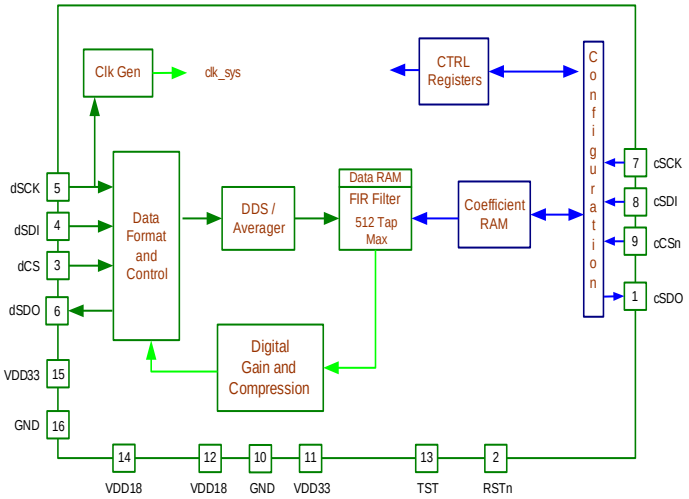
QF1Da512

Simple and versatile FIR engine (SavFIRe™)

APPLICATIONS

- **Audio Equalization, Crossovers, and 3D widening**
- **Televisions**
- **Docking Stations**
- **Stereo Headsets (both wired and wireless)**
- **Home Theatre Speakers**
- **Digital Volume Control**
- **Other powered speaker products**



DESCRIPTION

A programmable digital filter designed for seamless insertion in a serial data path or used as a coprocessor. The device can be programmed using the Quickfilter Pro™ Design Software which supports a variety of FIR digital filter algorithms plus import and curve fitting (for equalization and other unique transfer functions). The FIR filter has 512 taps capable of easily generating audio equalization and crossover filters. The QF1Da512 supports a wide variety of data sources. It also has **digital gain** and **programmable compression allowing the part to also serve as a distortion-less digital volume control**. The QF1Da512 can be configured to filter one or both channels of an I2S data stream. The filter can operate over a broad range of data rates from 10sps up to 500ksps and can support data resolutions ranging from 12 to 24 bits.

ORDERING INFORMATION

Device	Package
QF1Da512-QN-T QF1Da512-QN-B	16-Pin QFN - Tape & Reel (Reel qty 1000) - Trays (tray qty 496)
QF1D512-DK QF1Da512-DK	Base 1D Development Kit 1Da Audio Dragon Platform

FEATURES

- 512-tap symmetric or 256-tap non-symmetric FIR filter
- Up to 24 bit data words
- 32 bit coefficients
- Built in digital gain and compression
- Re-programmable in circuit
- Data Rate: Up to 500ksps (far exceeds audio needs)
- Data Interface: I^2S, SPI & synchronous serial modes
- SPI Configuration
- Multiple devices can be daisy chained, with programmable bypass mode.
- Programmable Averaging and Down-sampler pre-FIR, including bypass mode
- Low Power: < 3mW @ 44.1Ksps, <6mW @ 96Ksps
- 3.3V Digital I/O, 5 Volt Tolerant with 3.3V & 1.8V Supplies
- Industrial Temp -40C to +85C
- Package: 16-pin QFN (3 X 3 mm)

QUICKFILTER DEVELOPMENT ENVIRONMENT

QF1Da512-DK Audio Evaluation Platform

- Daughter card with stereo RCA jacks for input and output
- Also support SPDIF inputs and outputs

QF1D512-DK; Motherboard

- Host for configuring and programming QF1Da512-DK via USB

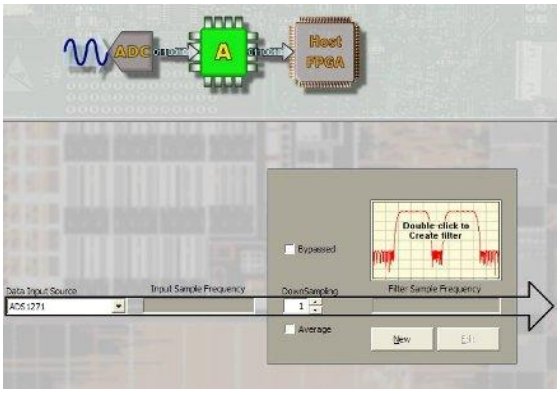


TABLE OF CONTENTS

1 SPECIFICATIONS

Absolute Maximum Ratings

Stresses above those listed under Absolute Maximum Ratings can cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

Parameter	Min	Max	Units
Storage Temperature	-40	150	ºC
Supply Voltage, V_{DD18} to GND	-0.3V	2.16V	V
Supply Voltage, V_{DD33} with respect to GND	-0.3V	4.0V	V
Digital Input Voltage with respect to GND	-0.3	5.8	V
ESD Immunity (HBM, JESD22-A114-D Class 1C)		7.5	KV
ESD Immunity (HBM, AEC-Q100-002D)		6.75	KV

This integrated circuit can be damaged by ESD. Quickfilter Technologies recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

Package Assembly

The QF1Da512 is offered in a “green” package (RoHS & no Sb/Br), assembled with enhanced environmentally compatible Pb-free and halide-free materials. The leads possess a matte-tin plating which is compatible with conventional board assembly processes or newer lead-free board assembly processes. The peak soldering temperature should not exceed 260°C during printed circuit board assembly.

Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Units
Supply Voltage 1.8	V_{DD18}	1.62	1.8	1.98	V
Supply Voltage 3.3	V_{DD33}	3.0	3.3	3.6	V
Digital Input Voltage		0		5.5	V
Ambient Temperature	T_A	-40	25	85	ºC

Note: Quickfilter guarantees the performance of this device over specified ranges by conducting electrical characterization over each range and by conducting a production test with single insertion coupled to periodic sampling.

Typical Performance Characteristics

Default Conditions: T_A = 25 C, V_{DD18} = 1.8V, V_{DD33} = 3.3V,

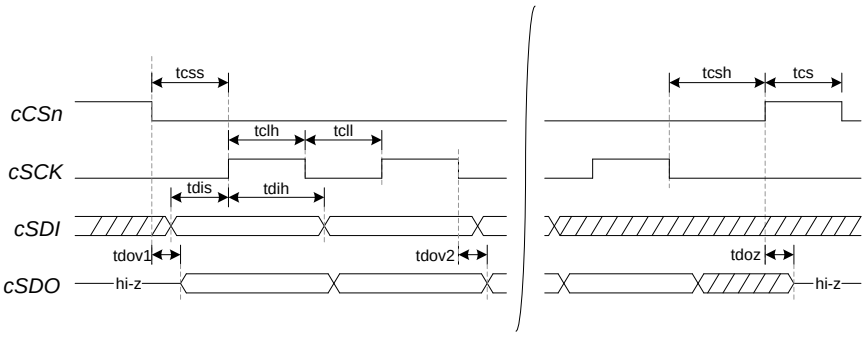
Electrical Characteristics

Symbol	Parameter (Condition)		Min	Typ	Max	Units	Note
Supply Current & Power Dissipation							
I_{DD18}	1.8V Supply Operating Current	- f_S = 50 kHz, 512 taps		1.3		mA	
		- f_S = 299 kHz, 512 taps		6.6		mA	
		- f_S = 500 kHz, 512 taps		11.6		mA	
		- Standby		50		nA	
I_{DD33}	3.3V Supply Operating Current	- f_S = 50 kHz		70		uA	
		- f_S = 299 kHz		200		uA	
		- f_S = 500 kHz		400		uA	
		- Standby		20		uA	
Digital Inputs (cSDI, cCSn, cSCK, dSDI, dCS, dSCK, RSTn)							
V_{IH}	High-level Input Voltage, V_{DD33} = 3.6V		2.0		5.5	V	
V_{IL}	Low-level Input Voltage, V_{DD33} = 3.0V		-0.3		0.8	V	
I_{IH}, I_{IL}	Input (leakage) current			+/- 1		uA	
C_{IN}	Input Capacitance			10		pF	
R_{PU}	Pull Up Resistance		40	75	190	KOhm	
R_{PD}	Pull Down Resistance		40	75	190	KOhm	
Digital Output (cSDO, dSDO)							
V_{OH}	High-level Output Voltage, DV_{DD33} = 3.0V, I_{OH} = -100uA		2.4			V	
V_{OL}	Low-level Output Voltage, DV_{DD33} = 3.0V, I_{OL} = 100uA				0.4	V	
C_O	Output Capacitance				10	pF	
I_{load}	Output Current			4	mA		
C_{load}	Load Capacitance			7	pF		

Timing Requirements

1.1.1 Configuration Timing Requirements

Timing



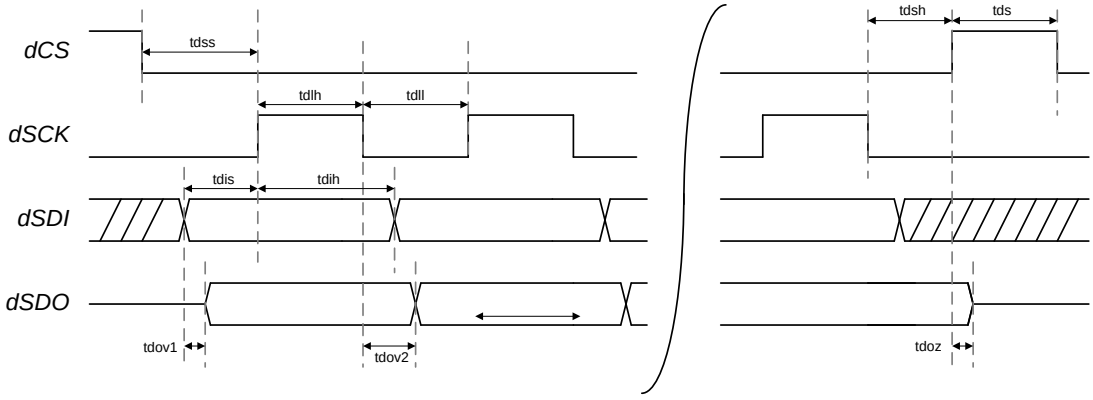
Parameters

Symbol	Parameter (SPI Input)	Min	Max	Units	Note
fs	Sample Frequency		500	KHz	
fsck	Serial Port Clock Frequency		20	MHz	
tcll	cSCK low time	12		nS	
tclh	cSCK high time	12		nS	
tcss	cCSn falling to cSCK rising* time	24		nS	
tcsh	cSCK falling to cCSn rising* time	24		nS	
tcs	cCSn high time	50		nS	
tdis	cSDI setup time	12		nS	
tdih	cSDI hold time	24		nS	
tdov1	cCSn falling to cSDO valid time	12		nS	
tdov2	cSCK falling* to cSDO valid time	12		nS	
tdoz	cCSn rising toc SDO high-Z time	12		nS	

* Using default cSCK capture/drive polarity.

1.1.2 Data Path Timing Requirements

Data Path Timing
(all modes)



Parameters

Symbol	Parameter(SPI Output)	Min	Max	Units	Note
fs	Sample Frequency		500	KHz	
fdck	Data Clock Frequency (dSCK)		24	MHz	
tdll	dSCK low time	12		nS	
tdlh	dSCK high time	12		nS	
tdss	Setup time dCS low to dSCK rising edge	24		nS	
tdsh	dSCK falling to dCS rising* time	24		nS	
tds	Hold time dCS high	50		nS	
tdis	dSDI setup time	12		nS	
tdih	dSDI hold time	24		nS	
tdov1	dSDI to dSDO valid (pass thru data)	8		nS	
tdov2	dSCK to dSDO valid (filtered data)	12		nS	
tdoz	dCS inactive to dSDO tri-state (Normal Mode)	12		nS	

NOTE: For tdov1 and tdov2, the output pin is set with I_{load} at 4mA and C_{load} at 7pf.

