Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
采购部

Register Map: Section 4.2

ZL40255 SmartBuffer™

3-Output Programmable Fanout Buffer with Multi-Format I/O and Dividers

Data Sheet

April 2016

Features

- **Four Input Clocks**
  - One crystal/CMOS input
  - Two differential/CMOS inputs
  - One single-ended/CMOS input
  - Any input frequency up to 1035MHz (up to 300MHz for CMOS)
  - Clock selection by pin or register control
- **Up to 3 Differential Outputs (Up to 6 CMOS)**
  - Output frequencies are any integer divisor up to 2^{32} of the input frequency (CMOS 250MHz max)
  - Each output has independent dividers
  - Low additive jitter <200fs RMS (12kHz-20MHz, for input frequencies ≥100MHz)
  - Outputs are CML or 2xCMOS, can interface to LVDS, LVPECL, HSTL, SSTL and HCSL
  - In 2xCMOS mode, the P and N pins can be different frequencies (e.g. 125MHz and 25MHz)*
  - Per-output supply pin with CMOS output voltages from 1.5V to 3.3V
  - Precise output alignment circuitry and per-output phase adjustment*
  - Per-output enable/disable and glitchless start/stop (stop high or low)*

Ordering Information

ZL40255LDG1	32 Pin QFN	Trays
ZL40255LDF1	32 Pin QFN	Tape and Reel

Matte Tin

Package size: 5 x 5 mm

-40°C to +85°C

- **General Features**
  - Automatic self-configuration at power-up from internal EEPROM; up to four configurations, pin-selectable
  - Crystal interface for frequency synthesis up to 60MHz
  - Four general-purpose I/O pins, each with many status and control options
  - SPI or I^2C processor Interface
  - Tiny 5x5mm QFN package

Applications

- Frequency synthesis up to 60MHz
- Fanout up to 1035MHz
- Format conversion, frequency division, and skew adjustment in a wide variety of equipment types

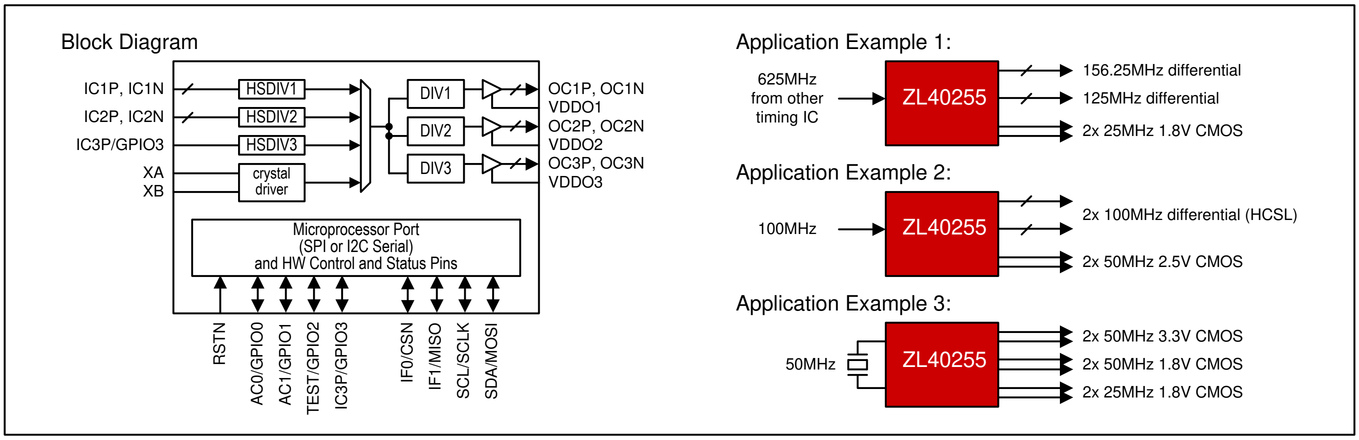


Figure 1 - Functional Block Diagram and Application Examples

* Some features require a higher-frequency input clock and enabling the output dividers.

Microsemi Corporation

Table of Contents

List of Figures

List of Tables

1. Pin Diagram

The device is packaged in a 5x5mm 32-pin QFN.

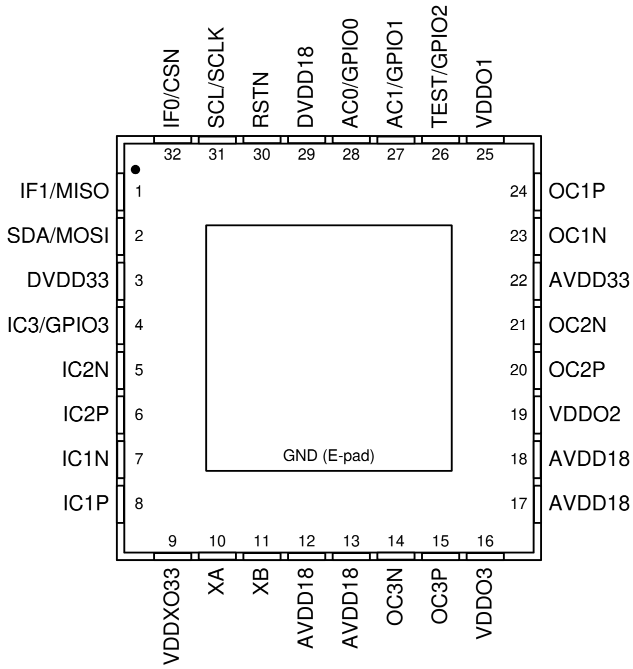


Figure 2 - Pin Diagram

2. Pin Descriptions

All device inputs and outputs are LVCMOS unless described otherwise. The Type column uses the following symbols: I – input, I_{PU} – input with 50kΩ internal pullup resistor, O – output, A – analog, P – power supply pin. All GPIO and SPI/I^2C interface pins have Schmitt-trigger inputs and have output drivers that can be disabled (high impedance).

Table 1 - Pin Descriptions

Pin #	Name	Type	Description
8 7 6 5 4	IC1P IC1N IC2P IC2N IC3P/GPIO3	I I I I I/O	**Input Clock Pins** Differential or Single-ended signal format. Programmable frequency. *Differential:* See Table 9 for electrical specifications, and see Figure 13 for recommended external circuitry for interfacing these differential inputs to LVDS, LVPECL, CML or HSCL output pins on neighboring devices. *Single-ended:* For input signal amplitude >2.5V, connect the signal directly to ICxP pin. For input signal amplitude ≤2.5V, AC-coupling the signal to ICxP is recommended. Connect the N pin to a capacitor (0.1µF or 0.01µF) to VSS. As shown in Figure 13, the ICxP and ICxN pins are internally biased to approximately 1.3V. Treat the ICxN pin as a sensitive node; minimize stubs; do not connect to anything else including other ICxN pins. *Unused:* Set ICEN.ICxEN=0. The ICxP and ICxN pins can be left floating. Note that the IC3N pin is not bonded out. A differential signal can be connected to IC3P by AC-coupling the POS trace to IC3P and terminating the signal on the driver side of the coupling cap. If not needed as an input clock pin, IC3P can behave as general-purpose I/O pin GPIO3, which is configured by GPIOCR2. Its state is indicated in GPIOSR.
10 11	XA XB	A / I	**Crystal or Input Clock Pins** *Crystal:* MCR1.XAB=01. An on-chip crystal driver circuit is designed to work with an external crystal connected to the XA and XB pins. See section 3.3 for crystal characteristics and recommended external components. *Input Clock:* MCR1.XAB=10. An external local oscillator or clock signal can be connected to the XA pin. The XB pin must be left unconnected.
24 23 20 21 15 14	OC1P OC1N OC2P OC2N OC3P OC3N	O	**Output Clock Pins** CML, HSTL or 1 or 2 CMOS. Programmable frequency and drive strength. See Table 10 and Figure 15 for electrical specifications and recommended external circuitry for interfacing to LVDS, LVPECL or CML input pins on neighboring devices. See Table 11 for electrical specifications for interfacing to CMOS and HSTL inputs on neighboring devices. See Figure 16 for recommended external circuitry for interfacing to HCSL inputs on neighboring devices.
30	RSTN	I_{PU}	**Reset (Active Low)** When this global asynchronous reset is pulled low, all internal circuitry is reset to default values. The device is held in reset as long as RSTN is low. See section 3.9.

