HOW TO CONTROL HOME ELECTRICAL EQUIPMENT USING INTERNET TECHNOLOGIES

POPA SORIN EUGEN[1]

[1] "Vasile Alecsandri" University of Bacau, Calea Marasesti 157, Bacau, 600115, Romania

Abstract: The paper presents the procedure for building a system based on Arduino and various Internet technologies to succeed remotely controlled power supply to some equipment in the house. The paper used an Arduino Ethernet development board, a board with two relays oldest being adapted to the control. The paper presents the algorithm used for processing online data taken from the Arduino board.

Keywords: Arduino Ethernet, web control, Internet, ajax, javascript

1. INTRODUCTION

This paper presents a system for remote control via a web page, the two relays that will feed different consumers. For this, the system must have the following characteristics: to connect to the Internet, order some digital relays, enabling the reading of digital and / or analog, and their values to be displayed online.

In our day there is the development of applications made by users to automate electrical equipment in the house. Basically, using a microcontroller, this connects various sensors that indicate the status of certain parameters and actuators, actuators or relays that operate some electrical appliances.

A large scale took an Arduino-based automation systems and Internet technologies. Plan is to create systems that allow control and monitoring of remote housing.

To achieve such a system automation we need a microcontroller, which is the brain of the automation system, an Internet connection enabling remote access with mobile devices (mobile phone, tablet, laptop), sensors for monitoring various parameters of the house (electric current sensors, voltage, gas, light, human presence, etc.) and actuators to trigger things: relay for power supply to washing machine, electrical systems heating, air conditioning, and others.

When designing such a system we must have the following aspects:cost scăzut: fiecare senzor trebuie să aibă un preţ scăzut;

- Flexibility: Arduino based projects allow anyone to extend it to cover the particular needs of any recipient;
- Reliability: these development boards are very reliable, have a long service life, many built sensors specifically for Arduino libraries including related software;
- The possibility of double feeds: so the electricity network of 220 V and battery.

2. EXPERIMENTAL SETUP

2.1. Hardware description

The device contains an Arduino Ethernet plate with two intermediate relays 12V DC, a senzot PIR, ultrasonic distance sensor, LED. All components are mounted in the housing of a source computer, in which case only were used, the fan and the power socket 220V.

Because Arduino does not have a real time clock was installed a module RTC (real time clock) based on the chip DS1307, communicating with Arduino using the I2C protocol, and is powered by an external battery CR1225 which ensures a long service about 5 years.

Relay board contains two relays 24VDC, DPDT, 5A, V23154-D0721-F104 powered from 12VDC through the one transistor BD137. Board relay is shown in Figure 12. The value of the coil resistance measured is 915Ω, so power consumption will be I = U / R = 13 mA current transistor successfully supported BD 137. The transistor is a bipolar transistor NPN BD137, power, used to boost the audio signal or switching function, as is our case. The maximum current of 1.5A is supported at a maximum voltage of 45V, according to [4].

