**7 Segment Display ile Sayı Yazdırma**

INF212 Project #2

Prepared by

1801022071

Alperen Arslan

INF 212 Algorithms and Programming II

Spring 2020

Electronic Engineering

Date Submitted: 17.04.2020

Evaluation/Değerlendirme

|  |  |
| --- | --- |
| Report layout /Rapor düzeni | 10 -- |
| Project Flowchart and functions flow chart /Projenin ve fonksiyonların Akış Diyagramı | 25 -- |
| Programming /Programlama Struct, Functions | 45 -- |
| Questions / Sorular | 20 -- |

|  |
| --- |
| Project Objective /Projenin Amacı |
| Projenin amacı öğrenci numarasının ilk iki ve son iki hanesinin hexadecimal karşılığını 7 segment display veya 8x8 dot matris sayesinde sırasıyla yazdırmak. |
| Description of Problem / Problem Tanımlama |
| 7 segment display veya 8x8 dot matris’in Arduino ile bağlantılarının yapılması ve öğrenci numarasının ilk iki ve son iki rakamlarını hexadecimal form’a çevirip sonrasında bu çevrilen rakamları sırasıyla 7 segment display veya 8x8 dot matris’e yazdırılması gerekmektedir. |
| **Description of Method / Metodunun Tanımlaması** |
| Proteus üzerinden 1 adet Arduino Uno, 2 adet 7 segment display ve 2 adet shift register olan 74HC595 test alınına yerleştirildi. Gerekli bağlantılar yapıldı. Arduino IDE üzerinde sistemi çalıştıracak kod yazıldı ve derlendi. Derleme sonucunda oluşan .hex dosyasının adresi kopyalandı ve Proteus üzerinden devredeki Arduino’nun özelliklerinden dosya yoluna yapıştırıldı. Bu sayede Proteus üzerinden test edilmek üzere kurulan devre için Arduino kod dosyasının konumunu bulabildi ve simülasyon gerçekleştirilebildi. |
| Text of Program/ Programın Kodunu Tanımlaması |
| Program kodu raporun sonuna eklenmiştir. |
| User’s guide / Kullanıcı Rehberi |
| Projenin elektronik kısmı iki adet 7 segment display, Arduino Uno ve 2 adet 74HC595 modelinde shift register’dan oluşuyor. Yapılması gereken devre şemasındaki bağlantıların yapılması ve sonrasında ise .ino dosyasındaki kodların Arduino Uno’ya yüklenmesi. Sonrasında Arduino’ya güç verildiği sürece 7 segment displayler sırasıyla kodun en üstünde yazılan öğrenci no’sunun ilk iki ve son iki rakamlarını hexadecimal forma dönüştürüp bu formdaki karşılıklarını 2 saniye aralarla gösterecektir. Kullanıcı sayıları değiştirmek istediğinde ise Arduino kodunun en üstünde define edilen numaraIlkIki ve numaraSonIki değişkenlerinin karşılığını değiştirmelidir. |
| Results of the solution / Programın Sonuçları Gerekli bağlantılar yapıldıktan sonra aşağıdaki gibi bir görsel karşımıza çıkıyor. Görselde simülasyon henüz başlatılmadı.  Aşağıdaki görselde ise simülasyon çalışmaya başladı. Arduino kodunda öğrenci numarasının ilk iki rakamı olarak 18 girildi. Kod 18 rakamını Converter şeklinde isimlendirilen kod sayesinde hexadecimal karşılığı olan 12 sayısını elde etti ve bunu ledYakma şeklinde isimlendirilen fonksiyon sayesinde 7 segment displaylere yazdırdı.  Aşağıdaki görselde ise öğrenci numarasının son iki hanesi olarak Arduino koduna yazılan 71 sayısının hexadecimal karşılığı olan 47 sayısının 7 segment display ekranlara yazdırıldığını görüyoruz. Program bunu yaparken yukarıda yazılan mantıkla çalışmaya devam ediyor.    Sistem sürekli güç verildiği takdirde 12 ve 47 sayılarını sırasıyla 7 segment displaylere yazdırmaya devam edecek. |
| Flowchart of the Program / Programın Akış Diyagramı |
|  |
| Conclusion and Remarks / Sonuç ve Notlar |
| Sistem gayet düzgün çalışmaktadır. Arduino’ya güç verildiği sürece kodun en başında ilk ve son iki rakamların hexadecimal karşılıklarına çevirip, bu karşılıklarını 7 segment displaylere yazdıracaktır. 7 segment displayin sadece tek bir rakamı göstermesi ve benim de tek seferde 2 rakam barındıran sayılar yazdırmam gerektiği için 2 adet 7 segment ortak anotlu display kullandım. 7 segment displayin ‘.’ yazdırmayı sağlayan bağlantısı yapılsa da programdan istenen çalışma mantığında yeri olmadığı için kullanılmadı. Eğer istenirse sadece kodu yazılıp çalıştırılabilir. Herhangi bir devre ayarlamasına gerek duymamaktadır. Arduino’da ne kadar az pin kullanırsam o kadar iyi şeklinde düşünerek 74HC595 shift registerlarını kullandım. Devrenin ve kodun görünürde herhangi bir sıkıntısı bulunmamakta. Tanımlanan 2 haneli sayıları hexadecimal çevirip yazdırıyor. 1 haneli veya 2 ve üstü haneli girilen sayılarda sistem düzgün çalışmayacaktır. Ama projenin amacında bizden 2 haneli sayıların çevrilip yazdırılması istendiğinden projenin amacına aykırı bir durum vey ahata olmamış oluyor. Gayet kullanıcı dostu bir kod. Proje bana hem elektronik bilgisi olarak hem kod ve algoritma yazma becerisi olarak hem de en önemlisi araştırma becerisi olarak çok şey kattı. 7 segment displayi kullanmak için internetden datasheet ve bağlantı şemalarını inceledim. Aynı şekilde shift registerları kullanmak için datasheet ve birsürü video ve blog sayfaları inceledim. Kod tarafında ise void fonksiyonlar hakkındaki bilgilerimi pekiştirirken özellikle fonksiyon içinde fonksiyon kullanmayı da öğrendim. |
| References / Kaynaklar |
| * Kodlakafa.com * Alldatasheet.com * Direnc.net * Diyot.net * Elektrikport.com * Stackoverflow.com * Algoritmaornekleri.com * Arduinolibraries.com * Github.com * Sanatsalbilgi.com * Theengineeringprojects.com * Kontrolkalemi.com * Elektronikhazirdevreler.blogspot.com * Rapidtables.com * Koddefteri.com * Erhanulman.com * Forum.donanimhaber.com * Forum.arduino.cc * Arduinobasics.blogspot.com * 1.bp.blogspot.com * Elektrikelektronikegitimi.blogspot.com * Maker.robotistan.com * Docs.labs.mediatek.com * Projehocam.com * Cagataycebi.com * Forum.csharpnedir.com * Edevre.net * En.wikipedi0.org * Hayaletveyap.com * Grafiboyz.blogcu.com * Aliozcan.org * Gelecegiyazanlar.turkcell.com.tr * Iucoders.com * Sanalkurs.net * Buraksenyurt.com * Microcore.com * Link24in.wordpress.com * Robotikvekodlama.com * Soyunmezoglubaris.blogspot.com * Motorobit.com * Mobilhanem.com * Arduinoturkiye.com * Kodumundunyasi.net |

