

Nome: _____ Matric.: _____

SD Card (Básico)

Para que serve o SD Card?

O SD Card serve basicamente para guardar informações em uma memória não-volátil, ou seja, os dados não irão se perder quando o Arduino for desligado. Isso é muito útil quando se está fazendo medições em lugares sem acesso a um computador, como em um campo ou mata. Desse modo, o SD Card é a melhor solução para criação de Dataloggers quando não se pode ter um computador nem uma rede junto do Arduino.



A Prática

Para uso do Cartão SD é necessário o uso das bibliotecas SPI e SD, que já estão nas bibliotecas padrão do Arduino.

Geralmente, o cartão SD é usado junto de um módulo Ethernet, que juntos formam o Shield Ethernet. Por ser um Shield, os pinos da placa já se conectam com o Arduino ao conectar o Shield. Os pinos necessários para o Shield são:

- D2 - Ethernet interrupt
- D4 - SD SPI CS
- D10 - Ethernet SPI CS
- D11 - SPI MOSI
- D12 - SPI MISO
- D13 - SPI SCK

A0 - SD Write Protect

A1 - SD Detect

Por enquanto não usaremos a parte Ethernet do Shield, porém as portas listadas acima continuarão inutilizáveis por nós.

O código a seguir mostra como inicializar um SD e como criar um arquivo:

```
#include <SD.h>
File myFile;

void setup() {
  // Abre a comunicacao serial:
  Serial.begin(9600);
  while (!Serial)
  {
    ; //espera o serial comunicar, necessario apenas no Leonardo
  }
  Serial.print("Initializing SD card...");
  // No Shield Ethernet, CS é o pino 4. É definido
  // Como saída por padrão.
  // Observe que, mesmo se ele não for usado como o pino CS, o
  // Pino SS do hardware
  // (10 na maioria das placas Arduino, 53 na mega) deve ser
  // Deixado como uma saída
  // Ou as funções da biblioteca SD não vão funcionar.
  comandosSD();
}

void loop () { }

void comandosSD(){
  if (!SD.begin(4))
  {
    Serial.println("inicializacao falhou!");
    return;
  }
  Serial.println("inicializacao completa!");
  if (SD.exists("exemplo.txt")) //verificando se o arquivo existe
  {
    Serial.println("exemplo.txt existe.");
  }
  else
  {
    Serial.println("exemplo.txt nao existe.");
  }

  Serial.println("Criando exemplo.txt...");

  // abre um novo arquivo e fecha ele em seguida:
  myFile = SD.open("exemplo.txt", FILE_WRITE);
  myFile.close();
}
```

Exercícios

1. Modifique o programa de exemplo para que o arquivo seja removido caso ele exista. Dica: use a função `SD.remove("arquivo")`;
2. Crie um novo programa que escreva um valor no arquivo.
3. Modifique o programa anterior para que o valor escrito tenha sido recebido via serial monitor.
4. Modifique o programa anterior para que o que for escrito no arquivo apareça no serial monitor.

Dicas: Use as funções `MyFile.read()`, `MyFile.println`, `Serial.write()`, `Serial.read`.