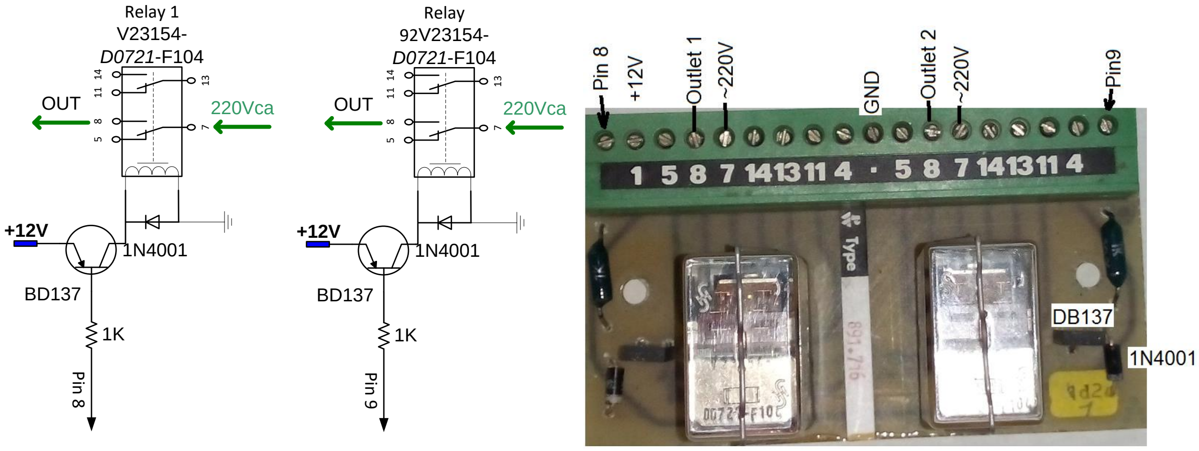


Fig.1. Power and control wiring diagram of the relays.

For further developments, they have mounted a PIR sensor and an ultrasonic sensor HCSR04. PIR sensor has three pins, with the following functional roles:

- Vcc - 5VDC supply voltage,
- GND - ground;
- AL - Alarm, here creates a voltage divider powered from 12V DC and a resistance of 10kΩ, as shown in Figure 2.

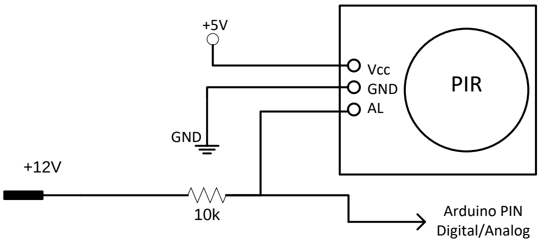


Fig. 2: Connection diagram for PIR sensor.

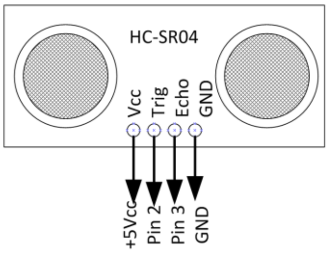


Fig. 3: Connection diagram for ultrasonic sensor

Figure 3 shows how to connect ultrasonic sensor HC-SR04 to Arduino board.

3. RESULTS AND DISCUSSION

The facility allows the command through a web interface of two relays, real-time display of values read from analog ports, displaying the distance measured by ultrasonic distance sensor. Pr The program consists of two parts, namely:

- Arduino microcontroller program that reads the sensors, calculations, LED lights and power transistors control relay coils and periodically send a string to the page index.htm, that is stored on the microSD card. Arduino Ethernet web server has the address 192.168.1.220 and opens the file index.htm call on microSD.

- Index.htm file that retrieves data sent by the microcontroller and displays them on the website and takes orders data page by the two buttons on the page.

3.1. Program Description of the Arduino
The program is divided into 3 main functional blocks, as shown in figure 4, notably:
1. definitions block and inclusion of libraries used;
2. Initial settings block;
3. The loop block, reading values and calculations

In Definitions block are included libraries for Ethernet, SD and SPI communication.
Defined parameters, their default / original and description are given in the following table:

Table 1: The elements used in the block definition program.

No.	*Parameter name*	*Default value*	*functional Description*
1	REQ_BUF_SZ	0	The size of the buffer used to capture http requests
2	trigPin	2	Used by ultrasonic distance sensor
3	echoPin	3	Used by ultrasonic distance sensor
4	LED1	6	Used by ultrasonic distance sensor
5	LED2	7	Used by ultrasonic distance sensor
6	HTTP_req[REQ_BUF_SZ]	0	Buffering HTTP requests
7	req_index	0	Index pentru bufferul HTTP_req
8	RELAY_state[2] = {0}	0	Stores relays state
9	distance	Int	Stores distance measured by the ultrasonic sensor.
10	PIRStare	LOW	Store PIR sensor state
11	previousMillis	0	
12	interval	500	
13	alarmValue	0	
14	Adresa MAC	DE-AD-BE-EF-FE-ED	Arduino Ethernet card's MAC address
15	Adresa IP	192.168.1.220	Arduino Ethernet card's IP address
16	Server port	80	