2 PINOUT and PIN DESCRIPTIONS

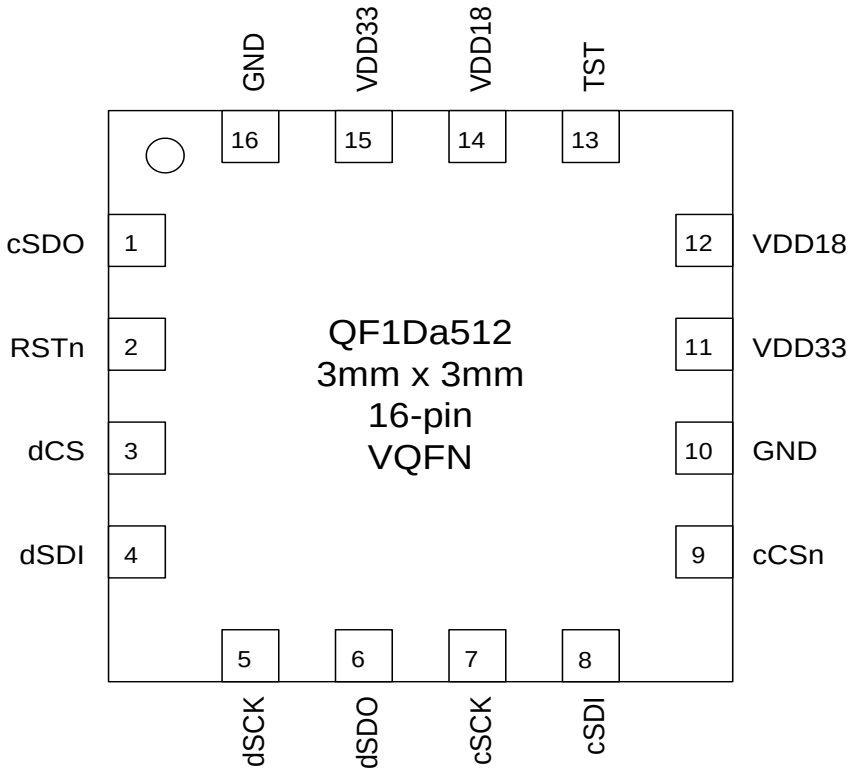


Figure 1: 16 Pin QFN Package Pin out

Table 1 Pin Descriptions

Pin	Signal Name	I/O	Type	Description
1	cSDO	O	Digital	Configuration serial data output
2	RSTn	I	Digital	Chip reset; active low; default internal pull-up
3	dCS	I	Digital	Data interface chip select; configurable as active low or high
4	dSDI	I	Digital	Data serial data input
5	dSCK	I	Digital	Data input clock used to capture dSDI; active capture edge either high, low or both
6	dSDO	O	Digital	Data serial data output
7	cSCK	I	Digital	Serial input clock used for configuration data transfers; active capture edge either high or low
8	cSDI	I	Digital	Configuration serial data input
9	cCSn	I	Digital	Configuration interface chip select; active low; default internal pull-up
10	GND	N/A	Return	Digital power supply return pin
11	VDD33	N/A	Power	Digital I/O voltage; 3.3V
12	VDD18	N/A	Power	Digital Core voltage; 1.8V
13	TST	I	Digital	Production test
14	VDD18	N/A	Power	Digital I/O voltage; 3.3V
15	VDD33	N/A	Power	Digital Core voltage; 1.8V
16	GND	N/A	Return	Digital power supply return pin

3 GENERAL DESCRIPTION

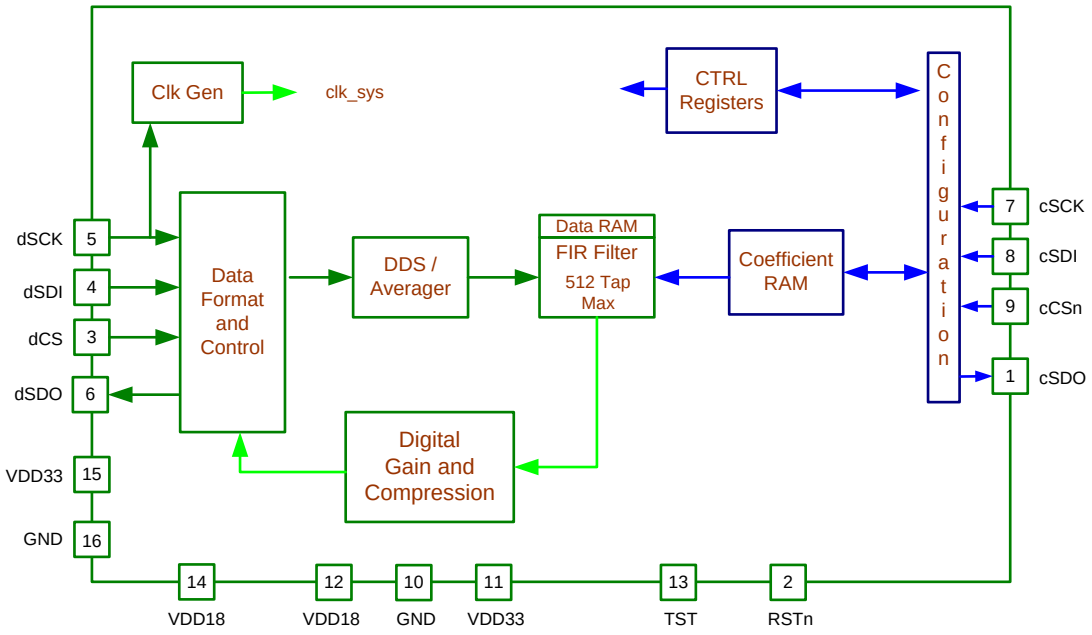


Figure 2: Functional Block Diagram

Data Format and Control

The Data Format and Control block accepts the digital serial data. It separates any header information and checks for data valid and then separates the data bits for processing. It also converts the data into the required format for the FIR filter. The data interface supports the standard I2S and SPI bus protocols. The data interface consists of the **dSCK**, **dCS**, **dSDI**, and **dSDO** pins and allows for digital data received on **dSDI** to be filtered or passed through to **dSDO**. If a header is used, it is passed unaltered along with the filtered data.

Averaging and Down-sampler

The Averaging / Down-sampler block down-samples the incoming data by a factor of 2 to 256. It also can be configured to average the down sampled data. Default is bypassed. This block also supports a special I2S feature of channel duplication which converts the standard left/right channel format to either all left or all right.

Finite Impulse Response Filter (FIR)

The FIR filter consists of up to 512 taps for a symmetric filter or 256 taps for an asymmetric filter.

Digital Gain and Compression (DGC)

DGC is used to maintain unity gain through the QF1Da512 for complex filters like those used in audio equalization. DGC is performed sample by sample with an instant attack and release. Both Gain and Compression (ratio) are comprised of 4 integer bits and 12 fractional bits. Quickfilter Pro™ can automatically determine Gain and Compression settings. Direct access and guidance is also available for advanced users.

Configuration Interface

The configuration interface supports the standard SPI bus protocol and operates in SLAVE mode. **cSCK** is capable of operating up to 20 MHz, although it may be run at much lower speeds.

The configuration interface consists of the **cSCK**, **cCSn**, **cSDI** and **cSDO** pins and is used to read and write the control registers and program the coefficient memory space.

4 SOFTWARE

Device Configuration

Before useful data can be output from the QF1Da512 it must first be correctly configured. Configuration parameters include the following:

1. Data mode of operation: I2S, SPI Normal, SPI Continuous, or Synchronous Serial mode
2. Data interface configuration: includes header information, use and size, and data size
3. Decimation and averaging configuration (if utilized)
4. FIR filter coefficients.
5. Gain and Compression (if utilized).

Quickfilter Audio Development Kit (QF1Da512-DK)

The Audio Development Kit is a development and evaluation platform for a stereo audio application utilizing two serial QF1Da512s. The QF1Da512-DK works in conjunction with the QF1D512-DK. The QF1D512-DK provides the interface to the Quickfilter Pro™ development software and hosts the QF1Da512 Audio Development Kit. Once programmed, the QF1Da512-DK can operate independent of the host.

The Quickfilter Pro™ Design Software allows all the necessary parameters to be generated in a quick and user-friendly manner. The user enters the desired characteristics (e.g. sampling rates, type of filter, cut-off frequencies etc.) for each channel and the software generates a configuration file for the devices. The configuration file can be immediately downloaded into the QF1Da512 on the evaluation platform, and the *actual hardware* ***device*** *performance* can be evaluated - in response to a user-applied signal source and sink. Device configuration parameters can be further adjusted, if necessary, until the optimum system performance is reached. Once the user is satisfied with the performance the Qf1Da512 configuration file can be saved for future use, for example to program devices in bulk prior to volume board manufacturing.

Quickfilter Host Development Kit (QF1D512-DK)

The Development Kit is a complete hardware and software combination which allows for rapid development of the QF1Da512 configuration parameters for a specific application.

The Quickfilter Design Software tool allows all the necessary parameters to be generated in a quick and user-friendly manner. The user enters the desired characteristics (e.g. sampling rates, type of filter, cut-off frequencies etc.) for each channel and the software generates a configuration file for the device. The configuration file can be immediately downloaded into the QF1Da512 on the QF1Da512-DK development board, and the *actual* ***filter*** *performance* can be monitored - either in response to a PC-generated noise source or to a user-applied signal. Device configuration parameters can be further adjusted, if necessary, until the optimum filter performance is reached.

Once the user is satisfied with the performance the QF1Da512 configuration file can be saved for future use, for example to program the QF1Da512-DK. The QF1Da512-DK can support 2 stereo configurations for comparisons.

5 GENERAL OPERATION

There are two modes of bus operation for the QF1Da512, “configure” mode, and “data” mode. The required mode is selected by writing the **FILT_EN** bit in the **CONTROL** register (03h).

“Configure” mode is used to set up or change options in the QF1Da512. In this mode it is possible to read/write the control registers and the coefficient RAM. To *configure* the device, set both the **FILT_EN** and **DIN_PT** bits to 0 (see Section 6.3).

“Data” Mode is used to interface with a data source (i.e, ADC, uP, DSP, digital data stream…) and run the FIR filter. In this mode it is possible to write the control registers. To *process data* with the device, set the **FILT_EN** bit high (see DATA INTERFACE). The format of the **dSDO** will match the format of the data coming into the **dSDI** pin.

Note: The Control registers can be written to and read from in either mode. However, to write the coefficients, the QF1Da512 must be in “configure” mode.

6 CONFIGURATION INTERFACE

SPI, Serial Peripheral Bus

6.1.1 Mode of Operation

The QF1Da512 is designed to interface directly with the serial peripheral interface (SPI) of microcontrollers and Digital Signal Processors in native mode. The QF1Da512 always operates in SPI slave mode using the **cSDI**, **cSDO**, **cSCK,** and **cCSn** pins. **cSDI** is the input serial data, and **cSDO** is the output serial data (if **FILT_EN** = **DIN_PT** = 0). **cSCK** is the input serial clock. Default is to capture on rising edge, but this can be changed to capture on falling edge by writing the **SCLK_POL** bit in the **DCONFIG** register (04h). The Control registers can be *written* while data is being processed.

6.1.2 Data Format

In order to address and read / write to the QF1Da512, **cCSn** is asserted low to select the device. When the device is in "configure" mode but is not selected, data will not be accepted via the serial input pin (**cSDI)** and the serial output pin (**cSDO**) will remain in a high impedance state.

