Inicia-se a conversão do código de C++ para Python

28/02/2024 – 03:51

[Código atual em C++](C:\\Users\\diego\\Desktop\\001-Desktop\\diegoPaladino\\Praditto\\MÁQUINAS\\CHARNEIRA\\CHARNEIRA_CODE\\PYTHON_CODE\\C++270220241139.txt)

O código que você compartilhou é um projeto C++ para Arduino, que envolve a utilização de vários componentes como um display LCD I2C, botões com debouncing, e controle de motores e LEDs. Para converter esse código para Python, usaremos a biblioteca gpiozero para a Raspberry Pi ou machine e esp8266 para microcontroladores ESP, dependendo do hardware que você planeja usar, além de RPi.GPIO para controle de GPIO em uma Raspberry Pi, se for o caso.

Aqui estão os passos principais e conceitos que aplicaremos na conversão:

Iniciar e Configurar Dispositivos: Converteremos as inicializações de dispositivos como o LCD e os pinos dos botões e motores de setup() para a inicialização em Python usando bibliotecas adequadas.

Debouncing de Botões: Utilizaremos funcionalidades existentes nas bibliotecas Python para implementar o debouncing de botões.

Leitura e Controle de GPIO: Converteremos as leituras de pinos e controle de dispositivos para o formato Python, usando gpiozero ou RPi.GPIO.

Loop Principal e Controle de Fluxo: Recriaremos a lógica de controle de fluxo do loop() principal, adaptando-a ao estilo Python.

Funções e Estruturas de Dados: Convertiremos as funções e estruturas, como a struct Movement, para Python, utilizando classes ou dicionários conforme necessário.

Para a conversão do código, precisaremos adaptar algumas partes específicas para Python, como a inicialização e uso do display LCD, que pode ser feito com uma biblioteca Python para LCDs I2C, e a substituição de Bounce2 para debouncing por uma abordagem Python, possivelmente com gpiozero ou manipulação manual de estados em Python.

[Versão 1.0 do código em Python](file:///C:\Users\diego\Desktop\001-Desktop\diegoPaladino\Praditto\MÁQUINAS\CHARNEIRA\CHARNEIRA_CODE\PYTHON_CODE\CharneiraPython-1.0.txt)