Table 2: How to use Arduino pins

Digital pin no.	IN/OUT	Description
0	IN	Rx - Front serial communication with PC
1	AUT	Tx - Transmission serial communication with PC
2	OUT	Trig ultrasonic distance sensor
3	IN	Echo for ultrasonic sensor
4	OUT	SD card
5	OUT	PIR sensor LED
6	OUT	ultrasonic distance sensor LED1
7	OUT	ultrasonic distance sensor LED2
8	OUT	Relay 1 control
9	OUT	Relay 2 control
10÷13		Reserved for Ethernet communication
Analogic pin no.		
0	IN	PIR sensor
1÷5	IN	unused

The table 2 shows how to use the Arduino Ethernet Pin.

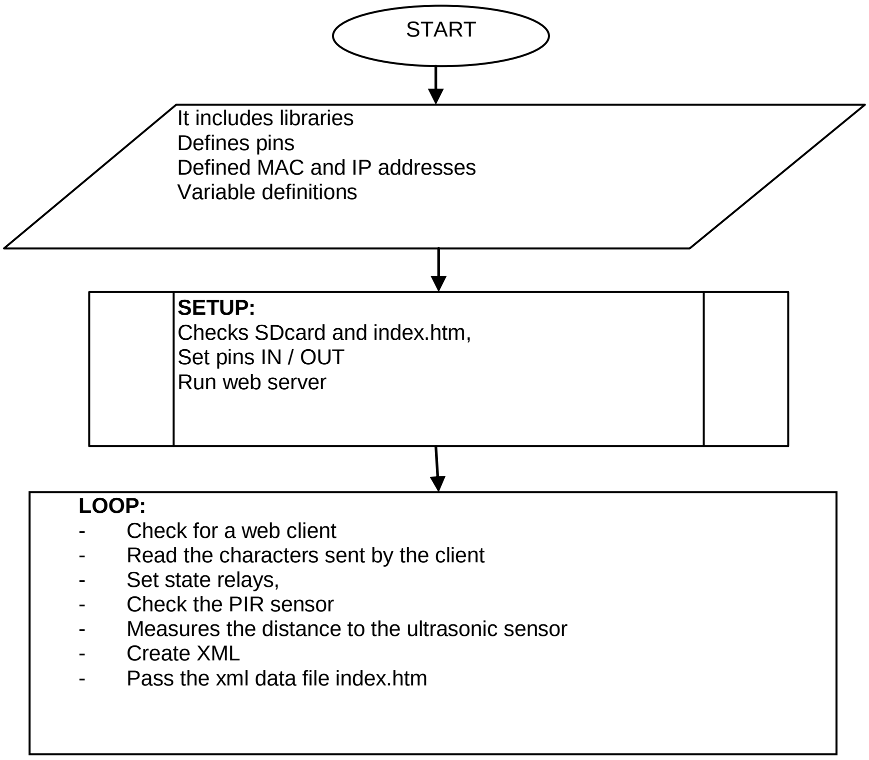
Blocul setup()
The following is an explanation of setup code block ().Se oprește chipul Ethernet;

- Checking for microSD card and index.htm file;
- Set the digital pins status according to table 2;
- Start Ethernet device;
- Start web server and expect customers.

Blocul loop()

Attempting to gain an Ethernet client, if any:

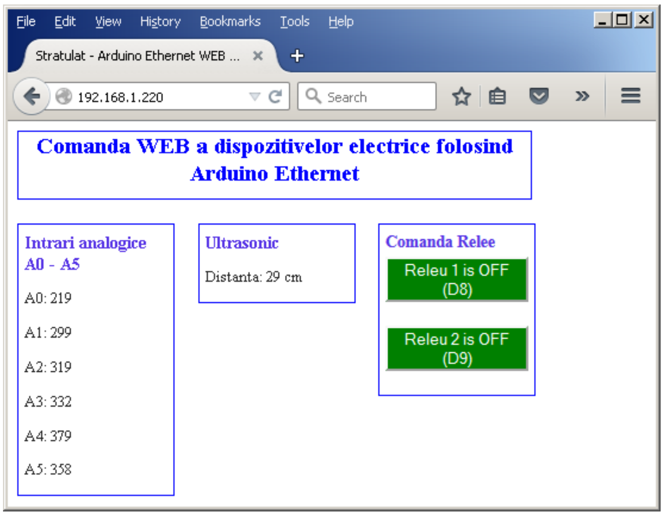
- If the client is available, read a character (byte) from the client (read character by character string data all received and saved in a buffer, taking care not to exceed the buffer size);
- Check if the client requests a web page or an XML page;
- If the client requests a page xml (ajax_inputs) then launches web server functions SetRELAYs (), (PIR) and ultrasonic () that sets the relay state after the command received from the web client and read status PIR sensor and ultrasonic sensor calculates the distance;
- Answering function XML_response web client (client) that retrieves data calculated / measured / set above and arranges them in a format that can be played by HMT index.htm file..
- Data is going to index.htm file stored on the microSD card.


