

Projeto: RelayBoard
Desenvolvedores: Gustavo Suim e Vitor Paganini
Incubadora: Laboratório de Garagem

Biblioteca - RelayBoard

#include <RelayBoard.h>

Declaração necessária para utilização da placa.

RelayBoard::RelayBoard(int date, int str, int clk, int numberboard)

- **Date:** Pino responsável pelo envio dos dados responsáveis pelo controle das saídas.
- **Str:** Pino responsável por disponibilizar nas saídas as entradas carregadas.
- **Clk:** Pino responsável pelo clock das placas.
- **Numberboard:** variável que informa para todo o sistema a quantidade de placas acopladas.

Ex.: RelayBoard relay(2,4,6,2); // cria o objeto "relay" com as devidas propriedades

Observação: TODOS os pinos desta função são configurados como OUT automaticamente.

void RelayBoard::set(int shield, int out, boolean status)

- **Shield:** Informa qual a placa que deverá ser utilizada
- **Out:** informa qual a saída da placa informada acima deverá ser acionada (ON/OFF).
- **Status:** Devemos informar "1" para LIGAR a saída ou "0" para DESLIGAR a saída.

Ex.: relay.set(0,3,1);

Observações:

- A contagem de placas e saídas iniciam-se com o valor 0 (zero).
- Esta função realiza o acionamento da saída sempre que for utilizada.

void RelayBoard::load(int board, int out, boolean status)

- **Board:** Informa qual a placa que deverá ser acionada
- **Out:** informa qual a saída da placa informada acima deverá ser acionada.
- **Status:** Devemos informar "1" para LIGAR a saída ou "0" para DESLIGAR a saída.

Ex.: relay.load(0,2,1);

Observação: Esta função carrega o STATUS desejado das saídas, porém NÃO faz o acionamento das mesmas.

void RelayBoard::go()

Após carregar as saídas solicitadas na função LOAD, a função GO faz o acionamento simultâneo das saídas solicitadas.