**UART**

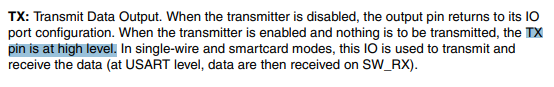
Интерфейс - совокупность унифицированных технических средств и правил (описаний, соглашений, протоколов), обеспечивающих взаимодействие устройств или сопряжение между системами.

* **U(S)ART --** Universal (synchronous) asynchronous receiver transmitter – периф. подситсема для передачи данных в соотв. со стандартом\*.
* Интерфейс передачи данных может быть **последовательным** (передавать бит за битом) или **параллельным** (передавать несколько битв в единицу времени). Также передача может быть **синхронной** и **асинхронной** (тут пример про биты, означающие начало передачи в обоих случаях).

Абсолютно параллельной передачи не бывает!!!

* Передача бывает одностронней (half-duplex) или двунаправленной (full-duplex).

\* -- приемник и передатчик дожны договориться о скорости передачи данных; когда они будут передаваться и т.п.\*\*; договоренность об уровне в канале при отсутсвии передачи (у нас высокий).



(RefMan 759)

* **Про ремаппинг будет ниже (мб)**

\*\* -- для передачи используются слова длиной 8 или 9\*\*\* бит + стоповые биты (0.5, 1, 1.5, 2 шт.).



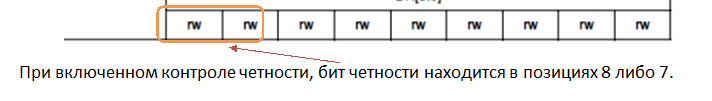
(RefMan 762)

Про нецелое кол-во стоповых:

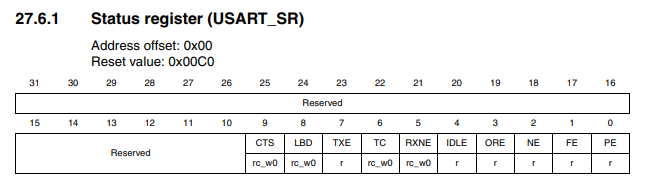
*In the RS-232 case, the number of “stop bits” refers to a*duration*, so 1.5 stop bits simply means 1.5 times the duration of a bit at the given rate.*

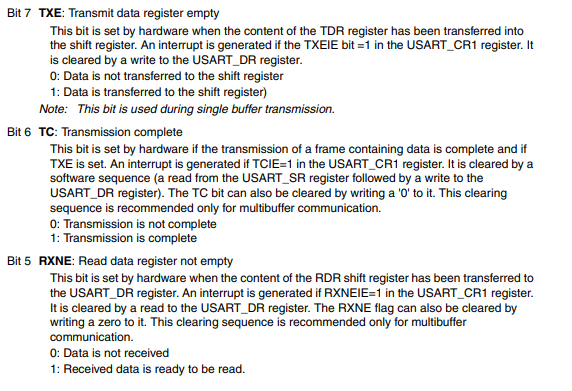
*\*\*\* -- речь про добавление избыточности, в частности, контрольных сумм.*

*Проверка чётности: добавляется бит, явл. суммой по модулю 2 всех битов сосбщения. Если при передаче бит обратился, сверка контрольных сумм поможет остледить нечетное количество ошибок. –> есть флаг Parity Error*

**

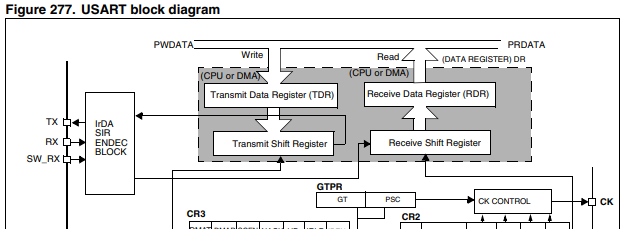
* Интересующие нас события в USART:
  + Receive buffer full (5ый флаг в SR -- **RXNE**: Read data register not empty)
  + Transmit buffer empty (7ой флаг в SR -- **TXE**: Transmit data register empty)
  + End of transmission (6ой флаг в SR -- **TC**: Transmission complete)





