**Course Code:** RME 2202

**Course Name:** Microcontroller and Programmable Logic Controller Lab

**Experiment No:** 02

**Experiment Name:** Simulating a DC Motor with 8051 Microcontroller Using Keil and Proteus Software

**Prepared By:** Rabeya Akter

Roll no: SK-092-015

2nd year 2nd semester

Department of Robotics and Mechatronics Engineering,

University of Dhaka

**Submitted to:** Dr. Shamim Ahmed Deowan

Chairman,

Department of Robotics & Mechatronics Engineering,

University of Dhaka

**Date of submission:** 04 January 2021

Objectives:

- To learn basic coding of 8051 micro-controller
- To be familiar with hardware and software interfacing of 8051 micro-controller
- To see how the program works in micro-controller

Theory:

Simulation is the imitation of the operation of a real-world process or system over time. The act of simulating something first requires that a model be developed; this model represents the key characteristics, behaviors and functions of the selected physical or abstract system or process. The model represents the system itself, whereas the simulation represents the operation of the system over time.

Simulation is used in many contexts, such as simulation of technology for performance optimization, safety engineering, testing, training, education, and video games. Often, computer experiments are used to study simulation models. Simulation is also used with scientific modeling of natural systems or human systems to gain insight into their functioning, as in economics. Simulation can be used to show the eventual real effects of alternative conditions and courses of action. Simulation is also used when the real system cannot be engaged, because it may not be accessible, or it may be dangerous or unacceptable to engage, or it is being designed but not yet built, or it may simply not exist.

Proteus is a simulation and design software tool developed by Labcenter Electronics for Electrical and Electronic circuit design. It also possess 2D CAD drawing feature. It deserves to bear the tagline “From concept to completion”. It is a software suite containing schematic, simulation as well as PCB designing.

ISIS is the software used to draw schematics and simulate the circuits in real time. The simulation allows human access during run time, thus providing real time simulation.

ARES is used for PCB designing. It has the feature of viewing output in 3D view of the designed PCB along with components. The designer can also develop 2D drawings for the product.

ISIS has wide range of components in its library. It has sources, signal generators, measurement and analysis tools like oscilloscope, voltmeter, ammeter etc., probes for real time monitoring of the parameters of the circuit, switches, displays, loads like motors and lamps, discrete components like resistors, capacitors, inductors, transformers, digital and analog Integrated circuits, semi-conductor switches, relays, microcontrollers, processors, sensors etc.

The microcontroller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as training or teaching tool.

A by-product of microprocessor which is smart, intelligent and programmable is the microcontroller. The same fabrication techniques and programming concepts that make possible the general purpose microprocessor also yielded the microcontroller. Nowadays many communication, digital entertainment, portable devices, are controlled by it.

Programming instructions or physical pin connections determine the use of any multifunction pins. The system designer decides the functions is to be used and designs the hardware and software affecting that pin accordingly. It has 40 pins. Among them, Program Store Enable named PSEN reads signal for external program memory & is normally active low. EA ,External Access Enable is active low to access external program memory locations 0 to 4K .To indicate that program code is stored in external Rom this pin must be connected to the ground. When an EA pin is connected to the Ground the 8051 fetches opcode from external ROM by using PSEN.

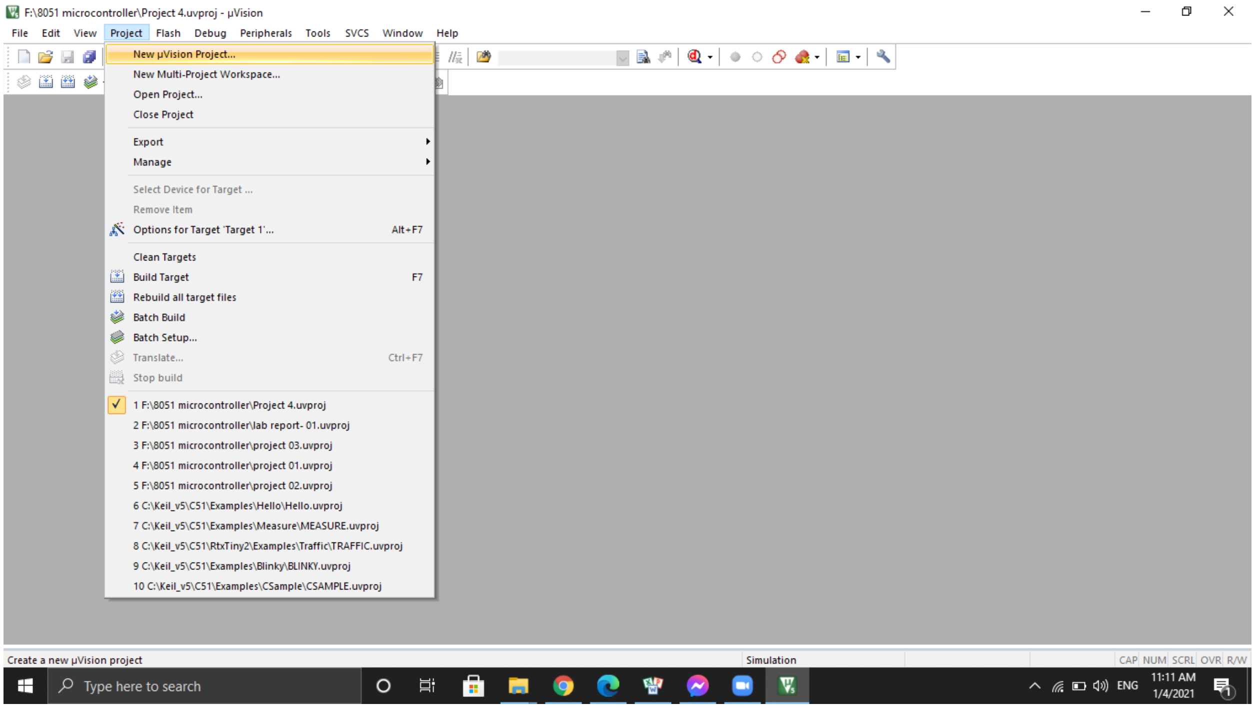
Equipment:

- Hardware- Desktop/ Laptop
- Software- 1.
  1. Keil uVision5
  2. Proteus 8 professional

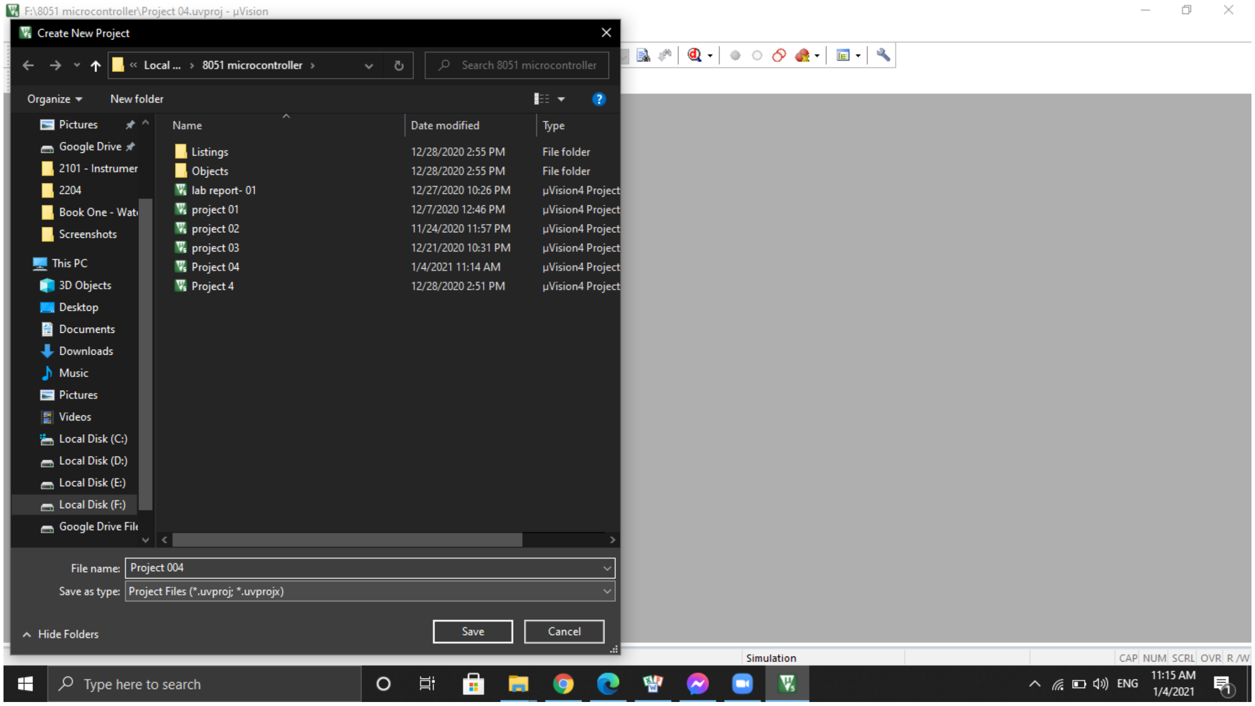
Procedure:

Keil uVision5 part: - For simulation purpose firstly the source code is needed to be written in Keil uVision5 and a hex file of the code is needed to be created. This is done step by step according to the following procedure-

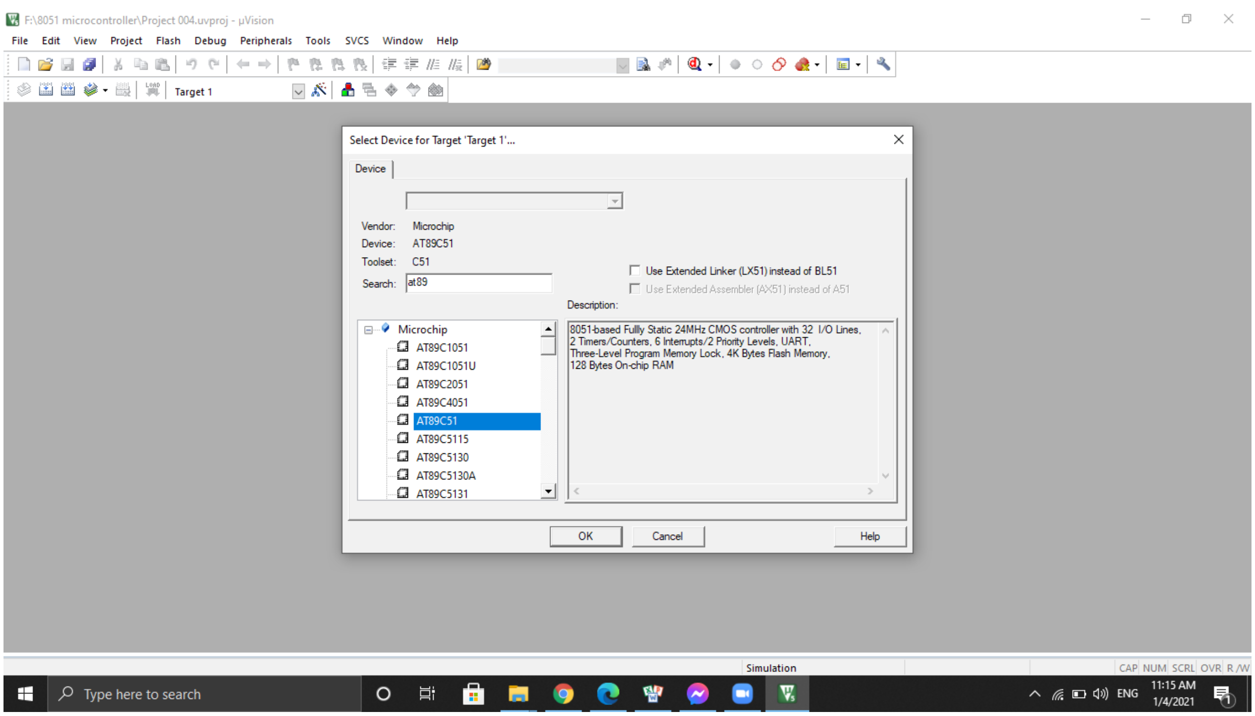
Step 1: At first I opened Keil software. Then from the following window I selected “New uVision project” from “Project”.



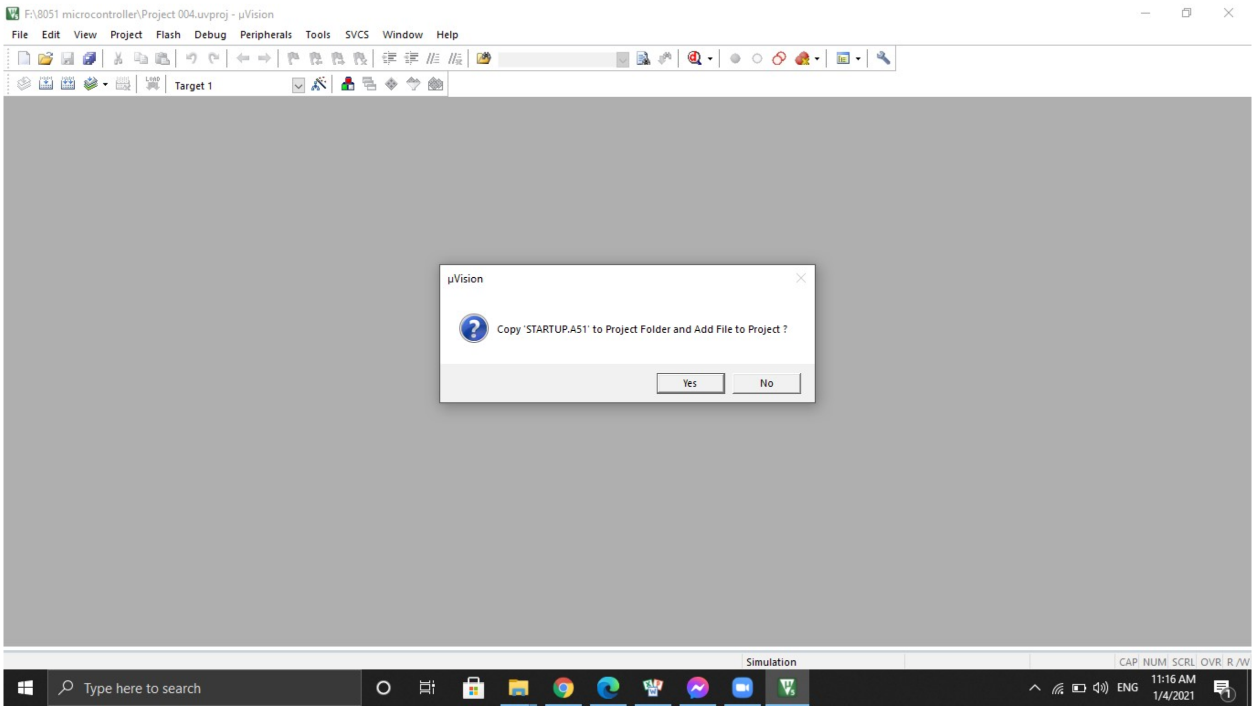

Step 2: Then the window appeared for saving the project by giving a name. I named the project “Project 004” and saved it as “.uvproj” type.



