

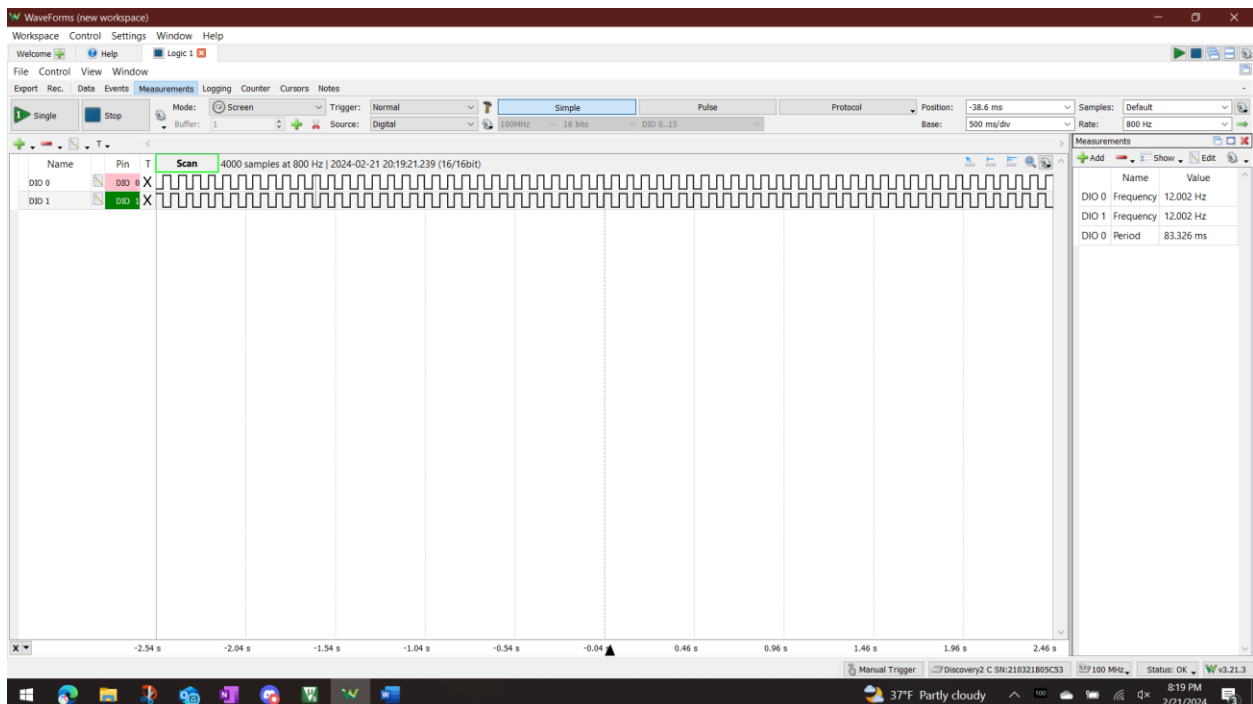
1. Using a timer clock source of 8 MHz, calculate PSC and ARR values to get a 60 Hz interrupt.
 - This is tricky because precisely 60 Hz is impossible with our system; instead, think about the process and minimize the error. Many combinations of PSC and ARR values work—not just one!

The PSC and ARR value combinations that I found to work for a ~60Hz interrupt is 7999 for the PSC and 17 for the ARR (16.666... in reality)

2. Look through the Table 13 "STM32F072x8/xB pin definitions" in the chip datasheet and list all pins that can have the timer 3 capture/compare channel 1 alternate function. • If the pin is included on the LQFP64 package that we are using, list the alternate function number that you would use to select it.
 - PE3
 - PA6 (GPIOA->AFR[1])
 - PC6 (GPIOC->AFR[0])
 - PB4 (GPIOB->AFR[1])

