STM32 Microcontrollers Course

Hamed Jafarzadeh

Summer 2016

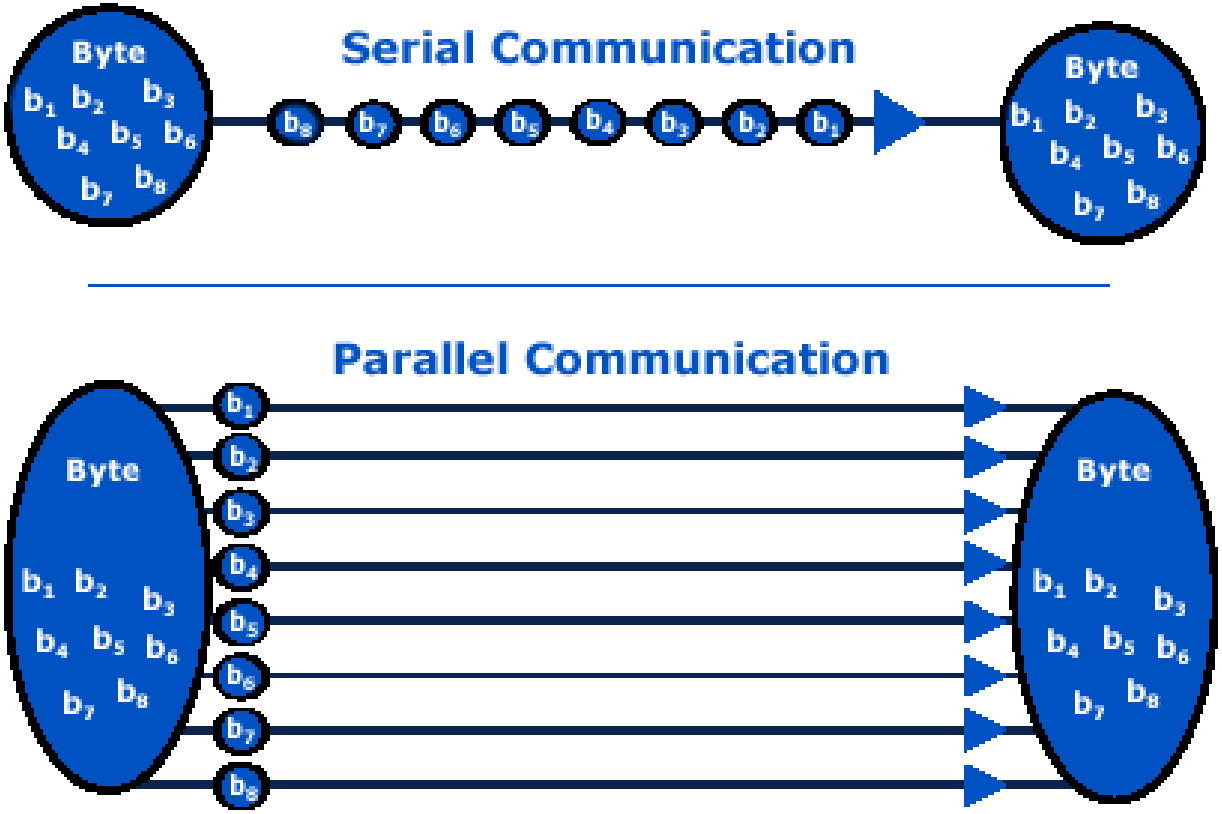
UART and External Interrupts

life.augmented

Protocol

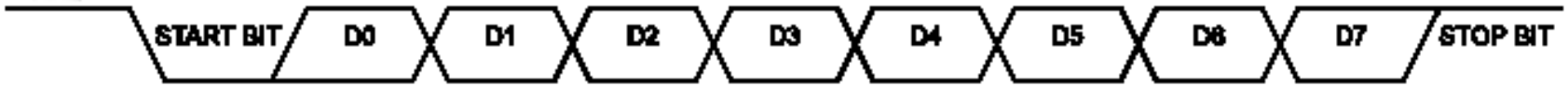
- Simply defines the Rules of talking between two things
- Famous Protocols :
  - TCP/IP
  - HTTP/HTTPS
  - Bluetooth Class 4 or 3
  - UART
  - SPI
  - I2C

Parallel vs Serial



Serial and UART

- The UART provides:
  - Parallel-to-Serial and Serial-to-Parallel conversion
  - Start and Stop Bit framing
  - Parity Generation
  - Baud-Rate Generation (2400-115.2kbps at 3.686 or 7.37MHz)
  - Interrupts
    - Transmit Complete
    - Transmit Data Register Empty
    - Receive Complete

