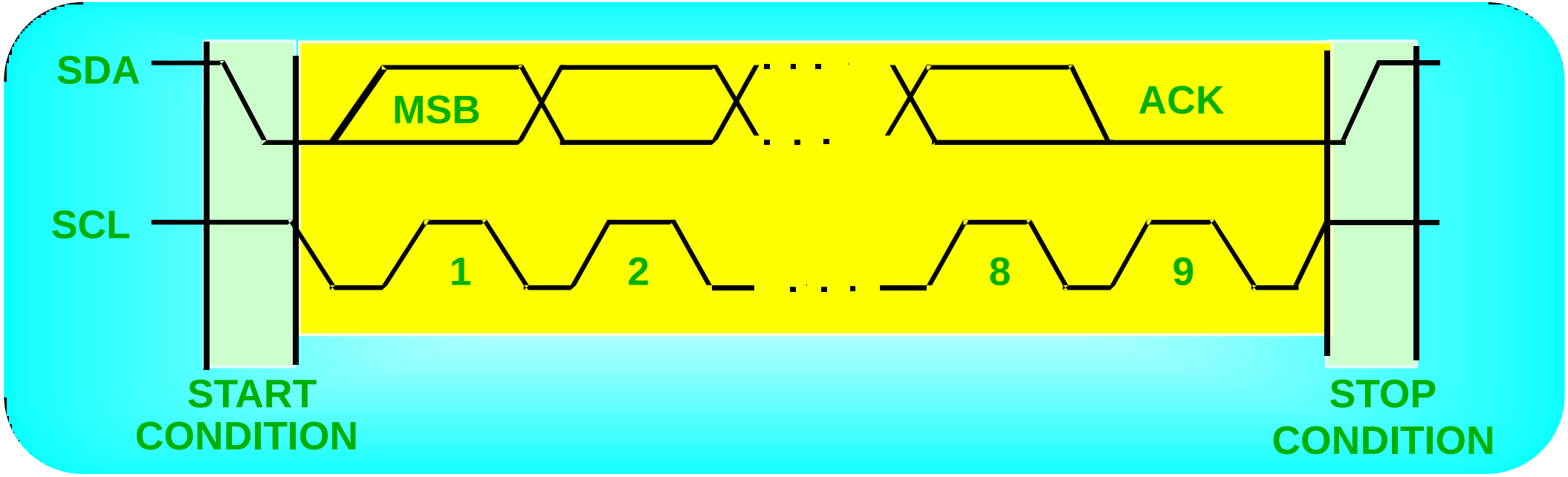
STM8S Communication Peripherals

May 2008

STM8S Technical Training

Inter Integrated Circuit (I^2C)

I2C Protocol