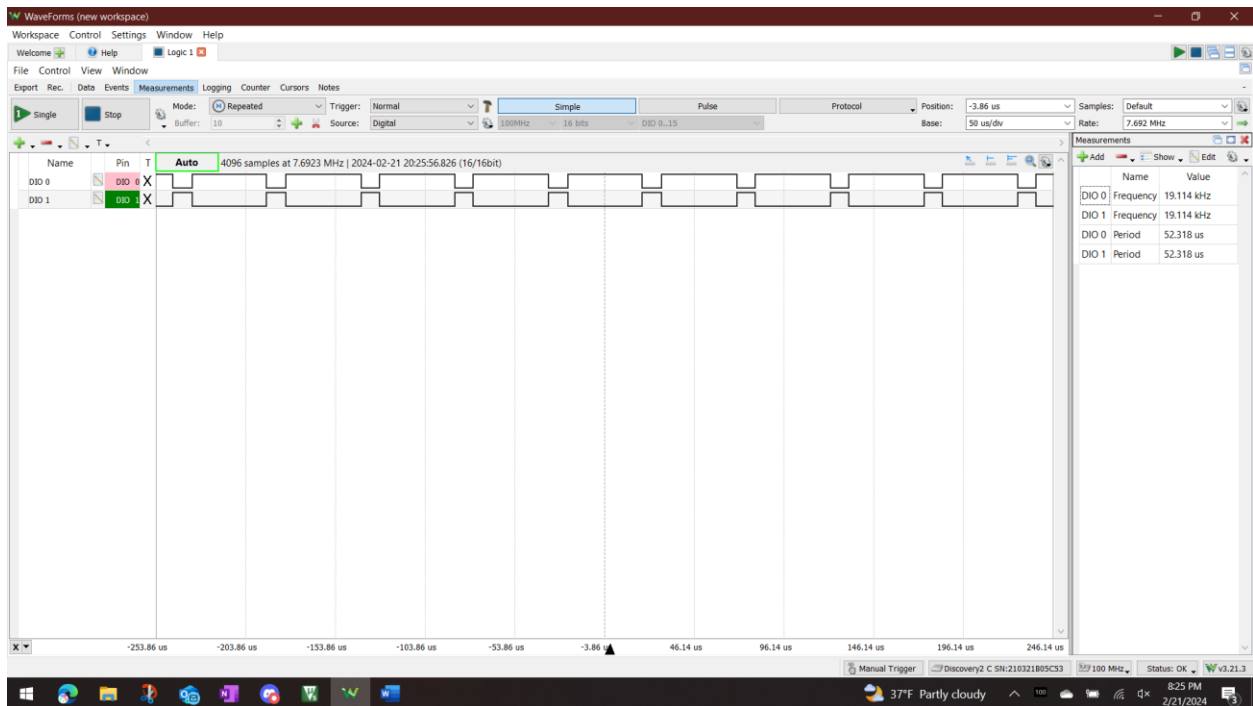
3. List your measured value of the timer UEV interrupt period from first experiment.

~83.326 ms (12 Hz; this microcontroller strangely appears to use the 24MHz clock by default rather than the 8MHz clock.)

