

1 Temmuz 2018'de güncellendi - GPIO0 çıkış olarak kullanıldığında yeniden programlamaya ilişkin not eklendi.

ESP8266-01 Ledleri göstermek için 2 Nisan 2018 tarihinde güncellendi. Bu ledleri bağlı oldukları pimlerden kontrol edebilirsiniz.

Giriş

ESP8266-01, çok düşük maliyetli bir WiFi özellikli yongadır. Ancak çok sınırlı bir G / Ç'ye sahiptir. İlk bakışta, programlamak için bir kez yapılandırırsanız tüm pinler kullanılır.

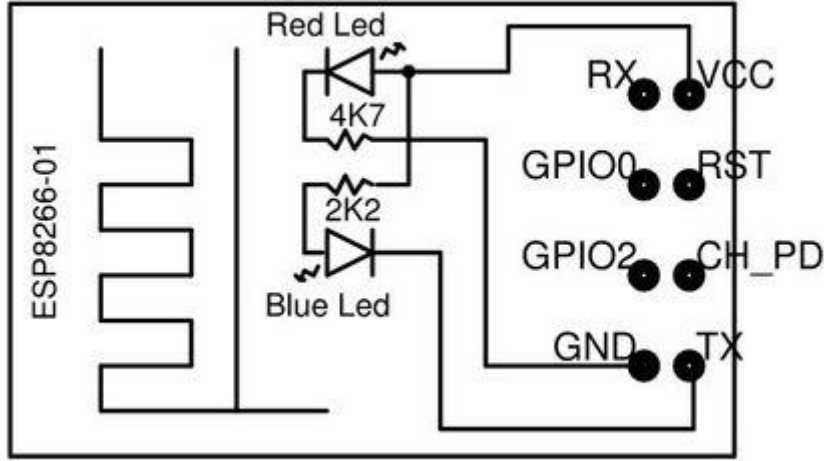
Bu eğitilebilirlik, sonraki ESP8266-01 projesinde size nasıl dört (4) kullanılabilir giriş / çıkış alabileceğinizi ve daha fazla giriş elde etmek için IC2'yi nasıl kullanabileceğinizi göstermek için ESP8266 GPIO0 / GPIO2 / GPIO15 pinlerini kullanma üzerine kuruludur.

Buradaki kod, <http://github.com/esp8266/arduino> adresinde açıklandığı gibi Arduino IDE kurulumunu kullanarak modülü programladığınızı varsaymaktadır. **Kurul Yöneticisiyle Kurmak.** Pano Yöneticisi'ni Araçlar → Pano menüsünden açıp Yazım'ı seçin. **Katkıda** ve esp8266 platformunu yükleyin.

Bu talimatlara www.pfod.com.au adresinden ESP8266-01 Pin Magic adresinden de ulaşılabilir.

Gereçler:

Adım 1: ESP8266-01 Pin



ESP8266-01, en küçük ESP8266 modülüdür ve yalnızca 8 pime sahiptir. Bu VCC'den GND, RST (reset) ve CH_PD (yonga seçimi), G / Ç pimleri değildir, ancak modülün çalışması gerekir. Bu, GPIO0, GPIO2, TX ve RX'i mümkün olan G / Ç pimleri olarak kullanılabilir durumda bırakır, ancak bunların önceden atanmış fonksiyonları vardır. GPIO0 ve GPIO2, modülün hangi modda başlatılacağını belirler ve TX / RX pimleri, modülü ve genel hata ayıklama amacıyla kullanılan Seri I / O için kullanılır. Modülün doğru şekilde başlatılmasını sağlamak için GPIO0 ve GPIO2 cihazlarının dirençleri bağlanmış olması gerekir.

