DAC Connections to the QF4A512 Programmable Signal Converter

1) Introduction

The topic of this application note is to describe how to interconnect the QF4A512 to a Digital to Analog Converter (DAC). The Texas Instruments DAC8555 was chosen for this application because it closely matches the QF4A512 data format, supporting 2's complement 16 bit and having 4 channels. The application circuit needs only minimal interconnect logic and the part chosen was the Xilinx XC2C32A.

This application note includes:

- Overview.
- Description
- Schematic
- CPLD Source Code
- Example
- Results
- Conclusion

2) Overview

Some applications require performing an analog filtering function, where the output of the circuit is an analog signal. But some of these applications also need filter performance that can not be achieved using analog filter techniques. This application note describes using the QF4A512 with a DAC and a small programmable logic part to create a four channel analog filter. Using the internal EEPROM of the QF4A512, configuration is done automatically, eliminating the need for a microcontroller.

3) Description

The test circuit implements the circuit using a Quickfilter QF4A512, a Texas Instruments DAC8555, and the logic required to interconnect both of these, using a Xilinx Coolrunner II XC2C32A CPLD. The DAC8555 was chosen because the device supports a 24 bit data word using 16 bit data field, using 2's complement format. Table 1 describes the two data formats:

Table 1: QF4A512 versus DAC8555 Data format

	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	Bits 15 - 0
QF4A512 (Run mode)	PAR	CH1	CH0	NDF	RES	RES	RES	RES	Data
DAC8555	0	0	LD1	LD0	X	DS1	DS0	PD	Data

PAR Parity bit
CH1:0 Channel ID bits
NDF New data flag (1: Indicates new data)
RES Reserved for future use
LD1:0 Load options (00: No DAC updated; 01: DAC selected by DS1:0 updated; 10: All DAC's updated)
DS1:0 DAC Channel ID bits
PD Power down (1: Powers down DAC output)

The data formats are very similar, the only differences being in the preceding 8 header bits of the 24 bit word. The function of the CPLD is to convert the QF4A512 output data into the format needed by the DAC8555 and to act as the SPI master. The QF4A512 updates the SDO pin on the rising edge of SCLK while the DAC8555 reads its SDI pin on the falling edge of SCLK. This allows the CPLD to use a common SCLK and /CS for both the QF4A512 and the DAC8555.

The QF4A512 stores the configuration and filter coefficients in internal EEPROM and the configuration is set for automatic load and start, so that on power up, the QF4A512 starts running. The CPLD monitors the DRDY signal from the QF4A512, reads the four channels of data, and simultaneously writes the data out to the corresponding DAC channel. The CPLD also has a host port interface implemented, so that an external processor can configure the QF4A512. The host interface has priority over the CPLD state machine and will interrupt it immediately.

4) Schematic

The schematic (supplied as a separate file) shows an implementation of the test circuit. The actual test circuit utilized the QF4A512-DK, so the additional circuit was implemented on a separate board with a ribbon connector to the QF4A512-DK board. The clock was to the CPLD was provided with an external signal generator. The schematic for a stand alone design is provided in PDF and ORCAD 9.2 format.

5) CPLD Source Code

The CPLD code was designed in VHDL and was implemented using the Xilinx ISE 7.1.04 tool. The XC2C32A was chosen to highlight the small size of the device needed to implement the design. The design utilized 91% of the resources of the XC2C32A. The source code and project files are provided for this applications note on our website (http://www.quickfiltertech.com/html/app_notes.php).

6) Example

The circuit was implemented by using the QF4A512-DK and a separate board containing the CPLD and DAC, interconnected with a ribbon cable.

