Freescale Semiconductor
Application Note

Document Number: AN3262
Rev. 1, 10/2009

Designing Hardware for the MC9S12X D-Family

by: Martyn Gallop and Lela Garofolo
Applications Engineering
Freescale Semiconductor

1 Introduction

This document contains hardware guidelines for designing with the MC9S12X A, B, and D families of microcontrollers from Freescale Semiconductor. This includes:

- Power supply connections
- Control pin connections
- I/O connections and considerations
- Default pinout overview

NOTE

Electrical parameters mentioned in this document are subject to change in individual device specifications. Check each application against the latest data sheet for specific target devices.

2 Block Diagram

See Figure 1 for a block diagram of the MC9S12X D-family.

Contents

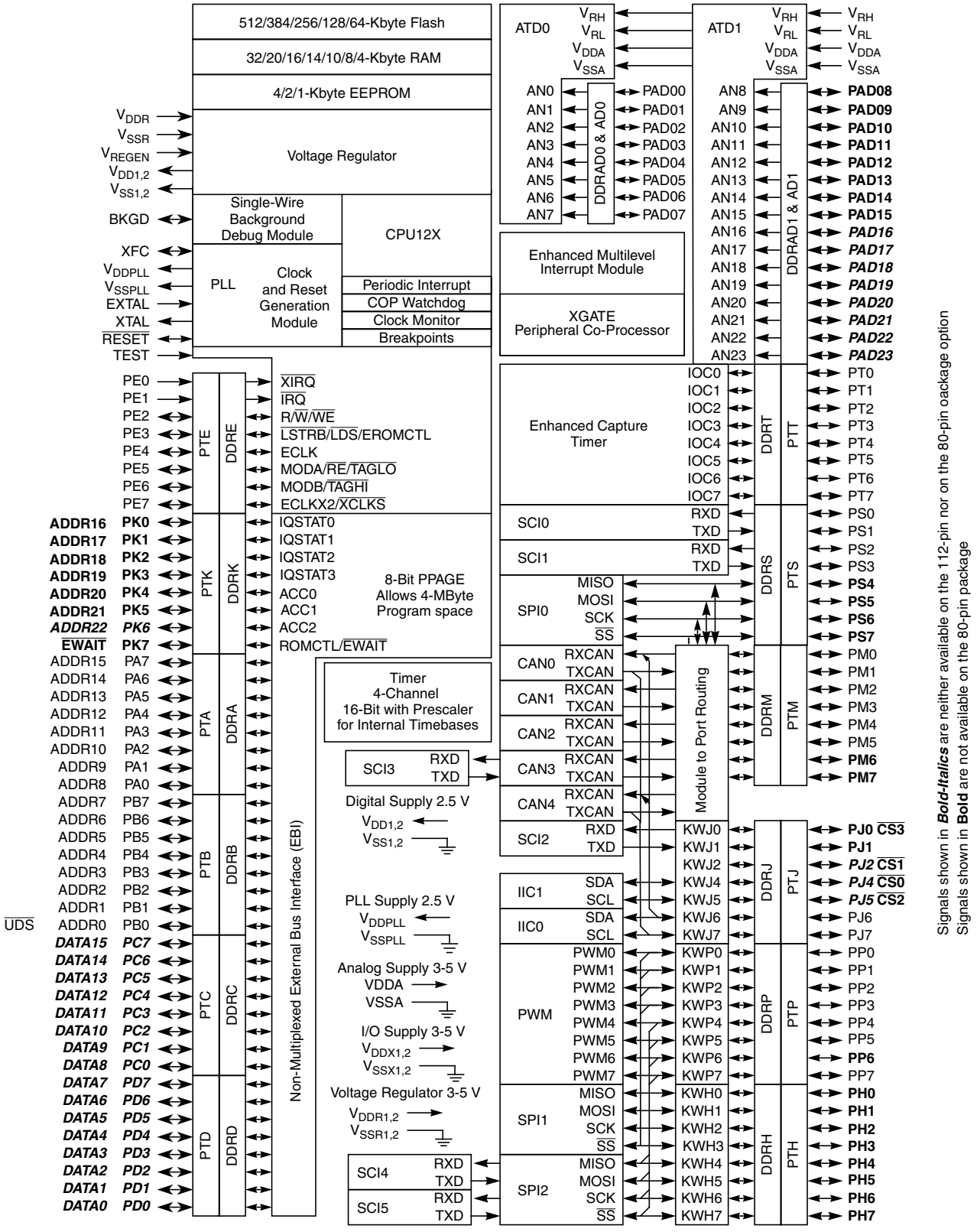


Figure 1. MC9S12X D-Family Block Diagram

3 Recommended Documentation

Other documentation useful for MC9S12X D-family hardware design can be found on the Freescale web site (http://www.freescale.com):

MC9S12XDP512 Data Sheet

Application Note AN2708 “An Introduction to the external Bus Interface on the MC9S12X”

Application Note AN2429 “Interfacing to the HCS12 ATD Module”

Application Note AN3328, “MC9S12XD Family Compatibility Considerations”

4 Power Supply Considerations

There are two power supply voltages:

- 2.5 V supply for logic, the CPU core, PLL, and oscillator
- 3–5 V supply (V_{DD5}) for the I/O buffers, voltage regulator, and ATD

In some cases, the different supplies may be connected internally on the chip by ESD protection diodes, but these connections are not intended for power distribution. Each pair of supply pins must be considered individually and **all power supply pins must be connected appropriately on the PCB.**

The core must not be powered down separately from the I/O.

The 5 V supply must not be switched off while the core is powered from an external 2.5 V supply.

When using the internal voltage regulator, the 2.5 V supply pins (V_{DD1}, V_{DD2} and V_{DDPLL}) should not be connected together on the PCB. They should be connected only to appropriate decoupling capacitors. **The internal voltage regulator is not designed to supply external circuitry.**

When using an external voltage regulator, the 2.5 V supply pins must be connected on the PCB with decoupling capacitors close to each supply pin pair — take steps to keep the V_{DDPLL} supply free from noise.

The PCB must be carefully laid out to ensure optimal operation of the internal voltage regulator as well as of the MCU itself. In general, it is advisable to route the MCU power and oscillator first.

4.1 Power Supply Layout Guidelines

Every supply pair must be decoupled by a ceramic capacitor, preferably surface mount, connected as close as possible to the corresponding pins. Keep the decoupling capacitors on the same side of the PCB as the MCU, preferably with no vias in the connections from the decoupling capacitor to the MCU pins.

Use low-impedance connections to V_{SS1}, V_{SS2}, and V_{SSR}. If not using a full board ground plane, connect the ground pins in a star arrangement with the central point of the ground star at the V_{SSR} pin. Where there is a full board ground plane, connect all V_{SS} supplies directly to the plane, except for V_{SSPLL} and the oscillator circuit ground.

Connect V_{SSPLL} directly to V_{SSR} (maintaining the star layout).

In general, the oscillator ground return should be considered as a separate ground, routed directly back to V_{SSPLL} and not connected elsewhere to the digital ground. The TEST pin is a static logic input that can also be connected to the oscillator ground to facilitate oscillator layout and minimize the impedance of the oscillator ground.

NOTE

The test pin should always be grounded in an application.

5 MC9S12X D-family Power Supplies

5.1 V_{DD1}, V_{DD2}

2.5 V supply for MCU core and peripheral logic.

If using the internal regulator, connect only to external decoupling capacitors.

5.2 V_{DDPLL}

2.5 V supply for oscillator and PLL.

If using the internal regulator, connect only to external decoupling capacitor.

5.3 V_{DDR}

Supply for regulator and ports A, B, E, H; connect to V_{DD5}.

Add 10 µF if running at the upper end of specification for bus speed.

5.4 V_{DDX}

Supply for all other ports; connect to V_{DD5}.

