**EN AZ SAYIDA BOZUK PARA VERME PROBLEMİ**

Özet

Bu projemizin amacı geçen dönem işlediğimiz sistem programlama dersindeki öğrendiklerimizi uygulamaya dökmektir. Bu aşamada derste kullandığımız tiva veya stellaris kartı breadboard üzerinde display ekran kullanarak uygun kodlarla çalıştırmaktır. Projemizde asıl amaç; bir otomat makinesinin minimum sayıda bozuk para üstü verecek şekilde çalıştırılıp tasarlanmasıdır. Örneğin 1. butona 2, 2. butona 3, 3. butona 4 defa basan biri otomata 6 TL atmış olacaktır ve ekranda (Display) "6TL para attınız" bilgisi gösterilecektir. Sonra bir miktar para girilip, 5 ürün çeşidinden biri butonlar yardımıyla şeçilmesi beklenmektedir. Bu ürünler su, çay, kahve, çikolata ve bisküvidir. Bir kişi 1 den fazla basabilir. Örneğin 1. butona 2 kez, 3. butona bir kez basan kişi 2 su, 1 kahve alacaktır anlamına gelir. Seçim işlemi bittiği zaman Bitiş butonuna basılacaktır. Tüm bu işlemlerden sonra 1-4 (1 ve 4 dahil) arasında random sayı üretilecektir. Eğer random sayı 2 olursa para sıkışma olacaktır. Para sıkışma işleminde kırmızı led yanacaktır, aksi durumda (random üretilen sayı 2 e eşit değilse) problem olmadığını belirten yeşil led yanacaktır. Para sıkışma durumunda kişiye para iadesi yapılacak ve tuşlanan ürünler stoktan düşmeyecektir. Tüm bu kontrollerden sonra Reset butonuna basılacaktır. 6. Son olarak kullanıcıya para üstü verilecektir. Otomat kasada bulunan bozuk paralardan en azını kullanarak kullanıcıya bozuk para verecektir. Örneğin otomata 1 TL atıldı ve ürün olarak su istendi. Bundan sonra eğer kasada varsa ilk önce 1 adet 50 kuruş geri verilecektir. Eğer yoksa 2 adet 25 kuruş verecektir. Kasada sürekli kontrol ve güncellene yapılacak ve güncellemeler urunler.txt dosyasının ilk satırında da güncellenecektir. Projemizde olan senaryolardan biri de kasada yeteri kadar para olmamasıdır. Örneğin kasada 75 Kuruş varsa ve kişiye 1 TL iade edilmesi gerekiyorsa ekranda (Display) "Kasada yeterli para yoktur" uyarısı gösteriliyordur.

# Giriş

Günümüzde kullanıcıların farklı isteklerini karşılayabilmeleri için çeşitli programlar geliştirilmektedir. Bu projede amacımız; bir otomat makinesinin minimum sayıda bozuk para üstü vererek çalışmasıdır.

**Temel Bilgiler**

Projenin tasarlama aşamasında sanal makinede açtığımız Eclipse idesi kullanılarak, tiva kartın kullanımı için gerekli kütüphaneler eklenmiştir.

# Geliştirilen Mimari

**3.1 Kullanılan sınıflar, metotlar ve bazı kütüphaneler:**

void Lcd\_Temizle(void);

lcd de yazılanları temizler

void Lcd\_init(void);

void Lcd\_Putch(unsigned char,int type);

void PortD\_init() :Ledlerin Bağlı Olduğu Portun ayarları burada yer alır.

struct urunler: Ürünlerin Id ve fiyat bilgisini tutar.

#include "inc/hw\_ints.h"

#include "inc/hw\_types.h"

#include "inc/hw\_memmap.h"

#include "driverlib/sysctl.h"

#include "driverlib/gpio.h"

#include <stdint.h>

#include "inc/tm4c123gh6pm.h"

**3.2 Karşılaşılan problemler ve çözüm yaklaşımları**

Karşılaşılan en büyük problem breadboard üzerine devre kurarken tiva kartın hata verip kilitlenmesiydi. Kilitlenen kart birdaha çalışmayıp kartı tekrar açmaya uğraştık. Bu konuda sistem programlama hocamızın duyurulara attığı kilitlenme sorunuyla ilgili doküman sayesinde tekrar kartımızı çalıştırdık.

