

Bancada de testes

Case Eletrônica

“Sagu” Amanda Jury Nakamura
Bacharelado em Sistemas de Informação
UTFPR – Curitiba, Centro





Tópicos

- Subsistema de eletrônica
- Sensores J13
- Protocolos de comunicação
- DAC e ADC
- Desenvolvimento do case
- Conclusão



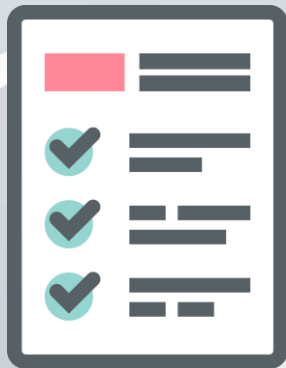
Introdução

- Subsistema de eletrônica
- Segurança
- Softwares
- Impressão 3D
- Aquisição e tratamento de dados



Problemática e Objetivos

- Testes e validação
 - Horários
 - Dependência
- Desenvolver uma bancada de testes
 - Simulação da eletrônica do veículo
 - Ondas digitais e analógicas
 - Avaliação dos sensores



Desenvolvimento

- Planejamento
- Pesquisa
- Ferramentas



Google Docs



Sensores J13



Sensor	Função	Módulo	Protocolo/Sinal
Reed Switch	RPM do motor		Analógico
Indutivo	Velocidade		Analógico
Infravermelho	Temperatura da CVT	MLX90614	I2C
Acelerômetro	Aceleração	MPU6050	I2C
Indefinido	Pressão do freio		ADC (ADS1115) -> I2C
Indefinido	Corrente		ADC -> I2C
Indefinido	Tensão		ADC -> I2C



Protocolos de comunicação

- Sinais digitais X Analógicos
- Protocolos de comunicação serial
 - Assíncronos
 - Síncronos
 - I2C