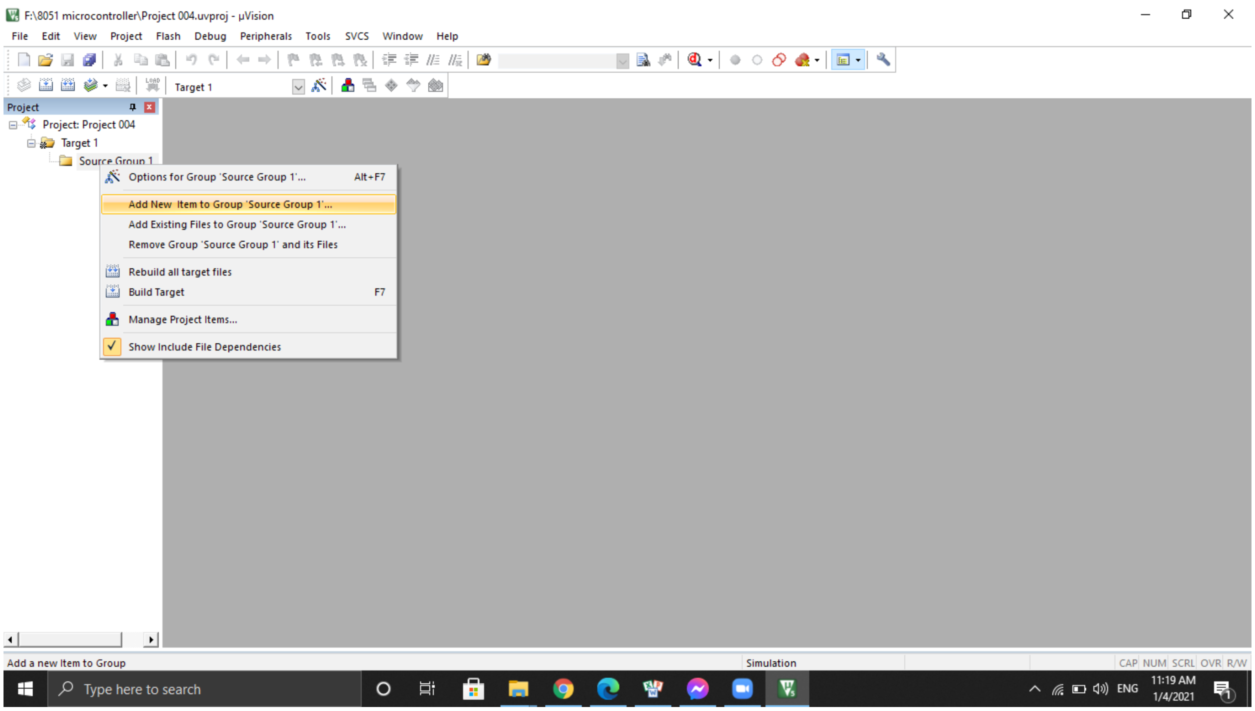

Step 3: I selected device “AT89C51” for target from the following window and pressed “OK”.



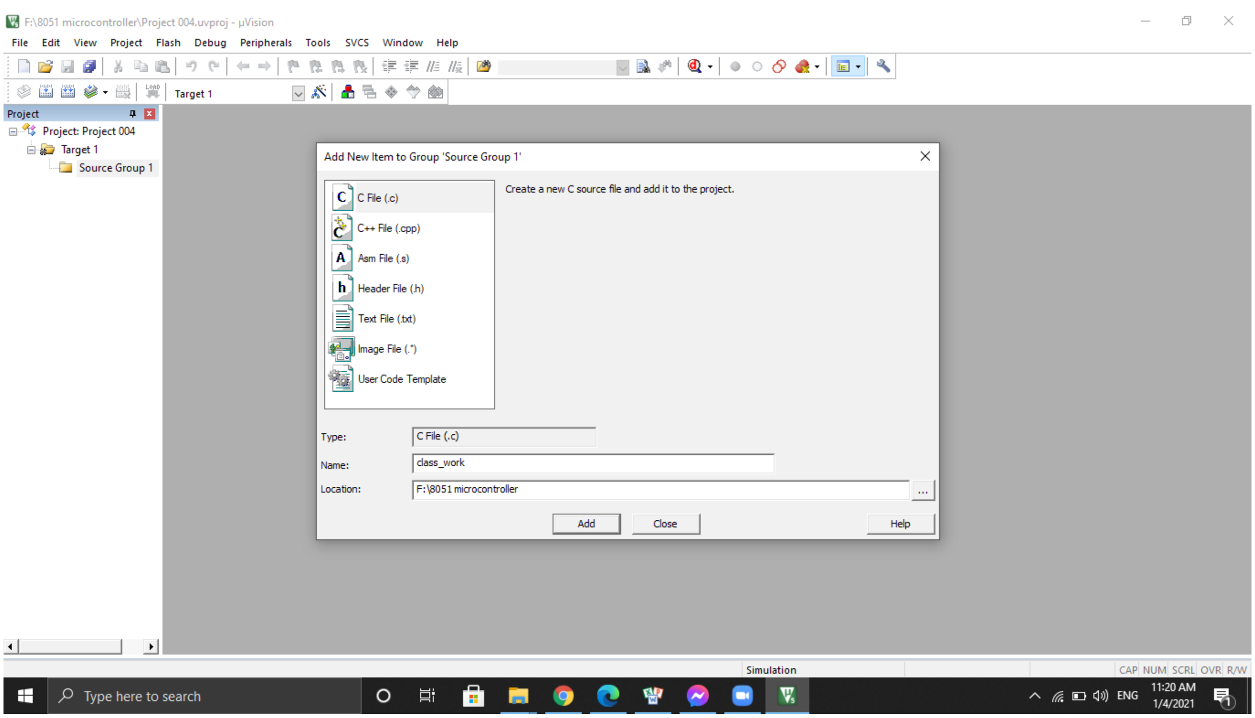
Step 4: The following window appeared and I selected “NO”.



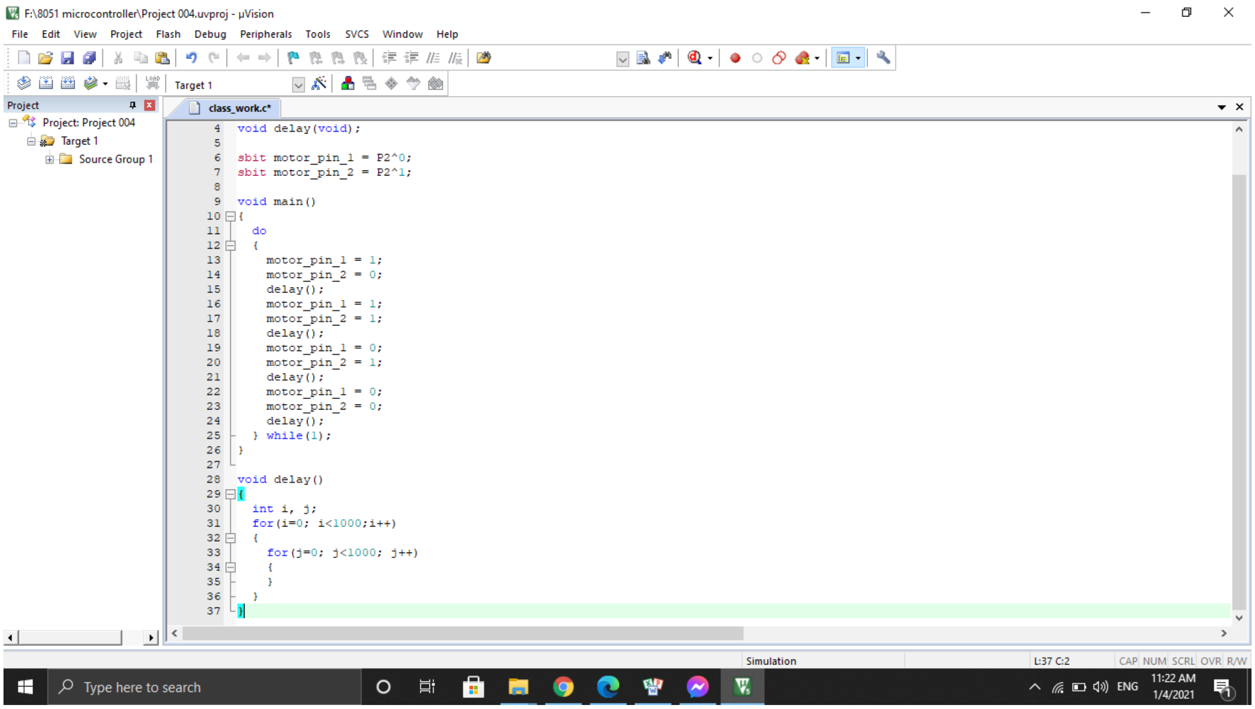

Step 5: Then from the appeared window I opened “Source Group 1” from “Target 1” and made a right click on “Source Group 1” and selected “Add New item to Group ‘Source Group 1’”.

