**8. Araç Kontrol Sistemi (AKS) ve Telemetri Sistemi**

Araç kontrol sistemi arabanın seyri esnasında güvenliği sağlamak adına gaz ve freni kontrol edebilen, sürüş modlarını ayarlayan ve sürüş performansı üzerinde etkili olan sistemdir. Araç kontrol sisteminin aracın tüm kısımlarını doğru bir şekilde koordine etmek için tasarlanmış alt sistemleri bulunur ve bu mekanizma araçtaki sistemlerin birbirleriyle uyum içerisinde çalışmasını sağlar. Telemetri Sistemi, araç içindeki verilerin dışarıya kablosuz aktarımından sorumludur.

**8.1 Araç Kontrol Sistemi**

**8.1.1 Araç Kontrol Sistemi Ana Fonksiyonları**

**a) Motor Tork Kontrolü**

AKS’de pedaldan ve bataryadan alınan veri motor sürücüye iletilir. Motor sürücü de işlenen veri tork kontrolü için kullanılmaktadır. Batarya voltajı ve sürücünün pedal girdisine bakarak uygun tork değeri motor sürücüye gönderilir.

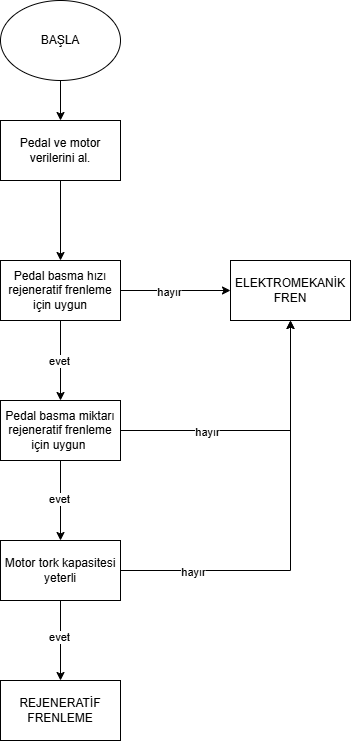


*Figure 1 Motor Tork Kontrolü hakkında blok diyagramı*

**b) Geri Kazanımlı Frenleme Optimizasyonu**

Elektrikli motorların verimini arttırmak harcanan enerjinin bir kısmının batarya geri dönmesini sağlayan optimizasyondur.

Sürücü ayağını gazdan çektiğinde veya belli bir hız ve basınçta pedala bastığında rejeneratif frenleme gerçekleşir. Burada önemli noktalardan biri de bataryanın doluluk oranıdır. Eğer batarya yüzde yüze yakınsa rejeneratif frenleme gerçekleşemez. Çünkü motorun bataryayı şarj edebilmesi için bataryadan daha yüksek bir voltaj farkı oluşturabilmesi gerekir. Aşırı düşük hızlarda dururken rejeneratif frenleme gerçekleşmez çünkü motorun ürettiği gerilim düşer ve bataryayı şarj edebilecek kadar gerilim üretilemez.



*Figure 2 Rejeneratif frenleme akış diyagramı*

**c) Araç içi Haberleşme Sistemi**

Gösterge panelinin temel amacı kullanıcıya araç durumunu göstermektir. Gösterge panelinde hız göstergesi, batarya sıcaklıkları ve gerilimi göstergesi, kalan batarya miktarı, rejeneratif frenlemenin durumu ve arıza teşhisi gibi özellikler bulunur. Araç içi ağ ile algılayıcılardan CANBUS aracılığıyla elde edilen verinin, dizayn edilen gösterge paneli aracılığıyla kullanan kişilere aktarılması ve işletilmesi işlemleri gerçekleştirilmiştir

**Haberleştiği Birimler**

Pedal ünitesi, Motor Sürücü ünitesi, Batarya Kontrol ünitesi, İzolasyon İzleme cihazı, Kontrol konsolu, ADAS Kontrol Modülüdür. Bu birimlerden gelen mesajlara göre gösterge panelinde sürücüye haber veren kullanıcı dostu bir arayüz tasarlanmıştır.

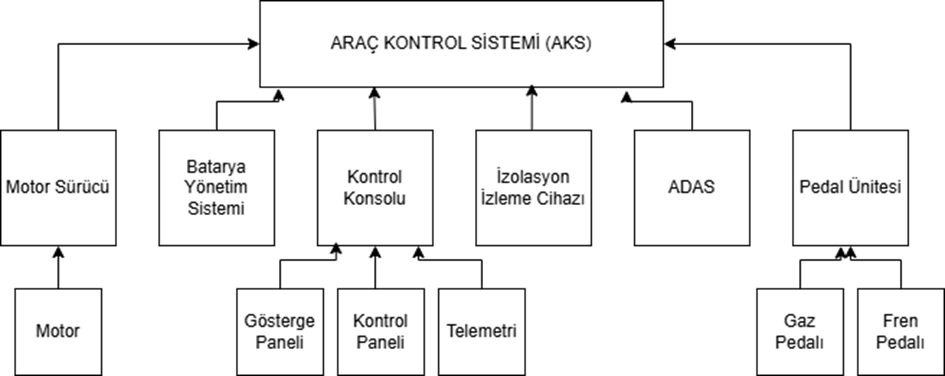
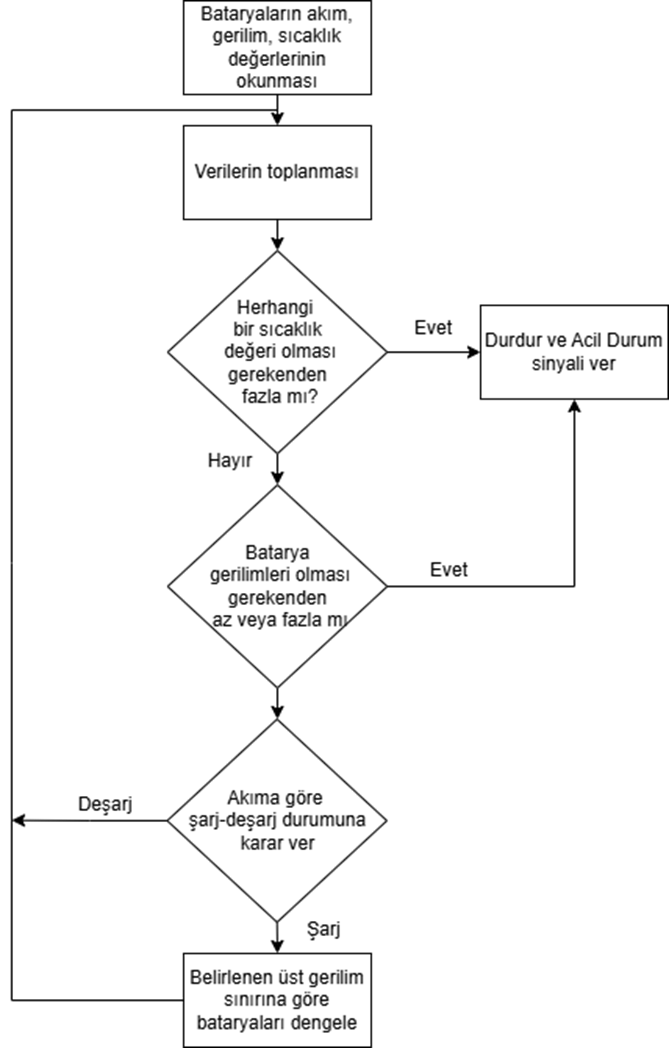