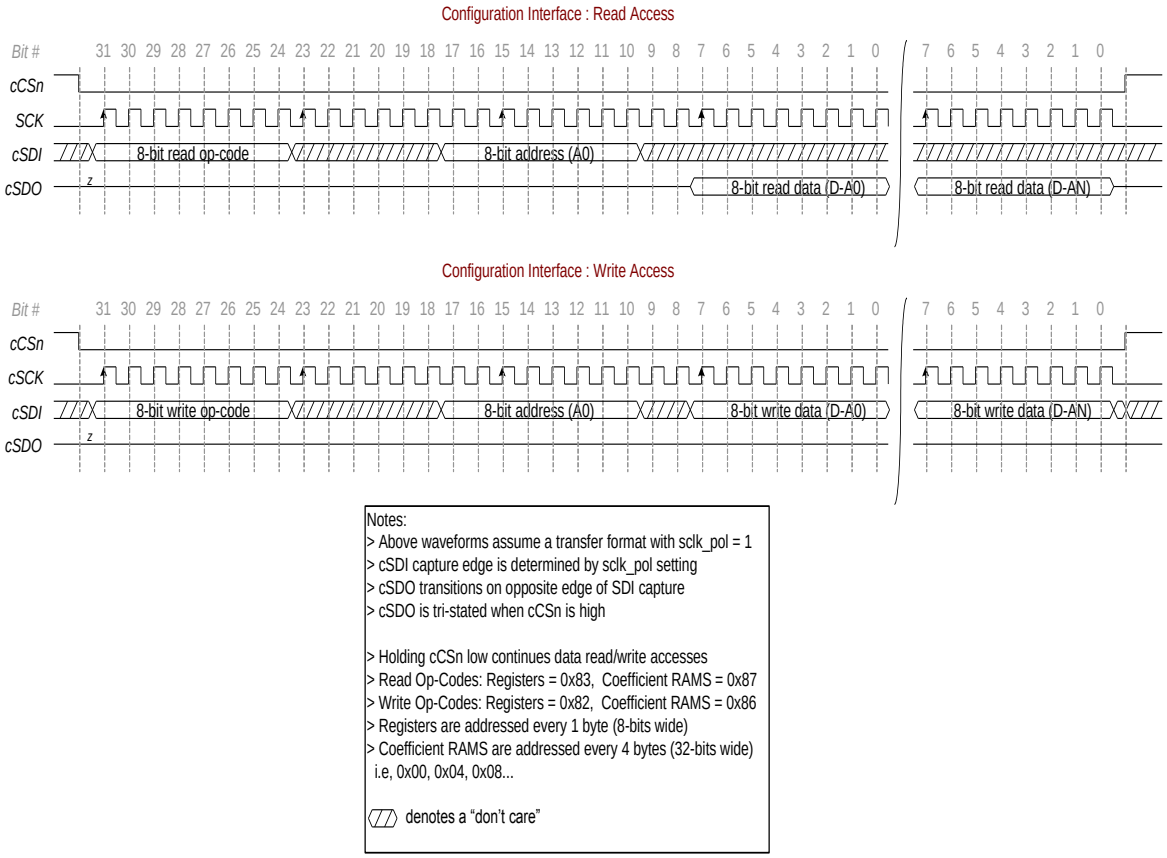


Figure 3: Configuration Data Timing

Note: There are 6 "don't care" clock cycles between the end of the op code and the 8-bit address. The value of cSDI during these clocks has no effect on the chip. There are also two "dead" clock cycles between the end of the address input and data output on the cSDO pin.

The Configuration SPI interface programs two different address spaces in the QF1Da512: a configuration register space and a coefficient memory space. The configuration register space is accessed when the Op-Codes 0x82/0x83 are used (refer to the notes above). The coefficient memory space is accessed when the Op-Codes 0x86/0x87 are used. Multiple successive registers or coefficient values may be programmed by simply appending 8-bit data values after the first data value is written. Thus the entire coefficient memory space may be written with a single Write access. Each 32-bit coefficient is written Most Significant Bit and Most Significant Byte first. Each 8-bit configuration register is written Most Significant Bit first.

The active edge of cSCK is programmable (**SCLK_POL** bit in the **DCONFIG** register) to suit the application system timing:

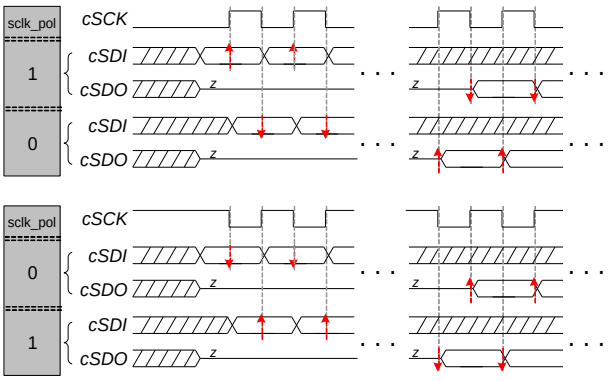


Figure 4: Configuration Interface Transfer Format

Sending Commands in Data Mode

If the QF1Da512 is in Data mode and it is required to reconfigure the device, it is first necessary to switch to Configure mode. This is done by writing to the **FILT_EN** bit using the configuration interface.

Note: While all of the configuration registers are accessible while the QF1Da512 is in Data mode, the only registers that should be written to are the CONFIG register FILT_EN, DIN_PT. Writes to other registers might result in corrupted filter data or other unpredictable behavior.

In configurations in which the **cSDI** and the **dSDI** pins have been tied together, allowing for single SPI port data filtering, it is necessary to ensure that the incoming data format does not match either the configuration register access format (see Table 4 Configuration Register Access Format) or the FIR coefficient memory access format (see Table 2 Register and Memory Format) as this may cause unintentional writes to configuration or coefficient memory. In other words, take care to ensure that the most significant byte of the data input does not match either of the WRITE opcodes.

7 DATA INTERFACE

Modes of Operation

The QF1Da512 supports I2S or SPI and is designed to interface directly with a variety of ADC converters.

1. I2S Mode
   A word select line (**WS** on the **dCS** input) indicates which of two time-division multiplexed data channels is being transmitted. The channels are transmitted alternately on the data line (**SD** on the **dSDI** input) and are synchronized by a clock line (**SCK** on the **dSCK** input).
2. SPI Mode
   a. Normal Mode
      The data is framed by the **dCS** input signal
   b. Continuous Mode
      Once **dCS** is active, data is output corresponding to clock bursts on the **dSCK** input
   c. Synchronous Serial Mode
      dCS is used as a trigger to indicate the start of each new sample period. **dSCK** runs continuously, filtered data is output on the next *n* cycles of **dSCK** according to the programmed word length.

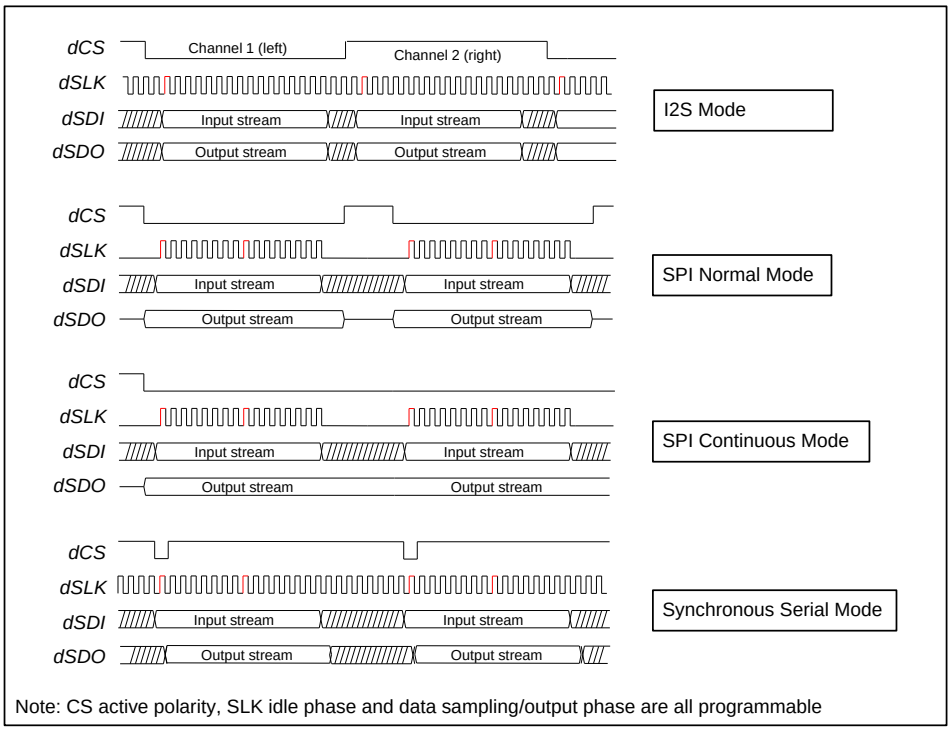


Figure 5: Interface Modes

The QF1Da512 supports both 2's complement and offset binary data formats. The data format is set by the **FORMAT** bit and the data mode is selected by the **MODE0** and **MODE1** bits in the **DCONFIG** register (04h).

MODE1	MODE0	Bus Mode
1	X	(Default) I2S Mode
0	0	SPI Normal Mode
0	1	SPI Continuous Mode
1	X	Synchronous Serial Mode

Figure 6: Data Format Selection

The data interface utilizes the **dSDI** (**SD** for I2S), **dSDO** (**SD** for I2S), **dCS** (**WS** for I2S) and **dSCK** (**SCK** for I2S) pins. **dSDI** is the input serial data, and **dSDO** is the output serial data (if **FILT_EN** = 1). **dSCK** is the input data clock and **dCS** is the framing signal for the input data:

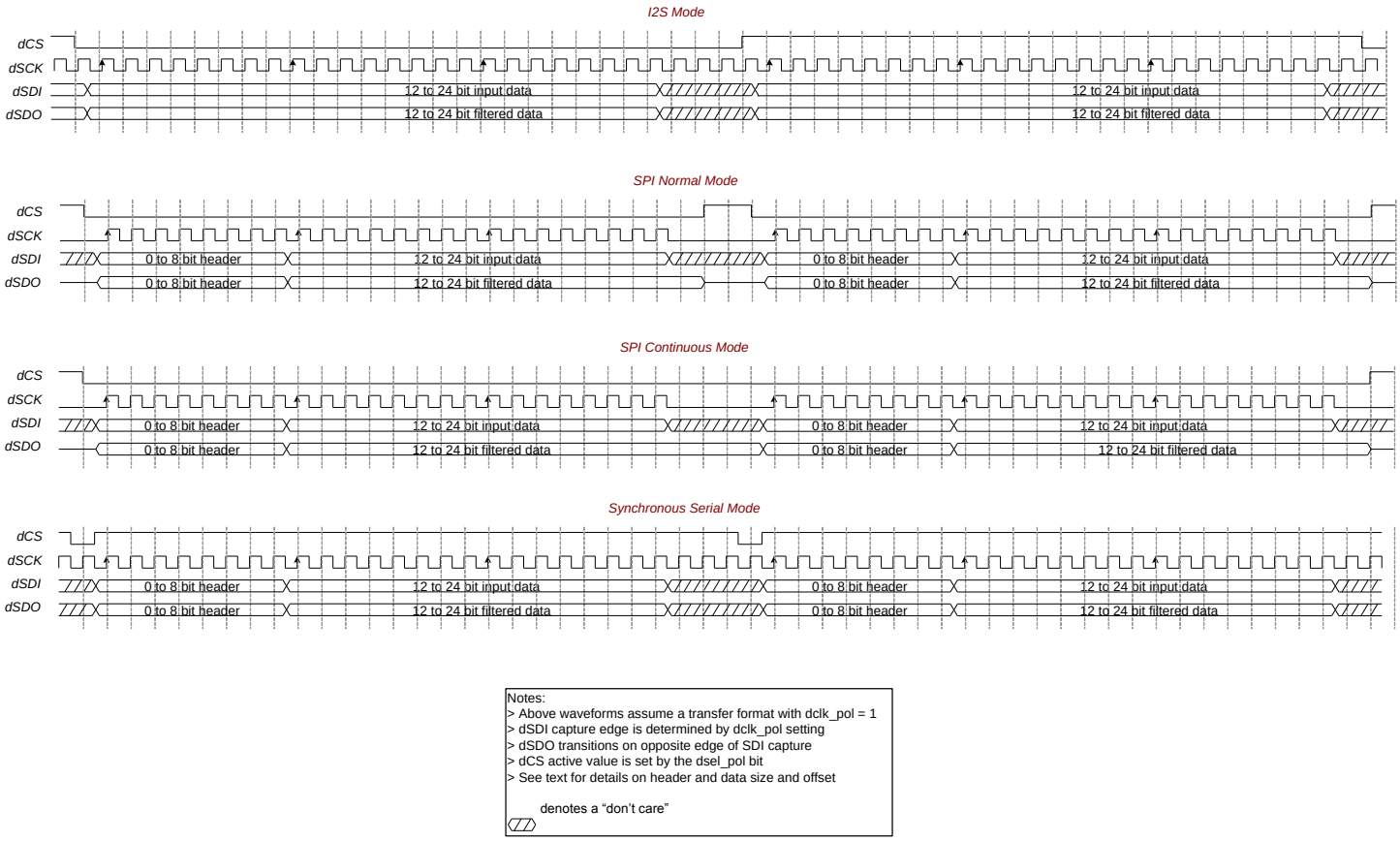


Figure 7: Data Interface Timing Diagram

Note: In "SPI Continuous Mode" the value of the dSDO output (lsb) at the end of the data sample will be held constant until the header is output for the next data sample.

