International Journal of Engineering and Innovative Research

http://dergipark.gov.tr/ijeir

DRIVING 0-10V CONTROLLED MOTOR SPEED CHANGER WITH ARDUINO BASED PROPORTIONAL CONTROL LOGIC

Özdemir Deniz[1*] Arif Emre Özgür[2]

[1]Isparta University of Applied Sciences, Uluborlu Selahattin Karasoy Vocational School, Department of Electronics and Automation, Isparta, Turkey.
[2]Isparta University of Applied Sciences, Faculty of Technology, Department of Mechanical Engineering, Isparta, Turkey.

[*]Corresponding Author: ozdemirdeniz@isparta.edu.tr
(**Received:** 10.02.2020; **Revised**: 02.04.2020 **Accepted:** 13.04.2020)

ABSTRACT: 0-10V control is a control protocol used in many industrial devices. The DC voltage supplied externally to the device allows some settings to be made automatically. The most known of these is the adjustment of the LED brightness known as Diming with this DC input. It is divided into many parts when it is considered as a brightness scale of 0 to 10V. 0V corresponds to an unlit lamp, while 10V corresponds to the brightest lamp can be lit. The voltage in between is used for situations between the Dimmest and brightest. Output value of 0-10V is also used for devices that can be in the form of such scales. In this study, the design of the Arduino-based 0-10V control circuit of the VFD (Variable Frequency Drive) device, which controls the speed of a compressor used in compressing the coolant in the heat pump, is explained. P (proportional) control method is used as the control method.

Keywords: Arduino, Motor Speed Changer,

1. INTRODUCTION

With the developing engineering technologies, development cards are used to reduce costs and increase the application in education [1]. There are microcontroller, input and output pins on development cards [2]. In this way, many environmental sensors can be included in development cards. Development cards differ according to their features such as microcontroller, working platform, number of input-output pins, memory [3]. One of these development boards is Arduino. Arduino is an open source code and hardware development board developed by Italian engineers [4-5]. Arduino has many models according to usage areas and number of input and output pins. Some models of Arduino development board are given in Table 1.

Table 1. Arduino models

Arduino DUE	Arduino Leonardo	Arduino Yun	Arduino DUE	Arduino Leonardo
Arduino Mega 2560	Arduino LilyPad	Arduino Industrial 101	Arduino Mega 2560	Arduino LilyPad
Arduino Esplora	Arduino Uno	Arduino 101	Arduino Esplora	Arduino Uno
Arduino Pro Mini	Arduino Mini	Arduino Tian	Arduino Pro Mini	Arduino Mini
Arduino BT (Bluetooth)	Arduino Nano	Arduino Zero	Arduino BT (Bluetooth)	Arduino Nano

Frequency inverter devices are a series of frequency inverters used to control the speeds of three-phase AC motors. There are various models with different features from single phase input of 370 W to three phase input of 970 kW. Frequency Inverters use the state-of-the-art Isolated Bipolar Circuit Transistor (IGBT) technique controlled by microprocessors [6]. This feature makes them reliable and versatile. By using a special pulse width modulation method, the frequency can easily be changed, and motors can be operated silently. Wide range of protection functions ensures perfect protection of the inverter and motor. Frequency inverters are ideal for a wide range of simple motor control applications. Today, frequency inverters can also be used for more advanced motor control applications through a comprehensive parameter list. Almost all frequency inverters can now be used in applications integrated with "Automation Systems" or in "one-unit" applications. The frequency inverter is literally the same as AC drives [7].

2. VFD MOTOR CONTROL UNIT

The installation of the system that uses the MCP4725 DAC to control the heat pump system has been designed and the task has been completed. In this control mechanism, the temperature information is read with the DS18B20 temperature sensor and a DC is produced by setting the frequency value for the VFD device with a proportional control (p control) by taking the difference of the incoming temperature information from the water temperature value designed to be taken at the heat pump output. For this, a circuit that will simulate this system was first established in the Proteus ISIS program. This circuit is shown in Figure 1.