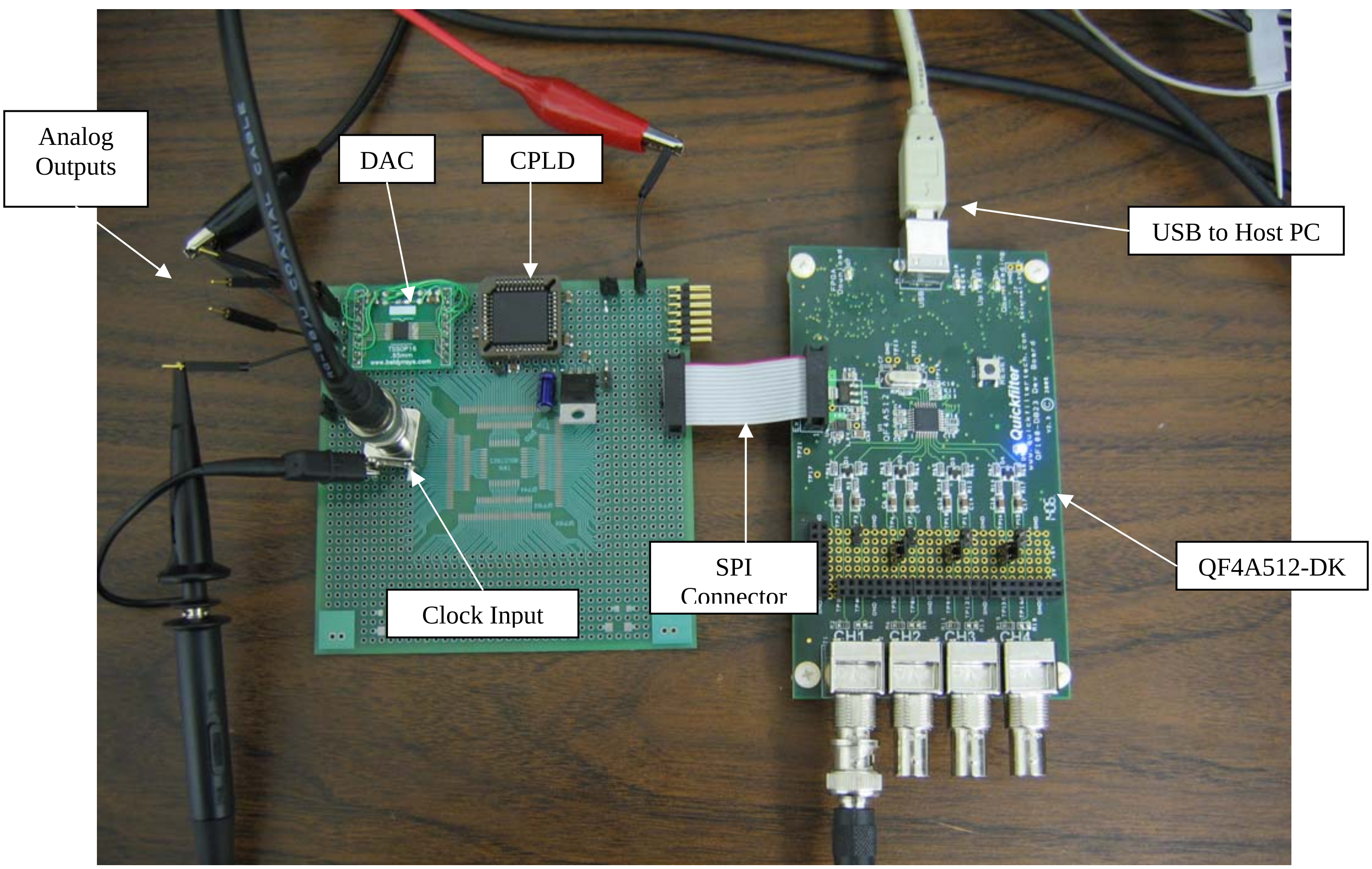

Figure 1: Test Configuration

The development board was modified so that the signal going into channel 1 is also connected to channels 2, 3 and 4.

