**EASYLCD\_MOBILE ile ESKİ TELEFONLARINIZLA, ESP MODÜLÜNE ELVEDA DİYEBİLİRSİNİZ. BASİT BİR EV OTOMASYON ÖRNEĞİ.**

Bu yazıda neler bulacaksınız öncelikle onlara bakalım. Birincisi eski telefonumuzu değerlendirmek için (APP INVENTOR ile yapılmış) LCD+ESP+RTC olarak kullanıyoruz.   
Bunun yanı sıra 1 adet App inventor ile yapılmış android telefon kullanıcı programı. ( Komut göndermek için) Ayrıca 1 adet merkezi kontrol ünitesi olarak kullanabileceğimiz ARDUINO programı.

EASYLCD\_MOBILE dediğimizde zaten LCD ve RTC programını düşünebiliriz. Bunlar sadece kütüphane gerektirmeyen akılda kalıcı birkaç komuttan ibarettir. Bu komutlar ile eski telefonunuzla haberleşip, telefon ekranına bilgi yazabilirsiniz ve sistem saatini öğrenebilirsiniz.   
Bunun yanı sıra, ESP modülü gibi kullanmak için bir TCP SERVER programı da eklendi. Bu program bir aracı olarak düşünülebilir. Merkez arduino olduğundan, kullanıcılarla arduino arasında aracılık edecektir. Merkez, 1 adet olmasına rağmen birden fazla kullanıcı olabilmektedir. Yani çoklu bağlantıya da destek vermektedir.

Program, App Inventor ile yapıldı ve tcp server ( TCP sunucu) uzantısı kullanıldı.

Bu programın çalışma şekli şudur.

1-) Kullanıcı devamlı aynı IP adresine bağlansın diye tcp server kurulan telefonun STATİK IP ayarını yapıyoruz.   
2-) Arduino ve tcp server haberleşsin diye Bluetooth bağlantısı sağlanır.   
ve tcp server port ayarlanır ve server açılır.

3-) Kullanıcı program SERVER IP ve PORT ayarlarına ayarlanır ve CONNECT butonu ile bağlantı sağlanır.

4-) Server bu bağlantıyı aldığında buna bir ID numarası bağlar. Ve komutlar için hattı dinlemeye başlar.

5-) Kullanıcı komut gönderdiğinde, bu komutu arduinoya yollar.

6-) Arduino bu komutu uygular ve uyguladığına dair bir geri bildirim verir.

7-) Server bu geri bildirimi alır ve Bağlantı sağlayan tüm kullanıcılara iletir. ( Tüm kullanıcılar dedim çünkü server çoklu bağlantı kabul ediyor. Fakat hangi bağlantıdan geldiğini bilmesine rağmen programı basit tuttuğum için ayırma işlemi yapmıyor. Bundan dolayı Arduinodan gelen cevabı tüm kullanıcılara iletiyor. )   
Sizler programı geliştirebilirsiniz. Hangi kullanıcıdan gelirse ID numarasına göre o kullanıcıya cevap verebilirsiniz.

Komut sistemini de anlamanız gerek. Çünkü EASYLCD de kullandığım komut sistemini kullandım. Bu komut sisteminde

3 karakter belirteç ( LCD, ESP gibi)   
2 karakter komut (WR,DD,SO,ON,OF;SD gibi)  
1,1 olarak yazılan numaralar koordinat olarak satır sütun olarak kullanılır. Bu programda birinci numara oda numarası ya da kat numarası gibi düşünüldü. İkinci numara ise kontrol edilecek cihazın numarası olarak düşünüldü. LCD olarak kullanırken satır ve sütun numaralarını gösterir.

(-) İşareti ise komut ile yapılacak işlemi ayırır.

KOMUT : ESPSD 1,1-ON ( ESPSD=esp datası, 1. Kat 1. Lambayı aç demektir. Burada kat ve cihaz sayıları değiştirilebilir. )

Arduino programı ise basit tutulmuştur. Sadece Karşılaştırma yapılırken “ESPSD 1,1-ON” gelirse bunu yap gibi. İstenirse programınızda komut ve koordinatları ayırıp daha faklı karşılaştırmalar yapabilirsiniz. Programınız daha da kısalacaktır.

Yukarıdaki işlem sırasına göre açıklamaya başlayayım.

1-) Telefonda sabit IP ayarlama. Bendeki 2 telefon farklı şekilde bunu yapıyor. Belki de android 2.3.6 ve android 6 versiyonlarında böyle olabilir. Bunun sebebi ise kullanıcı telefonunu bir kere IP adresine ayarladıktan sonra bir daha ayar yapmaması içindir.

Android 2.3.6 galaxy Y telefonumda aşağıdaki resimlerdeki gibi ayarlanacak. Önce kablosuz ayarlarınıza gireceksiniz sonrasında menüden gelişmiş seçeneğine tıklayıp resimdeki ayarları yaparsınız.