Table 1 - Pin Descriptions (continued)

Pin #	Name	Type	Description
28 27	AC0/GPIO0 AC1/GPIO1	I/O	**Auto-Configure [1:0] / General Purpose I/O 0 and 1** *Auto Configure:* On the rising edge of RSTN these pins behave as AC[1:0] and specify one of the configurations stored in EEPROM. See section 3.2. *General-Purpose I/O:* After reset these pins are GPIO0 and GPIO1. GPIOCR1 configures the pins. Their states are indicated in GPIOSR.
26	TEST/GPIO2	I/O	**Factory Test / General Purpose I/O 2** *Factory Test:* On the rising edge of RSTN the pin behaves as TEST. Factory test mode is enabled when TEST is high. For normal operation TEST must be low on the rising edge of RSTN. *General-Purpose I/O:* After reset this pin is GPIO2. GPIOCR2 configures the pin. It state is indicated in GPIOSR.
32	IF0/CSN	I/O	**Interface Mode 0 / SPI Chip Select (Active Low)** *Interface Mode:* On the rising edge of RSTN the pin behaves as IF0 and, together with IF1, specifies the interface mode for the device. See section 3.2. *SPI Chip Select:* After reset this pin is CSN. When the device is configured as a SPI slave, an external SPI master must assert (low) CSN to access device registers.
31	SCL/SCLK	I/O	**I^{2}C Clock / SPI Clock** *I^{2}C Clock:* When the device is configured as an I^{2}C slave, an external I^{2}C master must provide the I^{2}C clock signal on the SCL pin. *SPI Clock:* When the device is configured as a SPI slave, an external SPI master must provide the SPI clock signal on SCLK.
1	IF1/MISO	I/O	**Interface Mode 1 / SPI Master-In-Slave-Out** *Interface Mode:* On the rising edge of RSTN the pin behaves as IF1 and, together with IF0, specifies the interface mode for the device. See section 3.2. *SPI MISO:* After reset this pin is MISO. When the device is configured as a SPI slave, the device outputs data to an external SPI master on MISO during SPI read transactions.
2	SDA/MOSI	I/O	**I^{2}C Data / SPI Master-Out-Slave-In** *I^{2}C Data:* When the device is configured as an I^{2}C slave, SDA is the bidirectional data line between the device and an external I^{2}C master. *SPI MOSI:* When the device is configured as a SPI slave, an external SPI master sends commands, addresses and data to the device on MOSI.
12 13 17 18	AVDD18	P	**Analog Power Supply.** 1.8V ±5%.
22	AVDD33	P	**Analog Power Supply.** 3.3V ±5%.
29	DVDD18	P	**Digital Power Supply.** 1.8V ±5%.
3	DVDD33	P	**Digital Power Supply.** 3.3V ±5%.
25	VDDO1	P	**Output OC1 Power Supply.** 1.5V to 3.3V ±5%.
19	VDDO2	P	**Output OC2 Power Supply.** 1.5V to 3.3V ±5%.

Pin #	Name	Type	Description
16	VDDO3	P	**Output OC3 Power Supply.** 1.5V to 3.3V ±5%.
9	VDDXO33	P	**Analog Power Supply for Crystal Driver Circuitry.** 3.3V ±5%.
E-pad	VSS	P	**Ground.** 0 Volts.

3. Functional Description

3.1 Device Identification

The 12-bit read-only ID field and the 4-bit revision field are found in the ID1 and ID2 registers. Contact the factory to interpret the revision value and determine the latest revision.

3.2 Pin-Controlled Automatic Configuration at Reset

The device configuration is determined at reset (i.e. on the rising edge of RSTN) by the signal levels on five device pins: TEST/GPIO2, AC1/GPIO1, AC0/GPIO0, IF1/MISO and IF0/CSN. For each of these pins, the first name (TEST, AC1, AC0, IF1, IF0) indicates their function when they are sampled by the rising edge of the RSTN pin. The second name refers to their function after reset. The values of these pins are latched into the CFGSR register when RSTN goes high. To ensure the device properly samples the reset values of these pins, the following guidelines should be followed:

1. Any pullup or pulldown resistors used to set the value of these pins at reset should be 1kΩ.
2. RSTN must be asserted at least as long as specified in section 3.9.

The hardware configuration pins are grouped into three sets:

1. TEST - Manufacturing test mode
2. IF[1:0] – Microprocessor interface mode and I^{2}C address
3. AC[1:0] – Auto-configuration from EEPROM

The TEST pin selects manufacturing test modes when TEST=1 (the AC[1:0] pins specify the test mode). TEST=1 and AC[1:0]=00 configures the part so that production SPI EEPROM programmers can program the internal EEPROM (see section 3.11.3).

The IF[1:0] pins specify the processor interface mode and the I^{2}C slave address.

IF1	IF0	Processor Interface
0	0	I^{2}C, slave address 10110 00
0	1	I^{2}C, slave address 10110 01
1	0	I^{2}C, slave address 10110 10
1	1	SPI Slave

The AC[1:0] pins specify which of four device configurations in the EEPROM to execute after reset.

AC1	AC0	Auto Configuration
0	0	Configuration 0
0	1	Configuration 1
1	0	Configuration 2
1	1	Configuration 3

For more information about auto-configuration from EEPROM see section 3.11.1.

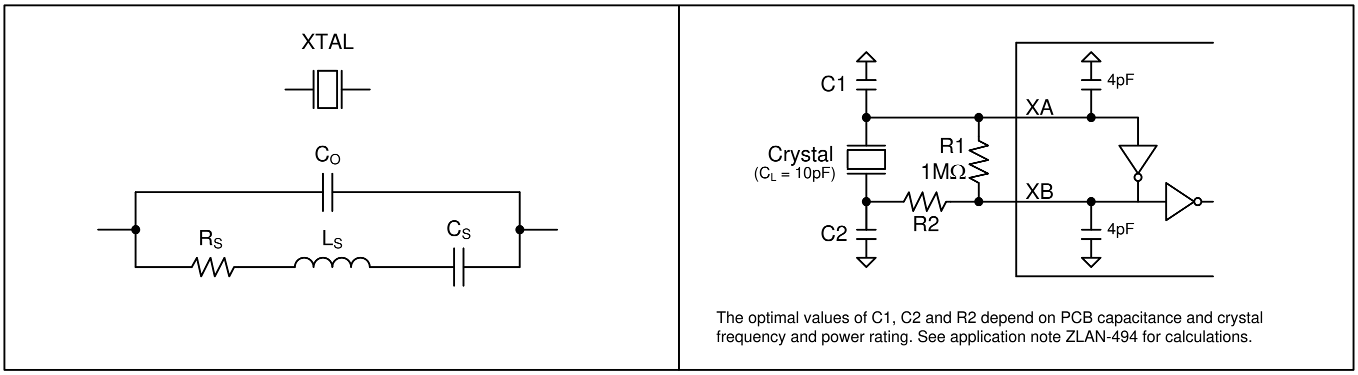
3.3 External Crystal and On-Chip Driver Circuit

The on-chip crystal driver circuit is designed to work with a fundamental mode, AT-cut crystal resonator. See Table 2 for recommended crystal specifications. To enable the crystal driver, set MCR1.XAB=01.

See Figure 3 for the crystal equivalent circuit and the recommended external capacitor connections. To achieve a crystal load (C_L) of 10pF, an external 16pF is placed in parallel with the 4pF internal capacitance of the XA pin, and an external 16pF is placed in parallel with the 4pF internal capacitance of the XB pin. The crystal then sees a load of 20pF in series with 20pF, which is 10pF total load. Note that the 16pF capacitance values in Figure 3 include all capacitance on those nodes. If, for example, PCB trace capacitance between crystal pin and IC pin is 2pF then 14pF capacitors should be used to make 16pF total.

The crystal, traces, and two external capacitors should be placed on the board as close as possible to the XA and XB pins to reduce crosstalk of active signals into the oscillator. Also no active signals should be routed under the crystal circuitry.

Note: Crystals have temperature sensitivies that can cause frequency changes in response to ambient temperature changes. In applications where significant temperature changes are expected near the crystal, it is recommended that the crystal be covered with a thermal cap, or an external XO or TCXO should be used instead.



The optimal values of C1, C2 and R2 depend on PCB capacitance and crystal frequency and power rating. See application note ZLAN-494 for calculations.

Figure 3 - Crystal Equivalent Circuit / Recommended Crystal Circuit

Table 2 - Crystal Selection Parameters

Parameter		Symbol	Min.	Typ.	Max.	Units
Crystal oscillation frequency[1]		f_{OSC}	25		60	MHz
Shunt capacitance		C_O		2	5	pF
Load capacitance		C_L		10		pF
Equivalent series resistance (ESR)[2]	f_{OSC} < 40MHz	R_S			60	Ω
	f_{OSC} > 40MHz	R_S			50	Ω
Maximum crystal drive level			100			µW

Note 1: Higher frequencies give lower output jitter, all else being equal.

Note 2: These ESR limits are chosen to constrain crystal drive level to less than 100µW. If the crystal can tolerate a drive level greater than 100µW then proportionally higher ESR is acceptable.