(RefMan 790)

* Интересующие контакты:
  + TX – выход трансмиттера, нужно соединить с ресивером другого устрйоства.
  + RX – аналогично
  + GND
* Как устроен USART:

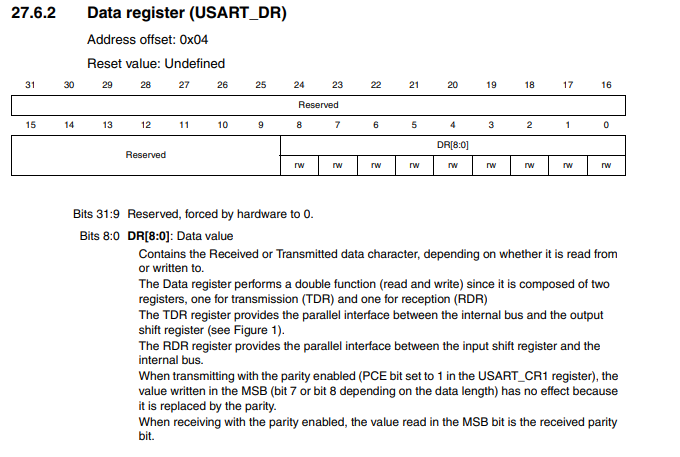


(RefMan 761)

* При передаче используется два регистра, причем, хоть они и называются одинаково, для приемника и передатчика своя пара этих регистров (см. картинку выше).

Пример для передатчика:

Data register – туда “кладутся” данные, которые требуются отправить.

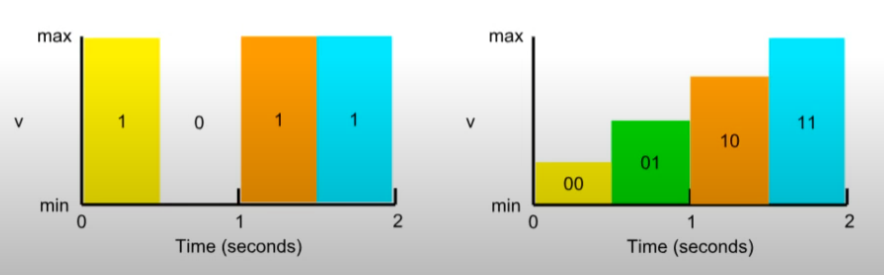


(RefMan 792)

Shift register – данные из DR падают сюда, и бит за битом сдвигаются в канал. В соотв. со сдвинутым битом, выставляется соотв. уровень на время, заданное скоростью.

//отсебятина: тут принцип FIFO

* Baud rate и Bit rate



Baud rate – количество изменеий уровней за секунду

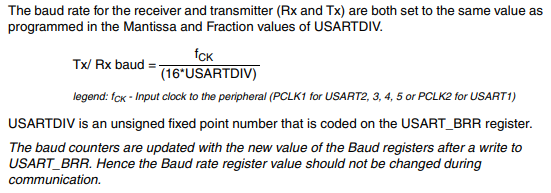
Bit rate -- количество битов за секунду

Bit rate >= Baud rate

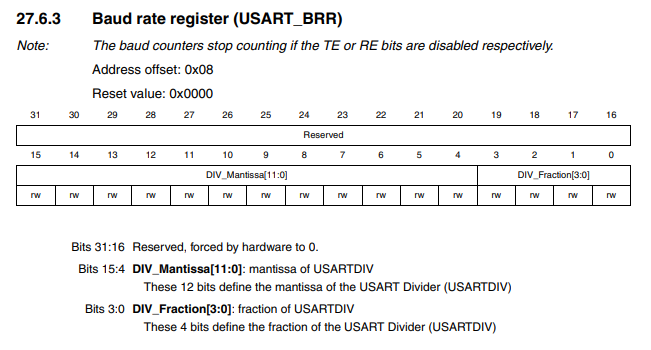
**ВАЖНО! Регистр с бод-рейтом является предделителем для этой периф. подсистемы!**

****

(RefMan 762)