- Only 2 wires
- Clock driven by the Master
- Up to 1023 slaves (with 10-bit address)

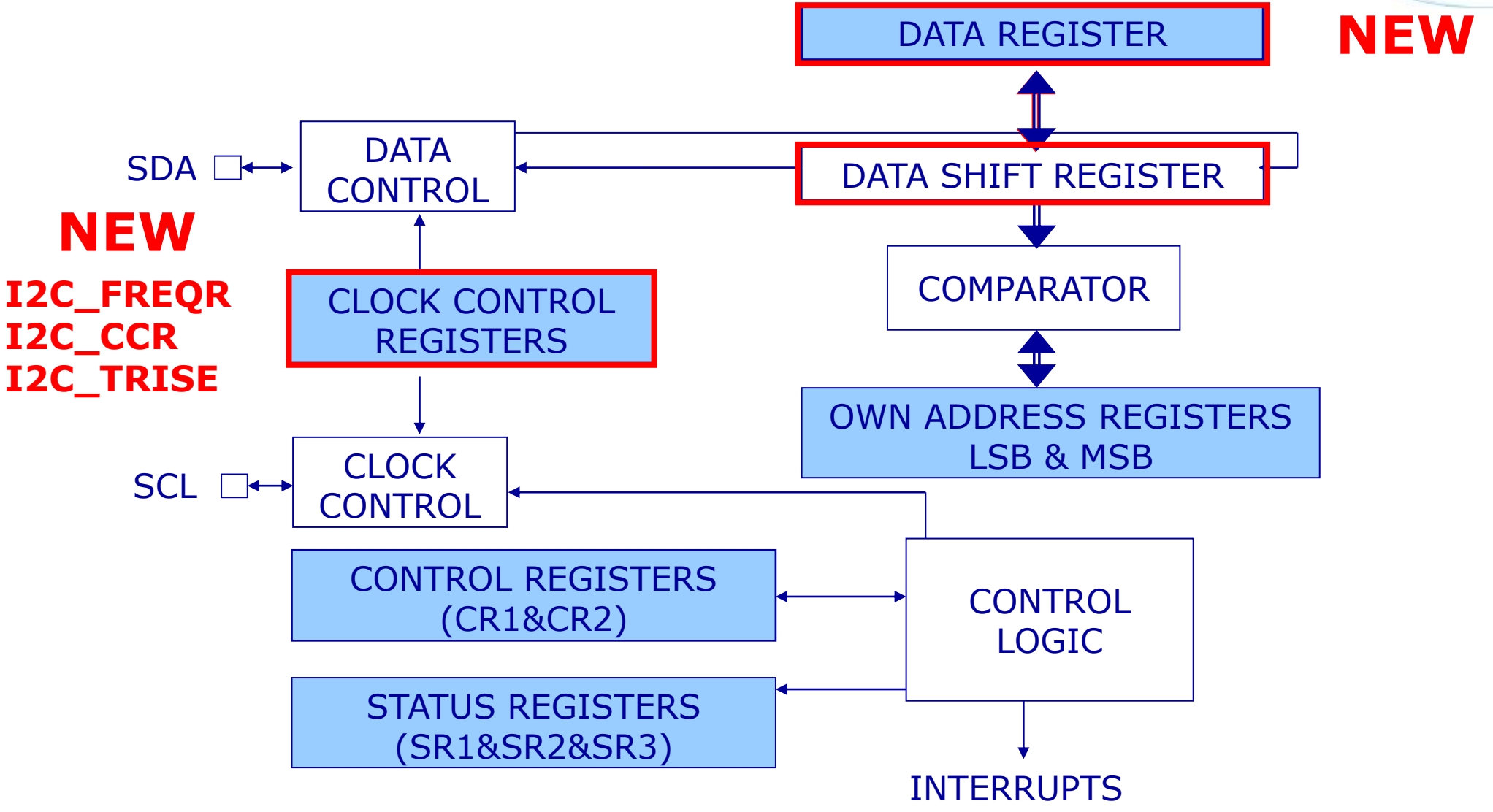
I2C Features (1/2)

