

Projeto 04

Controle por Notificação – Prática

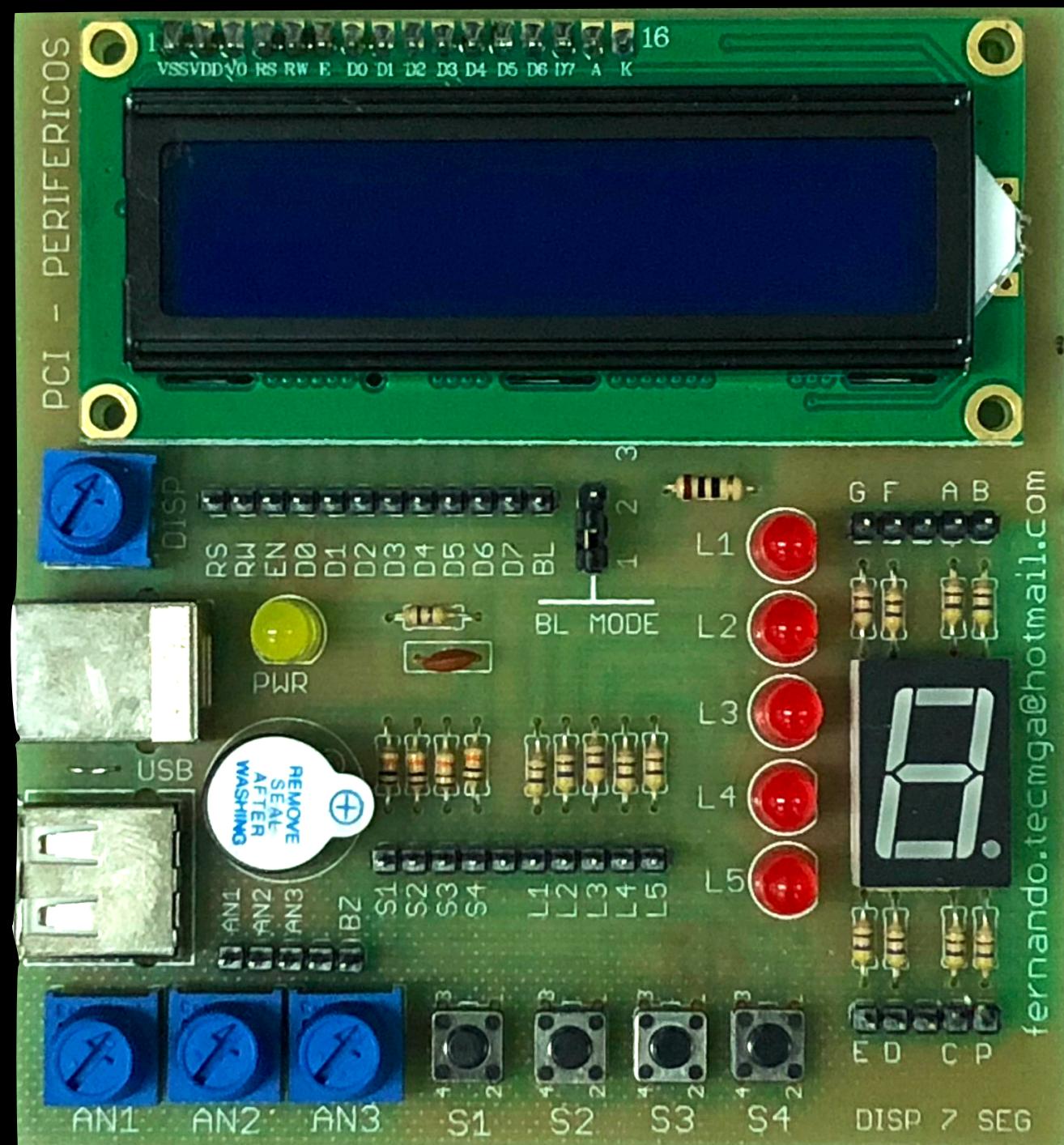
Jan K. S. – janks@puc-rio.br

ENG1419 – Programação de Microcontroladores

Testes Iniciais

GPIO 2, 3, 4, 5, 6, 7

GPIO 16



GPIO 11, 12, 13, 14

GPIO 21
GPIO 22
GPIO 23
GPIO 24
GPIO 25

Conexões com as Portas da GPIO



Testes Iniciais

Ao apertar o botão 1, exiba o texto "Gravando..." no LCD, **grave 5 segundos de áudio** e apague o texto. Abra o arquivo manualmente para ouvir a gravação.
↪ DICA: use a função **system**.

