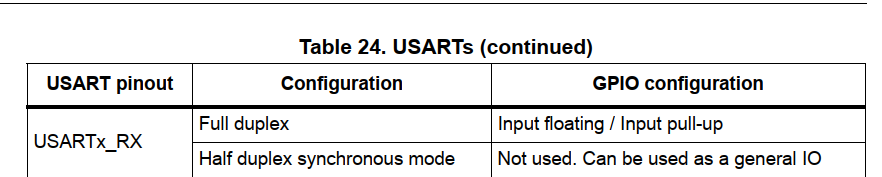
Determine the relation of what about UART connection

1. GPIO config
2. UART config

Hand\_on\_work()

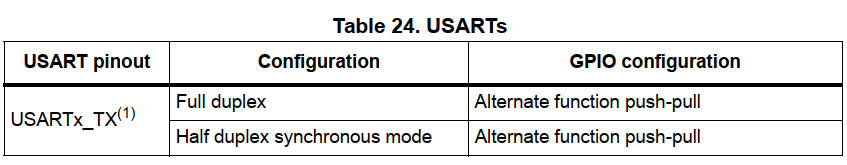
* 1. GPIO config:
  2. Alternative pin function: (STM32 F1)
* Input mode: (RX function)



CNF: input with pull-up/pull-down

MODE: input mode (reset state)

* Output mode: (TX function)



CNF: Alternative function output push-pull

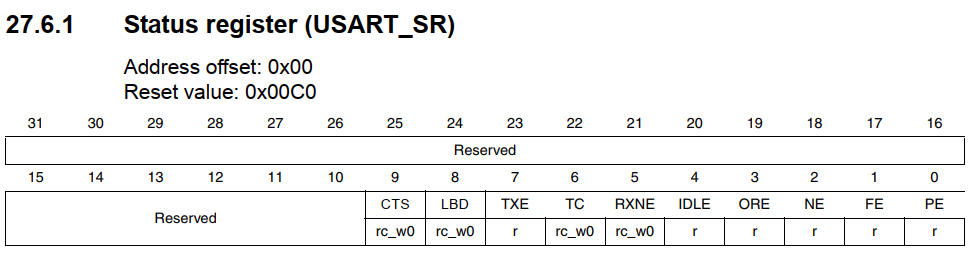
MODE: Output mode with frequency at 3 mode (10M, 2M, 50Mhz)

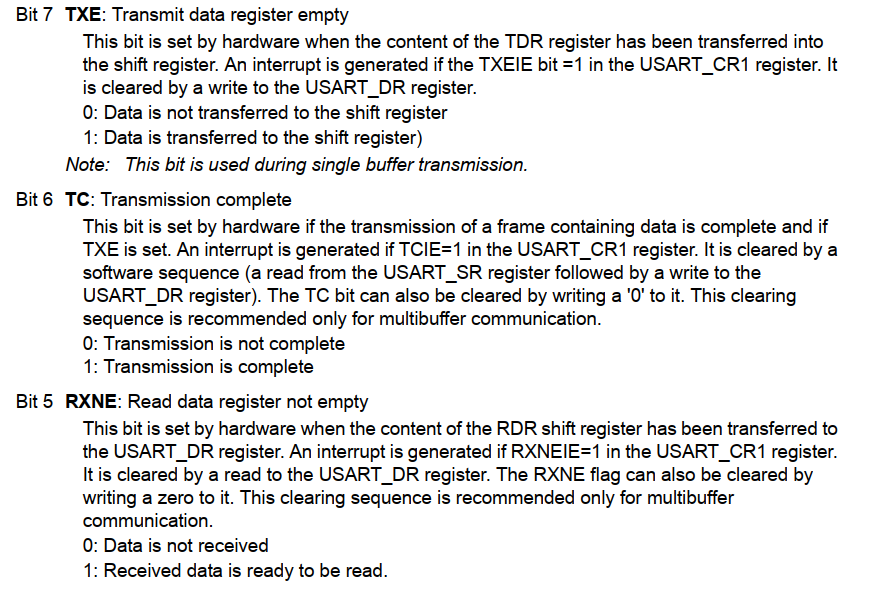
* 1. Alternative pin function: (STM32 F4)
* Depend on bus line or Mux port to define the UART pin on STM32 t
* Using this to find the number of port, configure may be in gpio register high/ low.
* With mode in Alternative pin that config directly in “gpio port mode register”.
  1. UART config
  2. UART/USART control register
* F1 series have 3 “control register - CR”, which we will use to config? That depend on function we want. In this circumstance, we want to enable USART, appear in bit 13 on CR1
* Need config: USART enable, Transmitter & Receiver, Word length config.
  1. Config BAUDRATE (USART\_BRR)
* To config Baudrate, we need to find the clock frequency that protocal UART work with. STM32 have 2 bus is (APB1 and ABP2) that have difference clock frequency.
  + 1. Config clock frequency of bus
    2. Set DIV of DIV\_Mantissa and DIV\_Fraction



After find the result, the integer is value need to be set for Mentissa, on the other hand Faction equal to (the decimal part cross to 16(bits)) this number will be rounded up, and set to the Faction register.

* 1. Sending and receiveing data





* + 1. Transmit phase:

**TXE** bit is set **once** data tranfer from Data register to Shift register.

After bit is shifted, now Shift register send this data to TX line. In the mean time, we can send another byte to Data register, and later that bytes will come.

Finished the transmition, bit **TC** will be set (after bit stop be sent)

* + 1. Receive phase:

With **RXNE** bit, it is set when having the data in DATA register.