



UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

RELATÓRIO COMPARATIVO DE SISTEMAS EMBARCADOS: ARDUINO VS ESP32 VS STM32

Gabriel Sampaio de Oliveira

Cruz das Almas

2023

UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

RELATÓRIO COMPARATIVO DE SISTEMAS EMBARCADOS: ARDUINO VS ESP32 VS STM32

Gabriel Sampaio de Oliveira

Relatório 1 da disciplina de Sistemas Embarcados (GCET528) - sob a orientação do Professor Dr. Igor Dantas dos Santos Miranda.

Cruz das Almas

2023

Sumário

1	INTRODUÇÃO	3
2	OBJETIVOS	4
3	FUNDAMENTAÇÃO TEÓRICA	5
3.0.1	Arduino ATmega2560	5
3.0.2	ESP32 NODEMCU-32s	7
3.0.3	STM32F4 Discovery	8
4	MATERIAIS E MÉTODOS	10
5	CONCLUSÕES	14
	REFERÊNCIAS	15

1 Introdução

Este relatório é constituído com base nos sistemas embarcados, que desempenham um papel fundamental em uma ampla gama de aplicações

O presente relatório apresenta uma sistematização da discussão acerca dos sistemas embarcados, na qual encontra-se uma ampla possibilidade de aplicações, de dispositivos IoT (Internet das Coisas) aos sistemas de controle industrial. Os sistemas utilizados no estudo foram: Arduino ATmega2560, ESP32 NODEMCU-32s e STM32F4 Discovery, os quais serão apresentados detalhadamente nas seções seguintes do relatório.

O objetivo do estudo é fornecer uma comparação prática e teórica da performance e da própria arquitetura dos sistemas mencionados. Cada sistema possui suas próprias características individuais e vantagens e desvantagens, desta forma, é fundamental compreendê-las para selecionar a plataforma mais apropriada para o desenvolvimento de um determinado projeto.

A seguir, apresentaremos uma visão geral de cada sistema e, posteriormente, detalharemos os experimentos realizados para avaliar seu desempenho e as informações relevantes sobre suas arquiteturas.

2 Objetivos

O estudo tem como objetivo geral analisar três sistemas embarcados amplamente utilizados: Arduino ATmega2560, ESP32 NODEMCU-32s e STM32F4 Discovery. Abordaremos tanto a perspectiva prática quanto a teórica, considerando o desempenho e a arquitetura de cada sistema.

Os nossos objetivos específicos são:

1. **Comparação Prática:** Implementar uma função seno baseada na série de Taylor em cada sistema e medir o período médio da oscilação.
2. **Análise Teórica:** Investigar detalhes relevantes de arquitetura, como processador, memórias, periféricos e consumo de energia.
3. **Avaliação de Custos:** Compreender o custo unitário para compras acima de 100 unidades de cada sistema.

3 Fundamentação Teórica

3.0.1 Arduino ATmega2560

Com um microcontrolador ATmega2560 desenvolvido pela empresa Microchip Technology, o Arduino Mega 2560 é projetado com 54 pinos de entradas e saídas digitais. Além disso, possui em sua placa uma memória que permite o uso de um número maior de programas do que o Arduino UNO.

