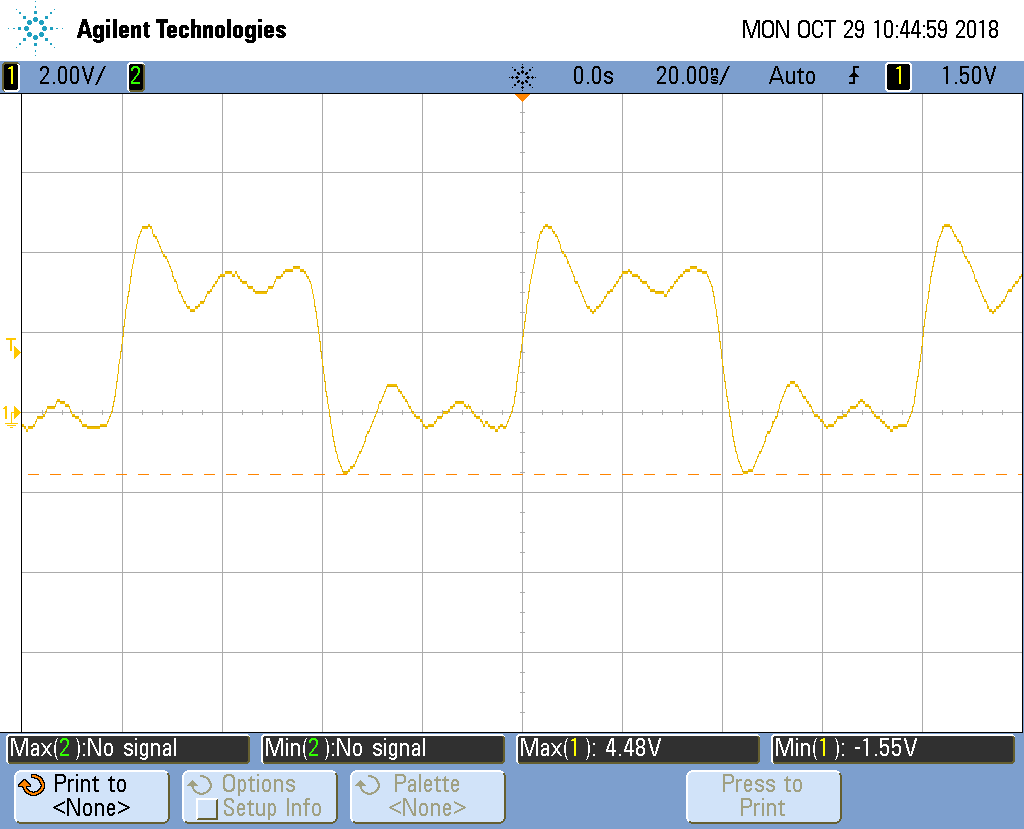
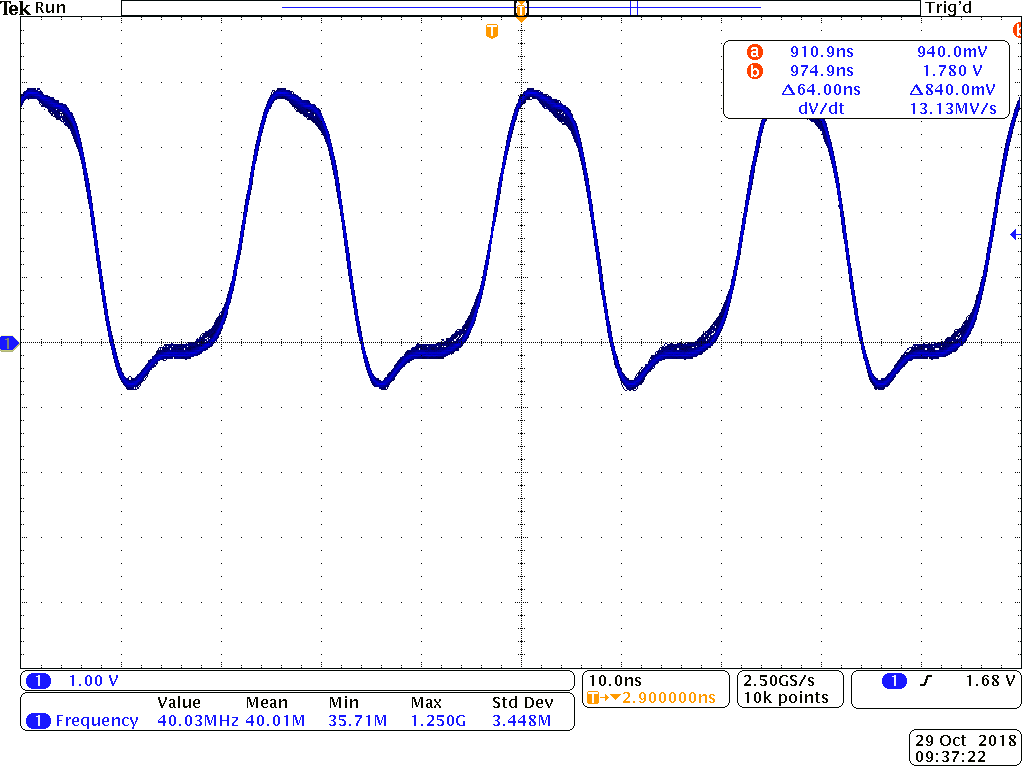
1. Blinking an LED



