In this write up, we will be using a resistor, capacitor and a Pulse Width Modulated (PWM) output to create a voltage. As was described previously, electrical circuits and fluid flow have similar character, so in an effort to understand how we can interpret the output voltage, consider the following system. In this we have a pump that will generate a pressure P_1, then a restricted pipe which supplies a tank. This system is set up in Figure 1.

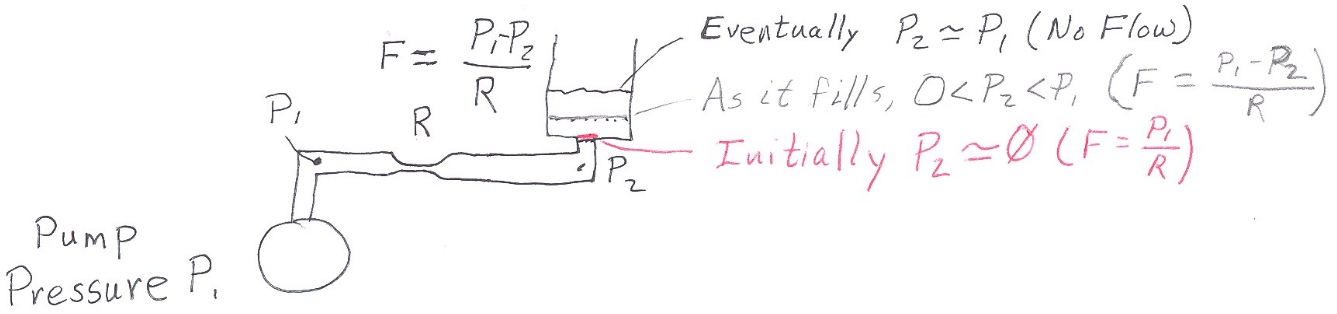

$F \simeq \frac{P_1 - P_2}{R}$
P_1
R
Eventually $P_2 \simeq P_1$ (No Flow)
As it fills, $0 < P_2 < P_1$ $\left(F = \frac{P_1 - P_2}{R}\right)$
Initially $P_2 \simeq \emptyset$ $\left(F = \frac{P_1}{R}\right)$
P_2
Pump Pressure P_1

Figure 1. Fluid Flow Analogy of an Resistor-Capacitor Circuit.

In this case, we can see that the flow will start out high then slow as the tank fills. In a similar fashion, the RC circuit, in Figure 2, will start charging up the capacitor quickly at first (high current). Then as the capacitor charges up and its voltage rises, the current drops off and the charging slows.

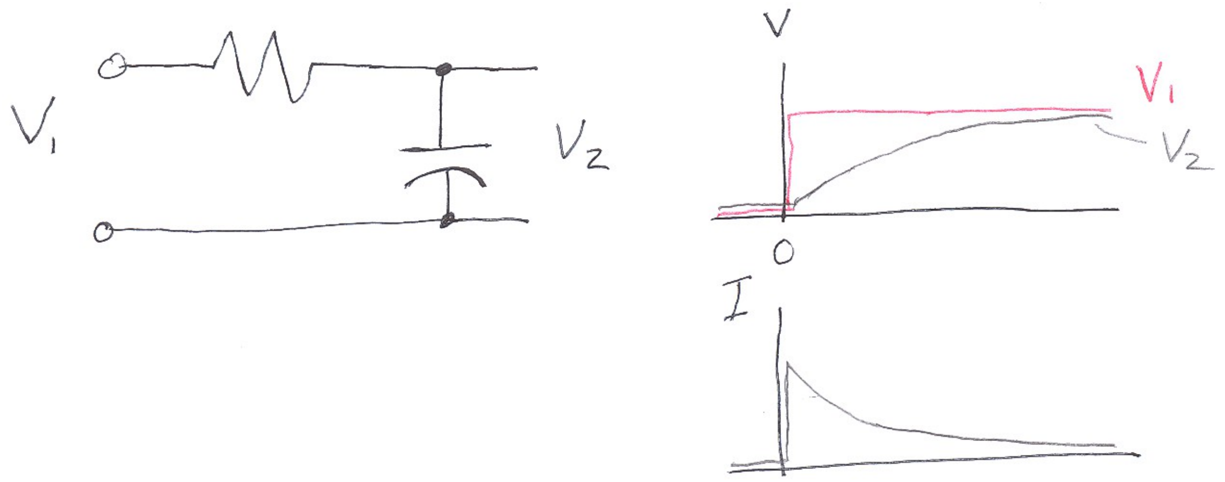


Figure 2. Resistor-Capacitor (RC) Voltage and Current Character.