Add 10 µF if large loads are switched, for example, where there are several inputs/outputs switching milliamp loads.

5.5 V_{DDA}

Supply to ATD, port ADx, and voltage regulator reference; connect to V_{DD5}.

Where possible, avoid connecting external pull devices or logic to V_{DDA} or V_{SSA} tracks.

5.6 V_{RH}

Reference for ATD; connect to ATD reference potential.

The following constraint must be met to obtain full-scale, full-range ATD results:

$$V_{SSA} \leq V_{RL} \leq V_{IN} \leq V_{RH} \leq V_{DDA}$$

If the input level goes outside this range it will be clipped.

V_{RH} should always be ≤ V_{DDA}, to maintain conversion accuracy.

Avoid tracking the reference voltage through digital supply planes, and do not connect external digital pull devices to V_{RH} or V_{RL}.

Any noise or ripple on the reference level will be reflected in the conversion result. In extreme cases, where high frequency system noise is present, low ESR (equivalent series resistance) series inductors and/or ferrite beads may be advantageous on V_{RH}. Series resistance should be avoided, as each ATD reference draws approximately 375 µA from the reference supply (V_{RH} = 5 V).

5.7 Ground

The oscillator return should be connected directly to V_{SSPLL}, which should in turn be connected to all other digital grounds via a low-impedance ground plane (preferred) or a ‘star’ configuration centred at V_{SSR}.

The oscillator ground should not be otherwise connected to the digital ground.

5.8 V_{SSA}, V_{SSR}

Where ATD accuracy is important, these should star separately from the external voltage regulator or voltage source.

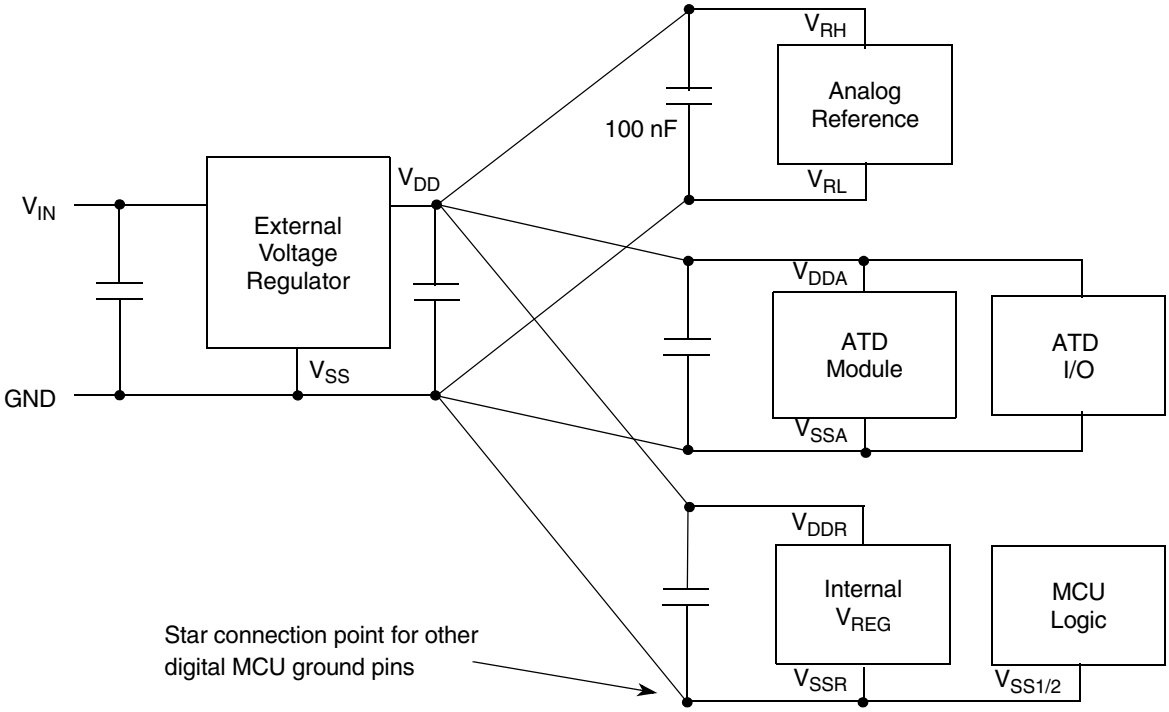


Figure 2. ATD Star Configuration

6 Pin Considerations Collection – Control

There are several control pins on the device that require special consideration at the design stage.

6.1 TEST Pin

This pin should always be grounded in an application.

This is a digital input and presents a static load; it can, therefore, be connected to the oscillator ground return, without concern.

NOTE

Failure to ground pin may result in unexpected operation.

6.2 $\overline{\text{RESET}}$ Pin

This is an open-drain, active-low, bidirectional control pin. It has an internal pullup for improved system reliability in the event of a PCB connectivity fault.

On assertion of any reset, the MCU releases internal control of the $\overline{\text{RESET}}$ pin after 128 SYSCLK cycles, and then samples the $\overline{\text{RESET}}$ pin after a further 64 SYSCLK cycles. If the pin reads low, the MCU determines that this was an external reset request and takes the external/POR reset vector. If it reads high, the MCU tests the internal reset sources and takes the appropriate reset vector.

If the time constant of external devices connected to the reset pin is too long, an internally generated reset may be detected as an external reset. If the desired application reset behavior is the same for all reset sources, this is not an issue; however, some applications may require different behavior for different reset sources.

Being able to differentiate the resources easily in an application can assist with debugging unexpected reset scenarios. To guarantee that the internal reset vectors (COP and CM) can be recognized by the reset logic, the rise time for the $\overline{\text{RESET}}$ pin to reach 3.15 V (a guaranteed input logic 1 for V_{DD5} = 5 V) must be less than 64 oscillator (SYSCLK) cycles or less than 11 µs (64 f_{SCM} cycles @ 5.5 MHz).

The internal voltage regulator supports a low voltage reset signal that will cleanly hold the device in reset if the internal core voltage drops below a safe level (approximately 2.25 V).

6.3 PE7: $\overline{\text{XCLKS}}$ Pin

During reset[1] PE7 is control pin, XCLKS, used to select the oscillator configuration. PE7 also has alternative functions for ECLKX2 and GPIO.

During rising edge of reset[1], PE7 becomes an input with a pullup device enabled, and the value at the pin is latched to select the oscillator configuration. An input of logic low configures the oscillator as a full-swing Pierce oscillator, or external clock drive on EXTAL. An input of logic high configures the oscillator as a loop-controlled Pierce oscillator.

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP.

The logic level must be clearly defined at reset[1] to select the oscillator configuration appropriate to the external oscillator component layout. Because there is a pullup device enabled during reset, the default oscillator configuration is loop-controlled Pierce oscillator. It is recommended to drive or pull the pin externally to ensure logic value and compensate for noise. Also, use caution when using alternative functions, especially GPIO, where the application also uses PE7 as GPIO following reset. In general, where PE7 is required as GPIO following reset[1], it is best suited as output with an external pulldown or pullup connected to the pin.

6.4 PE6: MODB and PE5: MODA

During reset PE6 and PE5 are control pins, MODB and MODA, used for selecting the user, or chip mode. See Freescale document MC9S12XDP512, *MC9S12XDP512 Data Sheet,* section 1.4, “Chip Configuration Summary,” for in-depth chip mode details. PE6 and PE5 also have an alternative function for GPIO.

During rising edge of reset[1], PE6 and PE5 become inputs with a pulldown device enabled, and the value at the pin is latched to select the chip mode. Because pulldowns are active during reset, the default chip mode is single chip, which is typically used by nearly all applications. It is recommended to drive or pull them externally in electrically noisy environments. Use caution when using alternative function GPIO, where the application will use PE6 and PE5 as GPIO directly following reset[1]. In general, if PE6 and PE5 are required as GPIO following reset[1], they are best suited as outputs with an external pulldown or pullup connected to the pin.