**statik\_IP\_galaxy\_1 ve statik\_IP\_galaxy\_2 jpg resimleri**

Android 6.0.1 versiyon ASUS zenfone II telefonumdaise aşağıdaki resimdeki gibi ayarlanıyor. Yine kablosuz ayarlarına girip bağlantı ağınızın üzerine basılı tutun. Sonrasında çıkan menüden AĞI DEĞİŞTİR seçeneğine basın. Resimler yol gösterecektir. Ayarlarken Önceden hangi IP adresini aldıysa onu verebilirsiniz. Çünkü başka IP yazarsanız diğer cihazlarınızla çakışabilir.

**statik\_IP\_asus\_1 statik\_IP\_asus\_2 statik\_IP\_asus\_3**

**Kendi telefonunuz farklı versiyonda ise internetten araştırabilirsiniz**.

2- Bluetooth ve tcp server bağlantısı. Bluetooth bağlantısı isteğe bağlıdır. İsterseniz ve cihazınızda OTG yi destekliyorsa USB OTG bağlantısı da yapabilirsiniz. Sadece haberleşme kısmını easylcd mobil v1 de olduğu gibi USB otg bloklarını eklemeniz yeterli olacaktır.  
 OSB OTG örneklerimden kopyalayabilirsiniz. **LINK versiyon 1 yazısı**

Bluetooth modülünüzü eşleştirdikten sonra TCP server start butonuna basıp bağlantıyı sağlarsınız. ( Haberleşme tamam ise arduino seri monitörden takip edebilirsiniz)

TCP server IP adresiniz ve port numaranız artık kullanıcılar için hazırdır.   
**(Bluetooth modüller konusunda bilgi ararsanız bu yazımı da okuyabilirsiniz.)**

3-) Kullanıcı programı IP ve PORT ayarları server IP ve port ayarlarına ayarlanarak CONNECT butonuna basılır ve bağlantı hazırsa ekranda kırmızı olan kutu yeşile döner. Ve kullanıcıya bilgi verilir.

**Client1 ve client2 jpg**

4-) Server bu bağlantıyı aldığında buna bir ID numarası bağlar. Ve komutlar için hattı dinlemeye başlar.

**LOG jpg**

5-) Kullanıcıdan alınan komut bilgisi (1,1-ON) komutu TCP server’a ulaştığında, bu komut ECHO 1,3-ON olarak kullanıcıya geri gönderilir bu komutu aldım yanlışlık var mı demek içindir. Fakat ben gelen ECHO bilgilerini programda kullanmadım. Sizler gelen veri yanlış ise doğrusunu gönderecek şekilde programı düzeltebilirsiniz.   
Ayrıca arduinoya ise ESPSD 1,3-ON LCD R olarak bilgi yollanır. LCD R geri bildirim kodudur. Bu LCD kullanımında kullanılır. Ben dikkate almadım. Fakat standart olsun diye yolladım. Şayet arduino LCD R bekliyorsa artık beklemesin ve gelen komutu işleme alsın diye LCD R eklendi.

**USER ECHO .jpg**

6-) Arduino bu komutu aldıktan sonra karşılaştırma yaparak Komutu uygular. Sonrasında ise Komutu uyguladığını geri bildirim ile tcp server’a yollar.( ESPSD 1,3-ON)

**Seri monitor.jpg**

7-) TCP server bu geri bildirimi alıp bağlı olan tüm kullanıcılara gönderir. Yukarıdaki LOG resmine bakınız.

8-) Kullanıcı programı gelen komutu alıp karşılaştırır. Şayet doğru bilgi gelmişse ve ON komutu yolladıysa o butonun rengini Yeşil yapar. Başka odada olsanız bile komutun uygulandığını buton renginden görebilirsiniz.

**Client 3,4,5**

Tüm çalışma mantığı budur.  
Program örnek olduğu için basit karşılaştırma metotları uygulandı. Ayrıca geri bildirim sadece görüldü fakat yanlış mı diye sorgulanmadı. Sizler programı geliştirebilirsiniz.

Resimler size yardımcı olacaktır. Bazı resimlerde **D:1,3-ON** gibi görünse de onlar eski versiyona ait. Son versiyonda kullanıcının göndereceği komut ( **1,1-ON 2,1-OFF** gibidir)

Bunun Yanısıra çoklu kullanıcıyı göstermek için video içerisinde ESP8266-01 modülünü kullandım. Yani illaki kullanıcı için başka telefon kullanmak zorunda değilsiniz. Mesela birkaç arduino da birbirini ESP8266-01 modülleriyle görebilir ve konuşabilir.   
Ben videoda kullanıcı için ESP8266-01 modülünü kullanırken AT komut setiyle kullandım. Bunu yapmamın sebebi deneme kolaylığı sağlayan FIRAT DEVECİ nin yaptığı programdı. Her şeyi rahatlıkla ayarladığımdan bunu seçtim.