So if we have an input voltage, V1, that is a PWM signal the output will be like that that appears in Figure 3.

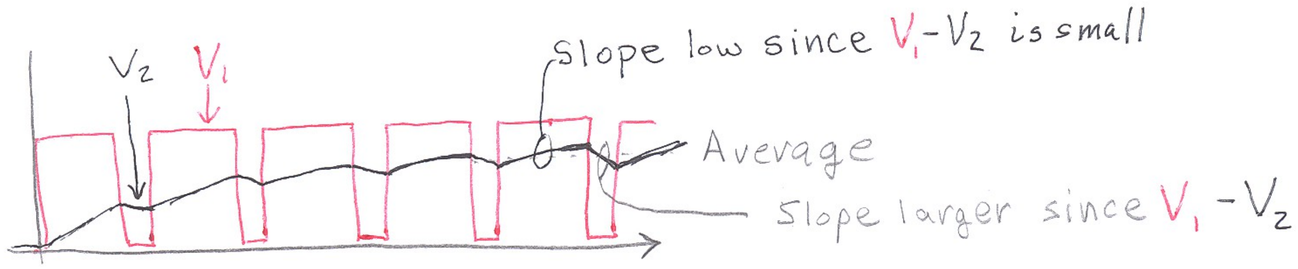


Figure 3. RC Circuit Response to a PWM Signal.

A PWM signal is characterized by its period, T, and its duty cycle. These parameters are shown in Figure 4.

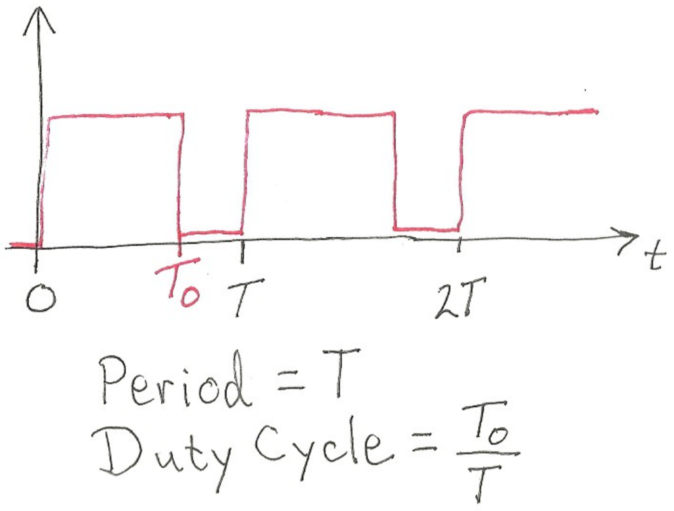
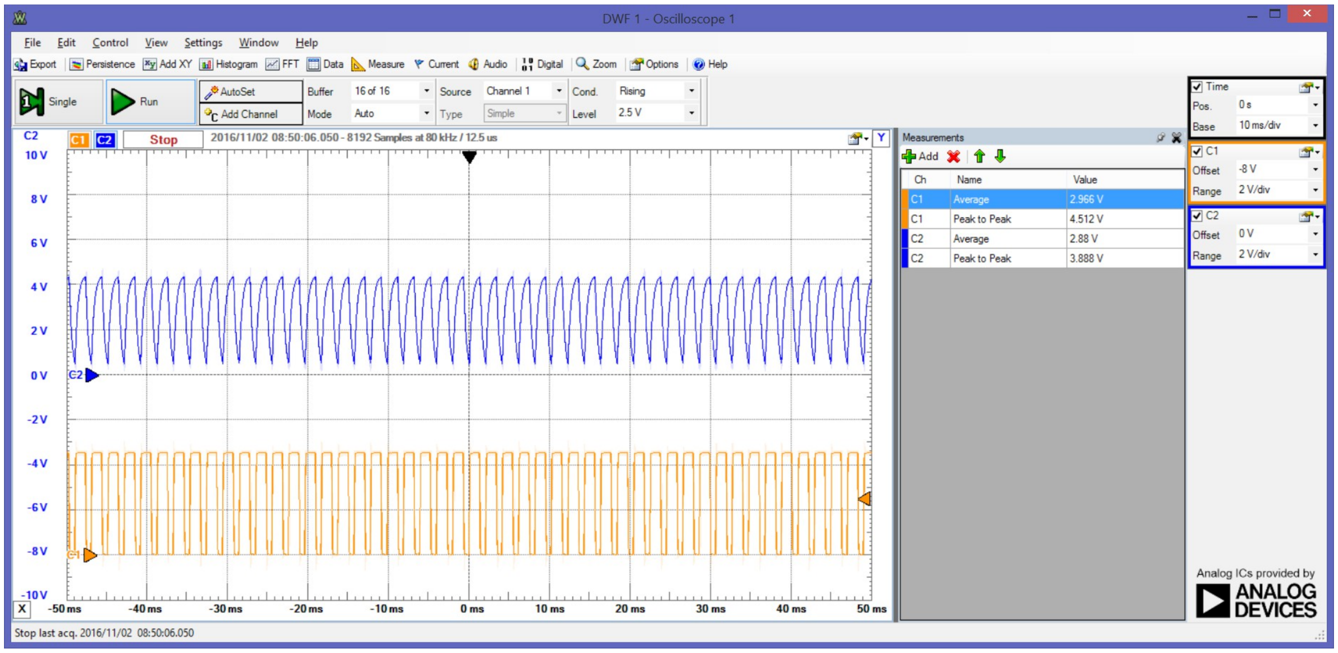