3.4 Input Signal Format Configuration

Input clocks IC1, IC2 and IC3 are enabled by setting the enable bits in the ICEN register. The power consumed by a differential receiver is shown in Table 6. The electrical specifications for these inputs are listed in Table 9. Each input clock can be configured to accept nearly any differential signal format by using the proper set of external components (see Figure 13). To configure these differential inputs to accept single-ended CMOS signals, connect the single-ended signal to the ICxP pin, and connect the ICxN pin to a capacitor (0.1µF or 0.01µF) to VSS. Each ICxP and ICxN pin is internally biased to approximately 1.3V. If an input is not used, both ICxP and ICxN pins can be left floating. Note that the IC3N pin is not present. A differential signal can be connected to IC3P by AC-coupling the POS trace to IC3P and terminating the signal on the driver side of the coupling cap. If not needed as an input clock pin, IC3P can behave as general-purpose I/O pin GPIO3.

3.5 Input Selection

The input to the device can be controlled by a GPIO pin or by the SRCCR3.INMUX register field. When SRCCR3.EXTSW=0, the SRCCR3.INMUX register field controls the input mux.

When SRCCR3.EXTSW=1, a GPIO pin controls the input mux. When the GPIO pin is low, the mux selects the input specified by SRCCR3.INMUX. When the GPIO pin is high, the mux selects the input specified by SRCCR3.ALTMUX. MCR2.EXTSS specifies which GPIO pin controls this behavior.

The polarity of an ICx input signal can be inverted by setting ICxCR1.POL.

Input clock frequencies above 850MHz must be divided by 2 using the input high-speed dividers configured by ICxCR1.HSDIV.

3.6 Output Clock Configuration

The device has three output clock signal pairs. Each output has individual divider, enable and signal format controls. In CMOS mode each signal pair can become two CMOS outputs, allowing the device to have up to six output clock signals. Also in CMOS mode, the OCxN pin can have an additional divider allowing the OCxN frequency to be an integer divisor of the OCxP frequency (example: OC3P 125MHz and OC3N 25MHz). The outputs can be aligned relative to each other and relative to an input signal, and the phases of output signals can be adjusted dynamically with high resolution and infinite range.

3.6.1 Output Enable, Signal Format, Voltage and Interfacing

To use an output, the output driver must be enabled by setting OCxCR2.OCSF≠0, and the per-output dividers must be enabled by setting the appropriate bit in the OCEN register. The per-output dividers include the medium-speed divider, the low-speed divider and the associated phase adjustment/alignment circuitry and start/stop logic.

Using the OCxCR2.OCSF register field, each output pair can be disabled or configured as a CML output, an HSTL output, or one or two CMOS outputs. When an output is disabled it is high impedance, and the output driver is in a low-power state. In CMOS mode, the OCxN pin can be disabled, in phase or inverted vs. the OCxP pin. In CML mode the normal 800mV V_{OD} differential voltage is available as well as a half-swing 400mV V_{OD}. All of these options are specified by OCxCR2.OCSF. The clock to the output driver can inverted by setting OCxCR2.POL=1. The CMOS/HSTL output driver can be set to any of four drive strengths using OCxCR2.DRIVE.

Each output has its own power supply pin to allow CMOS or HSTL signal swing from 1.5V to 3.3V for glueless interfacing to neighboring components. If OCSF is set to HSTL mode then a 1.5V power supply voltage should be used to get a standards-compliant HSTL output. Note that differential (CML) outputs must have a power supply of 3.3V.

The differential outputs can be easily interfaced to LVDS, LVPECL, CML, HCSL, HSTL and other differential inputs on neighboring ICs using a few external passive components. See Figure 15 for examples.

3.6.2 Output Frequency Configuration

Each output has two output dividers, a 7-bit medium-speed divider (OCxCR1.MSDIV) and a 25-bit low-speed output divider (LSDIV field in the OCxDIV registers). These dividers are in series, medium-speed divider first then output divider. These dividers produce signals with 50% duty cycle for all divider values including odd numbers. The low-speed divider can only be used if the medium-speed divider is used (i.e. OCxCR1.MSDIV>0).

Since each output has its own independent dividers, the device can output families of related frequencies that have an input frequency as a common multiple. For example, for Ethernet clocks, a 625MHz input clock can be divided by four for one output to get 156.25MHz, divided by five for another output to get 125MHz, and divided by 25 for another output to get 25MHz. Similarly, for SDH/SONET clocks, a 622.08MHz input clock can be divided by 4 to get 155.52MHz, by 8 to get 77.76MHz, by 16 to get 38.88MHz or by 32 to get 19.44MHz.

Two Different Frequencies in 2xCMOS Mode

When an output is in 2xCMOS mode it can be configured to have the frequency of the OCxN clock be an integer divisor of the frequency of the OCxP clock. Examples of where this can be useful:

- 125MHz on OCxP and 25MHz on OCxN for Ethernet applications
- 77.76MHz on OCxP and 19.44MHz on OCxN for SONET/SDH applications
- 25MHz on OCxP and 1Hz (i.e. 1PPS) on OCxN for telecom applications with Synchronous Ethernet and IEEE1588 timing

An output can be configured to operate like this by setting the LSDIV value in the OCxDIV registers to OCxP_freq / OCxN_freq - 1 and setting OCxCR3.LSSEL=0 and OCxCR3.NEGLSD=1. Here are some notes about this dual-frequency configuration option:

- In this mode only the medium speed divider is used to create the OCxP frequency. The low-speed divider is then used to divide the OCxP frequency down to the OCxN frequency. This means that the lowest OCxP frequency is the input high-speed divider frequency divided by 128.
- An additional constraint is that the medium-speed divider must be configured to divide by 6 or more (i.e. must have OCxCR1.MSDIV≥5).

3.6.3 Output Duty Cycle Adjustment

For output frequencies less than or equal to 141.666MHz, the duty cycle of the output clock can be modified using the OCxDC.OCDC register field. This behavior is only available when MSDIV>0 and LSDIV > 1. When OCDC = 0 the output clock is 50%. Otherwise the clock signal is a pulse with a width of OCDC number of MSDIV output clock periods. The range of OCDC can create pulse widths of 1 to 255 MSDIV output clock periods. When OCxCR2.POL=0, the pulse is high and the signal is low the remainder of the cycle. When POL=1, the pulse is low and the signal is high the remainder of the cycle.

Note that duty cycle adjustment is done in the low-speed divider. Therefore when OCxCR3.LSSEL=0 the duty cycle of the output is not affected. Also, when a CMOS output is configured with OCxCR3.LSSEL=0 and OCxCR3.NEGLSD=1, the OCxN pin has duty cycle adjustment but the OCxP pin does not. This allows a higher-speed 50% duty cycle clock signal to be output on the OCxP pin and a lower-speed frame/phase/time pulse (e.g. 2kHz, 8kHz or 1PPS) to be output on the OCxN pin at the same time.

An output configured for CMOS or HSTL signal format should not be configured to have a duty cycle with high time shorter than 2ns or low time shorter than 2ns.

3.6.4 Output Phase Adjustment and Phase Alignment

The device has flexible, high-resolution tools for managing the phases of the output clocks relative to one another. The key register fields for this are found in the PACR1 and PACR2 global configuration registers and the per-output OCxPH register.

Phase alignment and phase adjustment are done in the medium-speed dividers. Resoution is 0.5 periods (also known as unit intervals or UI) of the input clock after the high-speed divider. For example, for an input frequency of 800MHz, resolution is 625ps.

3.6.4.1 Phase Adjustment

A phase adjustment is a phase change for an output relative to that output's most recent phase. To cause the device to perform phase adjustment of an output clock, set PACR1.MODE=1, set OCxCR1.PHEN=1 to enable the output for phase adjustment, and write the phase adjustment amount to the output's OCxPH register. Then an arm/trigger methodology is used to cause the phase adjustment to happen.

The arm step tells the device that it is enabled to perform the phase adjustment when it sees the trigger stimulus. The source of the arm signal is specified by PACR2.ARMSRC. Options include the 0-to-1 transition of the PACR1.ARM bit or a transition on one of the GPIO pins.

The source of the trigger signal is specified by PACR2.TRGSRC. Options include 0-to-1 transition of the PACR1.TRIG bit or a transition on one of the GPIO pins. The trigger signal can be inverted by setting PACR1.TINV. With TINV=1, the same GPIO signal can arm on one edge and trigger on the opposite edge.

Any combination of outputs can be phase adjusted by the same trigger, and each output can be adjusted by a different amount. Only outputs with OCxCR1.PHEN=1 and OCxPH.PHADJ≠0 have their phases adjusted.

There are a few constraints on the range of possible phase adjustments. These have to do with the output's medium-speed divider value.

