**3-2 Journal: Peripheral Interfaces in Embedded Systems**

Chris Bridges

Southern New Hampshire University

CS 350: Emerging Sys Arch & Tech

Professor Bryant Moscon

19 July 2025

With this journal, I’ll compare three different interfaces: GPIO, A2D, and PWM. Each of these interfaces has its own advantages and disadvantages.

GPIO, which stands for General-Purpose Input/Output, allows a microcontroller to configure pins for both input and output. It is a common interface found on many microcontroller devices, such as the Raspberry Pi, Arduino, and the Flipper Zero I recently purchased. The pins allow the microcontroller to read signals from the input and output actions through the output pins. This works well with button states, relays, and LED lights since the input is binary and either on (1) or off (0).

A2D means analog to digital. This interface enables the conversion of an analog signal into a binary format for a microcontroller to interpret. This interface can take input from the real world, like light sensors and temperature sensors, and convert them into a format that a microcontroller can process. As a mechanic, I interact with sensors that use analog gears for a speedometer and convert them into a digital format for the speedometer and computer to interpret. This interface works well with A2D conversion but is limited to only that action.

PWM, or pulse width modulation, uses digital signals to manipulate outputs using a specified frequency. Duty cycle refers to the percentage of time that the signal is on. This interface can be used to control devices like electric motors, LED lighting, and solenoids using a digital signal.

Each of these interfaces excels at its designed tasks but is also limited by them. GPIO is a multi-purpose interface that allows inputs and outputs, while PWM is limited to outputs, and A2D is limited to inputs. If you wanted to use a microcontroller to control a fan, you could use GPIO to determine if the fan is switched on or off, PWM to control the fan motor’s speed, and use A2D to read data from a temperature sensor to control the fan speed dynamically. Using PWM over a GPIO’s binary output would allow the above fan’s speed to be adjustable rather than simply on or off.