Ao apertar o botão 2, tire **5 fotos com a webcam**, com 2 segundos de intervalo entre elas. Abra os 5 arquivos manualmente para visualizar as imagens.
↪ DICA: use um for, variando o nome do arquivo.

Pisque **uma vez** rapidamente o LED 1 após capturar cada foto no item anterior.

Configure o seu bot no Telegram, anotando a chave secreta. Em seguida, inicie uma conversa com ele pelo aplicativo, e obtenha o id do chat pelo navegador.
↪ DICA: existe uma versão web do aplicativo em web.telegram.org.

Ao apertar o botão 3, **envie uma mensagem de texto** qualquer para a conversa com o bot.

```
>>> endereco = endereco_base + "/sendMessage"
>>> dados_incompletos = {"text": "Olá!"}
>>> resposta = post(endereco, data=dados_incompletos)
>>> print(resposta.text)
{"ok":false,"error_code":400,"description":"Bad
Request: chat_id is empty"}
```

DICA: Mensagens de Erro do Telegram (Caso Algo Não Esteja Funcionando)

Implementação



Visitantes?

Controle de Acesso para Visitantes



notificação com foto

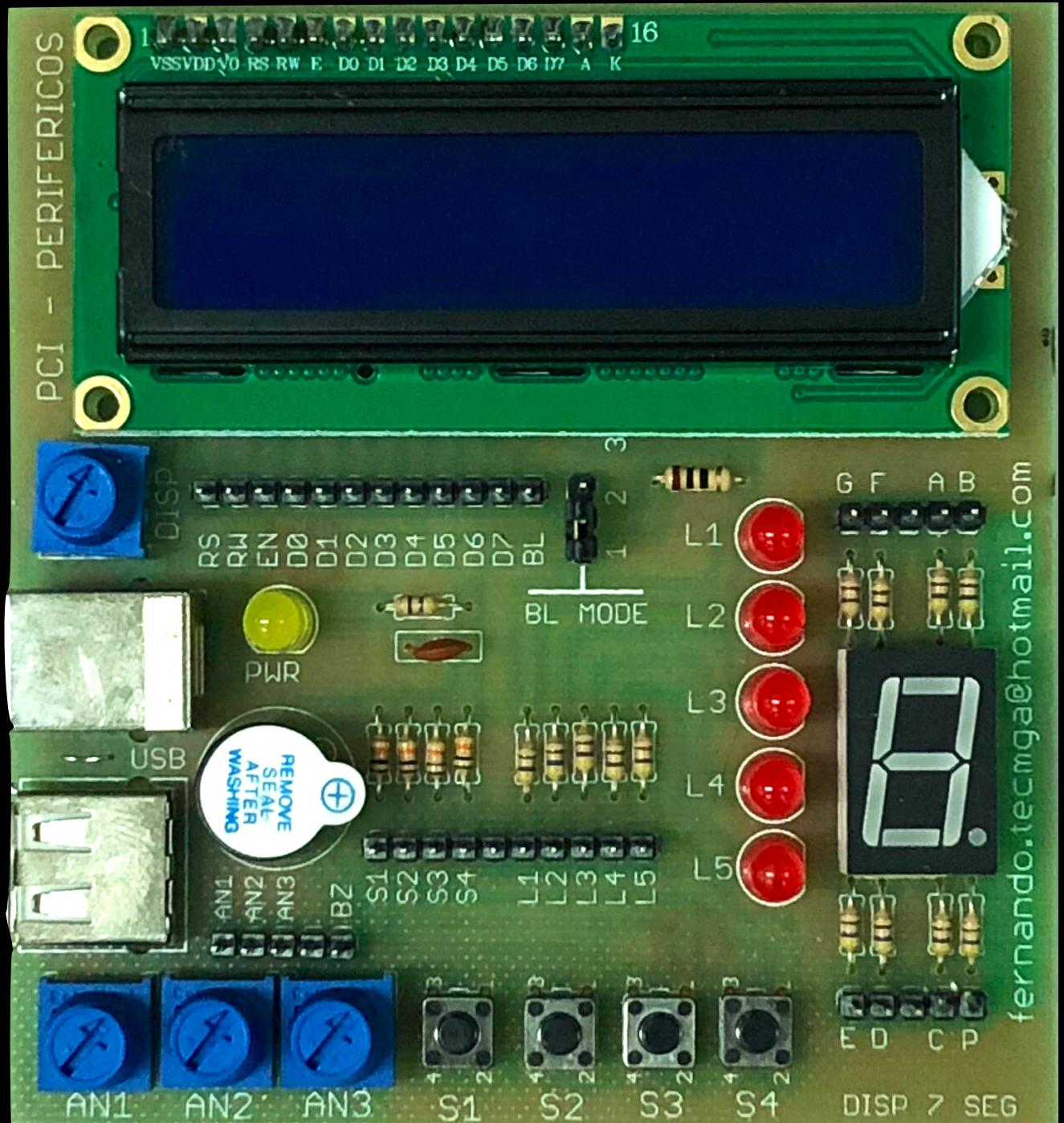


comando



Controle de Acesso por Notificação

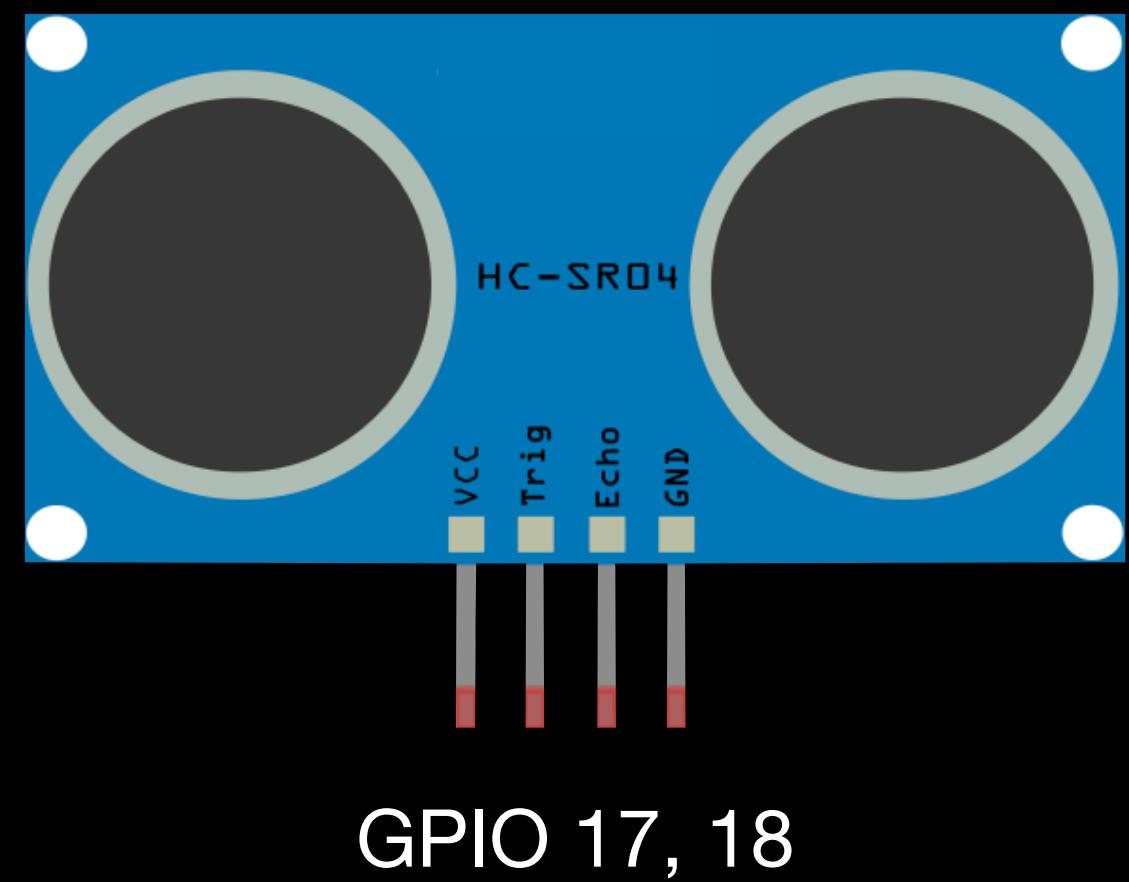
GPIO 2, 3, 4, 5, 6, 7



GPIO 16

GPIO 11, 12, 13, 14

GPIO 21
GPIO 22
GPIO 23
GPIO 24
GPIO 25



GPIO 17, 18

Conexões com as Portas da GPIO



Implementação

Ao apertar o Botão 1, ligue a campainha. Ao soltá-lo, desligue a campainha.

Também ao soltar o Botão 1, envie uma mensagem para o chat avisando que alguém está na porta.

Logo após a mensagem anterior, envie também uma foto tirada pela webcam, para mostrar quem está na porta.

Ao receber uma mensagem "Abrir", acenda o LED 1 (representando a abertura da porta). Ao receber "Alarme", toque a campainha 5 vezes rapidamente.

Ao apertar o Botão 2, apague o LED 1 (representando o fechamento da porta).