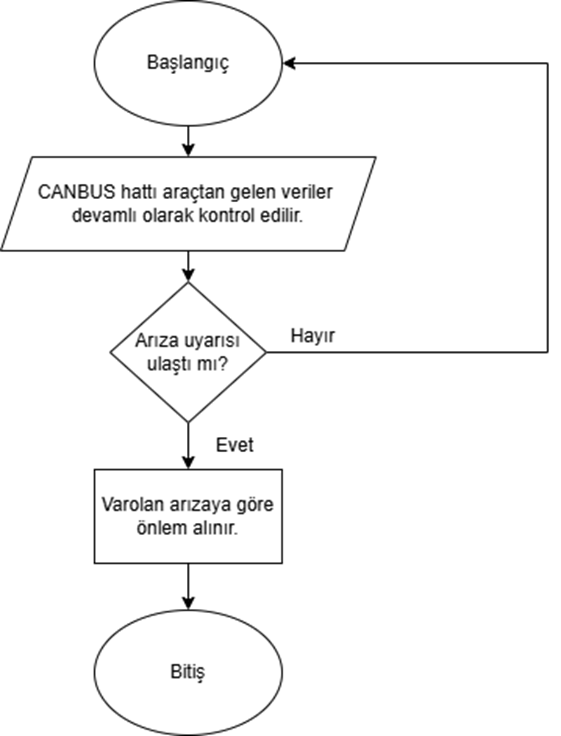
Figure 3 *AKS’ nin haberleştiği birimler hakkında blok diyagramı*

**d) Arıza Teşhisi**

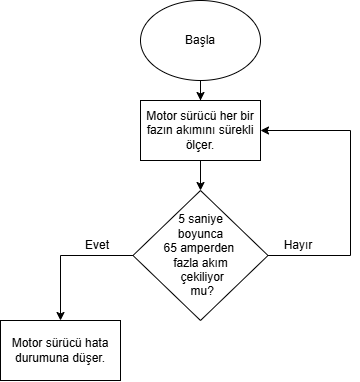
Batarya Yönetim Sistemi, motor sürücü gibi birimlerde oluşan arızaları bildirmek için vardır. Meydana gelebilecek arızalar ise aşırı akım, kısa devre, yüksek sıcaklık, aşırı gerilim, haberleşme kopuklukları gibi arızalardır.



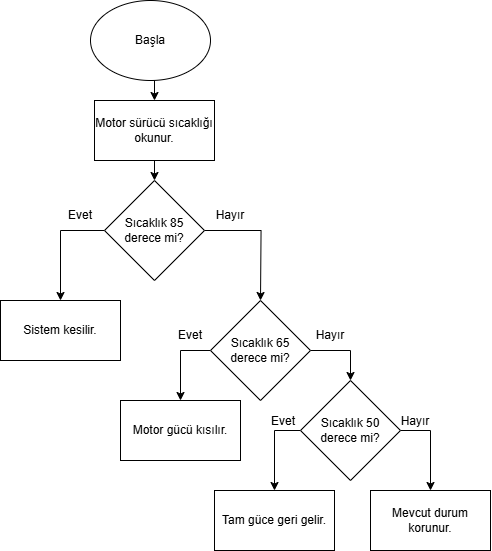
*Figure 4 Batarya Durumunun Arıza Teşhis Akış Diyagramı*



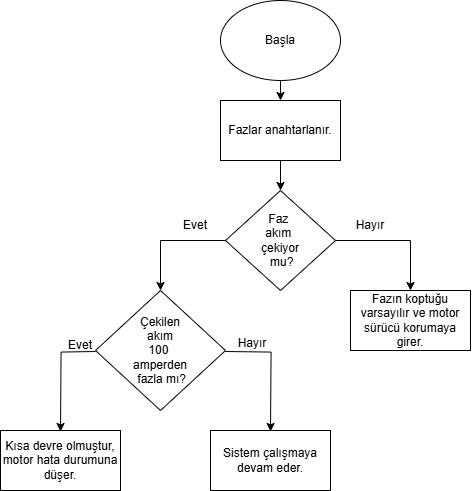
*Figure 5* *CANBUS Hattı Arıza Teşhis Akış Diyagramı*