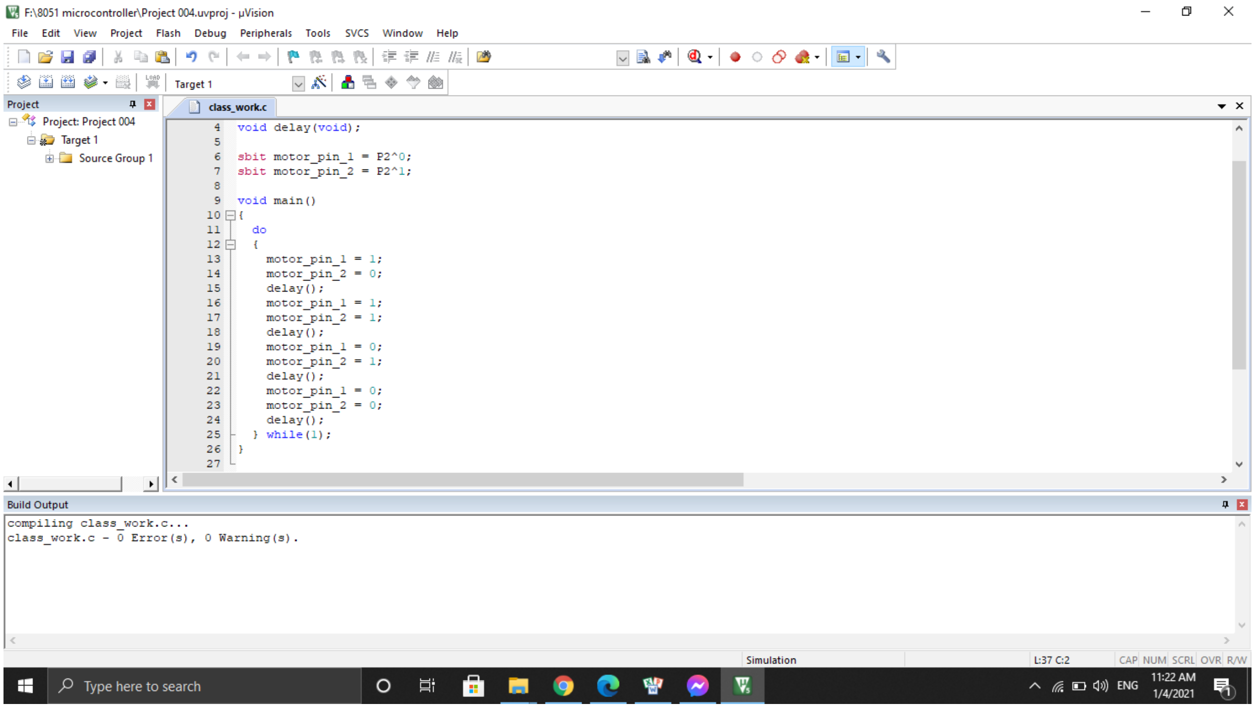



Step 6: The following window appeared and I selected “C File (c)” and named the file “class_work” and clicked “ADD”.