Adım 2: ESP8266 Programlama İpuçları (espcomm Başarısız)

Arduino IDE'yi kullanarak ESP8266'yı programlarken (bkz. ESP8266-01 Wifi Kalkanı), bazen (genellikle) Arduino IDE'de aşağıdaki gibi bir hata mesajı alırsınız: -

esp_com açılmadı

hata: COM33 açılmadı

error: espcomm_open başarısız oldu

hata: espcomm_upload_mem başarısız oldu

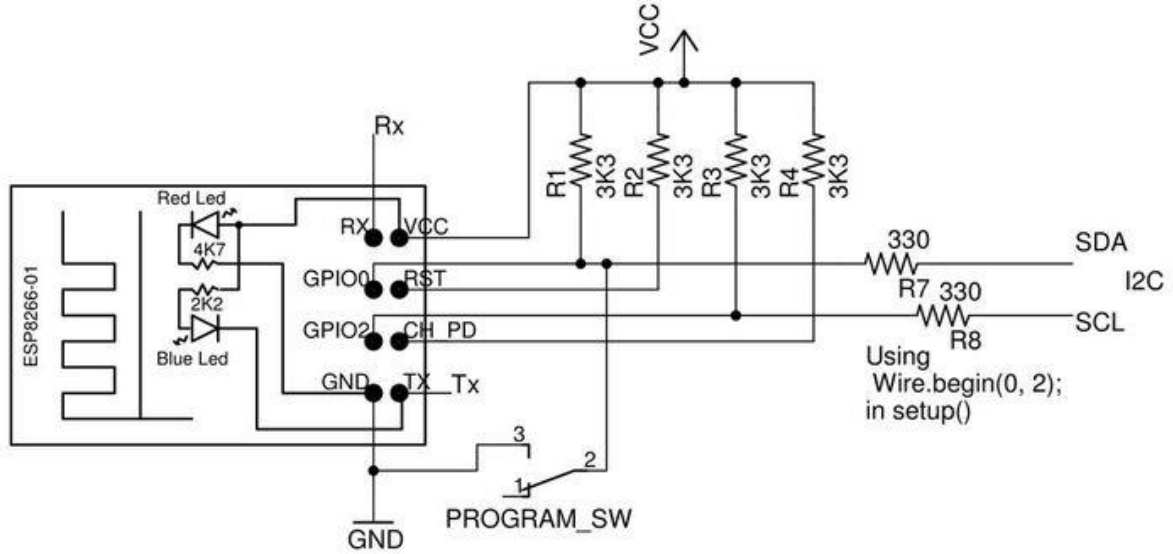
Bu durumda çalışmasını sağlamak için şu adımları izleyin: -

1. Arduino Tools menüsünden ESP8266 board seçilip seçilmediğini kontrol edin.
2. Arduino Tools menüsünden bir COM portu seçtiğinizden emin olun.
3. ESP8266'yı GPIO0 topraklamalı güç çevrimi (temiz güç uygulaması, aşağıya bakın)
4. 3) sorunu çözmezse, USB kablusunu bilgisayardan çıkarın, birkaç saniye bekleyin ve tekrar takın.

5. 4) sorunu çözmezse, USB kablosunu PC'den çıkarın, Arduino IDE'yi kapatın, Arduino IDE'yi açın, USB kablosunu tekrar takın.

ESP8266'ya güç uyguladığınızda, GPIO0'u toprakladıktan sonra, temiz bir şekilde uygulandığından emin olun. Bağlantıyı sallama. ESP8266 ledi sadece yanmalı ve herhangi bir yanıp sönmeden kalmalıdır.

Adım 3: En İyi Püf Noktası - I2C Kullanın



ESP8266-01'e fazladan girdi almanın en iyi yolu bir I2C arayüzü kullanmaktır.

Bir seçenek, GPIO0 ve GPIO2'yi I2C veri yolu olarak kullanmaktır.