*Figure 6 Motor Sürücü faz akımı Arıza Teşhis Akış Diyagramı*



*Figure 7 Motor sürücü sıcaklığı Arıza Teşhis Akış Diyagramı*



*Figure 7*

**f) Araç Durumunun izlenmesi ve kullanıcıya iletilmesi**

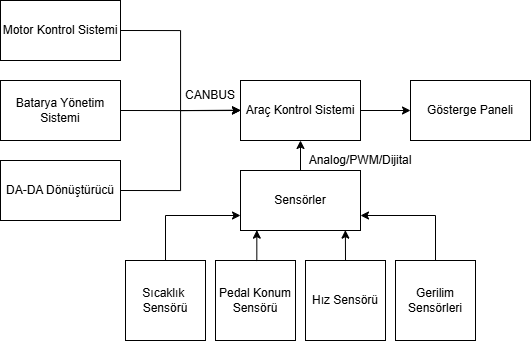
Bu araçta kullanılacak olan ekran Nextion NX1060P101-011C-I’dur. Nextion NX1060P101-011C-I, Nextion firmasının üretmiş olduğu dokunmatik özelliği olan bir TFT ekrandır. Proses, makine, cihaz arasında kontrol ve görselleştirme ile birlikte işlemciyi, bellek birimini ve dokunmatik ekranı tek bir platformda birleştiren bir HMI ürünüdür. Nextion ekran, donanım kısmında ise bir dizi TFT kartı ve yazılım kısmında ise Nextion’a ait olan kendi editöründen mevcuttur. Nextion Editor’de bulunan tut ve sürükle özelliği sayesinde grafik, buton, metin gibi bileşenler sayesinde tasarım kolayca yapılır. Ekran tarafını kodlamak için ASCII tabanlı komutlar bulunmaktadır.

**Özellikleri:**

* NX1060P101-011C-I (10.1 '' IPS RGB 65K kapasitif dokunmatik ekran)
* Ekran boyutu: 10.1
* Kart boyutu: 258\*152\*11.5 mm
* Ağırlık: 535 g
* Çözünürlük: 1024\*600
* Ekran Ömrü: 30.000 saat
* Güç Tüketimi: 5V / 2A
* Yerleşik MCU 200 MHz ve HMI projesi için 128 MB'a kadar
* Dahili RTC, EEPROM, 8 dijital GPIO (2 PWM özellikli)
* 4 pin (+ 5V, TX, RX, GND) TTL seri arayüz
* % 1 aralık ayarı ile 300 nit Parlaklık
* Ses arayüzü
* CE ve RoHS Uyumlu

**Tasarım**

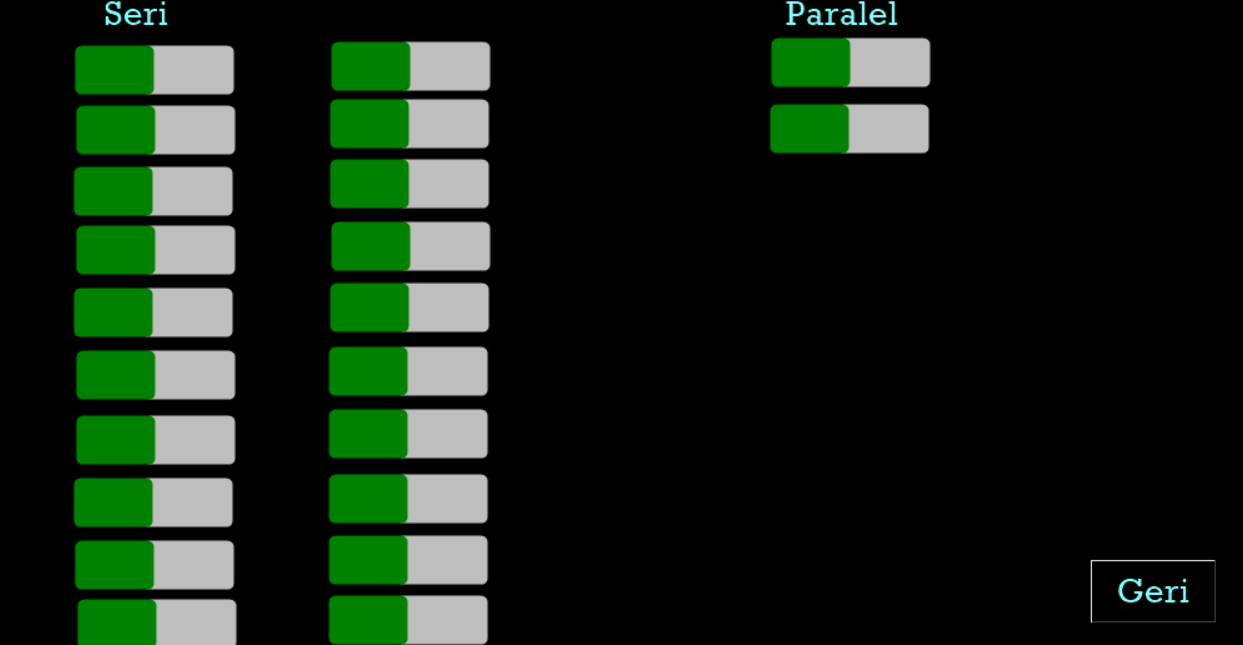
Nextion Editor, Nextion HMI cihazları için İnsan Makine Arayüzü GUI'lerini hızla oluşturmak için kullanılır.  Bileşenler ve sayfaların yükleme efektiyle, kullanıcı dostu dinamik ve mükemmel bir GUI en verimli şekilde oluşturulur. Tasarım yapılırken sayfa sayfa olacak şekilde düşünülüp modern ve kullanıcı için rahat kullanılan bir tasarım yapılmak amaçlanmıştır.

