

DC Güç Kaynağı

- 1 V – 36 V arası, potansiyometre yardımıyla ayarlanabilir çıkış gerilimi.

Cisim tespit projesi

- STM32F407 MCU kullanıldı.
- PWM sinyali ile servo motorlar kontrol edildi.
- HCSR04 ultrasonik mesafe sensörü ile cismin merkeze olan uzaklığı ölçüldü.
- Uzaklık bilgisi UART arayüzü ile dönüştürücü kullanılarak bilgisayara gönderildi.
- Matlab ile alınan uzaklık verisi ve servo motorun açısı bilgisi kullanılarak 2 boyutlu görsel arayüz oluşturuldu.

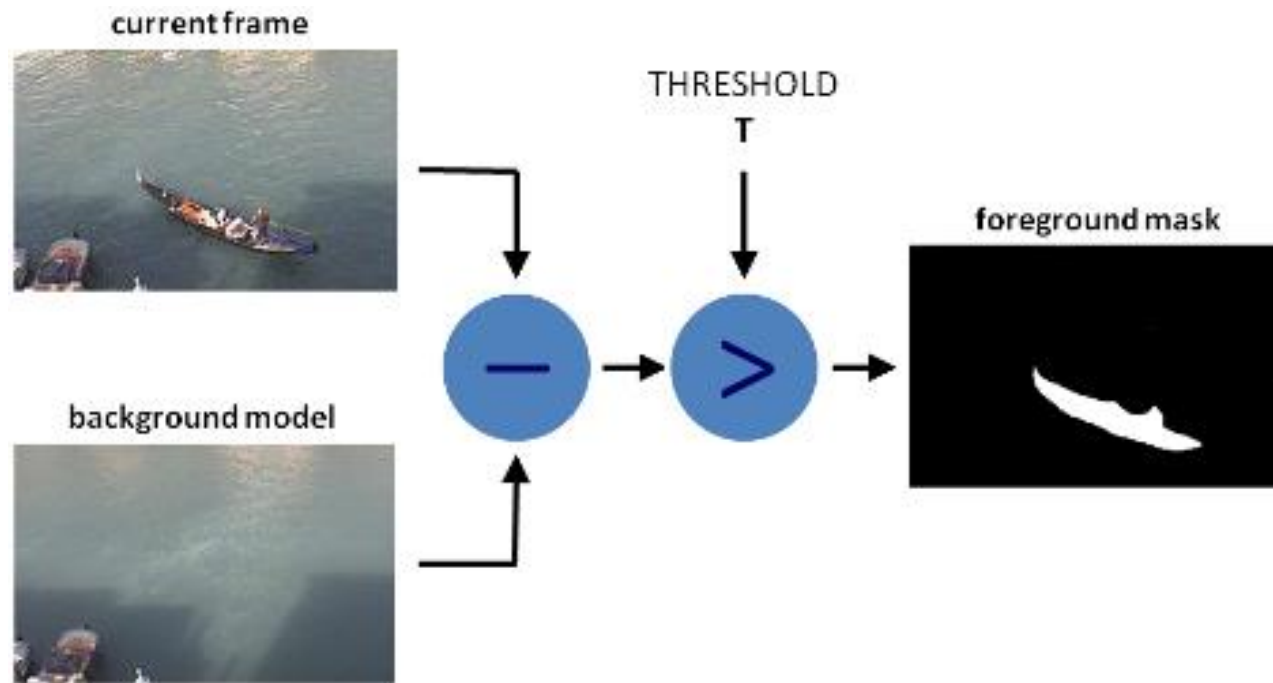


Araç Sayma Projesi

- OpenCV kütüphanesinin metotları kullanıldı.
- Bir bildiri takip edilerek verimli bir araç sayma yöntemi kullanıldı.
- ROI oldukça küçük belirlenerek bir çerçevenin işleme süresi oldukça düşürüldü.
- Çıkarılan hareketli nesnedeki gürültüleri önlemek ve yan yana giden araçları net bir şekilde birbirinden ayırmak için morfolojik işlemler uygulandı.