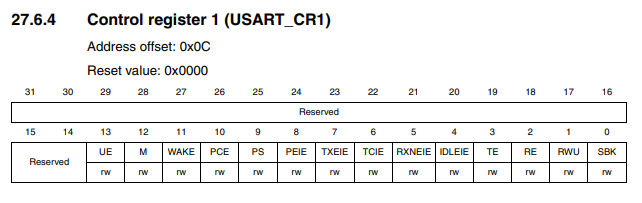
****

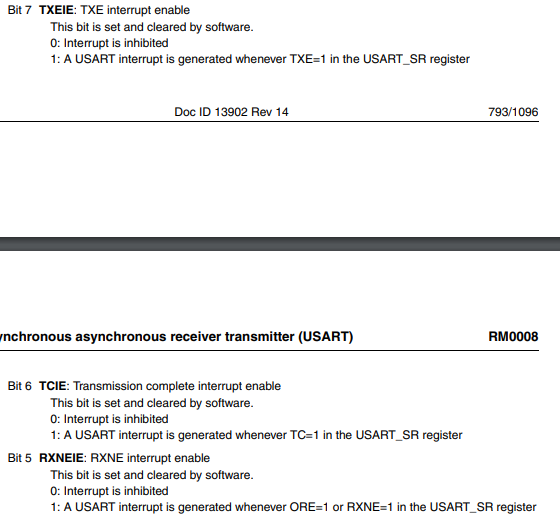
(RefMan 770)

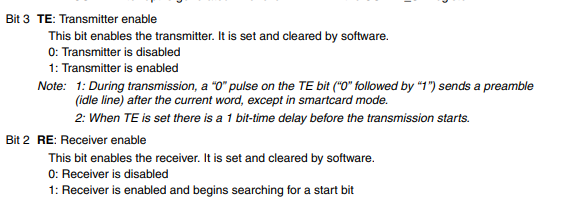


(RefMan 790)

* USART Control Register



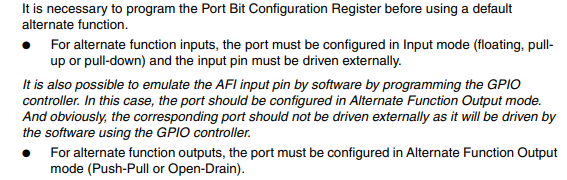




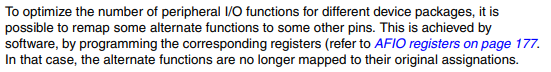
(RefMan 794)

* Alternate function

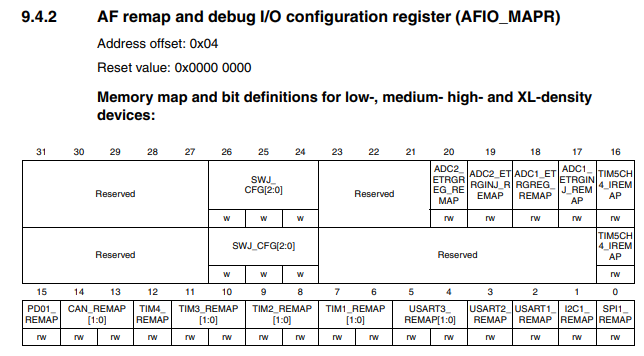
*In addition to acting as General Purpose I/Os, many of the pins on the STM32 microcontrollers have one or more alternate functions. These alternate functions are tied to subsystems inside the MCU, such as one or more SPI, I2C, USART, Timer, DMA or other peripherals.*

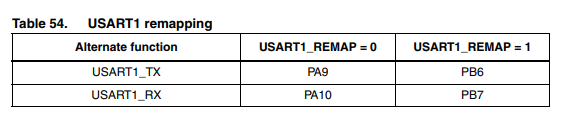


(RefMan 157)

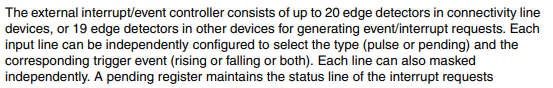


(RefMan 157)