- Multi Master and slave capability
- Controls all I²C bus specific sequencing, protocol, arbitration and timing
- Standard and fast I²C mode (up to 400kHz)
- 7-bit and 10-bit addressing modes
- Buffering of the data register
- Stretching of the clock can be disabled in slave mode
- Status flags:
  - Transmitter/Receiver mode flag
  - End-of-Byte transmission flag
  - I2C busy flag

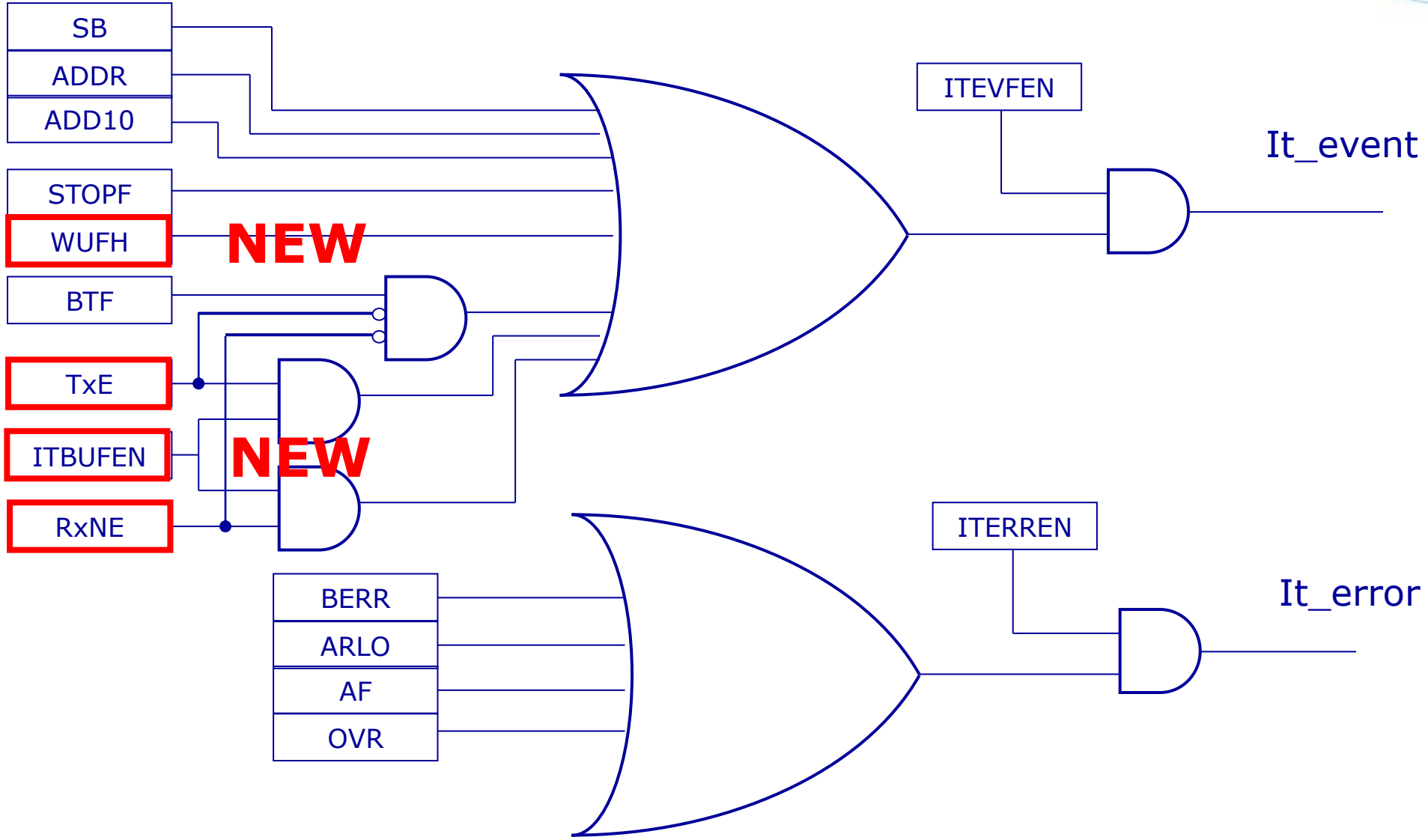
I2C Features (2/2)