Note that the expanded bus, or the S12X_EBI module, is only available on the 144 LQFP package option. The S12X_EBI module is essentially still on the 112 LQFP and smaller packages, but it is not fully implemented or tested to guarantee proper operation. Therefore, the expanded bus-associated pins used for control, PE6 and PE5, on the 144 LQFP that are on the 112 LQFP and smaller package options should still be considered. The pins should either be tied directly to ground, or else the user should follow the general recommendation given above for implementing as GPIO.

6.5 PE4: ECLK

In all modes except normal single chip mode, this pin defaults to ECLK enabled for bus clock out. The ECLK signal is useful for debugging — therefore it is recommended to run a test point to the signal. ECLK output can be enabled in normal single chip mode via the ECLKCTL register, and also allows control for ECLK dividing by two, three, or four. If using ECLK as an output in the end application, note that ECLK signal out is a relatively high frequency clock line.

Use caution when using PE4 as a GPIO in single chip mode, because out of reset ECLK defaults to enabled. Special single chip mode is enabled whenever a debug cable is connected and a reset performed.

6.6 PE1: $\overline{\text{IRQ}}$

PE1 is a general purpose input pin and optional maskable interrupt request input that can provide a means of applying asynchronous interrupt requests. This will wake up the MCU from Stop or Wait mode. By

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP.

default, this interrupt input is active low level sensitive but can be configured in software to falling edge sensitive in the IRQ control register (IRQCR).

6.7 PE0: $\overline{\text{XIRQ}}$

PE0 is a general purpose input pin and optional non-maskable interrupt request input that can provide a means of applying asynchronous interrupt requests. This will wake up the MCU from Stop or Wait mode. This interrupt input is active low level sensitive.

CAUTION

If an application requires Stop mode, this pin must be used with care (whether it is the source of the wake from the STOP signal or not). Behavior of the STOP instruction is directly linked to the $\overline{\text{XIRQ}}$ functionality. The X bit in the CPU condition codes register masks the $\overline{\text{XIRQ}}$ interrupt (preventing the interrupt vector from being taken) but it does not prevent the $\overline{\text{XIRQ}}$ pin from waking the MCU from Stop mode. As $\overline{\text{XIRQ}}$ is level sensitive, while this pin is low the MCU will not enter Stop mode, and failure to release logic low can cause the CPU to be stuck in the XIRQ interrupt.

6.8 BKGD/MODC

The BKGD/TAGHI/MODC pin is used as a pseudo open-drain pin for the background debug communication link.

The state of this pin is copied to the MODC bit at the rising edge of reset[1]. This pin controls whether the device enters special mode on release of reset. Internally, it has a permanently enabled pullup, to ensure that it enters normal mode if not connected. This pullup may not be strong enough to ensure adequate rise times for BDM communication with all development tools. To support BDM it is required to fit an external pullup resistor. See Section 7.2, “BKGD Pullup Value”.

Easy access to the $\overline{\text{RESET}}$, BKGD, 0V and V_{DD5} signals can facilitate debugging. It is recommended to include a standard 6-pin header in all design layouts to support in-system debug and reprogramming; the header need not be populated for production. See Section 7.1, “Standard Serial Debug Interface Connection”.

6.9 XFC

To use the PLL, an appropriate filter network should be fitted between the XFC pin and V_{DDPLL}. See Section 8, “MC9S12X D-family PLL”.

Fitting PLL filter components to all designs, whether they use the PLL or not, can be a good idea where BDM tools capable of high speed programming are being used.

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or full STOP.

6.10 V_{REGEN}

The V_{REGEN} pin controls enabling of the internal voltage regulator. It is sampled on the rising edge of the reset[1] pin and has an internal pullup (= internal regulator enabled) for improved system reliability in the case of a PCB connectivity fault.

7 Implementing a Standard Serial Debug Interface for the Background Debug Module (BDM)

The BDM module is a single-wire debug interface supported on MC9S12X D-family MCUs. Bidirectional communication is via a single pin on the MCU (the BKGD pin). Typically, a BDM interface cable will connect to four signals: BKGD, $\overline{\text{RESET}}$, V_{SS}, and V_{DD5}. V_{REGEN} at logic level high enables the internal regulator.

7.1 Standard Serial Debug Interface Connection

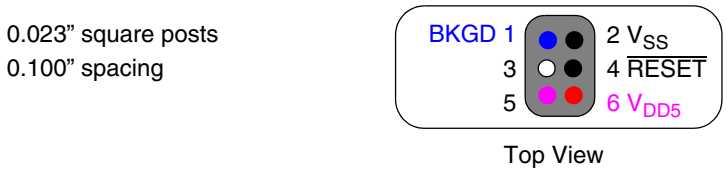


Figure 3. Connector Pinout

Although a debug cable may derive power from a source other than the target board, the target V_{DD5} and GND signals may be required with some cables to provide a reference level for the cable’s I/O buffers. Therefore, routing these supplies to the connector is also recommended.

7.2 BKGD Pullup Value

By default, the BDM module is clocked from the external oscillator clock (EXTAL); however, some BDM programming utilities may select the PLL as the clock source for the BDM.

Aim for t = R*C of about 20% of the maximum BDM speed with the BDM speed = 1/16 of the maximum expected BDM module clock.

So, for a 25 MHz bus => 1 µs bit time => at 100 pF (nominal load) => R = 200 ns / 100 ps = 2 kΩ.

This simplifies to R = 32 x 10^9 / BDM module clock frequency. For example, for 16 MHz, R = 2 kΩ.

Although a low impedance may not be necessary for communication at the application’s target bus speed, using a lower impedance value may be advantageous in electrically noisy environments. However, the resistor value should always be >> 600 Ω, to ensure that a low state will be detected.

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP.

The drive capability of the BDM tool used must also be considered — it must be able to drive the selected resistor and line capacitance low. Check any concerns with the BDM cable supplier.

8 MC9S12X D-family PLL

8.1 PLL Filter Circuit

The MC9S12X D-family PLL allows programmable bus frequencies to be generated from the oscillator clock.

With the PLL disabled:

$$\text{ECLK (bus) freq} = \text{Oscillator (Crystal) Freq} / 2$$

With the PLL enabled:

$$\text{ECLK (bus) freq} = \text{Oscillator (Crystal) Freq} \times \frac{(SYNR + 1)}{(REFDIV + 1)}$$

The PLL on the MC9S12X D-family requires a three-component passive filter to be connected to the XFC pin, as in Figure 4. The values of the filter components are application specific. The equations for calculating the filter are contained in the device data sheet or reference manual, but it is simpler to use the PLL filter calculator available from the Freescale web site at http://www.freescale.com.

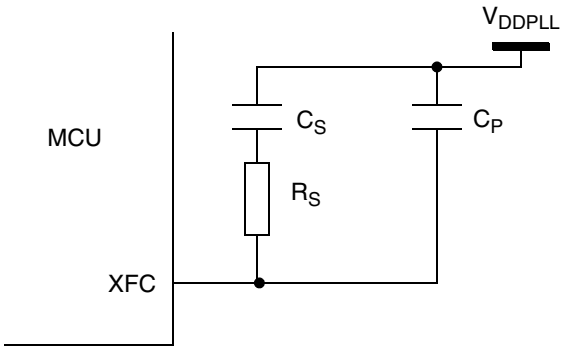


Figure 4. XFC Circuit

If, for some reason, it is decided not to fit a filter network, the XFC pin should be connected via a pullup resistor (1 kΩ to 5 kΩ) to V_{DDPLL}.

CAUTION

Never connect XFC to V_{SSPLL}.

If the application does not use the PLL, the software should clear the PLLON bit to disable the PLL following reset.

