



Prática 4: Uso de modulação em largura de pulsos (PWM)

Resumo

Introdução ao uso de periféricos embarcados na raspberry PI, como a modulação em largura de pulso (PWM) para controle de saídas.

Conceitos importantes:

Modulação, Controle, Níveis analógicos, Fator de trabalho.

Objetivo

O objetivo desta prática é a familiarização do uso de conexões externas a rasp, a partir de seus pinos de entrada e saída de uso geral, utilizando de saídas moduladas em largura de pulso.

O uso do PWM em python é feito utilizando-se o mesmo módulo importado para controle das GPIOs, e em seu uso deve-se definir a frequência de operação, fator de trabalho e pino a ser utilizado.

A frequência define como os pulsos serão operados, em geral os valores de frequência para o PWM são altos, de forma a garantir que o sinal na saída se assemelhe a um nível constante, além disso, o fator de trabalho, ou duty cycle, define, dentro do período de um pulso, qual o tempo em que o sinal estará em nível alto ou baixo. Desta forma, em “0” o sinal está sempre desligado, e no máximo valor sempre ligado. Em valores intermediários, aumenta-se o tempo em que o sinal está em nível alto, gerando na saída um sinal que possui funcionamento semelhante a um nível constante de tensão intermediário.

$$Duty\ Cycle = \frac{\text{tempo em nível lógico alto}}{\text{período de frequência}}$$

Equação 01

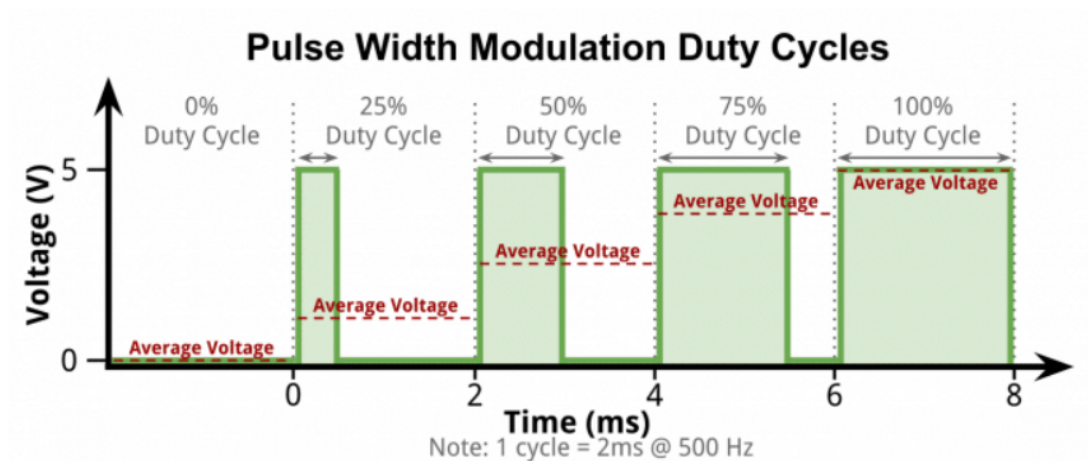


Figura 1 - Exemplo de sinal modulado com frequência de 500 Hz e 5 valores diferentes de *duty cycle*, 0%, 25%, 50%, 75% e 100%, (que pode variar entre 0% e 100). Fonte: Embarcados

Aplicação

Portanto, para a prática, deve-se executar um script em python, que seja capaz de realizar um controle de um pino na configuração de PWM, onde se possa alterar o fator de trabalho durante seu uso, além disso, para visualização, pode-se utilizar o osciloscópio presente na bancada, e a visualização a partir de um LED conectado a placa, lembre-se que a conexão do LED deve ser feita respeitando-se os limites de corrente do componente e da Raspberry PI, como segue na imagem a seguir:

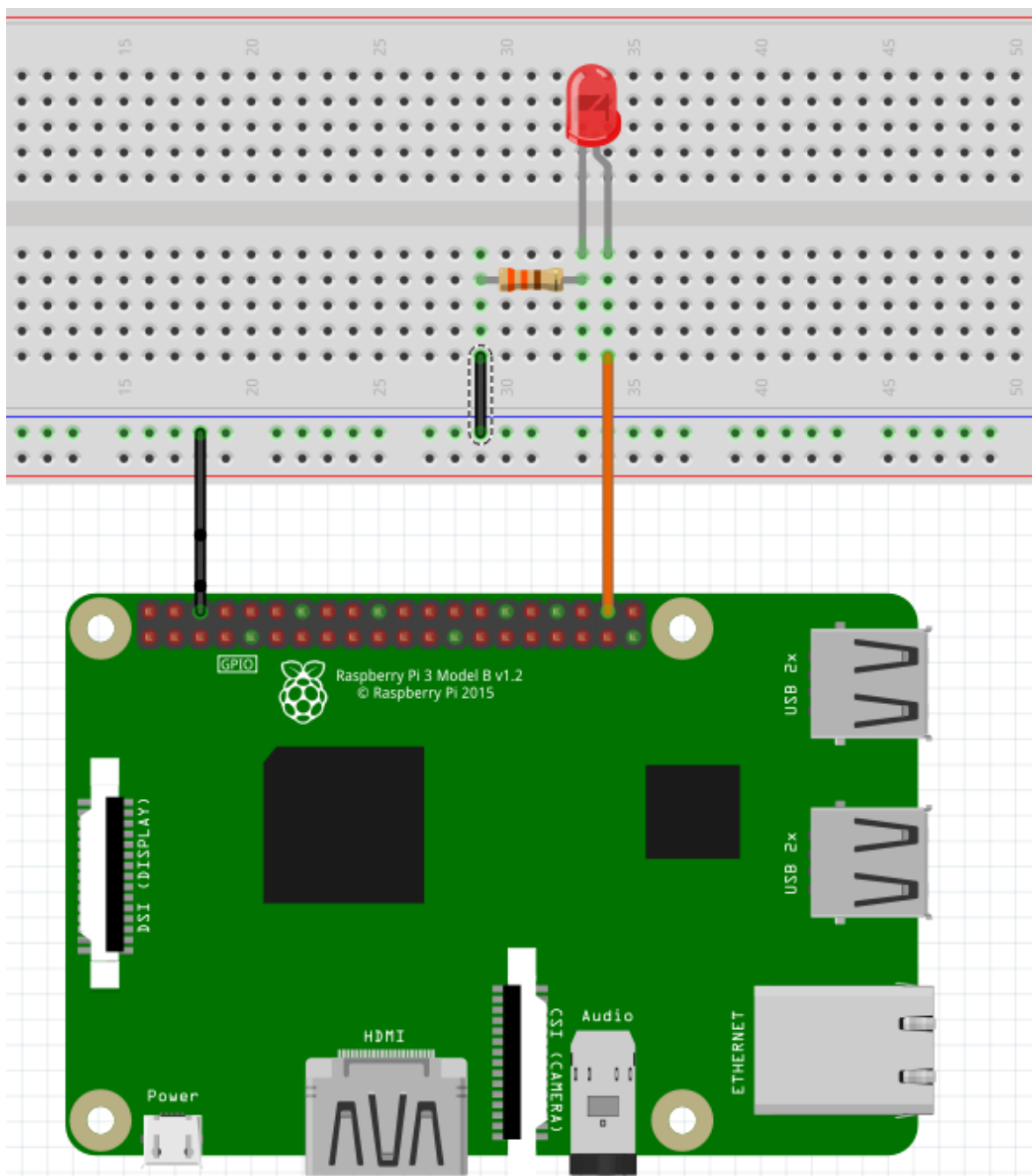


Figura 2 - Montagem para PWM - O pinout com descrição e numeração de cada um dos pinos pode ser consultado a partir da internet a partir do site de [pinout](http://pinout.raspberrypi.org) da rasp.

Motivação

O uso de PWMs é de extrema importância no contexto de sistemas embarcados, pois como visto na execução da atividade, o controle do PWM permite que os componentes conectados possam funcionar de forma a simular níveis de tensão menores do que os aplicados, sem necessariamente se reduzir a tensão. Esta utilização possui vantagens

frente a simples redução da tensão, visto que, em motores de corrente contínua, a redução de tensão provoca redução do torque do motor. Portanto, uma redução da velocidade por amplitude provocaria redução de torque, e em PWM, o torque é mantido. Além disso, em dispositivos semicondutores, como existe um limiar de tensão em que eles iniciam sua operação, é necessário que se faça um controle que não se altere o nível de tensão, porém, deve-se atentar que, apesar destes dispositivos aparentarem estar em um nível de tensão intermediário, como visto pelo osciloscópio, estes estão somente em um estado de ligar e desligar extremamente rápido. Sendo assim, aplicações muito sensíveis a altas frequências não podem ser controladas por este método. Por este motivo, os exemplos mais comuns são motores, em que há um filtro passa-baixa dentro deles que garante o nível intermediário, ou luzes, em que os olhos humanos por não possuírem alta frequência de aquisição, também operam como um filtro passa baixa na imagem recebida (por esta se tratar da soma das intensidades de energia recebidas pelo olho entre duas aquisições).

Roteiro

- Atente-se à conexão da placa, utilizando os pinos corretos, caso algum dos pinos já esteja em uso na placa, como um ground, utilize outro na placa seguindo a pinagem disponibilizada, e dê preferência a utilizar o ground para conexão do LED.
- Verifique se os estados iniciais dos pinos e do LED estão corretos e não ocasionarão nenhum curto circuito na placa.
- Importe a biblioteca RPi.GPIO para uso no projeto.
- As funções de PWM já estão desenvolvidas nesta biblioteca, veja a documentação ou exemplos na internet.
- Altere a frequência e observe a mudança ocasionada no projeto, com o LED espera-se que não seja vista alteração, porém, em osciloscópio, é possível verificar a diferença de frequência.
- Salve o histórico dos comandos em arquivo .txt.
- Desafio (para entrega futura em data a ser definida):
 - Seguindo a documentação de referência para controle PWM de servomotor:
 - <http://www.telecom.uff.br/pet/petws/downloads/apostilas/RaspberryPi.pdf>
 - <https://www.theengineeringprojects.com/2022/04/control-servo-motor-with-raspberry-pi-4-using-python.html>
 - Escreva um programa em Python que movimente um servomotor conectado a um Raspberry Pi começando em 0° e movendo-se continuamente em intervalos de 100 ms até 180° . Quando o servomotor chegar em 180° mova diretamente o ângulo para 0° e reinicie o movimento. Utilize a técnica PWM com a biblioteca RPi.GPIO. Utilize as funções de alteração de duty cycle, visando não ser necessária interrupção do programa e edição do texto a cada alteração do fator de trabalho.
 - Como não temos servomotores disponíveis para montagem prática, considere que o programa acima apenas irá simular os movimentos do

servo definindo um pino GPIO da Rasp e usando LED, conforme montagem mostrada na Fig. 2 acima.

- Bom projeto! e em caso de dúvidas contate o professor ou monitor da disciplina.