_{DD}$ is below $V_{DDMIN}$. After the decision and action has been taken, the software can put the MCU into reset by using external circuitry.
Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor 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 that may be provided in Freescale Semiconductor 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. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor 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 Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor 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 Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc. 2006. All rights reserved.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to http://www.freescale.com/epp.