Go, go, Speedracer! Feather/Arduino loop speed workshop (circuits fall 2020)

The main thrust of this little demo/workshop is to quantify just how fast your Feather 32u4 can execute different instructions. It is inspired by tutorial from Sparkfun: https://learn.sparkfun.com/blog/1687.

Inquiring minds want to know:

a) How fast can the CPU toggle a digital pin? Use `digitalWrite()` per code example below.
b) How fast can it make an analog read? Make an ADC reading using `analogRead()`.
c) How fast can it do math? for example, take an analog reading and multiply by 3.3/1023.
d) How fast can it transmit characters over the serial port to print messages or data on the serial monitor or plotter? Use `Serial.println()` to send messages of different length)

The answer to part a) can be empirically determined can be uploading the super simple code below to your feather board, then visualizing the resulting signal with your picoscope (as in Figure 1). You can similarly empirically determine answers to b, and d by modifying this code.

```
void setup()
{
    pinMode(13, OUTPUT);
}

void loop()
{
    digitalWrite(13, HIGH);
    digitalWrite(13, LOW);
}
```
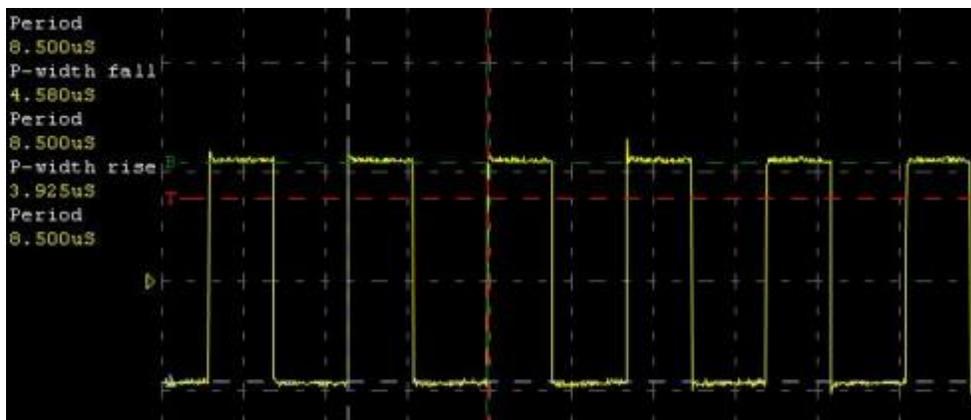

Figure 1. Square waveform resulting from continuously toggling a digital pin.

Work fast! Quantifying CPU speed on the Adafruit 32u4

Using the code provided below (also available as CPU_looptimeDemo.ino), one can tease out all sorts of information about CPU speed and timing consideration. Make sure to carefully study the code *line-by-line* so that you understand how it works. Most of it should look very familiar.

So what exactly can we learn?

a. How long does it take the cpu to execute instructions to make 2 analog reads and convert those to physical voltage? This is the result printed with Serial.println(instrTime). Measure and compute the mean and standard deviation. Is this "fast"? Relative to what?

b. How long does it take the cpu to print a numeric characters (a few bytes) to the Serial port? This is printed as serial.println(serialPrintTime). Quantify the mean and standard deviation. Is this "fast" or "slow"? Relative to what?

```
// initialize timestamp variables. these are globals.
unsigned long t1 = 0;
unsigned long t2 = 0;
unsigned long t3 = 0;

void setup()
{
  //turn on serial monitor; we'll need to print results
  Serial.begin(9600);
}

void loop()
{
  //print out timestamp as we enter loop on each iteration
  // comparing successive timestamps indicates how long each iteration takes to execute

 Serial.println(t1);

  t1 = micros(); //timestamp - start of executing the following instructions

  int a0 =     analogRead(A0);
  int a1 =     analogRead(A1);
  float Vin = a0 * 3.3 / 1023;
  float Vout = a1 * 3.3 / 1023;

  t2 = micros(); //timestamp - finsihed executing instructions above

  // amount of time (us) required to exectue instructinos bracketed by timestamps.
  float instrTime = t2 - t1;

  Serial.println(instrTime);

  t3 = micros();
  // compute amount of time required to print to serial.
  float serialPrintTime = t3 - t2;
  Serial.println(serialPrintTime);

  // Imagine we want to control the loop to iterate every 10000 us = 10 ms
  delayMicroseconds(10000); }
```