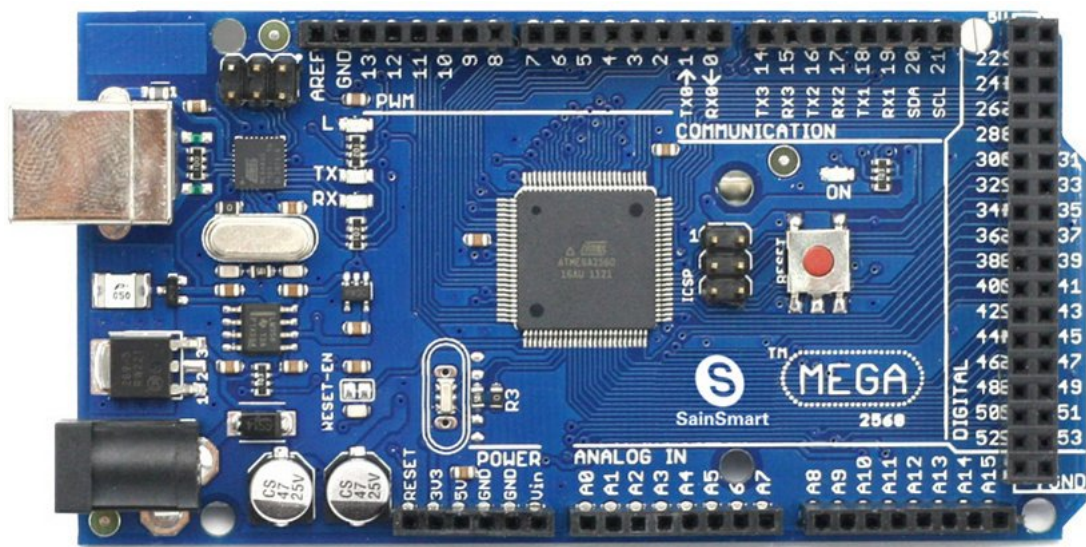


Figura 1 – Arduino ATmega2560

A seguir é apresentado, com base em sua documentação, as principais características de seu microcontrolador.

- Processador:
 - Palavra: O ATmega2560 tem uma palavra de 8 bits.
 - RISC/CISC: É baseado na arquitetura AVR-RISC de 8 bits.
 - Número de núcleos: O ATmega2560 tem um único núcleo.
 - Clock: Opera a uma frequência de 16 MHz.
 - MIPS: O ATmega2560 pode executar até 16 MIPS (milhões de instruções por segundo).
- Memórias:

- Tecnologias e capacidades: O ATmega2560 utiliza a tecnologia de memória NOR flash. Possuindo 256 KB de memória flash para armazenamento de programas e uma memória de inicialização de 1 KB usada para armazenar o bootloader. Além disso, possui 8 KB de SRAM para armazenamento de dados e 4 KB de memória EEPROM para armazenamento não-volátil dos dados (que podem ser lidos e escritos com a biblioteca EEPROM).
- Lista de Periféricos:
 - O ATmega2560 tem 54 pinos digitais de entrada/saída (dos quais 15 podem ser usados como saídas PWM), 16 entradas analógicas, 4 UARTs (portas seriais de hardware), 1 oscilador de cristal de 16 MHz, 6 timers de 16 bits, 1 timer de 8 bits, 2 interfaces SPI, 1 interface I2C, 1 conexão USB, 1 interface Ethernet, 1 interface CAN, canais ADC de 10 bits, entre outros.
- Potência consumida em standby e em operação máxima:
 - Consome cerca de 100 μ A a 500 μ A em standby, dependendo da tensão de alimentação. Em operação máxima, ele pode consumir até 30 mA em 5V. Vale ressaltar que a potência consumida pelo ATmega2560 varia de acordo com a frequência do clock e o tipo de operação realizada.
- Custo unitário para compras acima de 100 unidades:
 - Em média, o custo unitário do Arduino ATmega2560 pode variar em torno de \$8 a \$10 para compras de 100 unidades.

3.0.2 ESP32 NODEMCU-32s

O ESP32 NODEMCU-32s é um dispositivo que consiste de um microprocessador de baixa potência dual core fabricado pela empresa Espressif Systems com uma série de recursos integrados que o tornam adequado para diversas aplicações.



Figura 2 – ESP32 NODEMCU-32s

A seguir encontra-se as suas características principais.

- **Processador:**
 - Palavra: O NODEMCU-32s tem uma palavra de 32 bits.
 - RISC/CISC: É baseado na arquitetura AVR-RISC de 32 bits.
 - Número de núcleos: Dois núcleos.
 - Clock: Opera a uma frequência de até 240 MHz.
 - MIPS: É baseado no conjunto de instruções Xtensa LX6 da empresa Tensilica e pode executar até 600 MIPS.
- **Memórias:**
 - Tecnologias e capacidades: O ESP32 NODEMCU-32s utiliza a tecnologia de memória NOR flash. Possuindo 4 MB de memória flash para armazenamento de programas e uma memória de inicialização de 448 KB para o armazenamento do bootloader. Além disso, possui 520 KB de SRAM, 8 KB de memória RTC para armazenamento de dados em tempo real.
- **Lista de Periféricos:**
 - O NODEMCU-32s possui uma variedade de periféricos integrados, incluindo UART, GPIO, 18 canais ADC de 12 bits, DAC, SDIO, SD card, 16 PWMs, 2 interfaces I2C, 2 interfaces I2S, Wi-Fi 802.11b/g/n, Bluetooth v4.2 BR/EDR e BLE, entre outros.