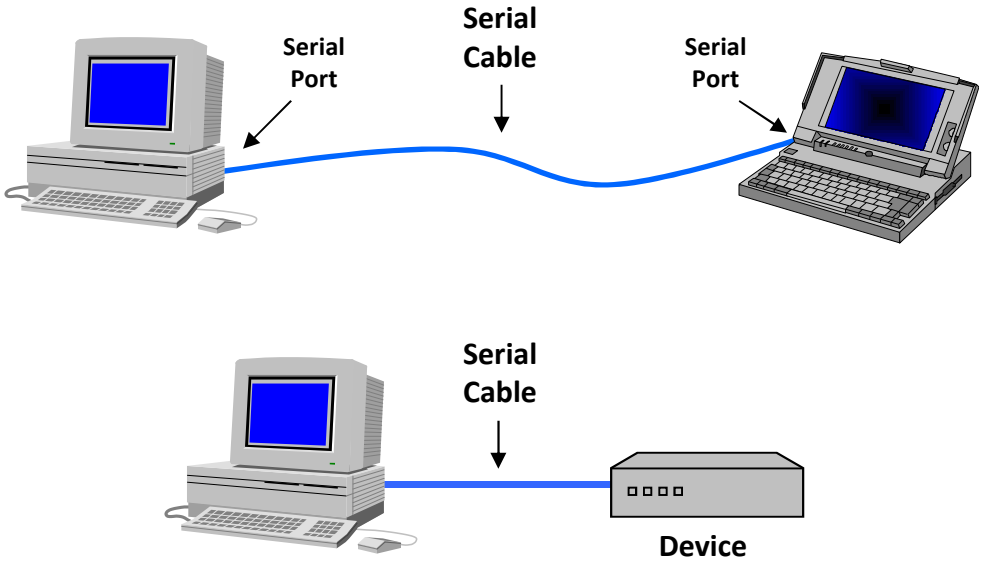


UART

- A UART may be used when:
  - High speed is not required
  - An inexpensive communication link between **two** devices is required
- UART communication is very cheap
  - Single wire for each direction (plus ground wire)
    - Asynchronous because no clock signal is transmitted
  - Relatively simple hardware
- PC devices such as mice and modems used to often use UARTs for communication to the PC

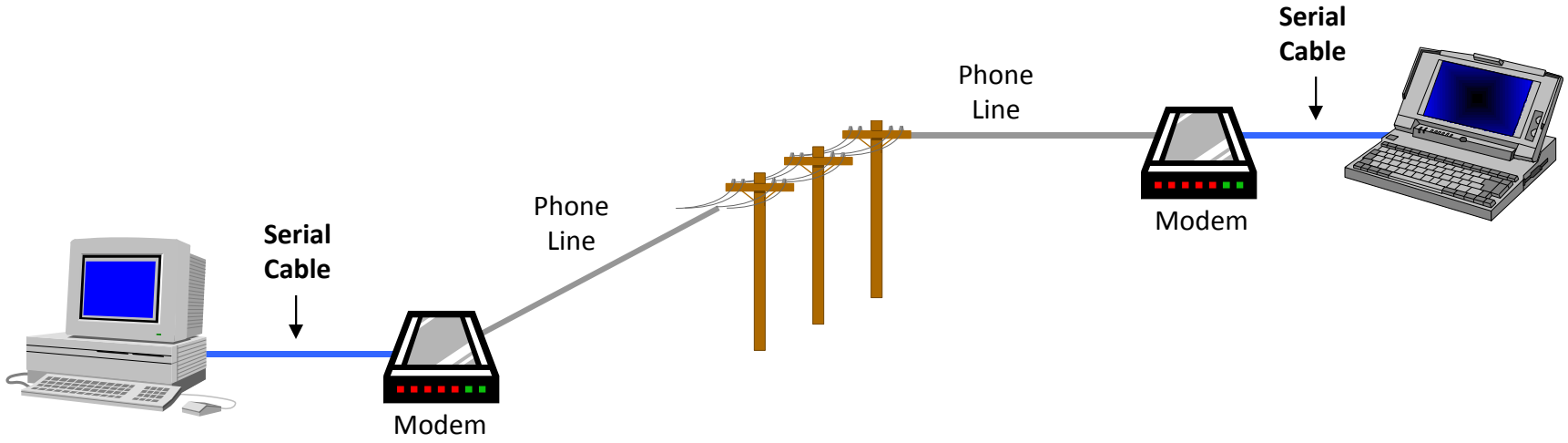
Serial Usage

- PC serial port is a UART!
  - Serializes data to be sent over serial cable
  - De-serializes received data



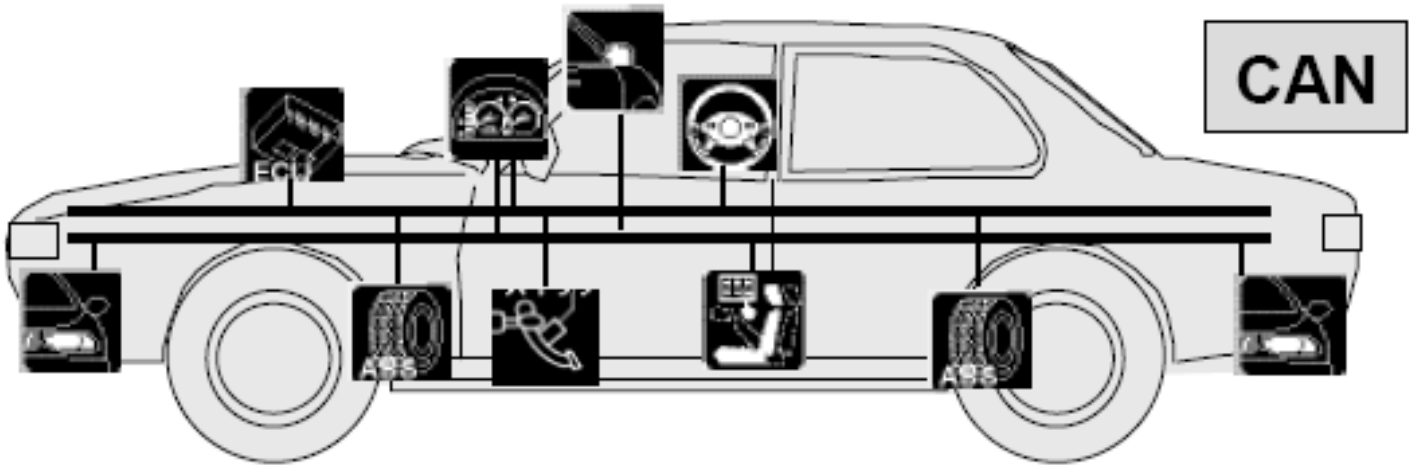
Serial Usage

- Communication between distant computers
  - Serializes data to be sent to modem
  - De-serializes data received from modem



Serial Usage

- A Controller Area Network (CAN bus) is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other in applications without a host computer. It is a message-based protocol, designed originally for multiplex electrical wiring within automobiles, but is also used in many other contexts.