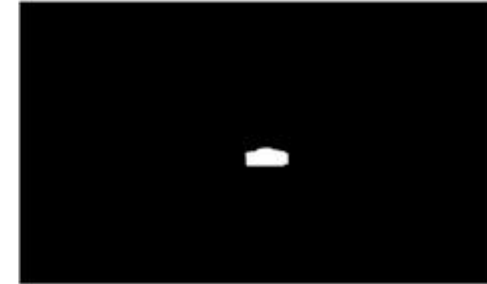
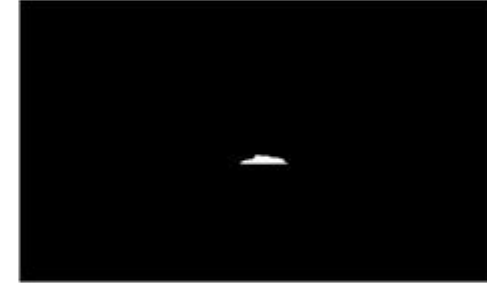
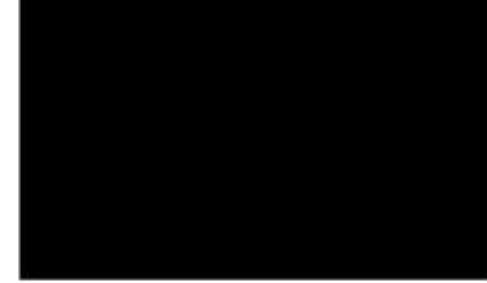
Ön plan maskesini elde etme (Hareketli nesneyi bulma)



Kaynak: https://docs.opencv.org/4.x/d1/dc5/tutorial_background_subtraction.html

Verimli Araç Sayma Yaklaşımı

- Bu yöntem herhangi bir tracking (nesneyi izlemeye devam etme) tekniği kullanılmadan yapılmış. Kesilmiş alana (ROI) giren nesneyi sürekli olarak takip etmemek için bir çözüm geliştirilmiş.
- ROI'ye giren nesnenin alanları her yeni frame için bir önceki frame ile kıyaslanıyor. Aradaki değişim belirli bir eşik değerinden fazla ise ROI'ye yeni bir araç girmiş demektir.



a

b

Verimli Araç Sayma Yaklaşımı

```
0  
0  
20  
750  
750  
750  
750  
940  
950  
1040  
1050  
1150  
1090  
1230  
1230  
1000  
1000  
950  
1000
```

```
if (vh1.area - p_vehicle.area > 550):  
    |   vh1.label += 1
```

Araç sayma doğruluğu

Video adı	Gerçek sayı	Sayı	Doğruluk (%)
Highway	27	29	92.31
Highway II	46	44	95.65
NUIII	1168	1115	95.46

$$\text{Error (\%)} = \frac{|Estimated - True NO|}{True NO} \times 100$$

$$\text{Presicion (\%)} = 100 - \text{Error (\%)}$$

Frame	Gerçek sayı	Sayı	Doğruluk (%)
1000 frame için	15	15	100.00
3000 frame için	44	42	95.45
5000 frame için	70	69	98.57

Ortalama işlenme süresi (Her bir frame için)

