

Add SysTick Delay Timer

Global Variables:

```
/****** Start Copy Below *****/
static volatile uint32_t DelayTimerTick = 0;
/****** End Copy *****/
```

Functions:

```
/****** Start Copy Below *****/
/* Initialize SysTick using CMSIS Core_CM4 function */
static void InitSysTick(void)
{
    SysTick_Config(120000); /* Configure the SysTick timer */
}

/* SysTick Interrupt Handler */
void SysTick_Handler(void)
{
    if (DelayTimerTick<0xFFFFFFFF) { DelayTimerTick++; } /* increment timer */
}

/* Delay Function based on SysTick Counter */
void Delay_SysTick(uint32_t SysTicks)
{
    DelayTimerTick = 0; /* reset timer value */
    while(DelayTimerTick < SysTicks); /* wait for timer */
}
/****** End Copy *****/
```

Main() Additions

```
/****** Start Copy Below *****/
    InitSysTick();
/****** End Copy *****/
```

Blinky LED

SDK Driver Include

```
/* Start Copy Below */
#include "fsl_gpio.h"
/* End Copy */
```

Peripheral Initialization

```
/* Start Copy Below */
void BOARD_InitPeripheral(void)
{
    /* GPIO Peripheral Init - Output Pin for LED */
    gpio_pin_config_t led_config = {
        kGPIO_DigitalOutput, 0,
    };
    GPIO_PinInit(BOARD_INITPINS_MY_LED_GPIO, BOARD_INITPINS_MY_LED_GPIO_PIN,
&led_config);
}
/* End Copy */
```

Main() Additions

```
/* Start Copy Below */
    BOARD_InitPeripheral();
/* End Copy */

/* Start Copy Below */
while (1)
{
    /* Blink LED every 250 SysTicks */
    Delay_SysTick(250);
    GPIO_TogglePinsOutput(BOARD_INITPINS_MY_LED_GPIO, 1u <<
BOARD_INITPINS_MY_LED_GPIO_PIN);
}
/* End Copy */
```

Interrupt Push Button

Peripheral Initialization (Add to existing function)

```
/****** Start Copy Below *****/
/* GPIO Peripheral Init - Input Pin for Push Button */
PORT_SetPinInterruptConfig(BOARD_INITPINS_SW2_PORT, BOARD_INITPINS_SW2_GPIO_PIN,
kPORT_InterruptFallingEdge);
EnableIRQ(PORTC_IRQn);

gpio_pin_config_t sw_config = {
    kGPIO_DigitalInput, 0,
};
GPIO_PinInit(BOARD_INITPINS_SW2_GPIO, BOARD_INITPINS_SW2_GPIO_PIN, &sw_config);
/****** End Copy *****/
```

SDK Driver Include

```
/****** Start Copy Below *****/
#include "fsl_port.h"
/****** End Copy *****/
```

Global Variables:

```
/****** Start Copy Below *****/
static volatile uint32_t g_count = 0;
/****** End Copy *****/
```

Functions:

```
/****** Start Copy Below *****/
/* Interrupt Handler for GPIO Port C Pins */
void PORTC_IRQHandler(void)
{
    /* Clear external interrupt flag. */
    GPIO_ClearPinsInterruptFlags(BOARD_INITPINS_SW2_GPIO, 1U <<
BOARD_INITPINS_SW2_GPIO_PIN);
    g_count++; /* Increment counter */
}
/****** End Copy *****/
```