Default operation is to capture on the rising edge of **dSCK** (**SCK** for I2S), but this can be changed to capture on falling edge by writing the **DCLK_POL** bit in the **DCONFIG** register (04h). The active polarity of **dCS** (**WS** for I2S) is also programmable via the **DCONFIG** register (**DSEL_POL** bit), which is especially useful for applications where the input pins are paired; e.g., **dSDI** (**SD** for I2S) and **cSDI**, **dSCK** (**SCK** for I2S) and **cSCK**, **cCSn** and **dCS** (**WS** for I2S)**.**

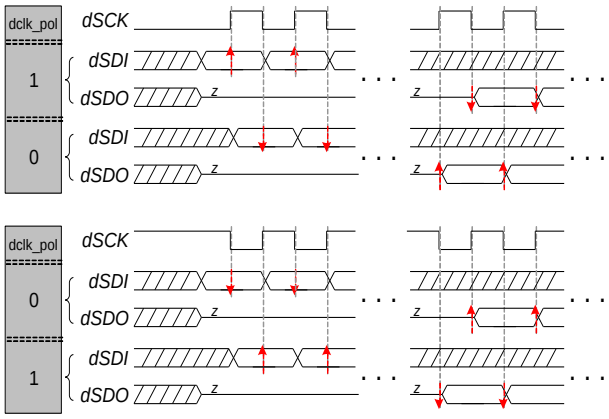
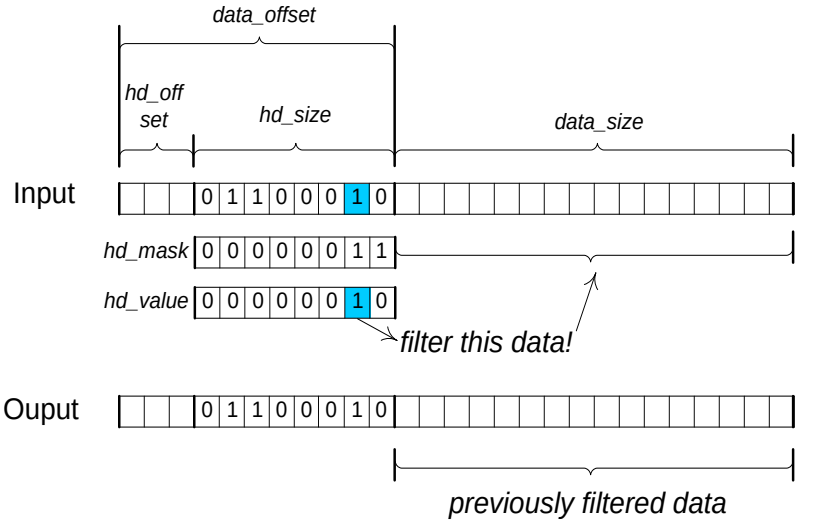


Figure 8: Data Interface Transfer Format

The QF1Da512 supports data widths in the range of 12 and 24 bits, and headers of up to 8 bits.

The format of the input data stream is specified in the **HD_OFFSET** (address 09h), **HD_SIZE** (address 0Ah), **DATA_OFFSET** (address 000Dh), and the **DATA_SIZE** (address 000Eh) registers. These registers set the header offset, header size, data offset, and data size respectively of the individual data sample. This allows for a large number of different data formats. Additionally, if the sample data contains a data valid qualifier, this field can be masked by configuring the **HD_MASK**, address 0Bh and **HD_VALUE**, address 0Ch registers. The filtering of the sample will occur only if a logical *and* of the header value and **HD_MASK** matches **HD_VALUE** , address 0Ch, otherwise the raw sample data is passed directly through to the **dSDO** pin unfiltered.



Header = 01100010,
Data valid qualifier bits are the two least significant bits, therefore hd_mask = 00000011
The data to be filtered has a data qualifier of 10, therefore the hd_value = 00000010
Data is filtered when a logical and of the header and hd_mask = hd_value

Figure 9: Data Header Masking Example

When configured for **FILTER** mode, **FILT_EN** bit set to '1'in the **CONTROL** register (03h), the output stream on the **dSDO** pin will have the same format as the input stream coming in on the **dSDI** pin. The header data is passed directly through to the **dSDO** pin while the filtered data has a latency associated with it due to the properties of the FIR filter. If no filtering is being done (data valid qualifier in header does not match **HD_VALUE**) coincident header and data will be passed through for possible use down-stream of the QF1Da512 device.

When averaging and / or down sampling is active, the data rate will be the same as the incoming data rate, but the number of valid samples will be reduced by the averaging / down sampling rate.

Data Passthrough

Sometimes it is required to pass-through the data present at **dSDI** directly to the **dSDO** output, for example when passing configuration data back from an ADC to the host controller. The data pass-through bit, **DIN_PT**, in the **CONTROL** register (03h), will force this to occur when it is set to 1. In this mode dSDO follows dSDI immediately, there is no clocking required. This control bit overrides all other settings for **dSDO** functionality (see next section) and should be set to 0 for either configuration mode or data filtering.

8 DIGITAL DOWN SAMPLING

Downconverter

The QF1Da512 provides the capability to down-sample the incoming data samples. The divider ratio is any integer value and ranges from 1 (no down-sampling) to 256. The value is selected in the **DECIMATE** register (address 08h). In addition to down-sampling and "throwing away" the intermediate samples, and option is provided to average the samples and forward the averaged value to the downstream FIR filter. The averaging block effectively adds up N samples, where N is the down-sampling rate, then divides by a power of 2 by right shifting the result. Note that if averaging is enabled with a down-sampling rate that is not a power of 2, system gain will be affected. The averaging block always divides by the next highest power of 2 when a *non* power of 2 down-sampling rate is detected.

Refer to the diagram below. This feature can be useful for noise reduction and possibly increasing bit resolution of the input data stream. To select filtering of the down sampled data, **DEC_AVG_EN**, bit 0 of the **FCONFIG** register (address 05h) is enabled.

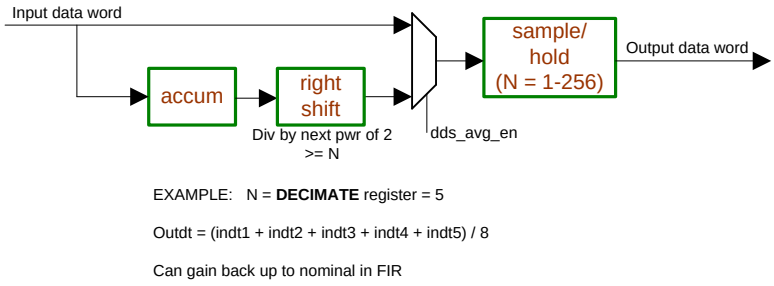


Figure 10: Averaging /Down-sampler Block Diagram

Channel Duplication

Channel duplication is a special feature in I2S mode (DECIMATE must also be set to 00h). Duplication uses the standard left-right channel framing to create 2 channels of either right or left data. Using 2 QF1Da512s, the first device will both filter the specified channel and duplicate (unaltered) the same channel's input data to the opposite channel's frame on dSDO. The second device will be programmed to filter the unaltered data channel (and the previously filtered data will be passed to dSDO). For example, if the first device is in Duplication mode, left-channel active and functions as a low pass filter, then the output of the first device will be Left-lowpass/Left-raw-data (channel-1/channel-2). The second device is in Standard mode, right-channel active and functions as a high pass filter, then the output of the second device will be Left-lowpass/Left-highpass.

9 FIR FILTER

Modes of Operation

The QF1Da512 filter can implement a symmetric 512-tap FIR filter or a 256-tap asymmetric FIR filter, which is used to define the precise filtering characteristics desired. The filter coefficients are 32 bits wide. The filtering provided by the software may be of the following types: low-pass, notched low-pass, high-pass, band-pass, dual band-pass, band-stop, and dual band-stop. Currently available filter algorithms include Parks-McClellan and Windowed Sync. User defined filters can also be implemented.

The QF1Da512 can be loaded with up to 256 filter coefficients. To implement a non-folded (asymmetric) filter the **FIR_FOLD** bit in the **FCONFIG** register (05h) should be set to 0, allowing a filter of up to 256 taps. A folded (symmetric) filter of up to 512 taps can be implemented if **FIR_FOLD** is set to 1.

Note: For a folded filter, if the number of coefficients (N) is even, a filter of 2N taps will be realized. If N is odd, a filter of 2N-1 taps will result.

The Quickfilter software allows the user to enter the filter characteristics required and see the predicted performance in terms of frequency and impulse response. Once the desired performance has been attained, the configuration can be downloaded to the QF1Da512, and the actual filter performance verified, by using the development kit. The development board can be fed with a white noise source (or other source as desired by the user) and the software can display an FFT of the QF1Da512's filter response.

FIR Latency

The delay introduced by the Qf1Da512 is dependent on the number of taps of the filter and the data rate of the samples coming into the FIR filter. The delay is calculated by the following equation:

Standard Mode

Filtered Channel: FIR Delay = (# taps – 1)/2 * $1/f_s$ + $8/f_s$

Bypassed Channel: Delay = 0

Duplication Mode

Filtered Channel: FIR Delay = (# taps – 1)/2 * $1/f_s$ + $12/f_s$

Replicated Channel: FIR Delay = $3/f_s$

In standard mode for data sampling at a 48 kHz rate, the latency will result in:

512 tap (maximum) filter:	Latency = 5.49ms
100 tap filter:	Latency = 1.20ms

In duplication mode for data sampling at a 48 kHz rate, the latency will result in:

Two 512 tap (maximum) filters in series:	Latency = 5.57ms
Two 100 tap filters in series:	Latency = 1.28ms

Note: If using the Quickfilter Pro development software the latency of a particular filter is calculated and displayed in the Information bar on the "Filter" tab.

Duplication mode is comprised of two serial QF1Da512s of which the first device is in duplication mode and the second device is in standard mode.

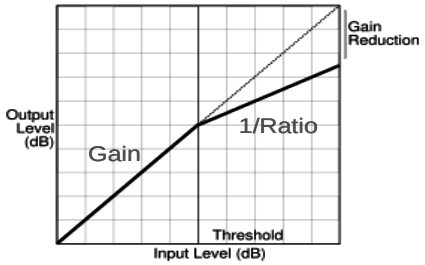
10 Digital Gain and Compression (DGC)

For standard FIR pass-band/stop-band filters, DGC is not necessary as the pass-band is designed for 0dB (unity). The Quickfilter Pro™ Freehand filter editor supports an audio mode where the gain and compression variables are automatically determined for the user. Optionally, Quickfilter Pro™ provides direct access to and guidance for all the DGC parameters.

Mode of Operation

When Compression is enabled, Gain is applied to the FIR output signal when the input signal is below a Threshold value. Above the Threshold value, Gain and Compression are applied to the FIR output signal. When the Compressor is bypassed (**THRES** = FFh), only Gain is applied to the FIR output signal. The default value for Gain is 1.

Both Gain and Compression (ratio) are comprised of 4 integer bits and 12 factional bits.



The DGC parameters are,

1. Threshold – signal levels above this value are compressed
2. Gain – the amount of gain to apply to the FIR output signal below the Threshold
3. Ratio – the inverse slope of the compression curve above the Threshold
4. Maximum – Maximum input amplitude value

Note – not all combinations of DGC parameters yield valid compression values settings. Therefore, it is best to use Quickfilter Pro™ to properly set these parameters. Also, care must be taken to avoid clipping when using the DGC function. Additional guidance can be found in section 14.

