Application overview

This application is based on the STM8S-DISCOVERY. It demonstrates how to use the STM8S GPIOs and interrupt controller to drive a set of LEDs.

Once the STM8S105C6T6 is powered up through an USB cable connected to the host PC, LEDs LD2 and LD5 start blinking meaning that the programming operation has completed successfully.

Each time the pushbutton is pressed, the interrupt controller asserts an interrupt that is used to control the I/Os, and change the LED behavior.

Reference documents

- STM8S-DISCOVERY evaluation board user manual (UM0817).
- Developing and debugging your STM8S-DISCOVERY application code user manual (UM0834).
Contents

1 Application description ........................................ 5
  1.1 Hardware requirements .................................. 5
  1.2 Application schematics .................................. 5
  1.3 Application principle .................................. 6

2 Software description ......................................... 7
  2.1 STM8S peripheral configuration ......................... 7
    2.1.1 GPIOs ........................................... 7
    2.1.2 EXTI ............................................. 7
  2.2 Standard STM8S standard firmware library configuration ............ 7
  2.3 Application software flowchart ........................... 7
    2.3.1 Main loop flowchart ............................... 7
    2.3.2 Interrupt function flowchart ....................... 9

3 Revision history ............................................ 10
List of tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>List of external components</td>
<td>5</td>
</tr>
<tr>
<td>Table 2</td>
<td>LEDs configuration</td>
<td>6</td>
</tr>
<tr>
<td>Table 3</td>
<td>Document revision history</td>
<td>10</td>
</tr>
</tbody>
</table>
List of figures

Figure 1. Application schematics ......................................................... 6
Figure 2. Main loop flowchart ................................................................. 8
Figure 3. EXTI_PORTB_IRQHandler() function flowchart ......................... 9
1 Application description

1.1 Hardware requirements

No on-board resources are required.  
*Table 1* gives the list of external components required by the application.

**Table 1. List of external components**

<table>
<thead>
<tr>
<th>External components</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD2, LD3, LD4, LD5</td>
<td>-</td>
<td>Standards LEDs</td>
</tr>
<tr>
<td>R2, R3, R4, R5</td>
<td>510 Ω</td>
<td>Protective resistors</td>
</tr>
<tr>
<td>R1</td>
<td>4.7 kΩ</td>
<td>Pull-up resistor</td>
</tr>
<tr>
<td>R</td>
<td>100 Ω</td>
<td>Debounce filter</td>
</tr>
<tr>
<td>C</td>
<td>100 nF</td>
<td></td>
</tr>
<tr>
<td>Pushbutton</td>
<td>-</td>
<td>Standard pushbutton</td>
</tr>
</tbody>
</table>

1.2 Application schematics

*Figure 1* shows how to interface the LEDs and the pushbutton with the STM8S-DISCOVERY.

For details on STM8S-DISCOVERY implementation, refer to the board schematics provided in the STM8S-DISCOVERY user manual (UM0817).

Protective resistors, R2, R3, R4, and R5, are mandatory to limit the current to a value that does not harm the LEDs.

The pushbutton requires a debounce filter (RC) and a pull-up resistor (R1) to avoid triggering several interrupts due to the mechanical bouncing of the button.
1.3 Application principle

At startup, LD2 and LD5 start blinking meaning that the STM8S105C6T6 Flash memory has been successfully programmed.

Pressing the pushbutton generates an interrupt which is handled by the application software to drive the LEDs.

Only one of the two pairs of LEDs, LD2/LD5 and LD3/LD4 blink at a time. A button event triggers the blinking of the other pair while switching off the first one.

The LEDs blinking conditions are described in Table 2.

Table 2. LEDs configuration

<table>
<thead>
<tr>
<th>Application</th>
<th>LED state</th>
</tr>
</thead>
<tbody>
<tr>
<td>At startup</td>
<td>LD2 and LD5 blink</td>
</tr>
<tr>
<td>On button event</td>
<td>The blinking LED pairs are swapped</td>
</tr>
</tbody>
</table>
2 Software description

The application software uses STM8S standard firmware library to control the general purpose features described in Section 2.1.

2.1 STM8S peripheral configuration

2.1.1 GPIOs

The application drives the MCU I/Os to interface the microcontroller with external hardware components. The GPIO_Init() function configures PB0 as floating input with interrupt to detect pushbutton events, and PB1/PB2/PB3/PB4 as output push-pull to control the LEDs.

2.1.2 EXTI

The external interrupt controller is configured through the EXTI_SetExtIntSensitivity() function to handle the external interrupts on PB0.

The external interrupt sensitivity is configured to trigger an interrupt each time a falling edge, and only a falling edge, is detected on PB0.

2.2 Standard STM8S standard firmware library configuration

The stm8s_conf.h file of the STM8S standard firmware library is used to configure the library by enabling the peripheral functions used by the application.

The following define statements must be present:

```c
#define _GPIO 1 /* enables the GPIOs */
#define _EXTI 1 /* enables the EXTI */
```

2.3 Application software flowchart

2.3.1 Main loop flowchart

The code main loop implements the algorithm that controls the LEDs according to pushbutton events.

The blinking LED pair is selected by setting the ButtonState flag.

Each time the pushbutton is pressed, an interrupt is triggered and ButtonState is complemented (see Section 2.3.2: Interrupt function flowchart). The main loop code tests ButtonState and selects the blinking LED pair according to its value (see Table 2).

The Delay() function generates a delay between the LED ON and OFF states so that we can see them blink.

*Figure 2* shows the flowchart of the application software main loop.
Figure 2. Main loop flowchart

- **Start**
  - **GPIO Configuration**: Configures PB1, PB2, PB3, PB4 as output push-pull low to drives LD2, LD3, LD4 and LD5. Configures PB0 as floating input with interrupts to handle the pushbutton.
  - ** EXTI Configuration**: Configures the external interrupt sensitivity to falling edge only on PB0.
  - **enableInterrupts()**: This function call enables all the STM8S interruptions.

- **ButtonState = 0?**
  - **No**
  - **Select LEDs LD2, LD5**
  - **Switch LEDs ON**
  - **Delay()**
  - **Switch LEDs OFF**
  - **Select LEDs LD3, LD4**
  - The selected LED pair starts blinking until the button is pressed again.
  - **Yes**
2.3.2 Interrupt function flowchart

Each time an interrupt is asserted, the EXTI_PORTB_IRQhandler() function complements the ButtonState flag and the main loop behaves accordingly (see Table 2).

*Figure 3* shows the flowchart of the EXTI(PORTB_IRQhandler() interrupt function.

*Figure 3. EXTI_PORTB_IRQhandler() function flowchart*
3 Revision history

Table 3. Document revision history

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>12-Nov-2010</td>
<td>1</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>