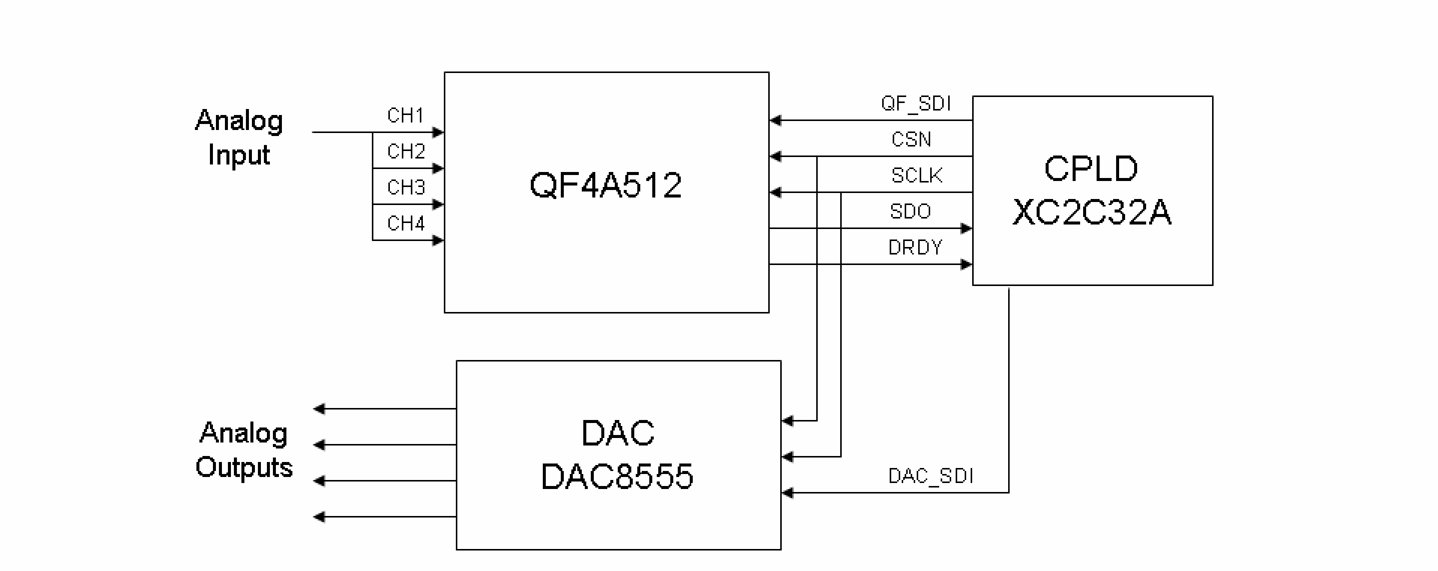

Figure 2: Block Diagram

Four filters were designed and loaded into the QF4A512. On channel 1, there is a low pass filter with a corner frequency of 500 Hz, sampled at 10 KHz. The sampling frequency is significantly higher than required, but this was done so that the output data rate would be high enough that the sampling effect would not distort the output DAC waveform significantly. In an actual implementation, a low pass filter might be added to filter out the sample clock noise.

