

ENERGY EFFICIENT SMART BUILDING DEVICES FOR HOME SECURITY APPLICATION USING GSM

AIM:

Design and development of Energy Efficient Smart Building Devices for Home security Application using GSM.

PURPOSE:

In modern times everything becomes smart. Maintenance of security also very important for homes and offices. There are some electronic system which can facilitate building management and security separately. Here we have both in one system. The proposed system is Energy efficient smart building devices for home security application using GSM.

DESCRIPTION:

This project includes GSM (Sim800C) module, which is connected to Arduino through UART interface. Temperature Sensor (LM35) and Light Sensor (LDR) connected to Arduino digital IO pins. DC Fan and LEDs connected to relays. Two IR sensors connected to Arduino to monitor visitors count. Buzzer will be connected to Arduino digital pin.

WORKING:

Here two IR sensors used to count number of persons. One IR sensor counts at entry side and another IR sensor counts at exit side. When Temperature sensor detects abnormal temperature more than 50 degrees, Fan will be ON. In LDR Dark condition LED will be ON and in Light condition LED will be OFF. All sensors status will be displayed on LCD. Also SMS will send to registered mobile number. Buzzer will be ON when temperature more than desired value.

TECHNICAL SPECIFICATIONS:

HARDWARE:

| | | |
|--------------------|---|-------------------|
| Microcontroller | : | Arduino Uno |
| Crystal | : | 16 MHz |
| LCD | : | 16X2 LCD |
| GSM | : | SIM800C |
| Temperature Sensor | : | LM35 |
| Light Sensor | : | LDR |
| Counting Sensors | : | IR |
| Power Source | : | 12v 2 amp Adaptor |

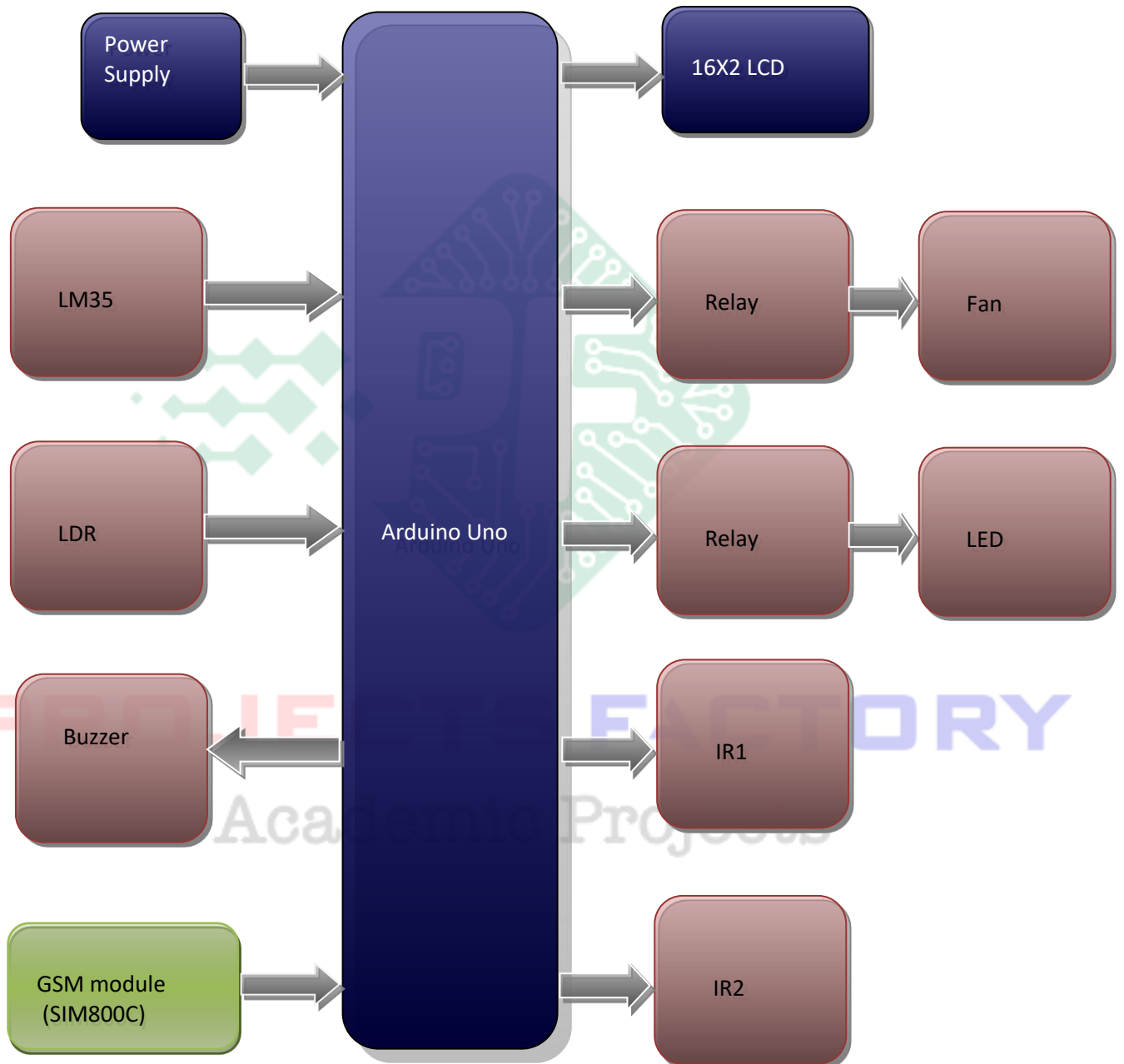
SOFTWARE:

Arduino IDE
Proteus based circuit diagram

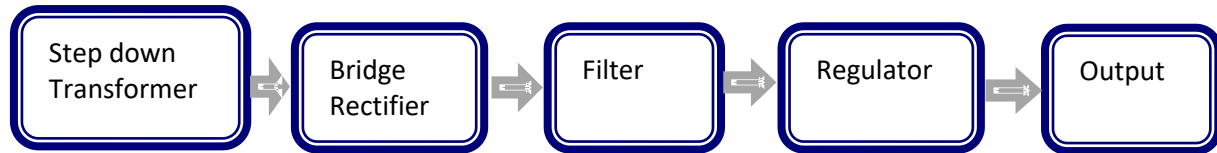
APPLICATIONS:

- Smart Buildings
- Smart Cities
- Security Systems

BLOCK DIAGRAM:



POWER SUPPLY BLOCKDIAGRAM:



INTERFACES COVERED:

- We have covered GSM (SIM800C) module interfacing
- IR, Lm35 and LDR sensors interfacing



PROJECTS FACTORY
Academic Projects