Serial Usage

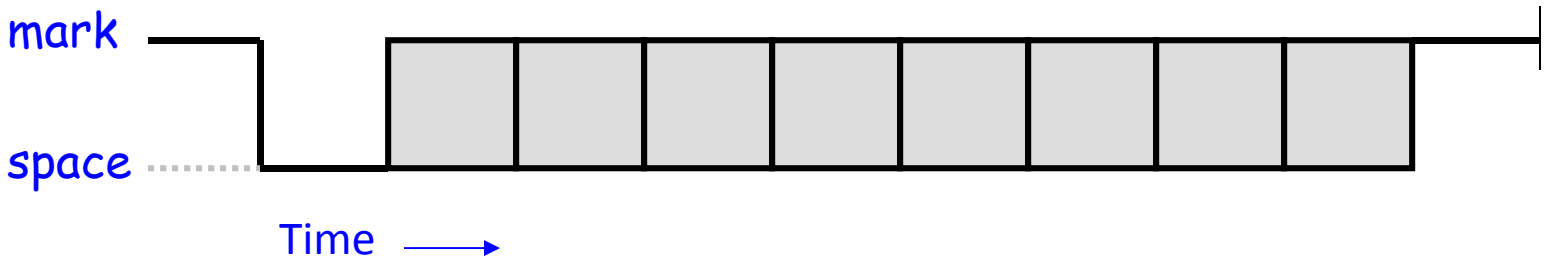
- USB
- SPI
- ADSL
- …

How UART Works ?

- Transmitter
  - Convert from parallel to serial
  - Add start and stop delineators (bits)
  - Add parity bit
- Receiver
  - Convert from serial to parallel
  - Remove start and stop delineators (bits)
  - Check and remove parity bit

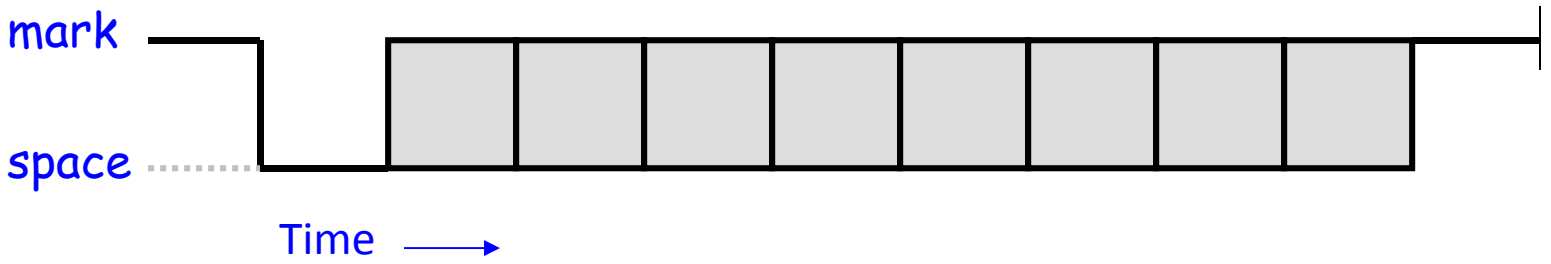
How UART Works ?

- Below is a timing diagram for the transmission of a single byte
- Uses a single wire for transmission
- Each block represents a bit that can be a **mark** (logic ‘1’) or **space** (logic ‘0’)