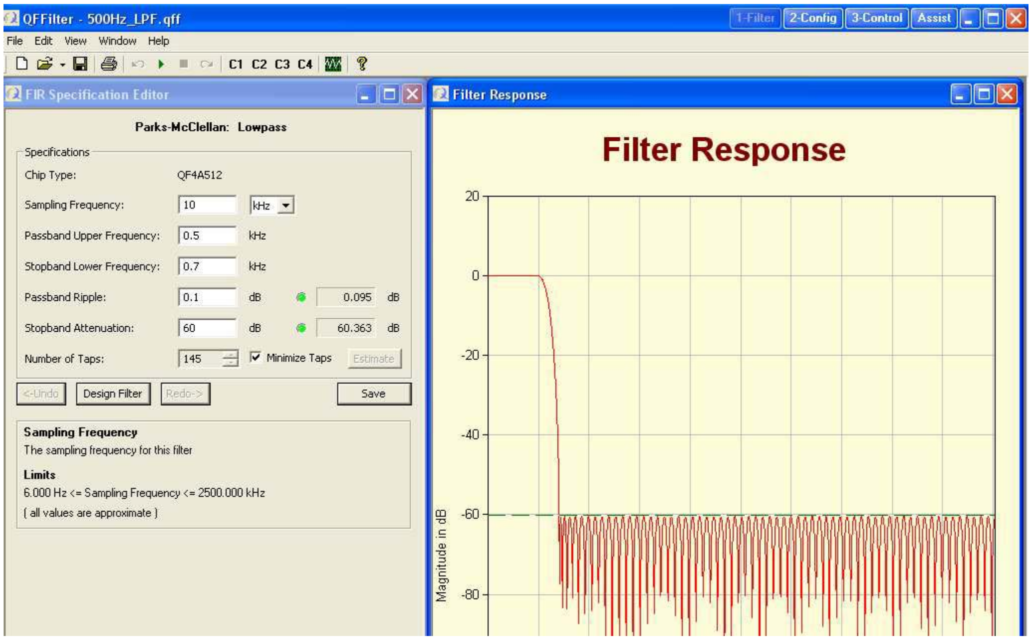

Figure 3: Channel 1, 500 Hz Low Pass Filter

Channels 2, 3, and 4 have band pass filters with center frequencies of 200, 300, and 400 Hz respectively.

