

The Minimum Number of Banknotes Exchange Project

Ferhat Açıklan

180202065@kocaeli.edu.tr

Berkay Yasin Çiftçi

190202056@kocaeli.edu.tr

I. GİRİŞ

Programlama Laboratuvarı 2 - II. Projesi olarak bizden “En Az Sayıda Banknot Üstü Verme ” adında bir otomatik araç yıkama makinesinin minimum sayıda para üstü vererek çalışması beklenen ve Tiva veya Stellaris veya Sanal Ortam kullanarak uygulama gerçekleştirmemiz beklenmektedir.

II. PROJENİN ÖZETİ

Biz proje için Proteus 8 CAD Connected Uygulamasını ve Arduino UNO R3 kartını kullandık. Bunları kullanma sebeplerimizden; açık kodlu, yüksek verimli, çok işlevli, yüksek seviyeli ve kullanımı kolay olması yatmaktadır.

Proteus platformu; temel olarak elektronik tasarım için kullanılan , baskılı devre kartı üretimi için ve elektronik baskı yapmak için kullanılmaktadır. Ve birçok simülasyonda gerçekleştirme imkanı kullanıcılarına sunmaktadır.

Proje dokümanında istekler bulunmaktadır. Bu isteklere göre projemizi şekillendirmiş bulunmaktayız. Projemizde 6 adet buton bulunmaktadır ve bu butonlar isteklerdeki butonlara uygun olacak şekilde tasarlanmıştır. Ve bir adet sıfırlamak için reset butonumuz bulunmaktadır. İki adet led bulunmaktadır.

III. TEMEL BİLGİLER

A. YAPILAN ARAŞTIRMALAR VE KARŞILAŞILAN SORUNLAR

Proje geliştirilirken karşılaşılan sorunlardan birkaçı aşağıda belirtilmiştir ve bu sorunları nasıl aştığımız ve yapılan araştırmalardan aşağıda detaylı olarak bahsedilmiştir.

Birinci olarak karşılaşılan sorun; Proteus Design Suit adlı program demo versiyonu yapılan çalışmaları kaydetme imkânı sunmuyordu. Biz de bu sorunu aşmak için internetten crackli sürümünü indirerek bu sorunumuza çözüm bulmuş olduk.

İkinci olarak karşılaşılan sorun; harita.txt dosyasında bulunan TL adetlerini , 4 adet hizmete ait bilgileri(HizmetID , HizmetAd , KalanHizmetAdet ve HizmetFiyat) okunması konusunda zorlandık.. Bu sorunu çözmek için kısa bir araştırma yaptıktan sonra gerekli fonksiyonları ve değişkenleri kullanarak projemize aktarmış olduk.

Üçüncü olarak karşılaşılan sorun; acaba kart mı alıp yapsak yoksa simülasyon şeklinde mi diye kafamızda soru işaretleri oluştu. Sonunda simülasyonun hem zaman açısından hem de hız açısından ve yanlış yapma ihtimali açısından daha iyi olacağına kanaat verdik ve simülasyon kullandık. Simülasyon uygulaması olan Proteus Design Suit de Arduino

UNO R3 kartı bulunmuyordu. Küçük bir araştırma sonucu Kütüphaneyi internetten edinerek programımıza tanıttık. Ve arduino UNO R3 kartımızı kullanmaya başladık.

Dördüncü olarak karşılaşılan sorun ise; bitiş butonu basıldığı zaman hizmetler bölümüne geçiyordu fakat o sırada 2. Ve 3. Butona basarken de hem para ekleme hem de hizmet ücreti ekleme işlemini yapmaktaydı. Yazdığımız kodu while döngüsü içine alıp bitiş butonuna bastıktan sonra if şeklinde break yaparak döngüden çıkartarak bu sorunu da çözmüş bulunmaktayız.

Beşinci olarak karşılaşılan sorun ise; random sayı üretirken her seferinde çalıştırıldığında aynı sayıyı üretmesiydi küçük bir araştırma sonucu randomSeed adlı fonksiyonu kullanarak çözülebileceği kanısına vardık başta az deneme yaptığımızdan hallolduğunu düşündük hâlbuki her çalıştırmada aynı sayıları üretiyordu. O yüzden AnalogRead adlı fonksiyonu kullanarak bu işin de üstesinden gelmiş bulunmaktayız. Ve randomSeed içine milisaniye kullanrak bu sorundan da üstesinden geldik.

B. YAZILIM MİMARİSİ

Projede gerekli olan fonksiyonlar tanımlandı. Fonksiyonlar oluşturulduktan sonra işlevleri hakkında bilgi sahibi olmak istersek;

parayukleme():

Bu fonksiyon kullanıcının butonlar yardımı ile sisteme para aktarımı yapması için yazılmıştır ve işi bittiğinde bitiş düğmesine basarak bir sonraki menü olan hizmetler menüsüne geçiş yapmasını sağlamaktadır.