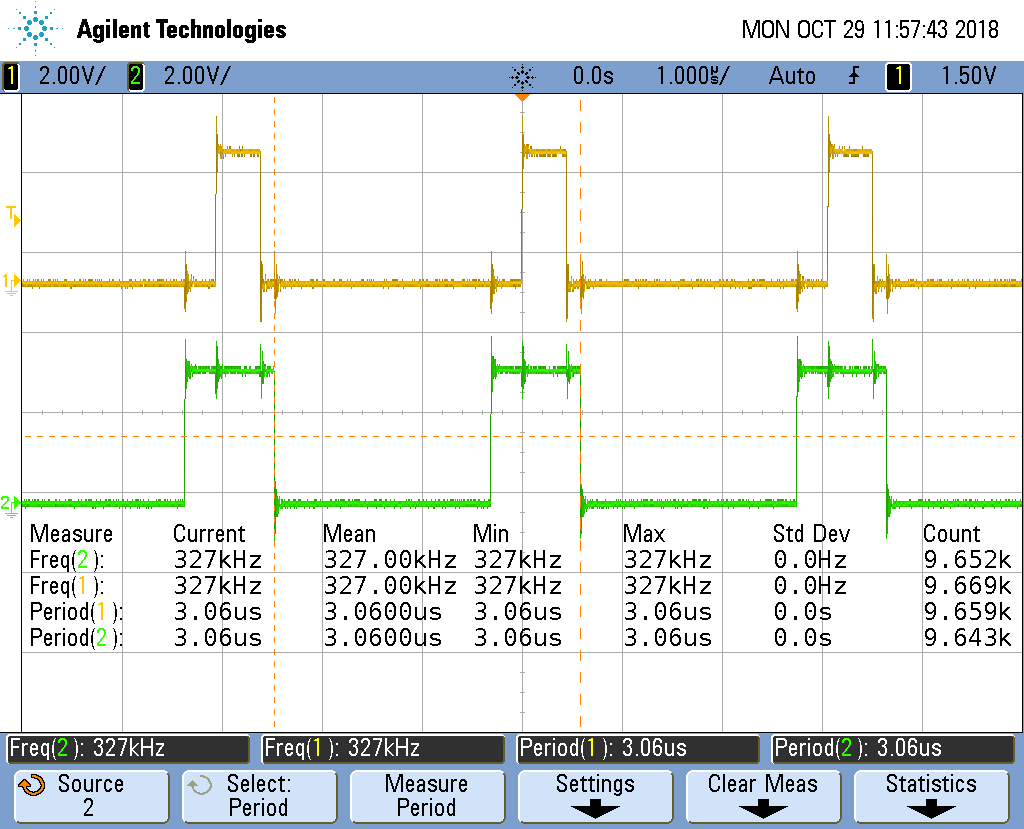
This is a scope capture of the toggle code with \_\_delay\_cylces() set to 0. The fastest you can toggle a pin using gpio is 25MHz. there is a decent amount of jitter. Yes it is stable.