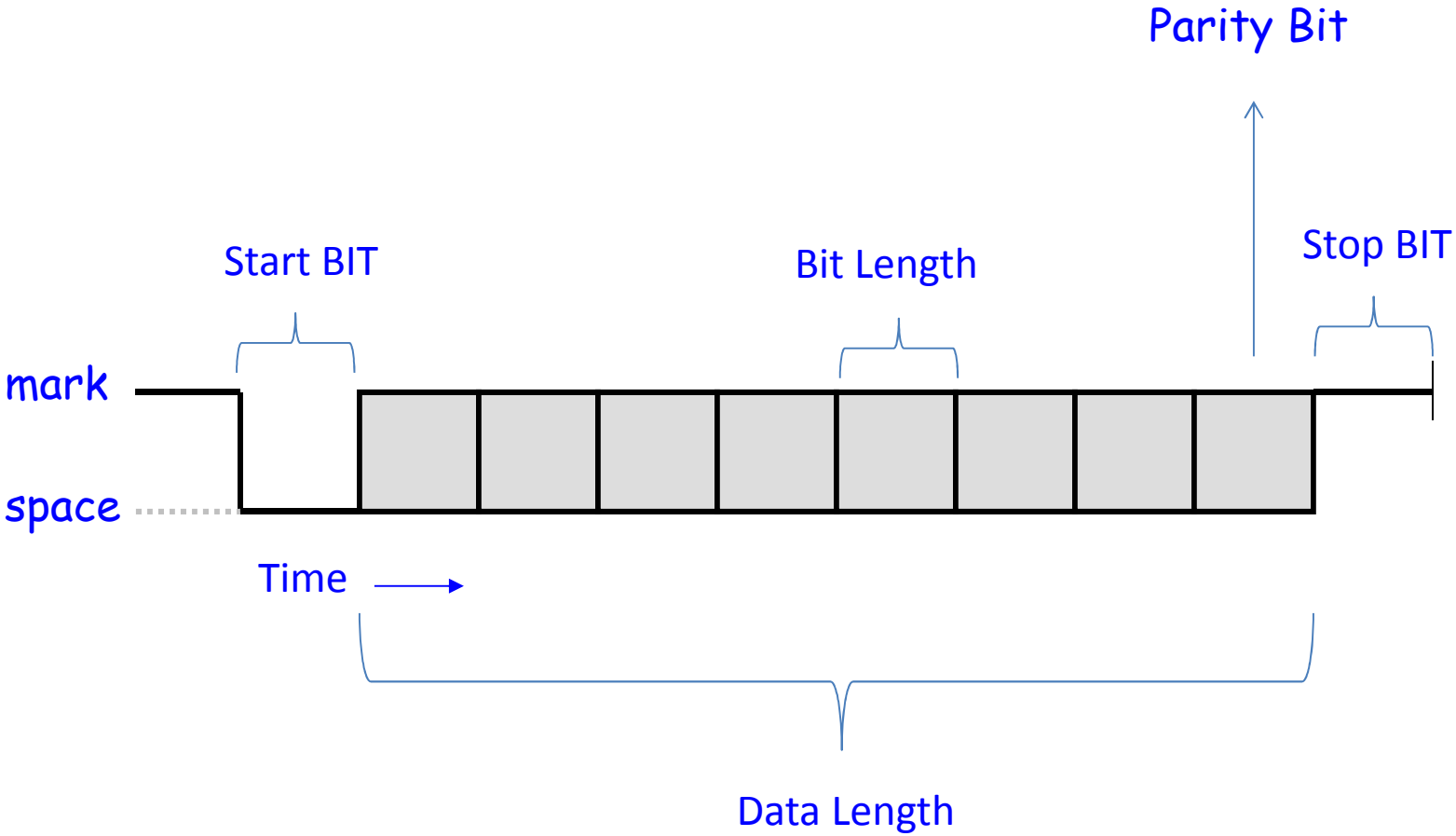


How UART Works ?

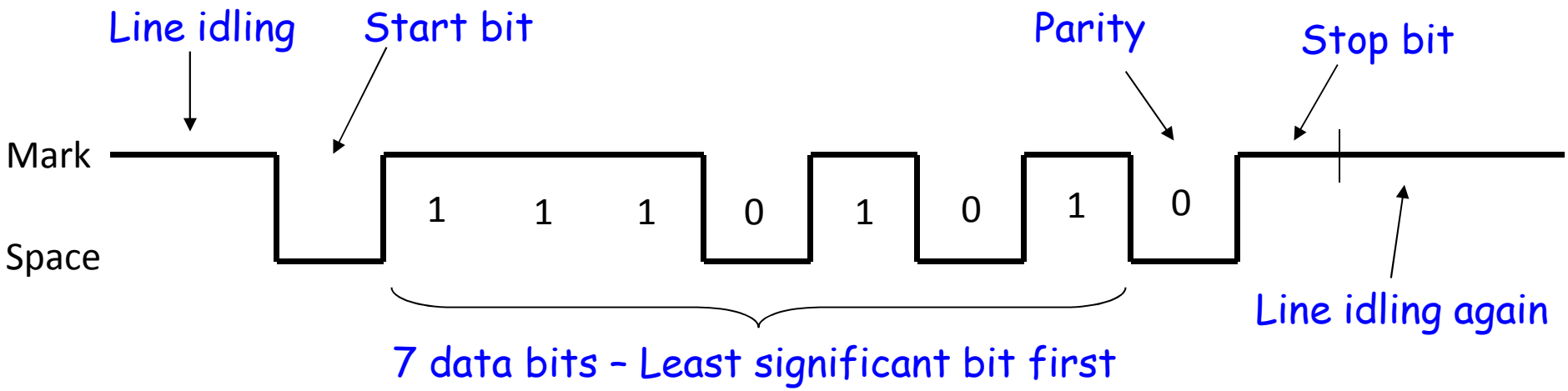
- Each bit has a fixed time duration determined by the transmission rate
- Example: a 1200 bps (bits per second) UART will have a 1/1200 s or about 833.3 µs bit duration
- Transmission rate called Baud rate



How UART Works ?



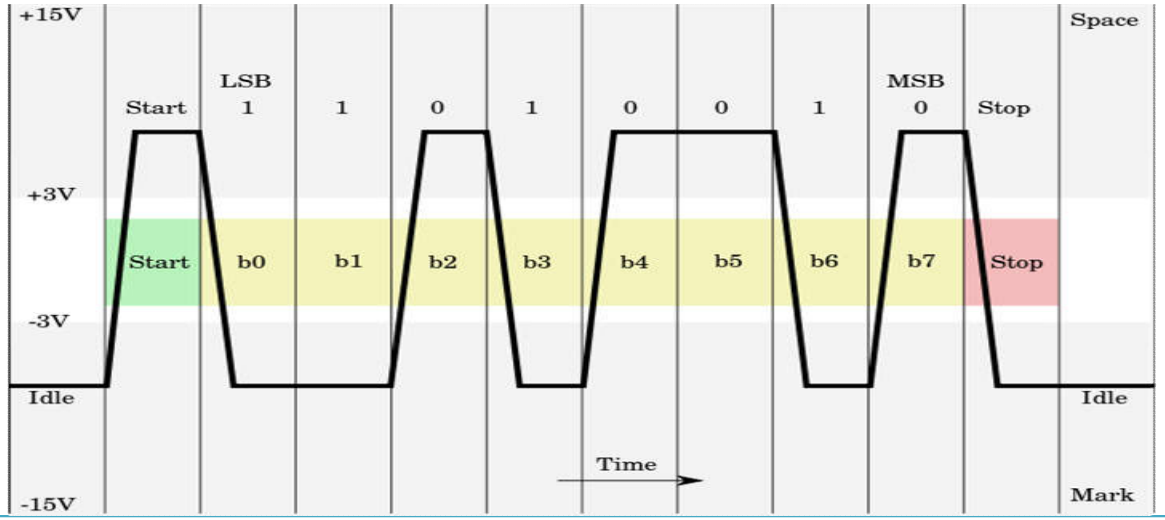
How UART Works ?