- Error flags:
  - Arbitration lost condition for master mode
  - Acknowledgement failure after address/ data transmission
  - Detection of misplaced start or stop condition (Bus error)
  - Overrun/Underrun if clock stretching is disabled
- 2 Interrupt vectors:
  - 1 Interrupt to manage addressing and data communication
  - 1 Interrupt for error condition

Block Diagram



I²C Interrupt mapping diagram



Quiz

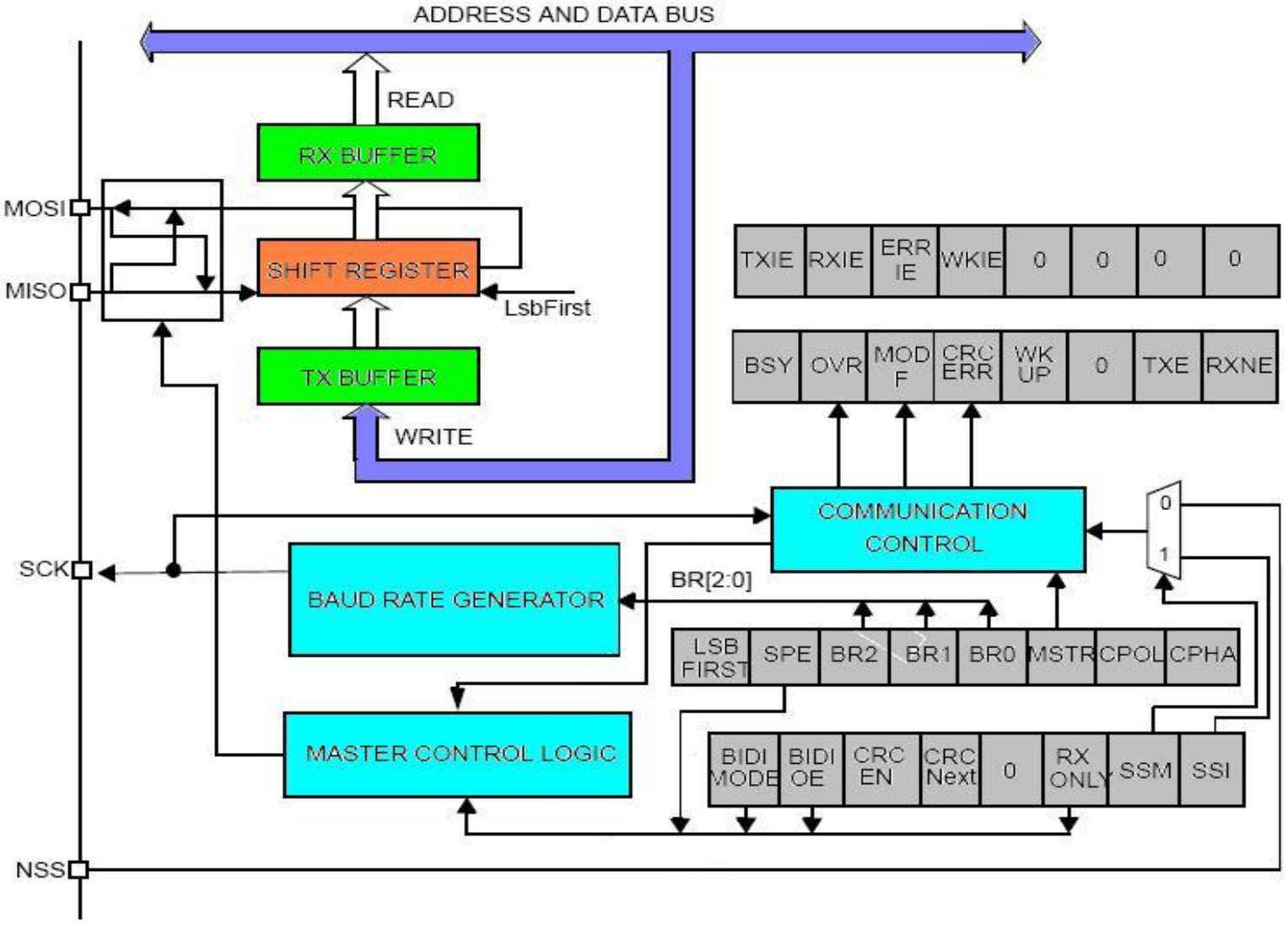
- How is defined start and stop condition of the I2C protocol and who performs it ?
- Who and how confirms any I2C transmission ?
- How many addressing modes are possible and what is their difference in I2C protocol ?