Sinal digital

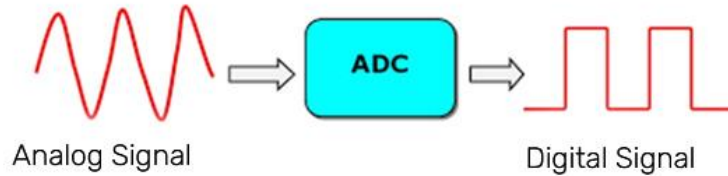


Sinal analógico

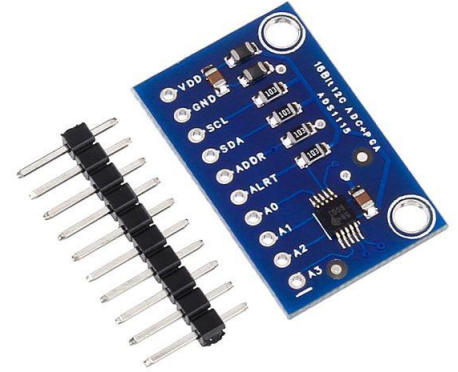


DAC e ADC

- Conversor analógico-digital
- Conversor digital-Analógico

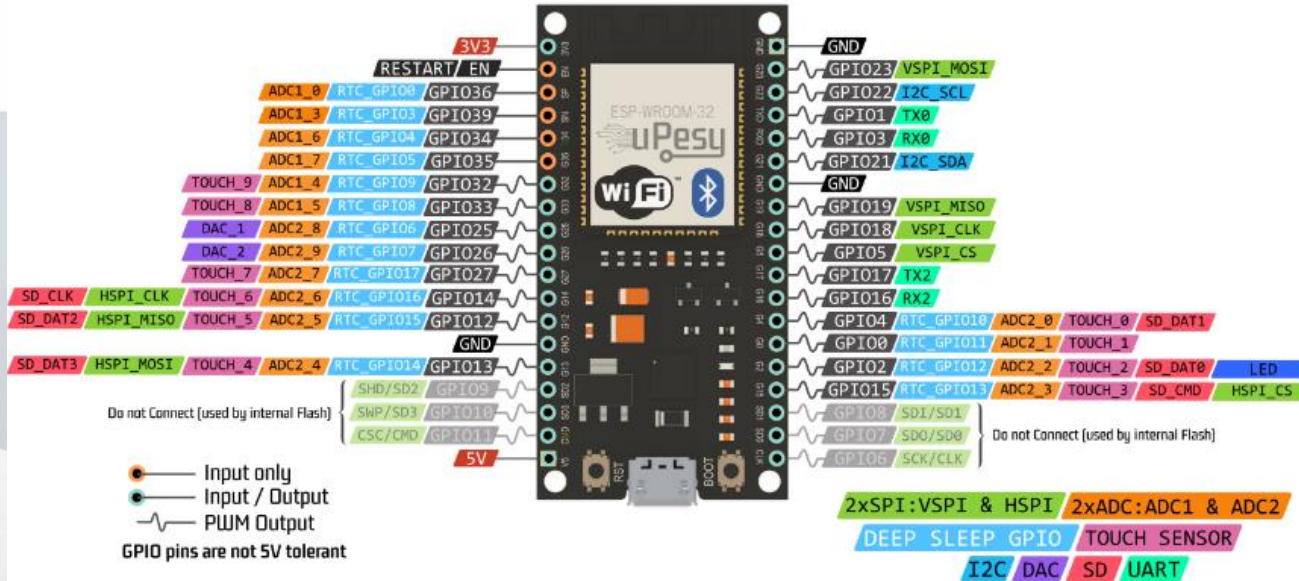


ADC and DAC Output Signal



Escolha dos pinos GPIOs

ESP32 Wroom DevKit Full Pinout



GPIO	Input	Output	Notes
0			outputs PWM signal at boot, must be LOW to enter flashing mode
1			debug output at boot
2			connected to on-board LED, must be left floating or LOW to enter flashing mode
3			HIGH at boot
4			
5			outputs PWM signal at boot, strapping pin
6			connected to the integrated SPI flash
7			connected to the integrated SPI flash
8			connected to the integrated SPI flash
9			connected to the integrated SPI flash
10			connected to the integrated SPI flash
11			connected to the integrated SPI flash
12			boot fails if pulled high, strapping pin
13			
14			outputs PWM signal at boot
15			outputs PWM signal at boot, strapping pin
16			
17			
18			
19			
21			
22			
23			
25			
26			
27			
32			
33			
34			input only
35			input only
36			input only
39			input only

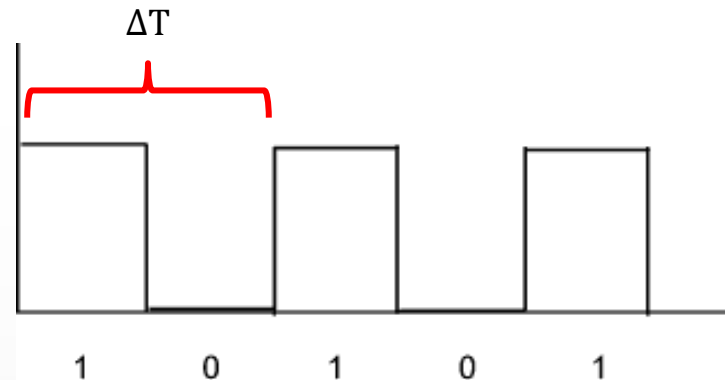
Código

- Interrupção por timer
 - Velocidade e RPM
 - Chamada da função que alterna o estado da onda
 - Frequência e período

$$\Delta T = \frac{COMP * k * 3,6}{vel}$$

$$\Delta T = \frac{60000}{rpm}$$

COMP: Comprimento da roda
k: constante referente a caixa de redução = (1000/0.7)
velocidade: velocidade escolhida para a simulação