11 CONTROL REGISTERS

Overview

The QF1Da512 internal data is separated into configuration registers and coefficient memory.

Table 2 Register and Memory Format

Registers	Address Range	Data Size	Write OpCode	Read Opcode
Device Configuration	0x0000 – 0x0021	8 bit	0x82	0x83
FIR Coefficients	0x0000 – 0x00FF	32 bit	0x86	0x87

Table 3 Control Registers

Hex*	Register Name	Description
0000h	**TEST_RW**	Software Register, Test Reads and Writes
0001h	**CHIP_ID**	Chip ID Number
0002h	**VERSION**	Chip Version Number
0003h	**CONTROL**	Mode Control Register
0004h	**DCONFIG**	Data Format Control
0005h	**FCONFIG**	Filter Control and Status
0006h	**NUM_TAPS**	Number of Taps, lower bits
0007h	**NUM_TAPS**	Number of Taps, upper bits
0008h	**DECIMATE**	Down-sampling Rate
0009h	**HD_OFFSET**	Header Offset
000Ah	**HD_SIZE**	Header Length
000Bh	**HD_MASK**	Header Valid Mask
000Ch	**HD_VALUE**	Header Valid Value
000Dh	**DATA_OFFSET**	Data Offset
000Eh	**DATA_SIZE**	Data length
000Fh	**GAIN**	Digital Gain Value, lower bits
0010h	**GAIN**	Digital Gain Value, upper bits
0011h	**THRESH**	Compression Threshold, lower bits
0012h	**TREASH**	Compression Threshold, upper bits
0013h	**MULTI**	Compression Multiplicor, lower bits
0014h	**MULTI**	Compression Multiplicor, upper bits
0015h	**ADDOR**	Compression Addor, lower bits
0016h	**ADDOR**	Compression Addor, upper bits
0017h	**IO_TST**	TST Pin Configuration
0018h	**IO_RST_N**	RSTn Pin Configuration
0019h	**IO_DCLK**	dSCK Pin Configuration
001Ah	**IO_DSEL**	dCS Pin Configuration
001Bh	**IO_DIN**	dSDI Pin Configuration
001Ch	**IO_CS_N**	cCSn Pin Configuration
001Dh	**IO_SCLK**	cSCK Pin Configuration
001Eh	**IO_SDI**	cSDI Pin Configuration
001Fh	**IO_SDO**	cSDO Pin Configuration
0020H	**IO_DOUT**	dSDO Pin Configuration
0021H	**TEST**	Chip Test Register

Configuration Registers

Accessing the configuration registers uses an SPI access as shown below.

Table 4 Configuration Register Access Format

	OPCODE	ADDRESS	UNUSED	DATA
Number of bits	8 bits	14 bits	2 bits	8 bits per register†
Write Access	82h	0000h – 0021h	Don't care	00h – FFh
Read Access	83h	0000h – 0021h	Don't care	00h – FFh

Note: * Multiple registers can be written/read by extending the SPI access cycle.

Listed below are the detailed configuration register definitions.

→ 1 00h TEST_RW (User Register)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	D7	D6	D5	D4	D3	D2	D1	D0
D7-D0	Provided as a blank user byte for the programmer to read and write to as a test. This byte defaults to 0 at power up.							

→ 2 01h CHIP_ID (Chip ID) - READ ONLY

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	1	1	0	0	0	0	0	0
	R	R	R	R	R	R	R	R
Bits	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
ID7-ID0	This read-only byte contains a number describing the identification of the QF1Da512 device. Identification number of the QF1Da512, default = C0h.							

→ 3 02h VERSION (Version) - READ ONLY

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	1
	R	R	R	R	R	R	R	R
Bits	VER7	VER6	VER5	VER4	VER3	VER2	VER1	VER0
VER7-VER0	This read-only byte contains a number describing the version of the QF1Da512 device. Version number of the QF1Da512, default = 01h.							

→ 4 03h CONTROL (Filter/Pass Through Control)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	RW	RW
Bits	X	X	X	X	X	X	DIN_PT	FILT_EN
Bit7-2	Unused							
DIN_PT	Data In Pass Through (DIN_PT) : Puts the QF1Da512 in Bypass mode 0 = Normal operation, dSDO pin outputs the signal path data. 1 = Data pass through, dSDO pin outputs the data on the dSDI pin							
FILT_EN	Filter Enable (FILT_EN) : Enable the Signal processing blocks and data path 0 = Configuration mode, cSDO pin outputs the configuration register. 1 = Filter mode, dSDO pin outputs the signal path data.							

→ 5 **04h DCONFIG** (Data Format Control)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	1	0	0	1	0	1	1
	RW	RW	RW	RW	RW	RW	RW	RW
Bits	DUP	MODE1	MODE0	FORMAT	dCS_POL1	dCS_POL0	dSCK_POL	cSCK_POL
DUP	Duplicates (but does not alter) dSDI on channel selected by dCS_POL to output dSDO. Requires DECIMATE (08h) =00h and dCS as per above. 0 = Input data is not duplicated. 1 = Input data is duplicated.							
MODE1 – MODE0	Selects the data input mode. 00 = Input data mode is SPI Normal Mode. 01 = Input data mode is SPI Continuous Mode. 10 = Input data mode is Synchronous serial mode 10 = Input data mode is I2S Mode							
FORMAT	Selects the data format of the incoming data. 0 = Incoming data format is 2's complement. 1 = Incoming data format is offset binary.							
dCS_POL1-dCS_POL0	Selects polarity of the dCS pin for SPI and Synchronous serial modes. Also used in processing I2S data if the DUP bit is set. 00 = dCS is active low for SPI modes and Synchronous serial mode when only a single edge/level is being used as a trigger 01 = dCS is active high for SPI modes and Synchronous serial mode when only a single edge/level is being used as a trigger 1X = dCS is active both high and low. This is the normal mode for I2S data. If DUP = 1 (Duplication Mode for I2S formatted data) 10 = Filter the incoming left channel data (the data where dCS is low) and place the filtered data on the dSDO left channel output. Copy (unaltered) the incoming left channel data to the right channel on dSDO. 11 = Filter the incoming right channel data (the data where dCS is low) and place the filtered data on the dSDO right channel output. Copy (unaltered) the incoming right channel data to the left channel on dSDO.							
dSCK_POL	Selects the clock edge of dSCK on which the data on the dSDI pin is captured. dSDO is output on the opposite edge (Filter mode). 0 = dSDI is captured on the falling edge of dSCK. 1 = dSDI is captured on the rising edge of dSCK.							
cSCK_POL	The clock edge of cSCK on which the data on the cSDI is captured. cSDO is output on the opposite edge. 0 = cSDI data is captured on the falling edge of cSCK. 1 = cSDI data is captured on the rising edge of cSCK.							

→ 6 05h FCONFIG (Filter Control)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	1	0	0	0	0	1	0
	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	RW	RW	RW
Bits	X	X	X	X	X	OVF_FLAG	FIR_FOLD	DEC_EN
OVF_FLAG	Indicates that the FIR filter has overflowed. To clear, write a '0' to this bit. 0 = FIR filter has not overflowed. 1 = FIR filter has overflowed.							
FIR_FOLD	Selects whether the FIR filter is symmetric (folded) or asymmetric. 0 = FIR Filter is not folded (asymmetric, 256 taps maximum). 1 = FIR Filter is folded (symmetric, 512 taps maximum).							
DEC_EN	Selects the averaging of down-sampled data. Setting this bit causes a composite sum of the incoming sample values to be used instead of a single sample (where the others are ignored), as is typically the case with downsampling. When this bit is set, N samples are summed (determined by the DECIMATE register) as they come in. Then their value is divided by a POWER OF 2 based on the value of the DECIMATE register. This divide value is 2 raised to the DECIMATE register's most significant bit that is set, plus 1. For example, if the DECIMATE register contains FF hex, and DEC_EN is enabled, the Decimate block will sum 255+1 samples, then divide that value by 2 raised to the (7+1) = 256 value. This will result in a low pass filtered downsampling algorithm. Because the divide value is limited to powers of two, it is best to use this feature in conjunction with DECIMATE values that are 2^N – 1. DEC_EN must be disabled when the chip is used with any dual channel format. 0 = Averaging of down-sampled data is disabled. 1 = Averaging of down-sampled data is enabled.							

→ 7 06h NUM_TAPS (Number of FIR Filter Taps, Lower Bits)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	1	1	1	1	1	1	1	1
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	NTAP7	NTAP6	NTAP5	NTAP4	NTAP3	NTAP2	NTAP1	NTAP0
NTAP7-NTAP0	Sets the number of taps in lower byte of the FIR filter. Number of taps in the FIR filter, range from 0 to 511 (= 1 – 512 taps), default = 1FFh (512 Taps).							

→ 8 07h NUM_TAPS (Number of FIR Filter Taps, Upper Bit)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	1
	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	W/R
Bits	X	X	X	X	X	X	X	NTAP8
NTAP8	Sets the number of taps in upper bit of the FIR filter. Number of taps in the FIR filter, range from 0 to 511 (= 1 – 512 taps), default = 1FFh (512 Taps).							

→ 9 08h DECIMATE (Down-sampling Rate)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	DEC7	DEC6	DEC5	DEC4	DEC3	DEC2	DEC1	DEC0
DEC7-DEC0	Sets the down-sampling *rate* from 1 (no down-sampling) to 256. The downsampling block only processes every nth incoming sample, and ignores all of the others. If the input signal has frequencies above ½ of the downsampled rate, aliasing of the upper frequencies will occur into the lower band. This is usually not desirable. Note that the actual output frame rate is not reduced. Instead, the output value is duplicated X number of times on the dSDO output, X being the downsampling rate plus 1. The register must be set to 0 if DUP = 1 or if this chip is used with any dual channel formats. 00h = No down sampling, Else = Down-sampling rate, *range* varies from 1 to 255 (for down-sampling *rate* of 2 to 256 respectively)							

→ 10 09h HD_OFFSET (Offset of Header Field)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	HDOFF7	HDOFF6	HDOFF5	HDOFF4	HDOFF3	HDOFF2	HDOFF1	HDOFF0
HDOFF7-HDOFF0	Sets the offset in bits in the data word of the header. Header field offset in bits, range 0 to 255, default = 0h.							

→ 11 0Ah HD_SIZE (Size of Header Field)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	UNUSED	UNUSED	UNUSED	UNUSED	W/R	W/R	W/R	W/R
Bits	X	X	X	X	HDSIZ3	HDSIZ2	HDSIZ1	HDSIZ0
HDSIZ7-HDSIZ0	Sets the size of the header field Header field size in bits, range 0 to 8, default = 0h.							

→ 12 0Bh HD_MASK (Header Data Valid Mask)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	HDMSK7	HDMSK6	HDMSK5	HDMSK4	HDMSK3	HDMSK2	HDMSK1	HDMSK0
HDMSK7-HDMSK0	Sets the bits of the header which are used to indicate the data valid condition of the sample. Header data valid bits, default = 0h.							

→ 13 0Ch HD_VALUE (Header Data Valid Value)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	HDDV7	HDDV6	HDDV5	HDDV4	HDDV3	HDDV2	HDDV1	HDDV0
HDDV7-HDDV0	Sets the pattern in the header which indicates that the data is valid. This value has the mask defined in the HD_MASK register (0Bh) applied to it. Header data valid bit pattern, default = 0h.							

→ 14 0Dh DATA_OFFSET (Offset of Data Field)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	DAOFF7	DAOFF6	DAOFF5	DAOFF4	DAOFF3	DAOFF2	DAOFF1	DAOFF0
DAOFF7-DAOFF0	Sets the offset to the data field in the incoming data. Data field offset in bits, range 0 to 255, default = 0h. This value must be set to 01h when in an I2S mode of operation.							

→ 15 0Eh DATA_SIZE (Number of Bits in the Data Field)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	1	0	0	0	0
	Unused	Unused	Unused	W/R	W/R	W/R	W/R	W/R
Bits	X	X	X	DASIZ4	DASIZ3	DASIZ2	DASIZ1	DASIZ0
DASIZ4-DASIZ0	Sets the number of bits in the data field. Only values 12 through 24 are valid. Unpredictable results may occur for values outside this range. field size in bits, range 12 to 24 bits, default 10h (16 bits).							