**AT ve SDK çevrimi yazısını buraya ekle.**

**Esp8266user1.jpg user2.jpg**

Sizler Kendi ESP modülünüzü programlayıp 2 arduino arası konuşmayı sağlayabilirsiniz.

İlla ki bu benim yaptığım gibi buton led kontrolü olmak zorunda değil. ESP belirtecini **AAC** gibi (arduino arduino connect) olarak düşünüp kendi komutlarınızı oluşturup 2 arduino arasını wifi ve TCP server üzerinden konuşturabilirsiniz. Bunun için app inventor programında KODAYIR bloklarına kendi komutlarınızın nereye yönleneceğini belirtmeniz gerekecektir.

**Not:** ESP8266-01 USB adaptörü sadece kullanmak içindir. ESP8266-01’i programlamak için adaptörde değişiklik yapmalısınız. Bu değişiklikler internette bulunmaktadır. Ben kendi pcb mi kullandığımdan ve bu değişiklikleri denemediğimden tavsiye edeceğim bir link yok. İnternetten bulabilirsiniz.

Program geliştirme fikirleri: Yukarıda aralarda bahsettiğim ekleme yapabileceğiniz yerler dışında programda olan bir özellik WIFI on ve OFF durumunu arduino kontrol edebilir. Nasıl ki geceleri herkes yattıktan sonra modemler için gece uykusu yapılabiliyorsa programınızda da ev sisteminin wifi özelliğini gece kapatarak hem pil tasarrufu hem de ısı tasarrufu yapılabilir. Programınıza bunun ayarlarını girebilirsiniz. Zaten sistem saati her daim bağlı olan telefondan gelecektir. İllaki app inventor programına ek yapmanız gerekmiyor arduino programınıza da bunu ekleyebilirsiniz.

Ayrıca sistemin tamamının uykuya dalmasını sağlayabilirsiniz. App inventor programına belli saatlerde arduino ile haberleşme için bluetooth kapatmasını ve saati geldiğinde açmasını söyleyebilirsiniz. Sistemi kapattığında App inventora ekleyeceğini ses programı ile ya da titreşim programı ile yüksek sesi dinleyebilirsiniz ya da deprem gibi büyük hareketliliklerde telefon uyanıp bluetooth açıp arduinoya tüm ışıkları açmasını ve alarm ses özelliğini aktif etmesini söyleyebilirsiniz. Hatta ev sahipleri evde yoksa whatsapp üzerinden ev sahiplerine mesaj gönderebilirsiniz. Eski telefonunuzu Arduinoya bağlamakla yapacağınız bir çok olaydan bir kaçı budur. Sizler de programlarınıza bu tip davranışları ekleyebilirsiniz.

Gelelim OTOMASYON kelimesine. Bu yapmış olduğum programda her ne kadar bazı geri bildirimleri değerlendirmesem de, geri bildirimli çalışma olmazsa OTOMASYON kelimesini kullanamayız. Mesela 8 röleli kart satın aldınız bunları telefonla kontrol etmek için bir program yazdınız. AÇ ve KAPAT komutlarını kullandınız. Bu sadece komutun uygulamasıdır. Geri bildirim yoksa röle kontakları yapışmışsa bundan kontrol eden devrenin haberi olmayacaktır. Kapattım zannedecek fakat devre kapanmayacaktır.   
Otomasyon bir dizi sıralı işlemin yapılması ve çıkan sonuçların değerlendirilerek düzeltmelerle işlemin devamlılığının sağlanması durumudur. Burada geri bildirim olumsuz şartları ön görmüşse programa direkt bildirim olarak gönderilir. Şayet tehlikeli bir durum söz konusu olduysa, interrupt denilen kesicilerle iş akışı durdurulur ve müdahale edilmesi sağlanır.   
Burada basit bir ev otomasyonu dedim. Çünkü ilk hareketi biz versek de program gereği burada yapılacak ve denetlenecektir. Bu programda arduinoya bilgi gidip gitmediğini denetledim. Fakat, bu butonla kontrol ettiğimiz röle, bir motoru kontrol ediyorsa, bu motor bir su pompasıysa, o zaman motor çalışmaya başladığında su gelmesi gerekir. Bunu anlamak için arduinoya bağlı bir nem sensörü konulur ve su akışı sağlandıysa bu nem sensörü onaylar. Motor çalıştırıldığında, bu sensör hala suyun olmadığını söylüyorsa, arduino motoru durdurup kullanıcıya mesaj yollar. Bu da otomasyon uygulaması olacaktır. Dediğim gibi arduinodan sonrasını kontrol etmedim. Sizler kendi programlarınızı bu yönde geliştirebilirsiniz.

Aklınıza bir soru gelebilir. Kullanıcı butona bastı ve çalıştığını farz ederek bağlantı programını kapattı. O zaman arduino nasıl kullanıcıyı uyaracak?

Bu da okuyucularıma bir problem olarak kalsın cevabı verecek olanlar yoruma bırakabilir. Bakalım kaç yolu var.