

A reasonable smart home technology on the Arduino

Dmitry Lubianov^a, Kostiantyn Kasian^a, Mykola Kasian^a

^a National university "Zaporizhzhia polytechnic", Zhukovskogo street 64, Zaporizhzhia, 69063, Ukraine

Abstract.

The article demonstrates the results of the development and implementation of low cost smart home technology programmed on the FPGA Arduino. The purpose of the work is to program remote control gadgets, with which the user has the ability to control all devices that support infrared signal technology. The scientific novelty of the work results is the developed set of programmable gadgets can replace a significant number of different remotes and switches by using only one gadget as a control device, which is an alternative to a smart home. The practical value of the work results is that the developed set of programmable gadgets will ease the trivial life of the handicapped and the elderly.

Keywords 1

Android, Arduino, FPGA, IR-signal, Remote control, Smart home

1. Introduction

Nowadays, concepts like a smart vacuum cleaner, a smart refrigerator, and in general, a smart home is not something fantastic. Smart devices are approaching to the 21st century people. However, there are certain restrictions that inhibit "smart development" in human life. The first reason is people's skepticism about this kind of thing. The second is the high cost of such devices. Solving the first problem is much more difficult, so we need to focus on the second, which can affect the previous one.

The simplest option is to upgrade existing appliances in compare with replace them with new ones. In this way, a person gets a full-fledged smart device, or even a smart home.

A smart home is a wide concept that includes a system of home devices that capable of performing actions and solving certain everyday tasks without human activity. Electrical appliances of the building, which are controlled centrally (from the remote control). The devices are connected to a computer network, which allows us to control them with a PC and provide remote access via the Internet [1].

However, there is an even simpler and cheaper option. The only disadvantage is all appliances go into remote control mode (the previous method included automatic mode).

The obvious and effective solution to this problem is to replace all remote controls with a universal remote control or smartphone. Although the last option has been practiced in everyday life for some time, but not all devices are equipped with infrared (IR) technology.

Furthermore, the smart home technology will be really handy the handicapped and the elderly life who find using a large amount of different remotes very tough or even impossible. The system settles all those problems.

This article describes methods and tools for the development and deployment of a ready-made system of the smart house, which will ease the trivial life for all people, especially the elderly and the

handicapped including devices analysis, hardware and software system review and development and testing of the system.

2. Devices analysis

To develop a low cost “smart home” technology we need to purchase a device for encoding/decoding signals. There are no specialized devices for such purposes, but there are FPGAs, the functionality of which fully covers these requirements. Among all the options, the Arduino FPGA family famous of its ease of use and low price (compared to BroadLink RM Pro, which is several times more expensive than the chosen integrated board) [2].

FPGA is an electronic component used to create digital integrated circuits. Unlike conventional digital chips, the logic of FPGA operation is not determined during manufacture, but get set by programming. For this purpose, there are programmers and debugging environments are used, which allow to set the desiring structure of the digital device using one of the special languages of hardware description (Verilog, VHDL, AHDL and others) [3].

The most popular and one of the budget options is the purchase of FPGA family Arduino. To build a smart home, the Arduino UNO is commonly used as a motherboard, and the Arduino Mini / Nano as an additional option to control certain devices.

The essence of smart home technology is not difficult to deploy, we just need to connect the main board to a computer, laptop or smartphone that has Internet access and organize communication with other modules and sensors via Wi-Fi or Bluetooth. Thus, a properly configured system will be able to remotely read the values of the installed sensors (temperature, fire, moving, lighting etc.) and send them to the user's device through the Wi-Fi or Bluetooth modules [4]. In turn, the user can program the delayed start of devices, adjust their work mode or control electrical appliances remotely (vacuum cleaner, air conditioner, light switch, etc.).

Bluetooth’s big advantage for smart home products is low energy use. With Bluetooth LE, you can have a standalone device that does not need charging for weeks. That may come in handy when we get to the point where all our clothes, all our furniture, all our possessions are smart-home enabled.

