Jomo Kenyatta University of Agriculture and Technology

EMT 2405

Design Of Mechatronic Systems I

Low Cost Lighting Management

Kilyungi John Nduli
EN292-0386/2011

Omondi Andrew Eastman
EN292-1658/2011

March 31, 2015

Contents

Chapter 1

Introduction

1.1 Problem Statement

Electricity is expensive. This does not deter, however, people from misusing this resource. Electric lights are normally left on, even when no one is in the rooms to use them. Sometimes, people use the lights even when sunlight(green energy) is efficient enough for the lighting. The best way to curb this is to make a system that manages the use of electricity, with the ability to switch between it and sunlight effectively.

1.2 Objectives

The following are the objectives of the project:

- To provide an automatic system that controls lighting of the room
- To provide a means of lighting different sections using low power LEDs instead of lighting one whole room using high power bulb
- To provide automatic means of controlling home features i.e. door, window
- To provide easy to make mechanisms for the physical design part.

1.3 Justification

This project was undertaken because it provided a means of solving a common issue, electrical wastage. If looked at from one user, it seems insignificant. However, the combined effect of the problem results in a lot of losses. The following are some of the reasons why it is important:

1. It provides a base of controlling various resources of the home automatically. This is because the system can be adapted to include control of other features like electronic power supplies, water management. This will make the resources be used as efficiently as possible.
2. It can easily provide necessary data of power usage to the final consumer, thus helping him budget better for his or her life.
3. The project can be scaled upwards such that it provides easy means of managing electricity for institutions like colleges and universities. This can result in reduction in the power bills for the various institutions. Also the water management feature will also be of significant use.

This project, although seemingly easy, can have a butterfly effect if implemented on the large scale.

Chapter 2

Methodology

In summary, the following steps were followed:[1]

1. Identification of the functional requirement of the system
2. Determination of the nature of motion (i.e., planar, spherical, or spatial mechanism), degrees of freedom (dof), type, complexity of the mechanisms and control of the mechanism. Determination of electrical circuitry to be used in the system.
3. Identification of the structural characteristics associated with some of the functional requirements and electrical characteristics of the circuitry used.
4. Enumeration of all possible kinematic structures that satisfy the structural characteristics. Enumeration of the different electrical circuits to be used in the circuit.
5. Sketching of the corresponding mechanisms and evaluating each of them qualitatively in terms of its capability in satisfying the remaining functional requirements. This results in a set of feasible mechanisms. The various control system are also analysed in this step.
6. Selection of the most promising mechanism for dimensional synthesis, design optimization, computer simulation, prototype demonstration, and documentation. Computer simulation of the various electric circuits and control systems is also done.
7. Enter the production phase. [1]

2.1 Identification of Functional Requirements

The idea was analyzed and discussed. The basic concept was utilizing sun energy for lighting of rooms. As the discussions went on, however, the scope of the project became bigger and bigger. Brainstorming was also performed with members of the class. Considering the time scope of the project and the resources available, the following were the requirements of the System that would be made:

1. Automatically switch off and on the lights, with the options of a manual override of the automatic control(Circuit design).
2. Automatically open the curtains if there is enough sunlight outside the room(Mechanism design).
3. The room is divided into parts, each having a light. If the sunlight does not reach a part of the room, the light should remain on.
4. Control of brightness of the light
5. Automation of opening and closing of the door(Simple security feature)

2.2 Determination of the Nature of Motion and Circuitry

There were two main mechanisms to be designed. These were the curtain opening mechanism and the door opening mechanism.

2.2.1 Curtain opening

The opening of the curtains is a planar motion, with one degree of freedom. The degree of freedom is a translation motion in one axis only.

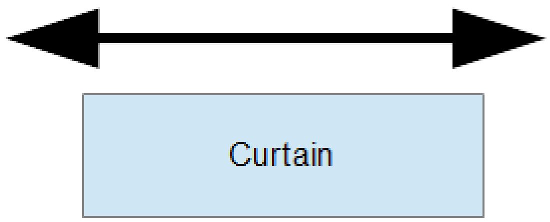


Figure 2.1: Curtain planar motion

2.2.2 Door opening

The opening of the door is a planar motion, with one degree of freedom. The degree of freedom is rotation about one axis of motion. The rotation should be limited, as in less than 90 degrees.

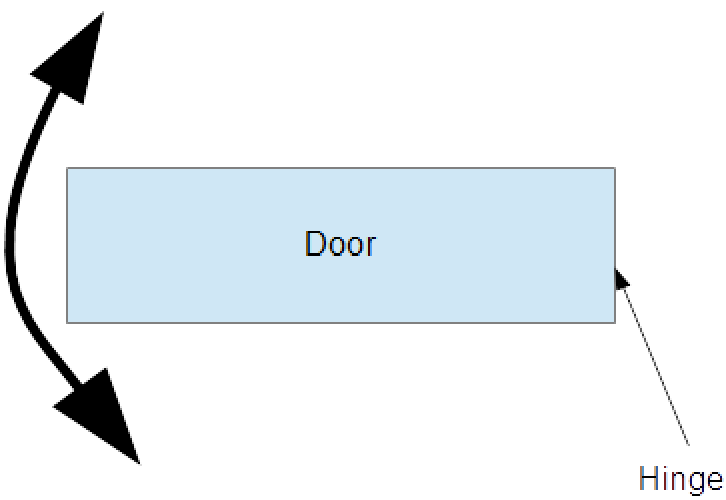


Figure 2.2: Door planar motion

2.2.3 Electrical Lights Arrangement

The room is divided into portions and the lights multiplexed to allow efficient usage of the microcontroller's pins. Each portion of the room has an LDR at an optimum location and an LED associated with this portion. If the LDR does not detect enough light, the LED is lit.

2.2.4 Motor drive circuit

A motor drive circuit that can rotate a DC motor in both clockwise and anticlockwise direction.

2.2.5 LIght Detection Mechanism

A means of getting whether there is enough sunlight outside to light the room.

2.2.6 Control and Display Circuit

Control of the various system should be performed. This will include control of the door, control of the curtains and lighting system control. It should also provide a means for overriding the system to provide full control of the lights to the end user.

2.3 Identification of the Structural Characteristics

The material of choice was wood. This is because it is relatively cheap and easy to get. Basel was chosen because the simulation software used had all of its features and characteristics. The dimensions chosen are relatively smaller than the normal dimension of the objects to cater for the possibility of making a prototype.

2.3.1 Curtain opening

The mechanism will be relatively long and thin. Thin because it should not block the windows themselves. The length used is 65cm, and its maximum height should not be more than 10cm.
Since window curtains have low weights,the mechanism should be able to withstand about 12kg. This value was got by assuming the maximum possible window curtain would be of 2m by 2m by 0.2cm dimensions and that the material it would be made of would be cotton (density 0f 1.5g / cm3).
Its motion should not be noisy and opening and closing should occur relatively fast.
It should also have a means of attaching the curtain onto it.

2.3.2 Door opening

The mechanism should open a door 70cm long.
The maximum weight to be supported during design is 20kg.
Opening and closing should not be noisy and should occur relatively fast.
It should also have a means of attaching a door to it.

2.3.3 Electrical Lights Arrangement

The room is divided into 6 different portions each a different distance from the main window.

2.3.4 Motor drive circuit

The motor should run for a set time and constant RPM(Revolutions Per Minute). This time is calculated and preset during coding. The time for the opening is set for 5 seconds in the code for demonstration purposes.

2.3.5 LIght Detection Mechanism

The mechanism used should provide a means of communicating the value of light brightness to the control unit. This value is compared to a reference voltage, and the digital value corresponding to the input read.

2.3.6 Control and Display circuit

This is used to inform the user about the system. It should be located at ideal locations, and have options to be easily expanded i.e. multiple others added to other different locations.

2.4 Enumeration of Possible Kinematic Structures and Electrical Circuits

2.4.1 Curtain opening

For the curtain opening, since it is a simple translatory motion, the following means can be used to make the mechanism:

Cam and Follower

A cam is a rotating machine element which gives reciprocating or oscillating motion to another element known as follower. The cam would be made such that the maximum distance travelled would be 70 cm, have a dwell to enable switching off, and have a return stroke of the same distance. However, this will result in a large cam and also the design will be complicated to cater for gravity prevent follower from falling since the operations is in a horizontal direction).

Crank and slider

A mechanism will be designed that receives rotary motion from a motor and translates it into linear motion using a 4 bar mechanism. The curtain will be attached to the slider.

Belt drive mechanism

As the pulleys rotate, the belt moves in a linear motion. The curtain can be placed on the pulley and careful control of motor rotation made, such that opening and closing is achieved.

2.4.2 Door opening

Depending on the door design, it can be opened in various ways:

crank and rocker mechanism

The crank receives the motor motion. Since the door is a hinge joint, the output link needs to rock through 90 degrees only otherwise breakage of the mechanism will occur.

Rack and pinion

This will only work for sliding doors, whereby the door is connected to the pinion and the gears moves it back and forth.

2.5 Sketching of Corresponding Mechanisms and Analyzing

2.5.1 Curtain opening

Crank and slider

The following 2D representation was done in linkage.

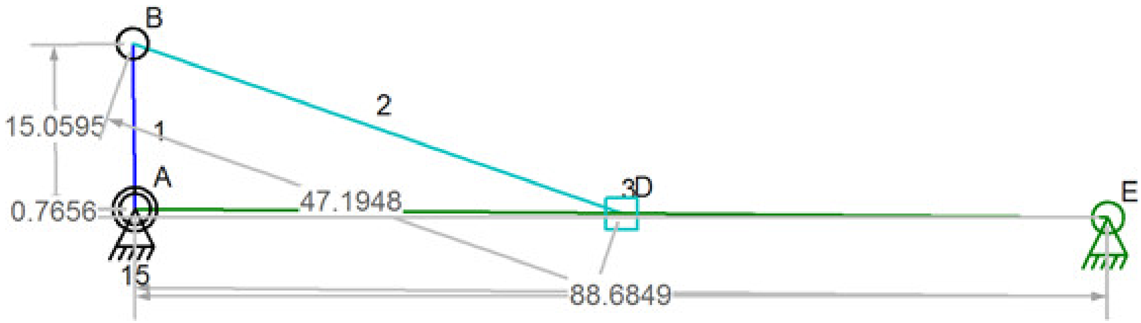


Figure 2.3: Slider mechanism as generated in Linkage

Belt drive mechanism

The following 2D representation was done in SAM.

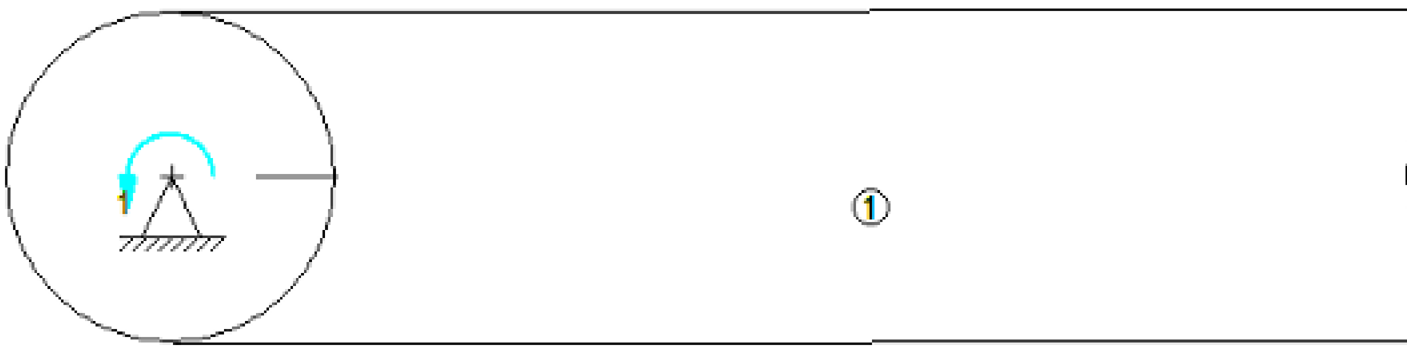


Figure 2.4: Belt as generated in SAM61

2.5.2 Door opening

Crank and Rocker mechanism

The figure is shown below:

2.5.3 Electrical Lights arrangement

The only option for the lights are LEDs because of their low power usage.

