

# Speech Control Systems for Electrical Technologies

Peter Apse-Apsitis<sup>1</sup>, Ansis Avotins<sup>2</sup>, Leonids Ribickis<sup>3</sup> (<sup>1,2,3</sup>*Riga Technical University*).

**Abstract** – Typically electrical apparatus and devices are controlled via hardware tools - buttons, sliders, and different sensors or by software tools. This article describes authors current development in use of human voice commands, that can be used to control robotic devices, converters, drives, lights etc. Two versions of the possible speech control systems for electrical technologies are further described. Implementation of such control method is an attractive way to use it as education tool for synthesis of Systems Control applications, Programming and other related courses within study program “Computer control of electrical technologies”.

**Keywords** – Converter control, education tool, lighting, motion control, robotics.

## I. INTRODUCTION

Despite of several available speech recognition computer programs like Windows Speech recognition [1], Dragon Naturally Speaking [2], speech data input for electronical device or equipment control still is rare. In the field of electrical technologies, such voice command controls mainly are used by hobbyists, or for special systems, like indoor lighting system controlled with hand clutch, but not more.

In the same time it could be recognized to use voice commands as possible control method for educational purposes or other existing devices and applications, like, three-wheel robot FESTO Robotino, DC drive or even use it for LED lighting and indoor illumination control.

The first attempt to implement voice control on Festo Didactic Robotino was made in 2010 [3]. Speech control system contains several Windows operating system based software modules, and was complicated and not enough reliable.

Further two versions of via voice commands controlled intelligent systems are described - earlier development of LED dimmers, remote controlled e-Learning workshop and others:

- first control method is based on Apple OS X software and hardware and Arduino family modules;
- second method is based on signal processing module EasyVR [4] and Arduino family modules;

Voice recognition started about 100 years ago when first telephone was designed and developed. On that time the main task was to reduce voice transmission bandwidth. Main developments in speech and voice recognition area started with digital systems development (computers) in 1960s. The main task in speech recognition area is to recognize individual human voice and speech and digitally transfer it to the written word. Usually both tasks are very close and there are no strict

borders between voice and speech recognition software or hardware.

A negative aspect is necessity to adapt user voice to speech recognition system in order to perform commands correctly. The process takes several minutes and each user voice profile is stored into software, for example - Dragon Dictate.

## II. STRUCTURE OF SYSTEMS

The further description of both voice control systems are designed for English language words. Several other wide spoken languages are also available, but Latvian language, unfortunately still is under development.

### A. OS X software and Arduino modules based system

The functional diagram of this control system is shown in Fig.1. System includes microphone, computer (Apple) running OS X 10.6 operating system, Dragon Dictate speech recognition software and FileMaker Pro 12 database software. It is also easy to use other Apple products, like iPhone in wireless microphone mode.

Speech recognition software Dragon Dictate converts spoken word into characters in FileMaker database text field. Then text field content is filtered, analyzed and converted as commands by FileMaker database software.

After analyzing process, commands are sent as a request to Arduino Ethernet module (board) via Internet. Arduino Board acts as Web server with fixed IP address. As it is connected to Arduino microprocessor board, the requests from FileMaker to Web server then are transformed to electric drives, control systems, actuators or other device control signals.

For example, spoken words “light 50” are converted to digital command to set LED lighting illumination to 50%. More complicated commands, like “100 steps forward, speed 50 second”, can be done, where mentioned command forces the step motor to perform 100 steps in forward direction with speed 50 steps per second.

Command structure still must be pre-defined and stored in FileMaker database, in this way each user can have own command profile for the same task. Profile is bounded to FileMaker Username, so no additional steps are necessary to find exact profile.

The proposed speech command system can be implemented into Energy Monitoring System [5] due to FileMaker application and similar structure in general.

### B. Arduino code

The full Arduino code to control via internet two RC servo motors and receive 3-axis accelerometer, readings form is given below. Fig. 2 shows the main algorithm of the Code.

