QF4A512 Single-Channel High-Speed Operation

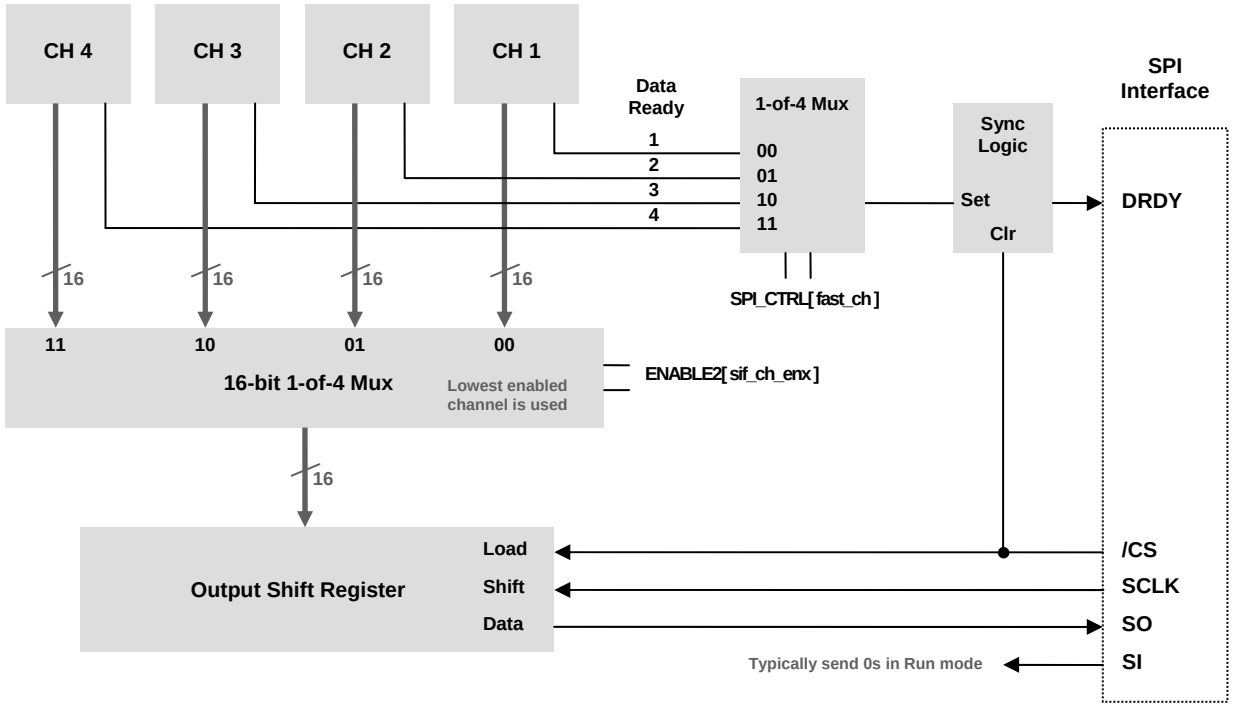
1) Introduction

The digital interface to the QF4A512 is a Serial Peripheral Interface (SPI) in the slave configuration. In Run mode, a continuous stream of filtered conversion results is available at the SPI port. Applications that require maximum speed and only one channel can use the Single-Channel High-Speed mode, which is the topic of this Application Note.

2) Overview

The simplest and fastest mode of operation is the Single-Channel mode, in which only one channel is in operation and only its data is sent (no flag bits).

Figure 1 – QF4A512 Interface Logic for Single-Channel Run Mode Operation



The operation sequence is as follows –

Figure 2 – QF4A512 Operation Sequence during Single-Channel Run Mode Operation

1. The lowest enabled channel in **ENABLE2** (register 0Bh) finishes processing data and sets its Data Ready signal high
2. If the same channel is selected in **SPI_CTRL[fast_ch]** (register 15h), DRDY will latch high, interrupting the host CPU
3. The host CPU activates /CS, which loads the channel data into the Output Shift Register and clears DRDY