2.5.4 Motor Drive Circuit

Different types of motors can be used. This can include:

1. DC Motors whereby timing is used for stopping and starting the motor. A circuit is also made to provide both clockwise and anticlockwise motion.
2. Stepper Motor whereby better control is achieved because of the properties of the Stepper i.e. 90 degrees rotations can be made.

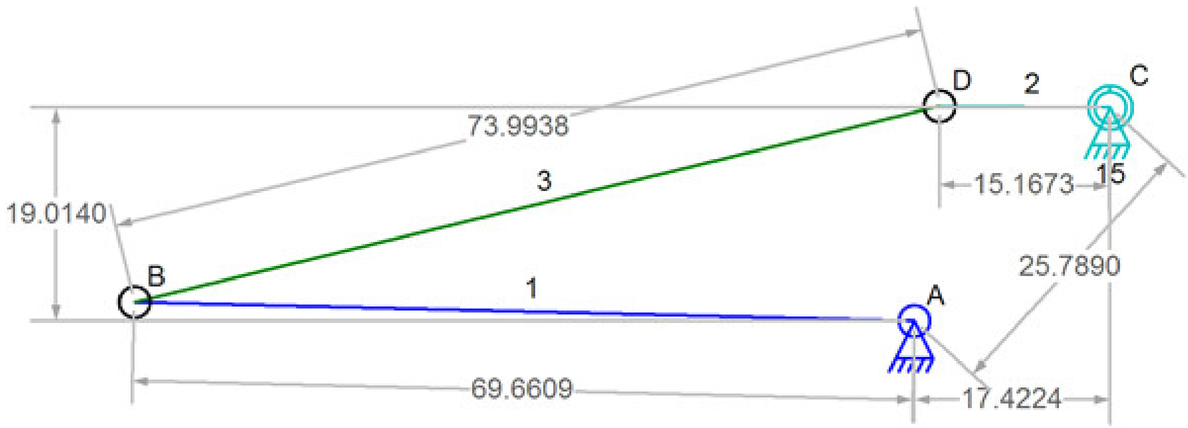


Figure 2.5: Door mechanism

2.5.5 Light Detection Mechanism

LDRs are used in a voltage divider system to provide relative voltage based on the brightness of the system.

2.5.6 Control and Display Circuit

The following are the options got for the display:

1. Laptop or PC whereby Rx Tx is used to communicate with the computer via USB. A special protocol will be made such that the computer will be able to understand the information got from the Control System. A user interface is made using Process or Java to provide easy to understand means of the information.

2. LCD whereby 8 pins of the Control System are used specifically to provide communication to the LCD(8 pin mode).

The control system is coded in C using AVR Studio. The microcontroller used is Atmega 328P.

2.6 Selection of the Most Promision Mechanism and Circuit Elements

For the curtain opening, the slider mechanism is better. The belt mechanism would have been the best, but it would have been more expensive and more difficult to set up from scratch.
For the door opening, the crank and rocker mechanism was chosen. This was because the main design was for a hinged door. If the door was sliding, the rack and pinion would have been the best option.
For the Electrical Lights, LEDs are used.
For the Motor Drive Circuit, DC motors are used with timing coded into the microcontroller. This is because it will be cheaper than using stepper motors.
For the Display Circuit, an LCD will be used because it will be cheaper and end up consuming less power. The microcontroller used is the Atmega 328P.

2.7 Production phase

Due to time constrains, this phase was not reached.

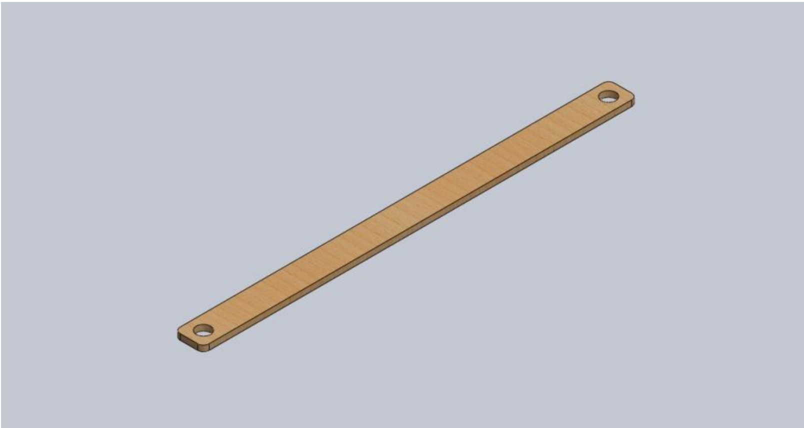
Chapter 3

Results and discussion

Solid works was used to make the above mentioned mechanisms. Attached are the reports generated by solid works:

3.1 Door mechanism

Attached is the analysis of the door mechanism performed by solid works. It provides stress analysis using Von Mises' Stress criterion, deformation analysis and allowable factors of safety. From the results, it can be seen that the mechanism is safe.



Simulation of doorlink

Date: 29 March 2015
Designer: Solidworks
Study name: SimulationXpress Study
Analysis type: Static

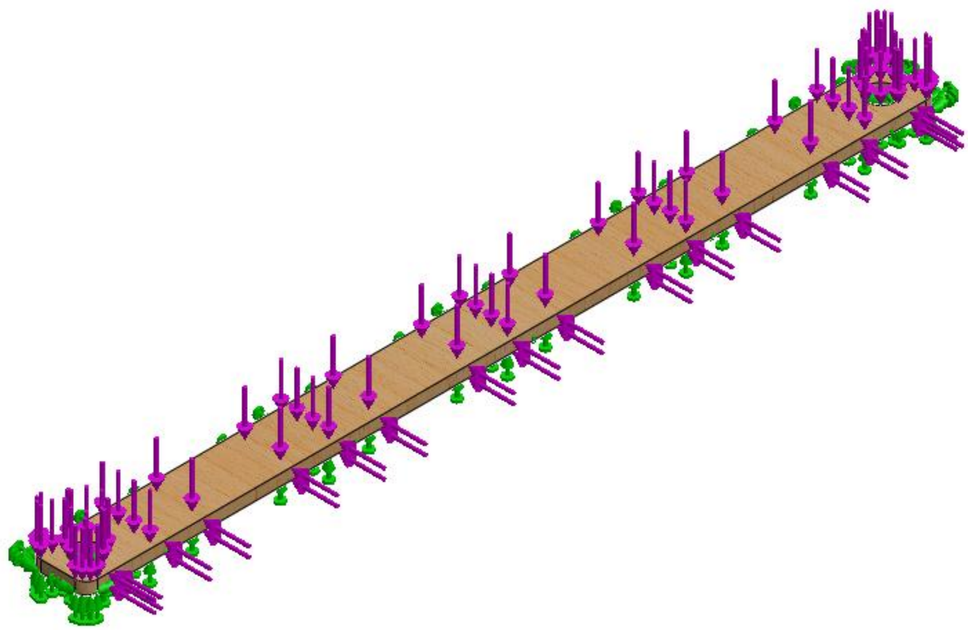
Table of Contents

Description

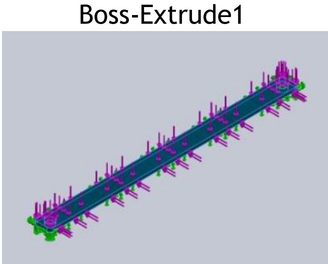
No Data

Assumptions

Model Information

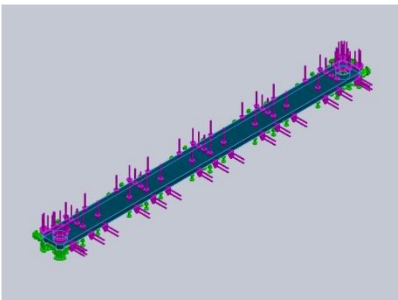


Model name: doorlink
Current Configuration: Default

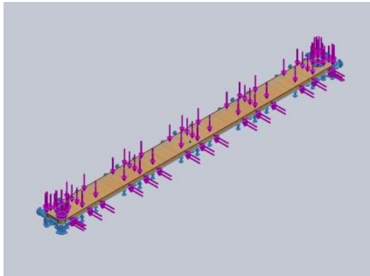
Solid Bodies			
<L_MdInf_SldBd_Nm/>	Treated As	Volumetric Properties	Document Path/Date Modified
Boss-Extrude1 	Solid Body	**Mass:0.0582882 kg** **Volume:0.000364324 m^3** **Density:159.99 kg/m^3** **Weight:0.571224 N**	**C:\Users\Sensei\Documents\Projects\SolidWorks\homeautomation\doormechanism\doorlink.SLDPRT** Mar 28 07:56:17 2015
<L_MdInf_ShlBd_Nm/>	<L_MdIn_ShlBd_Fr/>	<L_MdInf_ShlBd_VolProp/>	<L_MdIn_ShlBd_DtMd/>

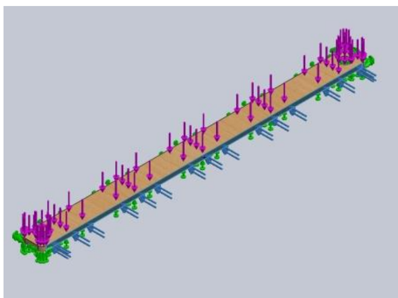
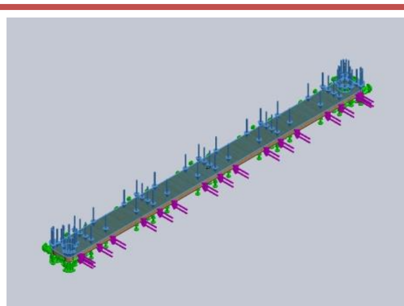
<L_MdInf_CpBd_Nm/>	<L_MdInf_CompBd_Props/>		
<L_MdInf_BmBd_Nm/>	<L_MdIn_BmBd_Fr/>	<L_MdInf_BmBd_VolProp/>	<L_MdIn_BmBd_DtMd/>

Material Properties

Model Reference	Properties	Components
	**Name:** Balsa **Model type:** Linear Elastic Isotropic **Default failure criterion:** Unknown **Yield strength:** 20 N/mm^2	SolidBody 1(Boss-Extrude1)(doorlink)

Loads and Fixtures

Fixture name	Fixture Image	Fixture Details
Fixed-3		**Entities:** 1 face(s) **Type:** Fixed Geometry