STM8S SPI PERIPHERAL

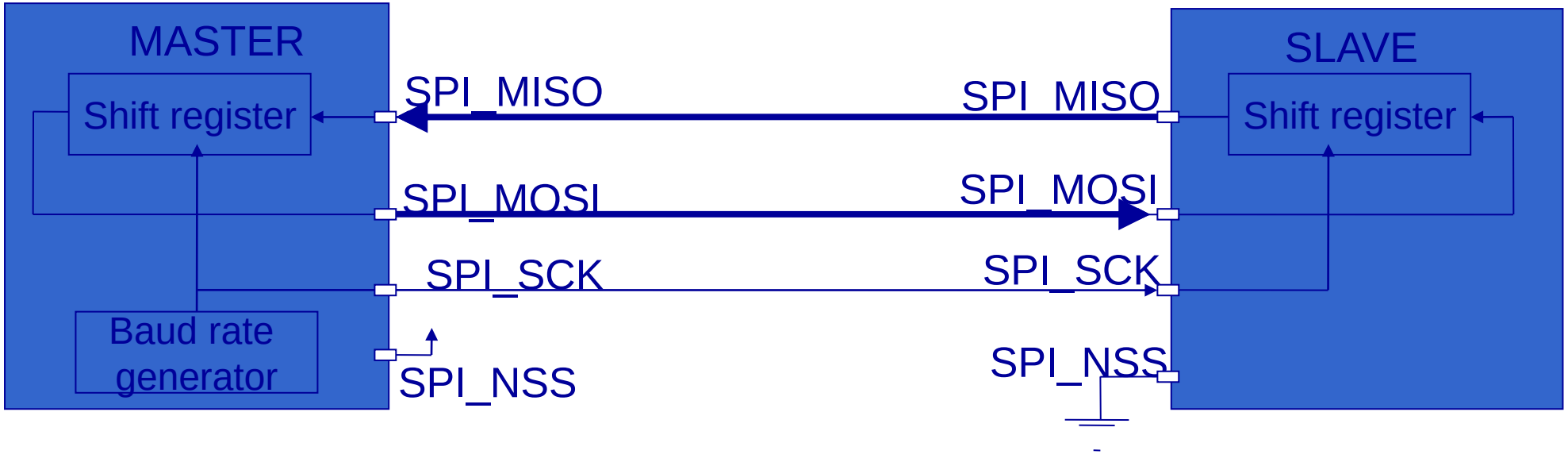
STM8S SPI features

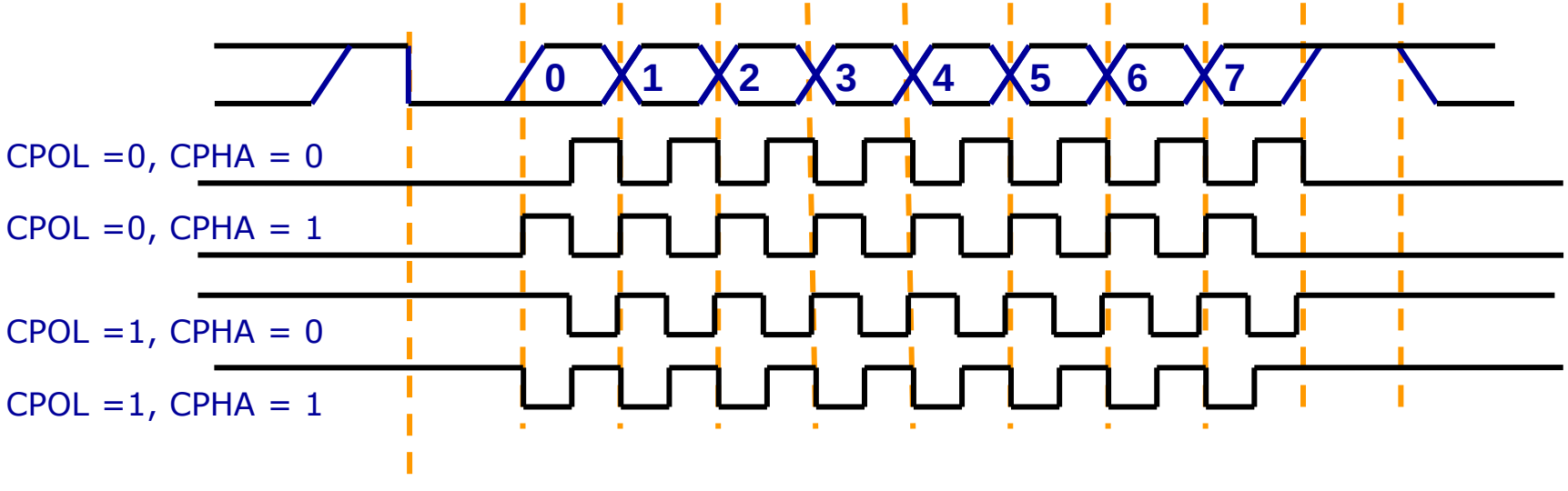
- Full-duplex three wires synchronous communication
- Simplex two wire synchronous communication with or without bidirectional data line
- 8 Master mode baud rates
- Baud rate up to $F_{master}/2$: (10MHz) – limited by the I/O maximum speed
- Software or hardware management of Slave Select pin
- Programmable polarity and phase
- Programmable bit order (MSB or LSB first)
- Up to 6 interrupt sources
- Hardware CRC capability
- Wake-up capability from low power mode