Generally, there are a lot of appliances that can be controlled remotely. For instance:

- TV
- audio devices
- slide projector
- conditioner
- purifier
- led-string
- smart-things and so on

However, the simplest option is to buy some of the cheapest FPGAs Arduino Mini / Nano, Bluetooth or WiFi modules and signal transmitters. This is the case considered in this work.

2.1. Hardware system review and development

The reasonable smart home technology development includes two components: hardware and software.

The hardware includes selecting and connecting the required components. For one-time and more convenient FPGA programming, the Arduino UNO board is selected (after that, all devices are connected to the Arduino Mini / Nano).

To get started the system, there are three additional modules are needed for Arduino including:

- IR-receiver VS1838B
- infrared LED
- bluetooth module HC-05

IR receivers, sometimes called IR sensors or IR detection diodes, usually come in two different form factors. You can either buy the diodes separately or mounted on a small breakout board.

They work exactly the same, so it does not matter which one you use. The only difference is that the breakout board often contains a small LED that blinks every time the receiver detects a signal which can be handy for debugging.

First, we need to connect the power supply to the board. Except for providing power, we need to connect an IR receiver.

In our case, the IR receiver (TSOP 1838) is connected to 11 pins and as well as to 5V and GND power supply. FPGA power supply, in our case, is provided by two options: from the 12V power supply or from the USB Type B connector [5].

The VS1838B infrared receiver is a receiver of IR signals that operates at a frequency of 38 KHz. The IR receiver can receive and process an infrared signal, which is an IR pulse of fixed frequency and a certain duration (a bundle of pulses) [6].

In IR signal modulation, an encoder on the IR remote converts a binary signal into a modulated electrical signal. This electrical signal is sent to the transmitting LED. The transmitting LED converts the modulated electrical signal into a modulated IR light signal. The IR receiver then demodulates the IR light signal and converts it back to binary before passing on the information to a Microcontroller.

The scheme of the button-code receiving through IR-receiver and the follow displaying them on the screen is shown in Figure 1.

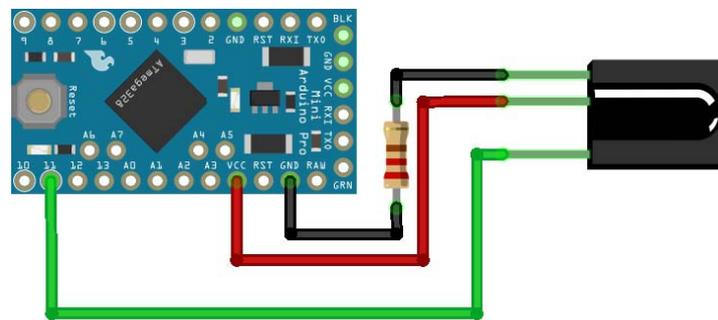


Figure 1: Arduino Mini connection scheme with IR-receiver VS1838B [7]

Features of IR receiver:

- frequency 38 KHz
- supply voltage 2.7-5.5 V
- current 0.4-1.5 mA
- signal reception angle: 90 °
- signal reception range up to 20 m

Infrared LED allows us to create a wireless optical method of data transmission, because infrared rays do not distract a person due to their invisibility [8].

From a physical point of view, an infrared LED is a diode or a simple semiconductor.

To operate other slave modules, it is necessary to connect an IR transmitter and, in our case, a Bluetooth module. The IR transmitter is connected to second pin and GND through a resistor.

The Bluetooth module HC-05 is connected to TXD, RXD, 5V and GND through a resistor. By the way, this module has one more pin called Key. If it is input low level or connect to the air, the module is at paired or communication mode. If it is input high level, the module will enter to AT mode.

Bluetooth module HC-05 is one of the best solutions for two-way Bluetooth communication of our Arduino device with a smartphone, laptop or other Bluetooth devices.

To compare with the HC-06 module, which can only be installed as a Slave, the HC-05 can also be installed as a Master (searching for Bluetooth devices and initiating a connection), which allows communication between two separate Arduino boards [9].

