STNM-GÖMÜLÜ SİSTEMLER(EMBEDDED SYSTEMS)-İLK PROJE(FİRST PROJECT)

1. **PROJE TANIMI**:

Bu proje, 8 DC motor kullanarak bir robotik sistemin 3 doğrusal yönde hareket etmesini sağlayacak şekilde tasarlanmıştır. Proje, elektronik devre tasarımı ve simülasyon adımlarını içermektedir. İlk aşamada, TinkerCAD platformunda devre tasarımı gerçekleştirildi ve sistemin çalışma prensipleri ve bağlantıları test edildi. Bu devre, Arduino mikrodenetleyici kullanılarak motor sürücüsü ve DC motorlar arasında etkileşim sağlanacak şekilde kontrol edildi. Motorların hareketini gözlemlemek için gerekli olan Arduino kodları yazıldı ve Tinkercad üzerinde simüle edilecektir.

İkinci aşamada yine TinkerCAD platformunda, tasarlanan devreye birkaç eklemede bulunuldu.Hareket kontrolünü sistem taraflı değil de kullanıcının yönetmesi amaçlı bir kızılötesi sinyaller alan kumanda eklenildi. Bu adımda, sistemin hareketleri sonsuz döngüde değil istenilen şekilde gerçekleştirildi. Projenin amacı, sanal ortamda sistemin hareketlerini ve devrenin doğru şekilde çalıştığını , motorların birbirleriyle ve kumanda komutlarıyla hareketini gözlemleyebilmektir . Bu sistem, robotik uygulamalar ve otomasyon sistemlerinde kullanılabilecek potansiyel bir sistem olmaktadır.

**2)PROJEDE KULLANILAN PLATFORMLAR:**

**TinkerCAD: Devre tasarımı** ve simülasyonu için kullanılır. Bu platform, **Arduino mikrodenetleyicisi**, **DC motorlar** ve **motor sürücüleri** ile devreyi tasarlayıp test etmeye olanak tanır. TinkerCAD’in ücretsiz ve erişilebilir olması da kullanımımı destekledi.

**3)KULLANILAN BİLEŞENLER:**

**- Motorlar:** Üç eksende hareketi sağlamak için kullanılan 8 adet DC motorlar.

* **X ekseni motoru:** Sağ-sol hareket.
* **Y ekseni motoru:** İleri-geri hareket.
* **Z ekseni motoru:** Yukarı-aşağı hareket.

**- Motor Sürücü:** Motorları yönlendirmek için kullanılan motor sürücü kartıdır.Projede 4 Adet L293D kullanılmaktadır. Her bir L293D iki motoru kontrol edebildiği için 8 motoru yönetmek adına bu sayı seçildi.

**- Kumanda:** Kullanıcıdan gelen komutları alarak motorlara ileten bir kontrol cihazıdır.

**- Arduino Uno:** Projenin beyni olan ve tüm hareketleri kontrol eden mikrodenetleyicidir.

**- Kütüphaneler:**

* **IRremote** kütüphanesi, projede **kızılötesi (IR) sinyalleri** alıp gönderebilmek için kullanılmaktadır. Bu kütüphane, IR alıcılar ve vericilerle iletişim kurmak amacıyla çok çeşitli işlevler sunmaktadır.

4) **Hangi Motorların Çalışması Hangi Yöne Hareketi Sağlıyor?**

Projemizde 8 adet DC motoru 3 eksende (x, y, z) hareket sağlayacak şekilde organize ettik. Motorların dağılımı ve görevleri şu şekildedir:

* **X Ekseni (Yatay Sağ-Sol Hareket):** Bu eksen için 2 motoru senkronize bir şekilde çalıştırdık. Motor 1, 2 ileri yönde çalıştırdığımızda sistem sağa, ters yönde çalıştırdığımızda ise sola hareket etti. İki motorun bir arada kullanılması, yükü dağıtarak daha dengeli bir hareket sağlamaktadır.
* **Y Ekseni (Yatay İleri-Geri Hareket):** Y ekseni için 2 motor (motor 3 ve 4) görevlendirildi. Bu motorlar ileri yönde çalıştığında sistem ileri, ters yönde çalıştığında geri gitti. Y ekseninde de birden fazla motor kullanarak gücü arttırdık.
* **Z Ekseni (Dikey Yukarı-Aşağı Hareket):** Z ekseni için kalan 4 motoru (motor 5,6,7 ve 8) kullandık. Dikey hareket daha fazla güç gerektirdiği için motor sayısını artırdık. Bu motorlar, örneğin bir kaldırma mekanizmasına bağlıysa, ileri yönde çalıştığında sistemi yukarı kaldırmakta, ters yönde çalıştığında ise aşağı indirmektedir.Bu gözlemi 3D ve gerçek hayatta daha iyi kavrayabiliriz.

Neden böyle bir dağılım kullandık? Her eksende birden fazla motor kullanarak hem hareketin stabilitesini artırdık hem de tek bir motorun aşırı yüklenmesini önledik. TinkerCAD simülasyonumuzdaki bu düzen, her yönde tutarlı bir hareket sağlamaktadır.Ayrıca kullandığımız kızılötesi kumanda ile beraber HEX kodlarına göre hareketleri gerçekleştirdik.

**5) Motor Sürücü Yerine ESC Kullansaydık Kodda Bir Değişiklik Olur muydu?**

Evet, motor sürücü (örneğin L293D) yerine ESC (Electronic Speed Controller) kullansaydık kodda bazı değişiklikler gerekirdi. Şöyle ki:

* **Motor Sürücü ile Kontrol:** L293D gibi bir sürücüde, motorun yönünü ve hızını kontrol etmek için iki giriş pini (IN1, IN2) ve bir PWM pini kullanıyoruz.Sınırlı pin sayısından dolayı da zorluklar yaşadık. Kodda bu pinleri ayrı ayrı ayarlayarak motoru ileri, geri veya durdurabiliyoruz.
* **ESC ile Kontrol:** ESC’ler ise genellikle tek bir PWM sinyaliyle çalışır ve bu sinyal hem hızı hem de yönü belirler (örneğin, 1000 µs geri, 1500 µs nötr, 2000 µs ileri). ESC kullanırsak, her motor için bir servo sinyali gibi davranmamız gerekirdi.
* **Kod Değişikliği:** ESC ile çalışsaydık kodun büyük kısmında değişiklikler olurdu.

Sonuç olarak, ESC kullanımı kodu biraz sadeleştirirdi ve işimizi daha kolaylaştırırdı ama TinkerCAD’in DC motor simülasyonuyla tam uyumlu olmayabilirdi ve ESC, ek bir servo kütüphanesi gerektirebilirdi. Bu yüzden motor sürücüyle devam ettik.

**6) Gerçek Hayatta Projemizi Yapsak Arduino Uno Yeterli Olur mu?**

Arduino Uno, projeyi gerçek hayatta hayata geçirmek için sınırda bir seçenek olurdu. Şöyle açıklayayım:

* **Pin Sayısı:** Uno’da toplam 14 dijital ve 6 analog pin var. 8 DC motoru kontrol etmek için motor sürücü başına en az 3 pin (2 yön, 1 PWM) gerektiğini varsayarsak, 8 motor için 24 pine ihtiyaç duyarız. Uno bunu karşılayamaz. Bu zorluklar sebebiyle projede motor sürücülerinin hız kontrolünü yapmak için pine bağlamadım.Doğrudan 5V’a bağlayarak motorları %100 hızda çalıştırdım. Böylece her bir motor sürücüsündeki 2 pin boşa çıkmış oldu.
* **Güç ve İşlem Kapasitesi:** Uno, 8 motorun kontrol sinyallerini üretebilir ama motorların güç ihtiyacını karşılamak için harici bir güç kaynağı ve motor sürücüleri şarttı.Bu yüzden 12V’luk bir güç kaynağı kullandık. Bu, motorların tam performansla çalışmasını sağladı.İşlem kapasitesi ise basit PWM kontrolü için yeterli.
* **Alternatif:** Gerçek hayatta Arduino Mega tercih edilebilir; çünkü 54 dijital piniyle daha fazla motoru doğrudan kontrol edebilir.Ayrıca hız kontrolü de gayet kolay şekilde yapılabilir.