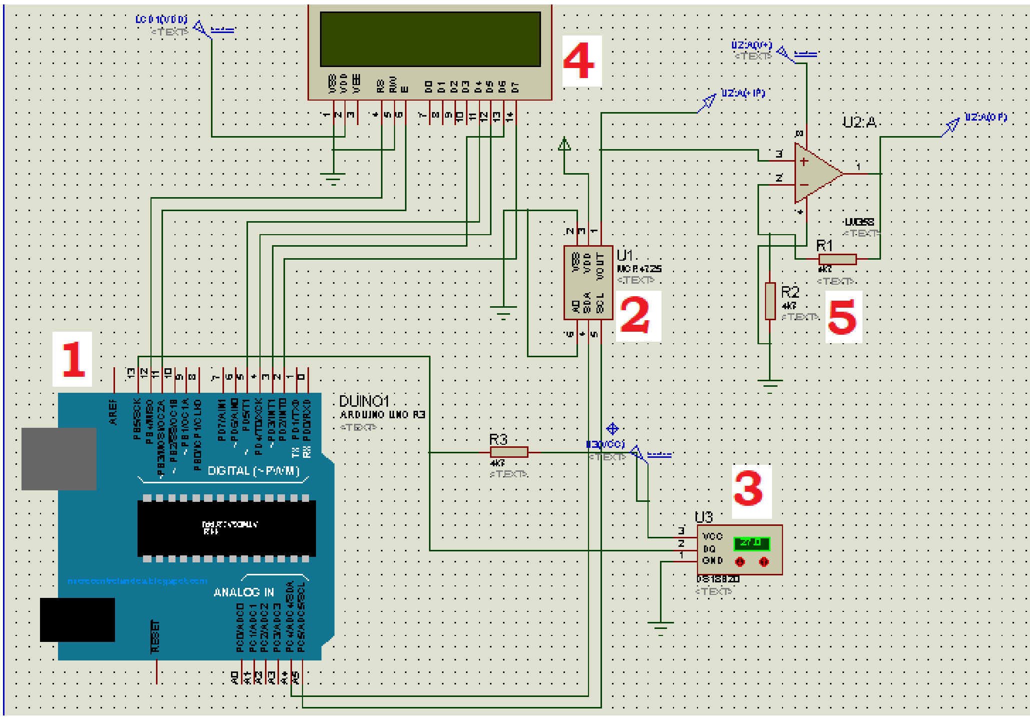

Figure 1. System simulation circuit

3. VFD CONTROL CIRCUIT DESIGN

The diagram of the VFD Circuit is given in the previous section (Figure 1). Here the circuit consists of 5 steps.

- The Arduino board controls the operation of all elements, makes the necessary calculations and ensures that the system operates at the desired values.
- MCP4725 DAC integration provides the 0-10V voltage required by the VFD device to be produced as 0-5V.
- The DS18B20 sensor measures the evaporator water outlet temperature and sends it to Arduino and enables Arduino to use the calculation to send the required voltage to VFD so that this temperature reaches the desired value.
- 2x16 LCD ensures that the required values (Temperature, voltage produced by DAC, frequency value corresponding to that voltage) is monitored on the screen.
- By operating the LM358 operational amplifier in inverted mode and using equal resistors, it provides a 2-fold increase according to the formula (1 + Rf / (R1)) to increase the 0-5V value in the AC output to the 0-10V range.

After the simulation of the circuit is done here, the circuit board construction is started. For this, the circuit was first transferred to CADSOFT Eagle Layout Editor program and shown in Figure 2

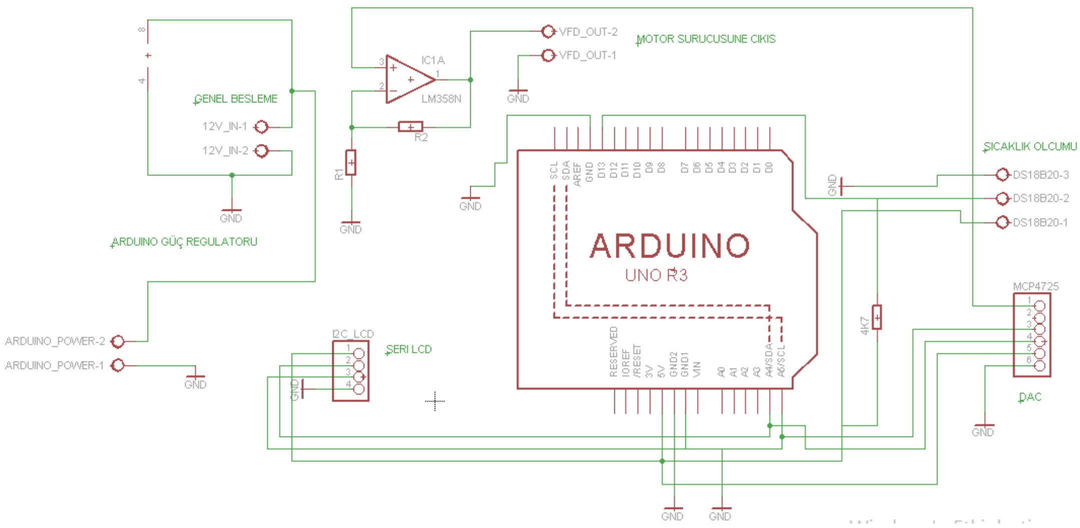

Figure 2. Transfer of vfd to cadsoft eagle layout editor program.