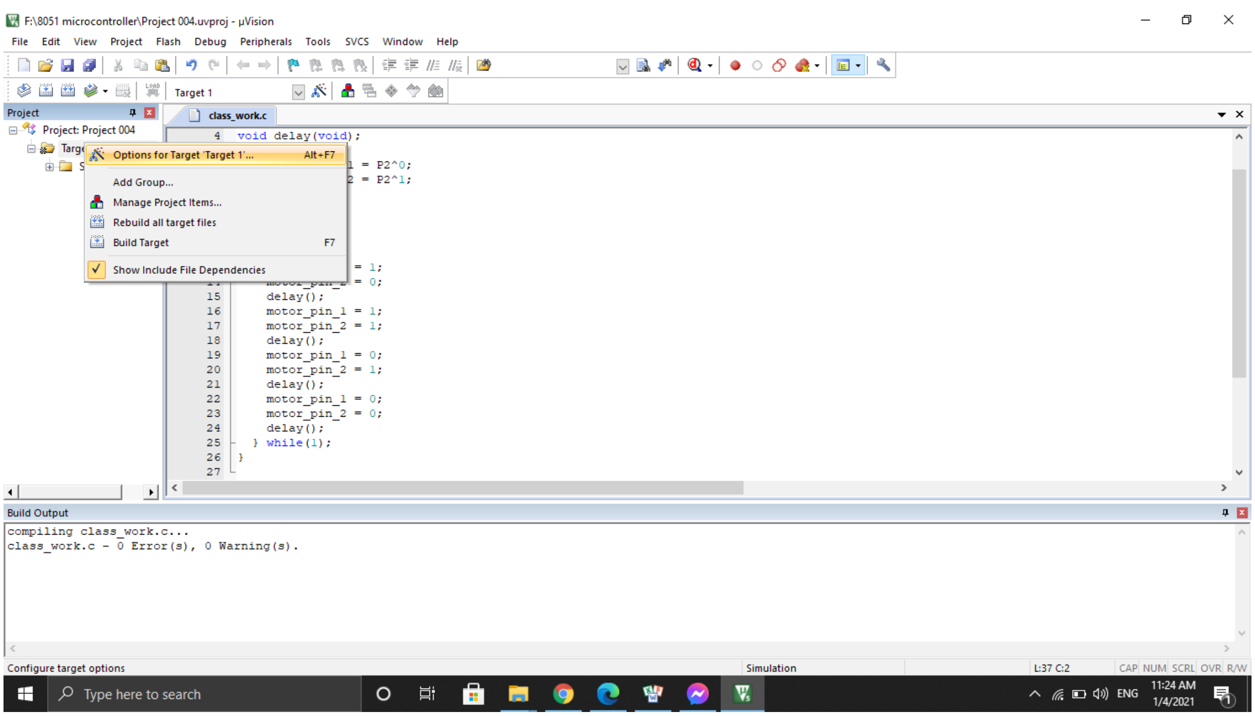


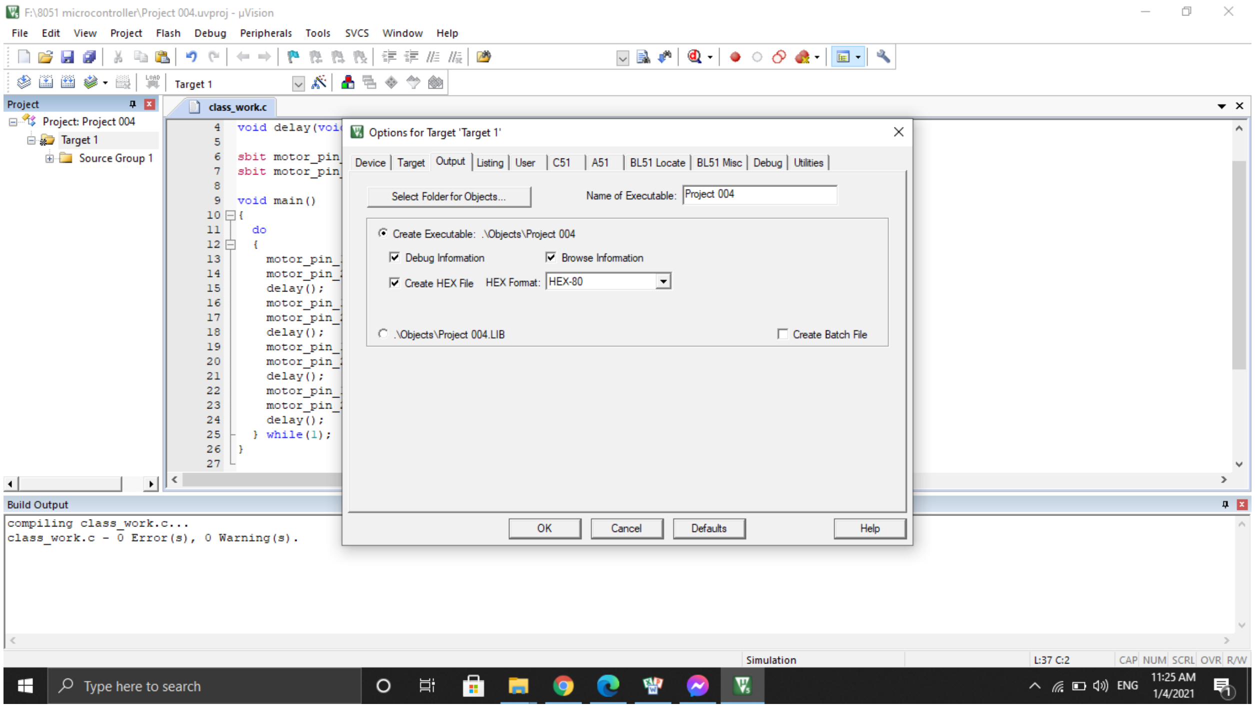
Step 7: The window of the c file created in the previous step appeared. Here I wrote the source code. Then I built the target and got the following result in “Build Output” shell giving “0 errors 0 warnings”.





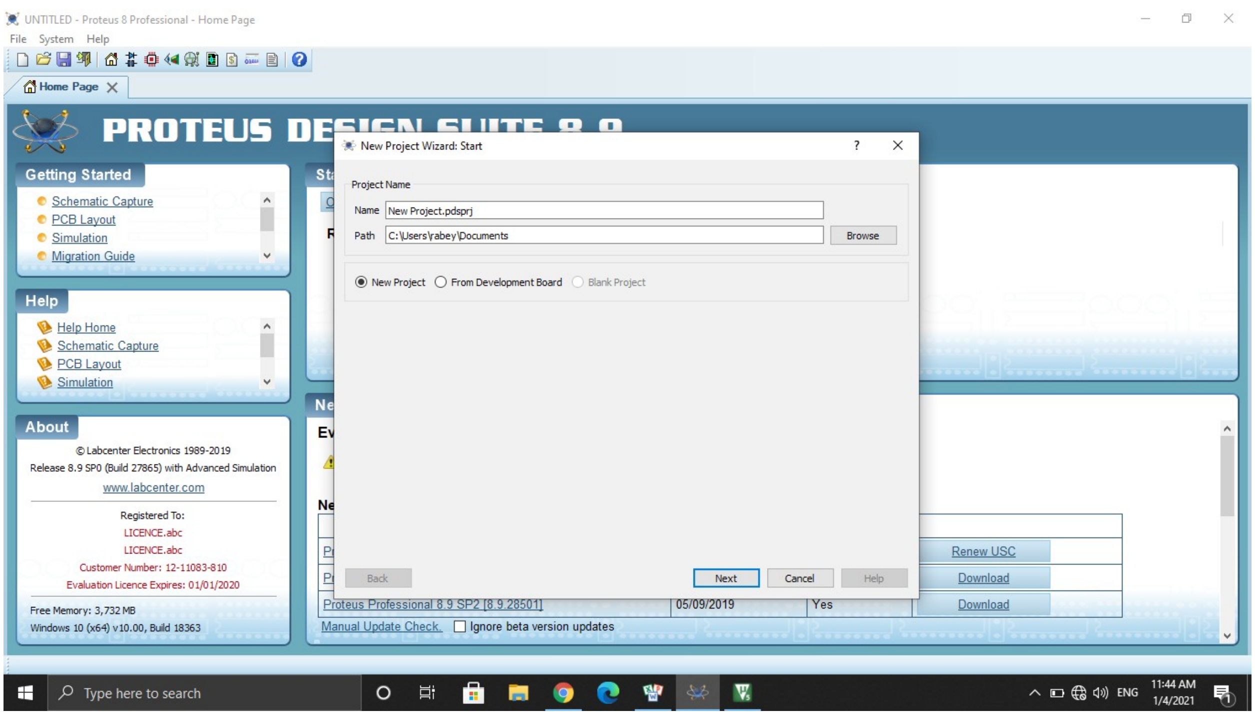

Step 8: I right clicked on “Target 1” and selected “Options for Target ‘Target 1’” from the options. Then the window appeared showing several options. I clicked on the “Output” option and selected “Create hex File”. Thus the hex file of the project was created which goes in the ‘Objects’ of the project file.



