

HW_5_Written_Section_Embedded_System

Question 1) The PWM interface of the GPIO pins allows for analog frequencies to be sent from the Pi to the device. Why would the Pi not allow reading analog frequencies on a GPIO pin interface?

Answer1) The GPIO pins is only capable of reading Digital Signal as it works on binary logic like most modern MCU or computers., which can only be configured in two states High and Low (0 and 1)with no other modes to modulate or play around with, PWM standing for Pusle Width Modulation, can simulate something close to Analog signal in a digital space, what it basically does it modulates the frequency and time of energy flowing imitating an analog wave.

Question2) Motors have diodes in the circuit to prevent the flow of electricity in a direction that is not desired to protect the circuit. What might we need to be able to do to allow the basic DC motor to operate forward and backwards? How would you propose to address that need in software, ignoring the hardware requirements?

Answer2) To control a DC motor using two GPIO pins, one pin can be used to set the direction, and the other can control the speed using PWM. The direction pin is configured as an output, where PI_HIGH sets the motor to move forward and PI_LOW sets it to reverse. The speed pin uses PWM to modulate the motor speed, with the duty cycle representing the speed percentage (0-100%). For example, a 50% duty cycle runs the motor at half speed. In software, a function can abstract these controls, taking inputs for the desired direction ("forward" or "reverse") and speed (0-100). This ensures efficient and compact control of the motor while minimizing GPIO usage. Proper cleanup of the GPIO pins ensures safety and prevents unexpected behavior when the program exits.