START
It includes libraries
Defines pins
Defined MAC and IP addresses
Variable definitions
SETUP:
Checks SDcard and index.htm,
Set pins IN / OUT
Run web server
LOOP:
- Check for a web client
- Read the characters sent by the client
- Set state relays,
- Check the PIR sensor
- Measures the distance to the ultrasonic sensor
- Create XML
- Pass the xml data file index.htm

index.htm file

Figure 5 shows the index.htm page, which monitors and controls the Arduino Ethernet.
This page is accessed from the local network, using Arduino Ethernet address, namely: 192.168.1.220.
Structurally, the page has three div's (work areas), namely:

- • The display of quantities read the analog ports as only the analog port A0 we PIR sensor attached only to this port has shown the significance, the rest to other ports values shown are random and have no physical significance. When feels PIR motion sensor, the value indicated by A0 tend to 0, otherwise it is more than 300 - this is due to the resistance value variation mounted in the circuit and the supply voltage - power supply voltage has stabilized;
- The display of the distance measured by the ultrasonic sensor, value expressed in centimeters;
- The display and control two relays (rather control transistor supplying relays).



Comanda WEB a dispozitivelor electrice folosind Arduino Ethernet

Intrari analogice A0 - A5

A0: 219

A1: 299

A2: 319

A3: 332

A4: 379

A5: 358

Ultrasonic

Distanta: 29 cm

Comanda Relee

Fig. 5: Website monitoring and control Arduino Ethernet.

The logical structure of the web page is:

- Some javascript marked by the <script> and </ script>which makes the following:
  - defines some variables that store: state relays and values: null, on and off;
  - GetArduino function (), which checks for Arduino Ethernet connection and if it receives an xml file from it;
  - Take the values for the tags <analogous> <ULTRASONIC> and <LED> which contains the values read on the analog ports, the distance measured by the ultrasonic sensor, the status of the two relays respectively;
  - GetButton function () acting when you press a button;
- Some of CSS, which is displayed in page format, framed by tags <style> and </ style>;
- The <body> htm web page, which shows information page.

4. CONCLUSIONS

The chosen solution using modern equipment and techniques, and requires a thorough knowledge of the programming of the microcontroller, the programming language C ++ and hardware used. As opportunities to develop / improve the operation of said apparatus:

- Using a hardware platform with a powerful microcontroller, enabling writing more code, thus adding additional functionality of the device - 78% of the current program uses code storage and 75% of dynamic memory allocation variables;
- The use of a temperature sensor placed inside the enclosure in which the Arduino is Ethernet, which actuates the fitted fan to overcome a critical temperature. Starting the fan is manually through a switch.

REFERENCES

[1] http://www.instructables.com/id/Uber-Home-Automation-w-Arduino-Pi/step1/
[2] http://web.ulbsibiu.ro/laurean.bogdan/html/Microcontrolere%20introducere.pdf
[3] WebServer example by David A. Mellis and modified by Tom Igoe
[4] http://www.datasheetcatalog.com/datasheets_pdf/B/D/1/3/BD137.shtml.
[5] SD card examples by David A. Mellis and Tom Igoe
[6] Ethernet library documentation: http://arduino.cc/en/Reference/Ethernet
[7] SD Card library documentation: http://arduino.cc/en/Reference/SD