$$\frac{\Delta T}{2 * 1000}$$



Código

```
void alterna1(void* arg) {
    estadoOnda1 = !estadoOnda1;
    digitalWrite(velPin1, estadoOnda1);
}

void setup() {
    Serial.begin(115200);
    pinMode(velPin1, OUTPUT);

    int t1 = (COMPRIMENTO * (1000 / 7.0) * 3.6) / V1;
    const esp_timer_create_args_t timer_args = {
        .callback = &alterna1,
        .arg = NULL,
        .dispatch_method = ESP_TIMER_TASK,
        .name = "timer1"
    };
    esp_err_t err = esp_timer_create(&timer_args, &timer1);
    if (err != ESP_OK) {
        Serial.println("Erro ao criar o timer");
        return;
    }
    err = esp_timer_start_periodic(timer1, t1/2*1000);
    // 500000 microsegundos = 500 ms
    if (err != ESP_OK) {
        Serial.println("Erro ao iniciar o timer");
    } else {
        Serial.println("Início do Timer com sucesso");
    }
}
```

→ Inverte o estado

→ Configura o pino

→ Configura o timer

→ Aciona a interrupção



Código

- Freio ➤ Funções millis () e Random()

```
1 unsigned long int timerFreio = 0;
2 void loop() {
3     //pressao freio -> afere acada 5s
4     pinMode(pinFreio, OUTPUT);
5     //timerFreio = tempo em ms da ultima execucao
6     //varacao de temp > 5s
7     if (millis()- timerFreio > 5000){
8         digitalWrite(pinFreio, random(0, 1));
9         timerFreio = millis();
10    }
```



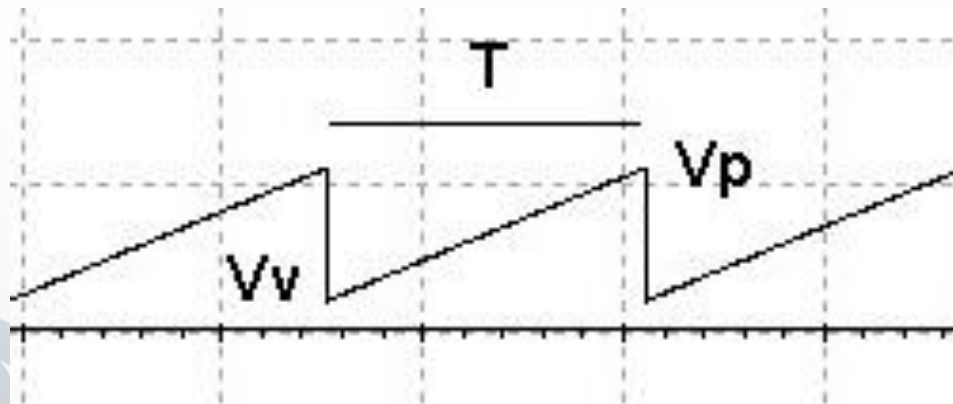
Código

- Teste do ADC – simular o DAC

```
int auxDAC = 0; //valor de entrada do DAC
//esse valor varia de 0 a 255 e volta para 0
void ondaTriangular () {
    dacWrite(pinDAC, auxDAC);
    auxDAC++;

    if (auxDAC == 256){
        auxDAC = 0;
    }
}

void loop() {
    ondaTriangular();
    delay (50);
}
```