→ 16 0Fh GAIN (Digital Gain Value, Lower Bits)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	GAIN7	GAIN6	GAIN5	GAIN4	GAIN3	GAIN2	GAIN1	GAIN0
GAIN7-GAIN0	Sets the Compressor gain when the signal is not being compressed. The format is unsigned with 4 integer bits and 12 fractional bits.							

→ **17** **10h GAIN** (Digital Gain Value, Upper Bit)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	1	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	GAIN15	GAIN14	GAIN13	GAIN12	GAIN11	GAIN10	GAIN9	GAIN8
GAIN15-GAIN8	Sets the Compressor gain when the signal is not being compressed. The format is unsigned with 4 integer bits and 12 fractional bits.							

→ **18** **11h TRESH** (Compressor Threshold Value, Lower Bits)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	1	1	1	1	1	1	1	1
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	THRES7	THRES6	THRES5	THRES4	THRES3	THRES2	THRES1	THRES0
THRES7-THRES0	Sets the threshold for compressor operation. The format is unsigned with 0 integer bits and 16 fractional bits.							

→ **19** **12h THRESH** (Compressor Threshold Value, Upper Bit)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	1	1	1	1	1	1	1	1
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	THRES15	THRES14	THRES13	THRES12	THRES11	THRES10	THRES9	THRES8
THRES15-THRES8	Sets the threshold for compressor operation. The format is unsigned with 0 integer bits and 16 fractional bits.							

→ **20** **13h MULTI** (Compressor Multiplicor Value, Lower Bits)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	MUL7	MUL6	MUL5	MUL4	MUL3	MUL2	MUL1	MUL0
MUL7-MUL0	Works in conjunction with the **GAIN** and **THRES** registers to set the compressor gain when the signal is being compressed. The format is unsigned with 4 integer bits and 12 fractional bits. Refer to the compression formula for more detail.							

→ **21** **14h MULTI (**Compressor Multiplicor Value, Upper Bit)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	1	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	MUL15	MUL14	MUL13	MUL12	MUL11	MUL10	MUL9	MUL8
MUL7-MUL0	Works in conjunction with the **GAIN** and **THRES** registers to set the compressor when the signal is being compressed. The format is unsigned with 4 integer bits and 12 fractional bits. Refer to the compression formula for more detail.							

→ **22** **15h ADDOR** (Compressor Addor Value, Lower Bits)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
ADD7-ADD0	Works in conjunction with the **GAIN** and **THRES** registers to set the compressor gain when the signal is being compressed. The format is unsigned with 0 integer bits and 16 fractional bits. Refer to the compression formula for more detail.							

→ 23 16h ADDOR (Compressor Addor Value, Upper Bit)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	0	0	0	0	0	0	0
	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R
Bits	ADD15	ADD14	ADD13	ADD12	ADD11	ADD10	X ADD9	ADD8
ADD7-ADD0	Works in conjunction with the **GAIN** and **THRES** registers to set the compressor gain when the signal is being compressed. The format is unsigned with 0 integer bits and 16 fractional bits. Refer to the compression formula for more detail.							

→ 24 17h – 1Eh IO_TDT, IO_RST_N, IO_DCLK, IO_DSEL, IO_DIN, IO_CS_N, IO_SCLK, IO_SDI

(TST, RSTn, dSCK, dCS, dSDI, cCSn, cSCK, cSDI Pin Configuration)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR IO_TDT (17h)	0	0	0	0	0	0	1	1
POR IO_RST_N (18h)	0	0	0	0	0	1	0	1
POR IO_DCLK (19h)	0	0	0	0	0	0	0	1
POR IO_DSEL (1Ah)	0	0	0	0	0	0	0	1
POR IO_DIN (1Bh)	0	0	0	0	0	0	0	1
POR IO_CS_N (1Ch)	0	0	0	0	0	1	0	1
POR IO_SCLK (1Dh)	0	0	0	0	0	0	0	1
POR IO_SDI (1EH)	0	0	0	0	0	0	0	1
	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	RW	RW	RW
Bits	X	X	X	X	X	PU_EN	PD_EN	SMT_EN
PU_EN	Enables the pull down resistor on the pin. 0 = No pull up resistor on input. 1 = 75K Ohm pull up resistor on input							
PD_EN	Enables the pull down resistor on the pin. 0 = No pull down resistor on input. 1 = 75K Ohm pull down resistor on input.							
SMT_EN	Enables the Schmitt trigger input on the pin. 0 = Normal input. 1 = Schmitt trigger input.							

→ 25 1Fh – 20h IO_SDO, IO_DOUT (cSDO, dSDO Pin Configuration)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR IO_SDO (1Fh)	0	0	0	0	0	0	1	0
POR IO_DOUT (20h)	0	0	0	0	0	1	0	0
	UNUSED	UNUSED	UNUSED	UNUSED	UNUSED	RW	RW	RW
Bits	X	X	X	X	X	DR2MA	DR4MA	SLEW
DR2MA, DR4MA	The output driver current for the pin. DR2MA DR4MA 0 0 2ma 1 0 4ma 0 1 6ma 1 1 8ma							
SLEW	0 = Slow slew rate. 1 = Fast slew rate.							

→ 26 21h TEST (Chip Test Register) – RESERVED

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POR	0	1	0	0	0	0	0	0
	RES	RES	RES	RES	RES	RES	RES	RES
Bits	X	X	X	X	X	X	X	X
Bit7-1	**The register is reserved and should not be accessed. Unpredictable results may occur if accessed.**							

Coefficient Memory

Accessing the coefficient memory uses an SPI access as shown below.

Table 5 FIR Coefficient Memory Access Format

	OPCODE	ADDRESS	UNUSED	DATA
Number of bits	8 bits	14 bits	2 bits	32 bits per memory location*
Write Access	86h	0000h – 00FFh	Don't care	0000h – FFFFh
Read Access	87h	0000h – 00FFh	Don't care	0000h – FFFFh

Note: * Multiple memory locations can be written/read by extending the SPI access cycle.

0000h – 0400h COEF_RAM (FIR Filter Coefficient Memory)

	Bit 31 – Bit 0
Address 0000h – 00FFh	Data 31 – Data 0

Description: FIR filter coefficient memory.

12 Operating Mode Configurations

Note: Data paths in **bold**, Control paths in regular. Filter data in **blue**, configuration data in **orange**.