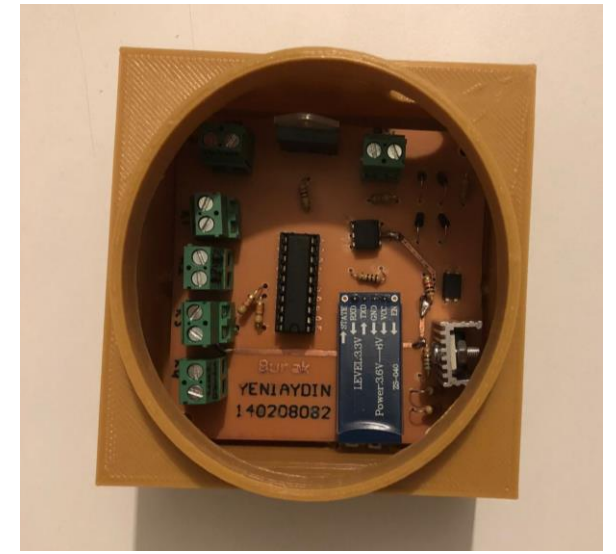
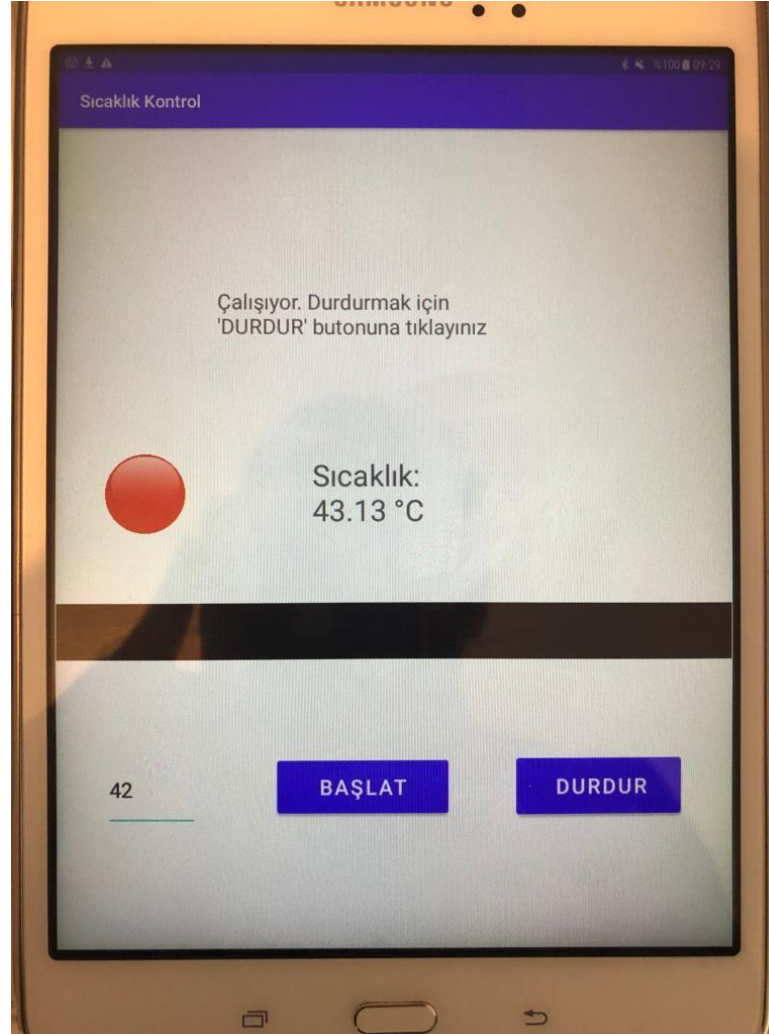
Video name	Avg. execution time (ms)
Highway	5.25
Highway II	5.27
M-30	7.50
M-30-HD	10.98
NU III	9.92
Average	7.78

```
PS C:\Users\Burak\Desktop\İmge sunum> py .\vehicle_counting.py  
Ortalama işleme süresi 0.005303
```

Çalıştığım video 60 fps.
 $60 * 0,005303 = 0.31818$ s
Program real time olarak çalışıyor.