...

```
print("Enviando foto...")
```

...

```
print("Foto enviada!")
```

...

```
print("Enviando mensagem de texto...")
```

...

DICA: Acompanhamento das Etapas com Print

Aperfeiçoamento



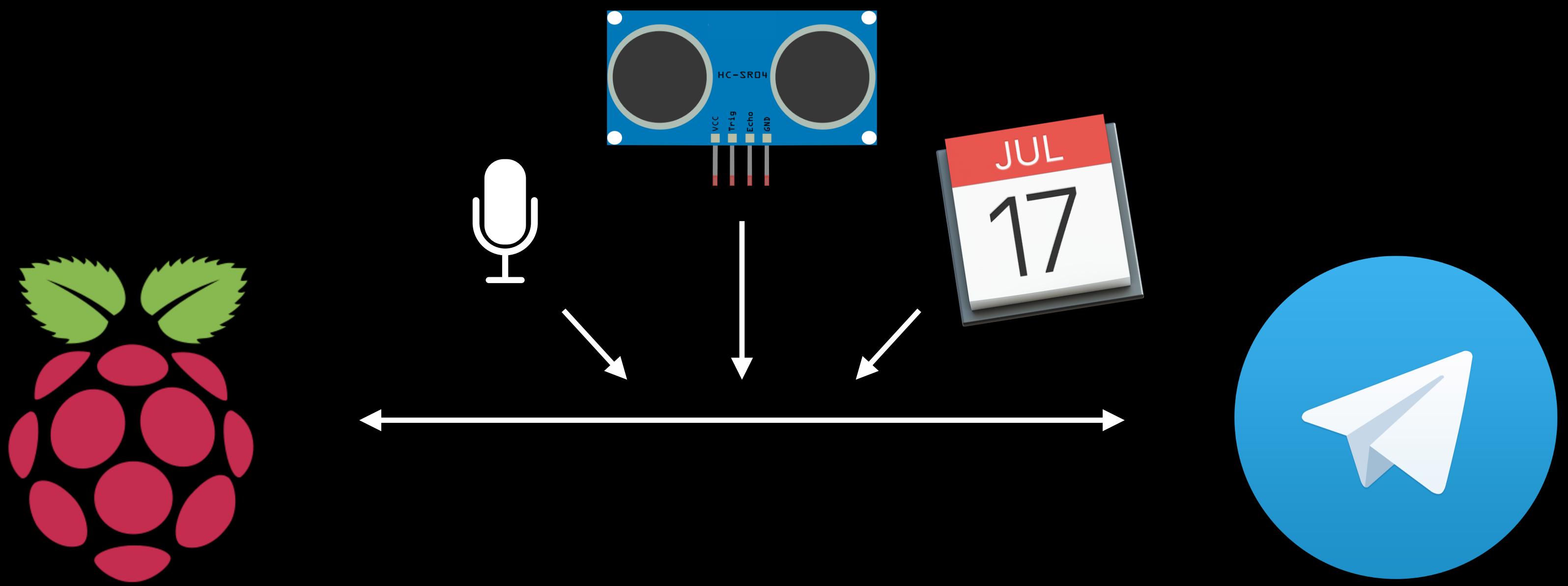
04b_implementacao.py

cópia
----->



04c_aperfeicoamento.py

Cópia do Código da Implementação para o Aperfeiçoamento



Inclusão de Áudio, Data/Horário e Distância na Comunicação



Aperfeiçoamento

Ao pressionar o Botão 3, comece a **gravar um áudio**. Ao soltar, **finalize a gravação**, converta o arquivo para OGG e envie-o para o chat.
↪ DICA: use o Popen.

Ao receber um **arquivo de áudio** do chat, toque-o no Raspberry Pi.
↪ DICA: use o mplayer.

Crie uma lógica que **impeça o envio de mais de um alerta+foto num intervalo de 15 segundos** ao soltar o Botão 1.

↪ DICA: use as funções de data/horário da aula passada.

Caso a pessoa tenha **ficado ao menos 10 segundos em frente à porta e saído**, envie uma mensagem de alerta "Pessoa saiu".

↪ DICA: use os eventos de proximidade do sensor de distância e as funções de data/horário.