9 Oscillator Options

On the MC9S12X D-family there are three oscillator options:

- Reduced power, loop-controlled Pierce (LCP) configuration recommended for the majority of applications.
- Full swing (2.5 V) Pierce (FSP) configuration for high frequency crystal operation.
- External clock source:
  - Uses the same MCU configuration as the full-swing Pierce option.
  - The oscillator is a 2.5 V module so an external clock should be ~ 2.5 V peak-peak.
  - When implementing a clock with a 5 V ‘canned’ oscillator, add a 50% potential divider to reduce the clock to 2.5 V.

The loop controlled Pierce configuration is suitable for most applications. It has good startup performance while providing lower power consumption than the FSP. It can reduce mechanical stress on the crystal due to its lower amplitude sinusoidal waveform and, as it does not require external resistors, it offers a lower cost solution and simpler PCB design.

The full swing Pierce is included primarily for compatibility with designs transitioning from HCS12 already using the FSP configuration and also for designs that demand a higher frequency oscillator.

The XCLKS signal selects the oscillator configuration. On the MC9S12X D-family, a 5 V level on PE7 ($\overline{\text{XCLKS}}$) will select the LCP oscillator option and, as this pin defaults to having an internal pullup enabled, this is the default if this pin is unconnected.

The oscillator configuration is selected during the reset low phase while a clock quality check is being performed. This is the case for:

- Power-on reset or low-voltage reset
- Clock monitor reset
- Any reset while in self-clock mode or full Stop mode

The selected oscillator configuration is frozen with the rising edge of the appropriate reset.

An external or COP reset during normal run, wait, or pseudo-stop mode operation will not modify the oscillator configuration. This relaxes the condition where an application is using PE7 as a GPIO following reset and the I/O level might result in an incorrect oscillator selection following a watchdog event.

The use of a fundamental resonator or crystal is always recommended in preference to an overtone resonator or crystal. Overtone crystals or resonators may be used with the FSP configuration following careful component selection. They are not suitable for the LCP configuration.

It is recommended that the suitability of a crystal or resonator for use in a particular application be confirmed with the crystal or resonator manufacturer.

The value of the load capacitors is specific to the crystal (and its use with the MC9S12X D-family oscillator). Load capacitor values should be confirmed with the crystal or resonator manufacturer.

9.1 Oscillator Layout Considerations

Good practice is important when laying out a PCB for any oscillator configuration and it is a good idea to lay out the oscillator first in any design.

The PCB layout is equally as critical when using a ceramic resonator as when using a quartz crystal resonator.

Keep the oscillator components on the same side of the PCB as the MCU and as close to the MCU as possible (allowing for the fan-out of any I/O used on the oscillator side of the MCU).

Parasitic capacitance between EXTAL and XTAL must be kept to a minimum. The EXTAL and XTAL tracks should be kept short and routed apart.

Take care when mounting a metal cased resonator directly on a PCB where the case of the oscillator will be in close proximity to the EXTAL and XTAL tracks on the board, as the case can introduce parasitic capacitance between the two signals. Where this might occur, consider mounting the resonator on its side, introducing some spacing between the resonator case and the PCB, or grounding the oscillator case.

A ground plane under the oscillator circuit is not a concern and can be advantageous as it can prevent signals on adjacent layer interfering with the oscillator (any increased parasitic loading from EXTAL and XTAL tracks to the ground plane can be evaluated and could be compensated for in the load capacitor values).

Where possible, keep the oscillator tracks on the same side of the PCB as the oscillator — minimize vias in the oscillator circuit.

Where there is no ground plane under the oscillator, avoid routing other signals on any layers in the region of the oscillator components or tracks. Place an exclusion zone on all layers around the oscillator.

Include a decoupling capacitor close to the V_{DDPLL}/V_{SSPLL} pins. Avoid vias in the VxxPLL supply tracks from the decoupling capacitor.

Treat the ground signal for the oscillator as a separate ground, connecting to the main digital ground at one point only, close by V_{SSR}. Where the main digital ground is implemented with a ground plane, it is recommended to keep the oscillator / PLL ground separate from the ground plane. Avoid connecting external components connected to digital IO (pull devices, filter circuits, for example) to the oscillator ground to prevent noise or transitions on the IO signals disturbing the oscillator.

Routing the oscillator ground via the TEST pin can simplify the layout. (The TEST pin is a static digital input that will not disturb the oscillator ground.)

Keep the adjacent $\overline{\text{RESET}}$ signal clean. Where it is connected off the board or to a long track in a noisy environment, consider adding some series resistance.

Avoid routing signals under the PLL components or tracks, to minimize cross-talk.

These guidelines are valid for single sided, double sided, and multi-layer boards. On boards with multiple layers, it may be possible to locate the oscillator closer to the MCU by fanning out the adjacent I/O underneath the MCU. Moving the PLL filter components to the underside of the board will also help simplify the tracking adjacent to the oscillator.

10 CRG — Loop Controller Pierce Oscillator

Recommended for 4 MHz to 16 MHz crystal or ceramic resonators.

On the MC9S12X D-family, pull $\overline{\text{XCLKS}}$ pin high at reset.[1]

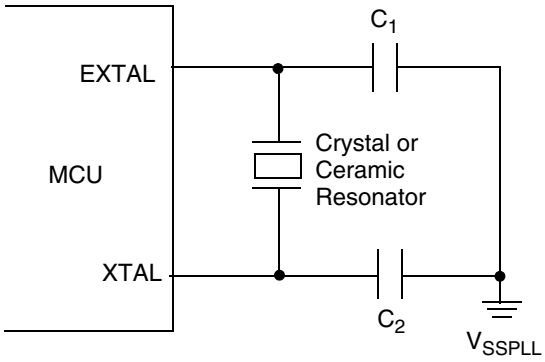


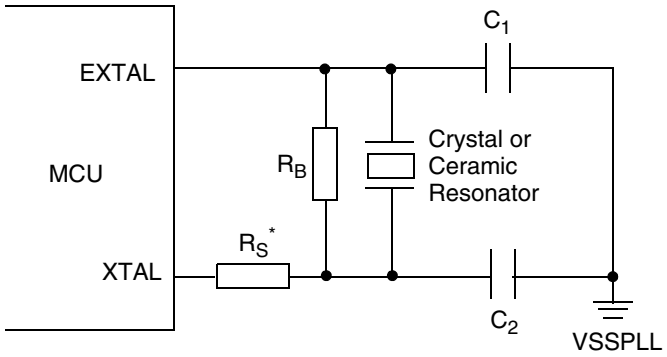
Figure 5. Loop Controlled Pierce Oscillator Configuration

C_1 and C_2 should be confirmed with the crystal or resonator manufacturer.

11 CRG — Full Swing Pierce Oscillator

For use with 500 kHz to 40 MHz crystal or ceramic resonators.

On the MC9S12X D-family, pull $\overline{\text{XCLKS}}$ pin low at reset. Typically used where a higher frequency oscillator is required, this is also the configuration to select when using an external 2.5 V oscillator (square wave) to drive EXTAL.



* R_S can be zero (shorted) when used with higher frequency crystals. Refer to manufacturer's data.

Figure 6. Full Swing Pierce Oscillator Configuration

R_B is required to bias the oscillator into its correct operating region. 1–2 MΩ is a reasonable value for R_B.

Values for R_S, C_1, and C_2 should be confirmed with the crystal or resonator manufacturer.

11.1 Oscillator Layout Example

For the loop controlled Pierce oscillator, R_S can be omitted or fitted with a zero ohm link and R_B can be omitted or not fitted.

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP.

