

LAB 2 – The capabilities and limits of a programmable multi-channel clock generator

Zachary Hoffman
Stanford University EE 133
(Dated: February 24, 2022)

Lab 2 of EE 133 focuses on putting together a programmable multi=channel clock generator, controlling a Si5351 clock generator over I2C using an itsy-bitsy. In the process, we discuss the function of PLLs (Phase Lock Loops) and the limitations of a cheap clock generator.

I. THEORETICAL BACKGROUND

Our clock generator main working system is a PLL (Phase Lock Loop). This is a system that uses closed loop feedback to be able to drive a voltage controlled oscillator (VCO) from an accurate reference. This system also allows us to output reference signals of various frequencies.

We can do so by changing our Loop Filter gain such that our phase detector, which drives the error signal to zero, can use the same reference. This is why we can use a single "fixed-frequency standard AT-cut crystal" that resonates at around 25 MHz to drive a wide range of frequencies. In the case of our clock generator Si5351, we have two PLLs running at $PLL_a = 900$ MHz and $PLL_b = 616.667$ MHz.

II. LAB PROCEDURE

This apparatus outputs a range of frequency from around 10 kHz to 160 MHz. It is able to accomplish this by using the PLLs and further dividing the clock frequency by some integer value.

We coded our clock generator, which has three outputs, such that $Clock0 = 112.5MHz$, $Clock1 = 13.553115MHz$, $Clock2 = 10.76kHz$. We connect the ItsyBitsy to our bread board and connected the respective SCL and SDA from the ItsyBitsy to the Si5351. This is the micro controller setup (with the appropriate libraries) that allows us to write python code to control our clock generator.

III. FINDINGS

This setup allows for a great deal of flexibility. For a given application, we could design our own PLL to work as an accurate clock. On the other hand, this programmable clock generator allows us a great deal more flexibility across many orders of magnitudes frequency range, while still having a great deal of accuracy because the clocks have PLLs backing them.

However, as we see in FIG. 3, this system is not without its flaws. Most notably, as we increase the frequency from Clock 2 to Clock 1 to Clock 0, our wave looks less and less square. This informs us that at high frequencies the effect of parasitics in our system becomes non-negligible.

This is often important, for example we may be using a clock generator to drive the LO of a switching mixer. In

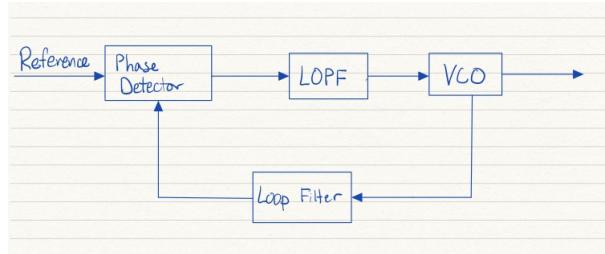


FIG. 1. Block diagram of a simple Phase Lock Loop.

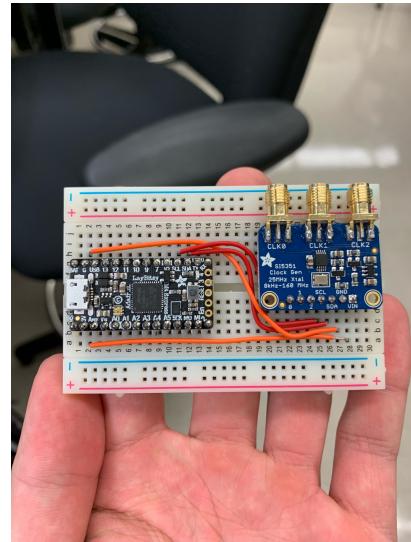


FIG. 2. ItsyBitsy and Si5351 Clock Generator on a solder-less breadboard, communicating over I2C.

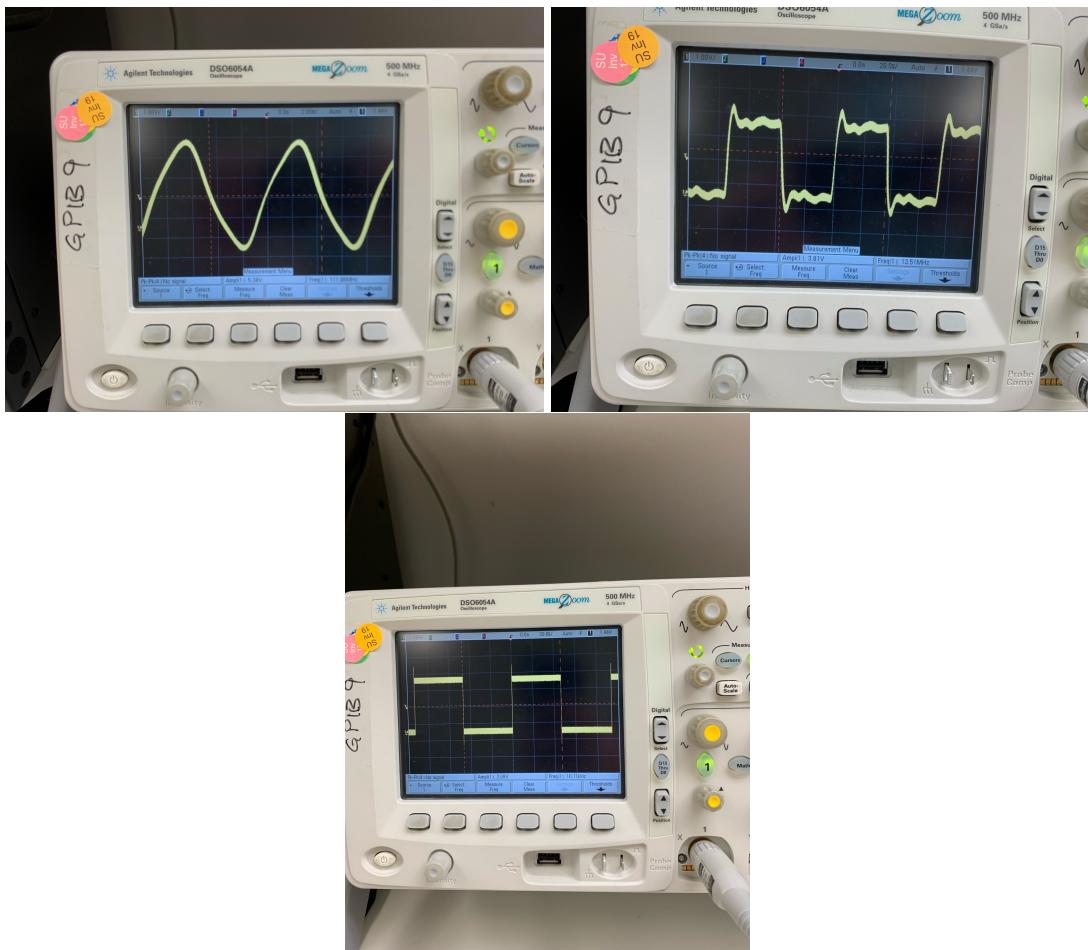


FIG. 3. From left to right, top to bottom: readings from the oscilloscope from Clock 0, Clock 1, and Clock 2.

certain constructions, a distorted square wave input to the LO will cause non-linearities in the IF signal.
