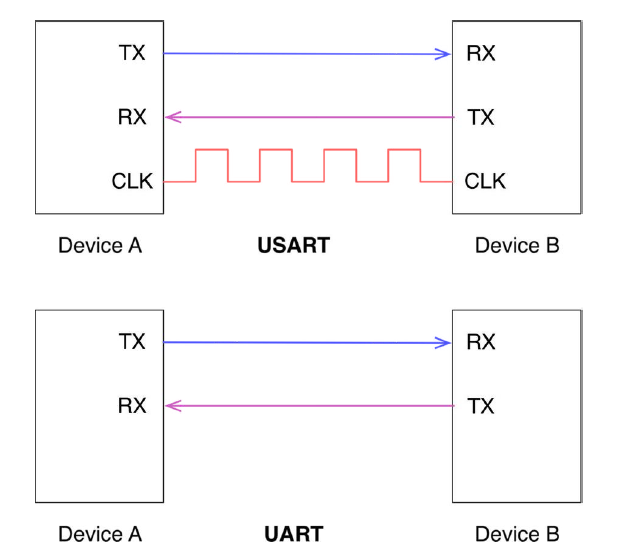
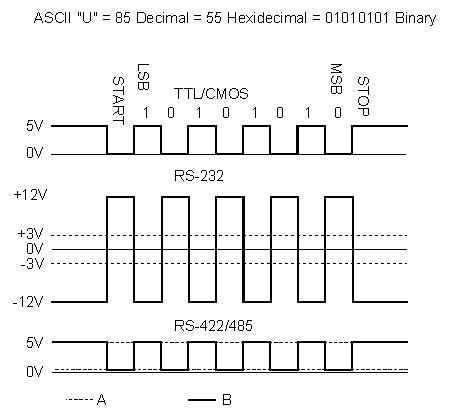
**UART&USART**

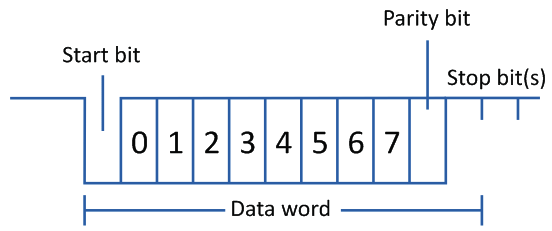
* UART (Universal Asynchronous Receiver Transmitter), bilgisayar veya mikrodenetleyicilerin çevre birimleri ile arasında haberleşmeyi sağlayan haberleşme protokolüdür. Asenkron olarak çalıştığı için herhangi bir “clock” ihtiyacı duymaz.
* USART (Universal Synchronous Asynchronous Receiver Transmitter) ise hem senkron hem de asenkron olarak çalışabilir. UART’a göre daha gelişmiş bir protokoldür. Haberleşme mantıklı aynı şekilde çalışır ancak USART aynı zamanda senkron haberleşmeleri de gerçekleştirebilir.
* Hemen hemen tüm STM32 MCU'ları en az iki UART / USART arabirimi vardır.



* UART haberleşmesini gerçekleştirirken ilk olarak **baudrate** (**veri taşıma hızı**) ayarlanması gerekir. Veri taşıma hızı çeşitli aralıklarda olabilir ancak piyasada yaygın olarak kullanılan baudrate’ler **4800, 9600, 57600, 115200** ve mikroişlemciler için çok fazla tercih edilmese de **921600**. Verinin saniyede ne kadarlık byte’ını taşıyacağını belirlememize yarar. Örneğin veri taşıma hızımızı 115200 seçersek bu bizim için saniyede yaklaşık olarak 115200 byte veri iletimi sağlayacaktır.
* **Veri gönderme için;**



* Haberleşmede 8 yada 9 bit olabilir bunlara star ve stop biti dahil değildir
* Baud rate(veri taşıma hızıdır) önemli bir detaydır. Haberleşmenin hızını belirtir hem receiver hem de transmitter da aynı olmalıdır.
* Start biti 1 den 0 a indirilir.
* Stop bitede 0 konumundan 1 konumuna çekilirse haberleşmemiz sonlanır.



0 1 0 1 0 1 1 0 P1

0 0 1 1 1 0 0 0 P2

Parity biti; Verinin içindeki 1 lerin sayısı çiftse P1 0 olur tek ise 1 olur. P2 1 olur. Gönderilen mesajın pratiy biti kontol edillir eger parity bitindeki belirtilen kurallar hatalı ise mesajın yanlış olduğu söylenir.

* **Kodlama da kullanıcak terimler;**
* BaudRate: Verinin saniyede ne kadarlık byte’ının taşınacağını ifade eder.

• WordLength:İletilen veya alınan veri bitlerinin sayısını belirtir. Bu alan UART\_WORDLENGTH\_8B veya UART\_WORDLENGTH\_9B değerini alabilir, yani 8 veya 9 veri biti içeren bir UART paketi üzerinden iletim yapabiliriz. Bu sayı, başlatma ve durdurma bitleri gibi iletilen bitleri içermez.

• StopBits: Bu alan iletilen stop bitlerinin sayısını belirtir. UART\_STOPBITS\_1 veya UART\_STOPBITS\_2 değerini varsayabilir; bu, çerçevenin sonunu belirtmek için bir veya iki durdurma biti kullanabileceğimiz anlamına gelir.

• Parity: Parity, hata denetiminin çok basit bir şeklidir. İki çeşittir: tek veya çift. Eşlik bitini üretmek için tüm veri bitleri toplanır ve toplamın düzgünlüğü bitin ayarlanıp ayarlanmadığına karar verir. Parity isteğe bağlıdır ve çok yaygın olarak kullanılmaz. Gürültülü ortamlar arasında iletim için yararlı olabilir, ancak aynı zamanda veri aktarımını biraz yavaşlatır ve hata işlemeyi uygulamak için hem gönderen hem de alıcı gerektirir.

-UART\_PARITY\_NONE

-UART\_PARITY\_EVEN

-UART\_PARITY\_ODD

• Mode: RX veya TX modunun etkin veya devre dışı olduğunu belirtir.

-UART\_MODE\_RX

-UART\_MODE\_TX

-UART\_MODE\_TX\_RX

• HwFlowCtl: RS232 Donanım Akış Kontrol modunun etkin veya devre dışı olup olmadığını belirtir.

-UART\_HWCONTROL\_NONE

-UART\_HWCONTROL\_RTS-Request To Send(RTS)

-UART\_HWCONTROL\_CTS -Clear To Send(CTS)

-UART\_HWCONTROL\_RTS\_CTS

• OverSampling: Yüksek frekansa sahip bir sinyali örnekleme tekniğidir. Aşırı örnekleme alanı, -UART\_OVERSAMPLING\_16 değerinin her kare biti için 16 örnek gerçekleştirdiğini veya -UART\_OVERSAMPLING\_8 değerinin 8 örnek gerçekleştireceğini varsayabilir.