Modülün doğru şekilde başlatılması için gereken çekme dirençleri, I2C veri yolu çekme dirençlerinin iki katı olabilir ve veri yolu üzerindeki diğer, bağımlı, bileşenler açık kollektördür ve bu nedenle veri yolunun açılmasında veri yolunu aşağı çekmemelidir. Bununla birlikte, bazı durumlarda, köleler, özellikle de batarya yedekli olanlar sıkışıp kalabilir ve veri yolunu aşağı tutabilir. Bu durumlarda, ESP8266 önyükleme aşamasından geçinceye kadar otobüsü izole etmeniz gerekecektir.

I2C veri yolu için TX ve RX kullanarak bu sorunu önleyebilirsiniz.

Unutulmaması gereken birkaç şey:

1. GPIO1 (TX) Veri hattı olarak kullanılır, çünkü GPIO1'de açılışta daima hata ayıklama çıktısı alırsınız. Bu çıktıyı bastırmanın bir yolu yoktur, ancak Saat çizgisi (RX) yüksek tutulacak, bu verilerin hiçbirisi köle olarak saatlenmeyecektir

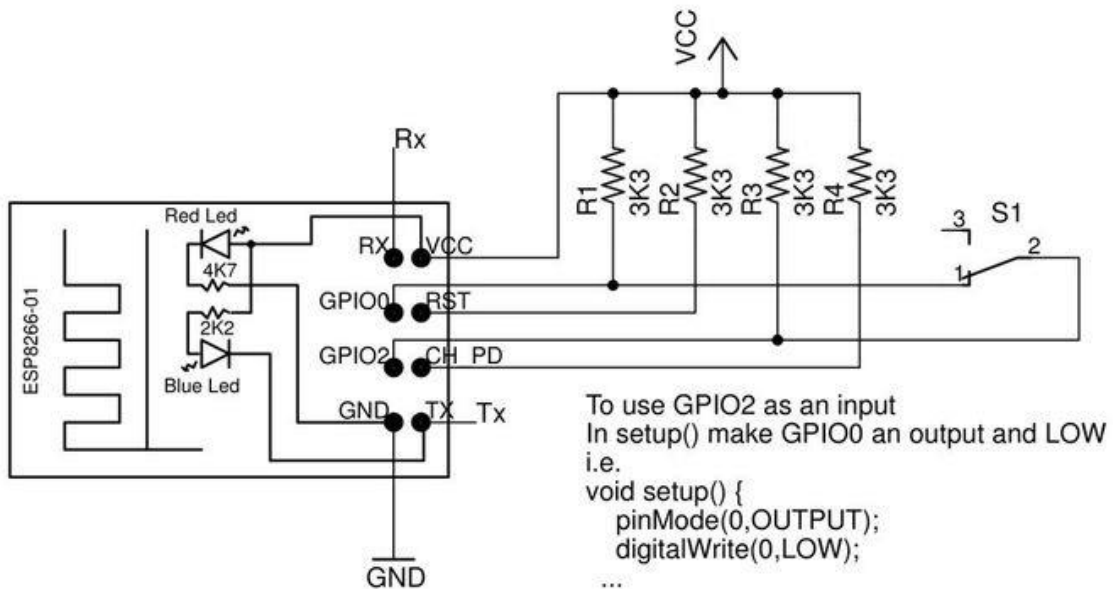
2. ESP8266'yı programlarken, RX hattı programcının çıkışına bağlanır. Programlamanın sonunda, ESP8266 yeniden başlatılır ve 330 Koruma direnci, RX'in programcının çıkış sürücüsünü kısa devre yapmasını önler.
3. I2C serisi dirençler, TX, RX için I2C veriyolundaki kısa devrelere benzer bir koruma sağlar

