1.1 LED

1.1.1

LED chosen = 160-1128-ND 5mm red LED, LTL-4224

LITEON LITE-ON ELECTRONICS, INC.

Property of Lite-On Only

Electrical / Optical Characteristics at T_A=25℃

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Condition
Luminous Intensity	I_V	29	90		mcd	I_F = 10mA Note 1,4
Viewing Angle	$2\theta_{1/2}$		16		deg	Note 2 (Fig.6)
Peak Emission Wavelength	λ_P		635		nm	Measurement @Peak (Fig.1)
Dominant Wavelength	λ_d		623		nm	Note 3
Spectral Line Half-Width	$\Delta\lambda$		40		nm	
Forward Voltage	V_F		2.0	2.6	V	I_F = 20mA
Reverse Current	I_R			100	μA	V_R = 5V
Capacitance	C		20		pF	V_F = 0 , f = 1MHz

We can see in the data sheet that the typical value for forward voltage is 2.0V and that for forward current is 20mA.

Let us now use this to calculate the minimum value of the resistor we can use.

Input Voltage (V_{in}) = 5.0 V

Forward Voltage (V_f) = 2.0 V

Forward Current (I_f) = 20 mA = $\frac{20}{1000}$ A

[from data sheet]

∴ Voltage drop across Resistor R_1 should be (V_{drop}) = $V_{in} - V_f$ — (1)

= 5.0 V - 2.0 V

= 3.0 V

Also, $V_{drop} = I_f \times R_1$ — (2)

∴ Using eqn (1) and eqn (2)

$\Rightarrow I_f \times R_1 = V_{in} - V_f$

$\Rightarrow R_1 = \frac{V_{in} - V_f}{I_f} = \frac{3.0}{20/1000}$

$\Rightarrow R_1 = 150\ \Omega$

∴ Smallest Resistor we can use is 150 Ω

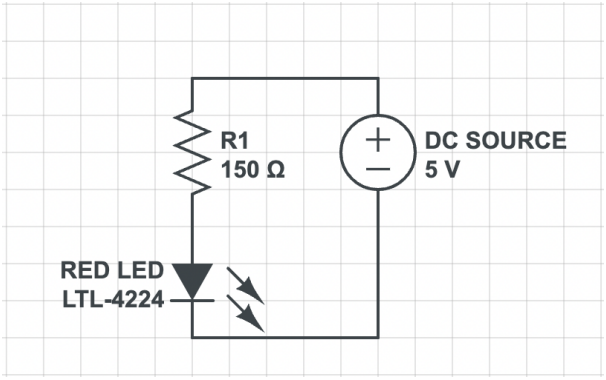
1.1.2

Theoretical value of R1 = 150 Ω
Observed value of R1 = 145.7 Ω

LED did not get too hot, start smoking or explode.

Finding value of resistance that gives very dim but still visible light -

Theoretical value of R_dim = 47000 Ω
Observed value of R_dim = 46490 Ω



Calculating current that gives dim light

$$V_{in} = V_f + I_{dim} R_{dim}$$

$$5V = 2V + (I_{dim} \times 46490\,\Omega)$$

$$I_{dim} = \frac{3V}{46490\,\Omega} = 0.0000645\ A$$

$$I_{dim} = 0.0645\ mA$$

1.1.3

LED 1 - LTW-2S3D7 (WHITE CLEAR)
LED 2 - LTL-4253 (YELLOW)

LED 1 datasheet

LITEON® **LITE-ON TECHNOLOGY CORPORATION**

Property of Lite-On Only

Electrical / Optical Characteristics at Ta=25℃

PARAMETER	SYMBOL	MIN.	TYP.	MAX.	UNIT	TEST CONDITION
Luminous Intensity	Iv	10000	16000		mcd	IF = 20mA Note 1,2,3 Iv Spec. Table
Viewing Angle	$2\theta_{1/2}$		15		deg	Note 4
Chromaticity Coordinates	x		0.30			IF = 20mA Note 5 Hue Spec. Table & Chromaticity Diagram
	y		0.30			
Forward Voltage	VF		3.3	3.6	V	IF = 20mA
Reverse Current	IR			100	μA	VR = 5V

We can see in the data sheet that for LED 1 the typical value for forward voltage is 3.3V and that for forward current is 20mA.