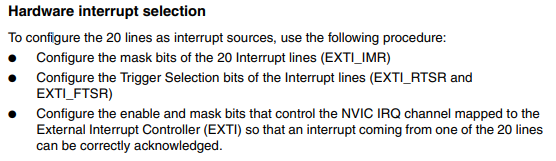




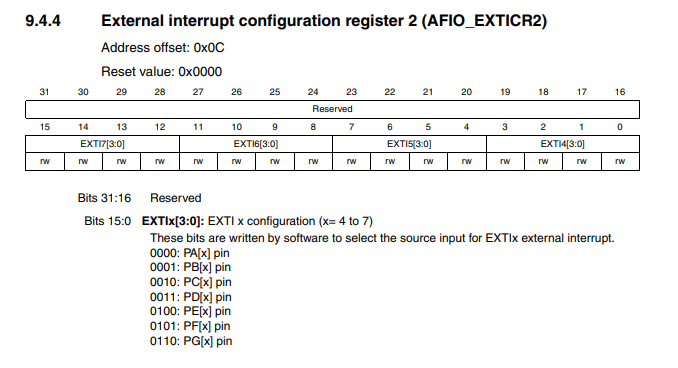
**EXTI**

****

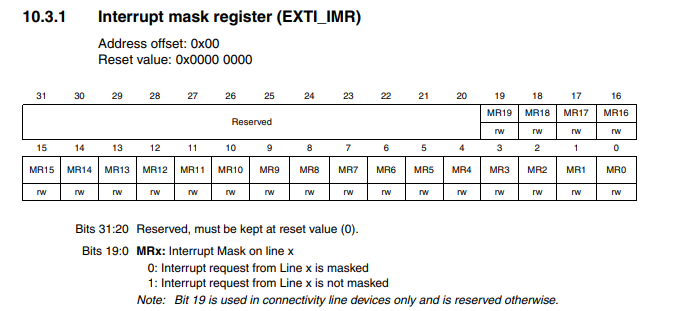
(RefMan 198)

****

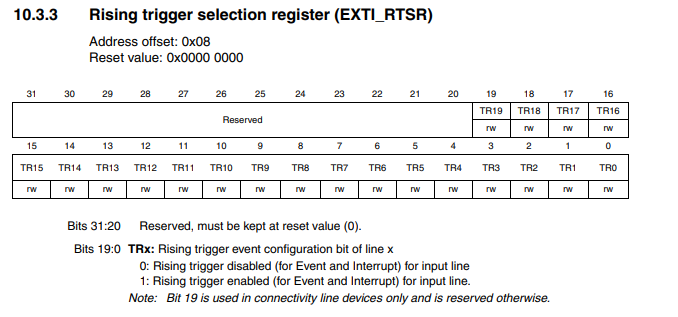
(RefMan 200)



(RefMan 185)

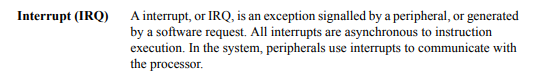


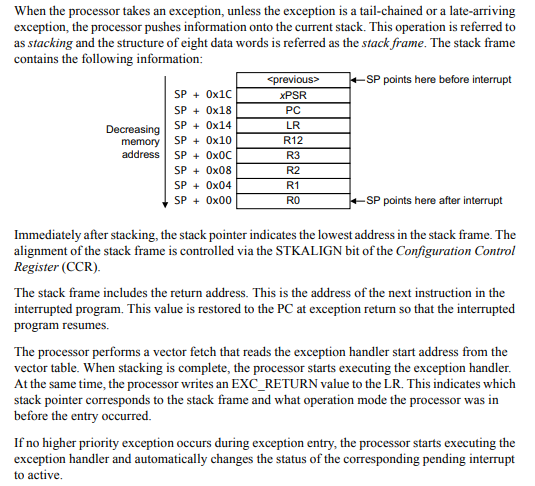
(RefMan 202)

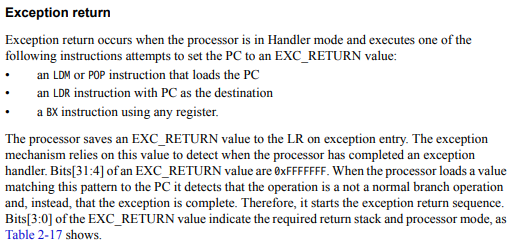


(RefMan 203)

Немножко про интеррапты







(CortexM3\_DevUserGuide ~39)