2.problem ise para şıkışma durumunda yanan kırmızı ve yeşil ledlerin yanması konusunda kodları yazarken zorlandık fakat internetten araştırdığımız gerekli led yakma kodları sayesinde bu problemi de kısa sürede çözdük. Para üstü verirken en az sayıda banknot hesaplamak için nesneye yönelik dersinde ilk projede yaptığımız gerekli bilgilerimizi tazelemiş olduk.

**3.3 Yazılım Geliştirme İçin Harcanan Süreler (kişi ve saat bazında)**

Yaklaşık kişi başı 40 saat harcanmış ve gerekli araştırmalar yapılmıştır.

**3.4 Projenin ek özellikleri:**

Kullanıcının yanlış seçim yapma ihtimaline karşı Reset butonumuz vardır. Reset butonu yapılan tüm seçimlerin iptal edilmesini sağlamaktadır.

**Örnek Kodlar:**

void Lcd\_Goto(char x, char y){

if(x==1)

Lcd\_Putch(0x80+((y-1)%16),0);

else

Lcd\_Putch(0xC0+((y-1)%16),0);

}

void Lcd\_Temizle(void){

Lcd\_Putch(0x01,0);

SysCtlDelay(10);

}

void Lcd\_Puts( char\* s){

while(\*s)

Lcd\_Putch(\*s++,1);

}

atilanPara=(float)(tempbirTL\*1+ tempelliKurus\*0.50 + tempyirmibesKurus\*0.25);

int para1= atilanPara;

int para2= ((int)(atilanPara\*100)%100)/10;

int para3= ((int)(atilanPara\*100)%100)%10;

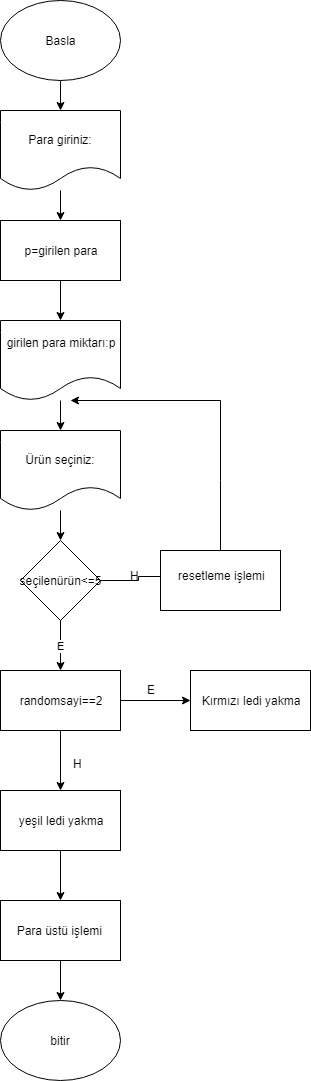
**Projenin Sistem Proglamlama Dersi ile İlgili Kısmı**

Bu derste kullandığımız tiva kartın özellikleri sayesinde breadboard üzerinde gerekli devre kullanarak kodumuzu yazmış olduk.