- Potência consumida em standby e em operação máxima:
 - Consome cerca de 5 μA a 20 μA em standby. Em operação máxima, ele pode consumir até 260 mA em 3.3V.
- Custo unitário para compras acima de 100 unidades:
 - Em média, o custo unitário do ESP32 NODEMCU-32s pode variar em torno de \$3 a \$4 para compras de 100 unidades.

3.0.3 STM32F4 Discovery

O STM32F4 Discovery é um dispositivo da família de microcontroladores baseados nos núcleos RISC ARM da fabricante STMicroelectronics e que utiliza ferramentas como o STM32CubeIDE, STLink Utility e STLink.

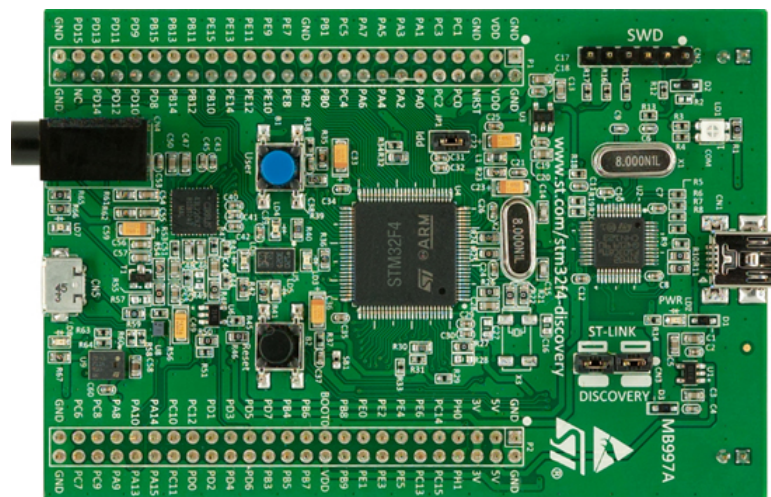


Figura 3 – STM32F4 Discovery

A seguir é apresentado suas características e principais recursos utilizados.

- Processador:
 - Palavra: O STM32F4 Discovery tem uma palavra de 32 bits.
 - RISC/CISC: Baseado na arquitetura ARM Cortex-M4, que é uma arquitetura RISC de 32 bits.
 - Número de núcleos: Um núcleo.
 - Clock: Opera a uma frequência de até 168 MHz.
 - MIPS: 210 MIPS (milhões de instruções por segundo).

- Memórias:
 - Tecnologias e capacidades: O STM32F4 Discovery utiliza a tecnologia de memória NOR flash. Possuindo 1 MB de memória flash para armazenamento de programas e uma memória de inicialização de 256 bytes para armazenar o bootloader. Além disso, possui 192 KB de memória SRAM, 4 KB de memória EEPROM para armazenamento não-volátil dos dados.
- Lista de Periféricos:
 - Possui em sua lista de periféricos integrados 4 interfaces UARTs, 3 ADCs de 12 bits, 2 DACs de 12 bits, 4 interfaces SPI, 3 interfaces I2C, 2 interfaces USB, 15 timers, unidade de debug ST Link – V2 on board, entre outros.
- Potência consumida em standby e em operação máxima:
 - Pode consumir cerca de 20 μA a 50 μA em standby, dependendo da tensão de alimentação. Em operação máxima, ele pode consumir até 150 mA em 3.3V.
- Custo unitário para compras acima de 100 unidades:
 - Em média, o custo unitário do STM32F4 Discovery pode variar em torno de \$5 a \$7 para compras de 100 unidades.

4 Materiais e Métodos

Para a realização do experimento comparativo na prática, a fim de avaliar o desempenho dos dispositivos embarcados citados na seção anterior, foi elaborado um código em C para ser implementado em ambos os sistemas. O código é simples e não foi utilizado nenhuma biblioteca.

O algoritmo representa a função seno com base na série de Taylor, considerando até o vigésimo termo da série.

$$\text{sen } x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots, \forall x \in \mathbb{R}$$

Dessa forma os algoritmos abaixo geram continuamente uma senoide com 1000 pontos de 0 a 2π , na qual não foi utilizado nenhuma função de atraso para que não ocorra interferências.