Código

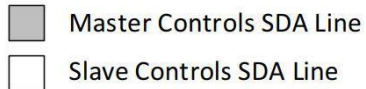
- I2C – aceleração e temperatura

Solicitação

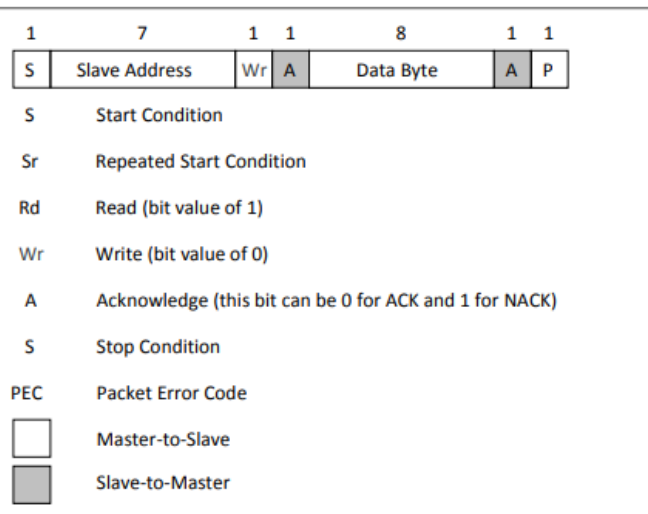
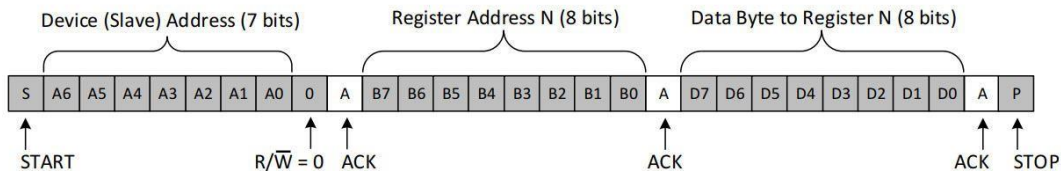
Endereço

Leitura ou escrita

Resposta



Write to One Register in a Device



Conclusão

- Simulação das ondas

- Velocidade ✓
- RPM ✓
- Freio ✓
- DAC ✓



- Planos futuros

- I2C
- Implementar e validar o código
- Testes dos sensores em bancada



Referências

1. Horowitz, P., & Hill, W. (2015). *The Art of Electronics* (3rd ed.). Cambridge University Press.
2. Tanenbaum, A. S., & Wetherall, D. J. (2011). *Computer Networks* (5th ed.). Pearson Education.
3. Harold, L. (2007). "UART Communications." In *Embedded Systems Design with the Atmel AVR Microcontroller* (pp. 115-136). Springer.
4. Smiley, J. (2011). "Using the I2C Bus." In *Arduino Workshop: A Hands-On Introduction with 65 Projects* (pp. 287-304). No Starch Press.
5. <https://www.crescerengenharia.com/post/como-instalar-driver-ch340#viewer-9os1q>
6. Franco, S. (1994). *Design with Operational Amplifiers and Analog Integrated Circuits*. McGraw-Hill.
7. Laker, K. R., & Sansen, W. M. C. (1994). *Design of Analog Integrated Circuits and Systems*. McGraw-Hill.
8. Schreier, R., & Temes, G. C. (2005). *Understanding Delta-Sigma Data Converters*. Wiley-Interscience.
9. <https://espressif-docs.readthedocs-hosted.com/projects/arduino-esp32/en/latest/api/i2c.html>
10. <https://randomnerdtutorials.com/esp32-i2c-scanner-arduino/>
11. <https://circuitdigest.com/microcontroller-projects/esp32-timers-and-timer-interrupts>
12. <https://randomnerdtutorials.com/esp32-i2c-communication-arduino-ide/>
13. Franco, S. (1994). *Design with Operational Amplifiers and Analog Integrated Circuits*. McGraw-Hill.
14. Laker, K. R., & Sansen, W. M. C. (1994). *Design of Analog Integrated Circuits and Systems*. McGraw-Hill.
15. Schreier, R., & Temes, G. C. (2005). *Understanding Delta-Sigma Data Converters*. Wiley-Interscience.



Obrigada!

Case eletrônica – Bancada de teste

Amanda Jury Nakamura “Sagu”