12.1 I2S Mode

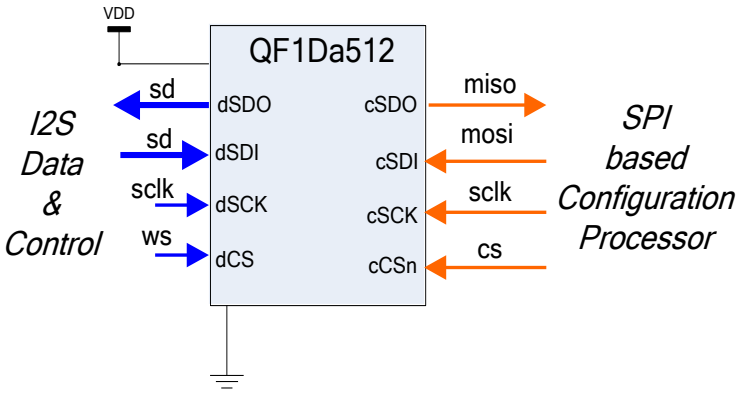


Figure 11: I2S Mode

12.2 SPI Coprocessor Mode

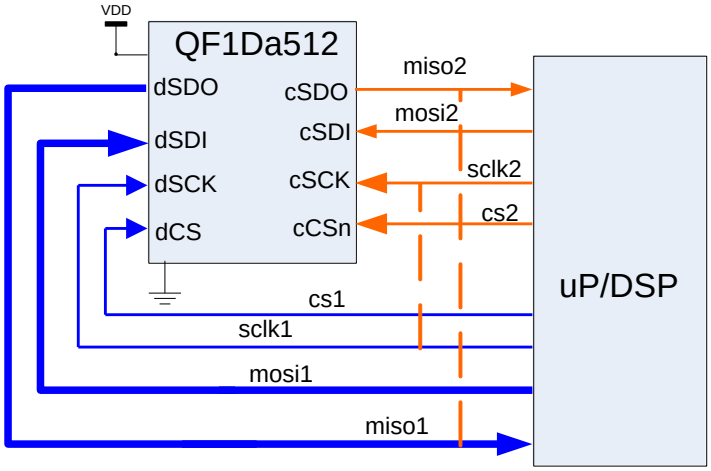


Figure 12: SPI Coprocessor Mode

Dashed lines are alternate connections

12.3 Normal SPI Mode

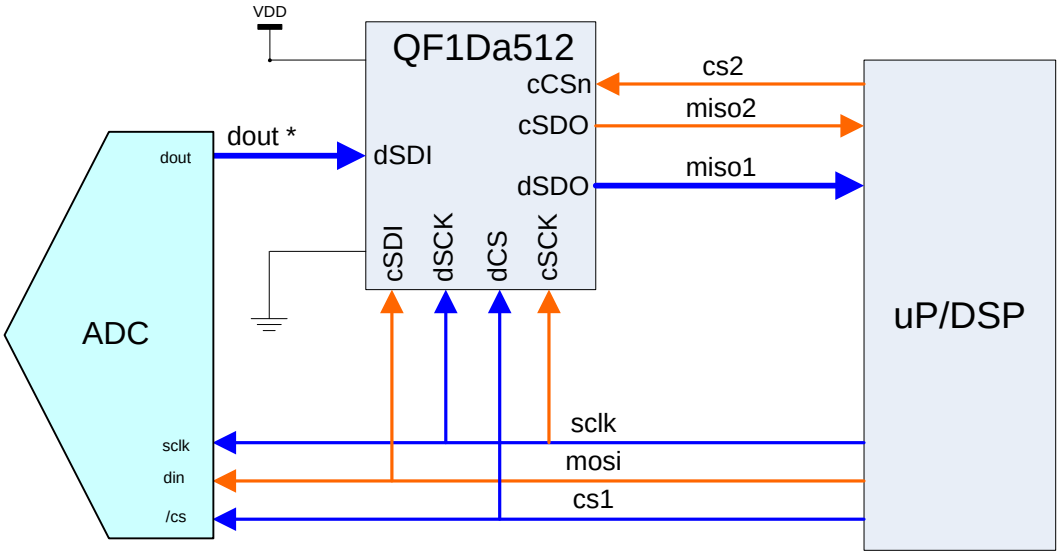


Figure 13: Inline Normal SPI Mode

12.4 Continuous SPI Mode (CS_N tied to GND)

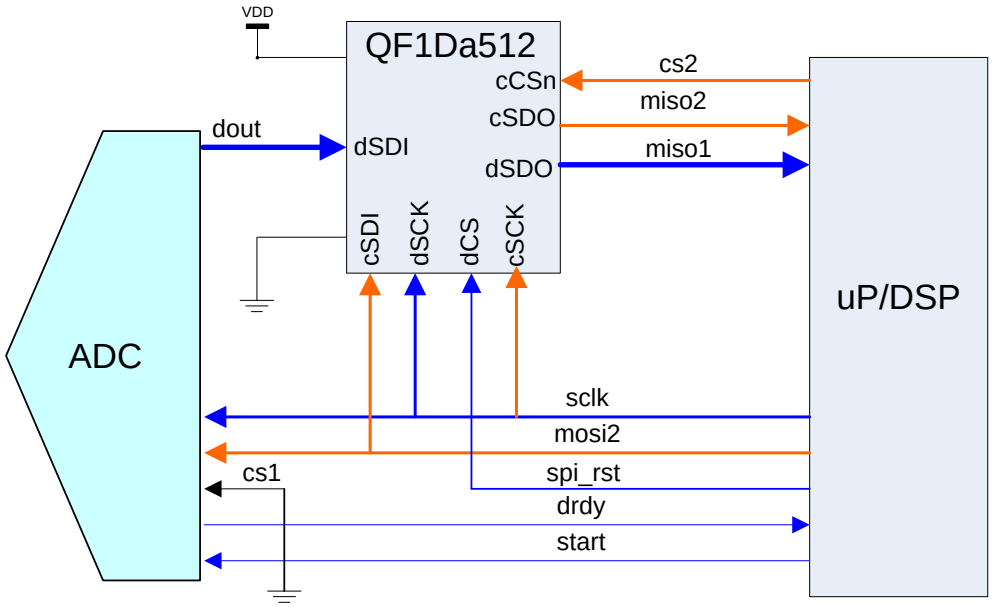


Figure 14: Inline Continuous SPI Mode

12.5 Synchronous Serial Mode (uP clock slave)

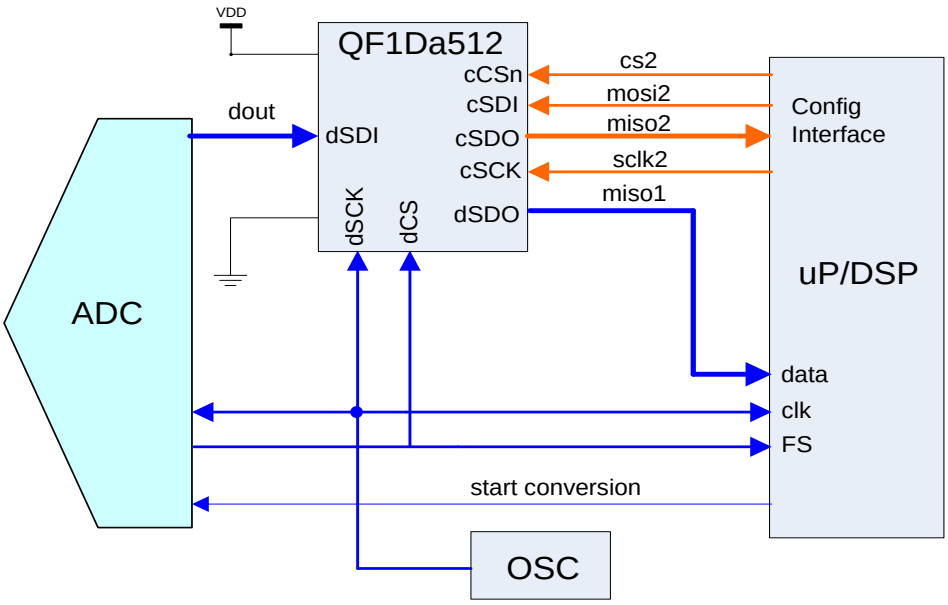


Figure 15: Inline Synchronous Serial Mode

12.6 Inline SPI DAC

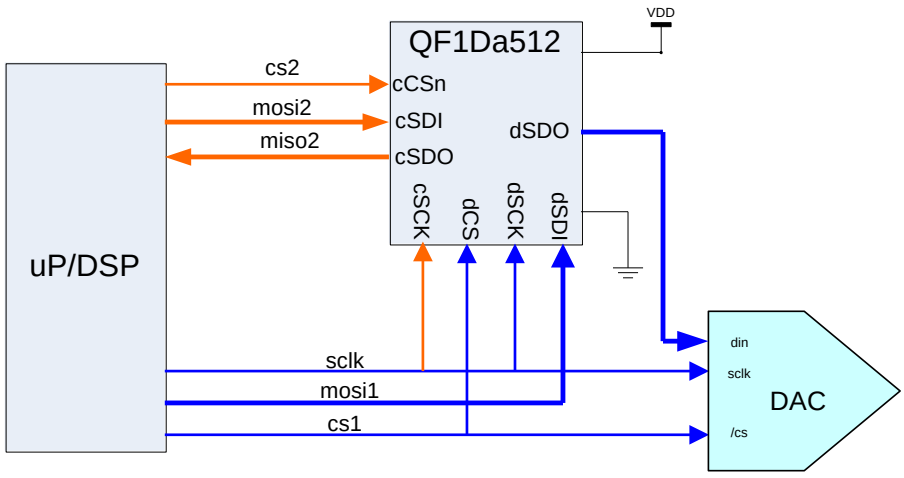


Figure 16: Inline SPI DAC

12.7 ADC Configuration Read / Write – QF1Da512 Passthrough

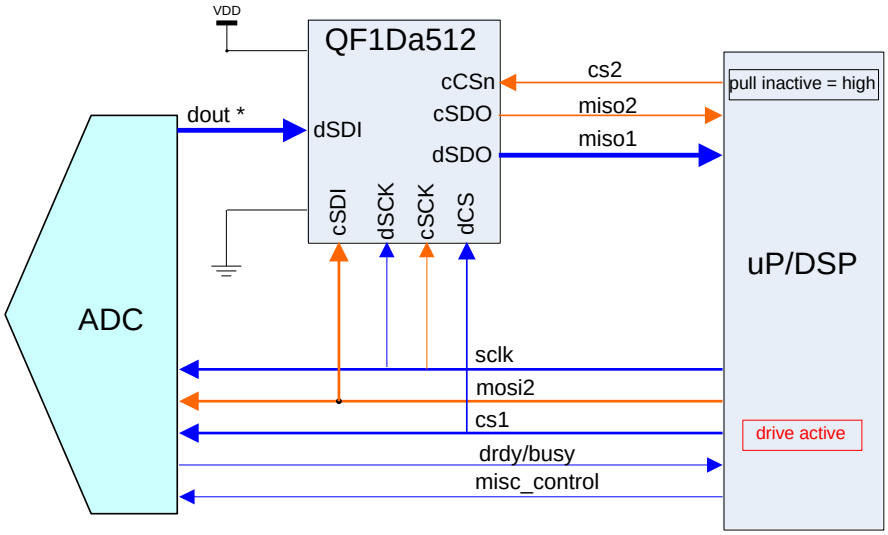


Figure 17: ADC Configuration Read / Write

12.8 QF1Da512 Configuration Mode

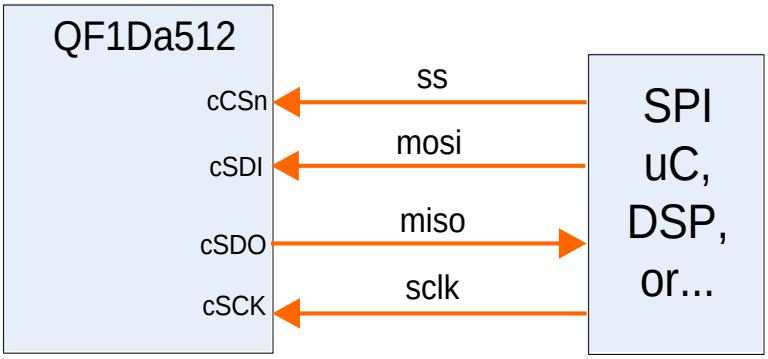


Figure 18: QF1Da512 Configuration, only SPI supported

13 I2C Option

13.1 Mode of Operation

It is relatively straightforward to interface to and control the QF1Da512 with an Inter-IC (I2C) master. Both I2C and I2S/SPI interface protocols use a Master/Slave hierarchy and synchronous data transmission, with the I2S/SPI interface using two separate data lines (one input and one output) and the I2C interface using a single bi-directional data line.

The most significant difference in these two interface mechanisms lies in their addressing schemes. Each I2S/SPI device has its own chip select (CS) input. The I2S/SPI master selects any given I2S/SPI slave device by activating its chip select (for the QF1Da512, pulling it low). At any given time, only one I2S/SPI slave will be selected, so any commands sent are responded to by that device. The I2C interface does not use chip selects. All I2C slave devices receive every command and the addressing scheme, imbedded in the software protocol, determines which slave is to respond to the command (see figure 19).

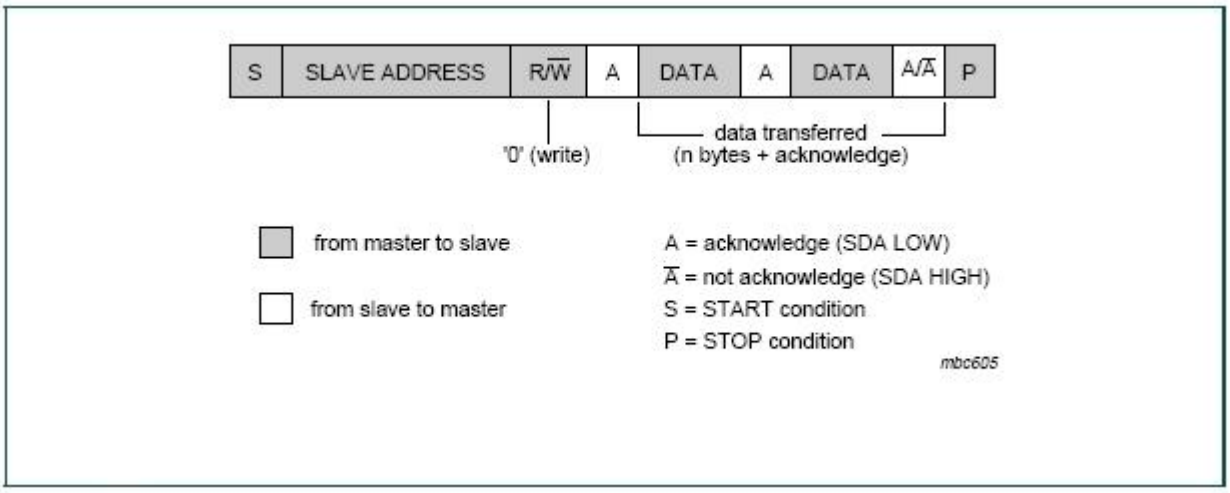


Figure 19: I2C Master/Slave Interaction

Each slave must detect its own address and respond only to those commands intended for it. For this reason, the I2C protocol includes arbitration and collision detection. While an I2S/SPI command is a single stream of bits from the master to the slave (and, for a READ command, a stream of bits back to the master), an I2C command is a succession of bit-string transfers and Acknowledge/Not-Acknowledge signals, interspersed throughout the command (see figure 20).

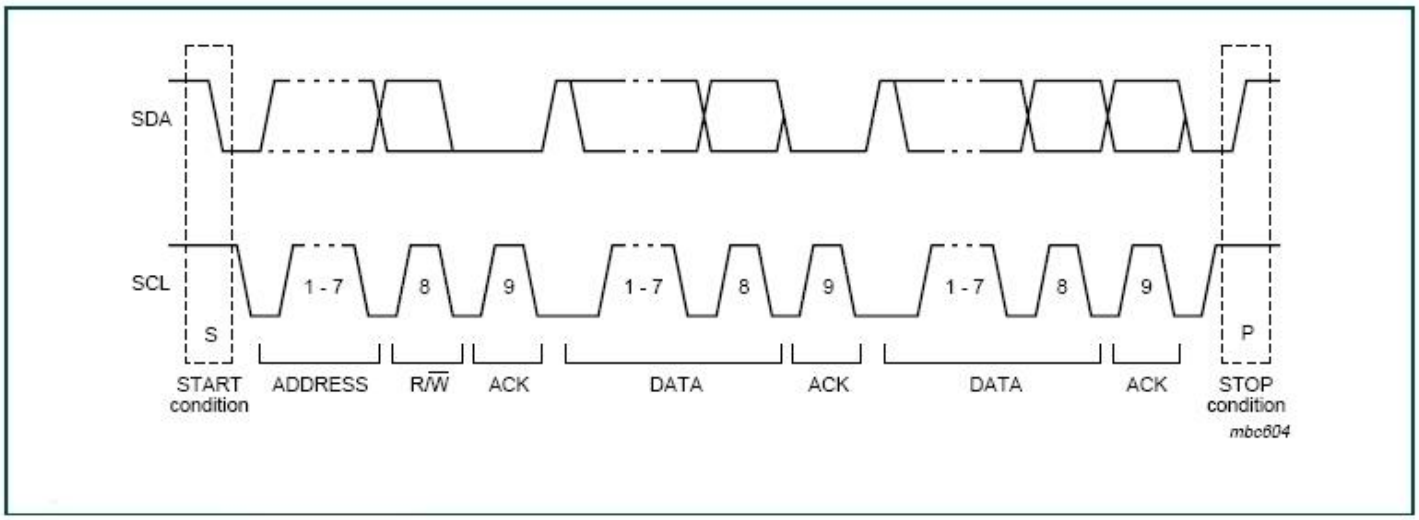


Figure 20: I2C Master/Slave Command/Response Timing

In interfacing the QF1Da512 to an I2C bus, the presence of an ACK/NAK bit approximately every 8th location in the I2C data stream, must be properly handled with respect to both the I2C master and the QF1Da512. This requires a response from the QF1Da512 (or at least, a simulated response from the interface to the QF1Da512). From the perspective of the QF1Da512, the presence of the ACK/NAK bits interferes with the QF1Da512's recognition of the address and data fields within the command and must be removed from the command stream.

