Brian Engel

Module 3 Journal

I decided to compare and contrast the three interfaces we have used in our projects so far, GPIO, PWM, and UART. GPIO or General Purpose Input / Output is designed to either send or receive a digital signal (0 or 1). On the input side it basically can tell whether an input is on or off, and on the output side it can turn an external component either on or off. PWM or Pulse Width Modulation is generally used as an output and can control the amount of power going to an external component by switching the power on and off at a specific interval known as the duty cycle. UART or Universal Asynchronous Receiver / Transmitter is used for serial communication between devices. Being asynchronous, the devices don’t need to sync their internal clocks, and use a baud rate to determine the rate the bits are transmitted and received.

GPIO would be used when you are trying to determine if the input coming in is either on or off, or if you want to set a component to on or off. We used this in the milestone this week to turn the light on and off. This was the correct choice because we didn’t have to set it to varying degrees of power and it was just on or off. PWM is used for output when you don’t want all or nothing. We used it in milestone one to set the lights to brightness levels of 90%, 10%, and 0%. This was done by setting the duty to the appropriate percentage. You can use it to turn a component all the way on and off as well, but it is more complicated than using GPIO. UART would be used for communicating between devices when more complicated communication is necessary. In milestone 2 we used UART to communicate between the microcontroller and the computer. GPIO would not have worked because we needed to communicate more than a 0 or a 1. UART made it possible to input a character on the computer and send it to the microcontroller, write it to the buffer, and then send the character back to the computer to write in the terminal.