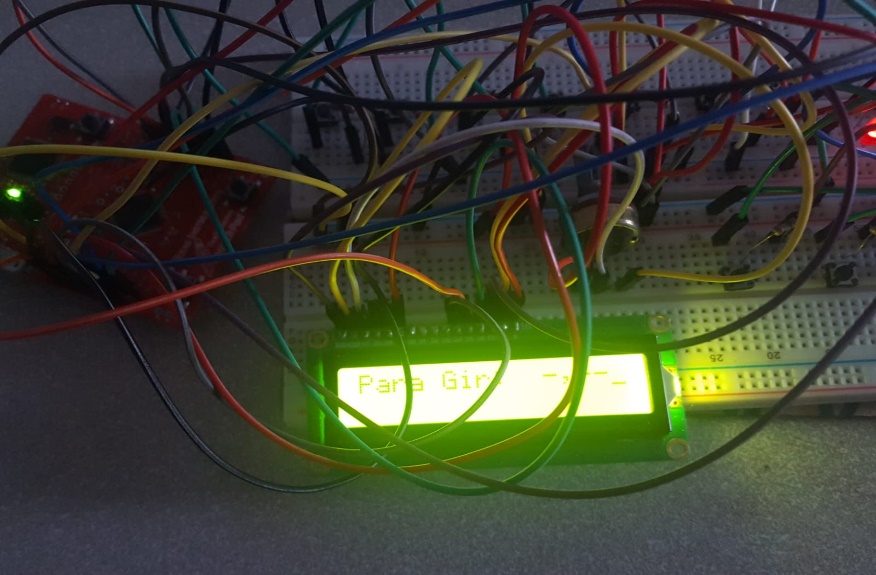
**Sonuçlar**

Uygulama içerisinde bulunan tüm fonksiyonlar istenildiği şekilde çalışmaktadır. Projemiz amacına ulaşmış olup, para üstünü en az biçimde doğru bir şekilde vermektedir. Para sıkışması olduğu durumda kırmızı led yanmakta olup, para verme işleminde sıkıntı olmadığı zamanlar yeşil ışık yanmaktadır. Devremiz buton ve ledlerden oluştuğu ve karmaşık bir yapıda olduğu için 2 adet breadboard üzerine kurulmuştur.

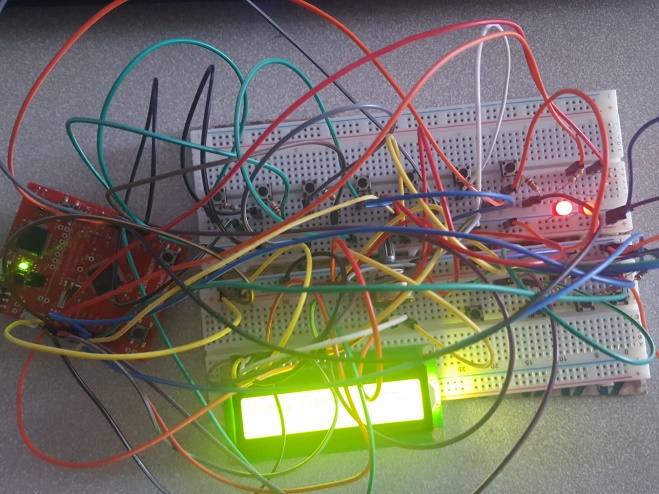
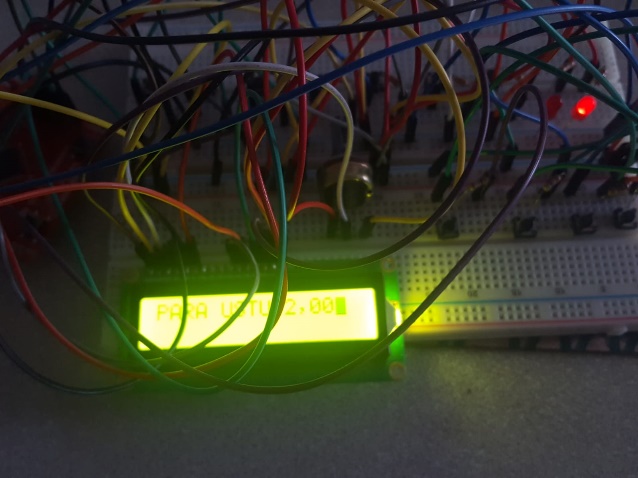
**ALGORİTMA ŞEMASI**



**EKRAN ÇIKTILARI=**

****

****

** **

**KAYNAKÇA=**

1. <http://www.teknikyazi.com/tiva-not-4-tiva-launchpad-osilator-ayarlari-io-kullanimi.html>
2. <http://www.resuldolaner.com/arduino_ile_lcd_kontrolu/>
3. <http://robotiktak.com/arduino-6-lcd-ekrana-yazi-yazdirma/>
4. <https://www.yazilimkodlama.com/etiket/c-para-ustu-hesaplama/>
5. <https://www.frmtr.com/c-ve-c-/3145303-atm-programi-c-dilinde-yapilacak-yardim-pls.html>
6. <https://docplayer.biz.tr/28038619-Sistem-programlama-deney-1.html>
7. <http://hayaletveyap.com/arduino-ile-push-buton-kullanimi/>
8. <http://forum.arduinoturkiye.com/index.php?p=/discussion/5878/butonla-doengue-kontrolue/p1>
9. <https://www.bilgigunlugum.net/prog/cprog/c_dosya>
10. <http://bilgisayarkavramlari.sadievrenseker.com/2008/10/22/c-ile-dosya-islemleri/>
11. <https://www.muhendisbeyinler.net/forum/konu/c-programlama-dili-dosyadan-veri-okuma-sorunu.1871/>
12. <https://www.bilisimogretmeni.com/visual-studio-c/visual-studio-c-uygulamalari-para-ustu-ve-banknot-adeti-hesaplama.html>