|  |  |
| --- | --- |
| Học trực tuyến CNTT, học lập trình từ cơ bản đến nâng cao |  |

**IOT303 – Lập trình C nhúng nâng cao cho vi điều khiển**

**THUYẾT MINH ASSIGNMENT 4**

**MÔ PHỎNG TÍNH NĂNG ĐIỀU CHỈNH ĐỘ SÁNG MÀN HÌNH TỰ ĐỘNG TRÊN ĐIỆN THOẠI**

|  |  |
| --- | --- |
| Tên học viên:  Mã học viên:  Ngày báo cáo: | **Nguyễn Hoàng Dũng**  **FX19603**  **07/06/2024** |

**HÀ NỘI, 06/2024**

**MỤC LỤC**

[1. Danh sách thư viện 4](#_Toc168699935)

[2. Danh sách các macro và các biến toàn cục 4](#_Toc168699936)

[3. Danh sách các hàm 5](#_Toc168699937)

[4. Hàm main ( ) 6](#_Toc168699938)

[4.1. Mô tả 6](#_Toc168699939)

[4.2. Tham số truyền vào 6](#_Toc168699940)

[4.3. Giá trị trả về 6](#_Toc168699941)

[5. Hàm AppInitManager ( ) 6](#_Toc168699942)

[5.1. Mô tả 6](#_Toc168699943)

[5.2. Tham số truyền vào 6](#_Toc168699944)

[5.3. Giá trị trả về 6](#_Toc168699945)

[6. Hàm LightSensor\_AdcInit ( ) 7](#_Toc168699946)

[6.1. Mô tả 7](#_Toc168699947)

[6.2. Tham số truyền vào 7](#_Toc168699948)

[6.3. Giá trị trả về 7](#_Toc168699949)

[7. Hàm LedControl\_TimerOCInit ( ) 7](#_Toc168699950)

[7.1. Mô tả 7](#_Toc168699951)

[7.2. Tham số truyền vào 7](#_Toc168699952)

[7.3. Giá trị trả về 7](#_Toc168699953)

[8. Hàm LightSensor\_AdcPollingRead ( ) 7](#_Toc168699954)

[8.1. Mô tả 7](#_Toc168699955)

[8.2. Tham số truyền vào 7](#_Toc168699956)

[8.3. Giá trị trả về 7](#_Toc168699957)

[9. Hàm TimerOCSetPwm ( ) 7](#_Toc168699958)

[9.1. Mô tả 7](#_Toc168699959)

[9.2. Tham số truyền vào 7](#_Toc168699960)

[9.3. Giá trị trả về 7](#_Toc168699961)

[10. Hàm ABL\_Process ( ) 7](#_Toc168699962)

[10.1. Mô tả 7](#_Toc168699963)

[10.2. Tham số truyền vào 8](#_Toc168699964)

[10.3. Giá trị trả về 8](#_Toc168699965)

[11. Hàm ABL\_StepBrightness ( ) 8](#_Toc168699966)

[11.1. Mô tả 8](#_Toc168699967)

[11.2. Tham số truyền vào 8](#_Toc168699968)

[11.3. Giá trị trả về 8](#_Toc168699969)

[12. Hàm ABL\_LedControl ( ) 8](#_Toc168699970)

[12.1. Mô tả 8](#_Toc168699971)

[12.2. Tham số truyền vào 8](#_Toc168699972)

[12.3. Giá trị trả về 8](#_Toc168699973)

[13. Luồng xử lý code. 8](#_Toc168699974)

[13.1. Lưu đồ thuật toán tổng quát 8](#_Toc168699975)