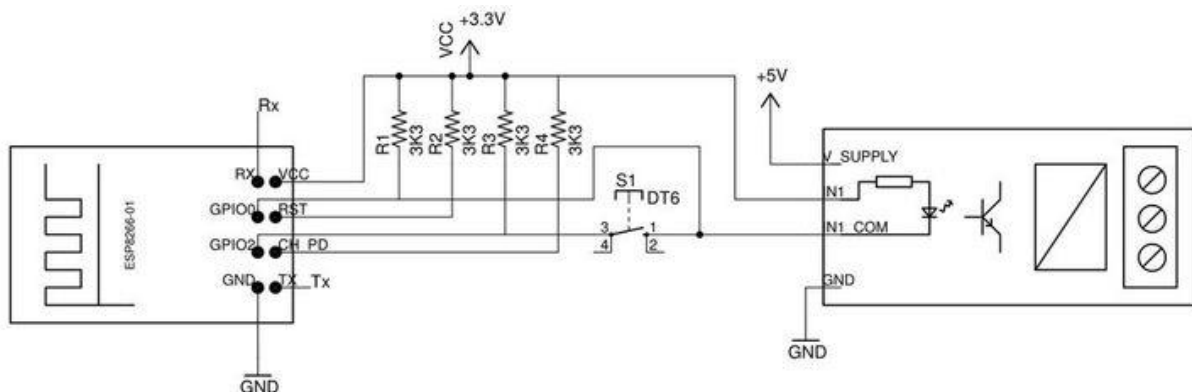
ESP8266, 3.3V'luk bir cihazdır, bu nedenle tercihen 3.3V I2C bağımlıları kullanın. Çoğu, ancak hepsi değil, I2C cihazları bugünlerde 3.3V'tur. "Genel olarak, bir cihazın diğerinden daha yüksek voltajda olduğu bir sistemde, iki cihazı I2C üzerinden aralarında seviye kayma devresi olmadan bağlamak mümkün olabilir. İşin püf noktası, çekme dirençlerini iki voltajın altına bağlamaktır." (SparkFun I2C eğitimi) 5V ve 3.3V cihazlarının bir karışımı için, çekme dirençlerini yukarıda gösterildiği gibi 3.3V hattına bağlayın.

I2C'yi kullanmak, ESP8266-01'e temelli bir modülün ADC girişini göstermeyen çok kanallı bir A'dan D'ye dönüştürücü eklemek için harika bir yoldur. Örneğin, Adafruit 12bit I2C 4 kanallı ADC kullanmak veya SparkFun'ın I2C DAC Breakout - MCP4725 kartı analog çıkışı için. I2C veriyollarında diğer birçok sensör tipi de mevcuttur.

I2C sorunlarını aşma konusunda daha fazla bilgi için <http://www.i2c-bus.org/i2c-primer/common-problems> ... adresini ziyaret edin. Ayrıca veri yolunun temizlenmesine yardımcı olmak için kısa bir yöntem için I2C Batarya Destekli RTC için Güvenli Başlangıç bölümüne bakınız.

Adım 4: OUTPUT için GPIO0 / GPIO2'yi ve INPUT için RX'i kullanma





Pimleri yapılandırmanın başka bir yolu da budur. Not: Bu numara sadece izole girişli bir röle modülüne sahipseniz çalışır (N1 ve N1-com). Bu sınırlama ve destek kodunun karmaşıklığı nedeniyle, bir girdi olarak RX kullanan önceki örnek tercih edilir.

ESP8266 GPIO0 / GPIO2 / GPIO15 pinlerinin kullanılması, ilave bir giriş elde etmek için GPIO0 / GPIO2'nin nasıl kullanılacağını zaten kapsıyor. Bu örnek, GPIO0'u röle sürücüsü çıkışı olarak ve GPIO0 / GPIO2'yi giriş olarak kullanmak üzere genişletilecektir.

İşte pdf olarak şematik.

Burada GPIO0, röleyi sürmek için bir çıkış olarak kullanılır ve GPIO0 / GPIO2, WiFi üzerindeki uzaktan kumandanın yanı sıra röleyi açıp kapatmak için manuel olarak geçersiz kılma olarak kullanılan anlık basma düğmesini okumak için bir girdi olarak kullanılır. bağ. Anlık basma düğmesi, güç uygulandığında basıldığında yapılandırma modunu etkinleştirmek için de kullanılır.