Código C utilizado para o Arduino ATmega2560 no Arduino IDE

```

1 int NUM_PONTOS = 1000;
2 double pi = 3.141592653;
3 double passo = (2 * pi) / NUM_PONTOS;
4 double angulo = 0;
5
6 double potencia(double base, int expoente) {
7     double resultado = 1.0;
8
9     for (int i = 0; i < expoente; i++) {
10         resultado *= base;
11     }
12
13     return resultado;
14 }
15
16 double fatorial(int n) {
17
18     if (n == 0 || n == 1) {
19         return 1.0;
20     }
21
22     double resultado = 1.0;
23
24     for (int i = 2; i <= n; i++) {
25         resultado *= i;
26     }
27
28     return resultado;

```

```
29 }
30
31 double seno(double x) {
32
33     double resultado = 0.0;
34
35     for (int i = 0; i < 20; i++) {
36         resultado += ((potencia(-1, i) * potencia(x, (2*i) + 1)) / fatorial
37             ((2*i) + 1));
38     }
39
40     return resultado;
41 }
42
43 void setup() {
44     pinMode(LED_BUILTIN, OUTPUT);
45     Serial.begin(115200);
46 }
47
48 void loop() {
49     angulo = 0;
50
51     for (int i = 0; i < NUM_PONTOS; i++) {
52
53         double valor_seno = seno(angulo);
54
55         if(valor_seno >= 0){
56             digitalWrite(LED_BUILTIN, HIGH);
57         }else{
58             digitalWrite(LED_BUILTIN, LOW);
59         }
60
61         angulo += passo;
62     }
63 }
```

O código acima foi implementado no Arduino ATmega2560 pelo Arduino IDE e também está disponível no [github](#), assim como os códigos utilizados no [ESP32 NODEMCU-32s](#) pelo Arduino IDE + pacote oficial da Espressif para ESP32 e no [STM32F4 Discovery](#) pelo STM32CubeIDE + STM32CubeProgrammer.

Desta forma, com um osciloscópio foi possível medir o período médio da oscilação e observar seus períodos na forma de onda. Abaixo está representado os sinais gerados por cada um dos sistemas embarcados estudados.