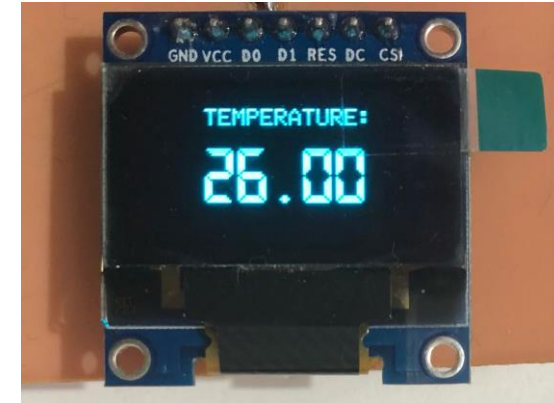
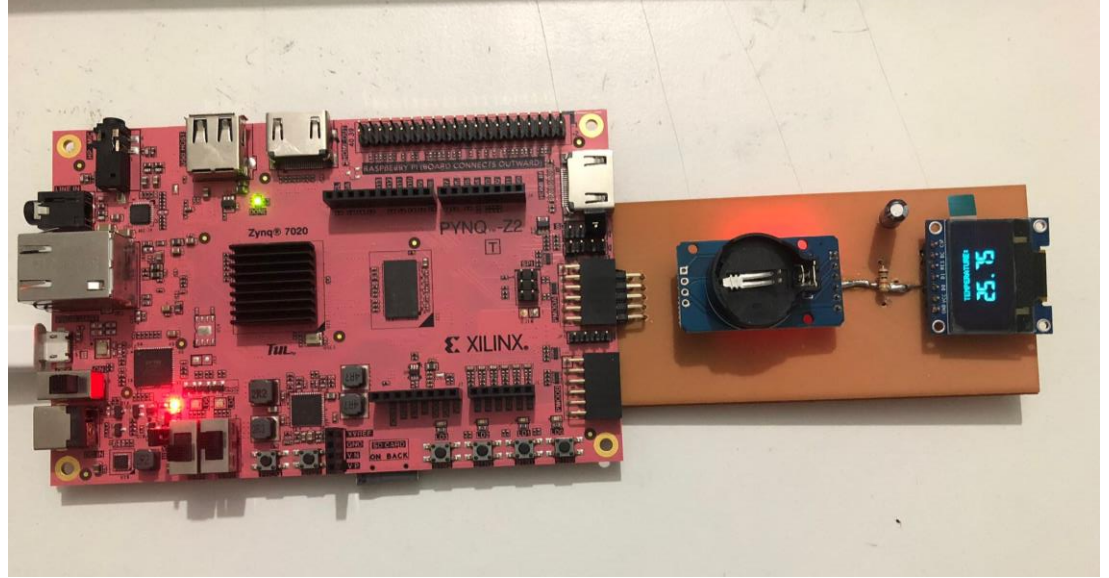
Sıcaklığı Ayarlanabilir Su Isıtıcı

- Android Studio ortamında Java dili kullanılarak Android uygulama tasarlandı.
- 220 V ile aç kapa mantığıyla çalışan ısıtıcı HW&SW eklentisiyle çalışma performansı ayarlanabilir hale getirildi.