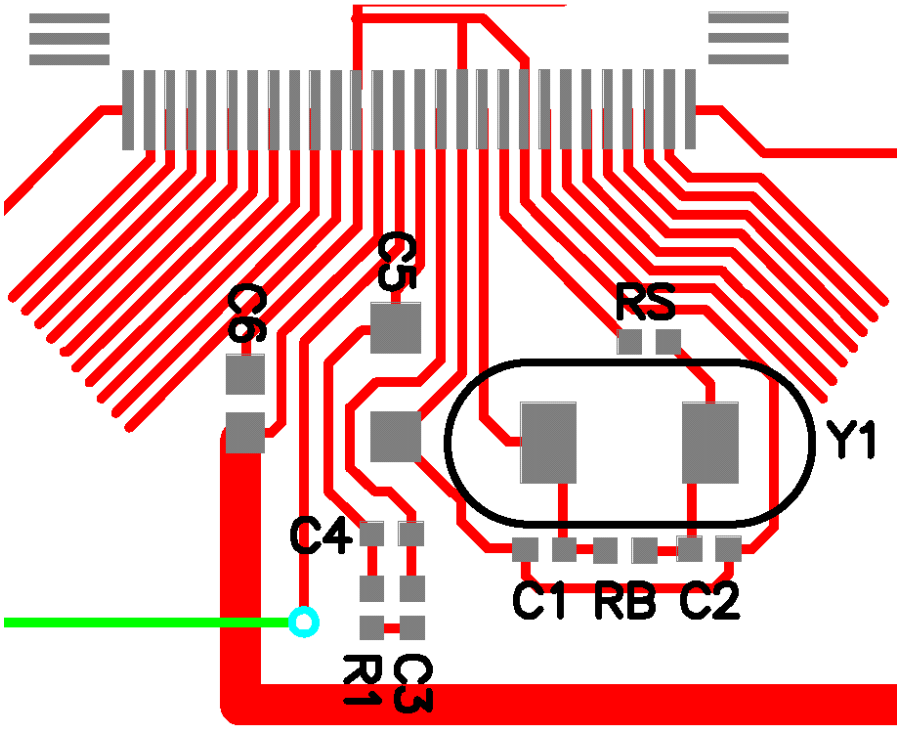


Figure 7. Oscillator Layout Example

12 Pin Considerations — Port Integration Module

The port integration module (PIM) controls the electrical pin properties and signal priorities of multiplexed functions. The PIM is the interface between peripheral modules and the I/O pins. Ports not being used by peripheral modules have the option to be used as general purpose input/output (GPIO) ports, and have controls associated to them.

12.1 MC9S12X D-family Core I/O Ports

Ports A, B, C, D, and K can be used as general purpose input/output (GPIO) ports. These ports also support expanded address and data for expanded bus modes. The expanded bus is available only on the 144 LQFP package option. For smaller package options, ports A, B, and K become dedicated for GPIO use.

CAUTION

For 112 LQFP and 80 QFP, the expanded bus S12X_EBI module is on the die. It is not fully implemented or guaranteed for functionality. Control pins for the S12X_EBI will still be active on the rising edge of reset[1]. For all packages, use caution when implementing pins that share control for the S12X_EBI module.

Registers for these ports are located in the Port Integration Module (PIM). These ports can also be used as GPIO in single chip modes (although Port E has some limitations, as detailed below).

The core I/O ports can be configured for internal pullups on a port wide basis (not all pins on Port E have pull devices). Control bits for the core I/O ports are in the Pullup Control Register (PUCR).

The core I/O ports can be configured for reduced drive strength on a port wide basis (not all pins on Port E have output capabiltiy). Reduced drive control bits are in the Reduced Drive Register (RDRIV).

Much of Port E can be used for GPIO. In expanded mode, Port E pins support bus control signals. Some Port E pins have additional control and configuration functions as detailed in Section 6, “Pin Considerations Collection – Control”.

12.1.1 Port E Control Pins

Port E pins, PE7:PE0, are associated with the expanded bus control signals and chip control signals XCLKS, IRQ, and XIRQ.

PE7 is the control for XCLKS, optional ECLKX2 output, and has the option to be used as GPIO. Caution should be used when choosing to use it as GPIO. On the rising edge of reset[1], PE7 becomes a dedicated input signal where the value of the pin is latched to select the oscillator configuration. Poor implementation of GPIO usage on this pin could cause the incorrect clock operation out of reset[1].

PE6 and PE5 are chip mode control signals for MODB and MODA, expanded bus (144 LQFP), and have the option to be used as GPIO. Caution should be used when using as GPIO. On the rising edge of reset[1], PE6 and PE5 become dedicated input signals where the value of the pins is latched to select the chip mode

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP..

(this applies to all package options). Poor implementation of GPIO on the pins could result in unexpected chip mode and operation out of reset[1].

PE4 is ECLK, output bus clock signal, that is enabled for all modes except normal single chip. Use caution when using pin as GPIO when in special single chip mode.

PE3 is the control signal for the option to enable or disable emulated flash out of reset. The input signal is don't-care unless the MCU is in an emulation mode. In general, emulation modes are mainly exploited by in-circuit emulators (ICE) which form a very special niche of applications.

PE0 is the control for the non-maskable interrupt request, XIRQ, that wakes up the MCU from Stop or Wait mode. PE0 as a general purpose pin signal has only input capability. Caution should be used when choosing to use it as GPI (general purpose input) in Stop or Wait mode. Poor implementation of GPI could cause an unexpected wakeup.

12.2 Port Integration Module (PIM) GPIO Ports

Ports H, J, P, M, and S support hardware interrupt functionality and alternative peripheral functionality. Registers for these ports are located in the port integration module (PIM). The PIM automatically switches control of each I/O as appropriate when a peripheral function is enabled for a specific pin.

Each GPIO port pin can be configured on a pin-by-pin basis for:

- I/P or O/P function
- Internal pullup / pulldown, approximately 100 µA load when driven by an external source.
- Full or reduced drive strength, useful for controlling EMC on SPI lines and PWM, for example.

Ports H, J, and P can also be configured on a pin-by-pin basis for:

- Edge sensitive interrupt inputs with glitch filtering. These can be used to wake the device from low power modes.

A useful feature of the PIM is that when an interrupt is enabled on one of the ports the appropriate pull device for the selected edge polarity is enabled:

- Falling edge = pullup enabled
- Rising edge = pulldown enabled

Ports M and S can also be configured on a pin-by-pin basis for:

- Open drain for wired-OR connections, useful for connecting multiple communications peripherals to the same bus.

All GPIO pins default to input on assertion of reset. Some are high impedance with no pull devices enabled, and some have pullups enabled. This is reflected in the default value of the associated data direction registers (xDDR).

To achieve minimum Stop or Wait I_{DD}, internal I/O pull devices should be configured by the application software so as not to conflict with external pin loads.

1. POR (Power on reset), Clock Monitor reset, reset in SCM (self clock mode), or reset that occurs while in full STOP.

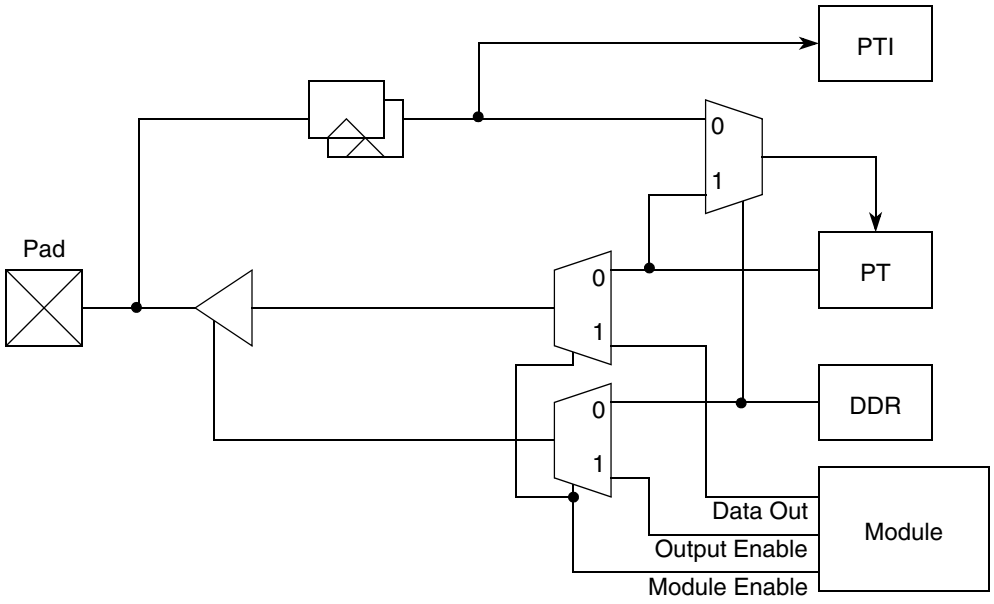


Figure 8. Illustration of Typical PIM GPIO Pin Functionality

12.2.1 Wakeup Pins

Some pins can generate hardware interrupts directly from I/O activity and can be used to bring a device out of low-power Wait or Stop modes.