Figure 4. PWM Signal Characteristics.

The problem that we will need to study is the values for the resistor, the capacitor and period of the PWM signal. We begin by noting that the analogWrite function in the Arduino IDE has a period that is really low, 490 Hz or 2.04 milliseconds. Also, it has an eight-bit duty cycle resolution (0 to 255). This will prove to be a problem as we look at the output.

1) The first case is an RC, with a 10KΩ 0 resistor and a 0.1μF capacitor. Also we are using analogWrite() to set the PWM. The resulting waveform has a huge ripple, which might be fixed by changing the R and C of the filter. However if we do that, the rate at which we can control the voltage will be reduced. This will be demonstrated later.


DWF 1 - Oscilloscope 1
File Edit Control View Settings Window Help
Export Persistence Add XY Histogram FFT Data Measure Current Audio Digital Zoom Options Help
Single Run AutoSet Add Channel
C2 C1 C2 Stop
10 V 8 V 6 V 4 V 2 V 0 V -2 V -4 V -6 V -8 V -10 V
-50 ms -40 ms -30 ms -20 ms -10 ms 0 ms 10 ms 20 ms 30 ms 40 ms 50 ms
Measurements Add
Analog ICs provided by ANALOG DEVICES

Buffer 16 of 16
Mode Auto
Source Channel 1
Type Simple
Cond. Rising
Level 2.5 V

2016/11/02 08:50:06.050 - 8192 Samples at 80 kHz / 12.5 us

Ch	Name	Value
C1	Average	2.966 V
C1	Peak to Peak	4.512 V
C2	Average	2.88 V
C2	Peak to Peak	3.888 V