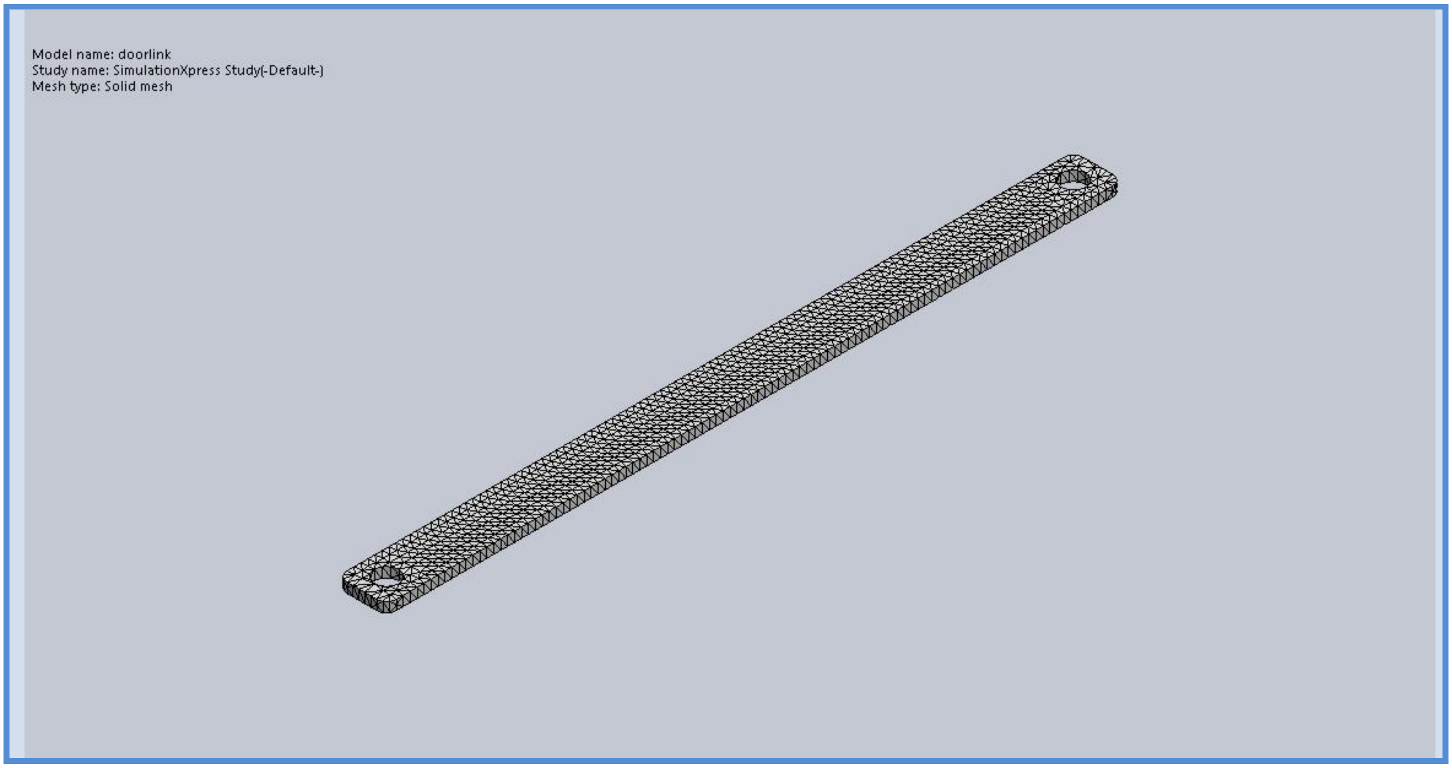
Load name	Load Image	Load Details
Force-4		**Entities:** 1 face(s) **Type:** Apply normal force **Value:** 20 N **Phase Angle:** 0 **Units:** deg
Force-5		**Entities:** 1 face(s) **Type:** Apply normal force **Value:** 10 N **Phase Angle:** 0 **Units:** deg

Mesh Information

Mesh type	Solid Mesh
Mesher Used:	Standard mesh
Automatic Transition:	Off
Include Mesh Auto Loops:	Off
Jacobian points	4 Points
Element Size	0.714404 cm
Tolerance	0.0357202 cm
Mesh Quality	High

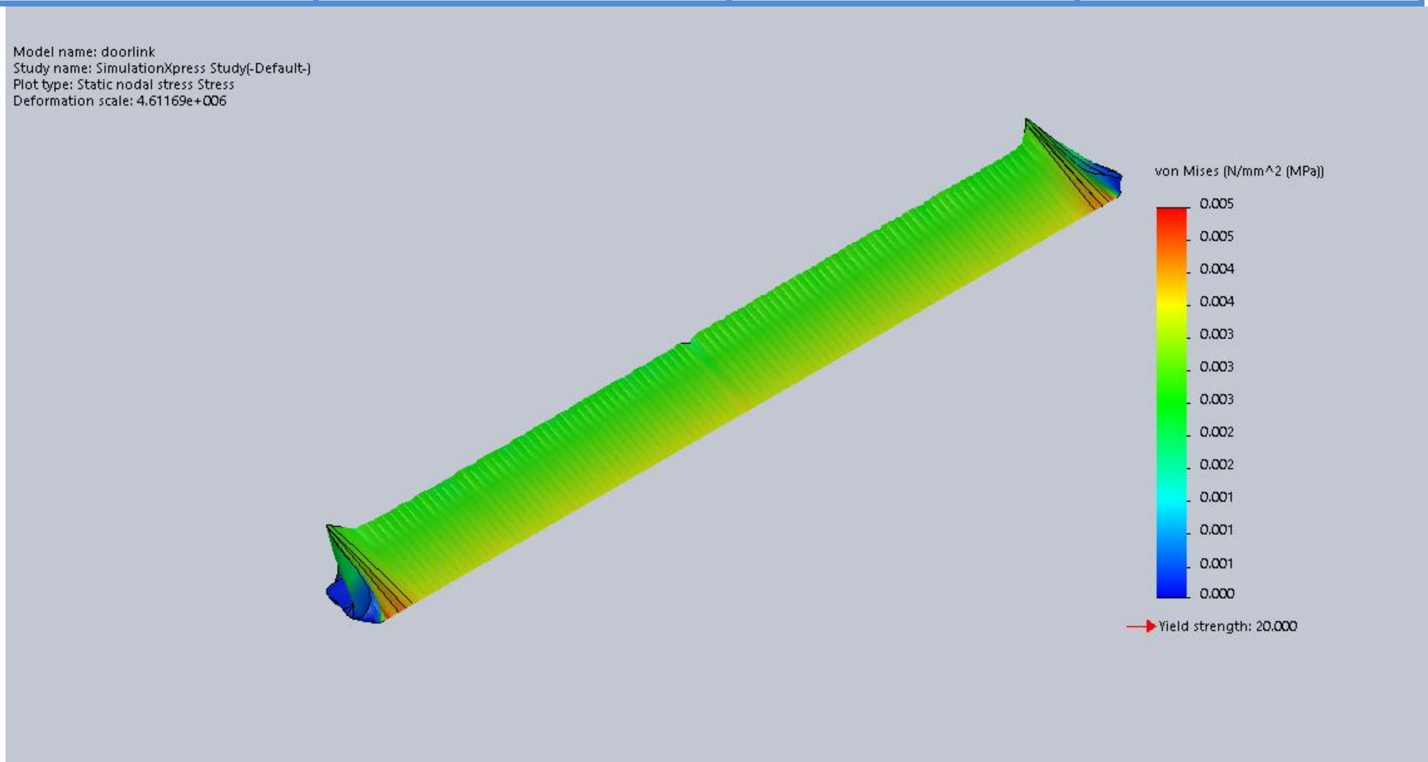
Mesh Information - Details

Total Nodes	14953
Total Elements	8488
Maximum Aspect Ratio	4.4992
% of elements with Aspect Ratio < 3	99.9
% of elements with Aspect Ratio > 10	0
% of distorted elements(Jacobian)	0
Time to complete mesh(hh;mm;ss):	00:00:01
Computer name:	LEXXY


Model name: doorlink
Study name: SimulationXpress Study(-Default-)
Mesh type: Solid mesh

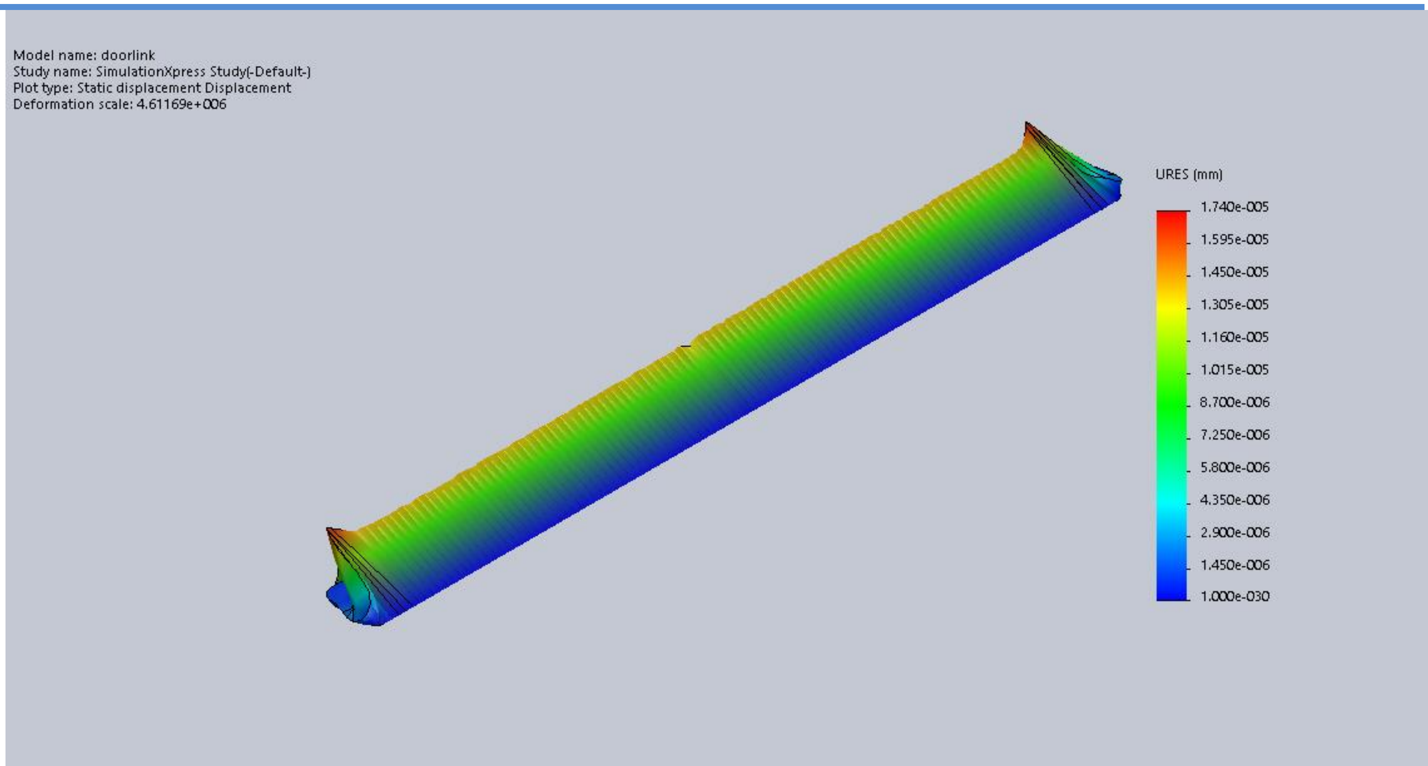
Study Results

Name	Type	Min	Max
Stress	VON: von Mises Stress	0.00011296 N/mm^2 (MPa) Node: 9864	0.00512363 N/mm^2 (MPa) Node: 9125



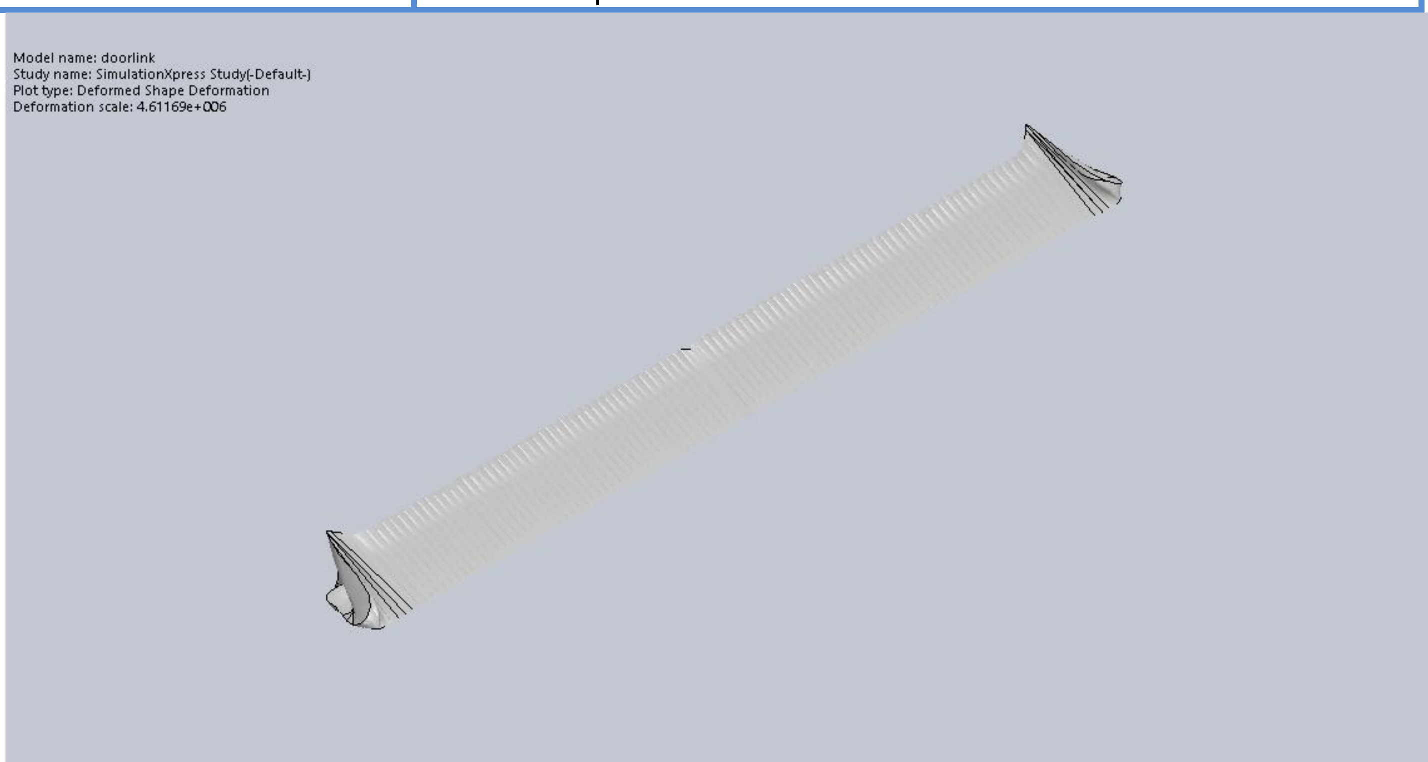
doorlink-SimulationXpress Study-Stress-Stress