Send the ASCII letter ‘W’ (1010111)

UART Definitions

- UART : Universal Asynchronous Receiver Transmitter
- USART : Universal Synchronous Asynchronous Receiver Transmitter
- RS232



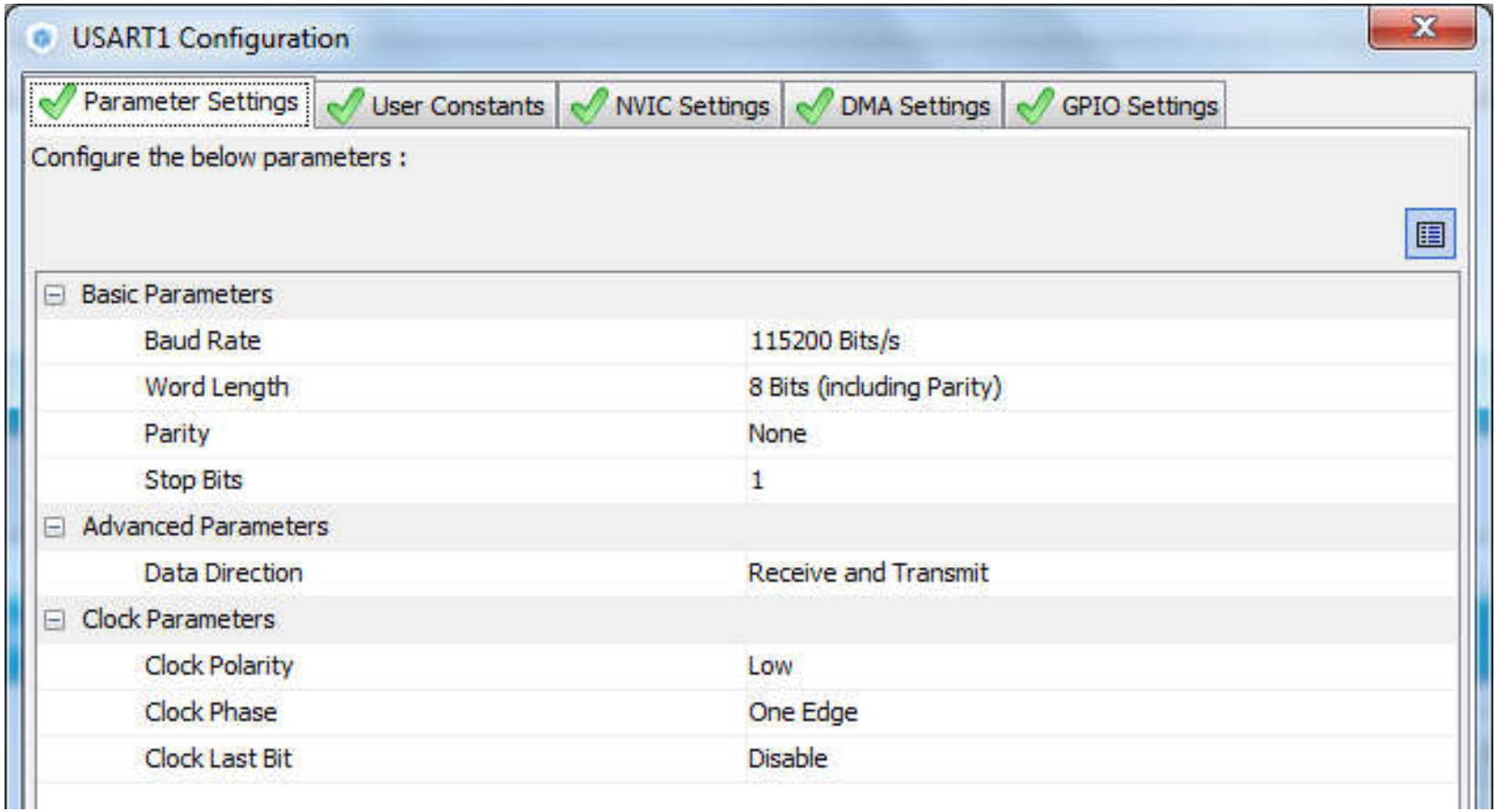
UART Definitions

- RX Pin : Receiving Port
- TX Pin : Transmitting Port
- Hardware Flow Control Pins:
  - RTS Flow
    - RTS : Request To Send
    - CTS : Clear To Send
  - DTR Flow
    - DTR : Data Terminal Ready
    - DSR : Data Set Ready
- Clock Pin