LED 2 datasheet

LITEON LITE-ON ELECTRONICS, INC.

Property of Lite-On Only LTL-4253

Electrical / Optical Characteristics at TA=25℃

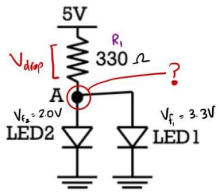
Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Condition
Luminous Intensity	I_V	5.6	19		mcd	I_F = 10mA Note 1,4
Viewing Angle	$2\theta_{1/2}$		36		deg	Note 2 (Fig.6)
Peak Emission Wavelength	λ_P		585		nm	Measurement @Peak (Fig.1)
Dominant Wavelength	λ_d		588		nm	Note 3
Spectral Line Half-Width	$\Delta\lambda$		35		nm	
Forward Voltage	V_F		2.1	2.6	V	I_F = 20mA
Reverse Current	I_R			100	μA	V_R = 5V
Capacitance	C		15		pF	V_F = 0 , f = 1MHz

We can see in the data sheet that for LED 2 the typical value for forward voltage is 2.1V and that for forward current is 20mA.

To find voltage at point A:

$V_{in} = 5\,V$

$I = 0.008\,A$ [value observed on Power supply]



∴ Theoretical voltage at point A (V_A) =

$$V_A = V_{in} - V_{drop}$$
$$= V_{in} - IR_1$$
$$= 5 - (0.008 \times 330)$$
$$= 2.34\,V$$

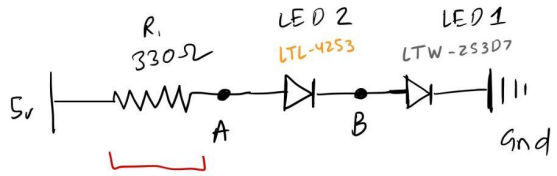
∴ Actual voltage measured at point A using Digital multimeter = 2.1 V

We know forward voltage for LED 1 (3.3V) is higher than that of LED 2 (2.1V) and that the voltage at point A is 2.1 V. Hence we can safely guess that LED 2 will be brighter as the voltage that is being received by the LEDs is closer to LED 2's forward voltage. This is due to the "limiting case" when diodes are connected in parallel.

We observe that LED 2 is in fact brighter.

The forward voltage of LED 2 in the datasheet explains the voltage at point A.

1.1.4



$V_{in} = 10\ V$

$V_{LED_2} = 2.1V$] datasheet

$V_{LED_1} = 3.3\ V$

Theoretically,

$V_{drop} = V_{in} - V_{LED_2} - V_{LED\,2}$

$= 10 - 2.1 - 3.3$

$= 4.6V$

$\therefore V_A = V_{in} - V_{drop} = 5.4\ V$

Observed

$V_A = 5.245V$

Theoretically,

$\therefore V_B = V_{in} - V_{drop} - V_{LED_2}$

$= 10 - 4.6 - 2.1$

$= 3.3\ V$

Observed,

$V_B = 2.96\ V$

1.1.5

If I were designing an LED desk lamp I would want

- a warm white LED with a color temp around 2700K - I find them conducive to focused work
- a peak brightness around 800 lm - I would be able to avoid straining my eyes while reading tiny text or working with small electronic components.

All these factors made me choose MHDEWT-0000-000N0HG227G.
This is a viable choice for the light source of our desk lamp as -

- It has a luminous efficacy of 113 lm/W, which is one of the highest for LEDs that output ≈800 Lumen
- It is sold as single units and I do not have to buy in bulk. It costs $2.68 to buy 1 unit of this LED, which is a competitive price.

1.2 ItsyBitsy Introduction

1.2.1

We are changing the Data Direction Register to determine the direction of data flow for the specific GPIO (General Purpose Input/Output) pin to which we have connected our LED.

1.2.2

LED chosen = LTL-4253

LITEON LITE-ON ELECTRONICS, INC.