Name	Type	Min	Max
Displacement	URES: Resultant Displacement	0 mm Node: 7	1.73991e-005 mm Node: 13964



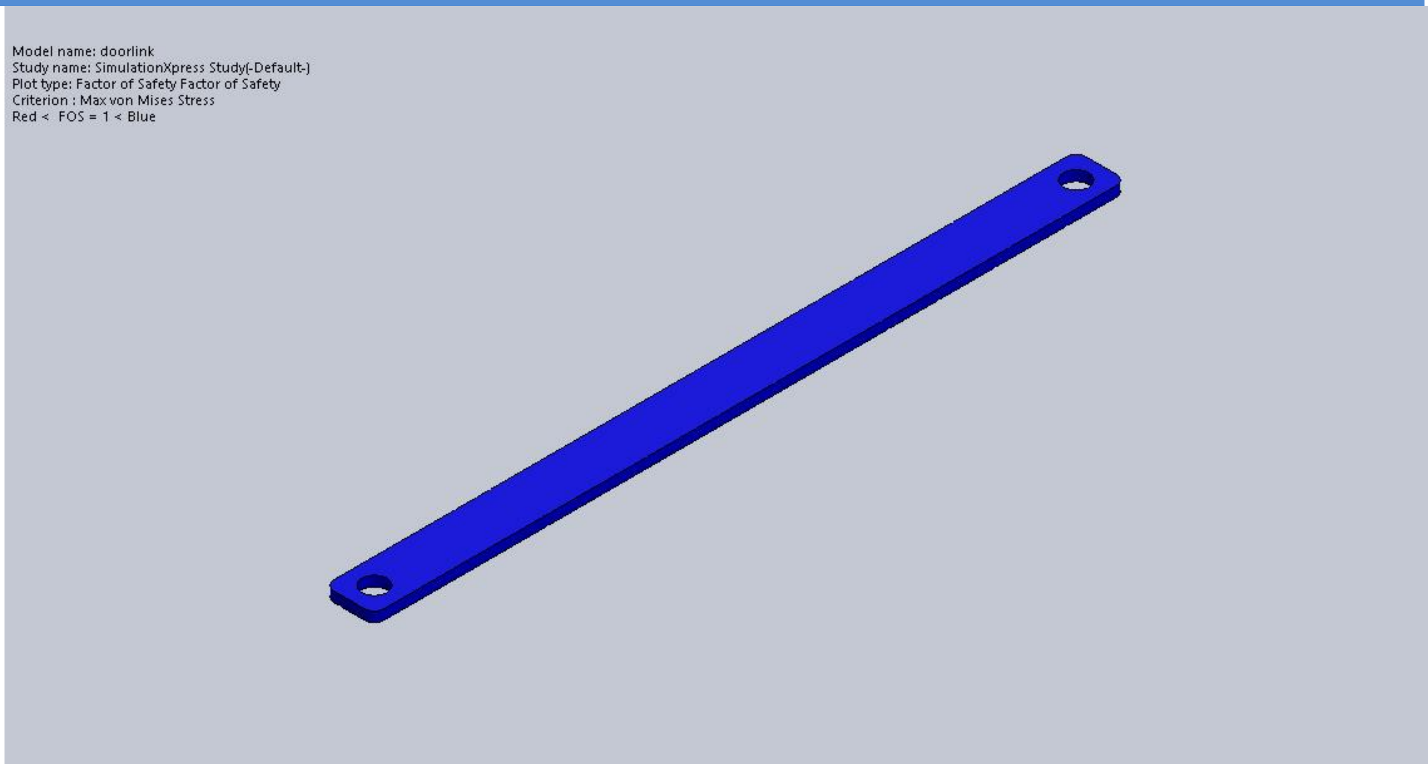
doorlink-SimulationXpress Study-Displacement-Displacement

Name	Type
Deformation	Deformed Shape



doorlink-SimulationXpress Study-Displacement-Deformation

Name	Type	Min	Max
Factor of Safety	Max von Mises Stress	3903.48 Node: 9125	177054 Node: 9864

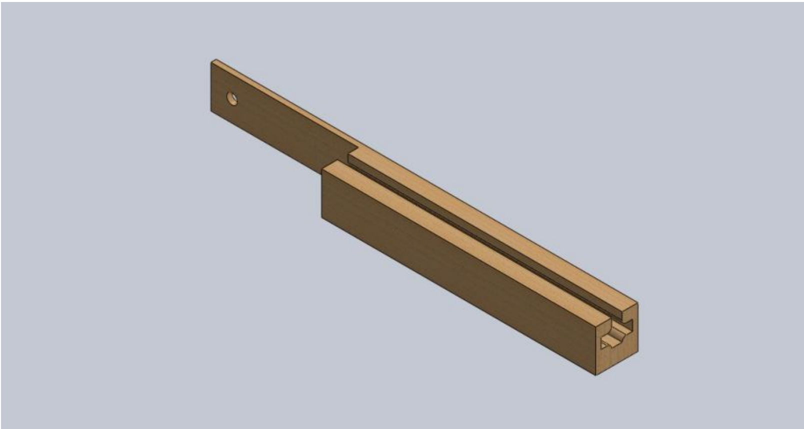


doorlink-SimulationXpress Study-Factor of Safety-Factor of Safety

Conclusion

3.2 Curtain Mechanism

Attached is the analysis of the curtain mechanism performed by solid works. It provides stress analysis using Von Mises' Stress criterion, deformation analysis and allowable factors of safety. From the results, it can be seen that the mechanism is safe.



Simulation of slidingcompartment

Date: 29 March 2015
Designer: Solidworks
Study name: SimulationXpress Study
Analysis type: Static

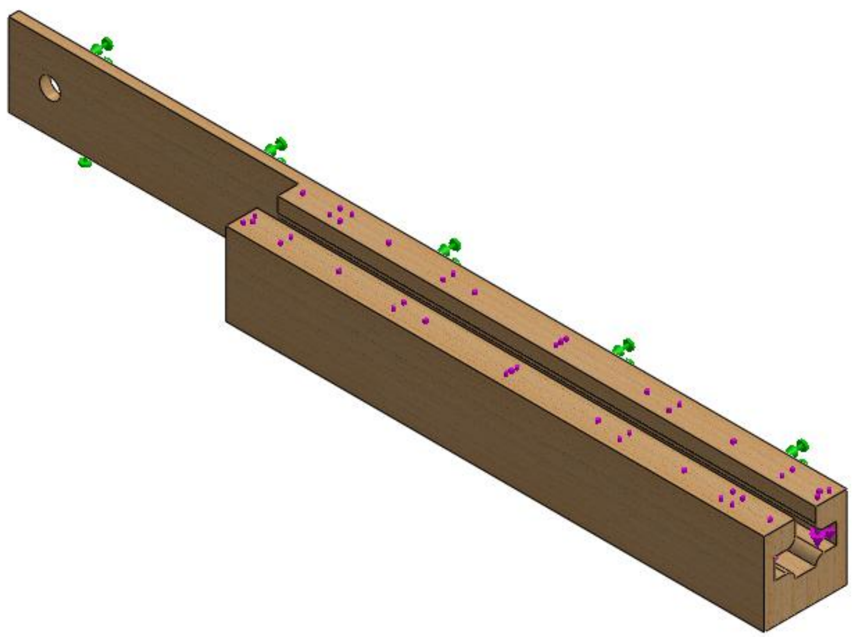
Table of Contents

Description

No Data

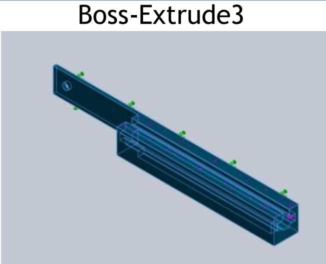
Assumptions

Model Information



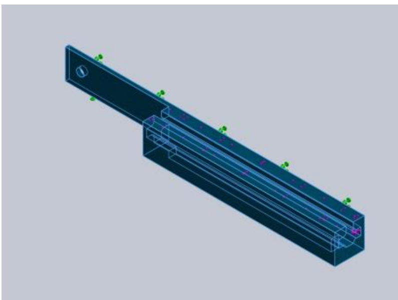
Model name: slidingcompartment
Current Configuration: Default

Solid Bodies

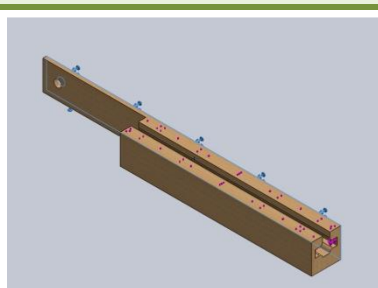
<L_MdInf_SldBd_Nm/>	Treated As	Volumetric Properties	Document Path/Date Modified
Boss-Extrude3 	Solid Body	Mass:0.721545 kg Volume:0.00450994 m^3 Density:159.99 kg/m^3 Weight:7.07114 N	C:\Users\Sensei\Documents\Projects\SolidWorks\homeautomation\curtainopeningmechanism\slidingcompartment.SLDPRT Mar 29 16:11:05 2015
<L_MdInf_ShlBd_Nm/>	<L_MdIn_ShlBd_Fr/>	<L_MdInf_ShlBd_VolProp/>	<L_MdIn_ShlBd_DtMd/>

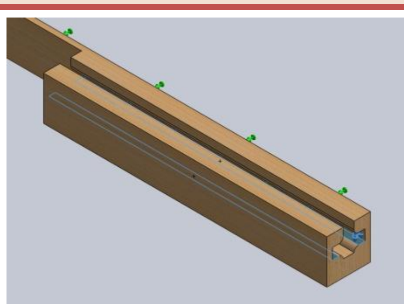
<L_MdInf_CpBd_Nm/>	<L_MdInf_CompBd_Props/>		
<L_MdInf_BmBd_Nm/>	<L_MdIn_BmBd_Fr/>	<L_MdInf_BmBd_VolProp/>	<L_MdIn_BmBd_DtMd/>

Material Properties

Model Reference	Properties	Components
	**Name:** Balsa **Model type:** Linear Elastic Isotropic **Default failure criterion:** Unknown **Yield strength:** 20 N/mm^2	SolidBody 1(Boss-Extrude3)(slidingcompartment)

Loads and Fixtures

Fixture name	Fixture Image	Fixture Details
Fixed-2		**Entities:** 1 face(s) **Type:** Fixed Geometry