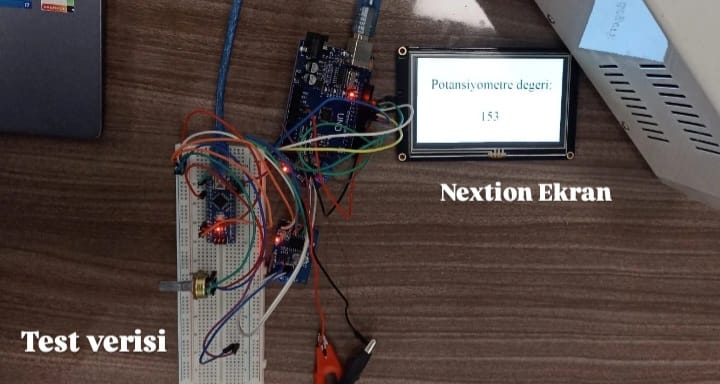


*Figure 8* *Gösterge Paneline gelen verilerin Blok Diyagramı*









Ekran test etme devresi

**Tablo 8.1.** Takım tarafından kendi tasarlanıp ve üretilen AKS’nin özellikleri.

|  |  |  |
| --- | --- | --- |
| **Özellik** |  | **Değer/Açıklama** |
| **AKS Fonksiyonları** | **:** | Motor Tork Kontrolü  Geri Kazanımlı Frenleme Optimizasyonu  Araç içi Haberleşme Sistemi  Arıza Teşhisi  Araç Durumunun İzlenmesi ve Kullanıcıya İletilmesi  Araç Verilerinin İzleme Merkezine Aktarılması |
| **Kontrolör Entegre Devresi** | **:** | ATmega328P mikrodenetleyici |
| **AKS Giriş/Çıkış sayısı** | **:** | 13 |
| **Elektronik Devre Tasarımı** | **:** |  |
| **Baskı Devre Kartı Tasarımı** | **:** |  |
| **Baskı Devre Kartı Üretimi** | **:** | Hala devam etmektedir. |
| **Yazılım Algoritması** | **:** | Gösterge paneli için Nextion Editor uygulaması kullanılmıştır. C/C++ kullanılmıştır. |
| **Deneysel Çalışma** | **:** | Ekran 5V 2A ile beslenip test edilmiştir. |
| **Boyut (PCB / Kutu)** | **:** | 10cm x 5cm |

**8.1.2 Devre Tasarımı**

**a) Araç Kontrol Sistemi Devresinin Tasarım Adımları**

Araç kontrol sistemi (AKS) devresi, elektrikli aracın sürüş esnasındaki kontrol ve güvenliğini sağlayan temel modülleri barındırır. Devre tasarımı aşağıdaki adımlarla gerçekleştirilmiştir:

1. **Fonksiyonel ihtiyaçların belirlenmesi:** Motor tork kontrolü, rejenaratif frenleme, haberleşme ve kullanıcıya veri sunumu gibi temel işlevler belirlendi.
2. **Mikrodenetleyici seçimi:** Arduino uyumlu, düşük güç tüketimli ve yeterli sayıda I/O hattına sahip **ATmega328P** seçildi.
3. Araç içi haberleşme protokolü olarak CAN-BUS altyapısı benimsendi. **MCP2515** CAN kontrolcüsü ve **TJA1050** sürücü entegreleriyle bu yapı oluşturuldu.
4. **Enerji dönüşümü ve besleme yönetimi**: 12V giriş gerilimi buck konvertörler ile 5V ve 3.3V’a düşürülerek sistemin beslenmesi sağlandı.
5. **İletişim birimleri**: LoRa modülü ile telemetri sağlandı. UART hattı üzerinden haberleşme gerçekleştirildi.
6. **Kullanıcı arayüzü geliştirme:** Nextion NX1060P101-011C-I ekran ile gösterge paneli oluşturuldu, tasarım Nextion Editor yazılımı ile geliştirildi.
7. PCB tasarımı **Altium Designer** üzerinde çizilerek üretime hazır hale getirildi.
8. Test ve simülasyon aşamasında örnek verilerle CAN-BUS ve LoRa üzerinden iletim testleri yapıldı.

**b) Malzeme Seçimi**

|  |  |  |
| --- | --- | --- |
| **Bileşen** | **Model / Değer** | **Açıklama** |
| **Mikrodenetleyici** | ATmega328P-AU | Sistem çekirdeği, düşük güç ve geniş destek |
| **CAN Controller** | MCP2515 + TJA1050 | CAN-BUS haberleşmesi için |
| **LoRa Modülü** | E32 433T20D | Telemetri için uzun menzil haberleşme |
| **Ekran** | Nextion NX1060P101-011C-I | HMI tasarım, kullanıcıya bilgi sunumu |
| **USB-UART Dönüştürücü** | CH340G | Programlama ve veri iletimi |
| **Gerilim Regülatörleri** | TLV1117-18, AP63200 | 12V → 5V / 3.3V dönüşümü |
| **EEPROM / RTC** | Dahili olarak Nextion ekran içinde | Hariciye gerek yok |
| **Kapasitörler** | 100nF, 10µF, 22pF, 100µF, 22µF | Filtreleme ve kararlılık için |
| **Dirençler** | 1kΩ, 10kΩ, 3.3kΩ, 120Ω terminatör | Pull-up/down, akım sınırlama |
| **Endüktörler** | 10µH | Güç regülasyonu (Buck) için |
| **Diyotlar** | SS16, PESD1CAN, SMBJ12A | ESD ve ters gerilim koruma |
| **LED’ler** | WL-SMCW 0805 Yellow / Green | Gösterge, durum LED’leri |
| **SD Kart Modülü** | MSD-1-A | Yedek veri kaydı için |
| **LoRa Anten** | 915 MHz 4.3 dBm SMA | Menzil artırımı |

*Tablo 1 Malzeme Seçimi*

**c) Devre tasarımına ait blok şemaları**

metin, ekran görüntüsü, diyagram, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