Desafio Extra



04c_aperfeicoamento.py

cópia
----->



04d_desafio.py

Cópia do Código da Implementação para o Aperfeiçoamento

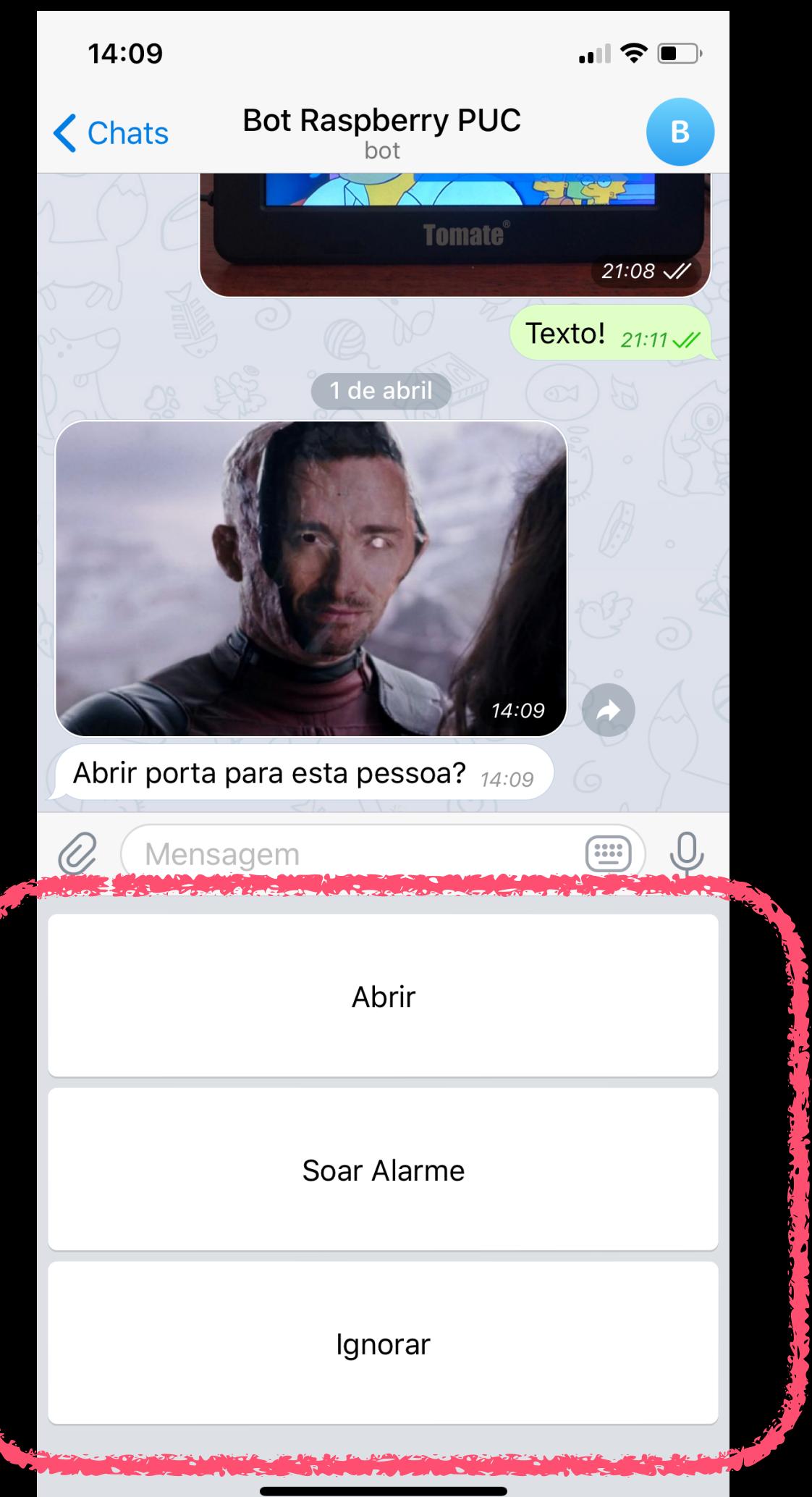
alerta piscante e sonoro



mensagem rolando



e



Teclado de Respostas Pré-Definidas e Outras Mensagens Recebidas no LCD



Desafio Extra

Ao receber um texto que não seja "Abrir", "Soar Alarme" ou "Ignorar", pisque duas vezes "Mensagem Recebida" no LCD e soe duas vezes a campainha. Em seguida, role a mensagem recebida uma vez no display, removendo os acentos antes.

↪ DICA: você pode usar a função sleep. Pesquise no Google como remover os acentos de uma string em Python.

Ao enviar o texto de alerta após apertar o Botão 1, configure opções pré-definidas de resposta para "Abrir", "Soar Alarme" e "Ignorar". A opção de alarme deve soar a campainha.

↪ DICA: consulte a documentação do Telegram em core.telegram.org/bots/api#sendmessage, e envie um dicionário como parâmetro extra nos dados da sendMessage.



janks.link/micro/projeto04.zip

Material do Projeto 04