**TỔNG QUAN CÁC BƯỚC**

**Bước 1:** Khai báo các thư viện; định nghĩa macro để sử dụng **BUTTON**, **LED, BUZZER.**

**Bước 2:** Khởi tạo **System Clock** và các ngoại vi: **TIMER, BUTTON, LED, BUZZER.**

**Bước 3:** Xây dựng **hàm điều khiển** LED, BUZZERvà **đọc trạng thái** BUTTON**.**

**Bước 4:** Xây dựng hàm **nháy LED** khi cấp nguồn.

**Bước 5: Xử lý sự kiện số lần nhấn** nút B3 trong **trình phục vụ ngắt** và **điều khiển** LED, BUZZER.

**Bước 6:** **Xử lý sự kiện giữ nút, nhả, nhấn liên tiếp**  nút B2, B4 trong **trình phục vụ ngắt** và **điều khiển** LED.

**THUYẾT MINH**

**Bước 1:** Khai báo các thư viện; định nghĩa macro để sử dụng **BUTTON**, **LED, BUZZER.**

**1. Khai báo các thư viện**

**+** Cấp xung clock: stm32f401re\_rcc.h

**+** Sử dụng Timer: timer.h

+ Sử dụng GPIO: stm32f401re\_gpio.h

**+** Sử dụng ngắt: misc.h, stm32f401re\_exti.h, stm32f401re\_syscfg.h

**2. Định nghĩa macro**

**Bước 2:** Khởi tạo **System Clock** và các ngoại vi: **TIMER, BUTTON, LED, BUZZER.**

**1. Khởi tạo System Clock:** SystemCoreClockUpdate().

**2. Khởi tạo LED, BUZZER**

/\*

\* @func LedBuzz\_Init

\* @brief To configure all parameters of LED and BUZZER

\* @param None

\* @retval None

\*/

**static** **void** **LedBuzz\_Init**(**void**)

**3. Khởi tạo BUTTON**

/\*

\* @func Button\_Init

\* @brief To configure all parameters of BUTTONs to use interrupt

\* @param None

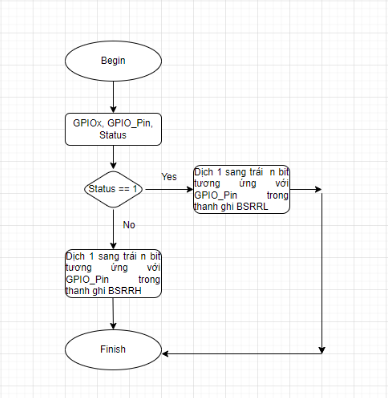
\* @retval None

\*/

**static** **void** **Button\_Init**(**void**)

**Bước 3:** Xây dựng **hàm điều khiển** LED, BUZZERvà **đọc trạng thái** BUTTON

**1. Hàm điều khiển LED**



/\*

\* @func LedControl\_SetState

\* @brief To turn on/off LED following state

\* @param

\* \*GPIOx - Address of GPIO port

\* GPIO\_PIN - The pin ordinal number of GPIO port

\* Status - ON/OFF LED

\* @retval None

\*/

**static** **void** **LedControl\_SetState**(GPIO\_TypeDef \*GPIOx, uint8\_t GPIO\_PIN, uint8\_t Status)

**2. Hàm điều khiển BUZZER (**thuật toán tương tự như đối với LED)

/\*

\* @func BuzzerControl\_SetBeep

\* @brief To blink BUZZER 2 times

\* @param

\* \*GPIOx - Address of GPIO port

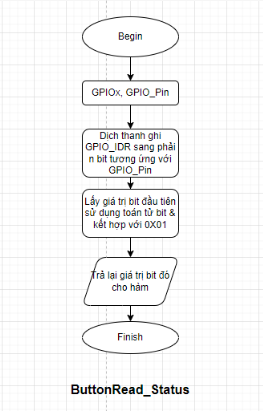
\* GPIO\_PIN - The pin ordinal number of GPIO port

\* @retval None

\*/

**static** **void** **BuzzerControl\_SetBeep**(GPIO\_TypeDef \*GPIOx, uint8\_t GPIO\_PIN)

**3. Hàm đọc trạng thái BUTTON**



/\*

\* @func ButtonRead\_Status

\* @brief To read the button state

\* @param

\* \*GPIOx - Address of GPIO port

\* GPIO\_PIN - The pin ordinal number of GPIO port

\* @retval 0 or 1

\*/

**static** uint8\_t **ButtonRead\_Status**(GPIO\_TypeDef \*GPIOx, uint32\_t GPIO\_PIN)

**Bước 4:** Xây dựng hàm **nháy LED** khi cấp nguồn.

/\*

\* @func Blinkled\_StatusPower

\* @brief To blink LED 4 times when the power turn on

\* @param None

\* @retval None

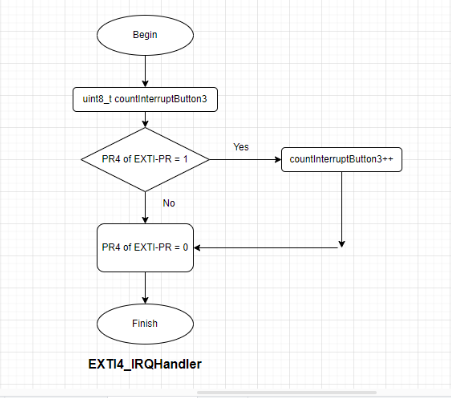
\*/

**static** **void** **Blinkled\_StatusPower**(**void**)

Trong hàm này sử dụng **vòng lặp for** kết hợp với hàm **LedControl\_SetState()** và **hàm Delay()** để tạo nháy LED.