Time
Pos. 0 s
Base 10 ms/div

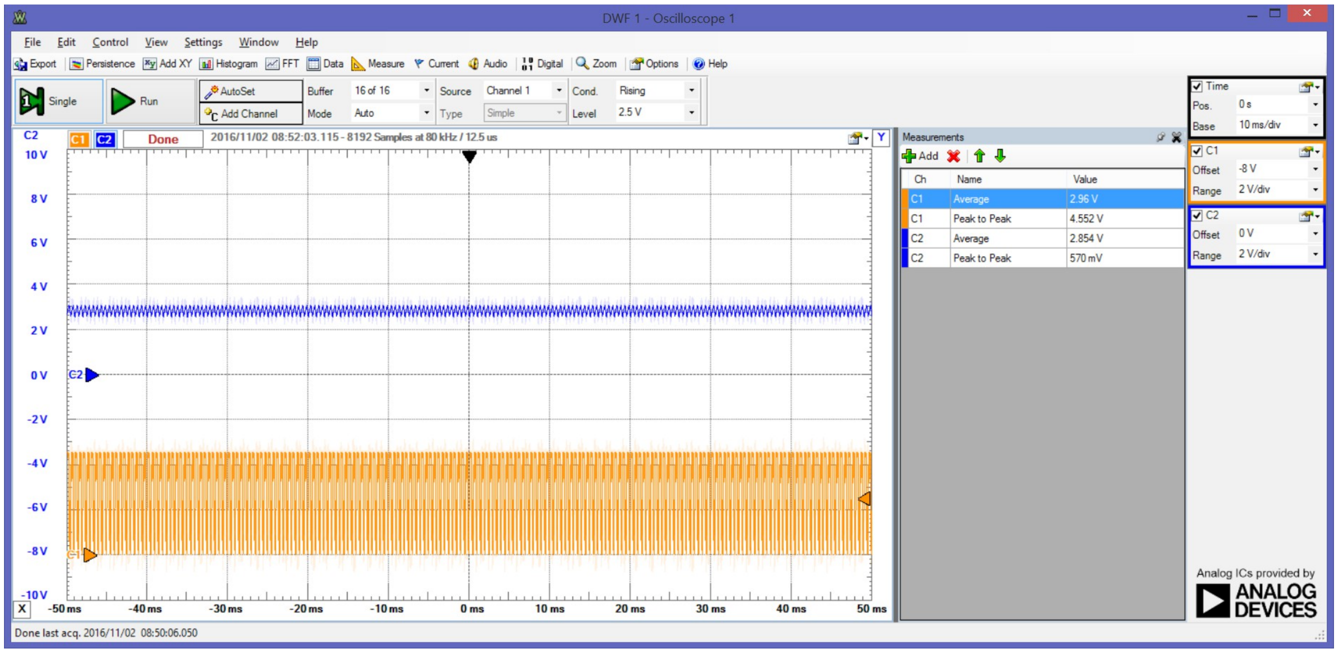
C1
Offset -8 V
Range 2 V/div

C2
Offset 0 V
Range 2 V/div

Stop last acq. 2016/11/02 08:50:06.050

2) If we change over to using to Timer1 to generate the PWM signal, we can set the period of our output to a smaller value. In this case a 500 µs period was chosen. Timer1 will also allow for a 10 bit resolution (0 to 1023) on duty cycle.

$$DutyCycle=\frac{Setting}{1023}$$


DWF 1 - Oscilloscope 1

Buffer: 16 of 16 · Mode: Auto · Source: Channel 1 · Type: Simple · Cond.: Rising · Level: 2.5 V