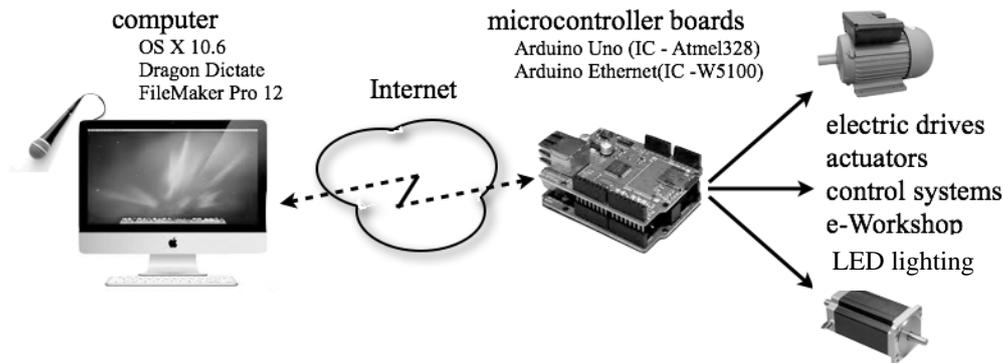


Fig. 1. Functional diagram of OS X based system

Two servo motors allow accelerometer positioning in to different angles against coordinate system X, Y, and Z axis. Display screen shoot is shown on Fig.3. [6].

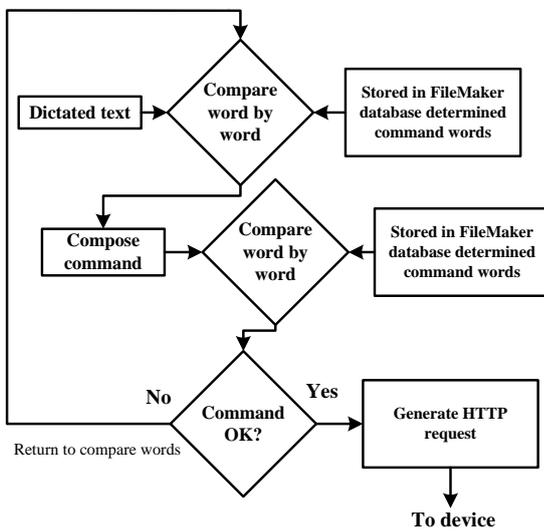


Fig. 2. Algorithm of the FileMaker program Code.

Main algorithm of the FileMaker program code is shown in Fig.2., where 3 main cycles/steps must be done before generating HTTP request and execute the command given by voice. Thus the first step is to compare word or words from recorded voice (dictated text) with predefined words in FileMaker database. The main task of this step is to convert recorded audio signal into a digital text, to be processed in next steps. When the words are recognized, then a next step can be realized – the composition of command. The recognized words now can be compared with command words that are predefined and saved in FileMaker database. The next step then evaluates the command, if command and command words match with predefined command in FileMaker database, then it is possible to generate HTTP request, else the command must be dictated again, and all cycle repeats, until it is recognized. The generated HTTP request is sent to the device – communication board, for example – Arduino Ethernet board, which is connected to microcontroller board - Arduino Uno or any board created and programmed by

student. Then the actuators can be controlled and the devices like electric drives, LED lighting, etc. can be controlled by voice. HTTP request is generated by FileMaker as standard HTTP request and include necessary commands. Commands correspond to dictated text in FileMaker text field. Further main programming code examples are described.

When HTTP request is received, the request is processed to find command characters (A, B, S, L) and values (servoValueA, servoValueB):

```

while (client.connected())
{
  if (client.available())
  {
    char c = client.read();//read char by char
    from HTTP request
    if (readString.length()<30)
    {
      readString +=c; //store characters to string
    }
    if (c=='\n') //if HTTP request has ended
    {
      pos = readString.indexOf('=');
      if( pos > 0)
      {
        simti=readString.charAt(pos+1)-48;
        desmiti=readString.charAt(pos+2)-48;
        vieni=readString.charAt(pos+3)-48;
        value=0;
        value=100*simti+10*desmiti+vieni;
        if(readString.charAt(pos-1)=='L')
        {
          ledValue=value;
          analogWrite(ledPin, ledValue);
        }
        if(readString.charAt(pos-1)=='S')
        {
          if(readString.charAt(pos-2)=='A')
          {
            servoValueA = value*10;
          }
          if(readString.charAt(pos-2) == 'B')
          {
            servoValueB = value*10;
          }
        }
      }
    }
  }
}

```

After command execution, read data from accelerometer and average between 3 readings:

```
X=0;
Y=0;
Z=0;
for(i=0;i<3;i++)
{
  X += analogRead(A0);
  Y += analogRead(A1);
  Z += analogRead(A2);
  delay(2);
}
X=X/3;
Y=Y/3;
Z=Z/3;
```