1) Phase adjustment is not available unless OCxCR1.MSDIV>0.
2) The largest negative phase adjustment magnitude in input HSDIV periods is:
   If OCxCR1.MSDIV is odd: (OCxCR1.MSDIV – 1) / 2
   If OCxCR1.MSDIV is even: (OCxCR1.MSDIV – 2) / 2
3) The largest positive phase adjustment in input HSDIV periods is:
   If OCxCR1.MSDIV is odd: (127 – OCxCR1.MSDIV) / 2
   If OCxCR1.MSDIV is even: (128 – OCxCR1.MSDIV) / 2

The implications of constraints 2) and 3) are shown in this table:

OCxCR1.MSDIV	**Largest Negative Phase Adjust,** HSDIV periods	**Largest Positive Phase Adjust,** HSDIV periods	Notes
1 or 2	0	63	no negative adjustment
3 or 4	1	62	
5 or 6	2	61	
…	…	…	
123 or 124	61	2	
125 or 126	62	1	
127	63	0	no positive adjustment

During a phase adjustment the MSDIV output period is changed for one period. The MSDIV output signal during that period will have longer high time (unless inverted) during a positive phase adjustment and shorter high time (unless inverted) during a negative phase adjustment. With negative phase adjustments care must be taken to not shorten the high time of the output clock signal to be too short for the components that receive the clock. There are several possible ways to avoid this issue including: (1) using small negative adjustments such as -0.5UI repeatedly instead of one larger negative adjustment, (2) using positive adjustments to "wrap around" to the desired negative adjustment, or (3) holding the components that receive the clock in reset during the phase adjustment.

An armed phase adjustment can be canceled before the trigger occurs by setting the PACR1.RST bit.

The PASR register has real-time status bits indicating whether a phase adjustment is armed and waiting for a trigger (ARMED bit) or in progress (BUSY bit).

Example: +1.0 HSDIV period phase adjustment for output OC1 using ARM and TRIG register bits:

OC1CR1.PHEN=1	(Enable phase adjust on OC1)
OC1PH.PHADJ=00000010	(Specify +1.0 HSDIV period phase adjustment)
PACR1.MODE=1	(Phase adjustment mode)
PACR2.ARMSRC=0001	(arm signal is PACR1.ARM bit)
PACR2.TRGSRC=0000	(trigger signal is PACR1.TRIG bit)
PACR1.RST=1	(reset phase adjust/align state machine after changing ARMSRC)
PACR1.ARM=1	(arm for phase adjust)
PACR1.TRIG=1	(do the phase adjust: add +1.0 UI to output phase)

repeat the next two writes as needed:

PACR1.ARM=1 .TRIG=0	(clear the trigger bit and arm again)
PACR1.TRIG=1	(add +1.0 UI to output phase again)

3.6.4.2 Phase Alignment, Output-to-Output

A phase alignment is a special case of phase adjustment where the MSDIV and LSDIV dividers for all participating outputs are reset just before the phase adjustment occurs. For output-to-output alignment the trigger can be the PACR1.TRIG bit.

To avoid glitches (i.e. “runt pulses”) on the output clock it is possible to manually stop the output(s), before triggering the phase alignment, and then restart the output(s) after the alignment (See section 3.6.5).

When aligning outputs, it is important to note that, by default, the phase of outputs configured as HSTL format or “two CMOS, OCxP inverted vs. OCxN” format is opposite that of CML outputs. For example, consider the case where OC1 is 100MHz CML format and OC2 is 100MHz HSTL format. When OC1 and OC2 are aligned then OC2N is high when OC1P is high. The polarity bit OCxCR2.POL can be used to change this as needed.

There are several rules when alignment is enabled for multiple outputs:

- All participating outputs must come from the same high-speed divider
- All outputs that use both medium-speed and low-speed divider must have the same MSDIV value, the same LSDIV value and PHADJ=0. Subsequent phase adjustment(s) can be used to move the output(s) to other phase(s).
- All outputs that only use medium-speed divider can have PHADJ values smaller than the period of the highest output frequency among them.
- When some outputs use only medium-speed divider and other outputs use both medium-speed and low-speed divider, all MSDIV values must be the same, and those output using low-speed divider must have PHADJ=0.

Contact Microsemi Timing Applications Support for help with alignment scenarios that don’t meet the rules listed above.

Example: OC1-to-OC2 alignment (+3.5 HSDIV UI offset):

OC1CR1.PHEN=1	(Enable phase adjust on OC1)
OC2CR1.PHEN=1	(Enable phase adjust on OC2)
OC1PH.PHADJ=00000000	(0.0UI)
OC2PH.PHADJ=00000111	(+3.5UI)
PACR1.MODE=0	(Phase alignment mode)
PACR2.ARMSRC=0001	(arm signal is PACR1.ARM bit)
PACR2.TRGSRC=0000	(trigger signal is PACR1.TRIG bit)
PACR1.RST=1	(reset phase adjust/align state machine after changing ARMSRC, TRGSRC)
PACR1.ARM=1	(arm for phase alignment)
PACR1.TRIG=1	(trigger phase alignment)
PACR1.TRIG=0	(clear trigger bit)

3.6.5 Output Clock Start and Stop

Output clocks can be stopped high or low. One use for this behavior is to ensure “glitchless” output clock operation while the output is reconfigured or phase aligned with some other signal.

Each output has an OCxSTOP register with fields to control this behavior. The OCxSTOP.MODE field specifies whether the output clock signal stops high, stops low, or or does not stop. The OCxSTOP.SRC field specifies the source of the stop signal. Options include the OCxSTOP.STOP bit, assertion of one of the GPIO pins, and the arming of a phase adjustment (which is indicate by PASR.ARMED).

When the stop mode is Stop High (OCxSTOP.MODE=01) and the stop signal is asserted, the output clock is stopped after the next rising edge of the output clock. When the stop mode is Stop Low (OCxSTOP.MODE=10) and the stop signal is asserted, the output clock is stopped after the next falling edge of the output clock. Internally the clock signal continues to toggle while the output is stopped. When the stop signal is deasserted, the output clock resumes on the opposite edge that it stopped on. Low-speed output clocks can take long intervals before being stopped after the stop signal goes active. For example, a 1 Hz output could take up to 1 second to stop.

OCxCR1.MSDIV must be > 0 for this function to operate since MSDIV=0 bypasses the start-stop circuits. Note that when OCxCR3.NEGLSD=1 the start-stop logic is bypassed for the OCxN pin, and OCxN may not start/stop without glitches.

When OCxCR2.POL=1 the output stops on the opposite polarity that is specified by the OCxSTOP.MODE field.

When OCxCR2.STOPDIS=1 the output driver is disabled (high impedance) while the output clock is stopped.

Each output has a status register (OCxSR) with several stop/start status bits. The STOPD bit is a real-time status bit indicating stopped or not stopped. The STOPL bit is a latched status bit that is set when the output clock has stopped. The STARTL bit is a latched status bit that is set when the output clock has started.

3.7 Microprocessor Interface

The device can communicate over a SPI interface or an I^{2}C interface.

Section 3.2 describes reset pin settings required to configure the device for these interfaces.

3.7.1 SPI Slave

The device can present a SPI slave port on the CSN, SCLK, MOSI, and MISO pins. SPI is a widely used master/slave bus protocol that allows a master and one or more slaves to communicate over a serial bus. SPI masters are typically microprocessors, ASICs or FPGAs. Data transfers are always initiated by the master, which also generates the SCLK signal. The device receives serial data on the MOSI (Master Out Slave In) pin and transmits serial data on the MISO (Master In Slave Out) pin. MISO is high impedance except when the device is transmitting data to the bus master.

Bit Order. The register address and all data bytes are transmitted most significant bit first on both MOSI and MISO.

Clock Polarity and Phase. The device latches data on MOSI on the rising edge of SCLK and updates data on MISO on the falling edge of SCLK. SCLK does not have to toggle between accesses, i.e., when CSN is high.

Device Selection. Each SPI device has its own chip-select line. To select the device, the bus master drives its CSN pin low.

Command and Address. After driving CSN low, the bus master transmits an 8-bit command followed by a 16-bit register address. The available commands are shown below.

Table 3 – SPI Commands

Command	Hex	Bit Order, Left to Right
Write Enable	0x06	0000 0110
Write	0x02	0000 0010
Read	0x03	0000 0011
Read Status	0x05	0000 0101

Read Transactions. The device registers are accessible when EESEL=0. The internal EEPROM memory is accessible when EESEL=1. After driving CSN low, the bus master transmits the read command followed by the 16-bit address. The device then responds with the requested data byte on MISO, increments its address counter, and prefetches the next data byte. If the bus master continues to demand data, the device continues to provide the data on MISO, increment its address counter, and prefetch the following byte. The read transaction is completed when the bus master drives CSN high. See Figure 4.