Technical features of the HC05 module:

- bluetooth chip: HC-05 (BC417143)
- radio frequency range: 2.4-2.48 GHz
- speed: Asynchronous: 2.1Mbps(Max) / 160 kbps, Synchronous: 1Mbps/1Mbps

- power supply: 3.3-5 V
- current: 50 mA
- range: up to 10 meters
- profiles: serial port
- modes: master, slave
- working temperature: -25... 75 ° C
- dimension: 27 x 13 x 2.2 mm

The scheme of the button-code receiving device on Bluetooth technology and its decoding in a signal is shown in Figure 2.

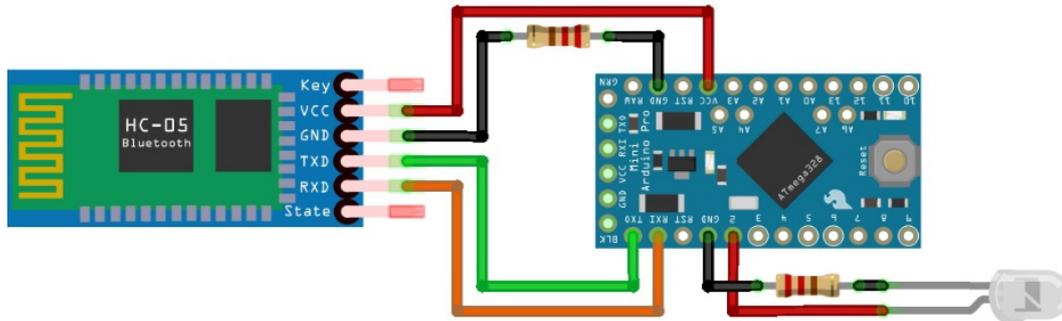


Figure 2: Arduino Mini connection scheme with IR-LED and HC-05 module [7]

By the way, we can combine two schemes, which shown in Fig.1 and Fig. 2, in only one. All we need is to connect IR-receiver to the last scheme and change listing a little. So that, we can adapt each board not only for transmitting signal but for receiving it through IR-technology.

To summarize, we need at least four the most significant modules to build the system (Arduino board, IR-receiver, IR-LED, Bluetooth module) and, of course, a few resistors to prevent a module failure because of the high amount of current, and some wires to combine all the components.

2.2. Software system review and development

The software part includes three components: writing a program for reading codes of different signals from remote controls; writing a program for remote signal transmission using Bluetooth technology; writing a program to receive, decode and transmit a signal to a household appliance.

Arduino programs are written in the C or C++ programming language. Users need to define only two functions in order to create a program that works on a cyclical basis [10]:

- `setup ()`: the function is executed only once at program start and allows to set initial parameters
- `loop ()`: the function is performed periodically until the board is turned off

An IR receiver was connected to the board to receive data from different remotes. It should be said that this operation can be performed without a computer. All we have to do is pre-load the required sketch to the Arduino board and all the codes are displayed on the FPGA monitor itself. However, this option is not so convenient due to the need for manual data recording [11].

An IR transmitter was connected to the Arduino UNO for preliminary testing. The program for decoding and encoding the remote control signal was written and the first results were obtained (in the 16th format) [12]. The principle of operation is to receive a signal from the parent remote control to the IR receiver of the board. After that, a sketch is performed and the recoded code is sent to the IR transmitter, which sends a signal to the device. An example of the obtained codes is shown in Table 1.

The first column is a short description of each command. The second is HEX code that was received from TV-box appliance and the third one is HEX code that was received from TV. The last column is testing buttons to send chosen command through the IR-LED.

To control household appliances with a smartphone, we need to write an application that transmits the received signals using Bluetooth technology to the HC-05 module. However, we can use a ready-made application, many of which are in the Internet and are very similar to each other.

First, we need to connect a Bluetooth module and an IR transmitter to our FPGA Arduino Mini / Nano.