Ports H, J, and P provide glitch filtered wakeup support and are configurable for polarity on a pin by pin basis. These are known as key wakeup ports because of their ability to wake the part from a 'sleep' mode and because they have traditionally been used to provide interrupt driven keypad support.

PE0, XIRQ (section 6.7) supports a low-level-sensitive interrupt and can wake the part up.

PE1, IRQ (section 6.6) supports a low-level-sensitive or falling edge interrupt and can wake the part up.

NOTE

Level sensitive interrupts are an effective means of extending the I/O interrupt capability of an MCU. Typically, multiple interrupt request lines are wire-or (open-drain) connected to the single input pin and a request/acknowledge protocol implemented. The interrupt source for each request is typically held low until the MCU has been able to determine which specific request is pending (out of those connected) and has acknowledged or serviced it.

Each of the MSCAN and SCI Rx lines can also be used to wake its respective module and the device up from Wait or Stop modes. This is intended primarily to enable a part to be wakened from sleep by activity on a CAN or LIN bus.

While the timer input capture pins can generate interrupts, this requires the timer to be active. Input capture lines can be used to wake a part from Wait mode but not from Stop mode (as the clocks to the timer will be halted).

12.3 Analog Ports

Ports AD0 and AD1 are associated with the ATD convertors and support both analog and digital functionality. Each pin can be configured for

- Analog input (default)
- Digital input (with optional pullup device)
- Digital output (with optional reduced drive)

Digital input enable mask registers (ATDDIEN) for these ports are located in the individual analog-to-digital modules, ATD0 and ATD1. These registers allow a digital input buffer to be enabled or disabled on a per-pin basis. Setting an ATDDIEN bit enables the corresponding digital input buffer continuously.

Disabling the digital input buffer on individual PTAD inputs performs two functions:

1. It immunizes any disturbance that reading the digital port may have on the analog sampling process.
2. It prevents an analog signal causing ‘cross over’ currents to flow in the digital input stage when the signal approaches the digital mid rail value.

Each port also has digital I/O control registers located in the port integration module and can be configured on a pin-by-pin basis for:

- I/P or O/P function (digital input requires the associated ATDDIEN bit be set)
- Internal pullup (approximately 100 µA load when driven by an external source)
- Full or reduced drive strength.

The pullup device is independent of the state of the associated ATDDIEN bit and should be disabled where a pin is being used for analog input.

If the ATD module is not enabled, the status of the ATD input stage will depend on the configuration of the ATDDIEN bits and the PIM register bits.

- With the ATDxDIENx bit = 0, the digital input stage is disabled. Unconnected inputs will have no effect.
- With ATDxDIEN = 1, the digital I/O is connected to the pin and unconnected pins should either have their internal pullup enabled or be configured as outputs.

12.4 ATD Connections

Analog conversion sequences can convert from one to eight or from one to sixteen channels at a time starting at any one of the channels. An analysis of the ATD sources to be converted may help utilize the flexibility of the ATD control and conversion structure. Sources with similar requirements can then be grouped onto adjacent ATD inputs, and the ATD configured appropriately for each conversion sequence.

The ATD converter’s accuracy is limited by the accuracy of the reference potentials. Noise on the reference potentials will result in noise on the digital output data stream; the reference potential lines do not reject reference noise. Ideally, the reference supply and ground should be routed separately to star ground configuration. See Section 5.5, “V_{DDA},” and Section 5.6, “V_{RH},” for more detail on V_{DDA} and V_{RH} connections.

Due to the sample-and-hold mechanism of the MC9S12X D-family ATD, charge-sharing between the external and the internal capacitances can cause a small voltage drop. Each analog input should have a capacitor, with good high frequency characteristics, between the input pin and V_{SSA}. The size of the external source capacitance will be application dependent; a basic guideline for minimizing the effect of this charge sharing is to keep the external capacitor greater than C_f as defined in the electrical characteristics section of the specific device user guide.

- For a maximum 10-bit sampling error of the input voltage ≤ 1 LSB, the external filter capacitor (C_f) should be ≥ 1024 * (C_{INS}-C_{INN}) or ≥ 12 nF.
- For an 8-bit conversion, 1 LSB is four times larger, so the minimum source capacitance for ≤ 1 LSB error is 256 * (C_{INS}-C_{INN}) or ≥ 3 nF.

The source impedance of the signal driver must also be considered when choosing the capacitor size. Optimizing the source impedance may be a compromise:

- External source impedance combined with the input capacitor will create a low-pass anti-aliasing filter, which can be used to attenuate unwanted frequency components and noise. Higher source impedance can result in rolling off of higher frequency components of interest in the input signals.
- Higher source impedance reduces current injection when the input exceeds the rail voltages.
- Lower source impedance avoids and reduces the error generated by input leakage (I_{in}). The maximum external source impedance of an analog signal is limited by the leakage into the pin.

A basic guideline for minimizing the effect of input leakage is as described in the data sheets. When $V_{REF} = V_{RH}–V_{RL}$ = 5.12 V, one 8-bit count = 20 mV and one 10-bit count = 5 mV

- For a maximum 10-bit error of < 1/2 LSB, R_S should be ≤ 2.5 kΩ (= 2.5 mV / 1 µA)
- For a maximum 8-bit error of < 1/2 LSB, R_S should be ≤ 10 kΩ (= 10 mV / 1 µA)

See application note AN2429 for further details and considerations on interfacing to the ATD ports.

13 MC9S12XDP512 MODRR Routing Options

The PIM module on the MC9S12X D-family can re-route the I/O connectivity for a number of communications peripherals depending on the value contained in the module routing register (MODRR). This is primarily intended to allow increased peripheral flexibility when using the 80-pin package, but can be used with the 112-pin and 144-pin packages to optimize PCB layout. Figure 9 and Figure 10 show MODRR routing options.