**NỘI DUNG THUYẾT MINH**

# Danh sách thư viện

|  |  |  |
| --- | --- | --- |
| **STT** | **Thư viện** | **Mô tả** |
|  | stdio.h | Thư viện hệ thống |
|  | stdint.h | Thư viện lưu trữ và làm việc với các dữ liệu ở cấp độ bit:   * uint8\_t: 8 bit * uint16\_t: 16 bit * uint32\_t: 32 bit |
|  | system\_stm32f4xx.h | Thư viện chứa các hàm khởi tạo hệ thống |
|  | timer.h | Thư viện cho phép quản lý các tác vụ theo thời gian |
|  | stm32f401re\_rcc.h | Thư viện chứa các macro và hàm liên quan tới RCC:   * RCC\_AHB1PeriphClockCmd(): hàm bật / tắt cấp clock cho ngoại vi của bus AHB1 * RCC\_APB1PeriphClockCmd(): hàm bật / tắt cấp clock cho ngoại vi của bus APB1 * ... |
|  | stm32f401re\_gpio.h | Thư viện chứa các macro và hàm liên quan tới GPIO:   * GPIO\_Init(): cấu hình các chân GPIO * ... |
|  | stm32f401re\_adc.h | Thư viện chứa các macro và hàm liên quan tới ADC:   * ADC\_Init(): Cấu hình chức năng cho ADC * ADC\_Cmd(): Cho phép ADC hoạt động * ... |
|  | stm32f401re\_tim.h | Thư viện chứa các macro và hàm liên quan tới Timer:   * TIM\_OC1Init (): Cấu hình chức năng cho Timer với các kênh tương ứng * TIM\_Cmd(): Cho phép Timer hoạt động   ... |
|  | kalman\_filter.h | Thư viện cung cấp các hàm để tạo bộ lọc |

# Danh sách các macro và các biến toàn cục

|  |  |  |
| --- | --- | --- |
| **STT** | **Biến toàn cục và macro** | **Mô tả** |
|  | ADC\_GPIO\_PORT | Địa chỉ cổng GPIO của ADC |
|  | ADC\_GPIO\_PIN | Địa chỉ chân GPIO của ADC |
|  | ADCx\_SENSOR | Địa chỉ của ngoại vi ADC1 |
|  | ADCx\_CLOCK | Clock cấp cho ngoại vi ADC1 |
|  | LED\_GREEN\_PORT | Địa chỉ cổng GPIO của Led GREEN |
|  | LED\_GREEN\_PIN\_1 | Địa chỉ của chân 0 |
|  | LED\_GREEN\_PIN\_2 | Địa chỉ của chân 11 |
|  | LED\_GREEN\_CLOCK | Clock cấp cho ngoại vi GPIOA |
|  | TIM\_PERIOD | Chu kỳ của Timer |
|  | VALUE\_LIGHT\_MAX | Giá trị cường độ ánh sáng tối đa mà cảm biến đo được |
|  | BRIGHTNESS\_STEP | Giá trị để tăng/giảm giá trị ADC |
|  | g\_AdcValue | Biến lưu giá trị ban đầu của ADC |
|  | g\_AdcValueUpdate | Biến lưu giá trị của ADC sau khi cập nhật |
|  | g\_idTimerStepBrightness | Biến lưu giá trị định danh ID của hàm TimerStart khi gọi hàm ABL\_StepBrightness |

# Danh sách các hàm

|  |  |  |
| --- | --- | --- |
| **STT** | **Danh sách hàm** | **Mô tả** |
|  | int main () | Hàm chính của chương trình |
|  | void AppInitManager (void); | Hàm khởi tạo các tài nguyên cần sử dụng |
|  | void LightSensor\_AdcInit (void); | Hàm cấu hình ngoại vi ADC ở chế độ quét liên tục polling |
|  | void LedControl\_TimerOCInit (void); | Hàm cấu hình ngoại vi TIMER ở chế độ Output Compare |
|  | uint16\_t LightSensor\_AdcPollingRead (void); | Hàm lấy giá trị cường độ ánh sáng từ cảm biến ánh sáng |
|  | void TimerOCSetPwm (uint32\_t Compare); | Hàm SET độ rộng xung của Timer |
|  | void ABL\_Process (void); | Hàm cập nhật giá trị cường độ ánh sáng với chu kỳ 100 ms |
|  | void ABL\_StepBrightness (void); | Hàm kiểm tra giá trị ADC thay đổi và cập nhật giá trị PWM |
|  | void ABL\_LedControl (uint8\_t dutyCycle); | Hàm điều khiển độ sáng Led |