Property of Lite-On Only

LTL-4253

Electrical / Optical Characteristics at T_A=25℃

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Condition
Luminous Intensity	I_V	5.6	19		mcd	I_F = 10mA Note 1,4
Viewing Angle	$2\theta_{1/2}$		36		deg	Note 2 (Fig.6)
Peak Emission Wavelength	λ_P		585		nm	Measurement @Peak (Fig.1)
Dominant Wavelength	λ_d		588		nm	Note 3
Spectral Line Half-Width	$\Delta\lambda$		35		nm	
Forward Voltage	V_F		2.1	2.6	V	I_F = 20mA
Reverse Current	I_R			100	μA	V_R = 5V
Capacitance	C		15		pF	V_F = 0 , f = 1MHz

Forward Voltage = 2.1V

Forward current = 20mA

1.2.2 Choosing a resistor

Input Voltage (V_{in}) = 5.0 V

Forward Voltage (V_f) = 2.1 V [from data sheet]

Forward Current (I_f) = 20 mA = $\frac{20}{1000}$ A

∴ Voltage drop across Resistor R_1 should be (V_{drop}) = $V_{in} - V_f$ — (1)

= 5.0V - 2.1V

= 2.9V

Also, $V_{drop} = I_f \times R_1$ — (2)

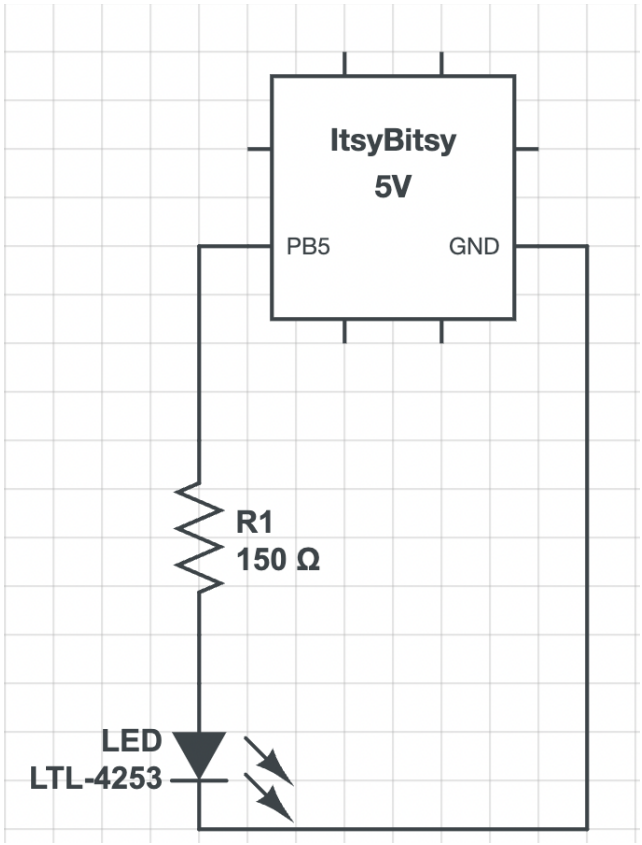
∴ Using eqn (1) and eqn (2)

$\Rightarrow I_f \times R_1 = V_{in} - V_f$

$\Rightarrow R_1 = \frac{V_{in} - V_f}{I_f} = \frac{2.9}{20/1000}$

$\Rightarrow R_1 = 145\ \Omega$

∴ Lets use a 150Ω Resistor



1.2.3

Users > virgoyal > Desktop > Blinky1.2.3 > src > C main.c > ...

```c
/* Name: main.c
 * Author: Vir Goyal
 */

#include "MEAM_general.h"  // includes the resources included in the MEAM_general.h file

int main(void)
{
    _clockdivide(0); //set the clock speed to 16Mhz
    DDRB |= 0b00100000; // Data Direction Register for port B and setting it to PB5
    for(;;){ // infinite loop

        PORTB |= 0b00100000;// Turn on the LED
        _delay_ms(1000); // Wait for 1 second


        PORTB &= ~0b00100000; // Turn off the LED
        _delay_ms(1000); // Wait for 1 second
    }
    return 0;   /* never reached */
}