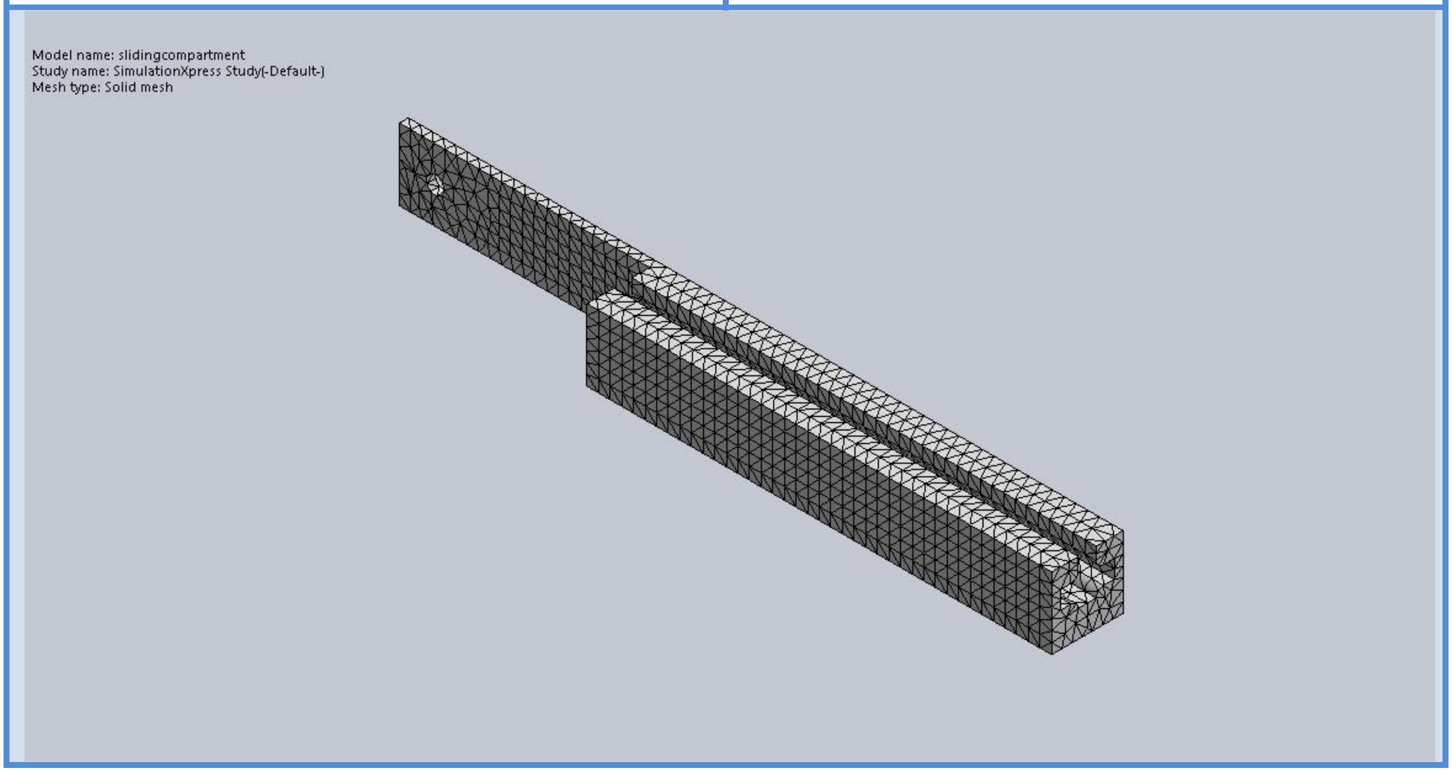
Load name	Load Image	Load Details
Force-2		**Entities:** 2 face(s) **Type:** Apply normal force **Value:** 60 N **Phase Angle:** 0 **Units:** deg

Mesh Information

Mesh type	Solid Mesh
Mesher Used:	Curvature based mesh
Jacobian points	4 Points
Maximum element size	0 cm
Minimum element size	0 cm
Mesh Quality	High

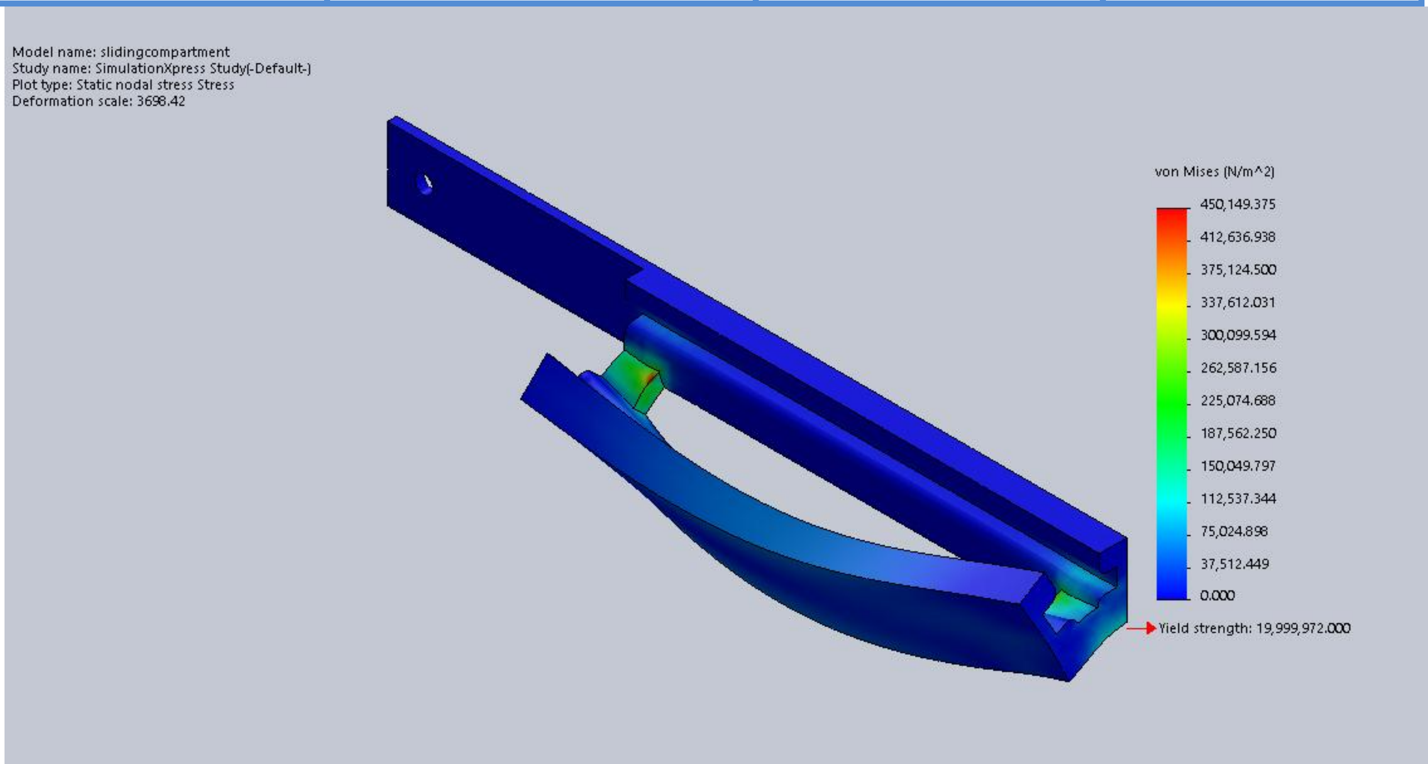
Mesh Information - Details

Total Nodes	15275
Total Elements	8400
Maximum Aspect Ratio	9.4266
% of elements with Aspect Ratio < 3	97.7
% of elements with Aspect Ratio > 10	0
% of distorted elements(Jacobian)	0
Time to complete mesh(hh;mm;ss):	00:00:01
Computer name:	LEXXY



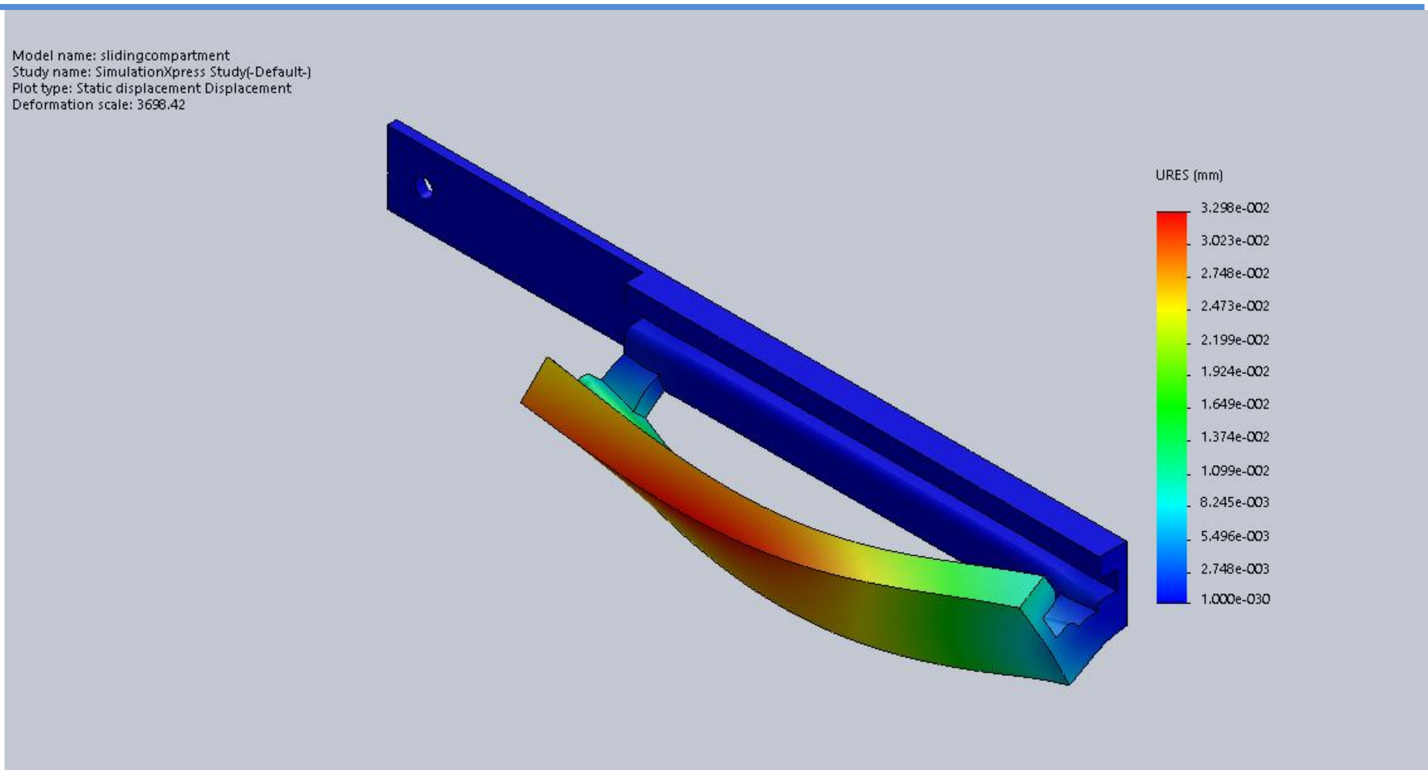
Study Results

Name	Type	Min	Max
Stress	VON: von Mises Stress	3.15749e-007 N/m^2 Node: 12462	450149 N/m^2 Node: 14814