1. PWM Generator



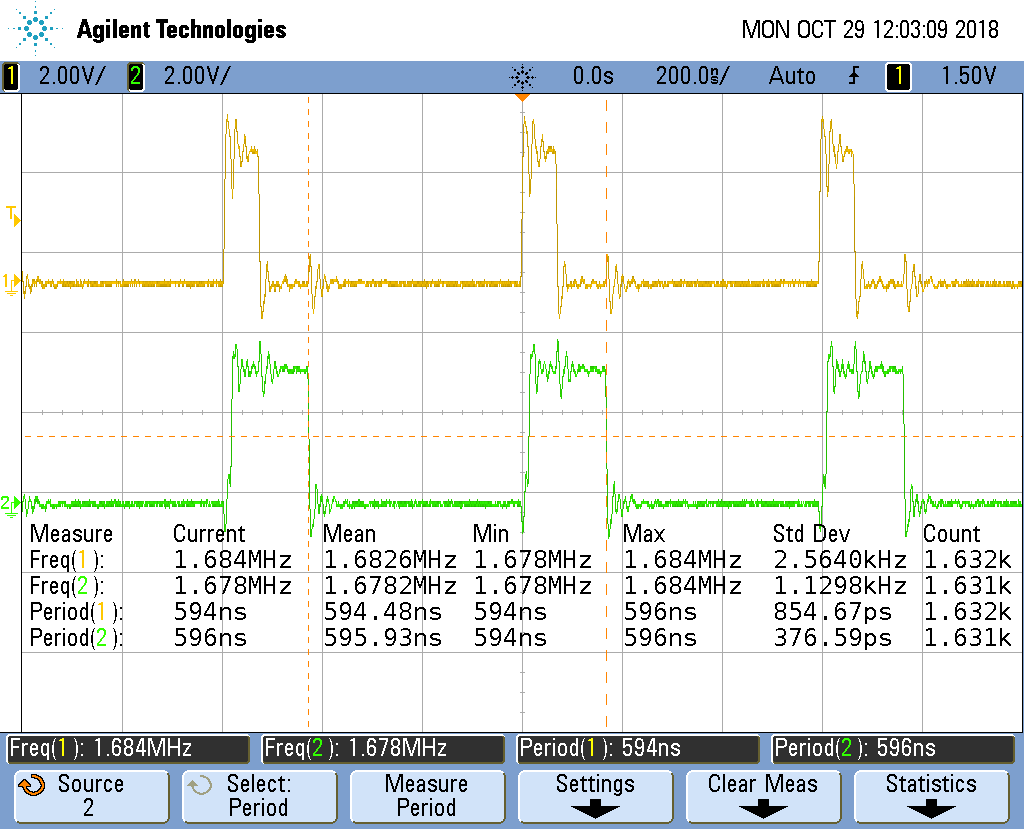
This is a scope capture of the pwm4 code with \_\_delay\_cylces() set to 0. My beaglebone could not get to 50MHz with this code so I had to use whatever the fastest I could get. The waveform doesn’t really resemble a rectangular waveform anymore, and the Std Dev is 3.448MHz which means that this waveform is not very stable. Also there is some jitter.