Buradaki hile, ESP8266 modülü başlatılırken GPIO0 ve GPIO2'yi yüksek tutarken tüm bunları yapmak.

Çekme dirençleri, R1 ve R3, bu iki pim için gerekli Yüksek değerini sağlar, ancak GPIO0 ve GPIO2'ye bağlı herhangi bir ekstra devrenin pimleri çekmemesini sağlamalısınız. Optik olarak izole edilmiş röle + 3.3V ile GPIO0 arasına bağlanır. Bu, GPIO0'un başlangıçta yüksek kalmasını sağlar, ancak GPIO'nun başlatmadan sonra bir çıkış yapmasını ve röleyi çalıştırmak için röle girişini topraklamasını sağlar. Modül başlatılırken anlık düğmeye basılmasının bir önemi yoktur, çünkü GPIO0'u GPIO2'ye bağlar ve her ikisini de çekme dirençlerine bağlar.

Yapılandırma Modunu Algılama

ESP8266'yı geçici bir erişim noktası olarak kullanarak, onu burada açıklanan şekilde bir web sayfası aracılığıyla yapılandırabilirsiniz. Bu işlemin bir kısmı, yapılandırma moduna girmek istediğiniz yazılıma işaret etmek için açılışta bir düğme veya kısa devre kullanmaktır.

ESP8266 modülü başlatıldıktan sonra, **kurmak()** kodu. Bu kodda, anlık basma düğmesine basılıp basılmadığını tespit etmek için, basmak için bir GND sağlamak üzere GPIO0 seviyesini düşürmeniz ve düşük olup olmadığını görmek için GPIO2 girişini kontrol etmeniz gerekmez. Bu kontrolün bir yan etkisi, ünitenin konfigürasyon moduna alındığı zaman rölenin daima çalıştırılmasıdır. Rölenin çalıştığını gördüğünüzde, girişi o zamana kadar algılanacağı için butonu bırakabilirsiniz. İşte bunu yapmak için bazı örnek kod **kurmak()**

```
boolean configMode = false; // config modunda değil normalde void setup ()
{pinMode (0, OUTPUT); digitalWrite (0, DÜŞÜK); // GPIO0 çıkışını düşük yapın // bu
```

```
düğmeye basıp basmadığını görmek için GPIO2 girişini kontrol edin configIO =  
configMode = (digitalRead (2) == LOW); if (configMode) { // AP'yi başlat ve config web  
sayfasını sunmaya hazır ol // // config modunda belirtmek için röleyi açık bırak. //  
.....} else { // normal kullanım // kapatmak için GPIO0'u YÜKSEK yap röle digitalWrite  
(0, HIGH); // .....} // kurulumun geri kalanı ()}
```

Manuel Geçersiz Kılma Basma Düğmesini Tespit Etme

Önceki bölüm, yapılandırma modunu etkinleştirmek için açma düğmesine basıldığında tespit edilmesini kapsıyordu. Röleyi WiFi bağlantısı üzerinden kontrol etmenin yanı sıra röleyi açıp kapatmak için bu düğmeyi manuel olarak geçersiz kılma olarak kullanmak istiyoruz.

Rölenin WiFi kontrolü burada ele alınmaz, ancak pfodApp kullanılarak kolayca yapılabilir. ESP8266 modülleri için pfodDesigner ile Arduino kodunun nasıl üretilceğini öğrenmek için OLIMEX Menü Jeneratörüne bakın.