Done · 2016/11/02 08:52:03.115 - 8192 Samples at 80 kHz / 12.5 us

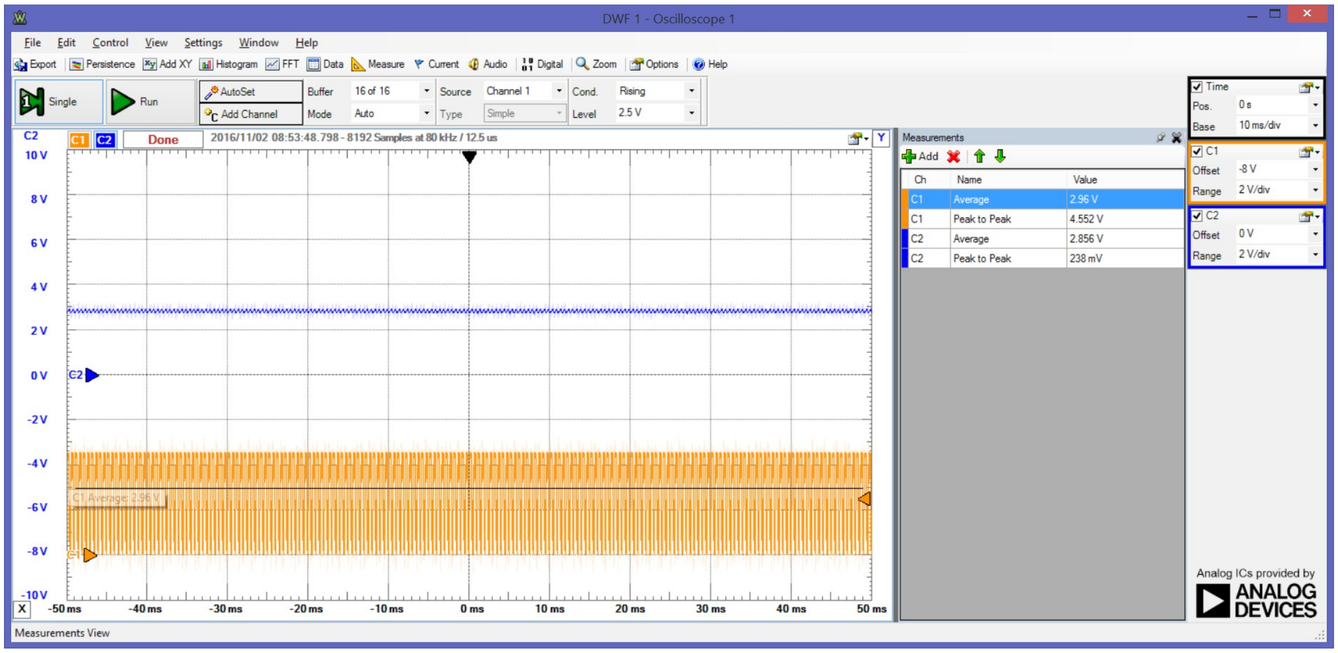
Time: Pos. 0 s · Base 10 ms/div
C1: Offset -8 V · Range 2 V/div
C2: Offset 0 V · Range 2 V/div

Measurements

Ch	Name	Value
C1	Average	2.96 V
C1	Peak to Peak	4.552 V
C2	Average	2.854 V
C2	Peak to Peak	570 mV

Done last acq. 2016/11/02 08:50:06.050

3) The next figure shows what happens if we increase the capacitor to 0.2 µF. We can see that the ripple has decreased significantly. So what is the problem? The next section will use a time varying input, to demonstrate the effects of R and C on the response time.


DWF 1 - Oscilloscope 1

Buffer: 16 of 16 · Mode: Auto · Source: Channel 1 · Type: Simple · Cond.: Rising · Level: 2.5 V

Done · 2016/11/02 08:53:48.798 - 8192 Samples at 80 kHz / 12.5 us

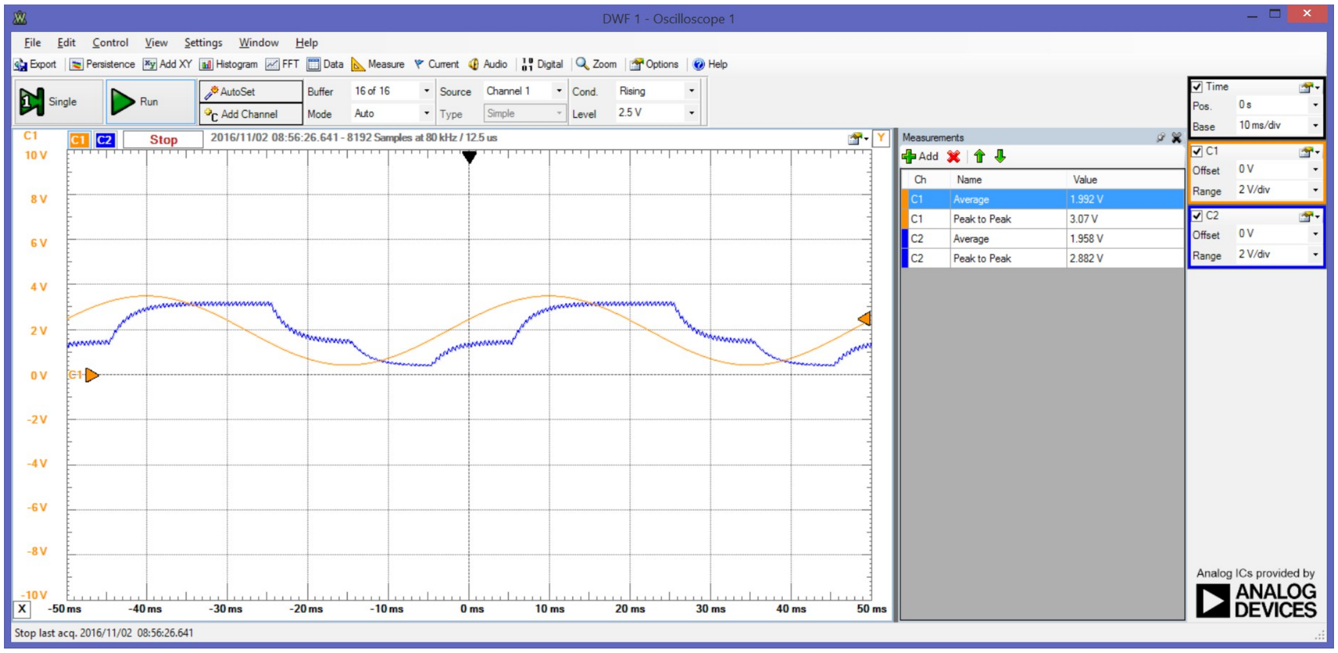
Time: Pos. 0 s · Base 10 ms/div
C1: Offset -8 V · Range 2 V/div
C2: Offset 0 V · Range 2 V/div