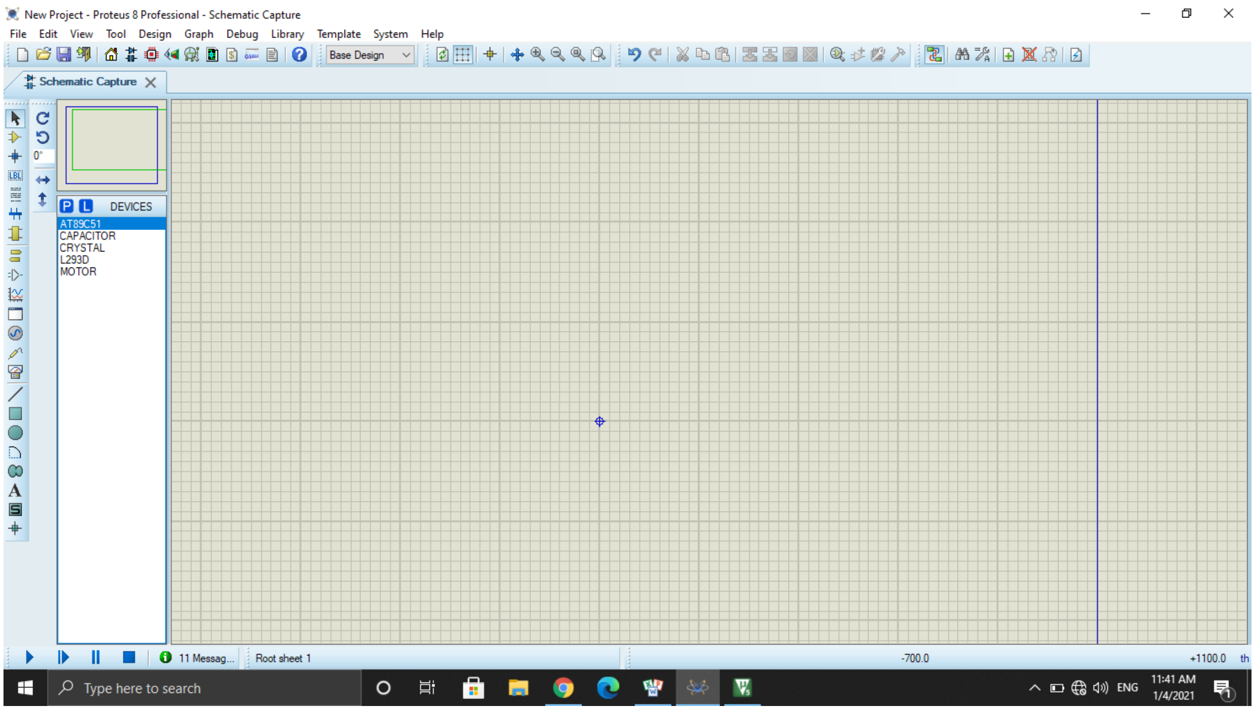



Proteus 8 professional part: -

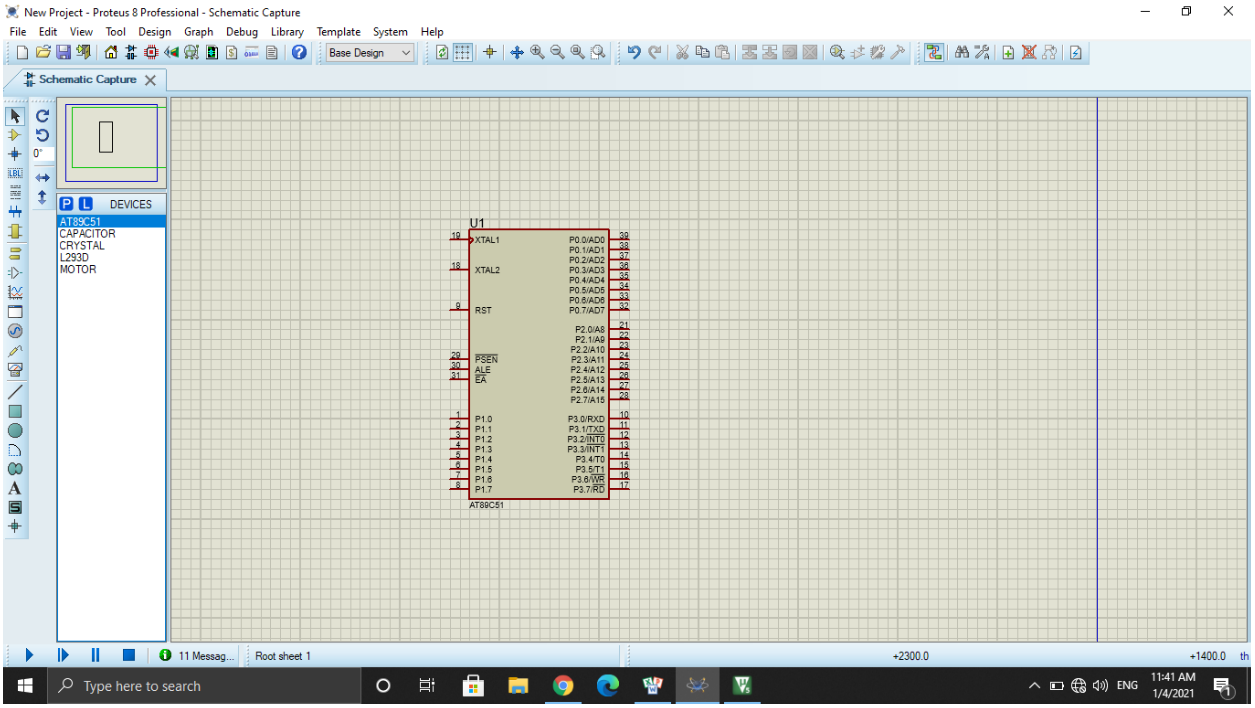
Step 1: I Opened ISIS software and opened a new project and saved it as “new project”.