*Figure 1 Sistemin Genel Yapısı*

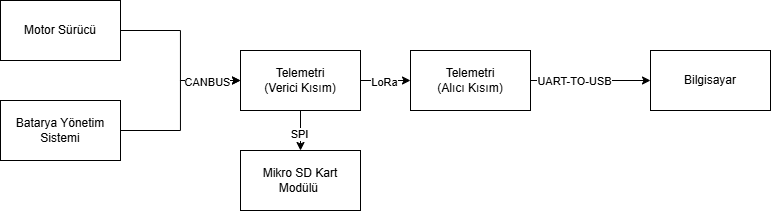
metin, diyagram, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

*Figure 2 Sistemin Uygulama Yapısı*

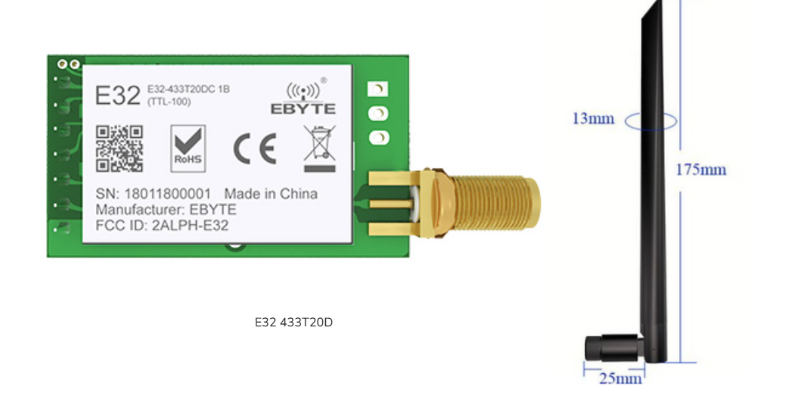
**8.2 Telemetri Sistemi**

Telemetrinin nihai amacı araç içindeki hız, batarya sıcaklığı, batarya voltajı ve bataryada kalan enerji miktarı gibi verileri aracın dışına, telemetri bilgisayarına, ulaştırmaktır. Araç hızı motor sürücüden ve diğer veriler BYS’den CAN hattına ulaşır. Bunun ardından, telemetri de bu verileri CAN hattında telemetri bilgisayarına aktarır.



CAN hattından verilerin alınması için *MCP2515* modülü, verilerin kablosuz bir şekilde araç dışına ulaştırılması içinse *LoRa E32 433T20D* modülü kullanılmıştır. Bu modül radyo frekansı üzerinden kablosuz haberleşme gerçekleştirir. *E32 433T20D* modülü 433Mhz bandını kullanır ve açık alanda menzili 5 kilometredir. Geliştirme raporundan itibaren modül değişikliğine gidilmiştir çünkü *E32 433T20D* modeli *E32 433T30D* modeline göre daha düşük seviye de enerji harcamakta ve daha dengeli bir veri aktarımı gerçekleştirmektedir. Güvenlik ve enerji verimliliği nedeniyle *E32 433T20D* modeli amacımıza daha uygun olduğunu belirleyip bu modeli kullandık.

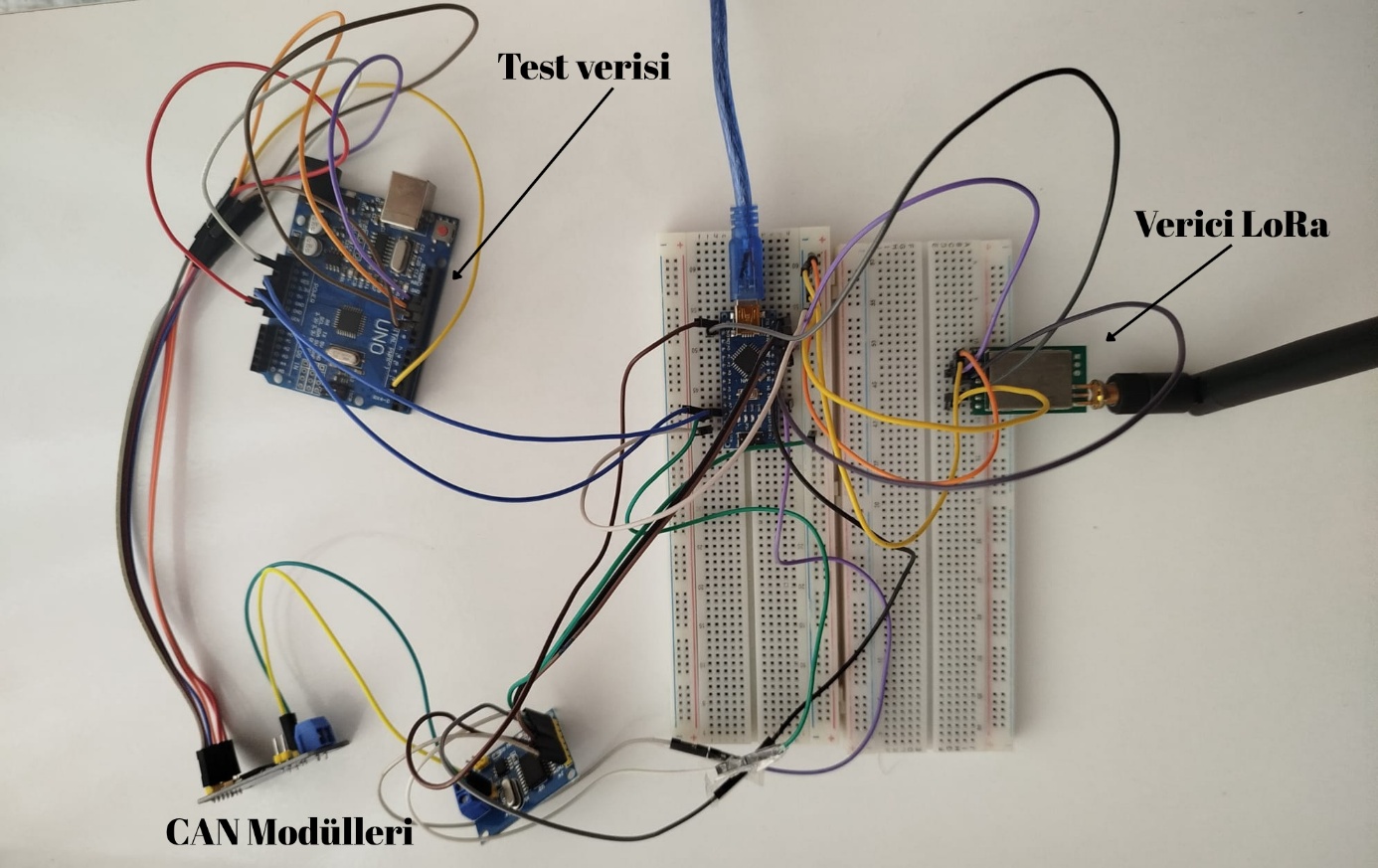
*LoRa* modülünün menzilini arttırmak ve daha güçlü sinyal oluşturmasını sağlamak için modüllerin frekansıyla uyumlu 433MHz frekansta çalışan band aralığı 410-450MHz olan *TX433-JKD-20P* model anten kullanılmıştır. Bu model oldukça hafiftir (21 gram) ve daha az enerji ile daha uzun menzillere veri ulaşımını sağladığı için oldukça verimlidir.