Measurements

Ch	Name	Value
C1	Average	2.96 V
C1	Peak to Peak	4.552 V
C2	Average	2.856 V
C2	Peak to Peak	238 mV

Measurements View

4) Consider Timer1 with 500 µs period, 10KΩ resistor, 0.2 µF capacitor, and a quickly changing input. Notice the slow change time. The response to a new input has a similar shape to the response of an RC circuit to a step change input, shown in Figure 2.


DWF 1 - Oscilloscope 1
File Edit Control View Settings Window Help
Export Persistence Add XY Histogram FFT Data Measure Current Audio Digital Zoom Options Help
Single Run AutoSet Add Channel
Stop
10 V 8 V 6 V 4 V 2 V 0 V -2 V -4 V -6 V -8 V -10 V
-50 ms -40 ms -30 ms -20 ms -10 ms 0 ms 10 ms 20 ms 30 ms 40 ms 50 ms
Analog ICs provided by ANALOG DEVICES

Buffer 16 of 16 | Mode Auto | Source Channel 1 | Type Simple | Cond. Rising | Level 2.5 V

2016/11/02 08:56:26.641 - 8192 Samples at 80 kHz / 12.5 us

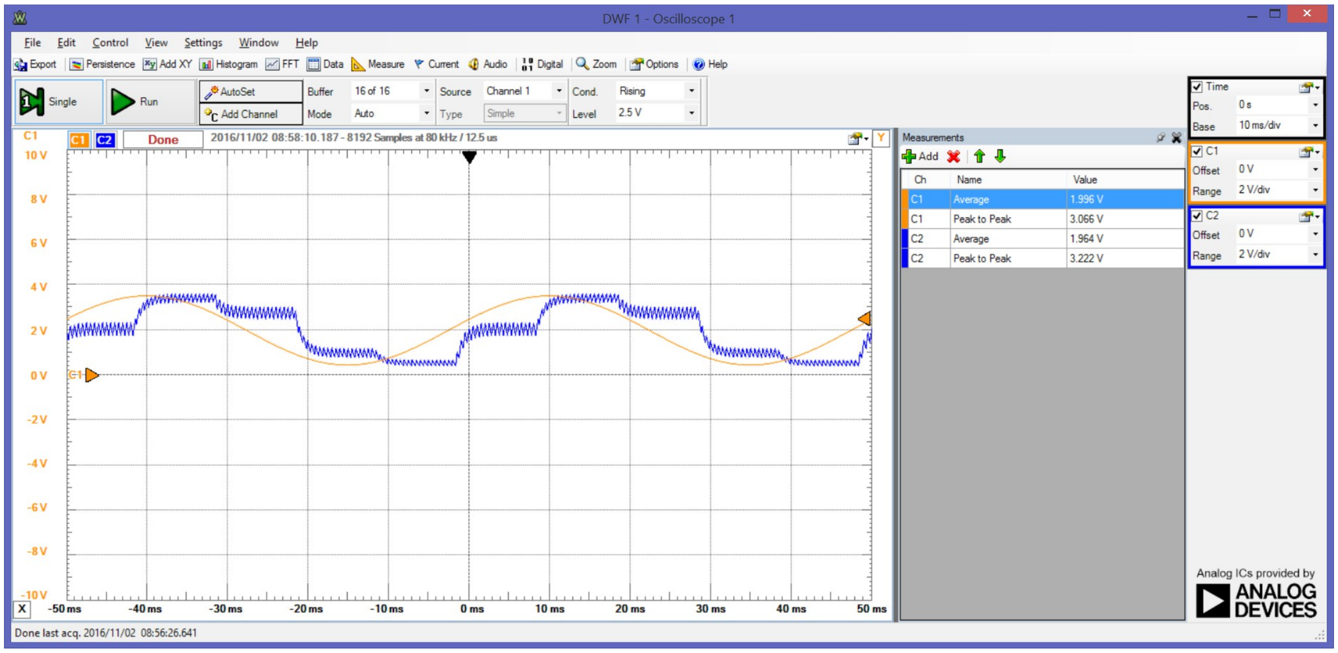
Time: Pos. 0 s, Base 10 ms/div; C1: Offset 0 V, Range 2 V/div; C2: Offset 0 V, Range 2 V/div