1. Controlling the PWM frequency



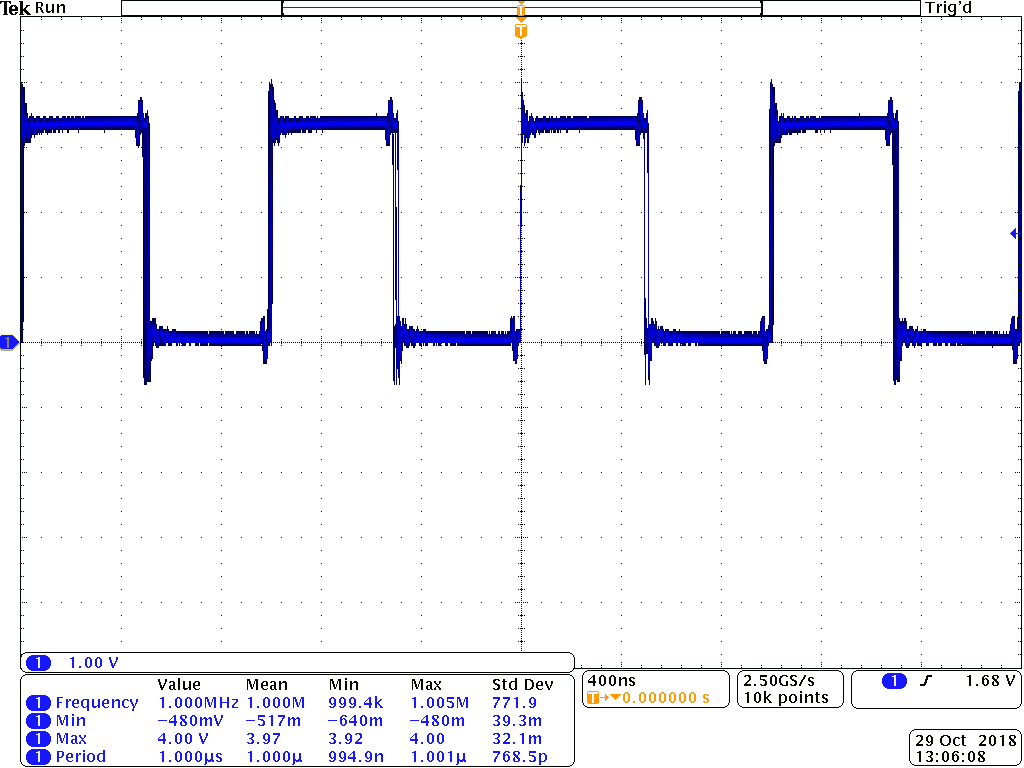
The highest frequency I could get was 327kHz with very little jittering. Also the pwm-test.c does work in changing the on and off times.