With the help of this software, necessary arrangements were made to transfer them to the printed circuit board and circuit paths were removed. These circuit paths are shown in Figure 3.

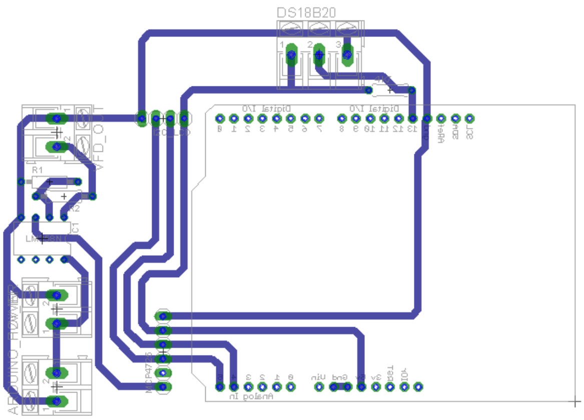

Figure 3. VFD printed circuit board.

Later, this scheme was printed with a laser printer on thin coated paper to be used as transfer paper in order to create the circuit board. This detail can be seen in Figure 4

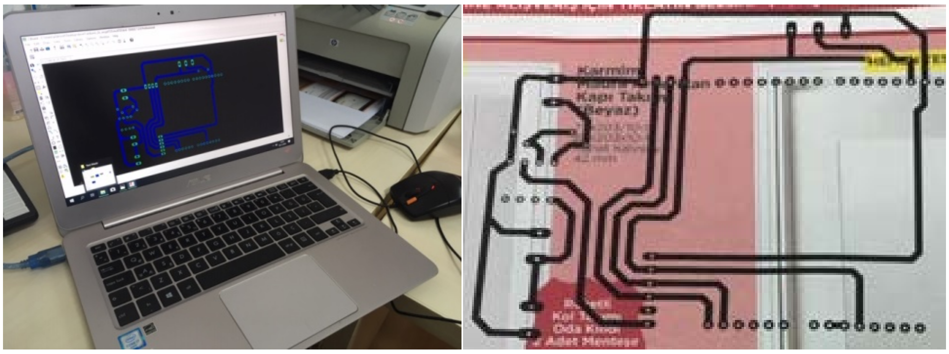

Figure 4. The circuit printed on coated paper.

The circuit diagram transferred on coated paper is attached to the copper plaque with the help of adhesive tape (so that the toner part and copper touch each other) and this time is placed in the Lamination machine heated to 200 ^{o}C to pass the toner onto the copper plaque. In the lamination machine, toners are glued to the copper plate by moving back and forth for 3-4 minutes. The lamination device used is shown in Figure 5.

Figure 5. Laminating device used for the printed circuit.

When the paper that came out of the lamination machine hot and adhered to the plaque was softened with water, the paper melted and the toners remained stuck on the plaque. While the copper plate was in this state, it was thrown into the 70% -30% Salt spirit-Peryhdrol mixture. This chemical reacted with copper and all the copper that cannot be sealed by the toner is dissolved. After this procedure, which lasts for 3-4 minutes, the card was removed from the chemical with a tong and without touching it manually, and the toner and toner were cleaned. The paths of the copper plate were removed in this way. After this stage, necessary holes were opened for the placement of the circuit elements. The visual is given in Figure 6

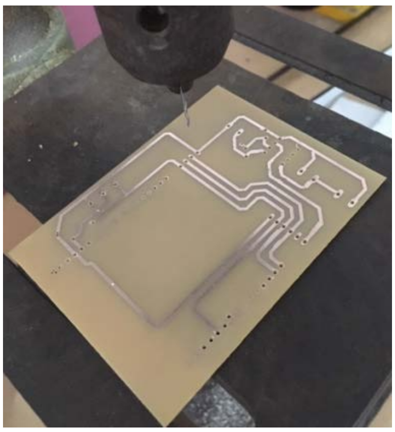

Figure 6. Opening of circuit paths in copper plate.