Fonksiyonumuz void tiplidir. Fonksiyonumuzda bulunan count5,count10,count20,count50,count100, gibi değişkenler global olarak projemizin başında kodlanmıştır. Bu değişkenler adından da anlaşılacağı gibi sisteme kullanıcı tarafında yüklenen paraların sayısını tutmaktadır.

Bu fonksiyonumuzda kullanıcımız her yükleme yaptığında ekranda ne kadar para attığını görmesini de sağlamaktadır.

Fonksiyonumuzdaki while döngüsünden çıkmak için bitiş kısmı olan state8 ve lastState8 kısmında break deyimi ile döngüden çıkılmaktadır ve diğer bölüm olan hizmetler kısmına geçilmektedir.

Fonksiyonumuzda temp adlı değişkenimiz global tanımlanmış olup kullanıcı tarafından girilen toplam paraların değerini tutmaktadır.

hizmetler():

Bu fonksiyon kullanıcının butonlar yardımı ile sistemden istediği hizmetleri seçmesi için yazılmıştır ve işi bittiğinde bitiş düğmesine basarak bir sonraki bölüme geçiş yapmasını sağlamaktadır.

Bu fonksiyonumuzun başka bir özelliği ise kullanıcımız her butona bastığında işlem ücretinin ne kadar olacağını göstermektedir.

Fonksiyonumuz void tiplidir. Fonksiyonumuzdaki while döngüsünden çıkmak için bitiş kısmı olan state8 ve lastState8 kısmında break deyimi ile döngüden çıkılmaktadır ve diğer bölüm olan hizmetler kısmına geçilmektedir. Hizmetler() adlı fonksiyon çıkışında state8 ve lastState8 durumu değiştiği için fonksiyonumuzun başında sıfırlama işlemi yapılmıştır. Fonksiyonumuzda tempHizmet adlı değişkenimiz global tanımlanmış olup yapılan ücretlendirmenin toplam değerini tutmaktadır.

paraustu():

Bu fonksiyonumuz minimum banknot kullanarak kullanıcımızın para üstünü vermeyi sağlamaktadır.

Fonksiyonumuz void tiplidir. Oduncektutar adında int değerinde değişken almaktadır. Parayattırma() ve hizmetler() fonksiyonumuzun çalışmasından sonra temp ve tempHizmet adlı değişkenlerimizin farkından çıkan sonucu bu fonksiyonumuza göndererek kullanıcıma kaç en az şekilde banknot vermek şartıyla para üstü verilmektedir.

If yapısını kullanarak kasadaki banknotların sayısının sıfırdan farklı olduğunu ve kullanıcıya verilecek banknotlar, her banknotun ayrı ayrı, sıfırdan büyük olup olmadığını kontrol edip ekrana ne kadar hangi banknottan verileceğini bastırmaktadır.

parasikisma():

Bu fonksiyonumuz kullanıcı para yükleyip hizmetini seçtikten sonra para sıkışıp sıkışmayacağını kontrol etmektedir.

Fonksiyonumuz void tiplidir. Ayrıca onksiyonumuzun diğer özelliği ise sıkışma olursa tuşlanan hizmetleri kalan hizmetlerden düşmeyecek şekilde ayarlanmıştır.

Fonksiyonumuz random sayı üretmektedir. Her seferinde farklı sayı üretmesi için randomSeed(millis()) şeklinde kullanımı yaparak her çalıştırıldığında farklı bir sayı üretmektedir.

Fonksiyonumuz üretilen randomsayı eğer ikiye eşitse para sıkışması olmaktadır ve kırmızı led yanmaktadır. Eğer 2 den farklı sayı üretirse para sıkışması olmamaktadır ve yeşil led yanmaktadır.

setup():

Bu fonksiyonumuz çalışma ortamını başlangıç için bizlere hazırlar ve görevini tamamladıktan sonra bir yenisinden başlatmaya kadar tekrar çalıştırılmaz.

Serial.begin(9600) adlı fonksiyonumuz ise arduino nun 9600 bit hızında iletilmesini sağlamaktadır.

Giriş ve çıkış ayarı için kullanılan komutu olan pinMode(), iki adet parametre almaktadır. Birincisi pin numarası diğer ise modu belirtmek için kullanılır. Void setup() fonksiyonu içinde yazılmalıdır. Gerekli pinlerimizin giriş ve çıkışları yapılmıştır.