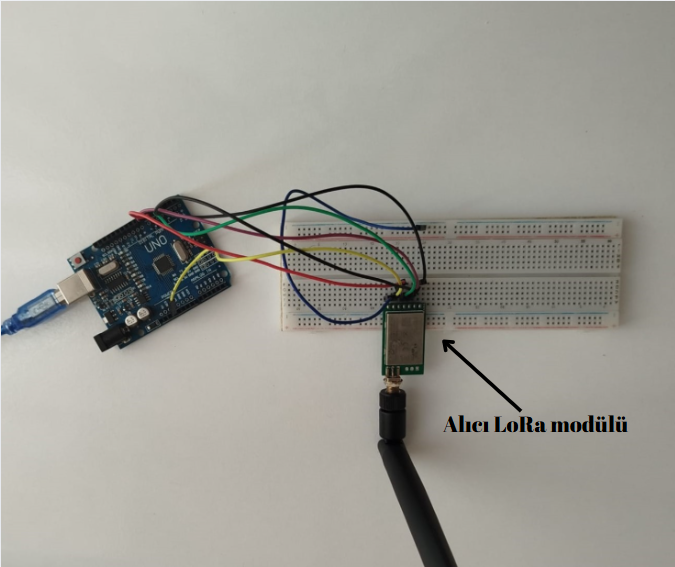


*LoRa* modüllerini kendi aralarında haberleştirmek için *Fixed (Sabit İletim) Modu* kullanıldı. Bu modda *LoRa* modüllerine kendilerine özgü adresleme yapıldı ve aynı kanal üzerinden haberleşmesi sağlanır. Her modüle 16 bitlik bir adres atanır. 16 bitlik adres 2 parçadan oluşur, *ADDRH* ve *ADDRL*. Veri aktarımı öncesinde her iki parça da koda entegre edilir. Bunların yanında koda haberleştirdiğimiz kanal numarası da eklenir. Bu sayede bir adresten diğerine aynı kanal üzerinden veri aktarımı yapmış oluruz.

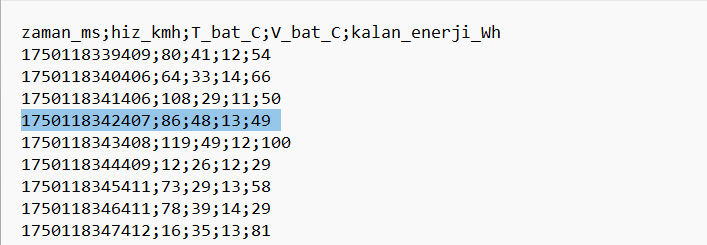
Sistemi araç içine yerleştirmeden önce hata tespiti yapabilmek adına sistemi simüle ettik. Bu aşamada oluşturduğumuz test verisini alıcı *LoRa*’ ya göndererek GUI da tüm verilerin doğru gösterilip gösterilmediği ve bilgisayar üzerinden verilerin kaydedilmesi denenmiştir.

Sistem bu aşamada şu şekilde görünmektedir:





Haberleşme tamamlandığında bilgisayardaki dosyaya kaydedilen veriler aşağıdaki gibi görünmektedir.