SPI block diagram

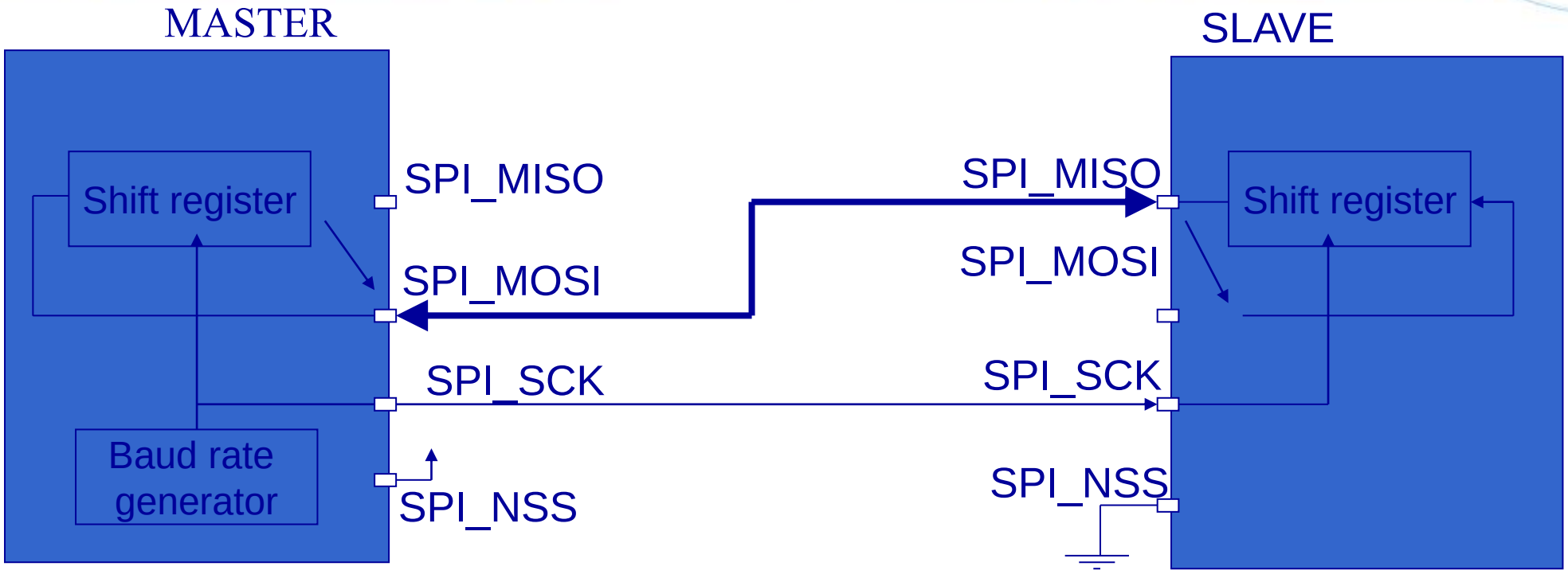


SPI connections - Duplex mode





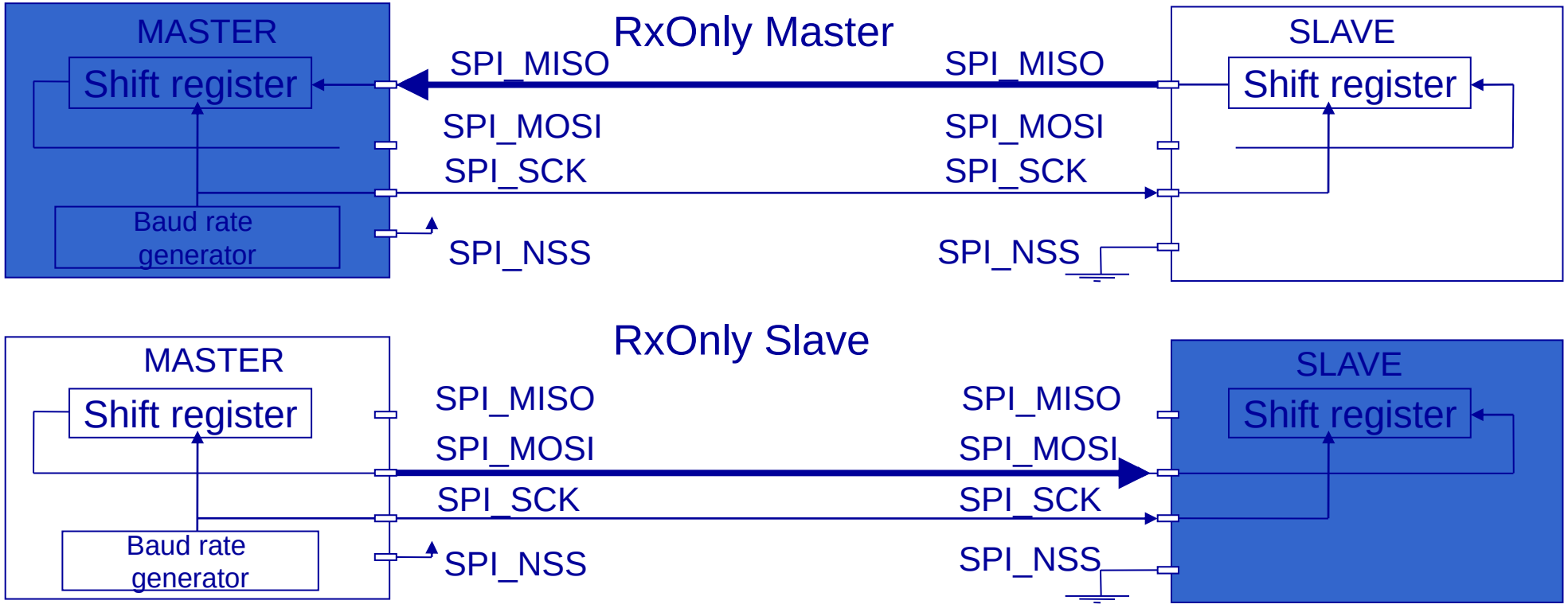
SPI connections – Simplex, Bidirectional mode