```

1.2.4

Users > virgoyal > Desktop > Blinky1.2.4 > src > C main.c > main(void)

```c
/* Name: main.c
 * Author: Vir Goyal
 */

#include "MEAM_general.h"  // includes the resources included in the MEAM_general.h file

int main(void)
{
    _clockdivide(0); //set the clock speed to 16Mhz
    set(DDRB,5); // Data Direction Register for port B and setting it to PB5.
    int dutycycle = 75 ;// dutycycle variable decides how long the LED will be in each (ON/OFF) state.
   // >>Assuming the total duration of a complete cycle is 1 seconds<<
   for(;;){ // infinite loop
       for (int i = 1; i <= dutycycle; i++){// loop that switches ON the LED for a duration set by dutycycle
           PORTB |= 0b00100000;// Turn on the LED
           _delay_ms(10); // Wait for 0.01 second
    }
       for (int i = 1 ; i <= 100-dutycycle; i++){// loop that switches OFF the LED for a duration set by dutycycle
           clear(PORTB,5);// Turn off the LED
           _delay_ms(10); // Wait for 0.01 second
   }
   }
   return 0;   /* never reached */
}

```

1.3 Timers

1.3.1


```
Users > virgoyal > Desktop > Blinky1.3.1 > src > C main.c > main()
1   /* Name: main.c
2    * Author: Vir Goyal
3    */
4   #include "MEAM_general.h"
5   #define COMPAREVALUE 6250 // calculated from the fact that clock source frrequency is 250kHz
6                             //and we want LED to blink at 20Hz
7   int main() {
8     DDRC |= 0x40;       //Port C6 as output
9     clear(TCCR3B,CS31);// Turn on counter 3 (1/64 prescaler)
10    set (TCCR3B,CS30);
11    set (TCCR3B,CS32);
12
13      for (;;) {//infinite loop
14        if (TCNT3 > COMPAREVALUE) {
15          toggle(PORTC,6);//switch state of LED
16          TCNT3 = 0; // Reset the timer to 0
17        }
18      }
19  return 0; //never reached
20  }
21
```

Prescaler = 1/64

∴ clock source freq = 250 kHz

if we want 20 Hz blinking

No. of cycles = $\frac{250,000}{20}$ = 12,500

∴ Compare value = $\frac{12,500}{2}$ = 6250

↳ ∴ each cycle needs an on and off state

1.3.2

I changed the prescaler to 1/256 and let the COMPAREVALUE remain 6250.

Prescaler = 1/256

∴ Clock Source freq = 62.5 kHz

∴ Blinking freq (x) = ?

$\Rightarrow \frac{62500}{x} = 6250 \times 2$

$\Rightarrow \frac{62500}{6250 \times 2} = x$

$\Rightarrow x = 5$

Blinking freq = 5 Hz

The output did change as we had expected it to.
The default system clock frequency is 16Mhz.

1.3.3

I am using a prescaler of 1/1024 and Timer mode 7 for my waveform generation.

Users > virgoyal > Desktop > Blinky1.3.3 > src > C main.c > main(void)

```c
/* Name: main.c
 * Author: Vir Goyal
 */
#include "MEAM_general.h"

int main(void) {
  set(DDRC,6); // Port C6

  set(TCCR3B,CS32);// Turn on counter 3 (1/1024 prescaler)
  clear (TCCR3B,CS31);
  set (TCCR3B,CS30);

  double dutycycle = 0 ;// dutycycle variable decides how long the LED will be in each (ON/OFF) state.
  double totcycle=1023;


  set(TCCR3B,WGM32);  // timer  mode 7
  set(TCCR3A,WGM30);
  clear(TCCR3B,WGM33);
  set(TCCR3A,WGM31);

  set (TCCR3A,COM3A1); //clear at OCR3A, set at rollover,
  clear (TCCR3A,COM3A0);

  OCR3A = totcycle*(dutycycle/100); // giving OCR3A value to compare to

  while(1){// run infinitely
    return 0;
  }

}
```