Bu bölüm, basma düğmesine basıldığında, kullanıcının röleyi değiştirmek istediğini belirten, yani KAPALI durumunun açık veya KAPALI durumdayken AÇIK duruma gelmek istediğini gösteren nasıl algılanacağını ele alacaktır. Şematik yukarıdakiyle aynıdır, tüm numaralar koddadır. Dikkate alınması gereken iki durum vardır: -

1. Röle KAPALI ve kullanıcı basma düğmesini kullanarak açmak istiyor,
2. Röle AÇIK ve kullanıcı basma düğmesini kullanarak kapatmak istiyor.

Röle KAPALI ve kullanıcı basma düğmeyi kullanarak açmak istiyor.

Bu durumda GPIO0'un çıkışı YÜKSEK. Gerçekte GPIO0 bu durumda bir giriş olabilir, çünkü R1 direncinin açılmamasını sağlar. İşin püf noktası bu. Bu durumda GPIO0'u bir Giriş yapın ve GPIO2 Çıkışını DÜŞÜK yapın ve ardından kullanıcı basma düğmesine bastığında iki şey olur: - a) GPIO2 tarafından sağlanan topraklama nedeniyle röle basılarak ve b) GPIO0 girişi düşük olacaktır. Kod GPIO0 Girişinin durumunu kontrol eder ve DÜŞÜK olduğunda kod kullanımın basma düğmesine bastığını bilir ve rölenin açık olmasını ister. Kod daha sonra basıldığında serbest bırakıldığında röleyi açık tutmak için GPIO0'u Çıkış DÜŞÜK yapar.

Röle AÇIK ve kullanıcı basma düğmesini kullanarak kapatmak istiyor.

Bu durumda, yukarıdaki durumdan sonra GPIO0, röleyi AÇIK tutan bir DÜŞÜK DÜŞÜK'tür. Şimdi bu durumda GPIO2'yi bir Giriş yapın (R3 tarafından çekilir) ve ardından kullanıcı basma düğmesine bastığında Giriş GPIO2'si GPIO0 üzerindeki DÜŞÜK Çıkış tarafından DÜŞÜK olarak çekilir. Kullanım basma düğmeyi bıraktığında kod, LOW - HIGH geçişini algılar ve ardından GPIO0'u bir çekme direnci (R1) nedeniyle röleyi serbest bırakır ve GPIO2'yi yukarıdaki i) durumunda ayarlamak için bir DÜŞÜK DÜŞÜK yapar.

Bir numara daha. İi) durumunda, röleyi KAPALI konuma getirmek için DÜŞÜK-YÜKSEK geçişi tespit eden bir Giriş olarak GPIO2'ya ihtiyacımız var. Ancak GPIO2'yi yapar ve i) durumunun sonunda giriş yaparsak, kullanıcı röleyi AÇIK konuma getirmek için bastırdıkları basma düğmeyi serbest bıraktıkça YÜKSEK DÜŞÜK bir geçiş alacağız. Röleyi derhal tekrar kapatmaktan kaçınmak için, röleyi açtıktan sonraki ilk YÜKSEK geçişi, röleyi AÇMAK için bastıkları düğmeyi bırakan kullanıcı olduğu için göz ardı edilir.

Örnek döngü () Röle Manuel Aşırı sürüş için kod

Örnek kod burada, ESP8266_01pinMagic_1.ino

Bu da Seri hata ayıklama için mevcut TX / RX pinlerini bırakır veya diğer I / O olarak kullanır.

Sonuç

Bu sayfa, ESP8266-01'de bulunan sınırlı pimlerin en iyi şekilde nasıl kullanılacağını gösterir. GPIO0 / GPIO2'yi I2C veriyolu olarak kullanmak en büyük genişletmeyi sağlar, ancak I2C kullanmıyorsanız, hala bir röle sürebilir ve GPIO0 / GPIO2 kullanarak bir buton girişi tespit edebilirsiniz. Her iki durumda da, TX / RX Seri hata ayıklama için de kullanılabilir veya WiFi bağlantısı üzerinden hata ayıklama yazdırma ifadeleri gönderirseniz, bu pinler genel G / Ç için de kullanılabilir.