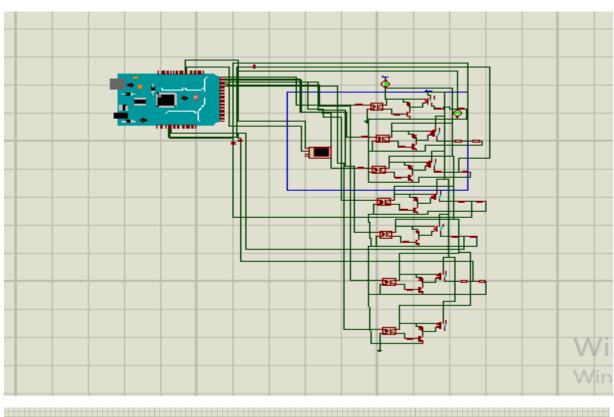
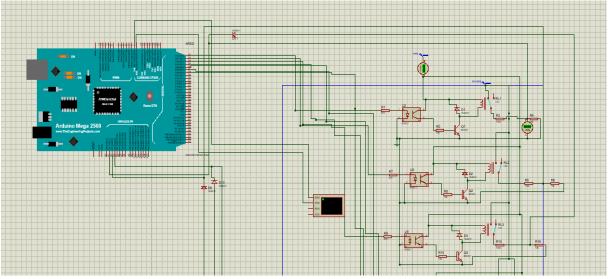
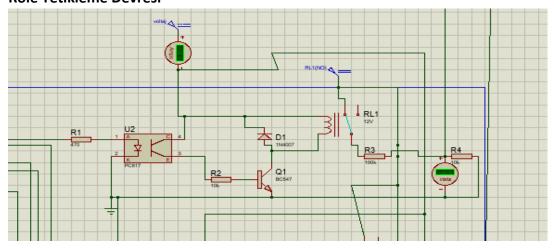
SLIP-RING KABLO TEST CIHAZI RAPORU

Oluşturulan Devrenin Ana Hat Tasarımı





Röle Tetikleme Devresi



KOD KISMI

Oluşturulan kablo test cihazı gerçekleşecek farklı kablo testlerine uyum sağlayabilmektedir. Kodlama üzerinde değişiklik yapılması gereken kısımlardan burada bahsedilecektir.

Yukarıda belirtilen dizilerde yapılacak değişikler ile 30 giriş 30 çıkış olmak üzere kablo testi gerçekleştirilebilmektedir.

```
const int 1[]={10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37};//Input Pins,
int a=28;    //number of input pins
```

Yukarıda "I []" ile belirtilen dizi 12V çıkışı sağlayan pinlerimizi isimlendirmek için kullanacağımız dizidir. Yapılacak yeni testlerde pin sıralaması doğru şekilde bu dizi üzerine kaydedilmelidir. "a" ile belirtilen integer ise kaç tane pine 12V çıkışı sağlayacağımız belirtmektedir.

```
int c[]={1,2,3,4,5,6,7,8,9,10,11,12,13,14,25,26,27,28,29,30};//55 pin read
const int d[]={1,2,3,4,5,6,7,8}; // 8 pin read
int y[]={1,2};//Fan Pin
```

Yukarıda belirtilen pinlerimiz okuma yaptığımız pinleri isimlendirmekte kullandığımız dizilerdir. Okuma yapılırken çakışan pinler olması sebebiyle farklı dizilerin oluşturulmasına ihtiyaç duyulmuştur.

"c[]" ile ifade edilen dizimiz 55 pinli konektörün girişlerini isimlendirmekte kullanılmaktadır. Yeni bir konnektöre geçtiğimizde bağlantılara göre bu dizide değişikliğe gidilmelidir.

"d[]" ile ifade edilen dizimiz 8 pinli konektörün girişlerini isimlendirmekte kullanılmaktadır.

"y[]" ile ifade edilen dizimiz fan konektörünün girişlerini isimlendirmekte kullanılmaktadır.