Readings are send to FileMaker as standart HTML code:

```
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println();
client.print("<body> <head> </head>");
client.print("LED= ");
client.print(ledValue);
-----
client.stop();
```

Servos are controlled by sending 1-2ms long control impulse every 20ms:

```
if(millis()-timeA >20)
{
  digitalWrite(servoPinA, HIGH);
  delayMicroseconds(servoValueA);
  digitalWrite(servoPinA, LOW);
  timeA = millis();
}
if(millis()-timeB > 20)
{
  digitalWrite(servoPinB, HIGH);
  delayMicroseconds(servoValueB);
  digitalWrite(servoPinB, LOW);
  timeB = millis();
}
```

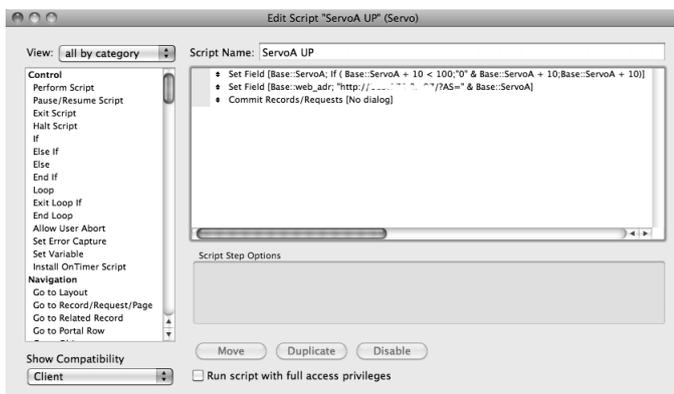


Fig. 3. FileMaker Scripting window and script to perform HTTP request.

FileMaker Script to perform HTTP request to Arduino Ethernet board via integrated FileMaker web browser are shown on screenshot (Fig.3.).

FileMaker scripting is similar to any programming language scripting or AppleScript. FileMaker Scripts operate with database fields and field content, layout Objects and object behavior. Any FileMaker supported mathematic operation or logic equitation easily can be included into FileMaker Script.

The written code is one example of possible variations of the speech - practically any set of words and any length of commands can be described and implemented. Command count is limited only by computer (running FileMaker) hardware (RAM size, disk size) limitations - practically unlimited.

Below are some examples for better explanation:

- received command “L=139” (in ASCII binary form) define LED illumination level via PWM to level 139/255 (PWM 100% = 255) can be decoded from voice command (before converting to digital form) “Light set one three nine go”;
- received command “SA=150” set first servo motor angle and corresponding voice command before converting are: “Servo A set one five zero go”;
- “SB=160” set second servo motor angle and corresponding voice command before converting are: “Servo A set one five zero go”.

Last word “go” determines end of the command. There can be any other word. When word “go” is detected, FileMaker software compiles necessary HTTP request to be sent to Arduino based Web server. FileMaker processes received reply from Arduino Web server, process received data and displays it if necessary.

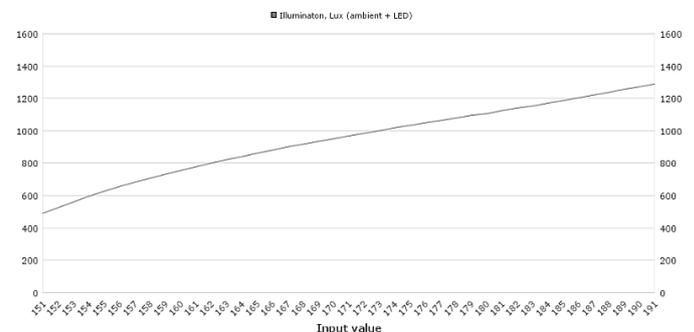


Fig. 4. Code input value vs LED illumination level in Lx.

Authors recently developed LED luminary dimmer allows controlling illumination digitally (Fig.4.). To control several LED lamp illumination in area, speech command can be composed like “Lamp three level five five go” to set lamp No.3 in numbered lamp network to illumination level 55% from current illumination level. Before changing lamp No.3 dimming value, FileMaker software send request to LED lamp No.3 to receive ambient light level. After receiving ambient light level, necessary dimming value according to graph is calculated and according code is sent to LED dimmer.

Also other dimming algorithms can be implemented according to pre-defined illumination task. Moreover, practically unlimited number of algorithms can be stored into FileMaker database. Well known communication protocols

like ZigBee/XBee, WiFi or radio-wave communication tools can be applied to replace Internet communication if necessary. In this case corresponding Arduino modules must be applied.