Register Write Transactions. The device registers are accessible when EESEL=0. After driving CSN low, the bus master transmits the write command followed by the 16-bit register address followed by the first data byte to be written. The device receives the first data byte on MOSI, writes it to the specified register, increments its internal address register, and prepares to receive the next data byte. If the master continues to transmit, the device continues to write the data received and increment its address counter. The write transaction is completed when the bus master drives CSN high. See Figure 6.

EEPROM Writes The EEPROM memory is accessible when EESEL=1. After driving CSN low, the bus master transmits the write enable command and then drives CSN high to set the internal write enable latch. The bus master then drives CSN low again and transmits the write command followed by the 16-bit address followed by the first data byte to be written. The device first copies the page to be written from EEPROM to its page buffer. The device then receives the first data byte on MOSI, writes it to its page buffer, increments its internal address register, and prepares to receive the next data byte. If the master continues to transmit, the device continues to write the data received to its page buffer and continues to increment its address counter. The address counter rolls over at the 32-byte page boundary (i.e. when the five least-significant address bits are 11111). When the bus master drives CSN high, the device transfers the data in the page buffer to the appropriate page in the EEPROM memory. See Figure 5 and Figure 6.

EEPROM Read Status. After the bus master drives CSN high to end an EEPROM write command, the EEPROM memory is not accessible for up to 5ms while the data is transferred from the page buffer. To determine when this transfer is complete, the bus master can use the Read Status command. After driving CSN low, the bus master transmits the Read Status command. The device then responds with the status byte on MISO. In this byte, the least significant bit is set to 1 if the transfer is still in progress and 0 if the transfer has completed.

Early Termination of Bus Transactions. The bus master can terminate SPI bus transactions at any time by pulling CSN high. In response to early terminations, the device resets its SPI interface logic and waits for the start of the next transaction. If a register write transaction is terminated prior to the SCLK edge that latches the least significant bit of a data byte, the data byte is not written. If an EEPROM write transaction is terminated prior to the SCLK edge that latches the least significant bit of a data byte, none of the bytes in that write transaction are written.

Design Option: Wiring MOSI and MISO Together. Because communication between the bus master and the device is half-duplex, the MOSI and MISO pins can be wired together externally to reduce wire count. To support this option, the bus master must not drive the MOSI/MISO line when the device is transmitting.

AC Timing. See Table 13 and Figure 17 for AC timing specifications for the SPI interface.

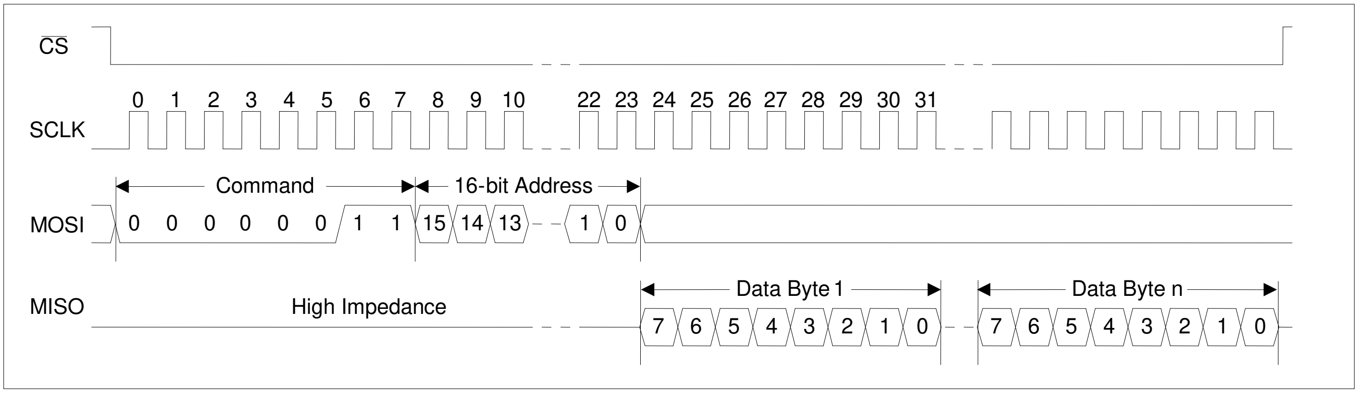


Figure 4 - SPI Read Transaction Functional Timing

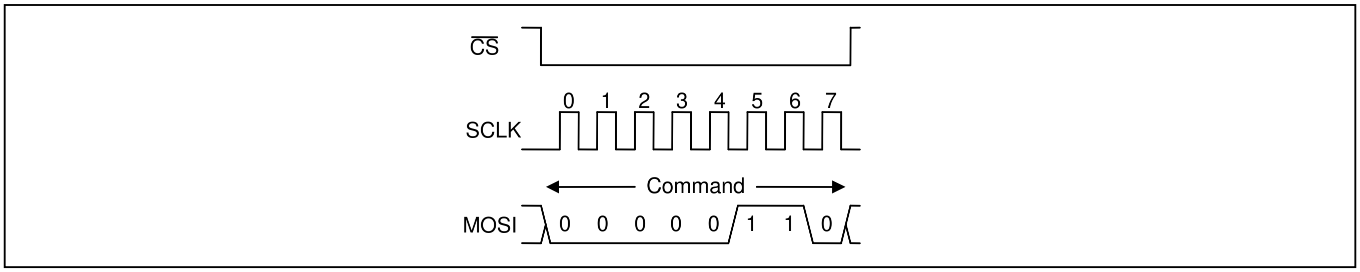


Figure 5 - SPI Write Enable Transaction Functional Timing

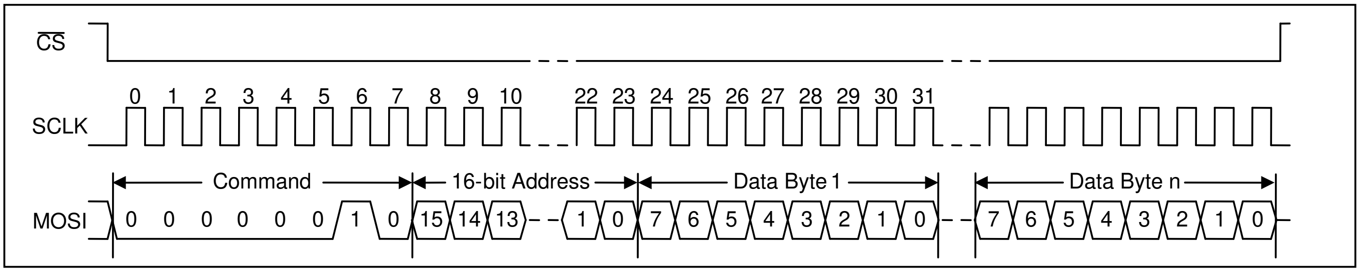


Figure 6 - SPI Write Transaction Functional Timing

3.7.2 I^2C Slave

The device can present a fast-mode (400kbit/s) I^2C slave port on the SCL and SDA pins. I^2C is a widely used master/slave bus protocol that allows one or more masters and one or more slaves to communicate over a two-wire serial bus. I^2C masters are typically microprocessors, ASICs or FPGAs. Data transfers are always initiated by the master, which also generates the SCL signal. The device is compliant with version 2.1 of the I^2C specification.

The I^2C interface on the device is a protocol translator from external I^2C transactions to internal SPI transactions. This explains the slightly increased protocol complexity described in the paragraphs that follow.

Read Transactions. The device registers are accessible when EESEL=0. The internal EEPROM memory is accessible when EESEL=1. The bus master first does an I^2C write to the device. In this transaction three bytes are written: the SPI Read command (see Table 3), the upper byte of the register address, and the lower byte of the register address. The bus master then does an I^2C read. During each acknowledge (A) bit the device fetches data from the read address and then increments the read address. The device then transmits the data to the bus master during the next 8 SCL cycles. The bus master terminates the read with a not-acknowledge (NA) followed by a STOP condition (P). See Figure 7. Note: If the I^2C write is separated in time from the I^2C read by other I^2C transactions then the device only outputs the data value from the first address and repeats that same data value after each acknowledge (A) generated by the bus master.

Register Write Transactions. The device registers are accessible when EESEL=0. The bus master does an I^2C write to the device. The first three bytes of this transaction are the SPI Write command (see Table 3), the upper byte of the register address, and the lower byte of the register address. Subsequent bytes are data bytes to be written. After each data byte is received, the device writes the byte to the write address and then increments the write address. The bus master terminates the write with a STOP condition (P). See Figure 8.

EEPROM Writes. The EEPROM memory is accessible when EESEL=1. The bus master first does an I^2C write to transmit the SPI Write Enable command (see Table 3) to the device. The bus master then does an I^2C write to transmit data to the device as described in the Register Write Transactions paragraph above. See Figure 9.

EEPROM Read Status. The bus master first does an I^2C write to transmit the SPI Read Status command (see Table 3) to the device. The bus master then does an I^2C read to get the status byte. In this byte, the least significant bit is set to 1 if the transfer is still in progress and 0 if the transfer has completed. See Figure 10.