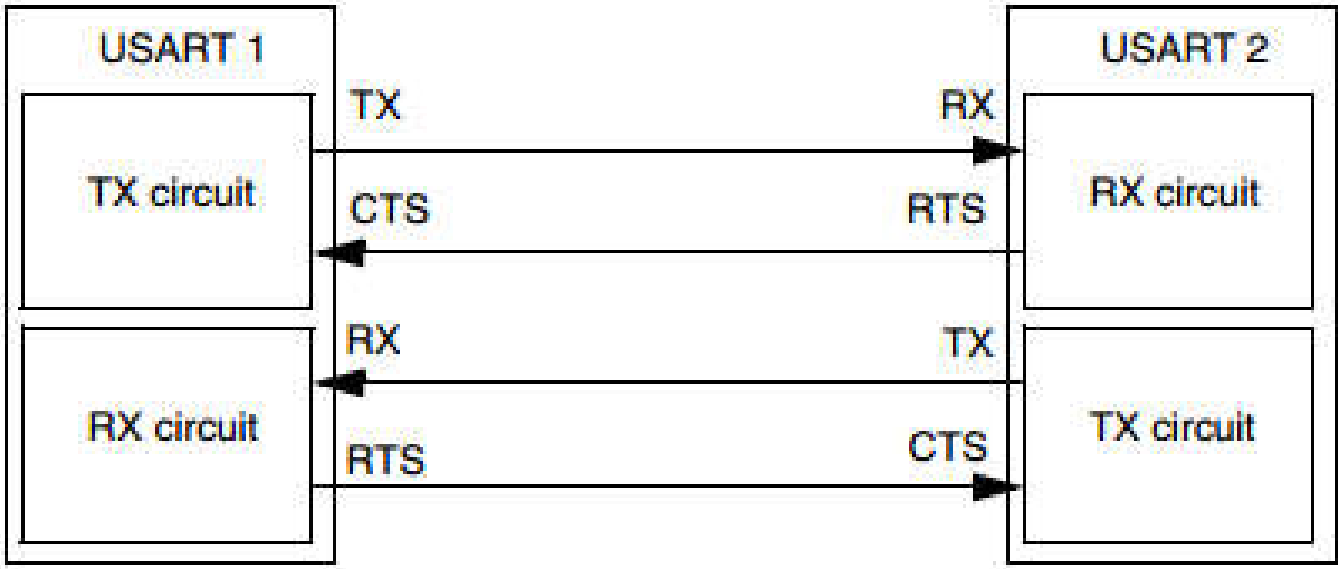
UART Definitions

⊟ Basic Parameters	
Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1
⊟ Advanced Parameters	
Data Direction	Receive and Transmit
Over Sampling	16 Samples

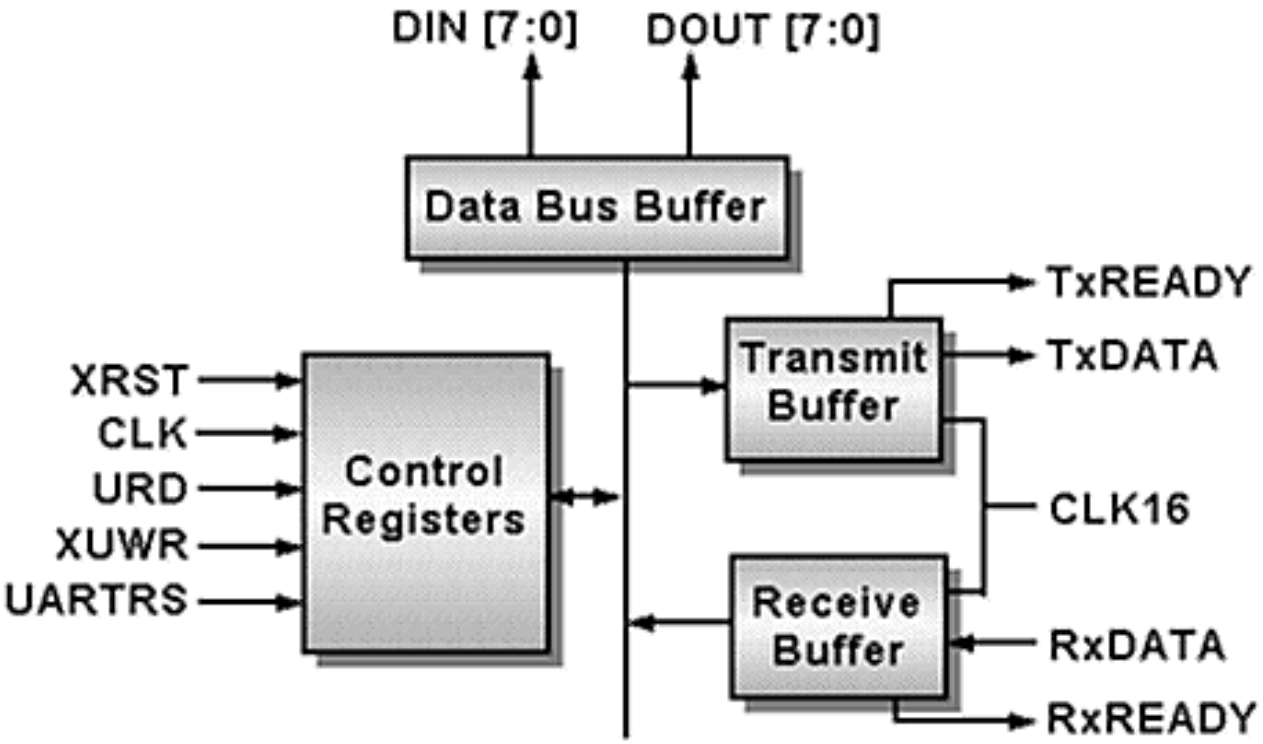
UART Definitions



UART Definitions



UART Subsystem



Reading a data receive register

- First step is finding corresponding registers via user manual

27.6.1 Status register (USART_SR)

Address offset: 0x00
Reset value: 0x00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
						rc_w0	rc_w0	r	rc_w0	rc_w0	r	r	r	r	r

Reading a data receive register

- First step is finding corresponding registers via user manual

27.6.2 Data register (USART_DR)

Address offset: 0x04

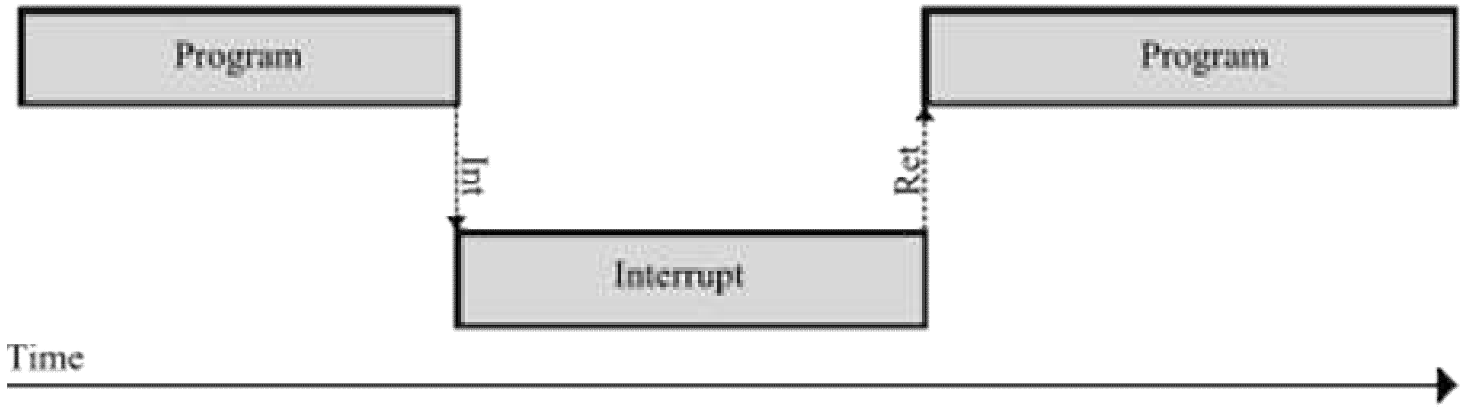
Reset value: Undefined

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							DR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