- PID kontrol algoritması MCU üzerinde gerçekleştirildi.
- HC-06 Bluetooth modülü, DS18B20 su geçirmez sıcaklık sensörü, MSP430 mikrodenetleyicisi ve diğer elemanları iki katlı baskı devre haline getirildi.



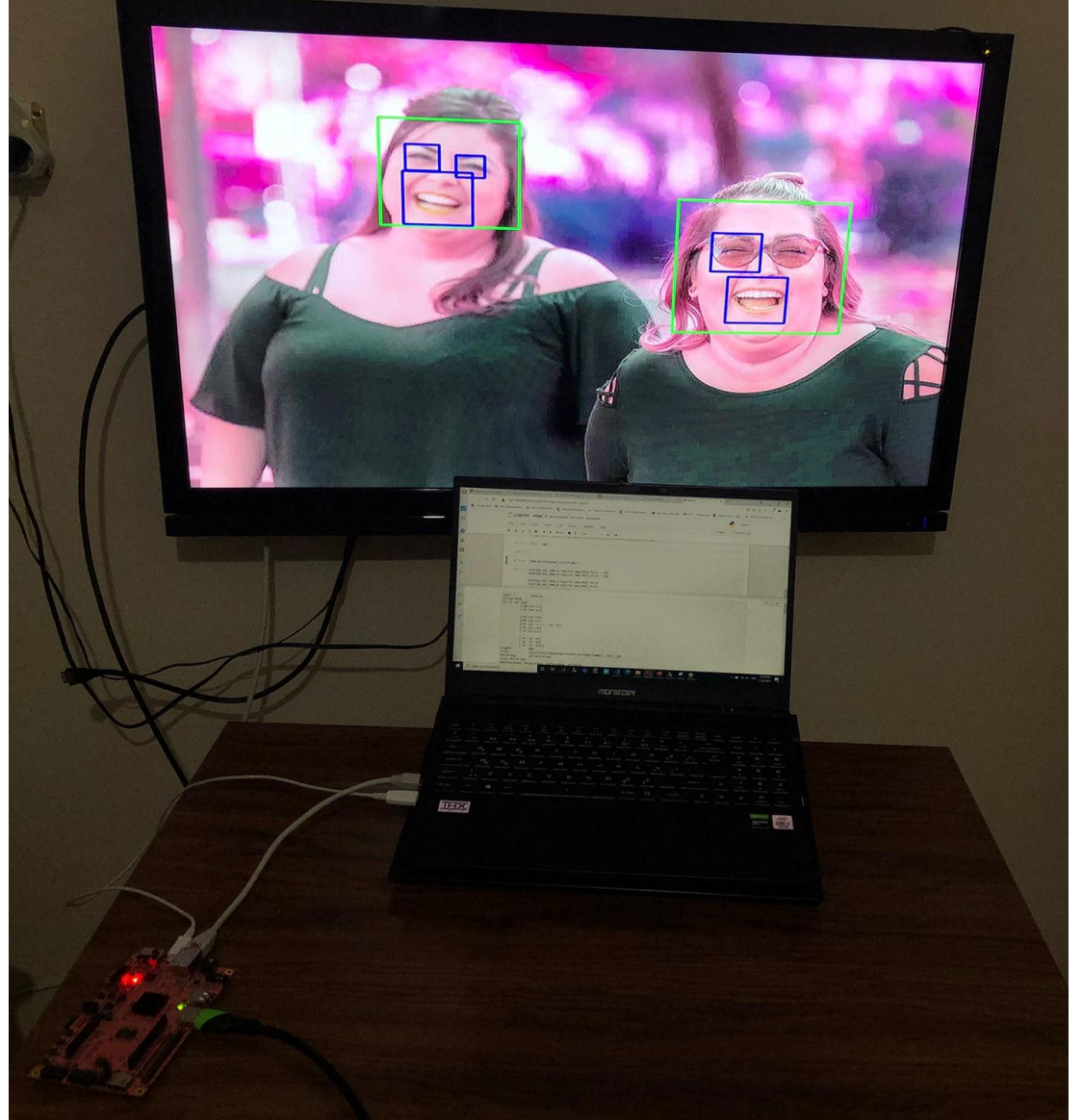
FPGA ile OLED'de ortamın sıcaklık bilgisini görüntüleme