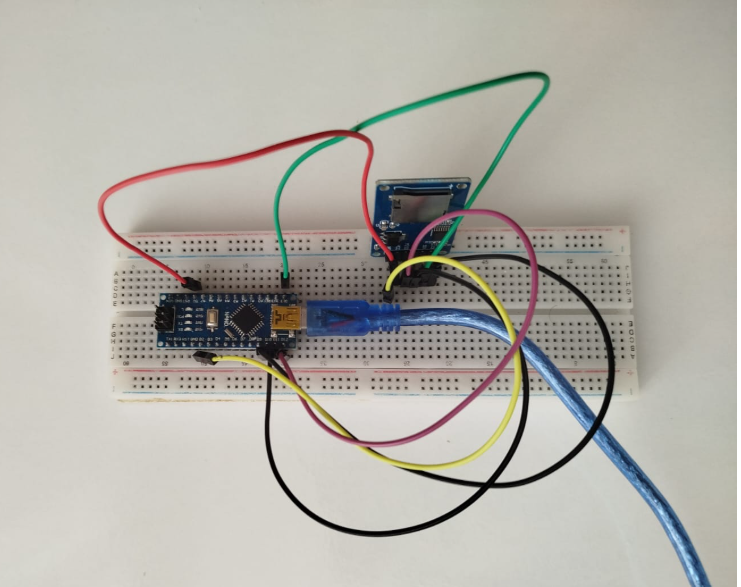


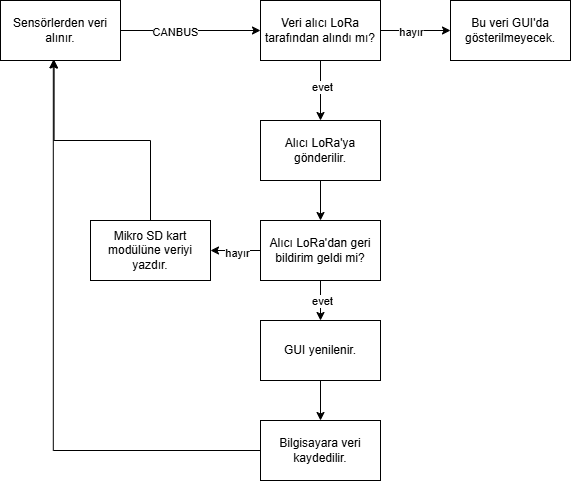
Telemetri bilgisayarı tarafında ise durum biraz daha farklıdır. Verileri alan *LoRa* modülü *USB-TTL dönüştürücü* ile bilgisayara bağlanır. Veriler *UART* ile bilgisayara aktarılır ve telemetri arayüzünde gösterilir.

Telemetrinin arayüzü *Qt Designer* ile tasarlanmış ve *Pyhton* dili ile programlanmıştır. *Qt Designer’da* hazırlanan *ui* dosyasını *Pyhton* koduna entegre edebilmek için *uic* kütüphanesi kullanılmıştır. Tasarım temelde *png* dosyasının üzerinde konumlandırılmış dijital sayı göstergelerinden ve veri akışını kontrol eden bir durum göstergesinden oluşmaktadır.*PNG* dosyasının entegresi için *QPixmap* , verilerin CANBUS üzerinden okunabilmesi için *QThread* ve *GUI’ nin* donmadan sürekli yeni verileri gösterebilmesi adına *pyqtSignal* kütüphaneleri eklenmiştir.



Tüm veri aktarımı doğru ve eksiksiz bir şeklide gerçekleşirse araç dışından araç içinde geri bildirim gönderilir fakat bu geri bildirimin gelmemesi durumunda veriler aracın içindeki SD kartta depolanacaktır.SD karta verileri yazdırma işlemi için mikroişlemci ile mikro SD kart modülü arasında *SPI* protokolü kullanılır. *SPI* protokolünü algoritmaya dahil etmek için *SPI.h* , mikro SD kart modülü üzerinde işlem yapabilmek için de *SD.h* kütüphanesi *Ardunio IDE* üzerinde kullanılır.