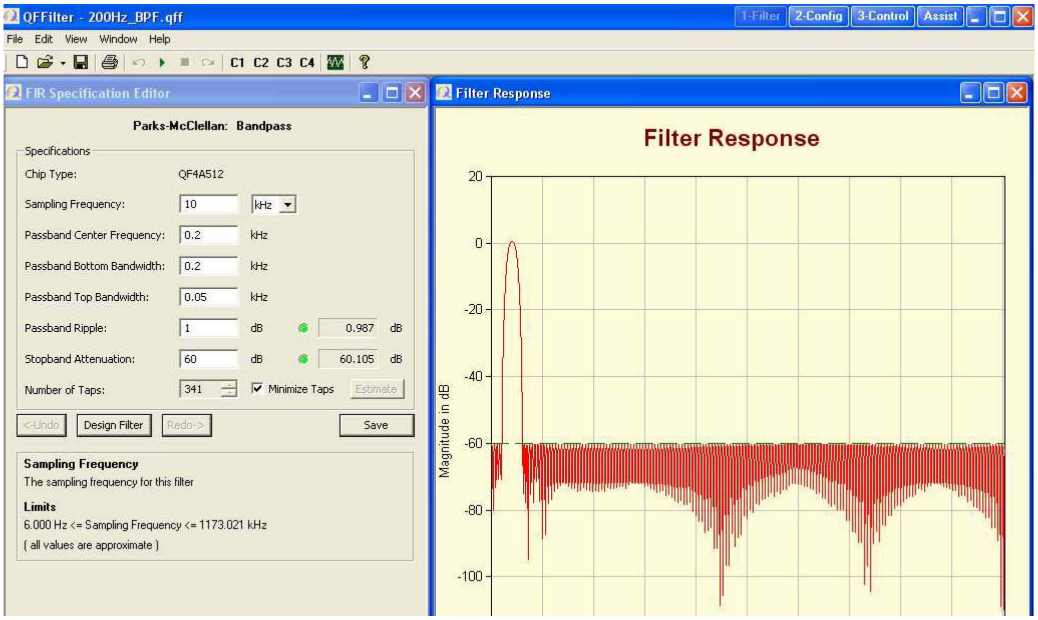

Figure 4: Channel 2, 200 Hz Band Pass Filter

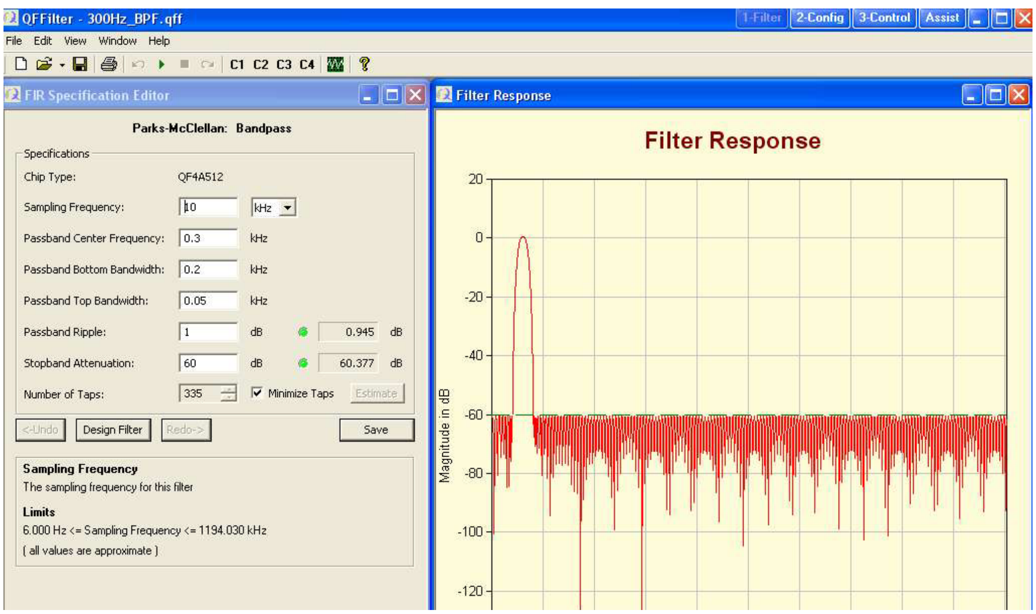

Figure 5: Channel 3, 300 Hz Band Pass Filter

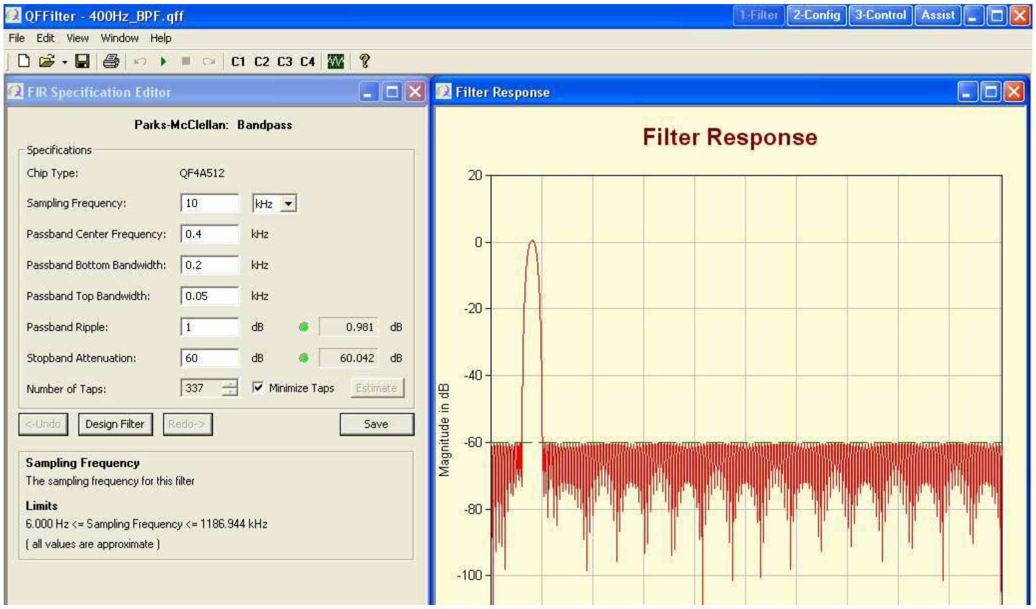

Figure 6: Channel 4, 400 Hz Band Pass Filter

The filter and device configuration files are provided for this application note. As mentioned previously, the configuration was stored in the EEPROM of the QF4A512 and the part was configured to automatically load and run on power up or reset.

The signal source was a PC sound card configured to generate three signals at 200, 300, and 400 Hz of equal amplitudes.

7) Results

The following oscilloscope outputs show the output of each of the channel filters compared to the input signal. Channel 1 of the oscilloscope is the top input signal (yellow) while Channel 2 of the oscilloscope is the output of the DAC8555 for each of the 4 channels.