I^2C Features Not Supported by the Device. The I^2C specification has several optional features that are not supported by the device. These are: 3.4Mbit/s high-speed mode (Hs-mode), 10-bit device addressing, general call address, software reset, and device ID. The device does not hold SCL low to force the master to wait.

I^2C Slave Address. The device's 7-bit slave address can be pin-configured for any of three values. These values are show in the table in section 3.2.

Bit Order. The I^2C specification requires device address, register address and all data bytes to be transmitted most significant bit first on the SDA signal.

Note: as required by the I^{2}C specification, when power is removed from the device, the SDA and SCL pins are left floating so they don't obstruct the bus lines.

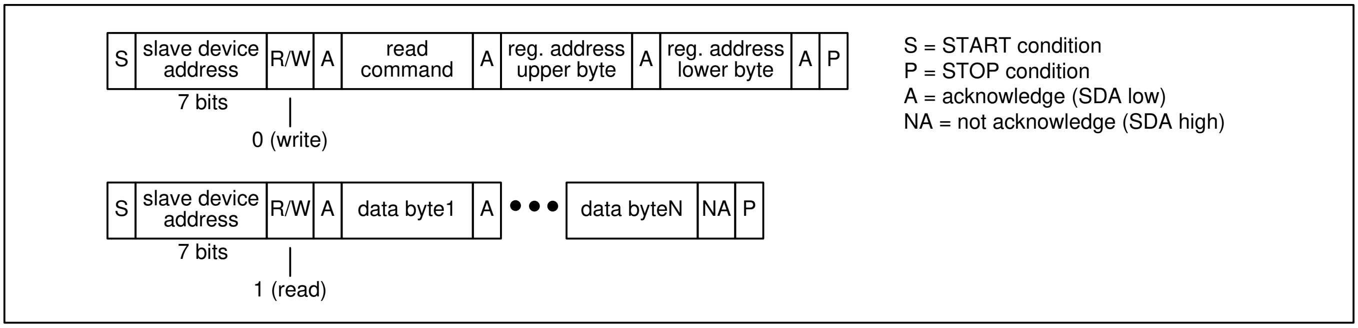


Figure 7 – I^{2}C Read Transaction Functional Timing

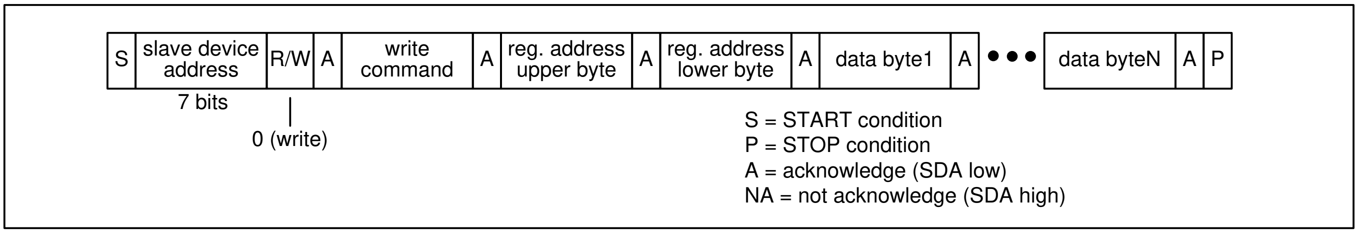


Figure 8 – I^{2}C Register Write Transaction Functional Timing

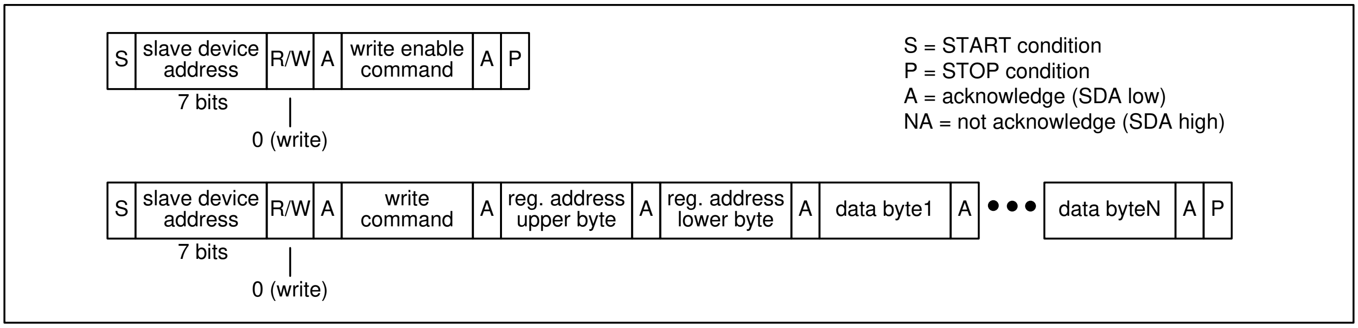


Figure 9 – I^{2}C EEPROM Write Transaction Functional Timing

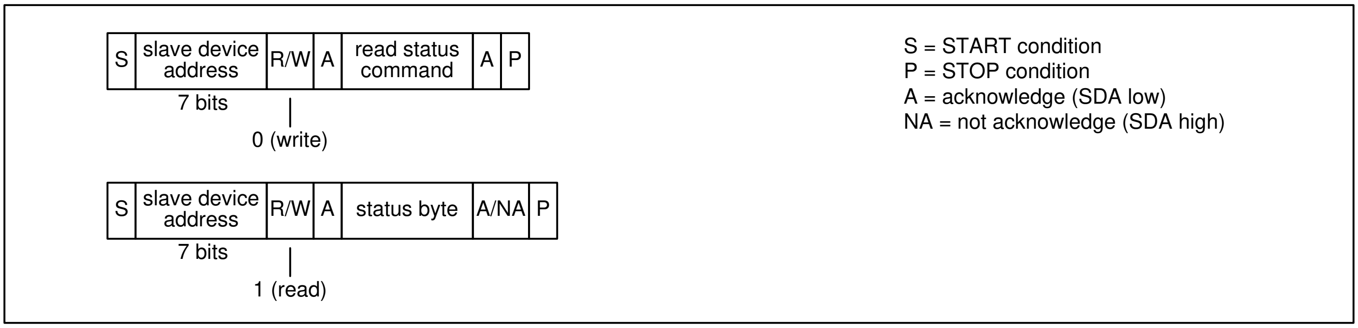


Figure 10 – I^{2}C EEPROM Read Status Transaction Functional Timing

Note: In Figure 7 through Figure 10, a STOP condition (P) immediately followed by a START condition (S) can be replaced by a repeated START condition (Sr) as described in the I^{2}C specification.

3.8 Interrupt Logic

Any of the GPIO pins can be configured as an interrupt-request output by setting the appropriate GPIOxC field in the GPIOCR registers to one of the status output options (01xx) and configuring the appropriate GPIOxSS register to follow the INTSR.INT bit. If system software is written to poll rather than receive interrupt requests, then software can read the INTSR.INT bit first to determine if any interrupt requests are active in the device.

Many of the latched status bits in the device can be the source of an interrupt request if their corresponding interrupt enable bits are set. The device's interrupt logic is shown in Figure 11. See the register map (Table 4) and the status register descriptions in section 4.3.2 for descriptions of the register bits shown in the figure.

