Zephyr RTOS Project: Hello World, LED Blink, and Button Press on SLTB010A Board

Board Used: SLTB010A (EFR32BG22 SoC)

Toolchain and Environment:

- OS: Ubuntu 20.04+
- Python: 3.10.18 (via virtualenv)
- CMake: 3.20.5
- Zephyr SDK: 0.17.0
- Zephyr version: 4.2.0-rc3
- West: 1.4.0
- J-Link Debugger: Version 8.50

---

1. Environment Setup

Create and activate Python virtual environment:

```
python3.10 -m venv ~/.venvs/zephyr-env
source ~/.venvs/zephyr-env/bin/activate
```

Install Zephyr dependencies and west tool:

```
pip install west
west init ~/zephyrproject
cd ~/zephyrproject
west update
west zephyr-export
pip install -r zephyr/scripts/requirements.txt
```

Install Zephyr SDK:

```
wget https://github.com/zephyrproject-rtos/sdk-ng/releases/download/v0.17.0/
zephyr-sdk-0.17.0_linux-x86_64.tar.xz
sudo tar xvf zephyr-sdk-0.17.0_linux-x86_64.tar.xz -C ~/
cd ~/zephyr-sdk-0.17.0
./setup.sh
```

---

2. Flashing and Debug Setup (J-Link)

Install SEGGER J-Link tools:

```
wget https://www.segger.com/downloads/jlink/JLink_Linux_V850_x86_64.deb
sudo dpkg -i JLink_Linux_V850_x86_64.deb
```

Reload udev rules:

```
sudo udevadm control --reload-rules
sudo udevadm trigger
```

Check JLink connection:

```
JLinkExe
```

3. Hello World Application

Create project directory:

```
mkdir ~/zephyrproject/hello_sltb
cd ~/zephyrproject/hello_sltb
```

Add required files:

`CMakeLists.txt`

```
cmake_minimum_required(VERSION 3.20.5)
find_package(Zephyr REQUIRED HINTS $ENV{ZEPHYR_BASE})
project(hello_sltb)
target_sources(app PRIVATE src/main.c)
```

`prj.conf`

```
CONFIG_PRINTK=y
CONFIG_SERIAL=y
```

`src/main.c`

```
#include <zephyr/kernel.h>

void main(void)
{
    printk("Hello SLTB010A!\n");
}
```

Build and Flash:

```
west build -b sltb010a --pristine
west flash
```

View Output:

Sudo minicom -s

press the rst button

---

4. LED Blink on USR0 (PA05)

Code (`src/main.c`):

```
#include <zephyr/kernel.h>
#include <zephyr/device.h>
#include <zephyr/drivers/gpio.h>

#define LED_NODE DT_ALIAS(led0)

static const struct gpio_dt_spec led = GPIO_DT_SPEC_GET(LED_NODE, gpios);

void main(void)
{
    gpio_pin_configure_dt(&led, GPIO_OUTPUT_INACTIVE);
    while (1) {
        gpio_pin_toggle_dt(&led);
        k_sleep(K_SECONDS(1));
    }
}
```

Update `prj.conf`:

```
CONFIG_GPIO=y
```

Build and Flash:

```
west build -b sltb010a --pristine
west flash
```

---

5. Button Press with LED Toggle

Code (`src/main.c`):

```
#include <zephyr/kernel.h>
#include <zephyr/device.h>
#include <zephyr/drivers/gpio.h>

#define LED_NODE    DT_ALIAS(led0)
#define BUTTON_NODE DT_ALIAS(sw0)

static const struct gpio_dt_spec led = GPIO_DT_SPEC_GET(LED_NODE, gpios);
static const struct gpio_dt_spec button = GPIO_DT_SPEC_GET(BUTTON_NODE, gpios);

static struct gpio_callback button_cb_data;

void button_pressed(const struct device *dev, struct gpio_callback *cb, uint32_t
pins)
{
    gpio_pin_toggle_dt(&led);
}
```

```
void main(void)
{
    gpio_pin_configure_dt(&led, GPIO_OUTPUT_INACTIVE);
    gpio_pin_configure_dt(&button, GPIO_INPUT);
    gpio_pin_interrupt_configure_dt(&button, GPIO_INT_EDGE_TO_ACTIVE);

    gpio_init_callback(&button_cb_data, button_pressed, BIT(button.pin));
    gpio_add_callback(button.port, &button_cb_data);

    while (1) {
        k_msleep(100);
    }
}
```

Update `prj.conf`:

```
CONFIG_GPIO=y
CONFIG_GPIO_INIT_PRIORITY=40
```

Result Summary

- Hello world successfully printed using RTT.
- LED blink on PA05 (USR0) tested.
- Button press toggles LED with interrupt.
- All features running on SLTB010A with Zephyr RTOS 4.2.0-rc3.