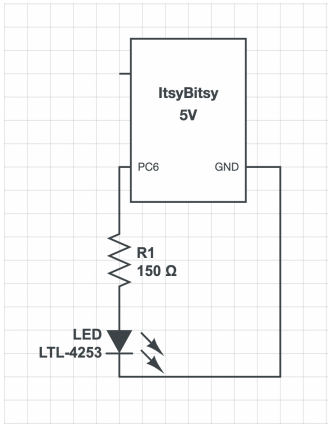
[DEMO VIDEO (50% duty cycle)]
[DEMO VIDEO (0% Duty Cycle)]
[DEMO VIDEO (100% Duty cycle)]

1.4 Practice with Loops

1.4.1

Users > virgoyal > Desktop > Blinky1.4.1 > src > C main.c > main(void)

```c
/* Name: main.c
 * Author: Vir Goyal
 */
#include "MEAM_general.h"  // includes the resources included in the MEAM_general.h file
int main(void){
  _clockdivide(0);
  set(DDRC,6); // setting Port C6 as output
  set (TCCR3B,CS30); //Setting prescaler to 1/1

  double dutycycle =0 ;// dutycycle variable decides how long the LED will be in each (ON/OFF) state.
  double duration_inc =300; // defines how long the intensity of brightness will increase
  double duration_dec =600;// defines how long the intensity of brightness will decrease
  double ds =100; // this is the step size, defines how smooth the transition betweeen 0 intensity to 100 intensity will be
  double totcycle=1023;

  // set timer on OCR3A mode 7
  set(TCCR3B,WGM32);
  set(TCCR3A,WGM30);
  clear(TCCR3B,WGM33);
  set(TCCR3A,WGM31);

  //clear at OCR3A, set at rollover
  set (TCCR3A,COM3A1);
  clear (TCCR3A,COM3A0);

  for(;;){//infinite loop
    for (int on = 0; on <= ds; on++){
    //if(dutycycle==100){set_led( TOGGLE);} // used to debug
    OCR3A = totcycle*(dutycycle/100);
    dutycycle=dutycycle+(100/ds);
    _delay_ms(duration_inc/ds);
    }
    for (int off = 0; off <= ds; off++){
    //if(dutycycle==0){set_led( TOGGLE);} // used to debug
    OCR3A = totcycle*(dutycycle/100);
    dutycycle=dutycycle-(100/ds);
    _delay_ms(duration_dec/ds);


    }
  }

}

```



[DEMO VIDEO]

1.4.2

Users > virgoyal > Desktop > Blinky1.4.2 > src > C main.c > totcycle

```c
/* Name: main.c
 * Author: Vir Goyal
 */
#include "MEAM_general.h"  // includes the resources included in the MEAM_general.h file

//GLOBAL VARIABLES
double dutycycle =0 ;// dutycycle variable decides how long the LED will be in each (ON/OFF) state.
                     //Also indirectly defines intensity.
double totcycle=1023;

void intensity_inc (double max_intensity){
  for (int on = 0; on <= 100; on++){
    /*if(dutycycle==100){set_led(OFF);}
    if(dutycycle==50){set_led(OFF);}*/
    OCR3A = totcycle*(dutycycle/100); //using OCR3A and waiting for timer to reach this value
    dutycycle=dutycycle+(max_intensity/100); // updating and increasing dutycycle to increase intensity
                                             //This is divided by 100 as loop will run 100 times
    _delay_ms(1); // waiting for 1 ms (this will run a 100 times thus total time = 0.1s)
  }

}

void intensity_dec (double max_intensity){
  for (int off = 0; off <= 100; off++){
  //if(dutycycle==(max_intensity/100)){set_led(ON);} // USED FOR DEBUGGING
  OCR3A = totcycle*(dutycycle/100); //using OCR3A and waiting for timer to reach this value
  dutycycle=dutycycle-(max_intensity/100); // updating and decreasing dutycycle to decrease intensity.
                                           //This is divided by 100 as loop will run 100 times
  _delay_ms(4); // waiting for 4 ms (this will run a 100 times thus total time = 0.4s)
  }
}

int main(void)
{
  _clockdivide(0);
  set(DDRC,6); // setting Port C6 as output
  set (TCCR3B,CS30); //Setting prescaler to 1/1

  // set timer on OCR3A mode 7
  set(TCCR3B,WGM32);
  set(TCCR3A,WGM30);
  clear(TCCR3B,WGM33);
  set(TCCR3A,WGM31);


  //clear at OCR3A, set at rollover
  set (TCCR3A,COM3A1);
  clear (TCCR3A,COM3A0);

  for(;;){
    intensity_inc(100); // increase intensity to 100% in 0.1s
    intensity_dec(100); // decrease intensity to 0% in 0.4s
    intensity_inc(50); // increase intensity to 50% in 0.1s
    intensity_dec(50); // decrease intensity to 0% in 0.4s
    clear(PORTC,6);   // just to  make sure LED is OFF
    _delay_ms(2000); // wait for 2 seconds with LED OFF
  }

}
```