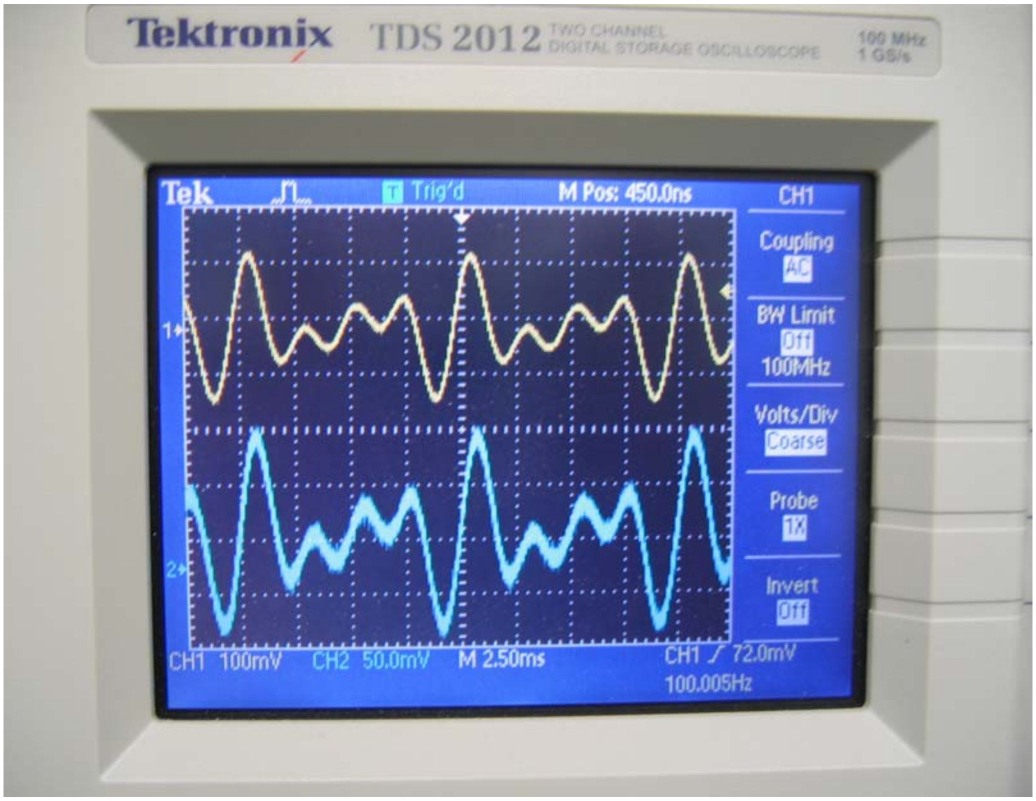

Figure 7: Signal Input versus Channel 1 Output (500 Hz Low Pass Filter)