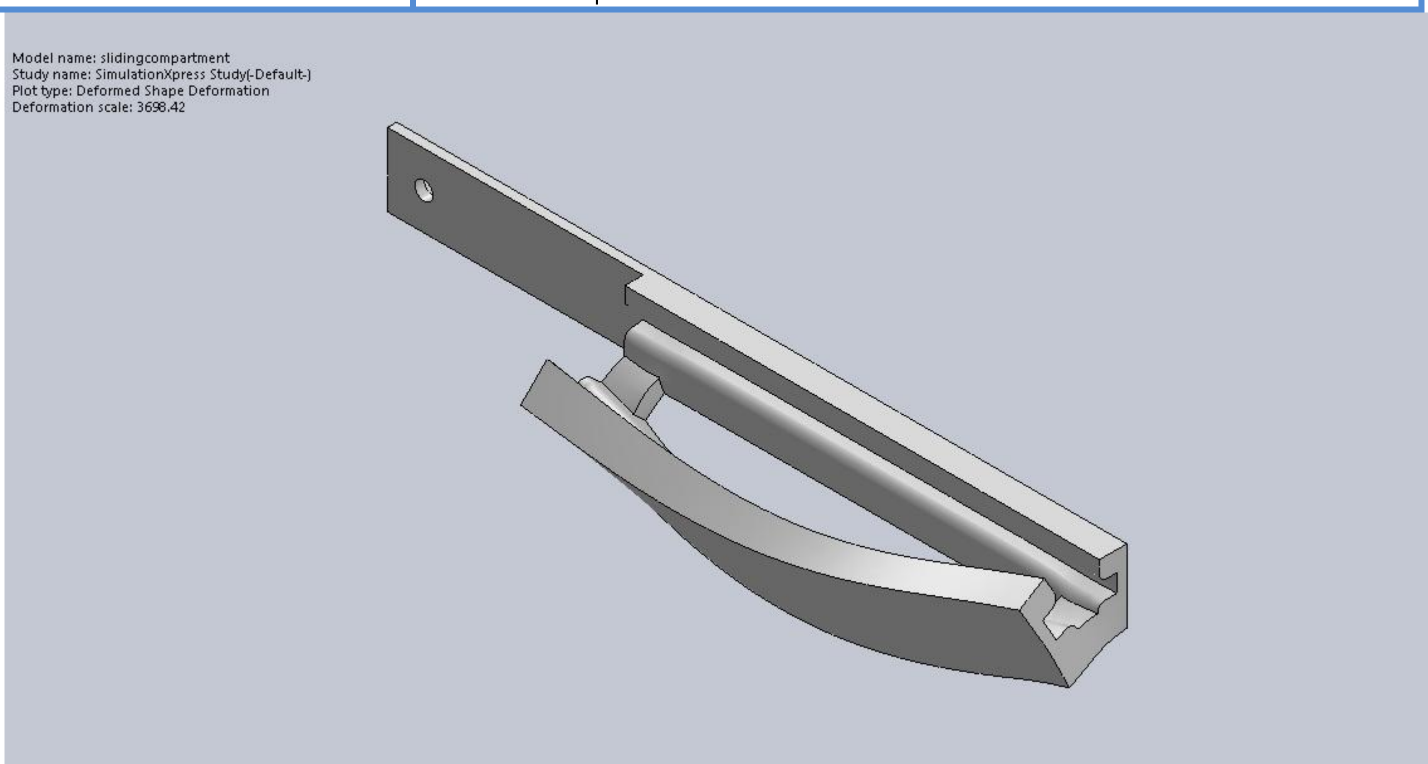
slidingcompartment-SimulationXpress Study-Stress-Stress

Name	Type	Min	Max
Displacement	URES: Resultant Displacement	0 mm Node: 1	0.0329788 mm Node: 8833



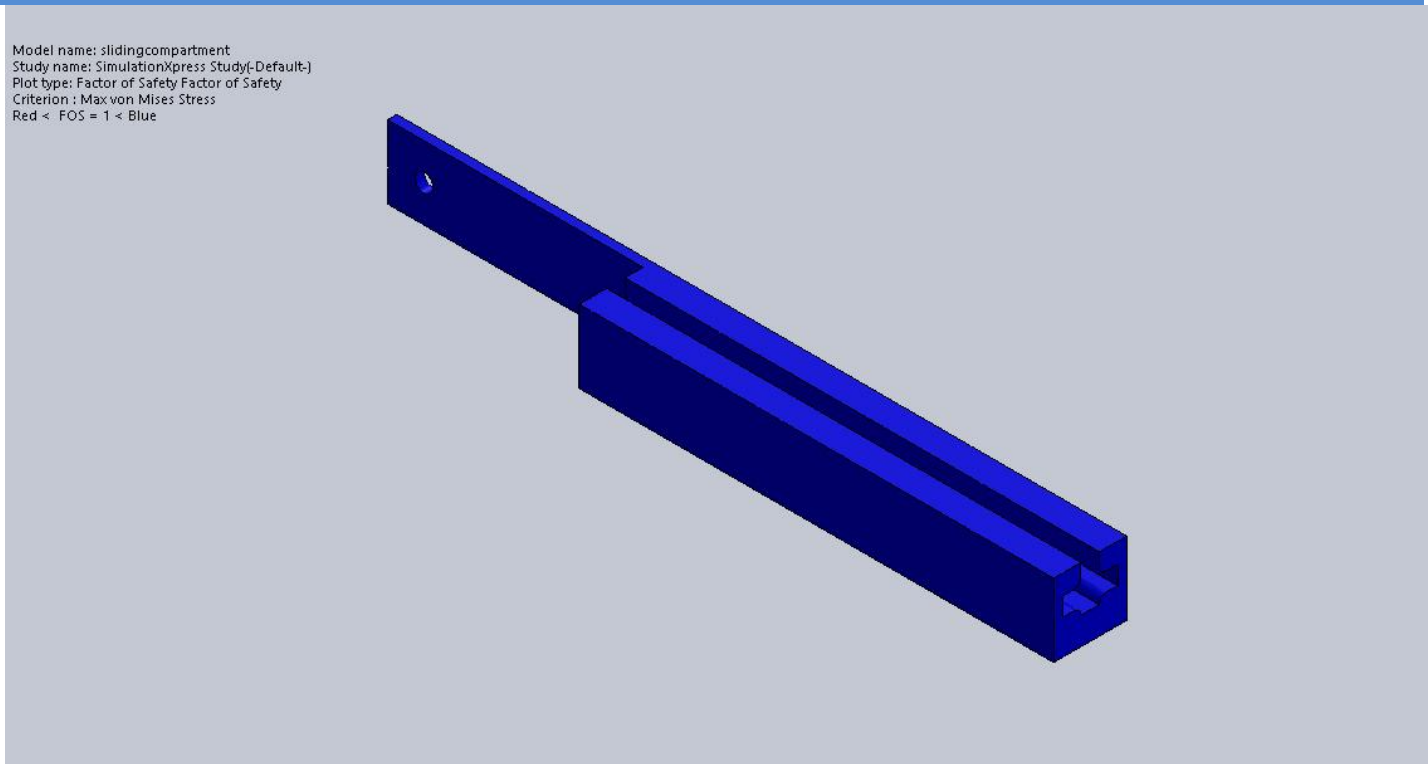
slidingcompartment-SimulationXpress Study-Displacement-Displacement

Name	Type
Deformation	Deformed Shape



slidingcompartment-SimulationXpress Study-Displacement-Deformation

Name	Type	Min	Max
Factor of Safety	Max von Mises Stress	44.4296 Node: 14814	6.33414e+013 Node: 12462



slidingcompartment-SimulationXpress Study-Factor of Safety-Factor of Safety

Conclusion

3.3 Lighting system

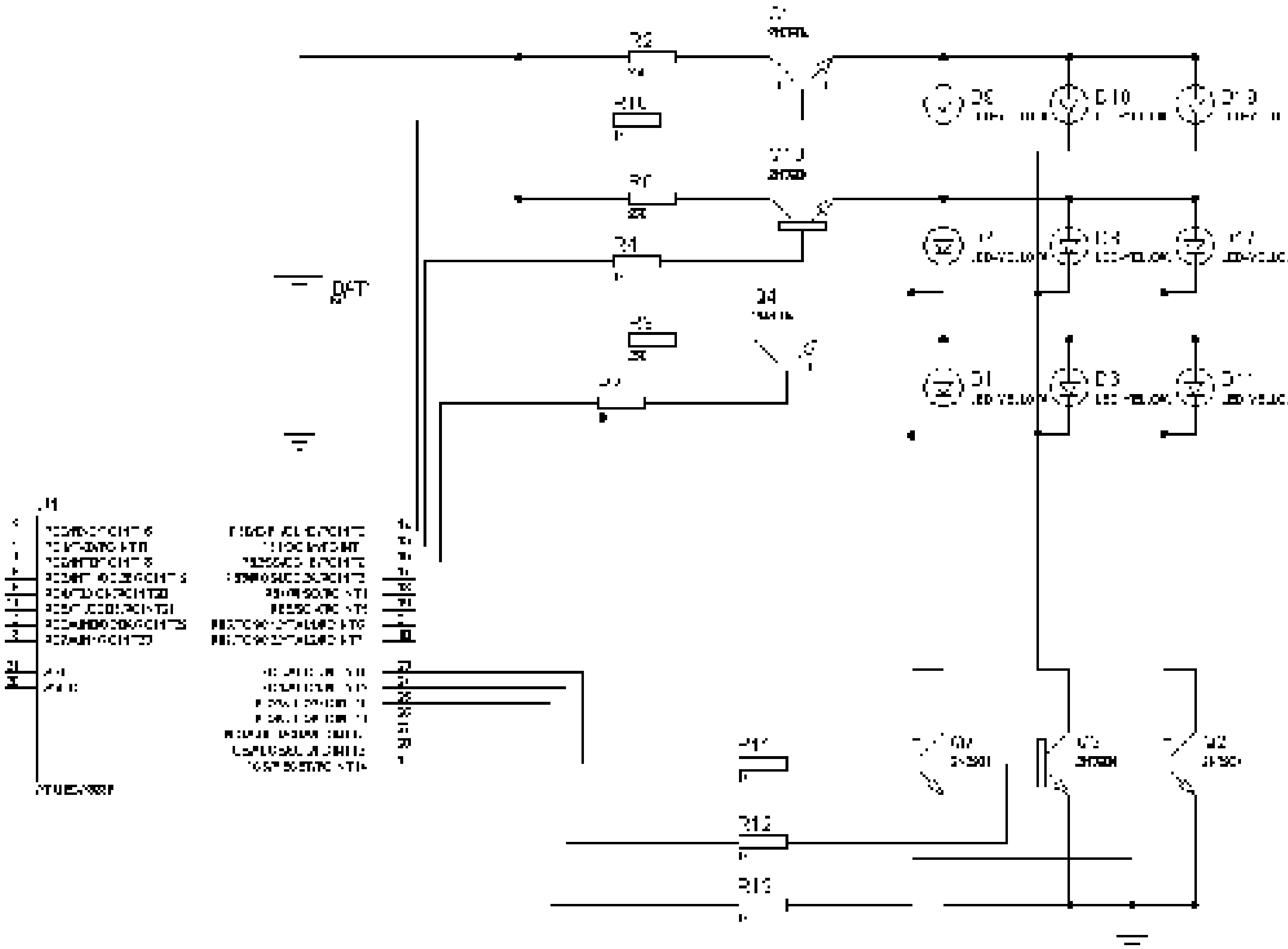


Figure 3.1: Light circuit

The figure shows the final circuit for the lights. It used demultiplexing, in that few pins are used to control many LEDs. Refer to the appendix for the code that lights this system. The code basically lights each LED for one second, before going to the next one. This is a proof of concept showing how easy it is to light individlual lights using the microcontroller. As the number of LEDs increase, the efficiency of this system can be seen eg. To control 36 LEDs only 12 pins are required. This 12 pins can be further demultiplexed in the same way to get that only 7 pins will be required at the end.