	Master	Slave	data flow
BDOE	1	0	M ⇨ S
BDOE	0	1	M ⇦ S

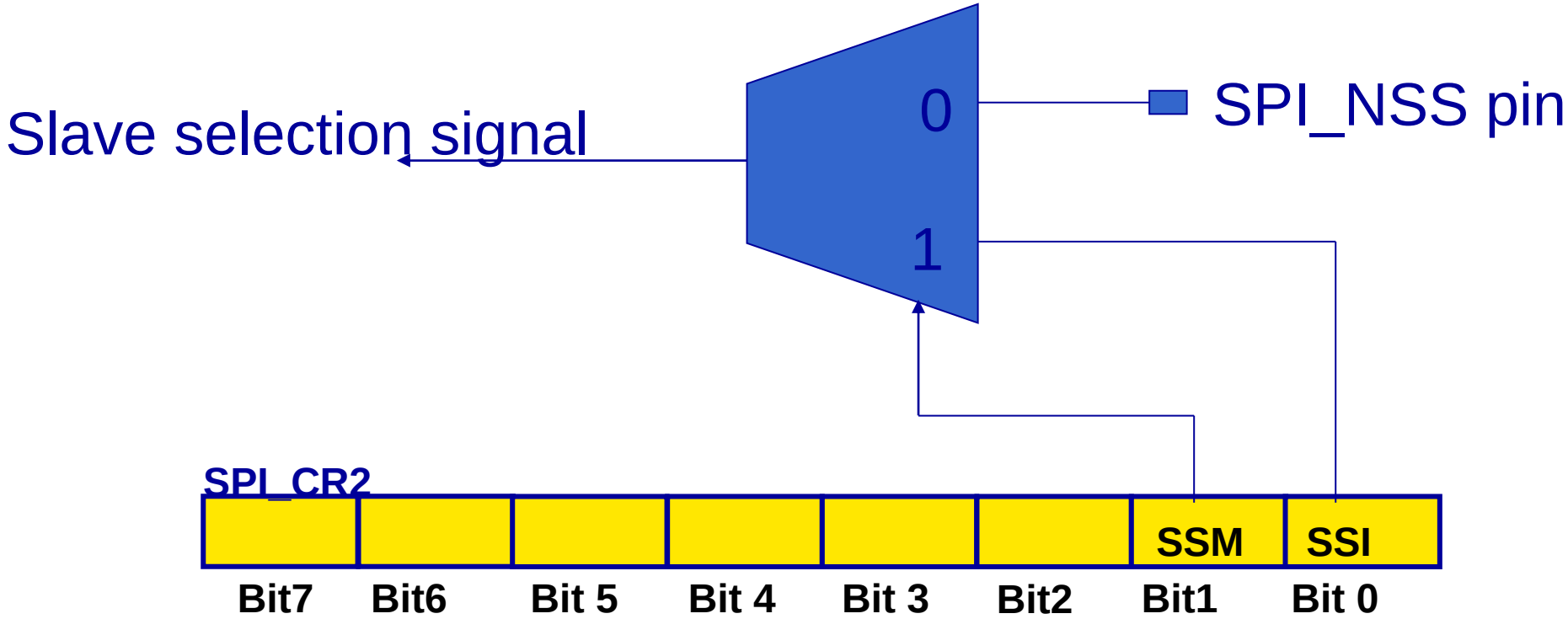
- Bidirectional mode is selected by setting BDM bit in SPI_CR2 register
- The exchange direction is set with BDOE bit
- MOSI is used on the Master and MISO is used on the slave

SPI connections – Simplex, RxOnly mode



- If the slave is put in RxOnly mode the MISO pin of the slave is freed for other purpose.
- Everything else operates exactly like in full duplex mode

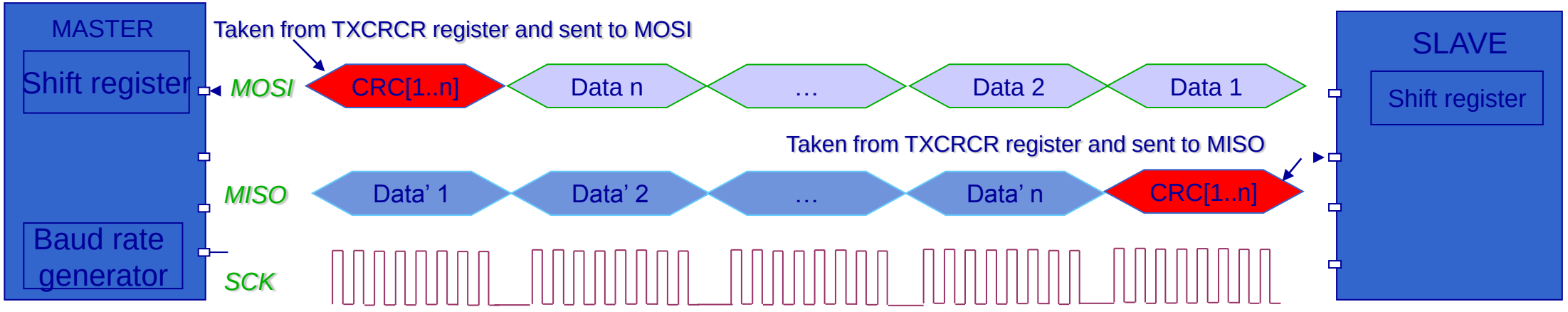
Software Slave Selection



Management of the Slave Select signal by software is alternative to using the NSS pin to control the Slave Select signal

CRC calculation

- A CRC calculation help improve the transmission reliability.
- Separate calculator are implemented for Tx and Rx data.
- The CRC is calculated using a programmable polynomial serially on each bit
- CRC is transmitted after the end of the pending transmission as soon as the CRCNEXT is set.
- If at the end of the transmission of the transmit CRC the contents of the computed reception CRC is different from the last received byte, a CRC error is generated.



Communication and Error Management

- **Communication management**
  - Transmission buffer empty (IT)
  - Receive buffer full (IT)
  - Busy flag
  - Wake-up flag (IT)

- **Error Management**
  - Overrun condition

    OVR bit is set if a new data is received but the previous one has not been read
  - Master Mode fault

    MODF bit is set if the device is the master and NSS pin (or SSI bit if SSM is set) goes low. The SPI cell then is automatically deactivated
  - CRC error

    If CRC is enabled, CRCERR bit is set whenever the received CRC does not match the one calculated.