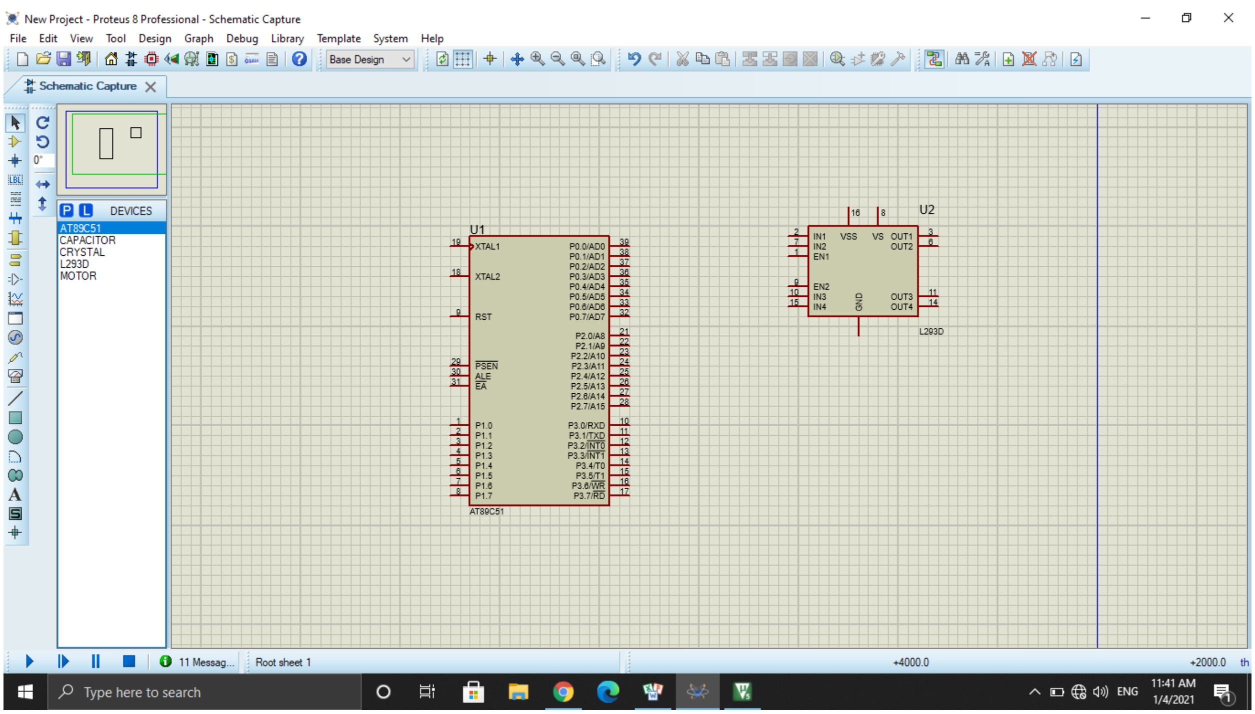


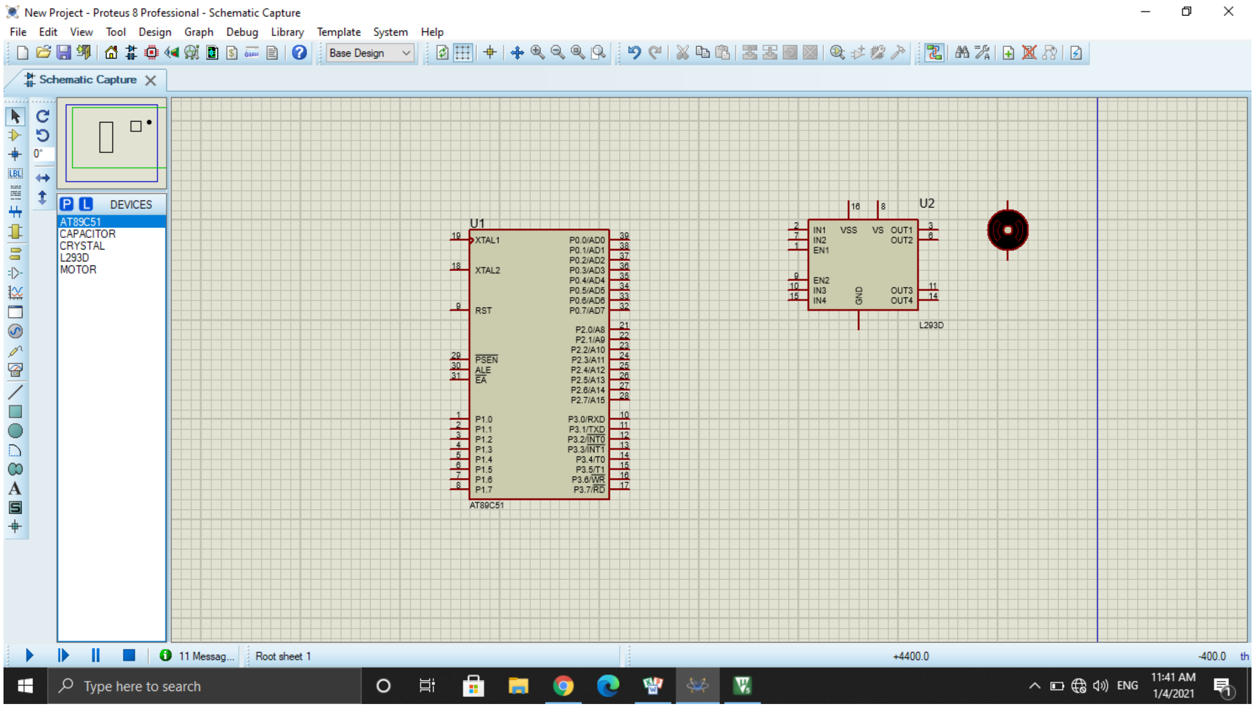
Step 2: I clicked on Pick from Libraries. It showed the categories of components available and a search option to enter the part name. Then I picked devices- AT89C51, L293D, MOTOR DC, CRYSTAL and CAPACITOR 33PF.

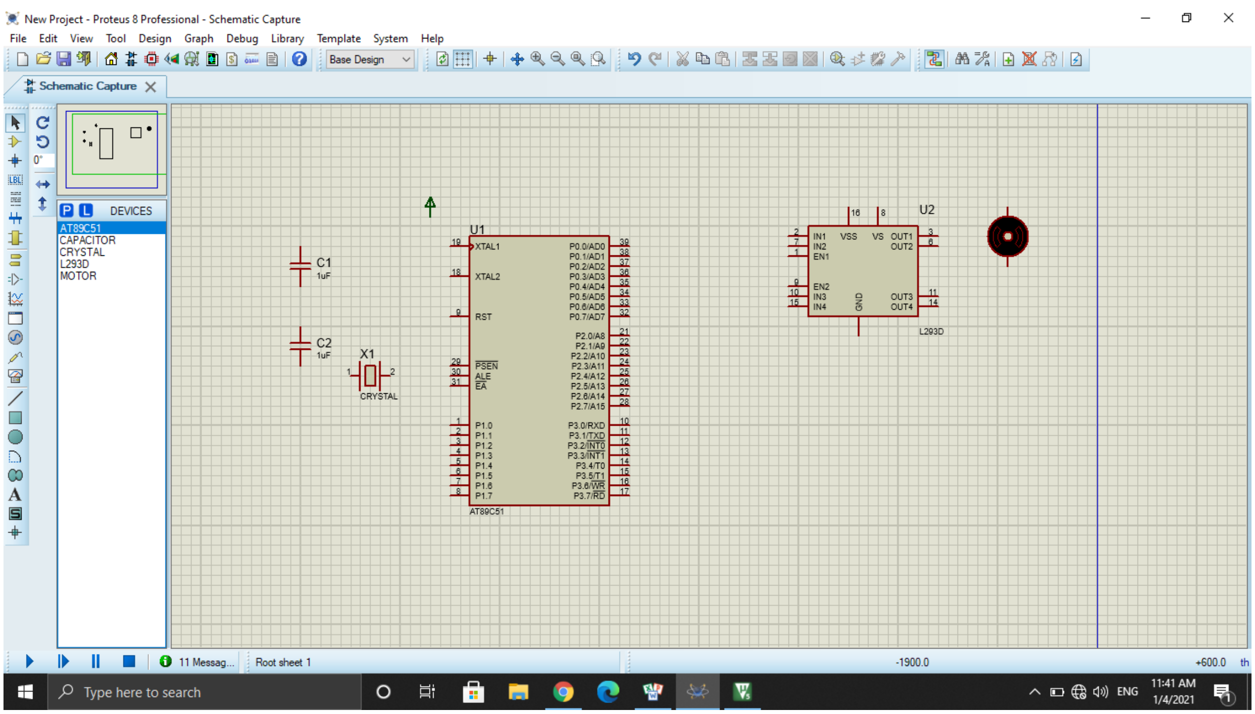


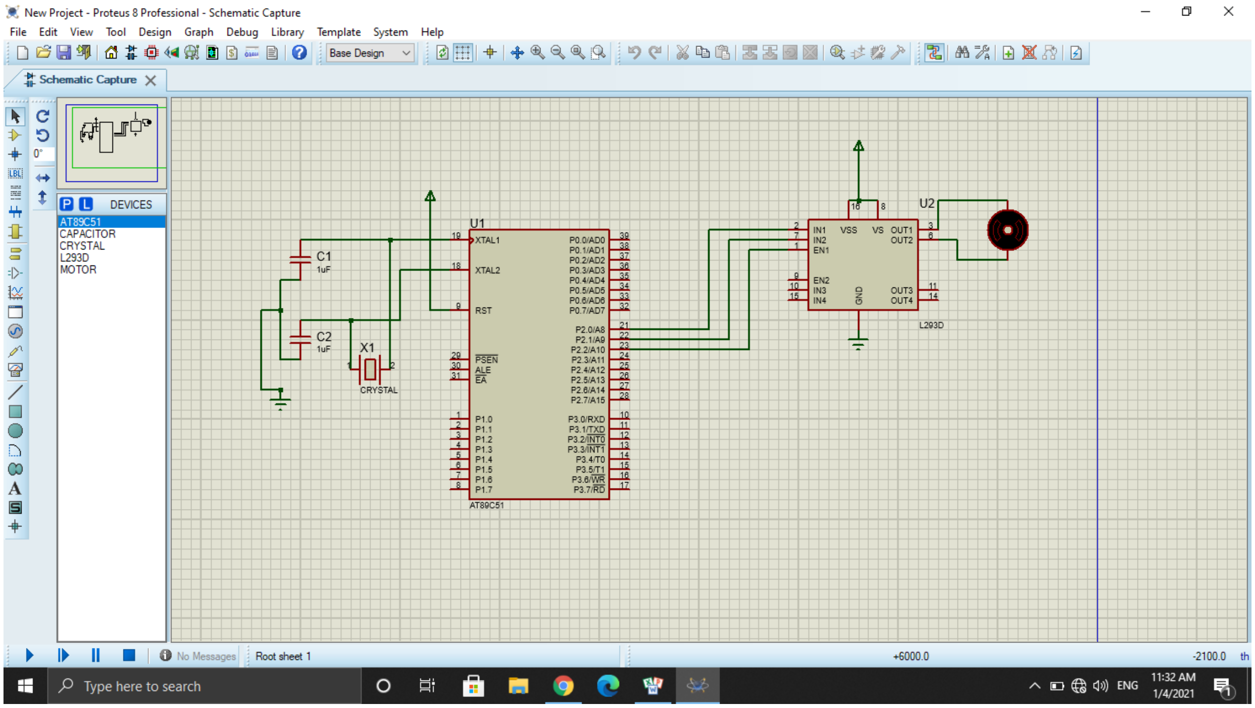

Step 3: The selected components appeared in the devices list. I selected the component and placed it in the design sheet by left-click.