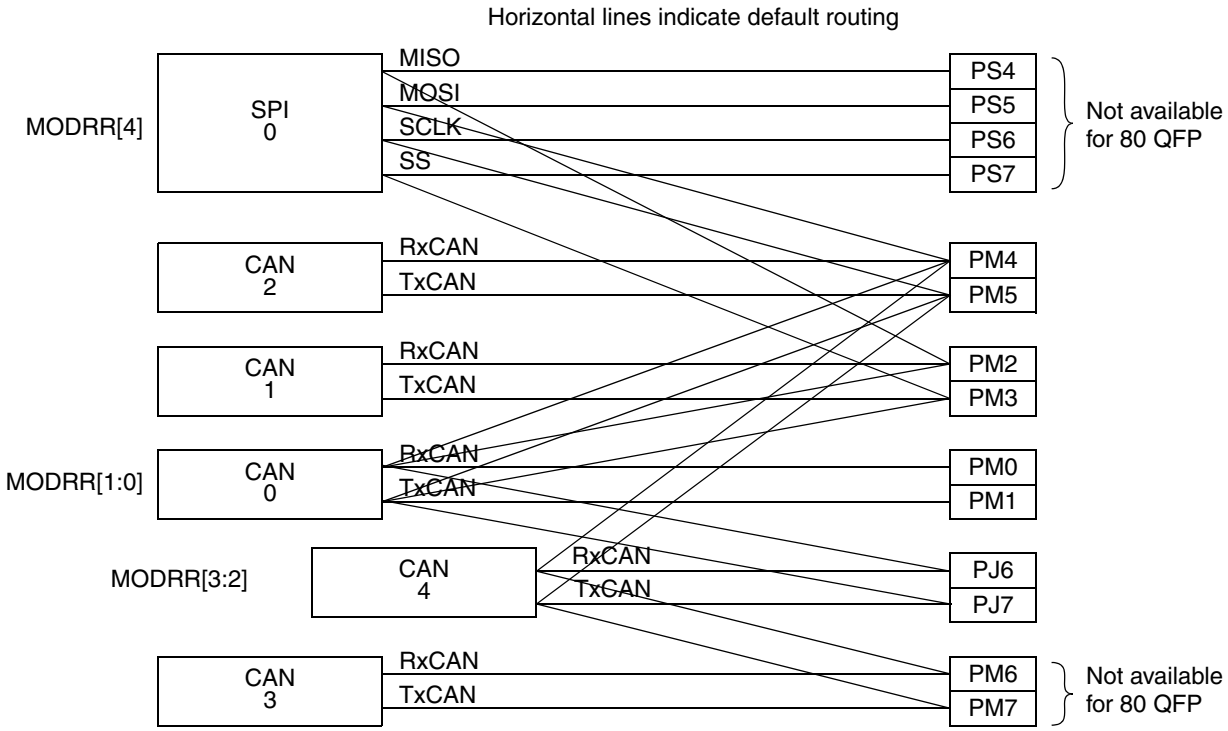


Figure 9. MODRR[4:0] Routing Options

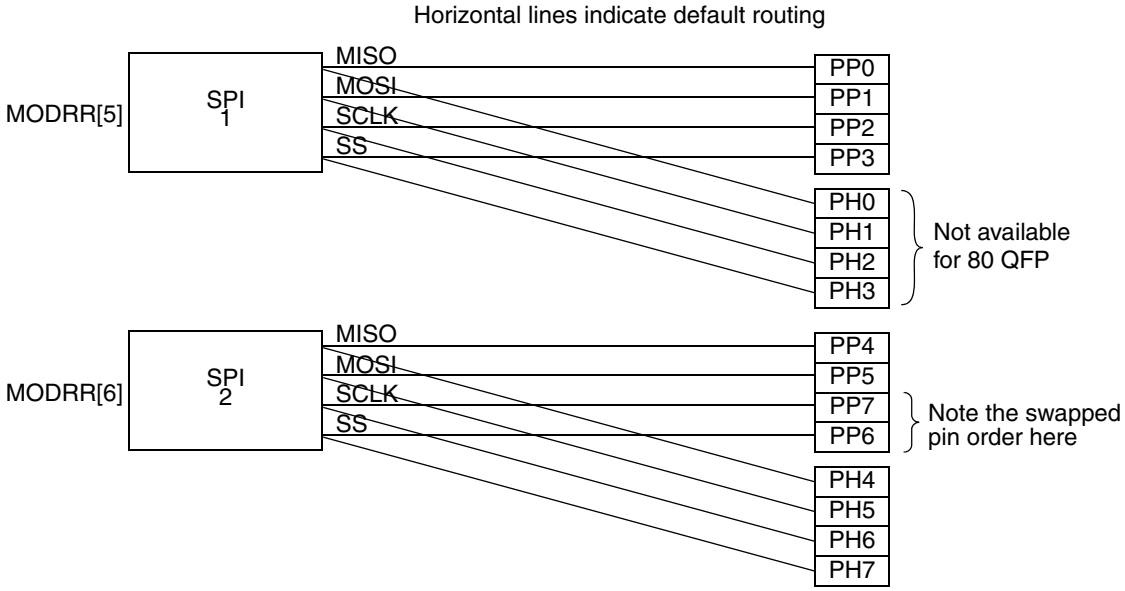


Figure 10. MODDR[6:5] Routing Options

14 Managing Unused Pins

If a general purpose input does not have a pull device enabled or is not driven externally, as the input approaches mid-rail (i.e. ~ 2.5 V) the digital input stage will be a linear region and a ‘cross-over current’ of ~ 2.5 mA can flow in the I/O stage. This ‘crossover current’ is on a the main I/O drive supply. This is

not a concern for the device itself but can directly impact the power supply demand and low power mode currents.

ATD inputs have a slightly different input stage and can be left open-circuit, although it is preferable to ground unused analog inputs to minimize pick-up of unwanted noise.

CAUTION

Leaving unused GPIO undefined is a common cause of unexpectedly high levels of Stop or Wait I_{DD}. This is not always obvious on every device or assembled module as this is dependent on small variations in the manufacturing process, operating temperature and voltage. Consequently, this is often not detected during development or qualification, but shows up in the application production test or in use where a greater number and range of operating conditions is encountered.

NOTE

MC9S12X D-family devices are often available in different packages (mostly 144-pin LQFP, 112-pin LQFP, and 80-pin QFP). When using a lower pin count variant, it is important to be aware that the I/O available on the largest pin count device is still present but unbonded and must be configured for low power modes. Either enable the internal pull devices or configure as outputs.

There are two approaches to managing unused GPIO: configuring them as inputs and configuring them as outputs; a combination of the two may be appropriate. In the event of code runaway, a GPIO port could be reconfigured unintentionally; protective software can reduce the impact of this, for either configuration, for example by periodically checking the PIM register and peripheral configurations.

14.1 Unused GPIO Configured as Outputs

This is a good solution as the pins will have a low impedance to rail. To minimize the impact of the ports being accidentally reconfigured as inputs, enable the internal pulldown devices (these will be active only when configured as inputs). It is recommended for devices in 80 QFP and 112 LQFP packages that all non-bonded pins be configured as outputs after reset, to avoid excess current draw.

14.2 Unused GPIO Configured as Inputs

Tie unused inputs to the supply rails, preferably with pull devices. In some cases, it may be an application requirement for unused inputs to be in a defined voltage state during reset; in such cases, all unused pins that default to high impedance must be pulled externally to a supply rail.

Several optional strategies for tying input pins are discussed below. Consideration should be given to the possibility of I/O conflict occurring, if unused input ports are reconfigured unintentionally as outputs, where:

- Two ports are connected together and might be driven with opposing polarities,

or

- A port is connected directly to a supply rail and might be driven to the opposite polarity from the supply rail.

In either case, the maximum I_{DD} specification for the pin will be exceeded, and the MCU will be damaged.

To minimize the impact of the ports being reconfigured accidentally as outputs, configure for reduced drive output and ensure that the port data registers match the polarity of the pull device(s) and supply rail connection.

There are several possible strategies:

1. Minimum risk, highest cost: pull each unused input pin to a supply rail with an individual pull resistor. This ensures no possibility of I/O conflict, as described above.
2. Highest risk, lowest cost: tie each unused input pin directly to a supply rail. This offers the highest risk of unintentional conflict with a supply rail.
3. High risk, low cost: common up all unused inputs to a single pull resistor to a supply rail. This offers the highest risk of unintentional conflict with another I/O pin.
4. Medium risk, medium cost: a better compromise of cost versus risk is to connect all unused inputs on each I/O port together, and to connect them to a separate pull resistor per I/O port.

NOTE

Some external peripherals tristate their outputs when disabled (including SPI interface lines, for example). Where this is the case, Stop and Wait I_{DD} can be minimized by enabling the internal pull devices on appropriate inputs while the external peripheral is disabled.

