

COE718 - Lab 2

Gabriel Casciano, 500744076
gabriel.casciano@ryerson.ca

Sept, 27, 2019

The code snippet, **Listing. 1**, shows how we can use the LED header file, to performs bit masking to enable the LEDS. Before the main runtime begins the LED init function must be called to ensure all pins are configured properly. The LED header masks the LPC_GPIOx->FIOPIN register, typically we mask with `—=` and `=` depending whether or not we are setting bits HIGH or LOW.

The code snippet, **Listing. 2**, implements an example of using the bit banding function. The address returned by the BitBanding function must be assigned to our pointer. Once it is assigned we can simply assign a 1 or 0 to turn it on or off.

$$\begin{aligned} \text{ByteOffset} &= 0x20000000 - \text{MemoryMappedAddress} \\ \text{BitBandAddress} &= \text{StartPoint} + (\text{ByteOffset} * 32) + (\text{Bit} * 4) \end{aligned}$$

The way the math is implemented in the program is different because it uses logical or, and left shift to allow be for quicker computation.

Using the Bit banding address function and hand calculating the addresses and assigning them to variables is functionally the same thing as using the bit banding function and can be observed in **Listing. 3**.

Calculations

I have calculated the Bit band addresses for both P1.31 and P2.2,

GPIO1_PIN Memory Addr= 0x2009C034

Bit = 31

StartPoint = 0x22000000

ByteOffset = 0x20000000 - MemoryMappedAddress = x9C034

```

Bit BandAddress = StartPoint + (ByteOffset * 32) + (Bit * 4) =
0x233806FC

```

```

GPIO2_PIN Memory Addr = 0x2009C054
Bit = 2
StartPoint = 0x22000000
ByteOffset = 0x20000000 - MemoryMappedAddress = x9C054
Bit BandAddress = StartPoint + (ByteOffset * 32) + (Bit * 4) =
0x23380A88

```

Listing 1: Masking Mode

```

#include "LED.h"

int main(void){
    LED_Init();
    LED_On(3); //on
    LED_Off(3); //off
}

```

Listing 2: Function Mode

```

#define ADDRESS(x) (*((volatile unsigned long *)(x)))
#define BitBand(x, y) ADDRESS((((unsigned long)(x) & 0xF0000000) |
    0x02000000 | (((unsigned long)(x) & 0x000FFFFF) << 5) | ((y) << 2)))

volatile unsigned long * GPIO1_LED28;

int main(void){
    GPIO1_LED28 = &BitBand(&LPC_GPIO1->FIOPIN1, 28);
    *GPIO1_LED28 = 1; //on
    *GPIO1_LED28 = 0; //off
}

```

Listing 3: Bit Banding Mode

```

#define GPIO1_LED31 (*((volatile unsigned long *) 0x233806FC))

int main(void){
    GPIO1_LED31 = 1; //on
    GPIO1_LED31 = 0; //off
}

```