XT32H05x

XT32 microcontroller GPIO

Application notes

Rev 0.0.0

Original Release Date: 12-Sep-2023

Revised :

Revision History

Release	Date	Author	Summary of Change
V0.0.0	12/09/2023	Shirling Liu	Initial

Contents

List of Figures

List of Tables

1 Introduce

This application note serves as a comprehensive guide for software developers, offering essential information on GPIO configurations for general-purpose input/output pins (GPIOs). It covers fundamental concepts and provides guidelines to ensure proper utilization of GPIOs in software development projects. Whether you're a beginner or an experienced developer, this document will equip you with the necessary knowledge and best practices to effectively configure and utilize GPIOs in your applications.

In this application, two LEDs and two buttons are utilized to demonstrate the configuration of GPIO pins as both input and output, as well as the implementation of interrupt mode for input GPIO pins. The purpose of this application is to showcase how to effectively set up GPIO functionality, including input and output capabilities, while leveraging interrupt mode for responsive input handling.

1.1 Required peripherals

This application involves PADI module and GPIO module.

Table 1. Modules in example

Sub-module	Peripheral use	Note
PADI	4 IO as GPIO	Call HAL_PADI_Init() in code
GPIO	2 pin as input to detect button, 2 pin as output to drive led	Call HAL_GPIO_Init in code

1.2 Compatible devices

This example is compatible with the devices in Table 2.

Table 2. Device list

Product	EVB
XT32H050	XB002823

2 Design description

2.1 Feature overview

Each port can be individually configured by software as GPIO pin in any of the following modes:

- Input with pull-up or pull-down or floating
- Input-interrupt capability for Port A(PAxx) group pins

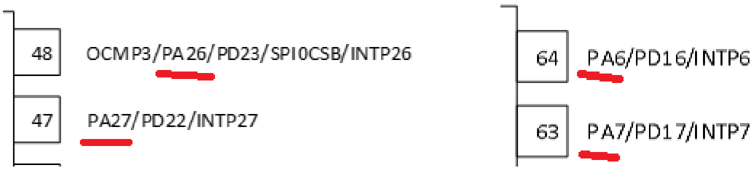
- Output with pull-up or pull-down or floating

2.2 Design steps

1. Select alternate function as GPIO from Peripheral PADI through PADI_InitTypeDef structure. This example uses PA26, PA27 as output to drive the LED, uses PA6 and PA7 as input for button detection.

   ➢ PADI_IDX_IO44_PA26, means select and enable the IO44(pin 48).

   ➢ PADI_CFG_IO44_PA26, means select GPIO Port A bit26 function for IO44.



   Note: please refer to XT32H0xxB—reference manual document to find the assignment relationship between pin 48 with IO44

2. Configure IO in output push-pull mode to drive external LEDs
3. Configure IO in input pull-down mode to detect external button
4. Configure IO in input pull-down mode and enable interrupt mode to detect external button。

➢ GPIO_MODE_IT_PL_SE_DE_SN means enable interrupt and low edge trigger.

- GPIO_MODE_IT_PH_SE_DE_SN means enable interrupt and high edge trigger.