### C. Voice Recognition module and Arduino modules based system

System's basic version is local in site. Internet version also is available but not described in this article. Version functional diagram are shown on Fig.5.

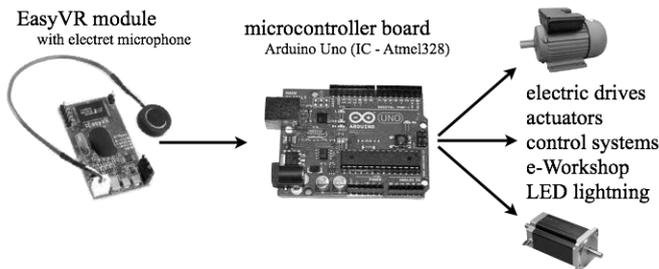


Fig. 5. EasyVR and Arduino board based system.

In general this solution is simple, less complicated and has lower potential comparing to solution described previously. Only 32 user defined command words are available and there is no data feedback available. Serial UART communication between EasyVR and Arduino board allows sending EasyVR recognized words as statements or commands to Arduino. Typical communication baud rate is 9600bps. Other typical UART communication baud rates are up to 115200bps in following frame: 8 Data bits, No parity, 1 Stop bit is available.

### III. CONCLUSIONS

Developed voice control systems can be applied to control different systems and gain the functionality and additional comfort, as well as to perform commands when hands are occupied and even in unsafe conditions or locations, that are potentially dangerous. Possible application of the system in noisy, dangerous, critical environments is not tested. Developed voice control system both versions must be tested by more users under different ambient noise conditions.

Speech localization module must be developed as well as possibility to use inexpensive system elements must be tested.

### REFERENCES

[1] Windows Speech Recognition. [Online]. Available: <http://www.microsoft.com/enable/products/windowsvista/speech.aspx> [Accessed: May 13, 2012].

[2] Dragon Dictation. [Online]. Available: <http://dragondictation.org/> [Accessed: May 13, 2012].  
 [3] Apse-Apsitis P. Voice controlled FESTO Robotino. [Online]. Available: [http://www.youtube.com/watch?v=Blgff14e\\_Fk](http://www.youtube.com/watch?v=Blgff14e_Fk) [Accessed: May 13, 2012].  
 [4] Easy VR user manual. [Online]. Available: [www.veear.eu](http://www.veear.eu) [Accessed: May 13, 2012].  
 [5] Apse-Apsitis P., Avotins A., Ribickis L., Zakis J. Development of Energy Monitoring System for SmartGrid Consumer Application, DoCEIS 2012 Conference Proceedings, Technological Innovation for Value Creation. - 3. Springer (2012) p. 347.-354.  
 [6] Apse-Apsitis P., Avotins A., Krievs O., Ribickis L., Practically Oriented e-Learning Workshop for Knowledge Improvement in Engineering Education, EDUCON 2012 Conference Proceedings.  
 [7] Galkin I., Avotins A., Suzdalenko A., LED dimmer as versatile hardware platform for practical exercises in power electronics and control courses, Power Electronics and Applications (EPE 2011), Proceedings of the 2011-14th European Conference on , vol., no., pp.1-9, Aug. 30 2011-Sept. 1 2011.



**Peteris Apse-Apsitis** is 3<sup>rd</sup> course PhD student, researcher of Riga Technical University Institute of Industrial Electronics and Electrical Engineering. Main fields of scientific interests are power electronics, robotics and mobile ICT applications. He is author of many industrial solutions and applications and patented inventions. A member of IEEE association.



**Ansis Avotins** is 3<sup>rd</sup> course PhD student at the faculty of Power and Electrical engineering study program „Computer Control of Electrical Technology”. Master degree of Science in Electrical Engineering achieved in 2005. Main field of study is autonomous lighting systems. From 2004 working as chief Laboratory Manager at RTU, IEEI. A member of IEEE and ACM associations.



**Leonids Ribickis** is a professor and Rector of the Riga Technical University. Received Dipl.El.Ing. qualification in 1970 (Riga Polytechnic Institute), in 1980 received PhD. In Belorussian Polytechnical Institute and in 1994 – Dr.Habil.Sc.ing. at RTU. Main research fields: Electric Drives, Power Electronics, Intelligent Electronic Equipment. Full member of Latvian Academy of Science, IEEE Latvia section Chairman, Board Member of EPE.