- FPGA ile I2C arayüzüne sahip DS3231 sensörü ve OLED kontrol edildi.
- Ortam sıcaklığı DS3231 modülünden okunarak OLED üzerinde gösterildi.
- FPGA Verilog HDL kullanılarak programlandı.
- OLED'de bastırılan her bir karakter BRAM'de tutuldu.
- Testbench kodları yazıldı.



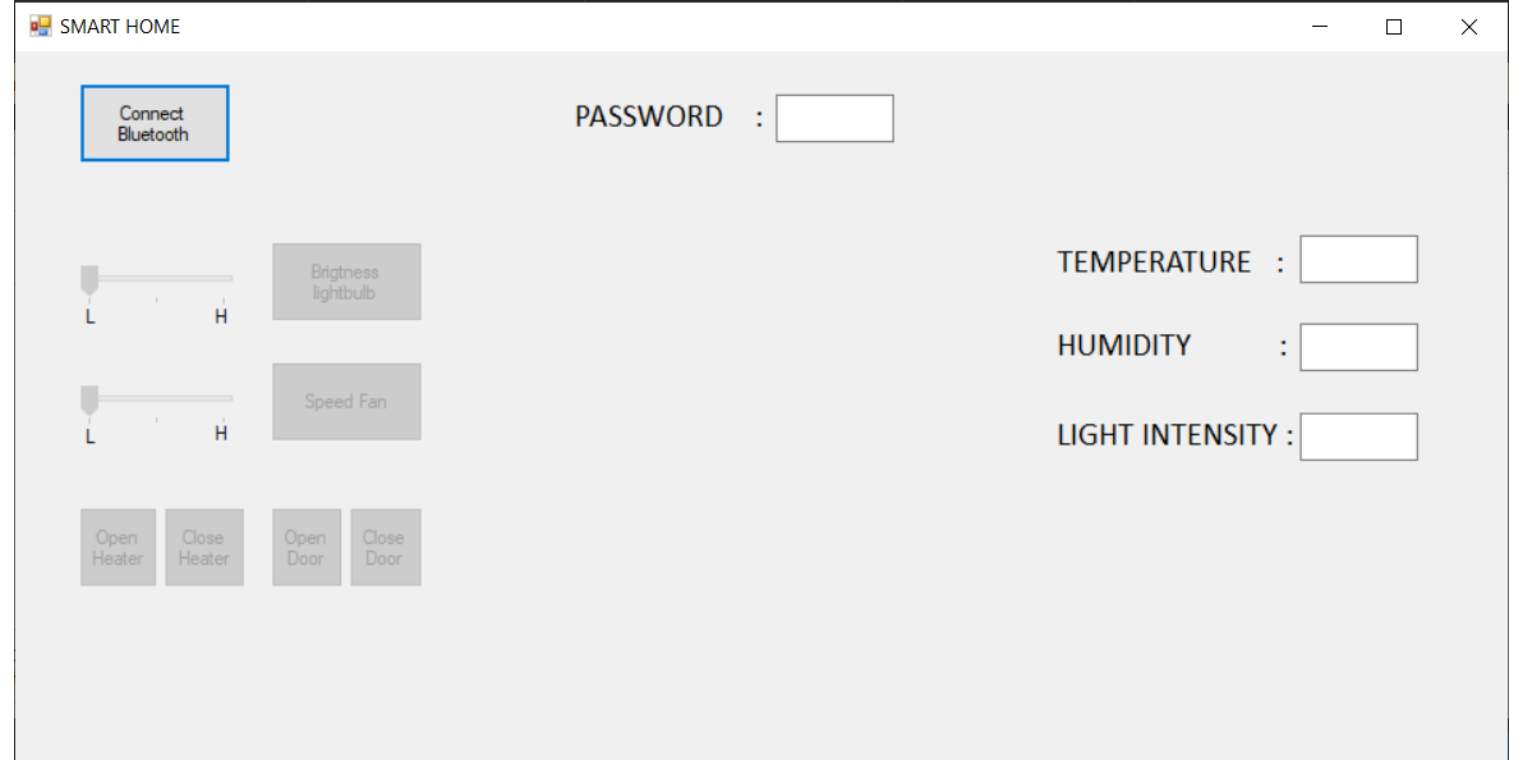
Pynq Z2 ile Face Detection uygulaması

- Pynq Z2 geliştirme kartı, içerisinde hem PL hem de PS tarafını barındıran Zynq7020 çipine (SoC) sahiptir.
- PS kısmında SD kartın içerisine yüklenen görüntü üzerinde Python ile yazılmış olan face detection kodları çalıştırıldı.
- PL kısmında, bazı Xilinx IP çekirdeklerini kullanarak PS tarafında işlenmiş olan görüntü kart üzerindeki HDMI çıkışı ile ekrana basıldı.