External Interrupts

Interrupt

- Pooling VS Interrupt
- Priority
- Flag Control
- Mask able Interrupts

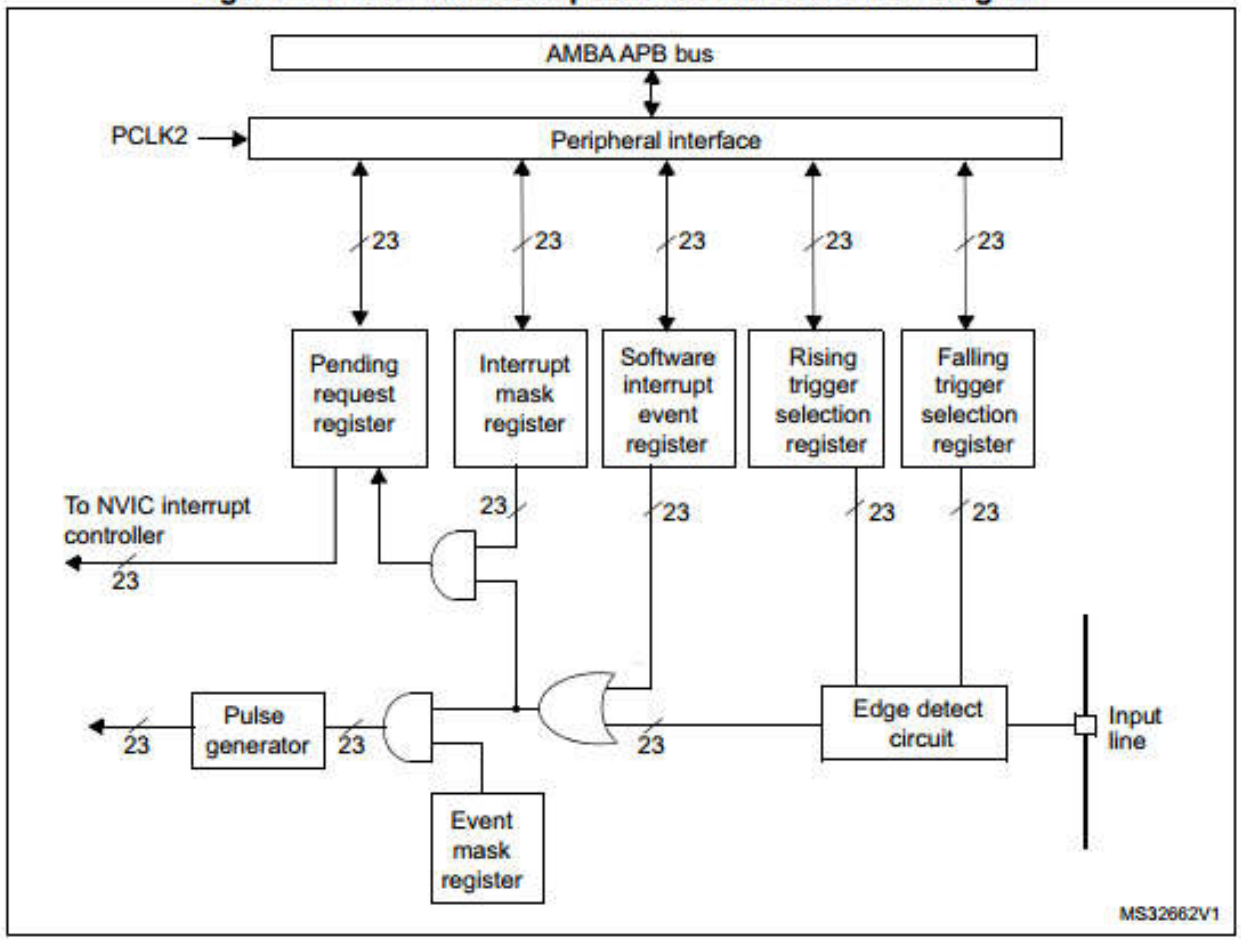


External Interrupts

- NVIC : Nested Vector Interrupt Controllers
- EXTI : External Interrupt/Event Controllers
  - Up to 23 Individual Interrupts
  - Interrupt on Rising , Falling or Both