In the next stage, the upper element placement of the circuit was printed and the installation was started. In the assembly stage, mounting bracket, solder, paste and solder wire and side cutter were used. The visuals of the manufacturing process are given in Figure 3.50.

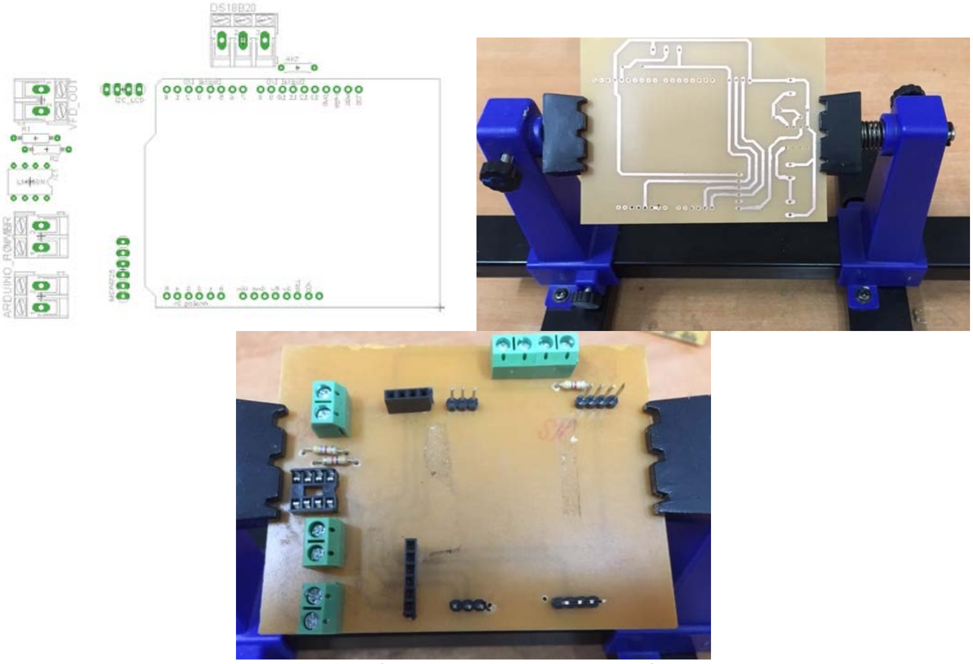

Figure 7. Sections from the VFD card manufacturing process.

After all the placements are made, the connection points reminders are written on the card with the printed circuit pencil and the assembly is completed. The final view of the finished card is shown in Figure 8.

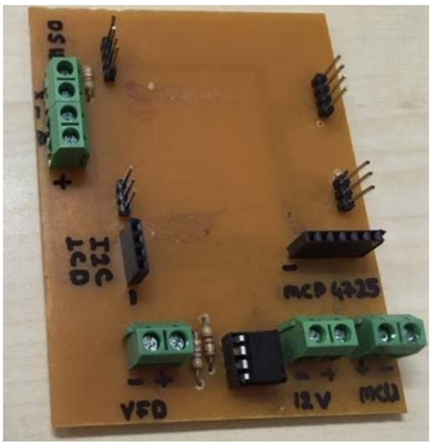

Figure 8. VFD card

After this stage, Arduino and other elements were placed on the card and the code to be written on Arduino started.

4. MODELING OF CONTROL SOFTWARE

It was previously described that certain DC voltages must be sent from the outside in order to control the VFD device remotely. By setting the parameters of the VFD device, the lowest frequency value is set to 40Hz and the highest frequency is set to 55Hz. This frequency range is considered to be 0-55Hz and 0-10V DC voltage that must be given externally to obtain this frequency is rated in a Table 2.

Table 2. DC voltage to be produced for the desired Frequency Value

Frequency Value	Volt Value
55	9,999
54,2	9,85356
53,4	9,70812
52,6	9,56268
51,8	9,41724
51	9,2718
50,2	9,12636
49,4	8,98092
48,6	8,83548
47,8	8,69004
47	8,5446
46,2	8,39916
45,4	8,25372
44,6	8,10828
43,8	7,96284
43	7,8174
42,2	7,67196
41,4	7,52652

The voltage value here will be created by the DAC integrated circuit and will be proportional to the temperature read at the evaporator outlet. The compressor will run at high frequency until it reaches the temperature read from the evaporator, the frequency value will decrease proportionally when it reaches the target temperature, and after the target temperature value is reached, the compressor will be accelerated and slowed down in small steps to keep this temperature constant. The necessary comparison for this to happen is prepared in Table 3.