Successfully interfacing the QF1Da512 to an I2C bus requires the use of a bridge circuit (see figure 21). The bridge can be implemented entirely in hardware (e.g., logic gates in a PLD or PAL) or through a combination of hardware and software (e.g., a low-cost microcontroller with embedded code). In either case, this circuit must recognize and respond to both the start and stop conditions; reply automatically back to the I2C master with ACK's (signifying that all is well and ensuring that the master continues with the transmission); and, strip the ACK's from the stream of data sent on to the 1Da (the data received by the 1Da will look like a normal I2S/SPI command).

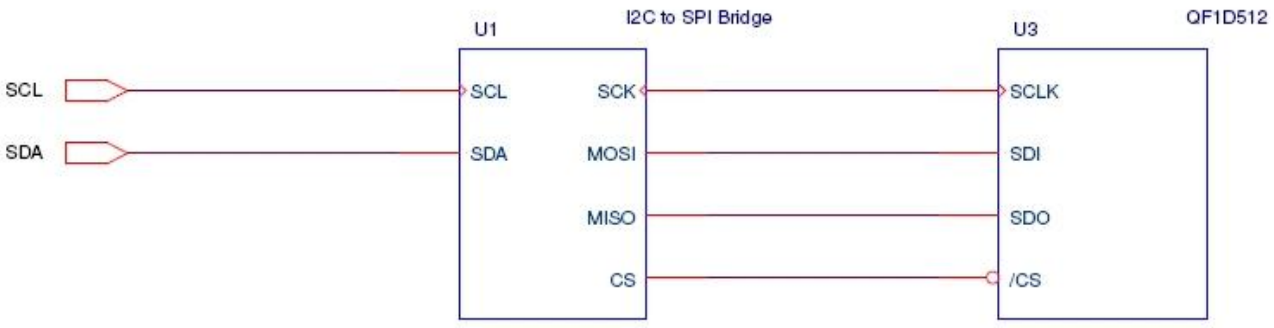


Figure 21: I2C to SPI Bridge Circuit

The I2C-to-I2S/SPI bridge will look to the I2C master like an I2C slave and to the QF1Da512 like an I2S/SPI master. It will intercept the I2C commands, responding appropriately as applicable, and send on the equivalent I2S/SPI command to the QF1Da512.

13.2 Data Format

The I2C bridge circuit must read and acknowledge the QF1Da512's I2C address and strip this address from the I2S/SPI command sequence sent on to the QF1Da512 (see figure 10). The first data byte must contain the command opcode (82h for Configuration WRITE, 83h for Configuration READ, 86h for Coefficient WRITE, or 87h for Coefficient READ) as it will be the first byte received by the QF1Da512. The subsequent data byte(s) will contain the actual data to be written or the data being read.

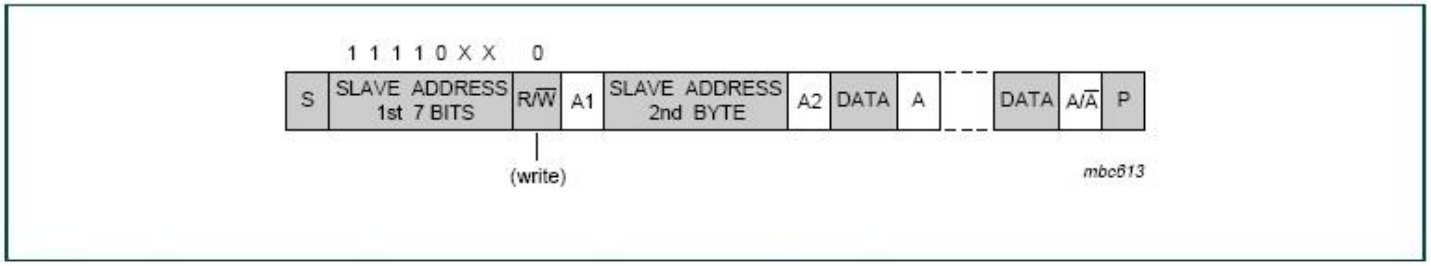


Figure 22: I2C Addressing Protocol

14 DGC User Guidance

T = Threshold

G = Gain

R = Ratio

M = Maximum expected input value

These four parameters are used to calculate two variables used by Quickfilter's algorithm.

Multiplicor = Gain / Ratio

Addor = Gain * Threshold * (1 – (1/Ratio))

These two variables are used to compute the compressed and gained value of the input. The Multiplicor is multiplied by the input while the Addor is added to the input to produce the output

If compression is disabled, Quickfilter Pro sets

T = 1

R = 1

And the user sets the gain limited to,

$$G \le \frac{1}{M}$$

If compression is enabled all values are set by user to the following limits

$$T \le M$$

$$1 < R < 16$$

$$G < \frac{R}{M + R \cdot T\left(1 - \frac{1}{R}\right)}$$

15 Packaging Information

3 x 3 x 0.9mm, VQFN 16, 0.8 mm Pull Back Lead (JEDEC)

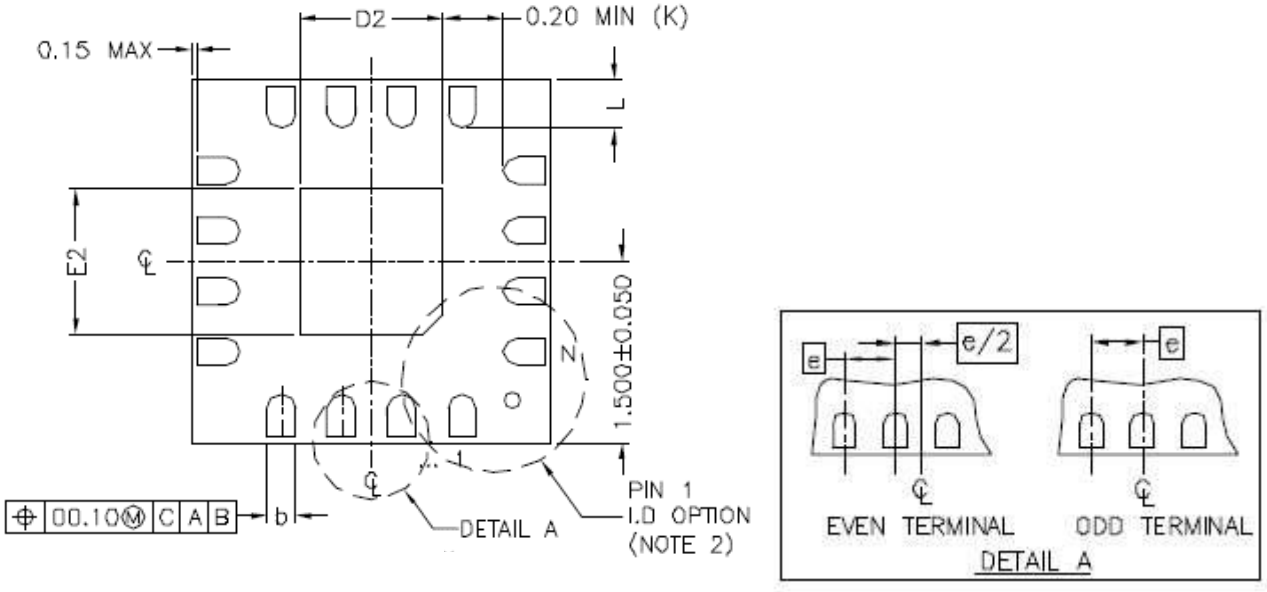


Figure 23: VQFN 16, Bottom View

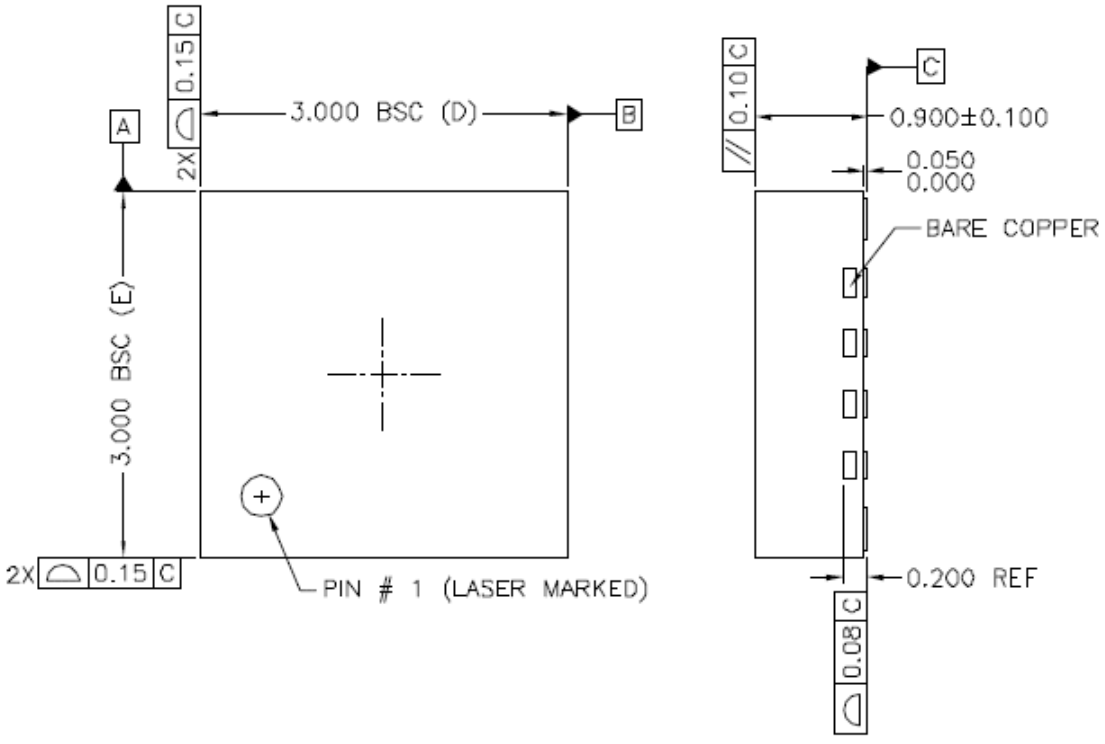


Figure 24: VQFN 16, Top/Side View

Table 6 VQFN 16, Dimensions

SYMBOL \ VARIATION		BB		
		MIN	NOM	MAX
e		0.50 BSC		
b		0.18	0.23	0.30
E2		1.60	1.70	1.80
D2		1.60	1.70	1.80
L		0.35	0.40	0.45
N		16 LD		
ND		4		
NE		4		
JEDEC VARIATION		VEED-4		
INTERNAL FEATURE		STANDARD		
LF P/N	PPF	16-016-003		
	AG SPOT	N/A		
	AG RING	F 16-016-091		
PKG CODE		VQ 016		

Notes:
1. Dimensions are in millimeters.
2. Interpret dimensions and tolerance per ASME Y14.5M-1994

16 Registers, Tables, and Figures

Control Register Listing

List of Tables

List of Figures

Quickfilter's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of Quickfilter's product could create a situation where personal injury, death, or severe property or environmental damage may occur. Quickfilter reserves the right to discontinue or make changes to its products at any time without notice.

1024 S. Greenville Ave
Suite 100
Allen, TX 75002-3344

Telephone: (214) 547-0460
Fax: (214) 547-0481

For inquiries, please contact Quickfilter at the above address,

Or visit our website:
http://www.quickfiltertech.com

Quickfilter, the Quickfilter logo and “Programmable Signal Converter”, and combinations thereof, are trademarks of Quickfilter Technologies, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.