Table 1. Codes from the two tested devices

Commands	SAT	TV	Code
power	9966DA25	FE50AF	0
mute	99667A85	FED02F	1
menu	99669A65	FE2AD5	2
exit	9966B24D	FEEA15	3
left	9966926D	FEDA25	4
right	996650AF	FE1AE5	5
up	9966D827	FE7A85	6
down	9966F00F	FE6A95	7
epg/source	9966708F	FECA35	8
ok	9966D02F	FE5AA5	9

A feature of the HEX encoding format is the almost 100% absence of signal code repetition on different devices [13]. Therefore, programming of all installed Arduino boards is made according to one template.

The algorithm of the program includes several key stages of obtaining, recognizing and decoding commands (Fig. 3). First, a signal transmits via Bluetooth technology to the HC-05 Arduino Mini / Nano module. Then the installed sketch works and after that, through the IR transmitter, the signal transmits the appropriate code to the IR receiver of the device.

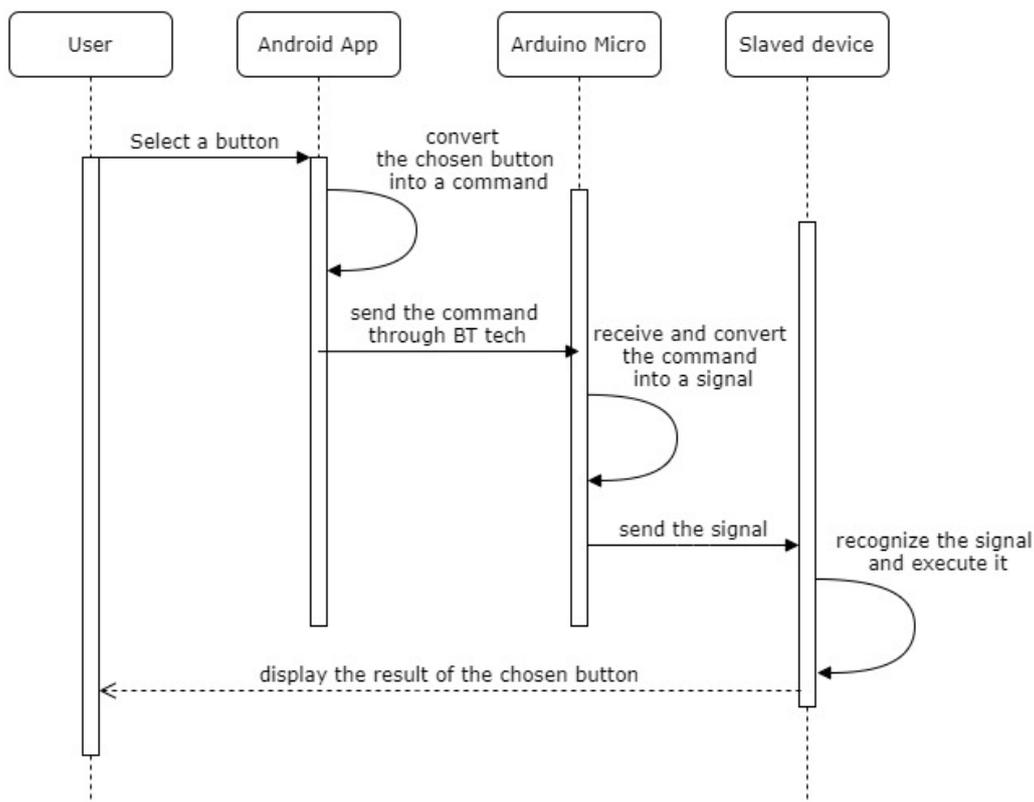


Figure 3: The main algorithm of transferring user's command

The main screen of the application has all the necessary functionality and an intuitive interface (Fig. 4), on which we can control the connected devices or perform other manipulations. The blue

Bluetooth icon means that the device is connected properly and is ready to use. The red Bluetooth icon, in contrast, means that there are problems with the connection [14].