External Interrupts

Figure 41. External interrupt/event controller block diagram

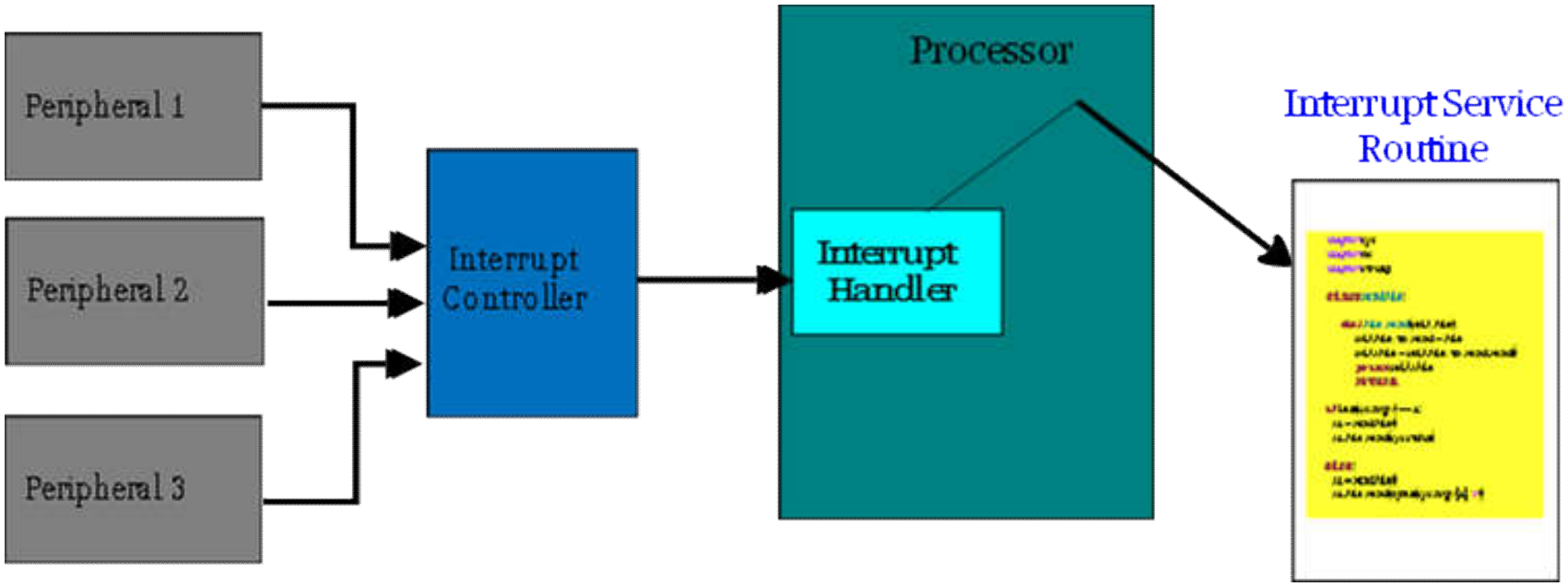


External Interrupts

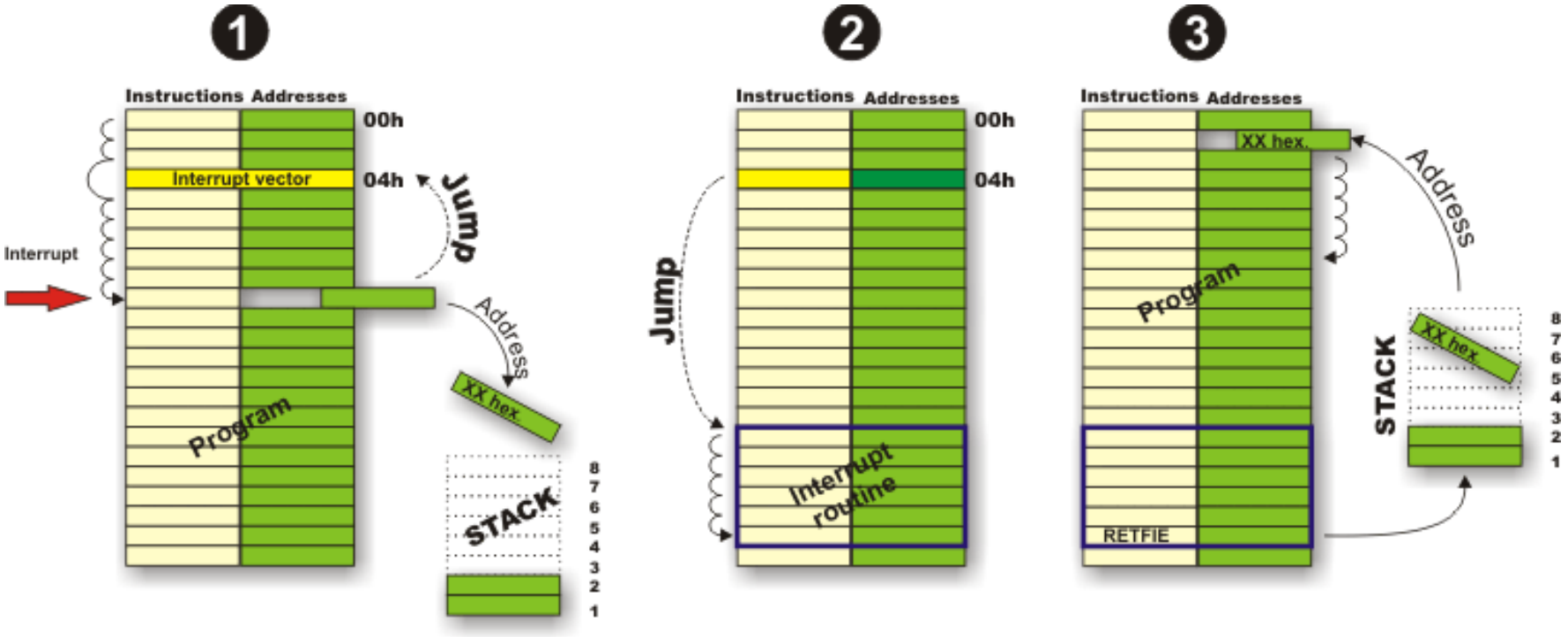
The seven other EXTI lines are connected as follows:

- EXTI line 16 is connected to the PVD output
- EXTI line 17 is connected to the RTC Alarm event
- EXTI line 18 is connected to the USB OTG FS Wakeup event
- EXTI line 19 is connected to the Ethernet Wakeup event
- EXTI line 20 is connected to the USB OTG HS (configured in FS) Wakeup event
- EXTI line 21 is connected to the RTC Tamper and TimeStamp events
- EXTI line 22 is connected to the RTC Wakeup event

Interrupt Service Routine



Nested Vector Interrupt



GPIO External Interrupts

External Interrupts