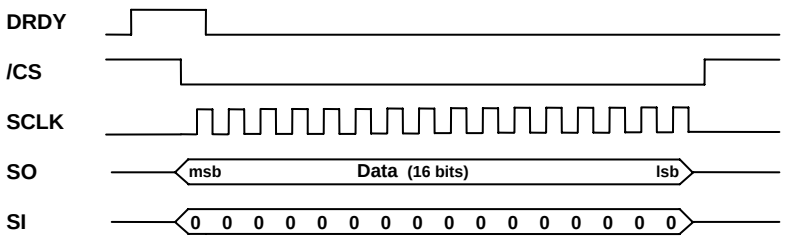
4. The host serially clocks the 16-bits of data out the SO pin, most significant bit first
5. The host completes the cycle by deactivating /CS.

Note that data *received* by the QF4A512 in Run mode is still evaluated by its Command Processor. For example, to exit Run mode the host has to send the mode change command while still in Run mode. But typically, unless intentionally controlling the device, the host CPU should always send zeroes. The 0 command is benign (it writes to the Scratch register, GLBL_SW).

Debug note

The clocks that drive DRDY and /CS are asynchronous to each other (DRDY is driven by the internal SYS_CLK and /CS is controlled by the Host CPU clock), so the QF4A512 must synchronize /CS to SYS_CLK before using it to clear DRDY. This synchronization can take up to three SYS_CLK cycles. In the rare design where SYS_CLK << SCLK, DRDY might not clear before /CS is de-asserted. Be sure the design allows /CS to be active (low) for at least four SYS_CLK cycles.

Figure 3 – QF4A512 SPI Bus Sequence during Single-Channel Run Mode Operation



Notes -
1. SPI ports only emit the SCLK when actually transferring data.
2. Data is valid on the rising edge of SCLK.

Background Info

The SPI bus is not officially promulgated by a standards body, but the implementation invented at Freescale (formerly Motorola) has become the *de facto* standard. See - www.freescale.com/files/microcontrollers/doc/ref_manual/S12SPIV4.pdf.

3) Calculating SCLK

The QF4A512 SPI interface is always in the slave configuration, with the Host CPU providing SCLK. To avoid data loss, a complete SPI read cycle must occur in less time than it takes the channel to produce consecutive samples. Expressing that concept as an equation yields

Equation 1 – Maximum Time to Remove SPI Data without Loss

$$\text{Time Between Samples} \geq \text{DRDY Interrupt to /CS Activated} + \text{SCLK Period x 16} + \text{Data Transferred to /CS Deactivated}$$

Solving for *SCLK Period*, produces

Equation 2 – Minimum SCLK Period to Remove SPI Data without Loss

$$\text{SCLK Period} \leq \frac{\text{Time Between Samples} - \text{DRDY Interrupt to /CS Activated} - \text{Data Transferred to /CS Deactivated}}{16}$$

SCLK Period should ultimately be expressed as *SCLK Frequency*, and *Time Between Samples* should be expressed as the *Effective Sample Rate*. Inverting *SCLK Period* and *Time Between Samples* converts them to the frequency/rate form. The equation changes to

Equation 3 – Minimum SCLK Frequency to Remove SPI Data without Loss

$$\text{SCLK Frequency} \geq \frac{16}{(1 / \text{Effective Sample Rate}) - \text{DRDY Interrupt to /CS Activated} - \text{Data Transferred to /CS Deactivated}}$$

Notes -
1. If SCLK Frequency < SYS_CLK, be sure /CS will be low for at least four SYS_CLK cycles (per DRDY timing discussion above)

Note that the Effective Sampling Rate is used, not the raw ADC sampling rate. The SCLK calculation provided by the QuickFilter software takes all of these factors into account.

Debug note

DRDY is asserted whenever data is ready, regardless of the state of /CS. If a low-to-high transition is observed on DRDY while /CS is low, it is likely that the bus is not running fast enough to keep up with the data output rate.

4) Example

Having laid the technical foundation, a practical example can now be considered. In this single-channel example, channel 2 is emitting filtered samples at a rate of 100,000/s and a 16-bit host CPU is tasked with reading the samples

without any data loss. What follows is a description of the SPI initialization and operation sequences that will fulfill that task.