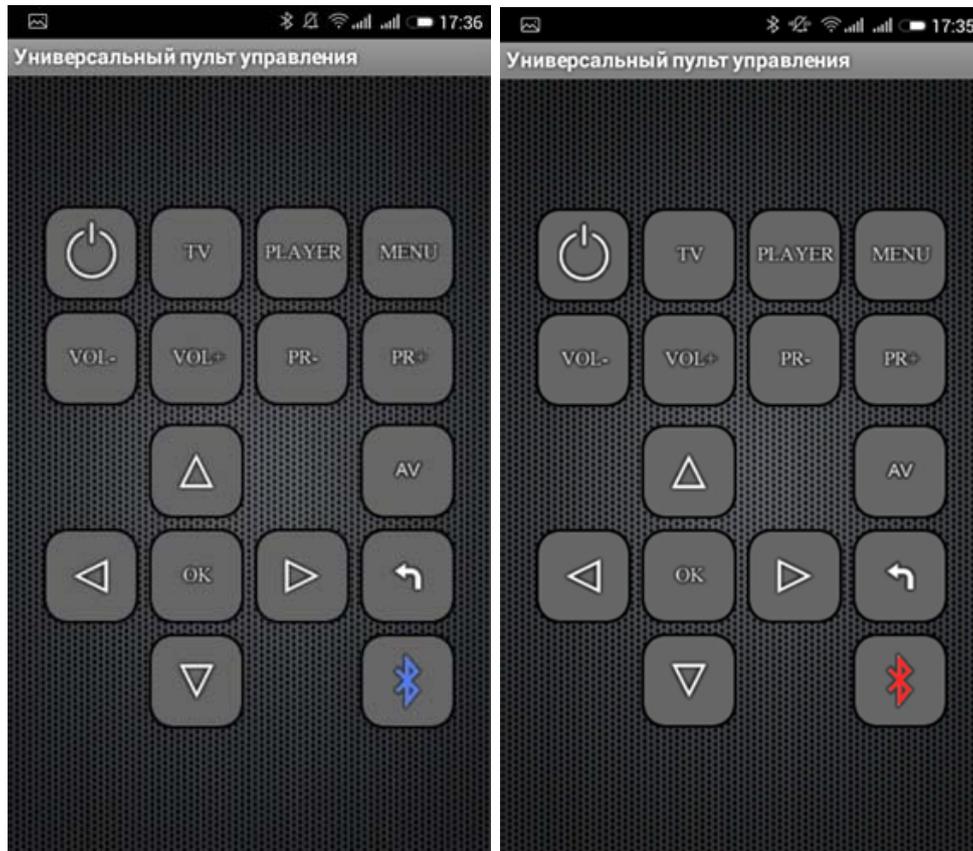


Figure 4: The main menu of the Android application [15]

As a result, we configured all the necessary components including hardware and software parts. We combined two schemes and programmed Arduino boards to execute two types of code: receiving and display code of signal and receiving and transmitting received signal.

3. Testing of the system

At the system testing stage, we need to make sure that the sent signal on the programmable remote control responds correctly and the device correctly receives the command.

First we need to establish connection with all the home appliances. We turn on the Bluetooth on a smartphone and open the application. Then we start searching Bluetooth devices around, connect to them and after that, devices are ready to use.

For testing features of the system, six devices were programmed: two different TVs, a set-top box, a CD player, a DVD player and an LED strip. We can connect up to 10 different devices (that amount was set at the programming stage), but these amount enough for an ordinary user.

The smart home structure is shown in Figure 5.