Measurements

Ch	Name	Value
C1	Average	1.992 V
C1	Peak to Peak	3.07 V
C2	Average	1.958 V
C2	Peak to Peak	2.882 V

Stop last acq. 2016/11/02 08:56:26.641

5) Changing back to 0.1 µF we have


DWF 1 - Oscilloscope 1
File Edit Control View Settings Window Help
Export Persistence Add XY Histogram FFT Data Measure Current Audio Digital Zoom Options Help
Single Run AutoSet Add Channel
Done
10 V 8 V 6 V 4 V 2 V 0 V -2 V -4 V -6 V -8 V -10 V
-50 ms -40 ms -30 ms -20 ms -10 ms 0 ms 10 ms 20 ms 30 ms 40 ms 50 ms
Analog ICs provided by ANALOG DEVICES

Buffer 16 of 16 | Mode Auto | Source Channel 1 | Type Simple | Cond. Rising | Level 2.5 V

2016/11/02 08:58:10.187 - 8192 Samples at 80 kHz / 12.5 us

Time: Pos. 0 s, Base 10 ms/div; C1: Offset 0 V, Range 2 V/div; C2: Offset 0 V, Range 2 V/div

Measurements

Ch	Name	Value
C1	Average	1.996 V
C1	Peak to Peak	3.066 V
C2	Average	1.964 V
C2	Peak to Peak	3.222 V

Done last acq. 2016/11/02 08:56:26.641

Thus we have a trade off of the response time versus the ripple on our output.

Appendix A: Code for PWM Test

```
// include libraries.
#include <TimerOne.h>
#include <MsTimer2.h>

// Switch between using analogWrite and Timer1.
// If commented out, code uses Timer1.
#define ANALOGWRITE

// Pins used
#define AnalogPin 0
#define AnalogOutputPin 10

// Interrupt Service Routine
// Samples analog input 0
// then sends it out on pin 10.
void Adc_Dac_ISR()
{
	int Input;
	PORTB |= 0x20; // Led On (indicates ISR active)

	// Read voltage from pin
	Input = analogRead(0);
#ifdef ANALOGWRITE
	analogWrite(AnalogOutputPin, Input / 4);
#else
	Timer1.setPwmDuty(AnalogOutputPin, Input);
#endif
	PORTB &= ~0x20; // Led Off (indicates ISR done)

} // End of Adc_Dac_ISR

// put your setup code here, to run once:
void setup()
{
	// Set ISR to fire every 10 milliseconds
	MsTimer2::set(10, Adc_Dac_ISR);
	MsTimer2::start();

	// Set output for analog signal.
	pinMode(AnalogOutputPin, OUTPUT);

#ifndef ANALOGWRITE
	// if using Timer1 for PWM.
	Timer1.initialize(500);
	Timer1.pwm(AnalogOutputPin, 512);
#endif
} // End of setup


void loop()
{
	// put your main code here, to run repeatedly:
	// Nothing, since all is handled in ISR.
} // End of loop.
```