1. Loop Unrolling for Better Performance

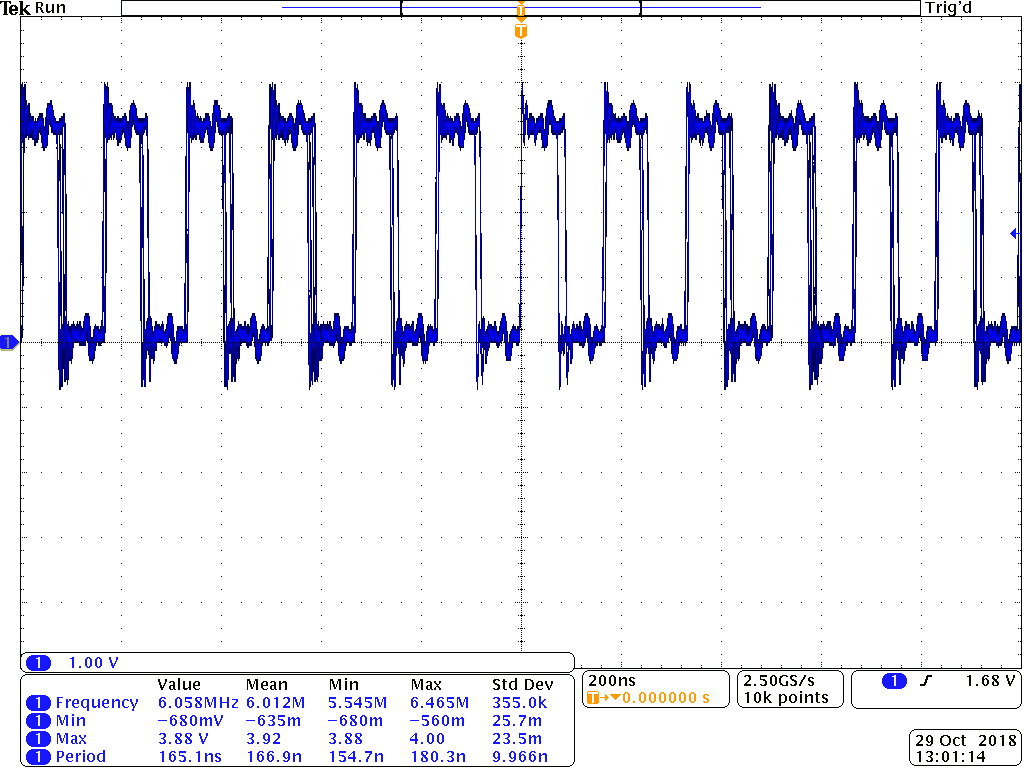


The fastest speed of the waveform increased by 5.15 times the previous fastest.

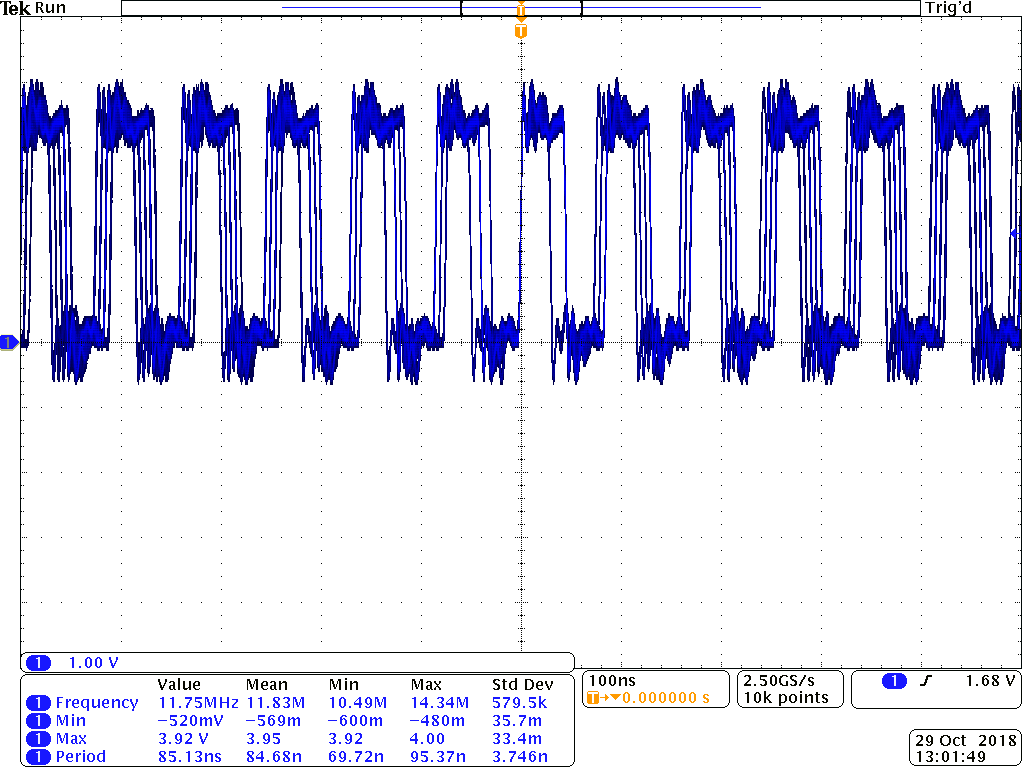
1. Reading an Input at Regular intervals



(a)



(b)



(c)

The three plots (a), (b), and (c) are the input1 code with a function generator at different frequencies. (a) at 1MHz is the most stable with very little jittering. (b) at 6.058MHz is the fastest that the beaglebone can transfer using the input1 code while still being relatively stable with some jitter. Finally, (c) at 11.75Mz the waveform is definitely not stable and a large amount of jittering can be seen.