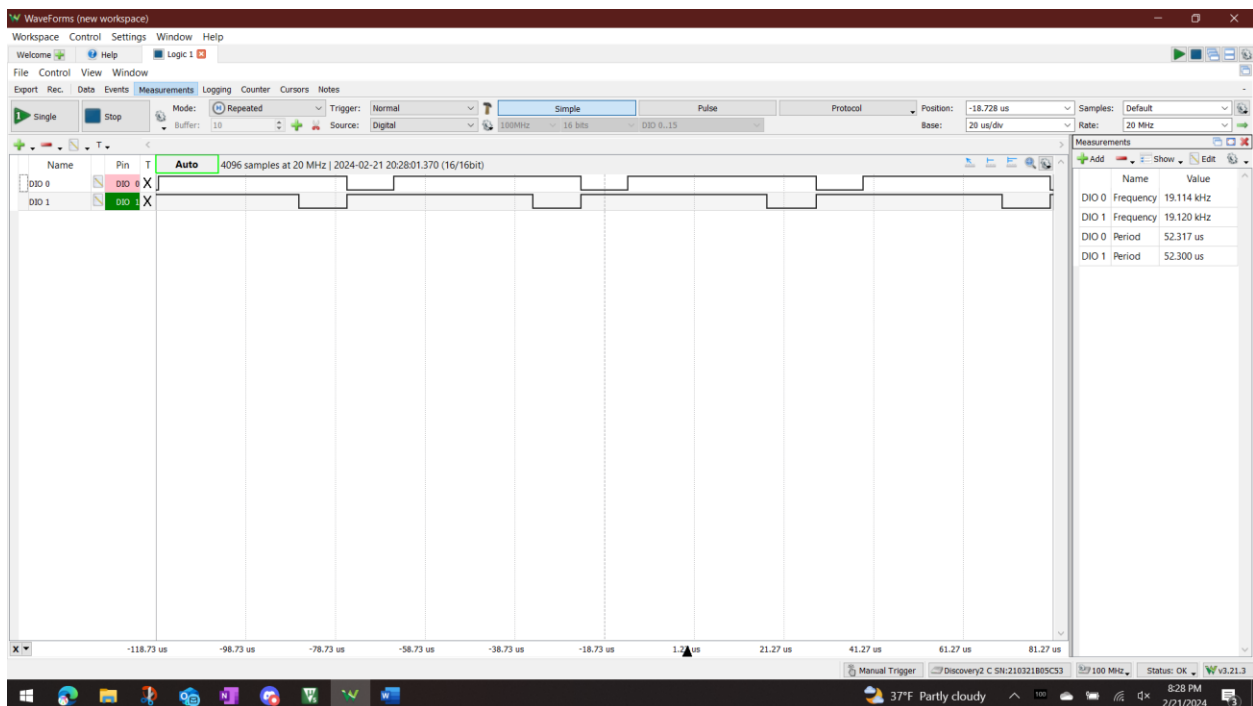


- Describe what happened to the measured duty-cycle as the CCRx value increased in PWM mode 1

Before: (value is 25)

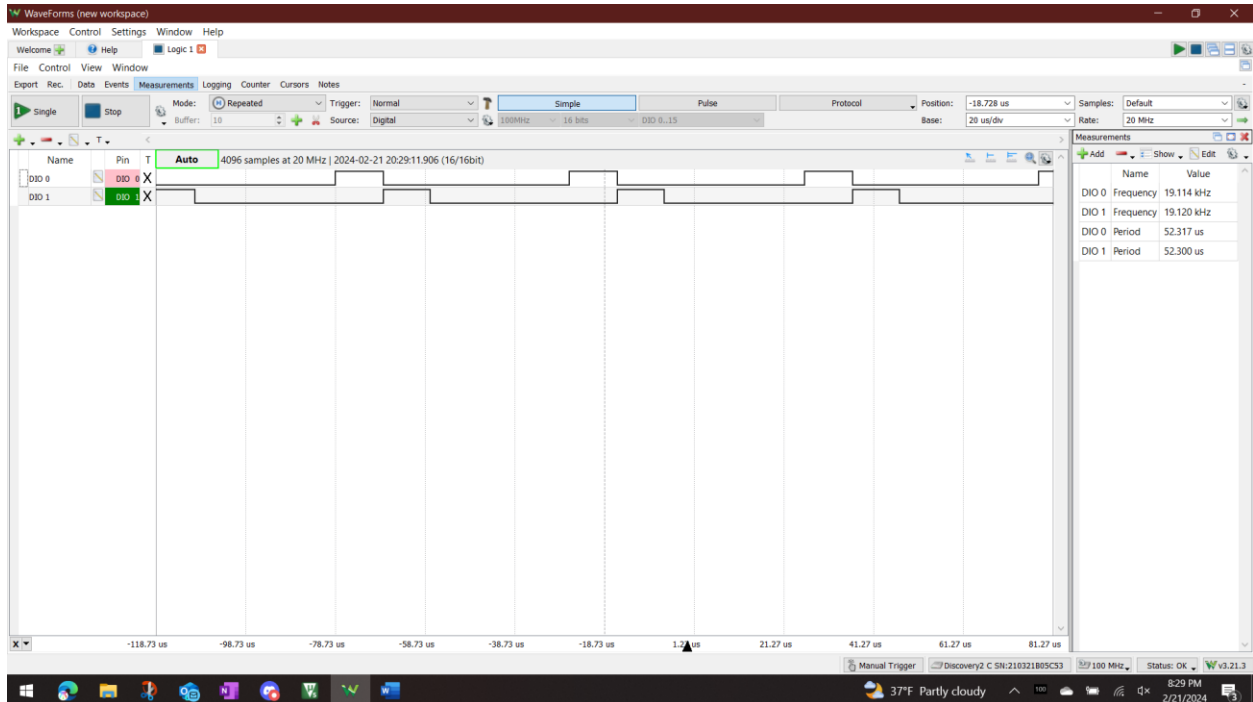


After: (value is 100)



The “on” period of the waveform very visibly lengthened.

5. Describe what happened to the measured duty-cycle as the CCRx value increased in PWM mode 2



The “off” period of the waveform visibly lengthened.

6. Include at least one logic analyzer screenshot of a PWM capture.
7. What PWM mode is shown in figure 3.6 of the lab manual (PWM mode 1 or 2)?

PWM mode 1; it is edge-aligned.