3.4 Motor circuit

The motor circuit is implemented in the opening and closing of the doors and curtains.A driver chip is used (The L293D chip) to control the rotation of the the motor depending on the control signals received from the microcontroller. In the event a forward rotation is is to be performed, bit 0 receives a high while bit one receives a low. In the event a reverse rotation is needed, bit 0 receives a low while bit 1 receives a high. In the event no rotation is required both inputs are given a low signal or

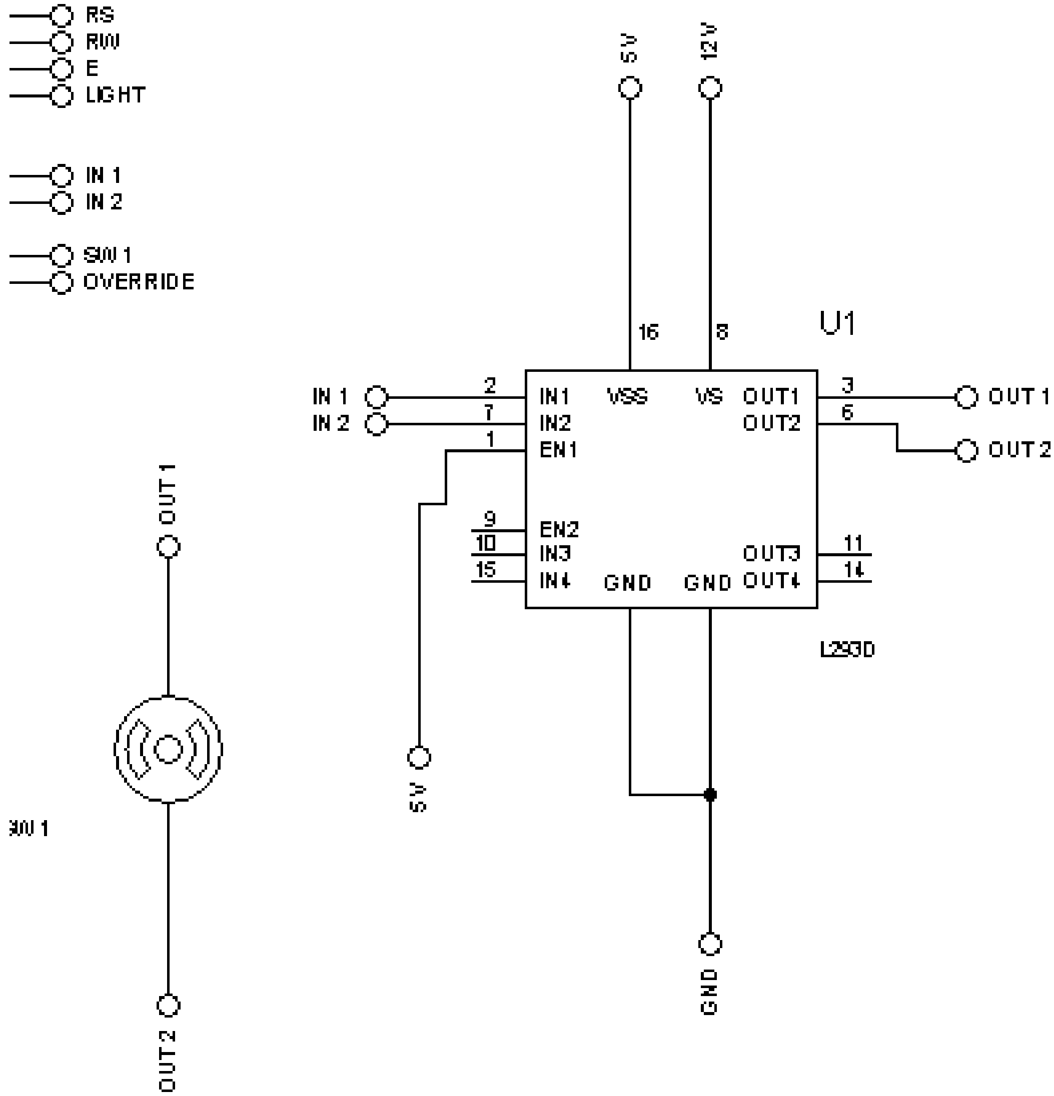


Figure 3.2: Section showing motor system

both inputs are given a high signal. For demontration purposes, a 12V Dc motor is used that will be rotating at a a contant speed.

3.5 The Display

The Display involves the use of a LCD screen to display the status of events in the system. The LM016L is used in this case. It consists of a 16x2 character display that is connected to the microcontroller. The LCD is connected to the Microcontroller in 8 bit mode and therefore 8 pins of the microcontroller are used to transfer data to be displayed and another 3 pins are used to send the control signals. The special commands used for the LCD for functions like initialization and clearing of the screen were obtained from the datasheet of the LCD and implemented in the code.

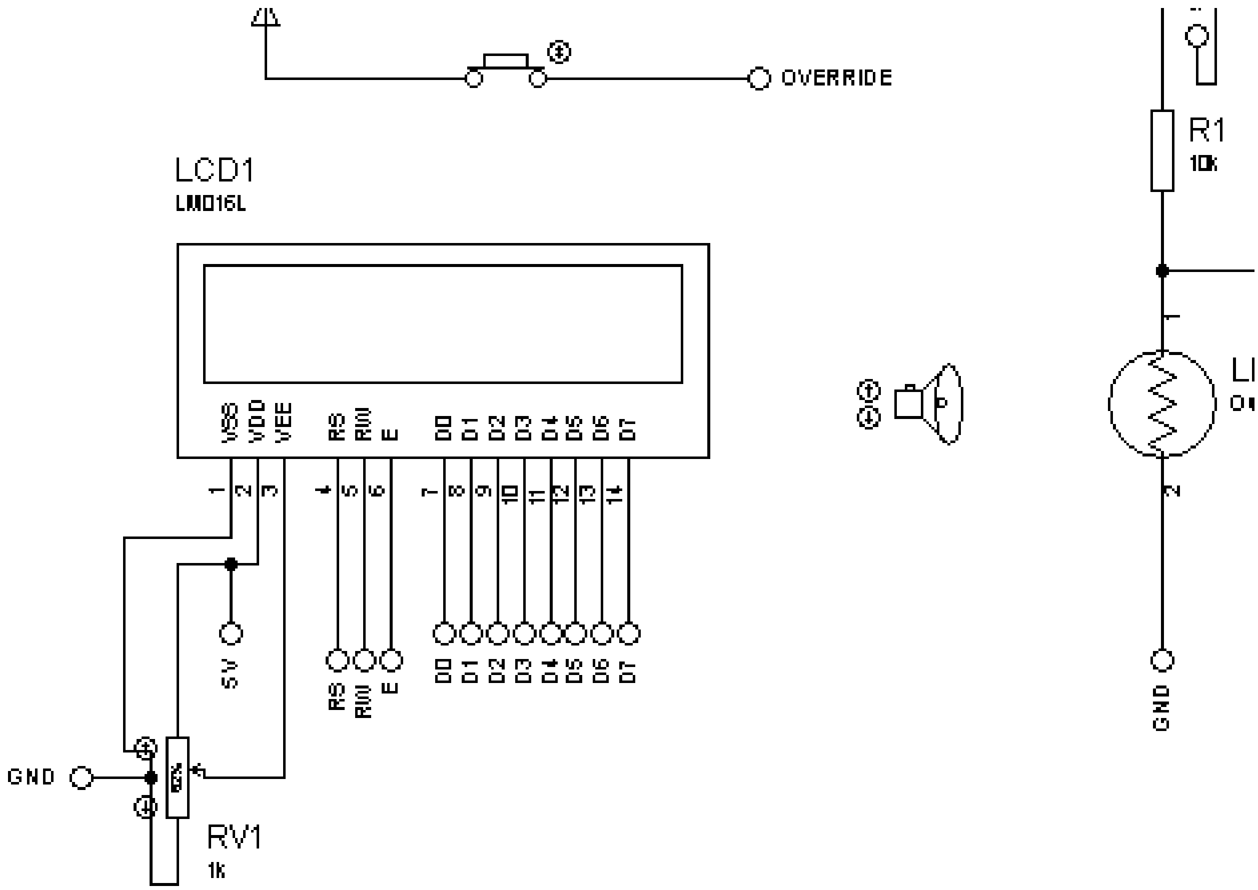


Figure 3.3: Section showing display system

3.6 The Motion Sensor

Due to the limitations of simulations and budgetary constraints, The motion sensor was not included in the simulations. However, the PIR(Passive Infrared Sensor) was identified for use in the event of the system implementation. Since it is a passive sensor it does not waste energy but reads changes in the surrounding area. Also, compared to the other types of sensors e.g the Microwave and Ultrasonic sensors, the PIR sensor proved to be much cheaper thus saving cost.

The PIR sensor works with the principle of measuring IR light radiating from objects in its view. Since all objects above absolute zero temperature emit heat energy in the form of heat energy, this can be detected by electronic devices since they travel at infrared wavelengths. Thus in the event a person moves about this reading increases from that of room temperature to that of body temperature to trigger that a person is present.

Chapter 4

Conclusion

The approximate cost of the system will be about 5000 Kshs. This is cheap. Analysis of the system also shows that the system works.

Appendix

.1 Generation of mechanisms

[2] First of all, each of the mechanisms has to follow Grashof's theorem.

$$s + l \leq p + q \tag{1}$$

To make the crank and rocker, the motor is attached to the shorter side, while the output link is the length of the motor. An arbitrary length of 15cm was used for the crank of the system and the other lengths got from linkage simulator as the links were connected based on the above criterion. Also the slider system was generated in a similar manner.

.2 Display, Control and Motor Drive Circuit Codes

Information to control the LCD was got from the datasheet [4].
The motor driver configuration was also got from the L293 driver IC dadtasheet [3].
The configuration of ports to use during code was got from the atmega 328 datasheet [5]

```
/*
 * DMSProject_2015.c
 *
 * Created: 08/03/2015 02:08:27
 *  Author: Kuzan
 */


#define F_CPU 1000000UL
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <avr/sfr_defs.h>
#define LCD_DATA PORTD                  // In my case PORTB is the PORT from which I sen
#define Control_PORT PORTB              // In my case PORTC is the PORT from which I set
#define En PORTB2             // Enable signal
#define RW PORTB1             // Read/Write signal
#define RS PORTB0             // Register Select signal
#define state_closed 0  //curtains state
#define state_open 1    //crrtains state
#define outside_bright 1      // state
#define outside_dark 0  // state
#define auto_mode 1
```

```
#define overide_mode 0

#define outside_threshold 53
static int state;
static int mode;
static int outside;
static uint16_t brightness;

void LCD_cmd(unsigned char cmd)
{
        LCD_DATA=cmd;
        Control_PORT =(0<<RS)|(0<<RW)|(1<<En);   // RS and RW as LOW and EN as HIGH
        _delay_ms(1);
        Control_PORT =(0<<RS)|(0<<RW)|(0<<En);   // RS, RW , LOW and EN as LOW
        _delay_ms(1);
        return;
}

void LCD_write(unsigned char data)
{
        LCD_DATA= data;
        Control_PORT = (1<<RS)|(0<<RW)|(1<<En); // RW as LOW and RS, EN as HIGH
        _delay_ms(1);
        Control_PORT = (1<<RS)|(0<<RW)|(0<<En); // EN and RW as LOW and RS HIGH
        _delay_ms(1);                           // delay to get things executed
        return ;
}

void move_forward(int time){
        int seconds=0;
        TCNT1 = 0;
        PORTB |= 1 << PINB6;
        while(seconds<time){//one second passes
                if (TCNT1 > 15625)
                {
                        TCNT1 = 0;
                        seconds++;
                }
        }
        PORTB &= 0 << PINB6;
}


void move_back(int time){
        int seconds=0;
        TCNT1 = 0;
        PORTB |= 1 << PINB7;
        while(seconds<time){
                if (TCNT1 > 15625)//one second passes
                {
                        TCNT1 = 0;
```

```
                    seconds++;
                }
        }
        PORTB &= 0 << PINB7;
}

void LCDWriteString(const char *msg)
{
        LCD_cmd(0x01); // clear LCD
        _delay_ms(1);

        LCD_cmd(0x0E); // cursor ON
        _delay_ms(1);

        LCD_cmd(0x80); // ---8 go to first line and --0 is for 0th position
        _delay_ms(1);
        int i=0;
        while (msg[i]!='\0'){
                LCD_write(msg[i]);
                i++;
        }
}

void init_LCD(){
        LCD_cmd(0x38); // initialization of 16X2 LCD in 8bit mode
        _delay_ms(1);

        LCD_cmd(0x01); // clear LCD
        _delay_ms(1);

        LCD_cmd(0x0E); // cursor ON
        _delay_ms(1);

        LCD_cmd(0x80); // ---8 go to first line and --0 is for 0th position
        _delay_ms(1);
}

init(){

        DDRB=0xFF; //set pin 0 and 1 of port B to be output for the motor
        DDRD=0xFF; //set port D to be outputs for LDR
        DDRC=0xF0; //set port C to be input
        TCCR1B |= 1<<CS10 | 1<<CS11; //set prescaler of internal clock to 64
        PORTC &=0<<PORTC5;
        state=state_closed;
        outside=outside_bright;
        mode=auto_mode;
        _delay_ms(200); //pause 2 seconds
        ADCSRA=0x8F;
        ADMUX=0x20;
        sei();
```

```
        ADCSRA|=(1<<ADSC);
}
int main(void)
{
        init();
        init_LCD();
        LCDWriteString("Welcome");
        _delay_ms(2000);
        while(1)
        {
                if(bit_is_set(PINC,1)){
                        mode=overide_mode;
                }
                else if(bit_is_clear(PINC,0)){
                        mode=auto_mode;
                }

                if(outside==outside_dark && state==state_open && mode==auto_mode){
                        LCDWriteString("Dark Oustside");
                        _delay_ms(1000);
                        LCDWriteString("Lights On");
                        PORTC |=1<<PORTC5;
                        _delay_ms(500);
                        LCDWriteString("Curtains Closing ...");
                        move_back(5);//close curtains
                        state=state_closed;
                        LCDWriteString("Curtains Closed");
                        _delay_ms(1000);
                        LCDWriteString("Dark Outside");

                        LCD_cmd(0x01); // clear LCD
                        LCDWriteString("Level: ");
                        LCD_write(brightness);
                        _delay_ms(1);
                        _delay_ms(1000);
                }
                else if(outside==outside_bright && state==state_closed && mode==auto_mod
                        LCDWriteString("Bright Oustside");
                        _delay_ms(1000);
                        LCDWriteString("Lights Off");
                        PORTC &=0<<PORTC5;
                        _delay_ms(500);
                        LCDWriteString("Curtains opening ...");
                        move_forward(5);//open curtains
                        state=state_open;
                        LCDWriteString("Curtains Open");
                        _delay_ms(1000);
                        LCDWriteString("Dark Bright");

                        LCD_cmd(0x01); // clear LCD
```

```
                LCDWriteString("Level: ");
                LCD_write(brightness);
                _delay_ms(1);
                _delay_ms(1000);
            }
            else if(mode==overide_mode){
                LCDWriteString("Override Mode");
                _delay_ms(1000);
            }else{
                LCD_cmd(0x01); // clear LCD
                LCDWriteString("Level: ");
                LCD_write(brightness);
                _delay_ms(1);
                _delay_ms(1000);
            }
        }
}

ISR(ADC_vect){

        uint16_t x= (ADCL>>6);
        uint16_t y=(ADCH<<2);
        x=x|y;
        brightness=(x/102.4)+48;
        if(brightness>outside_threshold){
            outside=outside_dark;
        }
        else{
            outside=outside_bright;
        }


        ADCSRA |= 1<<ADSC;

}
```

