

**Nome: Ana Julia Ferreira, Ana Clara e Gustavo Gabriel Cunha Manhães**

**Data: 11/11/2023.**

1.

- Disponibilidade de memória RAM disponíveis nos Arduino UNO e Mega:

Arduino UNO: o Arduino UNO contém 2kB de memória RAM.

Arduino Mega: o Arduino MEGA contém 8kB de memória RAM.

- Impactos no desempenho do Web Server devido à disponibilidade de RAM:

A quantidade de RAM disponível é crucial para o desempenho de um Web Server no Arduino. Se o programa consome toda a RAM disponível, pode ocorrer uma falha ou instabilidade no sistema. Além disso, a quantidade de RAM influencia a capacidade do Arduino em lidar com solicitações de rede, armazenar buffers de dados, e manipular outras operações relacionadas ao servidor.

- Velocidade de Processamento e Relação com o Clock:

A taxa de instruções por segundo (IPS) ou, mais comumente usada, a taxa de clock, é um indicador da velocidade de processamento do Arduino.

- Arduino UNO:

O ATmega328P do Arduino UNO possui uma frequência de clock de 16 MHz.

- Arduino Mega:

O ATmega2560 do Arduino Mega também opera a 16 MHz.

- Identifique a taxa de instruções por segundo (IPS) ou outra medida equivalente para os

Arduino.

- Arduino UNO: O ATmega328P, utilizado no Arduino UNO, opera a uma frequência de clock de 16 MHz. Isso significa que ele executa 16 milhões de ciclos de clock por segundo.
- Arduino Mega: O ATmega2560, no Arduino Mega, também opera a uma frequência de clock de 16 MHz.
- Importância da Velocidade de Processamento na Capacidade do Arduino em Lidar com Solicitações de um Web Server:

A velocidade de processamento é crucial para o desempenho do Web Server. Um Web Server lida com várias operações simultâneas, como receber e processar solicitações HTTP, gerenciar conexões, e enviar dados de resposta. Quanto mais rápido o Arduino pode executar essas operações, melhor será o desempenho do servidor, especialmente em situações de tráfego intenso.

- Relação entre a Velocidade de Processamento e a Frequência do Clock:

A variação na frequência do clock influencia diretamente a velocidade de processamento. Quanto maior a frequência do clock, mais instruções o processador pode executar em um determinado período de tempo. No entanto, é importante notar que um aumento na frequência do clock também pode aumentar o consumo de energia.

2.

#### Capacidade de Armazenamento no SD:

- A capacidade máxima de armazenamento de um cartão SD compatível com o Arduino pode variar, mas os Arduinos geralmente suportam cartões SD de até 32 GB. Esta capacidade pode variar dependendo do modelo específico de Arduino e do suporte do hardware.
- O uso de um cartão SD para armazenar dados, como páginas web, pode liberar a memória flash interna do Arduino. Isso é crucial porque a memória flash é usada para armazenar o programa (código) do Arduino. Ao transferir dados, como páginas web, para o cartão SD, você evita sobrecarregar a memória flash interna, deixando mais espaço disponível para o programa.

#### Eficiência e Limitações do Armazenamento em SD:

- Espaço adicional: Cartões SD oferecem muito mais espaço de armazenamento do que a memória flash interna do Arduino.
- Flexibilidade: Pode-se facilmente atualizar ou modificar conteúdos no cartão SD sem precisar reprogramar o Arduino.
- Velocidade: O acesso a dados no cartão SD pode ser mais lento em comparação com a memória flash interna, o que pode afetar o tempo de resposta do Webserver.
- Complexidade: O manuseio de arquivos em um cartão SD pode ser mais complexo do que o acesso à memória flash interna.
- O acesso aos arquivos no cartão SD pode impactar o desempenho do Webserver devido à velocidade de leitura e gravação no cartão. Operações frequentes de leitura/gravação podem introduzir atrasos nas respostas do servidor, especialmente em ambientes com alto tráfego.

#### Limitações e Considerações Importantes:

- Memória RAM: O Arduino possui limitações de memória RAM, e o uso de cartões SD pode ajudar a aliviar esse problema ao armazenar dados externamente.
- Velocidade de Acesso: A velocidade de acesso ao cartão SD deve ser considerada, especialmente em aplicações que exigem respostas rápidas.

#### Discussão em Sala de Aula:

- Estratégias de Otimização: Explorar estratégias para otimizar o uso da memória RAM, como minimizar o tamanho de buffers ou utilizar técnicas de compressão de dados.
- Caching: Discutir a implementação de estratégias de caching para minimizar operações frequentes de leitura no cartão SD.

- Trade-offs: Compreender os trade-offs entre o uso de memória flash interna, memória RAM e armazenamento externo no contexto de aplicações web.
- Promover a conscientização sobre essas limitações e discutir estratégias de otimização pode ser fundamental para o desenvolvimento eficiente de aplicações web utilizando Arduinos como Webserver.

**Data: 17/11/2023**

### **Questionário sobre Exercícios com Arduino e Node-RED**

Questão 1:

Qual biblioteca é comumente usada para estabelecer a conexão Ethernet no Arduino?

- ☒ a) Ethernet2.h
- ☐ b) WiFi.h
- ☐ c) Ethernet.h
- ☐ d) Internet.h

Questão 2:

Qual é a porta padrão em que um Webserver no Arduino geralmente escuta por requisições HTTP?

- ☐ a) 8080
- ☒ b) 80
- ☐ c) 8000
- ☐ d) 8888

Questão 3:

No Arduino, qual função é usada para definir um pino como saída?

- ☒ a) `pinMode()`
- ☐ b) `setOutput()`
- ☐ c) `outputMode()`
- ☐ d) `digitalWrite()`

Questão 4:

Em um endpoint Restful, qual método HTTP é geralmente usado para obter dados do servidor?

- a) POST
- b) DELETE
- c) PUT
- x) GET

Questão 5:

Qual é a diferença entre o método GET e o método POST em requisições HTTP?

- a) GET é para enviar dados e POST é para receber dados.
- x) GET é para receber dados e POST é para enviar dados.
- c) Ambos são usados para a mesma finalidade.
- d) GET e POST são métodos obsoletos em requisições HTTP.

Questão 6:

Qual comando no Arduino é usado para ligar uma saída?

- a) `output(HIGH)`
- b) `set(HIGH)`
- c) `digitalWrite(LOW)`
- x) `digitalWrite(HIGH)`

Questão 7:

Para controlar uma saída do Arduino via Webserver, qual caracter é tipicamente enviado na requisição para ligar a saída?

- a) 'ON'
- b) '1'
- x) 'HIGH'
- d) 'TRUE'

Questão 8:

Para configurar um fluxo no Node-RED para se conectar ao Webserver do Arduino, qual nó é comumente usado?

- a) TCP Request
- ☒ x) HTTP Request
- c) Ethernet Connection
- d) Web Connect

Questão 9:

Qual é o objetivo de criar endpoints Restful?

- a) Simplificar o acesso a servidores web.
- b) Permitir a comunicação de dispositivos sem fio.
- ☒ x) Organizar e padronizar a interação com recursos de um servidor.
- d) Conectar-se a APIs externas.

Questão 10:

Em um fluxo no Node-RED, qual nó seria utilizado para processar dados recebidos de um Webserver do Arduino?

- a) HTTP Response
- ☒ x) Function
- c) Debug
- d) Template