# Hàm main ( )

## 4.1. Mô tả

Hàm main ( ) thực hiện việc gọi các hàm sau:

* AppInitManager ( ): Khởi tạo các tài nguyên cần sử dụng
* ABL\_StepBrightness ( ): Kiểm tra giá trị ADC thay đổi và cập nhật giá trị PWM
* Trong hàm while (1) gọi các hàm:
* processTimerScheduler ( ): Xử lý các sự kiện theo thời gian đã cài đặt trước
* ABL\_Process ( ): Cập nhật giá trị cường độ ánh sáng với chu kỳ 100 ms

## 4.2. Tham số truyền vào

Không có

## 4.3. Giá trị trả về

Không có

# Hàm AppInitManager ( )

## 5.1. Mô tả

Hàm AppInitManager ( ) thực hiện việc khởi tạo các tài nguyên cần sử dụng:

* SystemCoreClockUpdate ( ): Cấu hình clock của hệ thống là 84 MHz
* TimerInit ( ): Khởi tạo timer system tick để xử lý các sự kiện thời gian
* LightSensor\_AdcInit ( ): Khởi tạo ngoại vi ADC
* LedControl\_TimerOCInit ( ): Khởi tạo ngoại vi Timer
* KalmanFilterInit ( ): Khởi tạo bộ lọc Kalman

## 5.2. Tham số truyền vào

Không có

## 5.3. Giá trị trả về

Không có

# Hàm LightSensor\_AdcInit ( )

## 6.1. Mô tả

Hàm LightSensor\_AdcInit ( ) thực hiện việc cấu hình ngoại vi ADC ở chế độ quét liên tục polling

## 6.2. Tham số truyền vào

Không có

## 6.3. Giá trị trả về

Không có

# Hàm LedControl\_TimerOCInit ( )

## 7.1. Mô tả

Hàm LedControl\_TimerOCInit ( ) thực hiện việc cấu hình ngoại vi TIMER ở chế độ Output Compare

## 7.2. Tham số truyền vào

Không có

## 7.3. Giá trị trả về

Không có

# Hàm LightSensor\_AdcPollingRead ( )

## 8.1. Mô tả

Hàm LightSensor\_AdcPollingRead ( ) thực hiện việc lấy giá trị cường độ ánh sáng từ cảm biến ánh sáng

## 8.2. Tham số truyền vào

Không có

## 8.3. Giá trị trả về

Biến result (kiểu dữ liệu uint16\_t): Giá trị của ADC sau khi chuyển đổi

# Hàm TimerOCSetPwm ( )

## 9.1. Mô tả

Hàm TimerOCSetPwm ( ) thực hiện việc SET độ rộng xung của Timer

## 9.2. Tham số truyền vào

Biến Compare (kiểu dữ liệu uint32\_t): Độ rộng xung

## 9.3. Giá trị trả về

Không có

# Hàm ABL\_Process ( )

## 10.1. Mô tả

Hàm ABL\_Process ( ) thực hiện việc cập nhật giá trị cường độ ánh sáng với chu kỳ 100 ms

## 10.2. Tham số truyền vào

Không có

## 10.3. Giá trị trả về

Không có

# Hàm ABL\_StepBrightness ( )

## 11.1. Mô tả

Hàm ABL\_StepBrightness ( ) thực hiện việc kiểm tra giá trị ADC thay đổi và cập nhật giá trị PWM

## 11.2. Tham số truyền vào

Không có

## 11.3. Giá trị trả về

Không có

# Hàm ABL\_LedControl ( )

## 12.1. Mô tả

Hàm ABL\_LedControl ( ) thực hiện việc điều khiển độ sáng Led

## 12.2. Tham số truyền vào

Biến dutyCycle (kiểu dữ liệu uint8\_t): chu kỳ hoạt động

## 12.3. Giá trị trả về

Không có

# Luồng xử lý code.

## 13.1. Lưu đồ thuật toán tổng quát