5. Process to detect button1 in polling mode and turn on/off led1, detect button2 in interrupt mode and turn on/off led2

2.3 Design considerations

2.4 Software flowchart

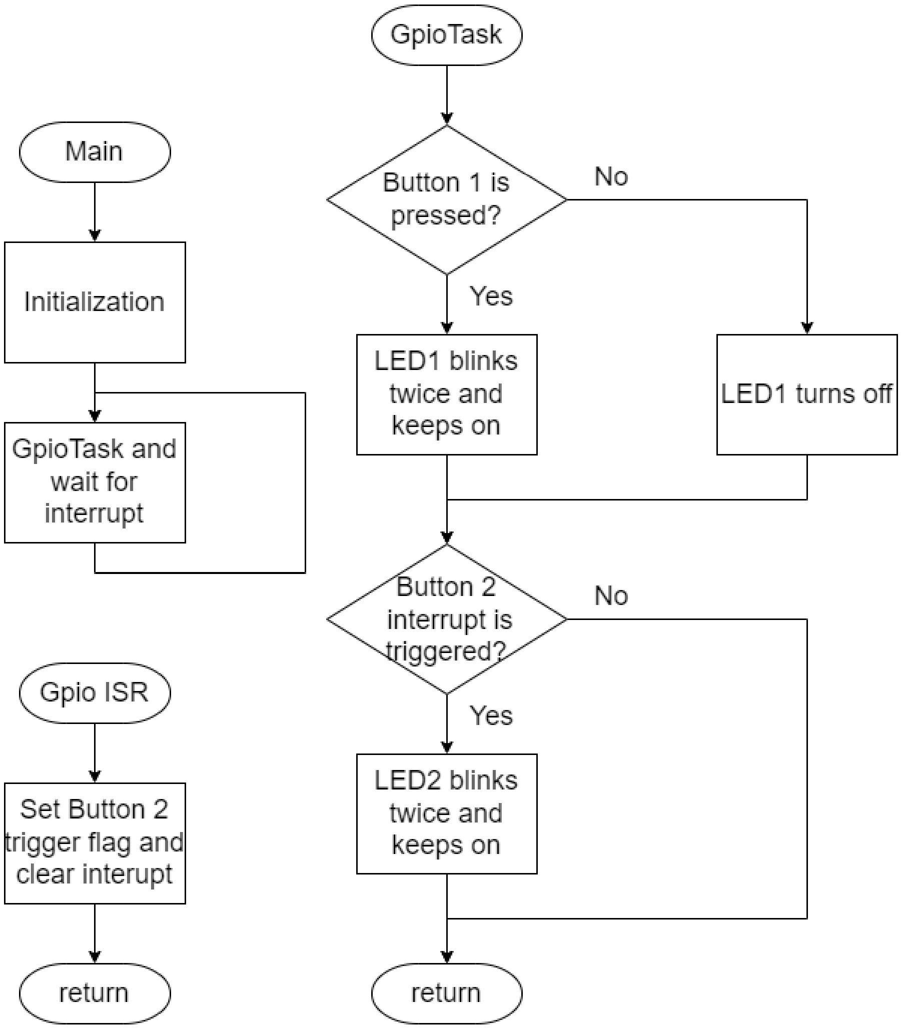

GpioTask
Main
Button 1 is pressed?
No
Initialization
Yes
LED1 blinks twice and keeps on
LED1 turns off
GpioTask and wait for interrupt
Button 2 interrupt is triggered?
No
Gpio ISR
Yes
Set Button 2 trigger flag and clear interupt
LED2 blinks twice and keeps on
return
return

Figure 1. Application flow

2.5 Reference code

```
    /* -1- IO option: Configure pin alternate function as GPIO */
    sPadConfig.Pad        = EVB_GPIO_INPUT_KEY_IO_IDX; //KEY1: SW3101
    sPadConfig.Alternate = EVB_GPIO_INPUT_KEY_IO_CFG;
    sPadConfig.Pull       = PADI_PULLUP;
    sPadConfig.DriveStrength = PADI_DS_HIGH;
    HAL_PADI_Init(PADI, &sPadConfig);
```

```
  /* -2- Configure IO in output push-pull mode to drive external led */
  GPIO_InitStruct.Pin      = EVB_GPIO_LED1_IO_PIN |EVB_GPIO_LED2_IO_PIN;
  GPIO_InitStruct.Mode     = GPIO_MODE_OUTPUT_SW;
  HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
```

```
void XT_Gpio_Task(void)
{
    uint8_t button_sta = 0;
    /*read the Key status*/
    button_sta = Get_Button1_State();
    if(button_sta==PRESSED) {/*Key pressed*/
        /*LED on*/
        Gpio_Led_On(LED_BLUE);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_BLUE);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_BLUE);
        HAL_Delay(200);
    }
    else if(button_sta==RELEASED){
        /*LED on*/
        Gpio_Led_Off(LED_BLUE);
    }

    if(button_trigged_flag)
    {
        button_trigged_flag =0;
        Gpio_Led_On(LED_GREEN);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_GREEN);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_GREEN);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_GREEN);
        HAL_Delay(200);
        Gpio_Led_Toggle(LED_GREEN);
        HAL_Delay(200);
    }
}
```

2.6 Additional resources

- XT32H0xxB--reference manual