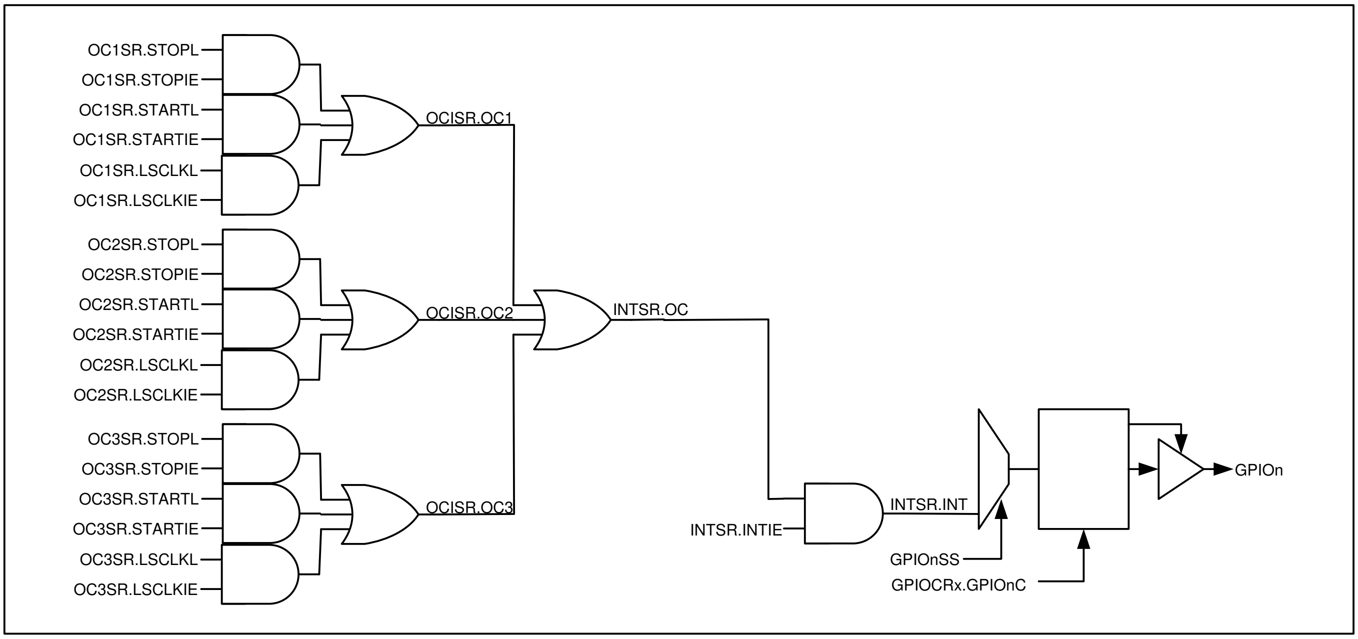


Figure 11 – Interrupt Structure

3.9 Reset Logic

The device has two reset controls: the RSTN pin and the RST bit in MCR1. The RSTN pin asynchronously resets the entire device. When the RSTN pin is low all internal registers are reset to their default values. **The RSTN pin must be asserted once after power-up.** At initial power-up reset should be asserted for at least 1µs. During operation, the RSTN assertion time can be as short as 1µs with one important exception:

> Consider each of these four pins: AC0/GPIO0, AC1/GPIO1, TEST/GPIO2 and IF1/MISO. If (1) the pin could be an output driving high when RSTN is asserted, and (2) an external pulldown resistor is used to set the at-reset value of the pin, then RSTN should be asserted for 100 milliseconds.

The MCR1.RST bit resets the entire device (except for the microprocessor interface and the RST bit itself), but when the RST bit is active, the register fields with pin-programmed defaults do not latch their values from, or based on, the corresponding input pins. Instead these fields are reset to the default values that were latched when the RSTN pin was last active.

Important: System software must wait at least 100µs after RSTN is deasserted and wait for GLOBISR.BCDONE=1 before configuring the device.

3.10 Power-Supply Considerations

Due to the multi-power-supply nature of the device, some I/Os have parasitic diodes between a <3.3V supply and a 3.3V supply. When ramping power supplies up or down, care must be taken to avoid forward-biasing these diodes because it could cause latchup. Two methods are available to prevent this. The first method is to place a Schottky diode external to the device between the <3.3V supply and the 3.3V supply to force the 3.3V supply to be within one parasitic diode drop of the <3.3V supply. The second method is to ramp up the 3.3V supply first and then ramp up the <3.3V supply. In some applications VDDOx power supply pins can be at other voltages, such as 2.5V or 1.5V. In these applications the general solution is to ramp up the supplies in order from highest nominal to lowest nominal voltage.

3.11 Auto-Configuration from EEPROM

3.11.1 Factory-Default Device Configurations

As shipped from Microsemi, the device auto-configures at reset as follows:

- IC1 and IC2 receivers enabled and input high-speed dividers set to 1 (don't divide). ICEN=0x03.
- External switching mode enabled with the GPIO3 pin as the control signal switching between between IC1 (GPIO3=0) and IC2 (GPIO3=1). MCR2=0x60, SRCCR3=0x51.
- OC1, OC2 and OC3 medium-speed and low-speed dividers enabled and set to 1 (don't divide). OCEN=0x07.
- OC1, OC2 and OC3 signal format specified at reset by the AC[1:0] pins as shown in the table below

AC1	AC0	OC1, OC2, OC3 Output Configuration	OCxCR2 (0x201,0x211,0x221)
0	0	3 full-swing CML outputs	0x01
0	1	3 HCSL outputs (apply 1.8V to all VDDO pins)	0x17
1	0	6 CMOS outputs (apply 3.3V or 2.5V to all VDDO pins)	0x04
1	1	Disabled	0x00

- Write SRCCR1=0x02 to set bit 2 to 1 and write OC1CR3=0x40 to set bit 6 to 1 as required.

3.11.2 Generating Other Device Configurations

EEPROM auto-configuration files for the ZL40255 must be generated by Microsemi factory application engineers. Contact your local Microsemi Field Applications Engineer (FAE) for assistance.

3.11.3 Direct EEPROM Write Mode

To simplify writing the internal EEPROM during manufacturing, the device has a test mode known as direct EEPROM write mode. The device enters this mode when TEST=1 and AC[1:0]=00 on the rising edge of RSTN. In this mode the EEPROM memory is mapped into the address map and can be written as needed to store configuration scripts in the device. Device registers are not accessible in this mode. The device exits this mode when TEST=0 on the rising edge of RSTN. Note: the device drives the MISO pin continually during this mode. Therefore this mode cannot be used when MOSI and MISO are tied together (as described in the *Design Option: Wiring MOSI and MISO Together* paragraph in section 3.7.1).

3.12 Power Supply Decoupling and Layout Recommendations

Application Note ZLAN-594 describes recommended power supply decoupling and layout practices.

4. Register Descriptions

The device has an overall address range from 000h to 1FFh. Table 4 shows the register map. In each register, bit 7 is the MSb and bit 0 is the LSb. Register addresses not listed and bits marked "—" are reserved and must be written with 0. Writing other values to these registers may put the device in a factory test mode resulting in undefined operation. Bits labeled "0" or "1" must be written with that value for proper operation. Register fields with underlined names are read-only fields; writes to these fields have no effect. All other fields are read-write. Register fields are described in detail in the register descriptions that follow Table 4.

4.1 Register Types

4.1.1 Status Bits

The device has two types of status bits. Real-time status bits are read-only and indicate the state of a signal at the time it is read. Latched status bits are set when a signal changes state (low-to-high, high-to-low, or both, depending on the bit) and cleared when written with a logic 1 value. Writing a 0 has no effect. When set, some latched status bits can cause an interrupt request if enabled to do so by corresponding interrupt enable bits. Status bits marked "—" are reserved and must be ignored.

4.1.2 Configuration Fields

Configuration fields are read-write. During reset, each configuration field reverts to the default value shown in the register definition. Configuration register bits marked "—" are reserved and must be written with 0.

4.1.3 Bank-Switched Registers

The EESEL register is a bank-select control field that maps the device registers into the memory map at address 0x1 and above when EESEL=0 and maps the EEPROM memory into the memory map at address 0x1 and above when EESEL=1. The EESEL register itself is always in the memory map at address 0x0 for both EESEL=0 and EESEL=1.

4.2 Register Map

Table 4 - Register Map

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Global Configuration Registers									
00h	EESEL	—	—	—	—	—	—	—	EESEL
09	MCR1	RST	—	—	—	—	—	XAB[1:0]	
0A	MCR2	—	EXTSS[1:0]		—	—	—	—	—
0C	ICEN	—	—	—	—	—	IC3EN	IC2EN	IC1EN
0D	OCEN	—	—	—	—	—	OC3EN	OC2EN	OC1EN
0E	GPIOCR1	GPIO1C[3:0]				GPIO0C[3:0]			
0F	GPIOCR2	GPIO3C[3:0]				GPIO2C[3:0]			
12	GPIO0SS	REG[4:0]					BIT[2:0]		
13	GPIO1SS	REG[4:0]					BIT[2:0]		
14	GPIO2SS	REG[4:0]					BIT[2:0]		
15	GPIO3SS	REG[4:0]					BIT[2:0]		
1B	PACR1	RST	TRIG	ARM	—	—	—	TINV	MODE
1C	PACR2	ARMSRC[3:0]				TRGSRC[3:0]			
Status Registers									
30	ID1	IDU[7:0]							
31	ID2	IDL[3:0]				REV[3:0]			
40	CFGSR	TEST	—	—	—	IF[1:0]		AC[1:0]	

