Laboratory session 1: Programming ATMega328P in C Language

Sadalinda B.G.S.,

2019/E/118,

Group D

Blinking LED

```
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-gcc -Os -DF_CPU=16000000UL -mmcu=atmega328p -o led led.c
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-objcopy -O ihex -R .eeprom led led.hex
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avrdude -F -V -c arduino -p atmega328p -P /dev/ttyACM0 -b 115200 -U flash:w:led.hex

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.00s

avrdude: Device signature = 0x1e950f (probably m328p)
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "led.hex"
avrdude: input file led.hex auto detected as Intel Hex
avrdude: writing flash (176 bytes):

Writing | ################################################## | 100% 0.04s

avrdude: 176 bytes of flash written

avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done.  Thank you.

eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avrdude -F -V -c arduino -p atmega328p -P /dev/ttyACM0 -b 115200 -U flash:w:led.hex

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.00s

avrdude: Device signature = 0x1e950f (probably m328p)
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "led.hex"
avrdude: input file led.hex auto detected as Intel Hex
avrdude: writing flash (176 bytes):

Writing | ################################################## | 100% 0.04s

avrdude: 176 bytes of flash written

avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done.  Thank you.
```

Exercise 1

1.

The avrdude function, the "-v" or "--verbose" option enables verbose output. When this option is used, avrdude provides more detailed information during its execution, which can be helpful for troubleshooting and understanding the process. verbose output in avrdude:

- Device Information
- Configuration Details
- Communication Messages
- Memory Operations
- Error Messages
- Timing and Statistics

Exercise 2.1

```
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-gcc -Os -DF_CPU=16000000UL -mmcu=atmega328p -o ex02_1 ex02_1.c
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-objcopy -O ihex -R .eeprom ex02_1 ex02_1.hex
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avrdude -F -V -c arduino -p atmega328p -P /dev/ttyACM0 -b 115200 -U flash:w:ex02_1.hex

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.00s

avrdude: Device signature = 0x1e950f (probably m328p)
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "ex02_1.hex"
avrdude: input file ex02_1.hex auto detected as Intel Hex
avrdude: writing flash (176 bytes):

Writing | ################################################## | 100% 0.04s

avrdude: 176 bytes of flash written

avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done.  Thank you.

eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$
```

Exercise 2.2

```
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-gcc -Os -DF_CPU=16000000UL -mmcu=atmega328p -o ex02_2 ex02_2.c
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avr-objcopy -O ihex -R .eeprom ex02_2 ex02_2.hex
eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$ avrdude -F -V -c arduino -p atmega328p -P /dev/ttyACM0 -b 115200 -U flash:w:ex02_2.hex

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.00s

avrdude: Device signature = 0x1e950f (probably m328p)
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "ex02_2.hex"
avrdude: input file ex02_2.hex auto detected as Intel Hex
avrdude: writing flash (254 bytes):

Writing | ################################################## | 100% 0.04s

avrdude: 254 bytes of flash written

avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done.  Thank you.

eng-co-mpl-001@engcompl001-HP-ProDesk-400-G4-MT:~/Desktop/Lab01_EC6020$
```

Exercise 3

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

void my_delay(int time) {
 for (int i = 0; i < time; i++) {
  for (volatile int j = 0; j < 3000; j++) {
  }
 }
}

int main(void) {
 my_delay(1000);
//the terminal function
 char avrdude_cmd[] = "avrdude -c <avrdude_options> -p <avr_model> -P <port> -U flash:w:<hex_file>";
 system(avrdude_cmd);

 return 0;
}
```

Exercise 4:

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

#define F_CPU 16000000UL
#define TIMER0_PRESCALER 64
#define TIMER0_COMPARE_VALUE (F_CPU / (TIMER0_PRESCALER * 500))

void timer0_init() {
  TCCR0A = 0;
  TCCR0B = (1 << CS01) | (1 << CS00);
  OCR0A = TIMER0_COMPARE_VALUE;
  TIMSK0 = (1 << OCIE0A);
}

ISR(TIMER0_COMPA_vect) {
  PORTB ^= (1 << PB5);
}

int main(void) {
  DDRB |= (1 << PB5);
  sei();
  timer0_init();
```

```
  while (1) {
    // Your main code goes here
  }
}
```