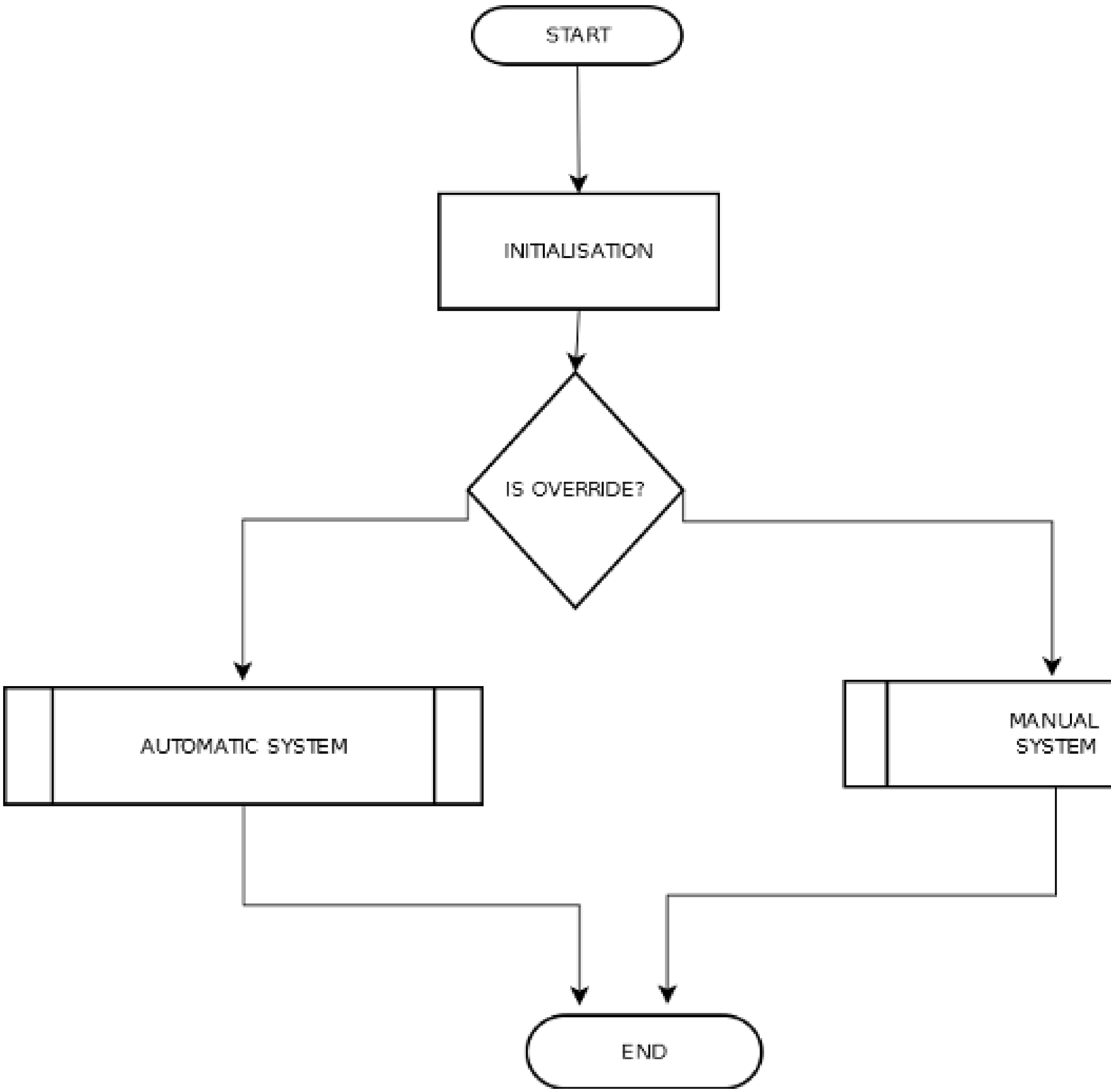
.3 Lighting system code

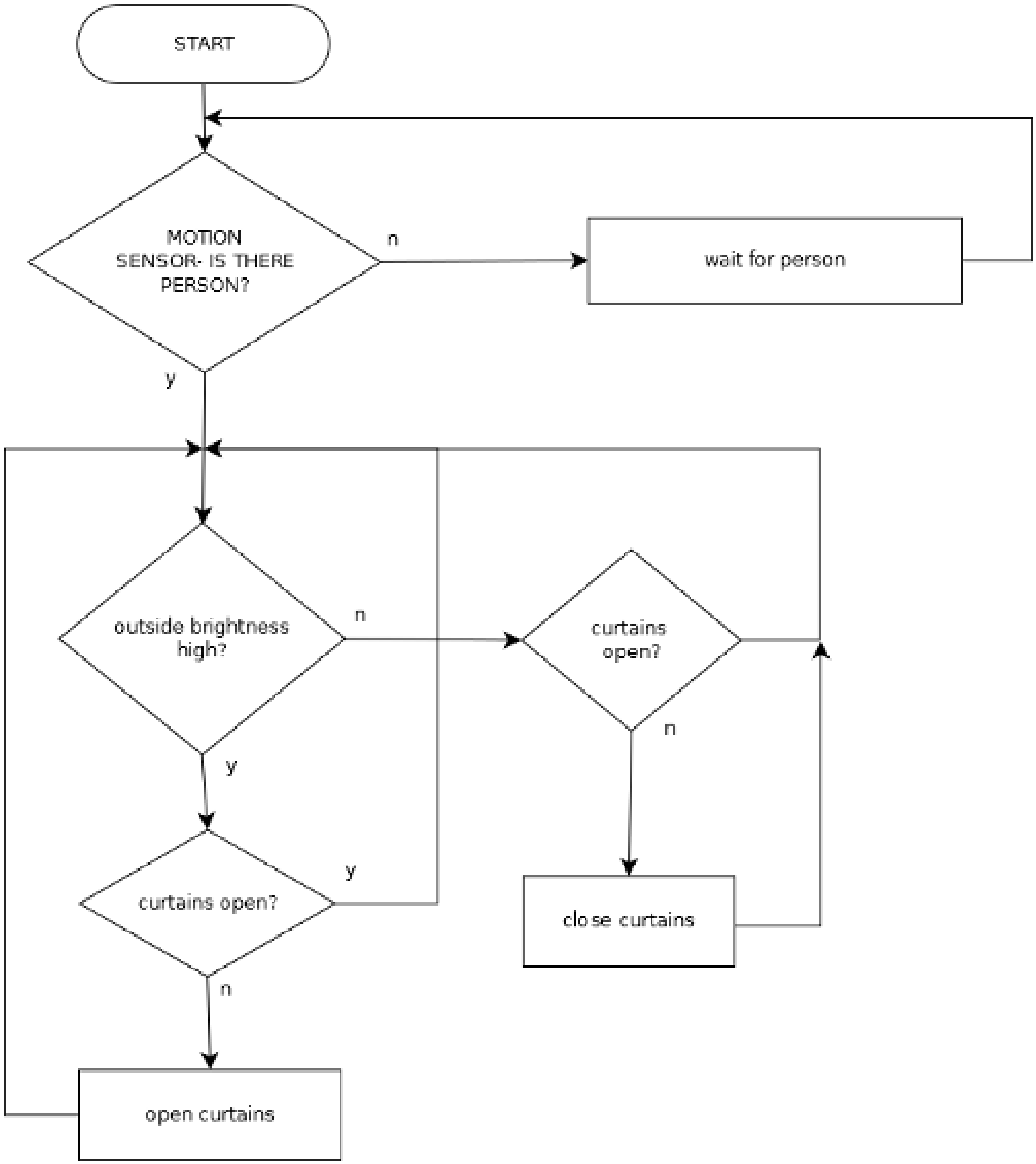
The pin configuration used was got from the atmega 328P datasheet [5].

```
/*
 * LEDlights.cpp
 *
 * Created: 30/03/2015 19:18:17
 *  Author: Sensei
 */


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

```
void LEDswitchOff(int vert, int horiz){
        switch(vert){
                case 0: PORTB &= ~ (1<<PB0); break;
                case 1: PORTB &= ~ (1<<PB1);break;
                case 2: PORTB &= ~ (1<<PB2); break;
        }
        switch(horiz){
                case 0: PORTC &= ~ (1<<PC0); break;
                case 1: PORTC &= ~ (1<<PC1);break;
                case 2: PORTC &= ~ (1<<PC2); break;
        }
}
void LEDswitchOn(int vert, int horiz){
        switch(vert){
                case 0: PORTB |= (1<<PB0); break;
                case 1: PORTB |= (1<<PB1);break;
                case 2: PORTB |= (1<<PB2); break;
                case 3: PORTB |= (1<<PB3); break;
                case 4: PORTB |= (1<<PB4); break;
        }
        switch(horiz){
                case 0: PORTC |= (1<<PC0); break;
                case 1: PORTC |= (1<<PC1);break;
                case 2: PORTC |= (1<<PC2); break;
                case 3: PORTC |= (1<<PC3); break;
        }
}

void looponebyone(){
        for(int i=0; i<3; i++){
                for(int j=0; j<3; j++){
                        LEDswitchOn(i,j);
                        _delay_ms(1000);
                        LEDswitchOff(i,j);
                }
        }
}

int main(void)
{
        DDRB |= (1<<PB2|1<<PB1|1<<PB0);
        DDRC |= (1<<PC2|1<<PC1|1<<PC0);

    while(1)
    {
        looponebyone();
    }
}
```

.4 Flowcharts


START
INITIALISATION
IS OVERRIDE?
AUTOMATIC SYSTEM
MANUAL
SYSTEM
END


START
MOTION SENSOR- IS THERE PERSON?
n
wait for person
y
outside brightness high?
n
curtains open?
y
curtains open?
y
n
n
close curtains
open curtains

Bibliography

[1] Lung-Wen Tsai, Mechanism Design Enumeration of Kinematic Structures According to Function, CRC Press, 2001

[2] Yi Zhang with Susan Finger and Stephanie Behrens, Introduction to Mechanisms, webpaper

[3] L293,L203D Quadruple Half H Drivers datasheet.

[4] LM016L-LM016XMBI LCD datasheet

[5] ATmega48A/PA/88A/PA/168A/PA/328/P datasheet