ADDR	REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
41	GPIOSR	—	—	—	—	GPIO3	GPIO2	GPIO1	GPIO0
42	INTSR	—	—	OC	—	—	—	INTIE	INT
43	GLOBISR	BCDONE	—	—	—	—	—	—	—
45	OCISR	—	—	—	—	—	OC3	OC2	OC1
4D	PASR	—	—	—	—	—	—	BUSY	ARMED
53	OC1SR	LSCLKIE	LSCLKL	LSCLK	STARTIE	STARTL	STOPIE	STOPL	STOPD
54	OC2SR	LSCLKIE	LSCLKL	LSCLK	STARTIE	STARTL	STOPIE	STOPL	STOPD
55	OC3SR	LSCLKIE	LSCLKL	LSCLK	STARTIE	STARTL	STOPIE	STOPL	STOPD
Source Selection Configuration Registers									
100	SRCCR1	—	—	—	—	—	—	1	—
102	SRCCR3	—	EXTSW	ALTMUX[2:0]			INMUX[2:0]		
Output Clock Configuration Registers									
	OC1 Registers								
200	OC1CR1	PHEN	MSDIV[6:0]						
201	OC1CR2	—	POL	DRIVE[1:0]		STOPDIS	OCSF[2:0]		
202	OC1CR3	SRLSEN	1	NEGLSD	LSSEL	—	—	—	LSDIV[24]
203	OC1DIV1	LSDIV[7:0]							
204	OC1DIV2	LSDIV[15:8]							
205	OC1DIV3	LSDIV[23:16]							
206	OC1DC	OCDC[7:0]							
207	OC1PH	PHADJ[7:0]							
208	OC1STOP	STOP	—	SRC[3:0]				MODE[1:0]	
	OC2 Registers								
210	OC2CR1	same as OC1 registers							
…	…								
218	OC2STOP								
	OC3 Registers								
220	OC3CR1	same as OC1 registers							
…	…								
228	OC3STOP								
Input Clock Configuration									
300	IC1CR1	—	POL	—	—	—	—	HSDIV[1:0]	
320	IC2CR1	—	POL	—	—	—	—	HSDIV[1:0]	
340	IC3CR1	—	POL	—	—	—	—	HSDIV[1:0]	

4.3 Register Definitions

4.3.1 Global Configuration Registers

Register Name: EESEL
Register Description: EEPROM Memory Selection Register
Register Address: 00h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	—	—	EESEL
Default	0	0	0	0	0	0	0	0

Bit 0: EEPROM Memory Select (EESEL). This bit is a bank-select that specfies whether device register space or EEPROM memory is mapped into addresses 0x1 and above. See sections 3.7 and 4.1.3.

0 = Device registers
1= EEPROM memory

Register Name: MCR1
Register Description: Master Configuration Register 1
Register Address: 09h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RST	—	—	—	—	—	XAB[1:0]	
Default	0	0	0	0	0	0	0	0

Bit 7: Device Reset (RST). When this bit is high the entire device is held in reset, and all register fields, except the RST bit itself, are reset to their default states. When RST is high, the register fields with pin-programmed defaults do not latch their values from the corresponding input pins. Instead these fields are reset to the default values that were latched from the pins when the RSTN pin was last active. See section 3.9.

0 = Normal operation
1 = Reset

Bits 1 to 0: XA/XB Pin Mode (XAB[1:0]). This field specifies the behavior of the XA and XB pins. See section 3.3.

00 = Crystal driver and input disabled / powered down
01 = Crystal driver and input enabled on XA/XB
10 = XA enabled as single-ended input for external oscillator signal; XB must be left floating
11 = {unused value}

Register Name: MCR2
Register Description: Master Configuration Register 2
Register Address: 0Ah

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	EXTSS[1:0]		—	—	—	—	—
Default	0	0	0	0	0	0	0	0

Bits 6 to 5: External Switch Source Select (EXTSS[1:0]). This field selects the GPIO source for the external switch control signal. It is only valid when SRCCR3.EXTSW=1. See section 3.5.

00 = GPIO0
01 = GPIO1
10 = GPIO2
11 = GPIO3

Register Name: ICEN
Register Description: Input Clock Enable Register
Register Address: 0Ch

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	IC3EN	IC2EN	IC1EN
Default	0	0	0	0	0	0	0	0

Bit 2: Input Clock 3 Enable (IC3EN). This bit enables and disables the input clock 3 differential receiver and input dividers. See section 3.4.
0 = Disabled
1 = Enabled

Bit 1: Input Clock 2 Enable (IC2EN). This bit enables and disables the input clock 2 differential receiver and input dividers. See section 3.4.
0 = Disabled
1 = Enabled

Bit 0: Input Clock 1 Enable (IC1EN). This bit enables and disables the input clock 1 differential receiver and input dividers. See section 3.4.
0 = Disabled
1 = Enabled

Register Name: OCEN
Register Description: Output Clock Enable Register
Register Address: 0Dh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	—	—	—	—	—	OC3EN	OC2EN	OC1EN
Default	0	0	0	0	0	0	0	0

Bit 2: Output Clock 3 Enable (OC3EN). This bit enables and disables the output clock 3 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section 3.6.1.
0 = Disabled
1 = Enabled

Bit 1: Output Clock 2 Enable (OC2EN). This bit enables and disables the output clock 2 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section 3.6.1.
0 = Disabled
1 = Enabled

Bit 0: Output Clock 1 Enable (OC1EN). This bit enables and disables the output clock 1 drivers, output dividers, phase adjustment/alignment circuitry and start/stop circuitry. See section 3.6.1.
0 = Disabled
1 = Enabled

Register Name: GPIOCR1
Register Description: GPIO Configuration Register 1
Register Address: 0Eh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GPIO1C[3:0]				GPIO0C[3:0]			
Default	0	0	0	0	0	0	0	0

Bits 7 to 4: GPIO1 Configuration (GPIO1C[3:0]). This field configures the GPIO1 pin as a general-purpose input, a general-purpose output driving low or high, or a status output. The current state of the pin can be read from GPIOSR.GPIO1. When GPIO1 is a status output, the GPIO1SS register specifies which status bit is output.

0000 = General-purpose input
0001 = General-purpose input - inverted polarity
0010 = General-purpose output driving low
0011 = General-purpose output driving high
0100 = Status output – non-inverted polarity
0101 = Status output - inverted polarity of the status bit it follows
0110 = Status output – 0 drives low, 1 high impedance
0111 = Status output – 0 high impedance, 1 drives low
1000 to 1111 = {unused values}

Bits 3 to 0: GPIO0 Configuration (GPIO0C[3:0]). This field configures the GPIO0 pin as a general-purpose input, a general-purpose output driving low or high, or a status output. The current state of the pin can be read from GPIOSR.GPIO0. When GPIO0 is a status output, the GPIO0SS register specifies which status bit is output.

0000 = General-purpose input
0001 = General-purpose input - inverted polarity
0010 = General-purpose output driving low
0011 = General-purpose output driving high
0100 = Status output – non-inverted polarity
0101 = Status output - inverted polarity of the status bit it follows
0110 = Status output – 0 drives low, 1 high impedance
0111 = Status output – 0 high impedance, 1 drives low
1000 to 1111 = {unused values}

Register Name: GPIOCR2
Register Description: GPIO Configuration Register 2
Register Address: 0Fh

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GPIO3C[3:0]				GPIO2C[3:0]			
Default	0	0	0	0	0	0	0	0

These fields are identical to those in GPIOCR1 except they control GPIO2 and GPIO3.

Register Name: GPIO0SS
Register Description: GPIO0 Status Select Register
Register Address: 12h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	REG[4:0]					BIT[2:0]		
Default	0	0	0	0	0	0	0	0

Bits 7 to 3: Status Register (REG[4:0]). When GPIOCR1.GPIO0C=01xx, this field specifies the register of the status bit that GPIO0 will follow while the BIT field below specifies the status bit within the register. Setting the combination of this field and the BIT field below to point to a bit that isn't implemented as a real-time or latched status register bit results in GPIO0 being driven low. The address of the status bit that GPIO0 follows is 0x40 + REG[4:0]

Bits 2 to 0: Status Bit (BIT[2:0]). When GPIOCR1.GPIO0C=01xx, the REG field above specifies the register of the status bit that GPIO0 will follow while this field specifies the status bit within the register. Setting the combination of the REG field and this field to point to a bit that isn't implemented as a real-time or latched status register bit results in GPIO1 being driven low. 000=bit 0 of the register. 111=bit 7 of the register.

Note: The device does not allow the GPIO status register bits in GPIOSR to be followed by a GPIO.

Register Name: GPIO1SS
Register Description: GPIO1 Status Select Register
Register Address: 13h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	REG[4:0]					BIT[2:0]		
Default	0	0	0	0	0	0	0	0

These fields are identical to those in GPIO0SS except they control GPIO1.

Register Name: GPIO2SS
Register Description: GPIO2 Status Select Register
Register Address: 14h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	REG[4:0]					BIT[2:0]		
Default	0	0	0	0	0	0	0	0

These fields are identical to those in GPIO0SS except they control GPIO2.

Register Name: GPIO3SS
Register Description: GPIO3 Status Select Register
Register Address: 15h

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	REG[4:0]					BIT[2:0]		
Default	0	0	0	0	0	0	0	0

These fields are identical to those in GPIO0SS except they control GPIO3.