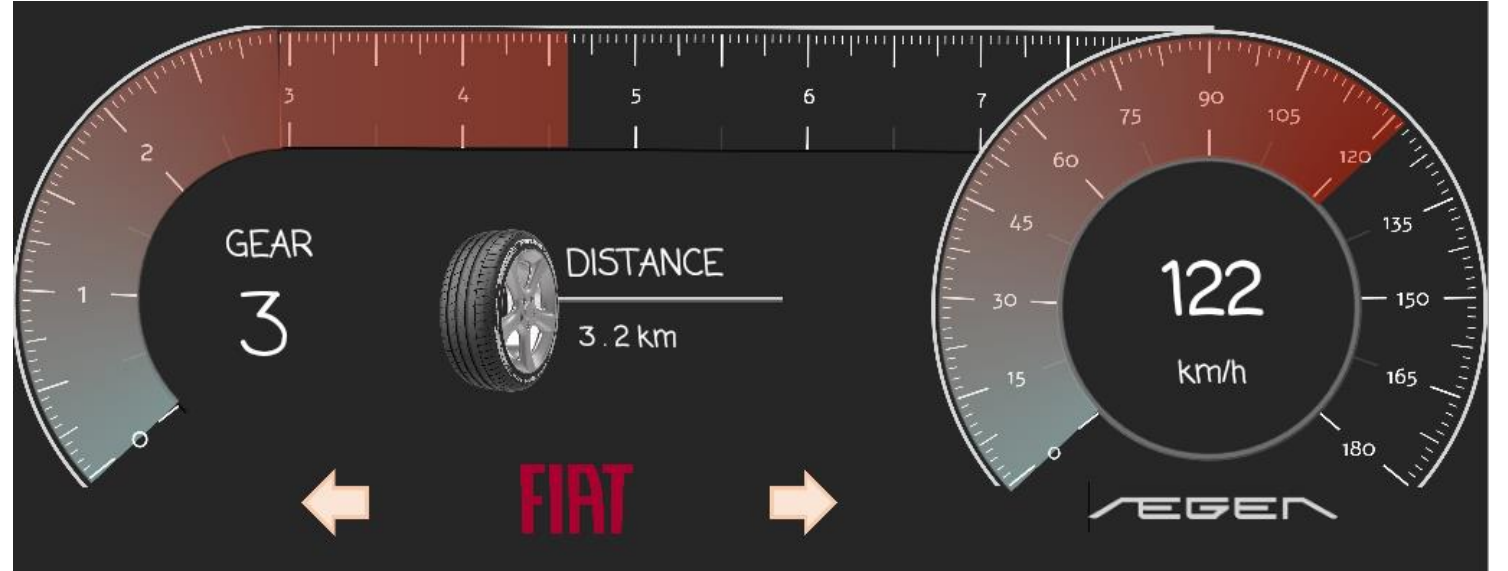
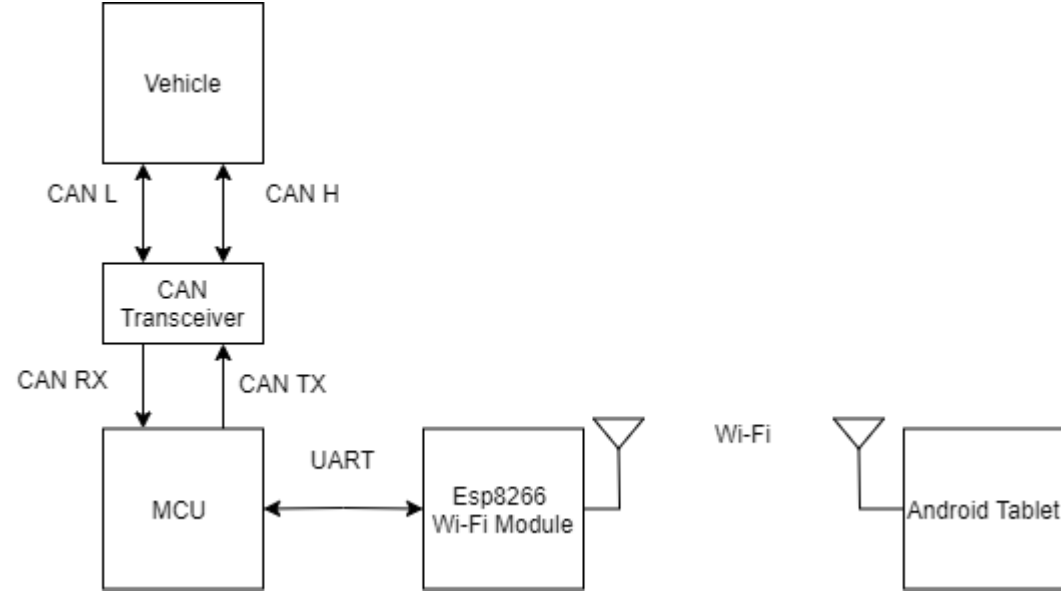
Akıllı Ev Projesi

- Kullanıcı kontrollü akıllı ev projesi gerçekleştirildi.
- Visual Studio ortamında C# ile kullanıcı arayüzü oluşturuldu.
- STM32F407 MCU ile HC-06 Bluetooth modülü, Keypad, 220V ampul, 12V fan ve ısıtıcı, servo motor, sıcaklık sensörü gibi bir çok elektronik bileşen kontrol edildi.



Aracın CAN-Bus verilerinin gerçek zamanlı gösterimi

- Aracın CAN-Bus'ındaki verileri alıp ilgili cihaza gönderebilecek HW&SW geliştirildi.
- ESP8266 MCU'dan komutlar gönderilerek AP (Access Point) modda çalıştırıldı.
- Emniyet kemeri takılı olup olmaması, Airbag'in açık veya kapalı olması, gaz fren debriyaj pedallarının analog değeri gibi bir çok bilginin hangi ID ile birlikte geldiği bulundu.



DİNLEDİĞİNİZ İÇİN TEŞEKKÜRLER