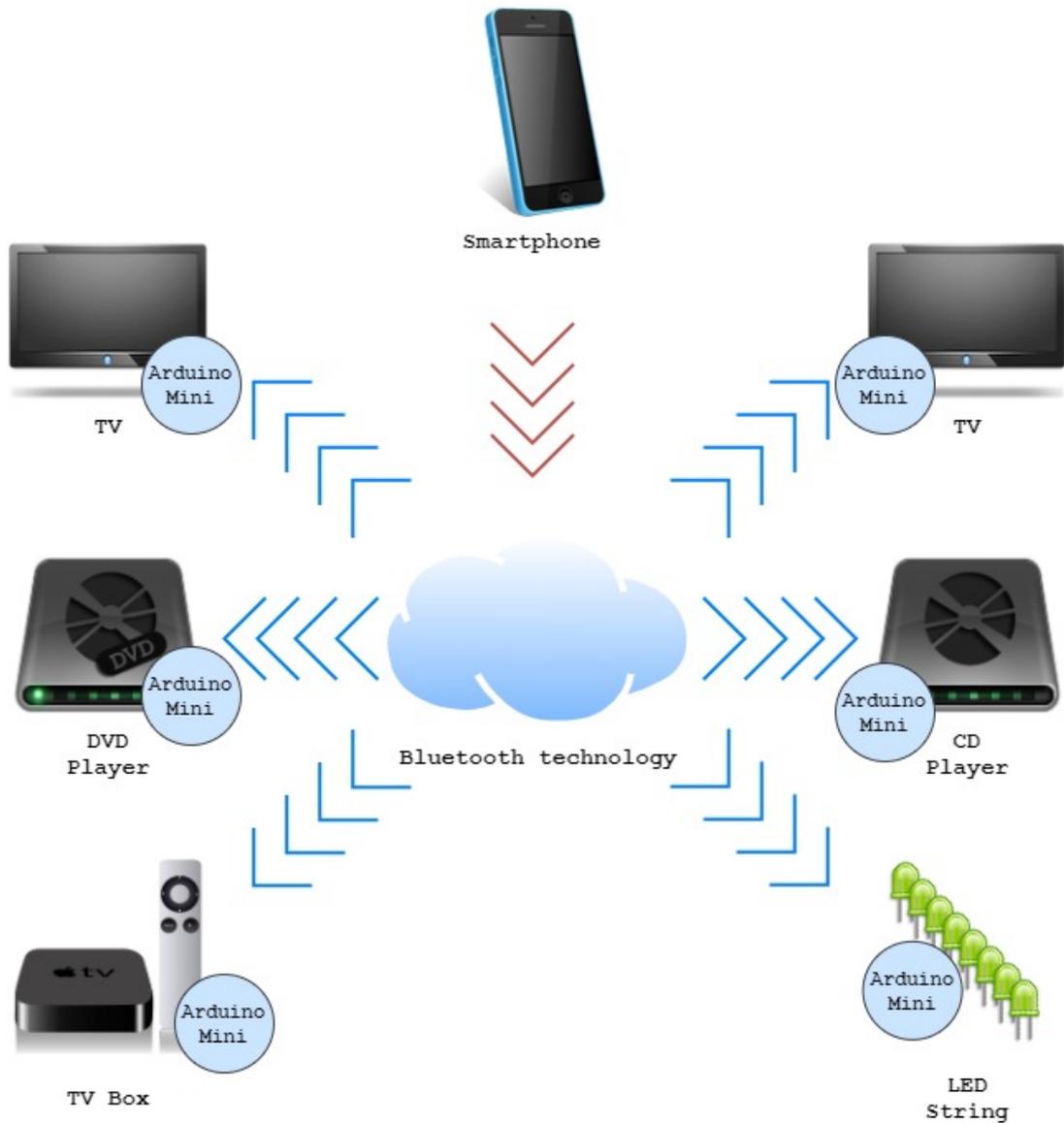


Figure 5: The smart home structure

In order to demonstrate correction of work we join the Arduino Mini system with HC-05 Bluetooth module to laptop and open "Port monitor" tab to see sent signals.

On Uno, Nano, Mini, and Mega, pins 0 and 1 are used for communication with the computer. Connecting anything to these pins can interfere with that communication, including causing failed uploads to the board.

We can use the Arduino environment's built-in serial monitor to communicate with an Arduino board. So that way, we can check which information transmitted through the Arduino board to the terminal device.