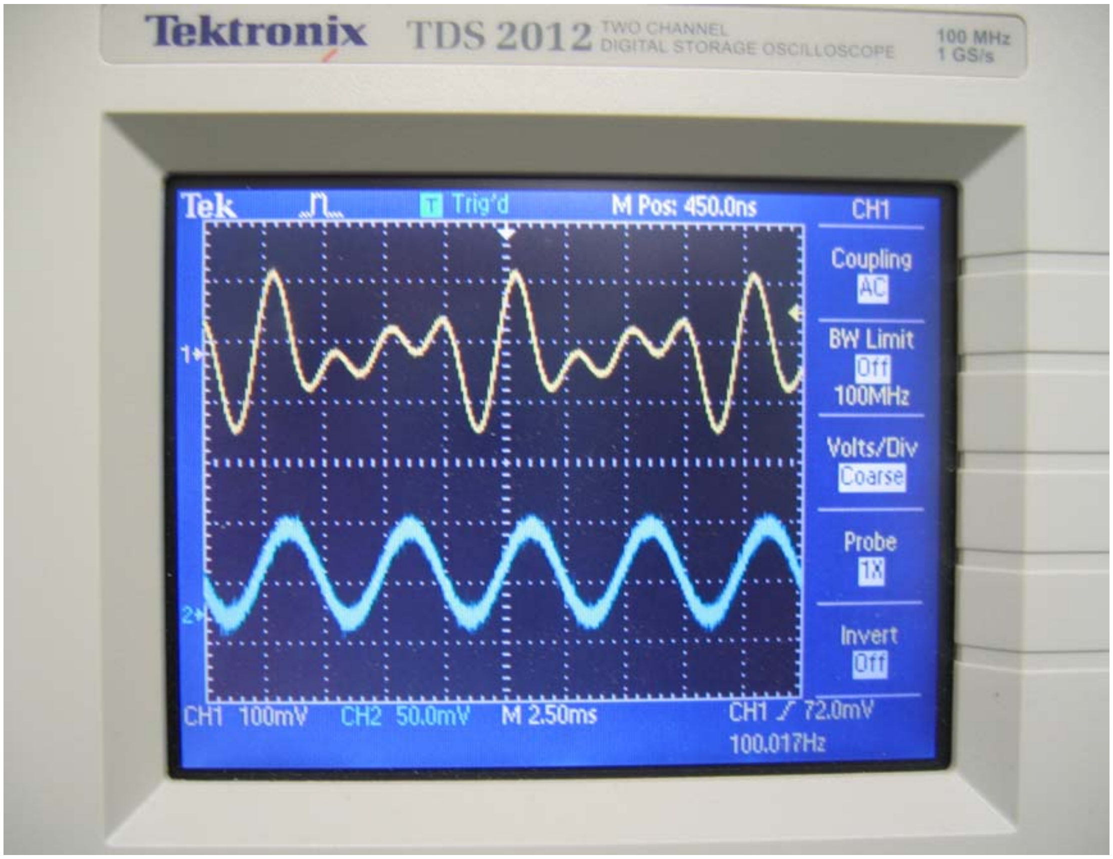

Figure 8: Signal Input versus Channel 2 Output (200 Hz Band Pass Filter)

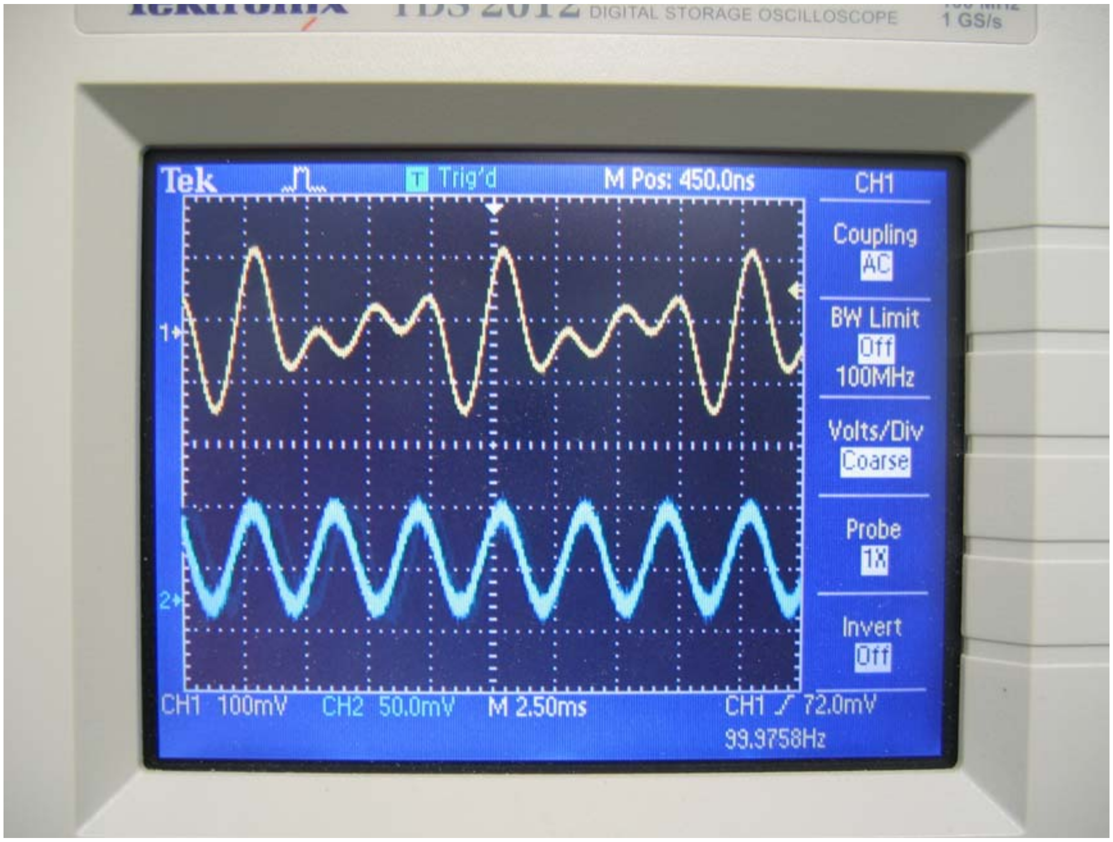

Figure 9: Signal Input versus Channel 3 Output (300 Hz Band Pass Filter)