[DEMO VIDEO]

1.4.3

Users > virgoyal > Desktop > Blinky1.4.3 > src > C main.c > totcycle

```c
/* Name: main.c
 * Author: Vir Goyal
 */
#include "MEAM_general.h"  // includes the resources included in the MEAM_general.h file

//GLOBAL VARIABLES
double dutycycle =0 ;// dutycycle variable decides how long the LED will be in each (ON/OFF) state.
                     //Also indirectly defines intensity.
double totcycle=1023;

void intensity_inc (double max_intensity){
  for (int on = 0; on <= 100; on++){
    /*if(dutycycle==100){set_led(OFF);}
    if(dutycycle==50){set_led(OFF);}*/
    OCR3A = totcycle*(dutycycle/100); //using OCR3A and waiting for timer to reach this value
    dutycycle=dutycycle+(max_intensity/100); // updating and increasing dutycycle to increase intensity
                                             //This is divided by 100 as loop will run 100 times
    _delay_ms(1); // waiting for 1 ms (this will run a 100 times thus total time = 0.1s)
  }

}

void intensity_dec (double max_intensity){
  for (int off = 0; off <= 100; off++){
  //if(dutycycle==(max_intensity/100)){set_led(ON);} // USED FOR DEBUGGING
  OCR3A = totcycle*(dutycycle/100); //using OCR3A and waiting for timer to reach this value
  dutycycle=dutycycle-(max_intensity/100); // updating and decreasing dutycycle to decrease intensity.
                                           //This is divided by 100 as loop will run 100 times
  _delay_ms(4); // waiting for 4 ms (this will run a 100 times thus total time = 0.4s)
  }
}

int main(void)
{
  _clockdivide(0);
  set(DDRC,6); // setting Port C6 as output
  set (TCCR3B,CS30); //Setting prescaler to 1/1

  // set timer on OCR3A mode 7
  set(TCCR3B,WGM32);
  set(TCCR3A,WGM30);
  clear(TCCR3B,WGM33);
  set(TCCR3A,WGM31);

  //clear at OCR3A, set at rollover
  set (TCCR3A,COM3A1);
  clear (TCCR3A,COM3A0);

  for(int lifeline =0 ; lifeline<=19;lifeline++){
    // lifeline variable defines the degree by which the intensity of LED decreases every iteration of loop.
    // let x denote how much the sum of how much the intensity has decreased by.
    intensity_inc(100-(lifeline/100)); // increase intensity to 100%-x in 0.1s.
    intensity_dec(100-(lifeline/100)); // decrease intensity to 0% in 0.4s
    intensity_inc(50-(lifeline/50));  // increase intensity to 50%-x in 0.1s
    intensity_dec(50-(lifeline/50));  // decrease intensity to 0% in 0.4s
    clear(PORTC,6); // just to  make sure LED is OFF
    _delay_ms(2000); // wait for 2 seconds with LED OFF
  }

}
```

[DEMO VIDEO]

1.5 Retrospective

1.1 - 2 hours
1.2 - 3 hours
1.3 - 5 hours
1.4 - 7 hours

Material used in lab was discussed after it was assigned. Was a little confusing when i did not yet have the tools needed.