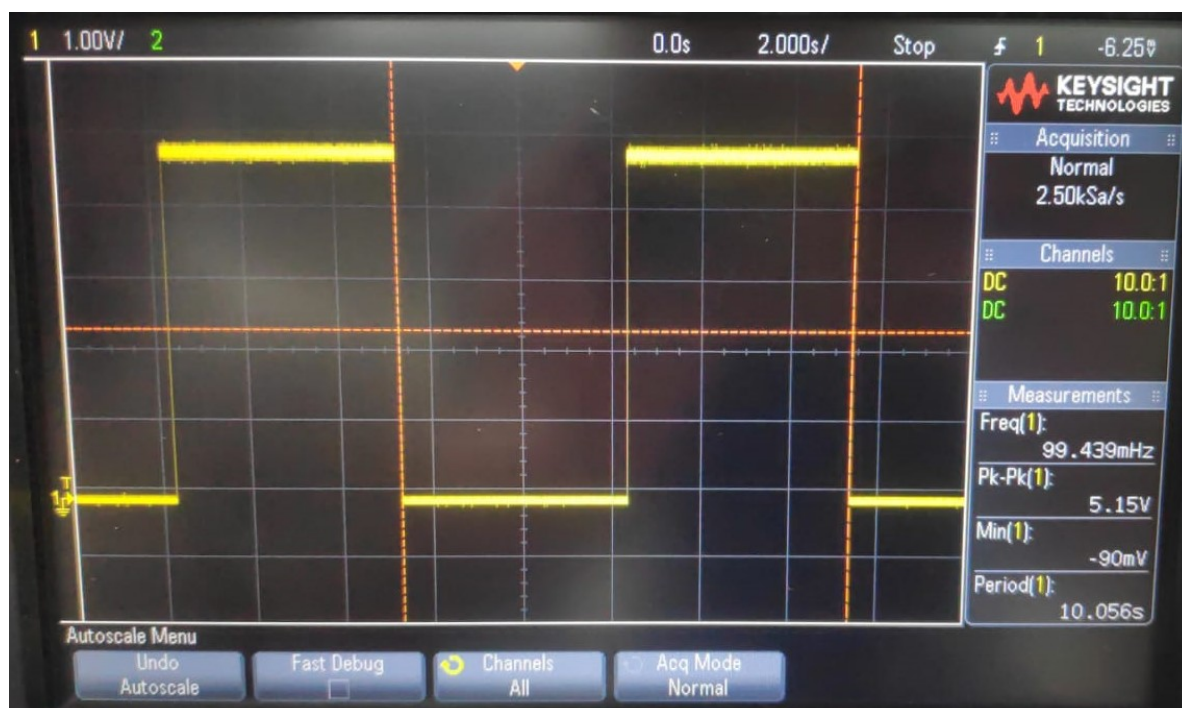


Figura 4 – Sinal produzido pelo experimento com Arduino ATmega2560 no osciloscópio.



Figura 5 – Sinal produzido pelo experimento com ESP32 NODEMCU-32s no osciloscópio.



Figura 6 – Sinal produzido pelo experimento com STM32F4 Discovery no osciloscópio.

5 Conclusões

Para uma melhor visualização e compreensão dos resultados obtidos, podemos observar na Tabela 1 que o dispositivo com melhor desempenho em nosso experimento foi o ESP32 NODEMCU-32s, com um período de 487.56 ms. Tendo assim uma melhor velocidade de processamento para executar o algoritmo proposto em comparação aos demais dispositivos.

Dispositivo	SRAM	Flash	Núcleos	Clock	MIPS	Resultado
Arduino ATmega2560	8 KB	256 KB	1	16 MHz	16	10.056 s
ESP32 NODEMCU-32s	520 KB	4 MB	2	240 MHz	600	487.56 ms
STM32F4 Discovery	192 KB	1 MB	1	168 MHz	210	7.566 s

Tabela 1 – Especificações dos sistemas analisados.

Tendo em vista o resultado da pesquisa e a análise feita a cerca das arquiteturas dos sistemas analisados, podemos explicar que a diferença de desempenho, perante o nosso experimento específico, pode ser dada por uma série de fatores, como a quantidade de núcleos, de memórias, de instruções realizadas por segundo, a frequência de clock do processador e principalmente a arquitetura do processador, e entre outros.

Como sinalizado, a arquitetura do processador pode ter uma maior relação com a diferença de desempenho. O ESP32 NODEMCU-32s que obteve o melhor resultado utiliza um processador Xtensa single-/dual-core 32-bit LX6 da família de processador RISC de 32-bit, que é muito mais poderoso que o processador baseado na arquitetura AVR utilizada pelo Arduino ATmega2560, que por sua vez é relativamente simples e sem muitos recursos avançados, e que também acaba sendo melhor que o STM32F4 Discovery que possui uma arquitetura ARM Cortex-M4.

Referências

ATMEL CORPORATION. *MEGA-2560 Datasheet*. [S.l.]. Disponível em: <<https://www.alldatasheet.com/datasheet-pdf/pdf/1425035/ETC/MEGA-2560.html>>. Nenhuma citação no texto.

ESPRESSIF SYSTEMS. *ESP32 Series Datasheet*. [S.l.], 2024. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Nenhuma citação no texto.

OLIVEIRA, E. *Conhecendo o NodeMCU-32S ESP32*. Disponível em: <<https://blogmasterwalkershop.com.br/embarcados/esp32/conhecendo-o-nodemcu-32s-esp32>>. Nenhuma citação no texto.

SHENZHEN AI-THINKER TECHNOLOGY CO. *Nodemcu-32s Datasheet*. [S.l.], 2019. Disponível em: <https://docs.ai-thinker.com/_media/esp32/docs/nodemcu-32s_product_specification.pdf>. Nenhuma citação no texto.

STM32F4 Discovery. Disponível em: <<https://stm32f4-discovery.net/stm32f429-discovery/>>. Nenhuma citação no texto.

STMICROELECTRONICS. *STMicroelectronics STM32F4DISCOVERY Datasheet*. [S.l.]. Disponível em: <<https://octopart.com/datasheet/stm32f4discovery-stmicroelectronics-20382043>>. Nenhuma citação no texto.