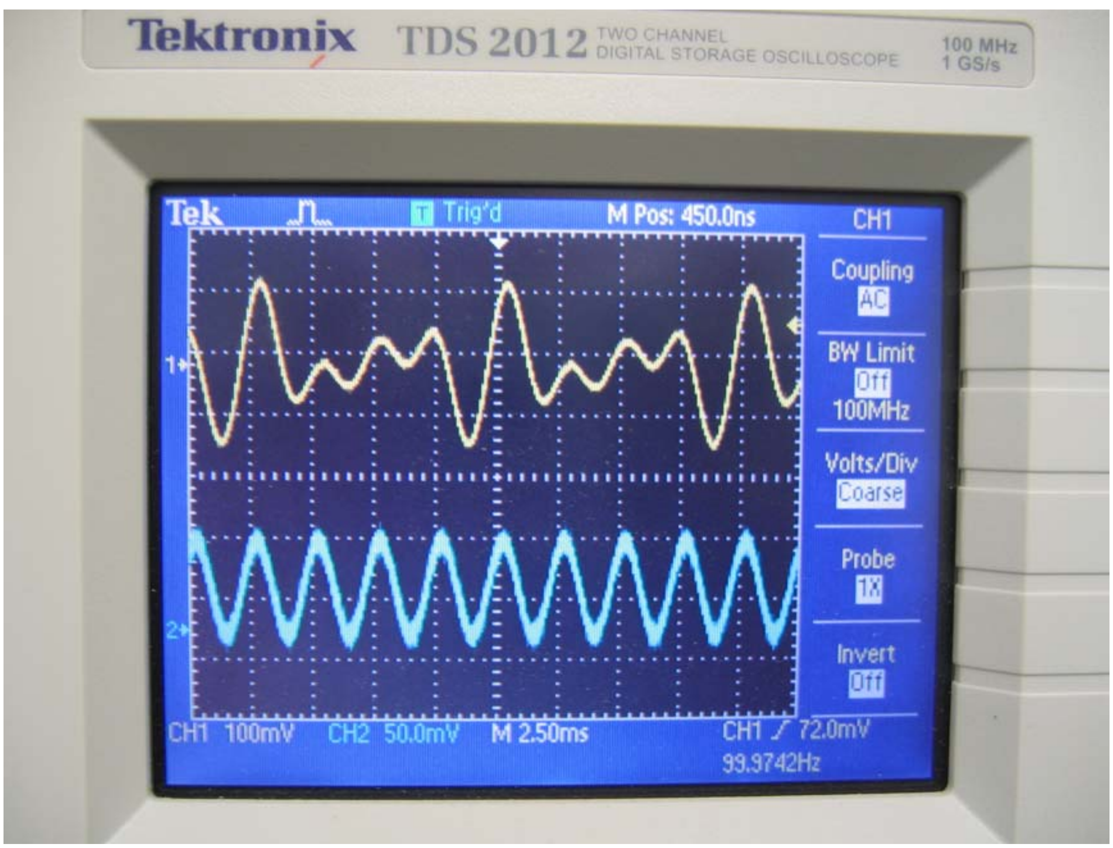

Figure 10: Signal Input versus Channel 4 Output (400 Hz Band Pass Filter)

As seen in the oscilloscope image, the component frequency signals are extracted in the band pass filters.

8) Conclusion

The QF4A512 can be used directly with a DAC converter to give an analog to analog filter, with the performance that digital filtering can provide. The circuit required for this application is easy to implement with standard DAC devices and a minimum amount of interconnect logic.

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