Yukarıda belirtilen diziler yapılan testte alınması gereken sonuçların işleneceği dizilerdir. Bu dizilere 12V verilen pine karşılık okunması gereken pin ismi girilerek doğruluk testinin çalışması sağlanmaktadır. Arduino okuma yaptığı pin ismini buradaki dizi ile karşılaştırmakta hata bulduysa hata ekranına geçiş yapmaktadır.

CİHAZ GÖRÜNÜMÜ



Cihaz önden görünüşü yukarıda verilmiştir. Buradaki fotoğrafta cihazımızın açılış ekranı ve 12V çıkış sağlayacağımız 37 pinli konektör görülmektedir.



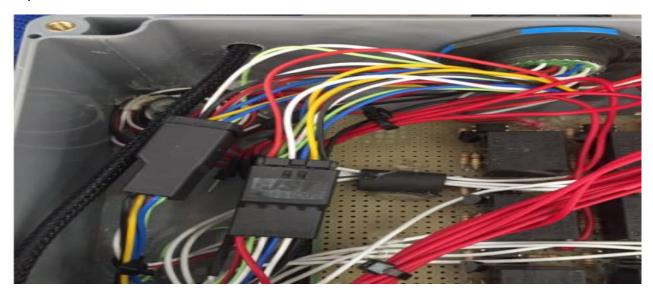
Bu fotoğraf cihazımızın arka yüzünü göstermektedir. Bu fotoğrafta 55 pinli konnektör, 8 pinli konnektör , fan konnektörü, usb bağlantı kablomuz, DC besleme kablomuz ve reset butonumuz görülmektedir.

Usb bağlantı kablomuz ile Arduino kartımızı beslemekte ve yazılım değişikliklerinde kullanılmaktadır. Besleme kablomuz üzerinden röle beslemelerimiz gerçekleşmektedir. Besleme kablomuzda kırmızı '+', beyaz ise '-' ucumuzu temsil etmektedir. Reset butonumuz yeniden başlatma işlemini gerçekleştirmektedir.

KONNEKTÖR DEĞİŞİMİ



Yukarıda belirtilen fotoğrafta sırayla 12V ile beslenen 37 pinli konnektör ucunu rölelere bağlayan JST konnektörler görülmektedir. Bu konnektörler olası değişimlerde yeni yapılacak ucun takılmasıyla yeni uca uyum sağlayacaktır. JST konnektörlerde kablo bağlantıları üzerlerinde yazan sıralamaya göre yapılmıştır. Uygun karşılıkları hazırlanarak yeni konnektörler test edilebilir. Aynı zamanda yapılan değişiklikler kodlama üzerine kaydedilmelidir.



Yukarıda okuma yapılmak için kullanılan JST konnektörler görülmektedir. Burada bağlantılar yapılırken yine JST üzerinde yazılı olan numaralara göre yapılmıştır. Bağlantılar değiştirildiğinde isimlendirmenin doğru yapılması için kodlama üzerinde giriş pinlerini adlandıran dizide gerekli değişiklik yapılmalıdır.

CİHAZ ÇALIŞMA DURUMU

Cihaz çalışırken aynı zamanda ekran üzerinde hangi pin üzerinde çıkış verdiğini ve okuma aldığını göstermektedir. Yazılan diziler ışığında karşılaştırma yaparak hata olduğunda hata ekranı vermektedir. Hata oluştuğunda test durmaktadır.



Ekran çıkıştan besleme yaptığında aldığı okuma pinini göstermektedir eğer belirtilen dizilere uyum sağlanmaz ise hata ekranına geçiş yapar. Burada 'HATA 12' kodu bize 12 numaralı çıkıştan besleme yaptığında hatalı bir pinden okuma yaptığını veya hiç okuma yapamadığı belirtmektedir. Bir hatayla karşılaşmaz ise test tamamlanarak 'TEST BITTI' yazısı ekrana verilmektedir.