New Project - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
AT89C51
CAPACITOR
CRYSTAL
L293D
MOTOR
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
AT89C51
11 Messag...
Root sheet 1
+2300.0
+1400.0
Type here to search
ENG
11:41 AM
1/4/2021

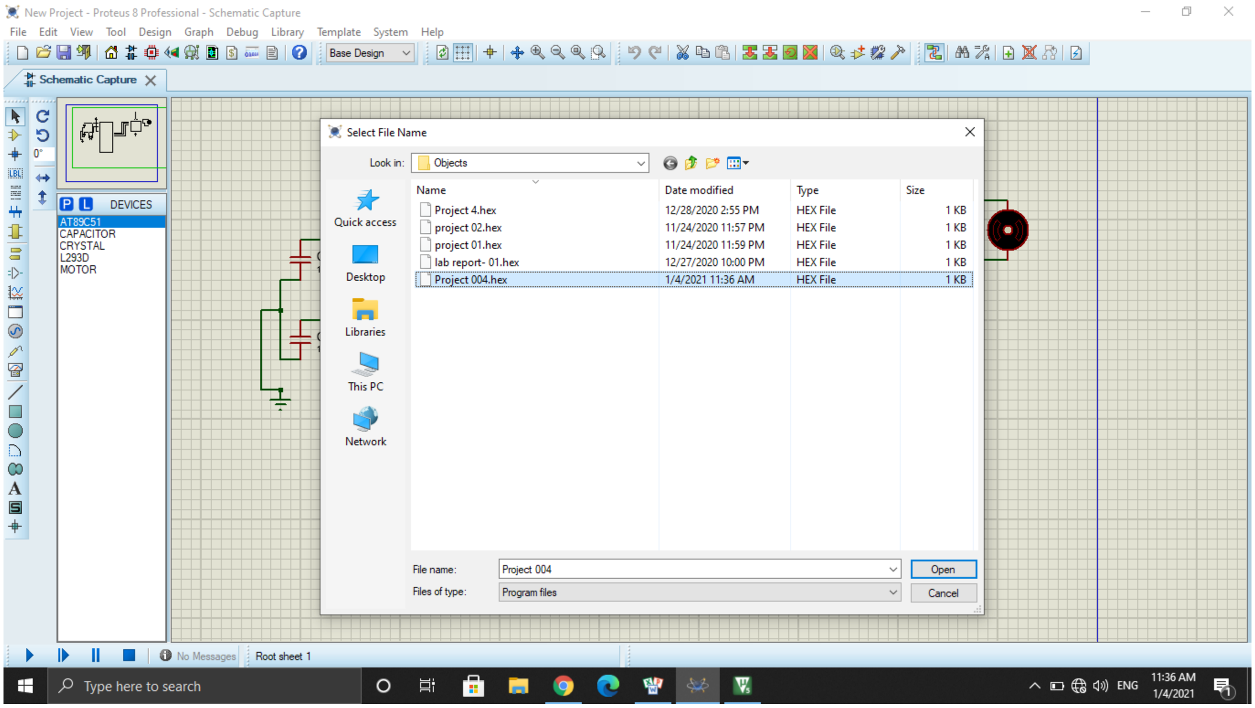

New Project - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
AT89C51
CAPACITOR
CRYSTAL
L293D
MOTOR
U1
XTAL1
XTAL2
RST
PSEN
ALE
EA
AT89C51
U2
IN1
IN2
EN1
EN2
IN3
IN4
VSS
VS
OUT1
OUT2
OUT3
OUT4
GND
L293D
11 Messag...
Root sheet 1
+4000.0
+2000.0
Type here to search
ENG
11:41 AM
1/4/2021


New Project - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
AT89C51
CAPACITOR
CRYSTAL
L293D
MOTOR
U1
AT89C51
U2
L293D
11 Messag...
Root sheet 1
+4400.0
-400.0
Type here to search
11:41 AM
1/4/2021

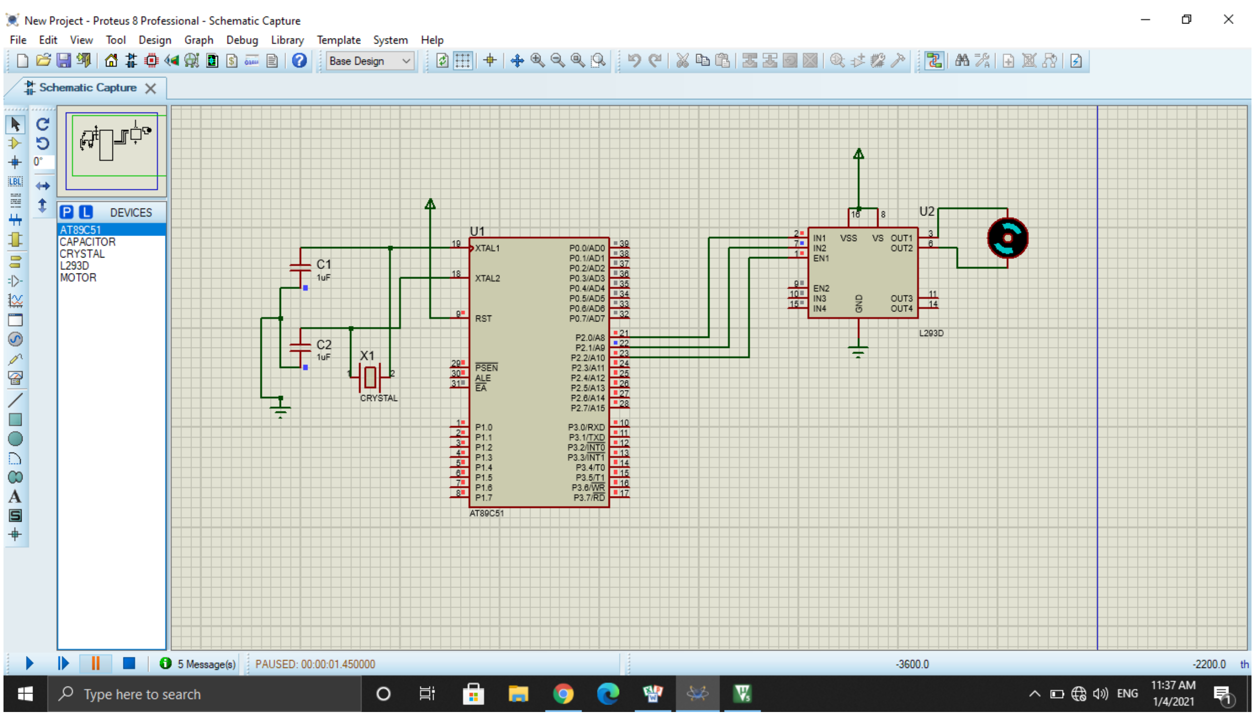

New Project - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
DEVICES
AT89C51
CAPACITOR
CRYSTAL
L293D
MOTOR
C1
1uF
C2
1uF
X1
CRYSTAL
U1
AT89C51
U2
L293D
11 Messag...
Root sheet 1
-1900.0
+600.0
Type here to search
11:41 AM
1/4/2021



Step 4: By double clicking on the AT89C51 device the following dialogue box appeared and I put the hex file of the project in “Program File”.



Step 5: When I run the program I got successful simulation. The motor was rotating perfectly.



Source code :

```
#include<reg51.h>
#include<stdio.h>

void delay(void);

sbit motor_pin_1 = P2^0;
sbit motor_pin_2 = P2^1;

void main()
{
	do
	{
		motor_pin_1 = 1;
		motor_pin_2 = 0;
		delay();
		motor_pin_1 = 1;
		motor_pin_2 = 1;
		delay();
		motor_pin_1 = 0;
		motor_pin_2 = 1;
		delay();
		motor_pin_1 = 0;
		motor_pin_2 = 0;
		delay();
	} while(1);
}

void delay()
```

```
{
	int i, j;
	for(i=0; i<1000;i++)
	{
		for(j=0; j<1000; j++)
		{
		}
	}
}
```

Conclusion:

A simple function for rotating a motor was implemented in the experiment. The experiment ran as expected and the motor changed the direction at a delay. So, this experiment can be considered as successful.