# *Kod:*

# #define numaraIlkIki 18 //Ogrenci numarasinin ilk ve son iki numarasini buraya yazar

# #define numaraSonIki 71

# #define LATCH 2 //STCP

# #define DATA 3 //DS

# #define CLK 4 //SHCP

# #define MR 5 //MR

# #define NOKTA 128

# #define HARF\_A 119 //1110111

# #define HARF\_B 124 //1111100

# #define HARF\_C 57 //0111001

# #define HARF\_D 94 //1011110

# #define HARF\_E 121 //1111001

# #define HARF\_F 113 //1110001

# byte cc\_rakamlar[10] = {

# 63, //0 (0b00111111)

# 6, //1 (0b00000110)

# 91, //2 (0b01011011)

# 79, //3 (0b01001111)

# 102, //4 (0b01100110)

# 109, //5 (0b01101101)

# 125, //6 (0b01111101)

# 7, //7 (0b00000111)

# 127, //8 (0b01111111)

# 111 //9 (0b01101111)

# };

# void setup(){

# pinMode(LATCH, OUTPUT); //Atanan pinlerin modları belirtildi

# pinMode(DATA, OUTPUT);

# pinMode(CLK, OUTPUT);

# pinMode(MR, OUTPUT);

# digitalWrite(MR, LOW); //Setup her calistirildiginda Master Reset e reset atmak gerekir

# digitalWrite(MR, HIGH); //Once LOW sonra HIGH vererek reset atiyoruz

# }

# void loop(){

# converter(numaraIlkIki);

# delay(1000);

# converter(numaraSonIki);

# delay(1000);

# 

# }

# void converter(int number){ //Bu fonksiyon kendisine gonderilen iki basamakli sayilari once

# int bolum = number / 16; // Hexadecimal e cevirir sonra ise ledYak fonksiyonuna gonderip

# ledYak(bolum); // cevirilen hexadecimal sayiyinin 7 segment displayde yazdirilmasini

# int kalan = number%16; // saglar

# ledYak(kalan);

# }

# void ledYak(int sayi){ // Bu fonksiyon converter fonk. dan aldigi sayilari displaye yazdirir

# if(sayi == 10){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_A);

# digitalWrite(LATCH, HIGH);

# }

# else if(sayi == 11){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_B);

# digitalWrite(LATCH, HIGH);

# }

# else if(sayi == 12){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_C);

# digitalWrite(LATCH, HIGH);

# }

# else if(sayi == 13){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_D);

# digitalWrite(LATCH, HIGH);

# }

# else if(sayi == 14){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_E);

# digitalWrite(LATCH, HIGH);

# }

# else if(sayi == 15){

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,HARF\_F);

# digitalWrite(LATCH, HIGH);

# }

# else{

# digitalWrite(LATCH, LOW);

# shiftOut(DATA, CLK, MSBFIRST,cc\_rakamlar[sayi]);

# digitalWrite(LATCH, HIGH);

# }

# }