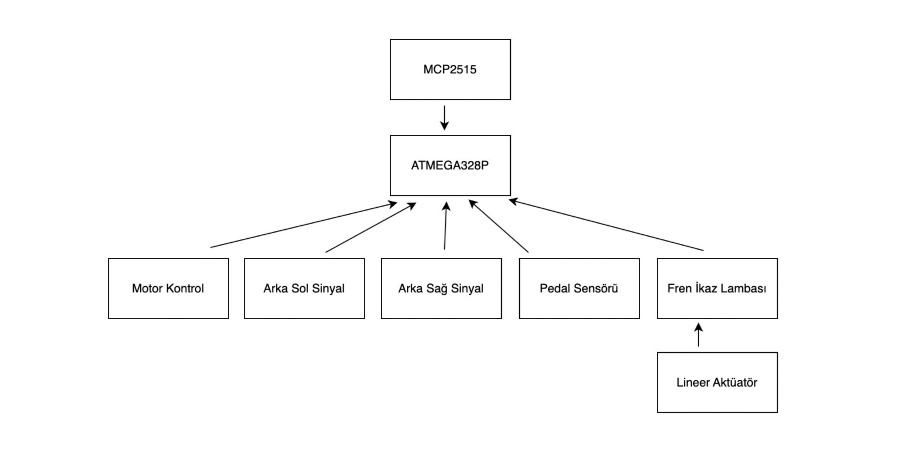




Telemetri sisteminin akış diyagramı

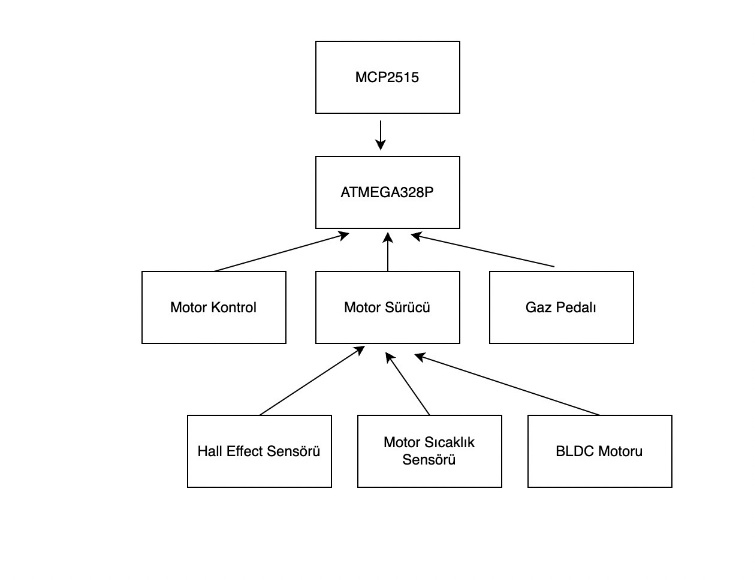
Canbus Blok Diyagramları

AKS



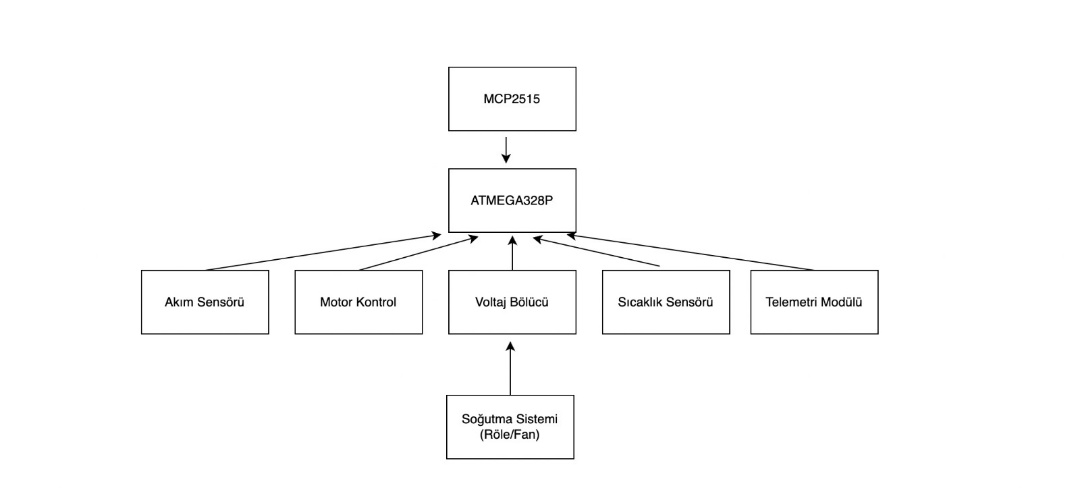
ATMEGA328P mikrodenetleyici, MCP2515 CAN modülünden gelen mesajları AKS birimi ile işler. Pedal sensörü, fren ikaz lambasını tetikler ve lineer aktüatörü kontrol ederek mekanik bir frenleme yapabilir. Mikrodenetleyici, motor kontrol sistemini ve arka sağ ve sol sinyal lambalarını da kontrol eder. Bu nedenle CAN ağı, frenleme, yönlendirme sinyalleri ve ikaz ışıklarını tek bir kontrolcü üzerinden entegre eder.

Motor Sürücü



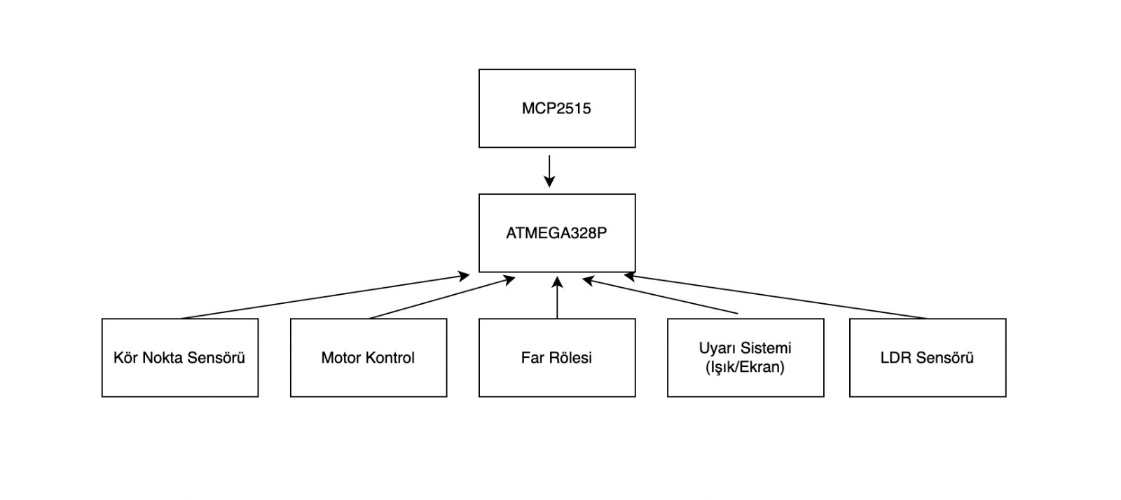
ATMEGA328P, gaz pedalından gelen kullanıcı komutlarını motor sürücü birimine aktarır ve BLDC motoru motor sürücü (ESC) tarafından kontrol eder. Hall effect sensörü motorun dönüş hızını ve sıcaklığını ölçer. Bu bilgiler mikrodenetleyiciye gönderilir ve motorun performansını izleyebilir ve aşırı ısınma gibi durumlarda önlem alabilirsiniz. Bu veriler CAN hattı üzerinden diğer birimlere aktarılır ve MCP2515 modülü merkezden izlenir.

Batarya



ATMEGA328P mikrodenetleyici, MCP2515 modülü ve batarya, CAN ağına entegre edilir. Bu birimde bulunan voltaj bölücü ve sıcaklık sensörleri, batarya durumunu anlık olarak izlemek için kullanılır. Sıcaklık belirli eşikleri aşarsa röle veya fan devreye girer ve ölçülen veriler telemetri modülüne aktarılır. Bu, hem güvenliği hem de batarya ömrünü korur. Bununla birlikte, motor kontrol birimine enerjiyle ilgili bilgiler sağlanır.

ADAS



ADAS (İleri Sürücü Destek Sistemi), çevresel bilgileri analiz ederek sürücülere yardımcı olmak için tasarlanmıştır. Bu sistem, LDR (ışık) sensörleri ve kör nokta sensörleri kullanarak bilgileri işler. ATMEGA328P işlemcisi ve MCP2515 modülü, CAN ağına bağlanmıştır. Otomatik far sistemi, ortam karardığında far rölesi aracılığıyla çalışır. Körü körüne bir tehlike algılandığında, sürücü uyarı sistemi tarafından bilgilendirilir. Tüm bu işlemler CAN hattı üzerinden diğer birimlerle senkronize edilmiştir.