**Bước 5: Xử lý sự kiện số lần nhấn** nút B3 trong **trình phục vụ ngắt** và **điều khiển** LED, BUZZER.

**1. Trình phục vụ ngắt (**đếm sự kiện ấn nút B3)



/\*

\* @func EXTI4\_IRQHandler

\* @brief To process interrupt at EXTI4

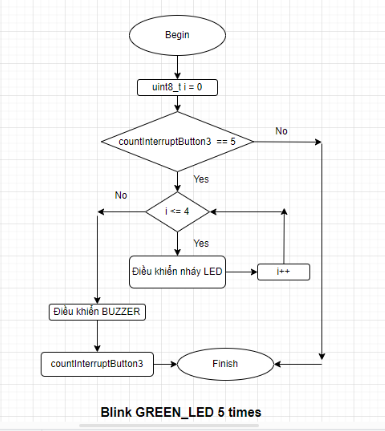
\* @param None

\* @retval None

\*/

**void** **EXTI4\_IRQHandler**(**void**)

**2. Điều khiển LED, BUZZER**



/\*

\* @func LedControl\_SetStatus

\* @brief To blink LED 5 times and BUZZER 2 times

\* @param

\* \*GPIOx1 - Address of LED GPIO port

\* \*GPIOx2 - Address of BUZZER GPIO port

\* GPIO\_PIN1 - The LED pin ordinal number of GPIO port

\* GPIO\_PIN2 - The BUZZER pin ordinal number of GPIO port

\* @retval None

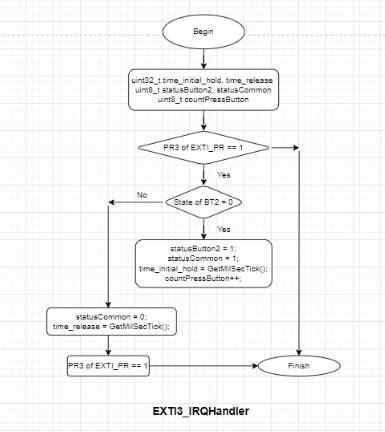
\*/

**static** **void** **LedControl\_SetStatus**(GPIO\_TypeDef \*GPIOx1, GPIO\_TypeDef \*GPIOx2, uint8\_t GPIO\_PIN1, uint8\_t GPIO\_PIN2)

**Bước 6:** **Xử lý sự kiện giữ nút, nhả, nhấn liên tiếp**  nút B2, B4 trong **trình phục vụ ngắt** và **điều khiển** LED.

**1. Trình phục vụ ngắt**

**a) Nút nhấn B2**



/\*

\* @func EXTI3\_IRQHandler

\* @brief To process interrupt at EXTI3

\* @param None

\* @retval None

\*/

**void** **EXTI3\_IRQHandler**(**void**)

**b) Nút nhấn B4 (**thuật toán tương tự như nút nhấn B2)

/\*

\* @func EXTI0\_IRQHandler

\* @brief To process interrupt at EXTI0

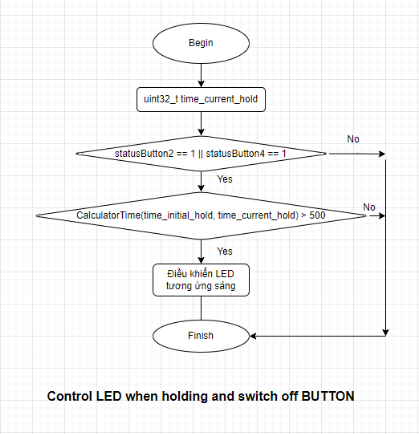
\* @param None

\* @retval None

\*/

**void** **EXTI0\_IRQHandler**(**void**)

**2. Điều khiển LED**



/\*

\* @func LedControl\_TimPress

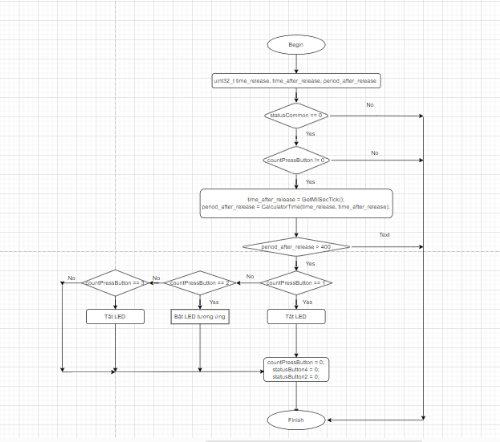
\* @brief To control LED1\_BLUE, LED2\_RED when you press BT2, BT4 and hold them

\* @param None

\* @retval None

\*/

**static** **void** **LedControl\_TimPress**(**void**)



/\*

\* @func LedControl\_TimOutPress

\* @brief To control LED1\_BLUE, LED2\_RED when you press BT2, BT4 continuously

\* @param None

\* @retval None

\*/

**static** **void** **LedControl\_TimOutPress**(**void**)