Table 3. Frequency and DAC output values versus temperature values

Tset	Tan	Tfark	Kp	Frequency Value	Volt Value	DAC Value
40	20	20	0,4	55	9,999	4094,59459
40	22	18	0,4	54,2	9,85356	4035,03686
40	24	16	0,4	53,4	9,70812	3975,47912
40	26	14	0,4	52,6	9,56268	3915,92138
40	28	12	0,4	51,8	9,41724	3856,36364
40	30	10	0,4	51	9,2718	3796,8059
40	32	8	0,4	50,2	9,12636	3737,24816
40	34	6	0,4	49,4	8,98092	3677,69042
40	36	4	0,4	48,6	8,83548	3618,13268
40	38	2	0,4	47,8	8,69004	3558,57494
40	40	0	0,4	47	8,5446	3499,0172
40	42	-2	0,4	46,2	8,39916	3439,45946
40	44	-4	0,4	45,4	8,25372	3379,90172
40	46	-6	0,4	44,6	8,10828	3320,34398
40	48	-8	0,4	43,8	7,96284	3260,78624

40	50	-10	0,4	43	7,8174	3201,2285
40	52	-12	0,4	42,2	7,67196	3141,67076
40	54	-14	0,4	41,4	7,52652	3082,11302

The values in Table 3 are explained as follows.

- T_{set}: The target temperature to be reached at the evaporator outlet
- T_{an}: Instantaneous temperature (calculated and proportioned with the values that can be encountered in the table)
- T_{fark}: The value difference between target and instantaneous temperature.
- K_p: Proportional, that is, the proportion coefficient as "p" type (proportional) control is performed.
- Frequency Value: the frequency value that the compressor should operate according to the current temperature
- Volt Value: The estimated DC voltage value that must be sent to the VFD for the VFD to operate at the current frequency
- DAC value: The decimal equivalent of the 12-bit digital signal that must be sent to the DAC circuit in order to generate the signal that can give the volt value (12 bit, 2^{12} = 4096, in this case, DAC generates voltage between 0-5V in response to the digital information sent between 0-4095 and this value is doubled with the operational amplifier. exponentially increased to the range of 0-10V).

Here the formula FR = 47 + (T_{fark} x K_p) is used for the frequency value, the formula V = FRx0,1818 for the Volt Value and the DAC = V / (0.002442) for the DAC value.

After the proportional control is done in this way, serial LCD control codes and DS18B20 temperature sensor read codes are added to the coding. The information read from the sensor is matched with the T_{an} variable and coding is completed.

5. CONCLUSION

In the application work carried out, the 0-10V control circuit was designed with the Arduino Uno development board in order to reach the desired temperature of the VFD device, which controls the speed of a compressor. P control method was used as the control method. Frequency and DAC output values were checked against the temperature values in the control processes of the control card. In addition, the DC voltage to be produced according to the desired frequency value has been determined.

REFERENCES

[1] Banzi, M., & Shiloh, M. (2014). Getting started with Arduino: the open source electronics prototyping platform. Maker Media, Inc..

[2] Louis, L. (2016). WORKING PRINCIPLE OF ARDUINO AND U SING IT. International Journal of Control, Automation, Communication and Systems (IJCACS), 1(2), 21-29.

[3] Barbon, G., Margolis, M., Palumbo, F., Raimondi, F., & Weldin, N. (2016). Taking Arduino to the Internet of Things: the ASIP programming model. Computer Communications, 89, 128-140.

[4] Parameswaran, G., & Sivaprasath, K. (2016). Arduino based smart drip irrigation system using Internet of Things. Int. J. Eng. Sci, 5518.

[5] Süzen, A. A., Ceylan, O., Çetin, A., & Ulusoy, A. (2017). Arduino Kontrollü Çizim Robotu. Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 8(Özel (Special) 1), 79-87.

[6] Dijle, M., Ertay, M. M., & Yücedağ, İ. (2016). The inverter structures used in dynamics compensation devices: STATCOM and DSTATCOM topologies. Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 20(2), 265-279.

[7] Inverter, Erişim Tarihi: 15.12.2019, Erişim Linki: http://www.inverter plc.net /motor_kontrol/ frekans _inverteri_nedir.htm