Kısacası Uno ile yapılabilir ama ek donanım ve dikkatli pin yönetimi gerekir.

**7) Projemiz Nasıl Daha İşlevsel Hale Getirilebilir?**

Projemiz zaten sağlam bir temel sunuyor, ama birkaç geliştirmeyle daha işlevsel hale getirebiliriz:

* **Sensör Entegrasyonu:** Mesafe sensörleri (HC-SR04) veya ivmeölçer (MPU6050) ekleyerek sistemin çevresine duyarlı hale gelmesini sağlayabiliriz. Örneğin, bir engele yaklaştığında otomatik durabilir.
* **Hızlı Geri Bildirim:** LCD ekran veya Bluetooth modülü (HC-05) ekleyerek kumandadan bağımsız bir durum göstergesi oluşturabiliriz. Bu, sistemin pozisyonunu veya motor durumunu anlık olarak gösterir.
* **Modüler Tasarım:** Mekanizmayı bir robot koluna veya taşıyıcı bir platforma dönüştürebiliriz. Örneğin, z eksenine bir tutucu ekleyerek nesne taşıma özelliği kazandırılabilir.
* **Enerji Verimliliği:** Motorların sadece gerektiğinde çalışmasını sağlayan bir uyku modu veya enerji tasarruf algoritması eklenebilir.
* **Kablosuz İyileştirme:** Kumandayı RF veya Wi-Fi tabanlı bir sisteme yükselterek kontrol mesafesini artırabiliriz.

Bu gibi özellikleri eklemek projeyi hem daha çevre dostu hem de daha kullanıcı dostu yapabilir.

**8) Projede Karşılaştığım Sorunlar Nelerdi?**

**-IR kumanda algılama sorunu:** Projemin başlangıç aşamasında, kızılötesi (IR) kumanda sinyallerini almak ve işlemek için **eski bir sürüm IRremote kütüphanesi** kullanılmıştır.Bu yüzden HEX kodları tamamen yanlış olarak belirlenmiştir.Bu sorunu her bir tuşun HEX kodları farklı şekilde olmadığında anladım ve araştırdım.Bu şekilde sorun halledilmiş oldu.

**-Motorlarda beklenmeyen hareket sorunu:** Motor bağlantılarında ve pinlerinde birkaç yanlışlık sebebiyle 1.motoru çalıştırmaya çalışırken 2.motor pini kullandığımı fark ettim.Bağlantı ve kodlarımı gözden geçirerek hatayı hallettim.

**-Kullanılan pin sorunu**: Projede 0 ve 1 numaralı dijital pinleri motor kontrolü için kullandım fakat bu pinler Seri Haberleşme (Serial Communication) ile çakışmayaneden olmaktaymış.Bu çakışmadan dolayı motorun düzensiz bir şekilde çalıştığını fark ettim ve motorun bağlı olduğu dijital pinleri değiştirdim.

**-Kod kaynaklı sorunlar:** İlk öncelikle kodlarımdaki hatayı tespit edebilmek için adım adım ilerledim. Hatanın bağlantı kaynaklı mı yoksa kodsal mı olduğunu belirlemek benim için en zor kısımdı.Bunun için sonrasında parça parça deneme yaptım.İlk önce dc motorların sonsuz döngüde çalışmasını tamamladım.Daha sonrasında ayrı bir projede IR kumandanın çalışmasını gözlemledim ve HEX kodlarının internette bulduğum kodlarla aynı olmadığını gördüm. Bu yüzden kendi kumandamın kodlarını tek tek kaydettim ve iki kodu da birbirine entegre ettim.Hangi tuşun hangi hareketi kontrol ettiğini kodlarımda da belirttim ve switch case(anahtarlama) yapısını kullandım.Ara ara çalışmalarımda hata olduğunu belirten -monitöre- yazılar yazdırdım.Bu şekilde hatalarımı ayıkladım ve hallettim.

Projede emeği geçen ve yardım eden herkese teşekkür ederim .Daha güzel,başarılı projelere imza atmak umuduyla…

Proje Yöneticisi- Project Management

Merve KARAGÜLLE