Kullanacağımız pinlerin dijital giriş olarak tanımlanan bir pin üzerindeki değeri okumamız için digitalWrite() adlı fonksiyonumuz kullanılmıştır. Ve bu fonksiyon bir sonuç döndürür. Ya HIGH ya da LOW değerlerini döndürmektedir. Parayukleme(), hizmetler(), parasikisma gibi oluşturduğumuz fonksiyonlarımız burada kullanılmaktadır.

dosyaokuma():

Bu fonksiyonumuz SD kartta yer alan hizmet.txt dosyasını okuyup, bu dosyada yer alan banknot sayısı ve hizmetlerle ilgileri okuyup programa aktarır. İlk satırı parçalayıp sırasıyla, varolan banknot sayılarını banknot_count[] dizisine ekler. Diğer satırlarda verilen hizmetlerle ilgili olan bilgileri, hizmet_adlari[], hizmet_kalan[], hizmet_ucretleri[] dizilerine sırasıyla ekler.

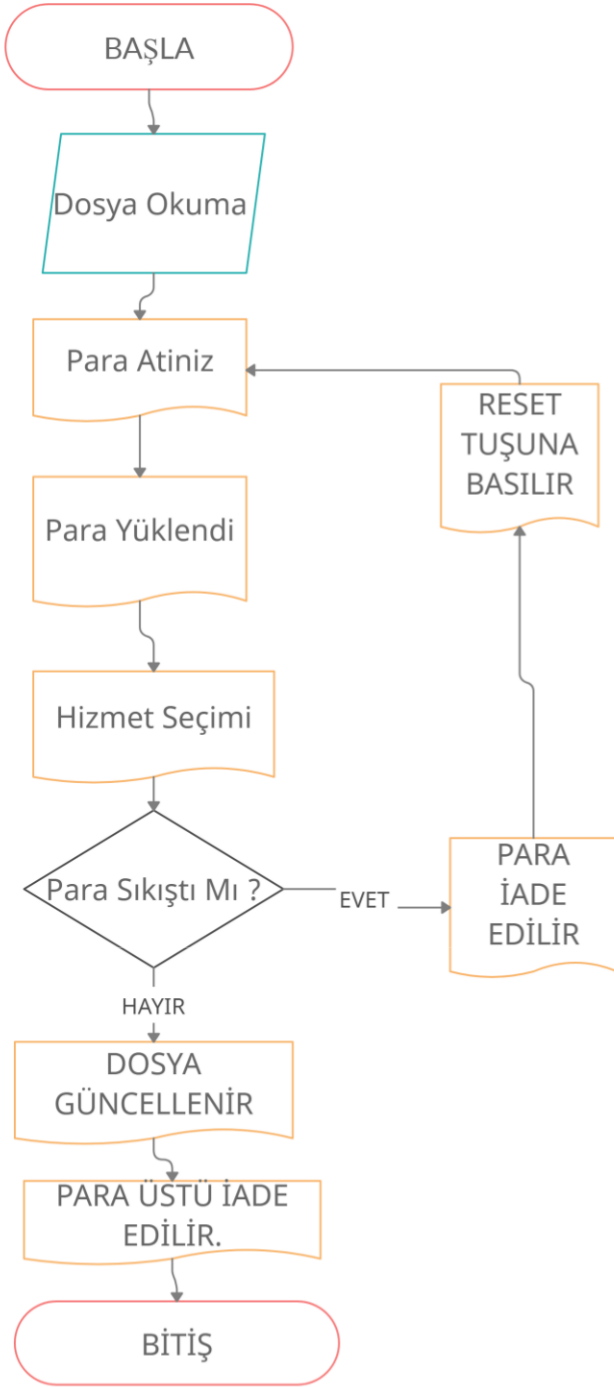
dosyaesitle():

Bu fonksiyon tüm işlemler bittikten sonra değişen bilgilere göre hizmet.txt dosyasını güncellemeye yarar. SD kartta yer alan hizmet.txt dosyasını siler. Yeni dosya oluşturur. banknot_count[], hizmet_adlari[], hizmet_kalan[], hizmet_ucretleri[] dizilerindeki güncellenmiş verilere göre dosya güncellenir.