The result of work is showed in Figure 6.

```

Decoded NEC: 9966708F (32 bits)
Raw (68): 8950 -4400 600 -1600 650 -450 650 -500 600 -1600 650 -1600 600 -500 600 -450 700 -1550 650 -500 600 -1600 650 -1600
9966D02F
Decoded NEC: 9966D02F (32 bits)
Raw (68): 8950 -4350 600 -1650 600 -500 600 -500 600 -1600 650 -1600 600 -500 650 -450 650 -1600 600 -550 600 -1600 600 -1600
FE50AF
Decoded NEC: FE50AF (32 bits)
Raw (68): 8950 -4450 600 -500 600 -550 600 -500 600 -500 650 -450 650 -500 600 -550 600 -500 600 -1600 650 -1550 650 -1600 60
FED02F
Decoded NEC: FED02F (32 bits)
Raw (68): 9000 -4400 600 -500 650 -500 600 -500 650 -450 650 -500 650 -500 600 -500 600 -500 650 -1600 600 -1600 650 -1550 65
FE2AD5
Decoded NEC: FE2AD5 (32 bits)
Raw (68): 9000 -4400 600 -500 650 -500 600 -500 650 -500 600 -500 650 -500 600 -500 600 -500 650 -1600 600 -1600 650 -1550 65
FE1AE5
Decoded NEC: FE1AE5 (32 bits)
Raw (68): 8950 -4450 600 -500 600 -500 650 -500 600 -500 650 -500 600 -500 600 -500 650 -500 600 -1600 650 -1550 650 -1600 60
FEDA25
Decoded NEC: FEDA25 (32 bits)
Raw (68): 8950 -4450 600 -500 600 -500 650 -450 650 -550 600 -500 600 -500 650 -450 650 -500 600 -1600 650 -1600 600 -1600 60
FE1AE5
Decoded NEC: FE1AE5 (32 bits)
Raw (68): 9000 -4400 650 -450 650 -500 600 -500 650 -500 600 -500 650 -500 600 -500 650 -500 600 -1600 600 -1600 650 -1550 65
FE7A85
Decoded NEC: FE7A85 (32 bits)
Raw (68): 8950 -4450 600 -500 600 -500 650 -450 650 -500 600 -500 650 -500 600 -500 650 -500 600 -1600 600 -1600 650 -1600 60
FE6A95
Decoded NEC: FE6A95 (32 bits)
Raw (68): 9000 -4400 600 -500 650 -500 600 -500 650 -500 600 -500 600 -550 600 -500 600 -550 600 -1550 650 -1600 600 -1600 65
FECA35
Decoded NEC: FECA35 (32 bits)
Raw (68): 8950 -4400 650 -450 650 -500 600 -500 650 -500 600 -450 700 -500 600 -550 600 -450 650 -1600 600 -1600 650 -1600 60
FESAA5
Decoded NEC: FESAA5 (32 bits)
Raw (68): 9050 -4400 600 -500 650 -500 600 -500 650 -450 650 -500 600 -550 600 -500 600 -500 650 -1550 650 -1600 600 -1600 65

```

Figure 6: The result of sent signals through IR technology

As we can see, the codes that we tested and saved earlier received and transmitted correctly. By the way, except Decoded NEC code format we also used the Raw format [16]. We have to use such a format to send the complex signal to such devices as a conditioner and so on. Because these types of appliances get some amount of following codes that going one by one, to apply all necessary signals to different part of system, like humidity, temperature, velocity of wind flow and so on.

All of them work properly and during testing, we definitely had the convenient and benefits of controlling all devices with only one gadget.

Nevertheless, managing devices with the Android application has its advantages and disadvantages.

On the one hand, we get rid of any remote control as a control device and transfer all its functions to a smartphone; what is more, there is no need to point to the controlled device each time. Bluetooth technology works within a certain radius of up to 10 m in any direction [17]. On the other hand, the devices are controlled by sending an IR signal from the Arduino board, so the FPGA must be in direct line of sight to the controlled device.

After all, we got the fully functional system that works correctly and passed all the tests successfully.