14.3 Debugging Low Power Mode I_{DD}

While unexpectedly high I_{DD} is often identified during development, it is not unknown for the following issues to be discovered only after a module is in full production.

Many such issues can be avoided by effective planning at the functional hardware design phase by creating an I/O check list identifying each I/O on the device (including unbonded ports) and defining how each of them will be configured by the hardware or software during run and low power modes. If the application allows for it, it is recommended to run a software check on I/O configuration before entering low power mode.

14.3.1 Undefined inputs

The most common reason for high Stop or Wait I_{DD} is where unused I/O, bonded or unbonded, are not driven or pulled to VDDX or VSSX. In this case some of the undefined inputs can generate unwanted I_{DD} as detailed in Section 14.

Any undefined digital input has the potential to float and draw unplanned current - this does not mean that it will in your prototype but it might under different conditions at some later date. To ensure that you get lowest Stop I_{DD} ALL unused ports (bonded and unbonded) must be set to output or pulled high or low.

14.3.1.1 Debugging a Floating Input

Where you think that an I/O might not be being configured (or maybe being erroneously re-configured as an undefined input by the software), one way to check this is to make up a symmetrical potential divider between VDDX and VSSX (using ~1M resistors), connect the centre point of the divider to a scope probe or volt meter and measure each of the I/O on the device in turn. All of the pin voltages should read ~VDDX or ~VSSX. A floating, high impedance input will be pulled mid-rail by the divider and measure ~VDDX/2.

14.3.2 Driving an Internal Pull Device

Another typical reason is where an external signal drives an input with an internal pull device enabled. Where the pin is driven to the opposite polarity of the pull device this can result in an undesired I_{DD} overhead while in a low power mode. An I/O check list can help identify this condition.

14.3.3 Tristate of a Signal Driving an Input With No Internal Pull Device Enabled

One less usual reason for increased I_{DD} is where an external device driving an input tristates its output when put into a low power mode. In this case, the software should enable an internal pull device as part of the low power entry routine, to prevent the pin floating, and then disable it in the low power recovery routine.

15 Output Drive Currents

The MC9S12X D-family has an “instantaneous maximum current single pin limit for all digital I/O” of ± 25 mA. If the current on a pin exceeds 25 mA peak at any time, the I/O structure may become damaged or suffer degradation. It is strongly recommended to stay well below the 25 mA limit, to avoid peaks exceeding this limit during switching.

The device maximum I/O current is limited by its power dissipation and will be application dependent. The $P_{IO} = \Sigma R_{DSON} \times I_{IO}^2$ term in the data sheet power dissipation calculation indicates the heating effect of the I/O current. Ensure that the total I/O power dissipation plus the internal device dissipation combined does not cause the device junction temperature to exceed the appropriate limit (for C, V, or M specification) in the application environment.

In production, I/O limits are tested by holding each pin at the V_{OL} and V_{OH} limits and measuring that the I/O current exceeds the I_{OL} and I_{OH} specification limits respectively (actually a measure of R_{DSON} of the output drivers).

From the V_{OHL} specification it can be seen that in the conduction range the I/O driver has a maximum $R_{DSON} = 0.8\ V / 10\ mA = 80\ \Omega$ (at the maximum temperature specified for the device). Typically, R_{DSON} will be lower than this; it will also be reduced at lower temperatures.

When driving high impedance loads such as logic devices the voltage dropped across the output R_{DSON} will be low, and the V_{OH} and V_{OL} levels will be much closer to the appropriate supply rail voltages.

16 I/O Injection Currents

All digital I/O pins are internally connected to V_{SSX}/V_{DDX}, V_{SSR}/V_{DDR}, or V_{SSA}/V_{DDA} via protection diodes. Taking an input pin above V_{DD5} or below V_{SS} by greater than a diode drop will cause current to flow to or from the internal device supply rails, via the protection diodes. This is known as “current injection” and is valid as long as the application limits the injected current to within specified limits.

16.1 Continuous (IICS)

Up to ± 2.5 mA per pin. MCU functionality is not guaranteed if any single pin exceeds this value.

16.2 Device Total (IICP)

The sum of the injected currents on all pins = $\Sigma \left| \text{IICP} \right|$. Maximum = 25 mA.

16.3 Instantaneous Maximum (ID, IDL)

Short duration injection current maximum ± 25 mA (on any single pin). MCU functionality is not guaranteed but the device will not be damaged. The IICP limit must be respected.

Keep current injection on any individual input less than IICS (2.5 mA) and the total device current injection less than IICP (25 mA). If any pin transient exceeds IICS, MCU functionality is not guaranteed but, as long as the individual pin current and the total device current remains less than 25 mA, the device will not be damaged.

NOTE

The power supply must maintain regulation within operating V_{DD} or V_{DDX} range during instantaneous and operating maximum current conditions. If positive injection current ($V_{IN} > V_{DD}$ or $V_{IN} > V_{DDX}$) is greater than I_{DD} or I_{DDX}, the injection current may flow out of V_{DD}/V_{DDX} and could result in the external power supply going out of regulation. Make sure that the external V_{DD}/V_{DDX} load will shunt current greater than maximum injection current. This will be the greatest risk when the MCU is not consuming power. Examples are: when no system clock is present, as in Stop mode; when the clock rate is very low, which would reduce overall power consumption.

16.4 Analog Input Considerations

In addition to the above constraints, current injection on ATD inputs may cause additional errors in conversions of adjacent pins. A portion of the injected current will also be picked up by the adjacent channels (coupling ratio K), generating an error voltage proportional to the source resistance of the input being converted.

The additional input voltage error on the converted channel can be calculated as $V_{ERR} = K * R_S * I_{INJ}$, with I_{INJ} being the sum of the currents injected into the two pins adjacent to the converted channel.

There are two coupling ratios specified:

- Kp = 10^{-4} for positive current injection, where the input is taken above V_{DDA}
- Kn = 10^{-2} for negative current injection, where the input is taken below V_{SSA}.

17 Connecting Capacitors Directly to Output Pins

1. Avoid connecting capacitors directly to output pins (Figure 11) in an attempt to prevent system noise reaching the MCU pin. The capacitor will appear as low impedance to transitions of the output, resulting in fast rising pulses of current and EMC noise. A simple solution is to add some series resistance to the MCU side of the filter. This will increase the rise time of the signal on the capacitor (due to the time constant of the increased effective output impedance and the filter capacitor) and reduce the EMC generated.
2. Connecting a large capacitor directly to an output pin (to create a long time delay, for example) can result in exceeding the maximum I_{DD} specification for the pin and damage to the MCU. Ensure that there is enough series resistance from the MCU to any capacitor to limit the peak current to << 25 mA.

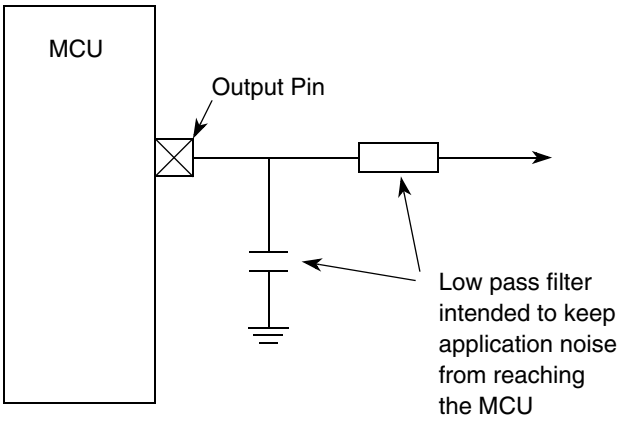


Figure 11. Connecting Capacitors Directly to Output Pins

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: AN3262
Rev. 1
10/2009

freescale™
semiconductor