Initialization

This Section describes the initialization of the SPI-related features of the QF4A512. Configuration of other chip features, such as clocks, sampling rates and FIR filters are described in other Application notes. The QF4A512 Datasheet shows the timing for Configure mode cycles.

We have done some experimenting with our Host CPU timing and have come up with values for the terms in Equation 3. Those values are given in Figure 4 below.

Figure 4 – SPI Timing Factors for Example 1

1. Effective Sample Rate = 100,000 samples/s
2. DRDY Interrupt to /CS Activated = 1 µs
3. Data Transferred to /CS Deactivated = 1 µs

Inserting the timing factors from Figure 4 into Equation 3, SCLK must be greater than or equal to 2 MHz. However, the design must guarantee that the data read cycle will always complete prior to the next DRDY signal (to avoid data loss). Since DRDY and /CS are asynchronous, SCLK is increased to account for manufacturing tolerances in the clocks that drive those signals. In this example increasing SCLK to 2.1MHz allows for a 5% margin.

Figure 5 - Host CPU Initialization Sequence Description

1. Clear the cycle counter used in the SPI Receive Interrupt handler
2. Select an initial 16-word data buffer for the SPI Receive Interrupt handler
3. Set the SCLK divider to 2.1 MHz
4. Initialize SPI Master mode
5. Enter QF4A512 Configure mode and setup the SPI features

RUN_MODE[run_mode] = 0	Enter Configure mode
SPI_CTRL[sif_singlech] = 1	Enable single channel mode
SPI_CTRL[fast_ch] = 2	Set Channel 2 as the source of DRDY
ENABLE2[sif_ch_en2] = 1	Enable Channel 2 (and only Channel 2
RUN_MODE[run_mode] = 1	Enable Run mode

6. Synchronize with data stream by and throwing away the first sample. To do this, activate /CS (don't shift any data), wait for DRDY to clear, and then enable DRDY interrupts to start on the beginning of the next cycle.

Operation

This Section details the steps required to acquire and process single-channel data from the Host CPU point of view. The host CPU in this example has no DMA capability, so acquisition is driven by two interrupt service routines, one for the DRDY signal, which will occur first, and another for SPI receive, which will complete the transfer cycle.

Higher level processing of the acquired data can be handled by a simple foreground polling loop or by a kernel task, depending on the host software design. Notification of higher level software occurs after each 16 cycles.

Figure 6 – DRDY Pin Interrupt Handler Sequence Description

1. Disable DRDY interrupts
2. Take /CS low
3. Clear SPI receive flag then enable SPI receive interrupts
4. Load the transmit register with 0 then start a dummy transmit cycle to clock the QF4A512 data into the SPI receive register

Figure 7 – SPI Receive Interrupt Handler Sequence Description

1. Disable SPI receive interrupts
2. Take /CS high
3. Store the received data in the current buffer
4. Increment the cycle counter
5. If this is the 16th cycle
    a. Alert foreground loop or task that a new block of data has arrived and indicate which buffer was used
    b. Select a new 16-word buffer
    c. Reset the cycle counter to 0
6. Clear DRDY flag then enable DRDY interrupts

Special Case

Hosts with only an 8-bit SPI port have some special considerations. First, two 8-bit cycles are required to transfer all 16-bits of a cycle. A delay between SPI bytes has no effect on the QF4A512 logic, just be sure the delay is factored into the overall timing to avoid data loss. Also, if the 8-bit Host SPI port has an automatically generated SPI Select output pin, it cannot be used for the /CS signal. /CS must stay low for all 16-bits of data transfer. A manually controlled general-purpose IO pin can provide the /CS signal.

5) Conclusion

The Single-Channel configuration sends out a stream of 16-bit data triggered by the DRDY signal, making it the fastest mode of the QF4A512. This Note described the Single-Channel configuration and presented a practical example of its use.

Contact Information:

Quickfilter Technologies, Inc.
1024 S. Greenville Avenue, Suite 100
Allen, TX 75002-3324

General: info@quickfilter.net
Applications: apps@quickfilter.net
Sales: sales@quickfilter.net
Phone: 214-547-0460
Fax: 214-547-0481
Web: www.quickfiltertech.com