In fact, we are able to change Bluetooth technology to Wi-Fi [18]. That will be even more effective because we can control all the devices not only via a smartphone, but also via a tablet, a laptop and so on.

The great advantage of this method is we can control all the devices “absolutely” remotely; there is no necessity to be near the devices like in case Bluetooth technology. So that way we can watch all the indicators and control all home appliances from different places.

Obviously, configure the whole system using Wi-Fi technology will be much more difficult because we have to build a working network system, join it to local network with global internet

access. Our option is simpler and reasonable, even without great abilities in programming a person is able to build and deploy the smart home technology.

4. Conclusions

The result of research work is a universal complex for controlling devices with an infrared sensor, which opens up a wide range of opportunities for its users. Devices are controlled by a smartphone, even if it does not have an infrared port. This allows the user to set up the technology of a smart home, which is very popular and relevant now, in any room.

It is perfect, fully functional, low cost and reliable replacement to regular smart home technology, because not every person can afford to deploy true smart home technology in their house.

All components to build the complex system, including several Arduino Mini/Nano boards, cost even less than only one branded smart device. Once you need to write a necessity code and then just upload it to all the boards. It takes a few minutes, but the result is terrific.

Besides, the developed set of programmable gadgets will help the handicapped and the elderly, who find it very difficult to operate multiple remotes or master a new one. The only thing they should get is getting used to the android application.

As a result, the reasonable smart home technology for controlling household appliances has been developed, programmed and tested.

5. Acknowledgement

The system was implemented in the work process at the LLC "STYLE MAX" enterprise and used as a device for testing different appliances at certain stages of production which controlled by an infrared signal.

6. References

- [1] Dmitry Lubyantsev, Max Tverdokhlib, Tatiana Dyachuk. "Complex of programmable gadgets for household appliances management". The modern science potential (part IV): materials of the IV International scientific-practical conference (2019): 49-50.
- [2] Simon Monk. "Programming Arduino. Professional work with sketches", 2016.
- [3] Jeremy Blum. Exploring Arduino: Tools and Techniques for Engineering Wizardry 1st Edition, (2016): 90-93.
- [4] Simon Monk. Arduino + Android Projects for the Evil Genius: Control Arduino with Your Smartphone or Tablet, 2017.
- [5] Becky Stewart. "Adventures in Arduino", 2015.
- [6] Simon Monk. Programming Arduino: Getting Started with Sketches (Tab) Second Edition (2011): 140-154.
- [7] Fritzing. URL: <https://fritzing.org/>.
- [8] Alasdair Allan, Don Coleman, Sandeep MistryMake. "Bluetooth LE Projects with Arduino, Raspberry Pi, and Smartphones", 2018.
- [9] John Boxall. Arduino Workshop: A Hands-On Introduction with 65 Projects, (2013): 35-43.
- [10] Jody Culkin, Eric Hagan. "Learn Electronics with Arduino: An Illustrated Beginner's Guide to Physical Computing (Make: Technology on Your Time) 1st Edition", 2014.
- [11] Richard Blum. Arduino Programming in 24 Hours, Sams Teach Yourself 1st Edition, (2014): 111-119.
- [12] Turner Ryan. Arduino Programming: The Ultimate Beginner's Guide to Learn Arduino Programming Step by Step Paperback, 2019.
- [13] Mark Geddes. Arduino Project Handbook: 25 Practical Projects to Get You Started 1st edition, 2016.
- [14] Jack Purdum. Beginning C for Arduino, Second Edition: Learn C Programming for the Arduino, (2015): 167-170.

- [15] A universal remote on the Arduino. URL: <https://habr.com/ru/post/254761/>.
- [16] Massimo Banzi, Michael Shiloh. Getting Started with Arduino: The Open Source Electronics Prototyping Platform (Make) 3rd Edition, 2014.
- [17] Mr. Lawrence Harte. Introduction to Bluetooth: Technology, Operation, Profiles, and Services 2nd Edition, 2019.
- [18] Marco Schwartz. Internet of Things with Arduino Cookbook, 2018.