Project #01(Pjt01_led)

- Memory Map of Atmega2560

Memory Mapped I/O, Isolated I/O

I/O Macros(Header Files)

GPIO PIN interface(LED Control)

Project #01(Pjt01_led)

Memory Map of Atmega2560

Figure 11. Program Memory Map

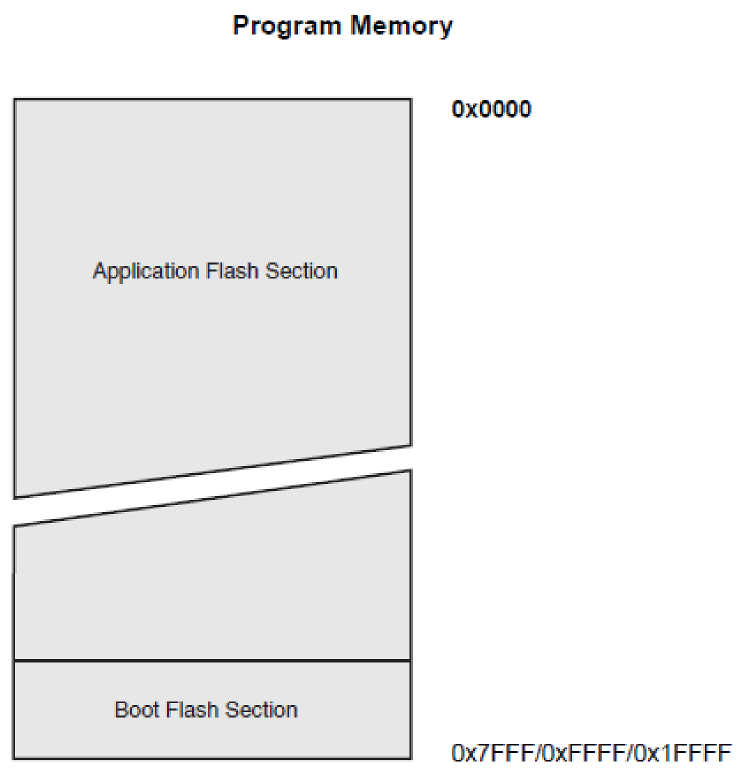
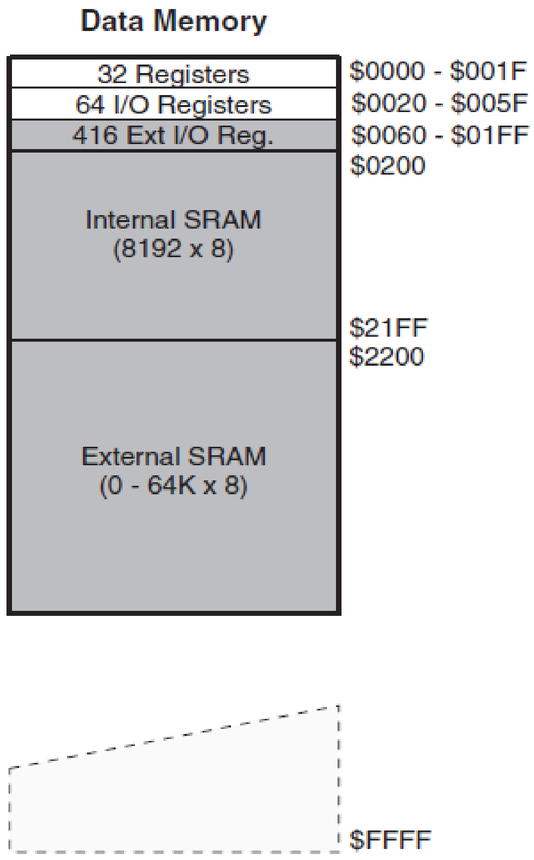


Figure 12. Data Memory Map



Project #01(Pjt01_led)

- Memory Mapped I/O, Isolated I/O

PORTL ?

★ /usr/local/avr/include/sfr_defs.h

```
#define _SFR_MEM8(addr)    _MMIO_BYTE(addr)
#define _MMIO_BYTE(addr)  (*(volatile uint8_t *)(addr))
```

★ /usr/local/avr/include/iomxx0_1.h

```
#define  PORTL  _SFR_MEM8(0x10B)
```

PORTL = 0xff; ➔ (*(volatile uint8_t *)(addr)) = 0xff;

Project #01(Pjt01_led) step 1

```
#include <avr/io.h>
int  main()
{
     DDRL = 0xff;
     PORTL = 0x00;
     while(1)
          ;
     return 0;
}
```

main.c

Project #01(Pjt01_led) step 2

```c
void  led();

int  main()
{
     led();
     while(1)
          ;
     return 0;
}
```

main.c

```c
#include <avr/io.h>

void  led()
{
     DDRL = 0xff;
     PORTL = 0x00;
}
```

led.c

Project #01(Pjt01_led) step 3

```
#include  “led.h”

int  main()
{
     led();
     while(1)
          ;
     return 0;
}
```

main.c

```
void   led();
```

led.h

```
#include <avr/io.h>

void  led()
{
     DDRL = 0xff;
     PORTL = 0x00;
}
```

led.c

Project #01(Pjt01_led) step 4

```
#include <avr/io.h>
#include <util/delay.h>
#include “led.h”
int main()
{
    led_init();
    while(1) {
        led_on_all();
        _delay_ms(1500);
        led_off_all();
        _delay_ms(1500);
    }
    return 0;
}
```

main.c

```
void led_init(void);

void led_on_all(void);

void led_off_all(void);

void led_on(uint8_t led_no);

void led_off(uint8_t led_no);

void led_set(uint8_t led_mask);
```

led.h

```
#include <avr/io.h>
#include “led.h”
static uint8_t  Leds;
void led_init() {
    DDRL = PORTL = 0xff;
}
void led_on_all(void) {
    Leds = PORTL = 0x00;
}
void led_off_all(void) {
    Leds = PORTL = 0xff;
}
void led_on(uint8_t led_no) {
    Leds &= (1 << led_no);
    PORTL = Leds;
}
void led_off(uint8_t led_no) {
    Leds ¦= (1 << led_no);
    PORTL = Leds;
}
void led_set(uint8_t led_mask) {
    Leds = ~led_mask;
    PORTL = Leds;
}
```

led.c