C. YALANCI KOD

1. Başla
2. Kullanılacak pinleri tanımla
3. SD kart fonksiyonlarını çağır
4. SD kartı oku
5. Hizmet.txt dosyasını aç
6. İlk satırı oku
7. Okunan değerleri banknot_count dizisine ekle
8. Hizmet bilgilerini oku
9. Hizmet adını hizmet_adlari dizisine ekle
10. Kalan hizmet bilgisini hizmet_kalan dizisine ekle
11. Hizmetlerin ücretlerini hizmet_ucretleri dizisine ekle
12. Dosya okumayı bitir
13. Para yükleme işlemine geç
14. Basılan buton değerine göre para yükle
15. Bitiş butonuna basılmadıysa 14. adıma git
16. Para yükleme işlemini bitir
17. Hizmet seçimine geç
18. Basılan butona göre hizmet seç
19. Kalan hizmet sayısını kontrol et
20. Bitiş butonuna basılmadıysa 18. adıma geç
21. Hizmet seçimini bitir
22. Para sıkışma durumunu kontrol et
23. Para sıkıştıysa atılan parayı geri ver
24. Para sıkışmadıysa para üstü ver
25. Yeterli banknot durumunu kontrol et
26. Banknot yeterliyse işlemi bitir
27. Para sıkıştıysa kırmızı ledi yak
28. İşlem başarılıysa yeşil ledi yak
29. Bitir

D. AKIŞ DİAGRAMI



E. DENEYSEL SONUÇLAR

```

Virtual Terminal
SD card okunuyor...
init done.
hizmet.txt:
28,21,12,36,25,
1.Kopukleme 0 15
2.Yıkana 4 10
3.Kurulana 1 5
4.Cilalana 4 50
Para Atınız

Virtual Terminal
SD card okunuyor...
init done.
hizmet.txt:
28,21,12,36,25,
1.Kopukleme 0 15
2.Yıkana 4 10
3.Kurulana 1 5
4.Cilalana 4 50
Para Atınız
10 TL para attınız.
30 TL para attınız.
80 TL para attınız.
180 TL para attınız.
280 TL para attınız.
Para yukleme isleni bitmistir.
1.Kopukleme
2.Yıkana
3.Kurulana
4.Cilalana
6.Bitis

Virtual Terminal
2.Yıkana 4 10
3.Kurulana 1 5
4.Cilalana 4 50
Para Atınız
10 TL para attınız.
30 TL para attınız.
80 TL para attınız.
180 TL para attınız.
280 TL para attınız.
Para yukleme isleni bitmistir.
1.Kopukleme
2.Yıkana
3.Kurulana
4.Cilalana
6.Bitis
hizmetlerin toplan bedeli = 50
hizmetlerin toplan bedeli = 100
hizmetlerin toplan bedeli = 110
hizmetlerin toplan bedeli = 120
hizmetlerin toplan bedeli = 130
Uzgunuz bu hizmeti sindilik veremiyoruz!

Virtual Terminal
280 TL para attınız.
Para yukleme isleni bitmistir.
1.Kopukleme
2.Yıkana
3.Kurulana
4.Cilalana
6.Bitis
hizmetlerin toplan bedeli = 50
hizmetlerin toplan bedeli = 100
hizmetlerin toplan bedeli = 110
hizmetlerin toplan bedeli = 120
hizmetlerin toplan bedeli = 130
Uzgunuz bu hizmeti sindilik veremiyoruz!
Hizmet Secin Isleni Bitmistir...
Para sikisti Paranız tanani geri iade ediliyor...
280 TL paranız iade edilmistir.
  
```

```
Virtual Terminal
20 TL para attiniz.
40 TL para attiniz.
60 TL para attiniz.
80 TL para attiniz.
100 TL para attiniz.
Para yukleme isleni bitmistir.
1.Kopukleme
2.Yikama
3.Kurulama
4.Cilalama
5.Bitis
hizmetlerin toplan bedeli = 50
hizmetlerin toplan bedeli = 100
Hizmet Secim Isleni Bitmistir...
Yeteri Kadar Para Atiladi...
100 TL paraniz iade edilmistir.

Virtual Terminal
1.Kopukleme
2.Yikama
3.Kurulama
4.Cilalama
5.Bitis
hizmetlerin toplan bedeli = 5
Uzunuz bu hizmeti sindilik vereniyoruz!
Uzunuz bu hizmeti sindilik vereniyoruz!
hizmetlerin toplan bedeli = 15
Hizmet Secim Isleni Bitmistir...
Para sikismasi olmadi para ustu iade ediliyor...
4 adet 100 TL
1 adet 20 TL
1 adet 5 TL
sekinde paraniz iade edilmistir.
```

F. SONUÇ

Bu proje sayesinde Arduino UNO R3 kartını kullanarak basit bir şekilde banknot otomasyon cihazını simüle etmiş bulunmaktayız. Sistem programlama dersinde öğrendiğimiz bilgilerle pratik yapmış olduk. Teşekkür ederiz.

G. KAYNAKÇA

<https://forum.arduino.cc/>

<https://www.arduino.cc/>

<https://www.tutorialspoint.com/arduino/index.htm>

<https://www.javatpoint.com/arduino>