Quiz

- What is the maximum speed of the STM8S SPI?
- How to reduce as much as possible the number of pins used by the SPI of the STM8S?
- What happens if the NSS pin goes to zero while the device is configured in master mode?
- What are the different error flags?

STM8S – USART PERIPHERAL
Universal Synchronous/Asynchronous Receiver Transmitter

STM8S U(S)ART Features overview (1/2)

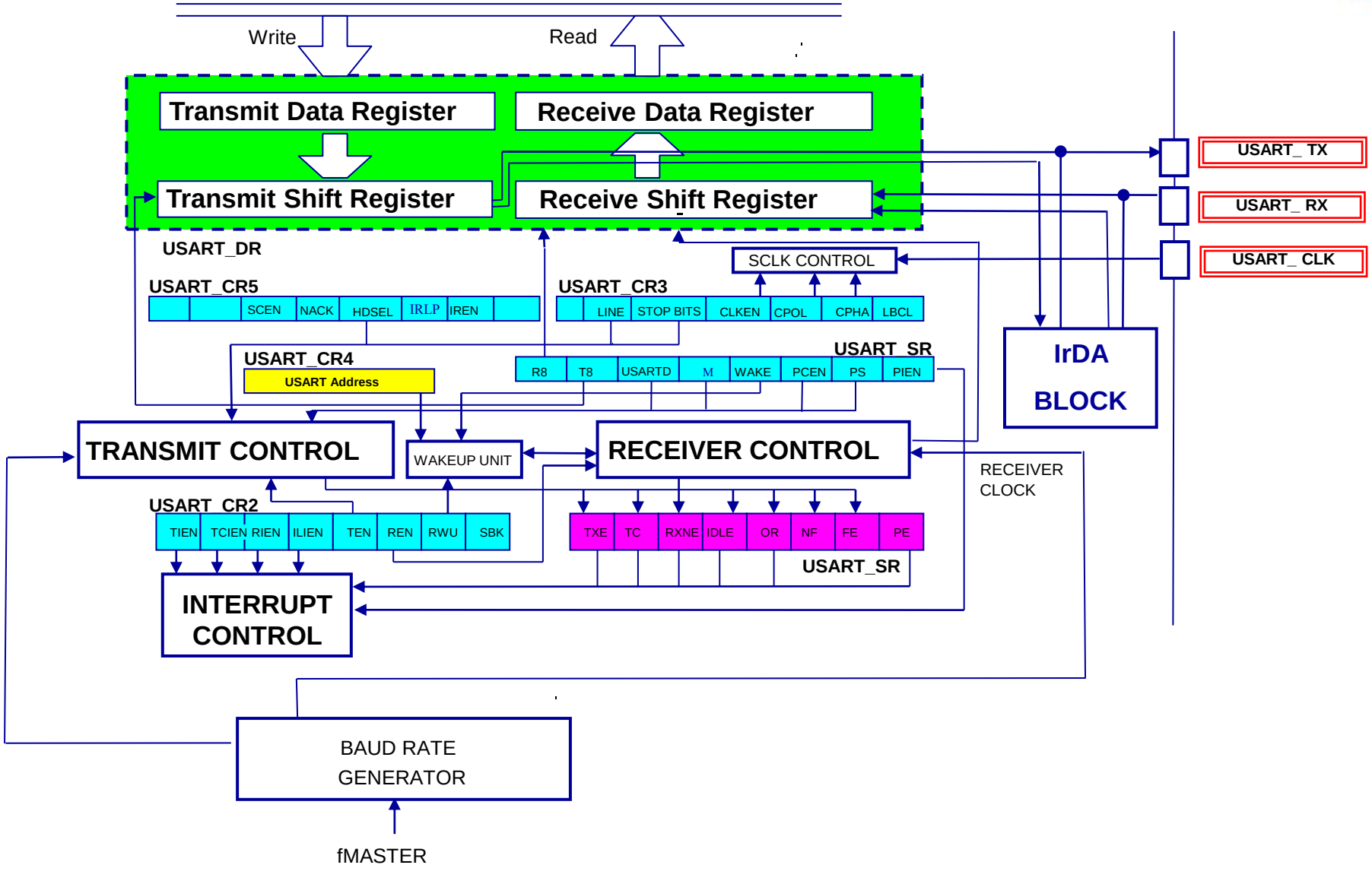
- Asynchronous full-duplex two wires communication
- Asynchronous half-duplex single wire communication
- Synchronous communication in master mode
- Multi-Processor communication using node address

- LIN Master mode
  - LIN Break and Delimiter generation
  - LIN Break and Delimiter detection with separate flag and interrupt source for readback checking
- Smartcard interface
- IrDA interface

STM8S U(S)ART Features overview (2/2)

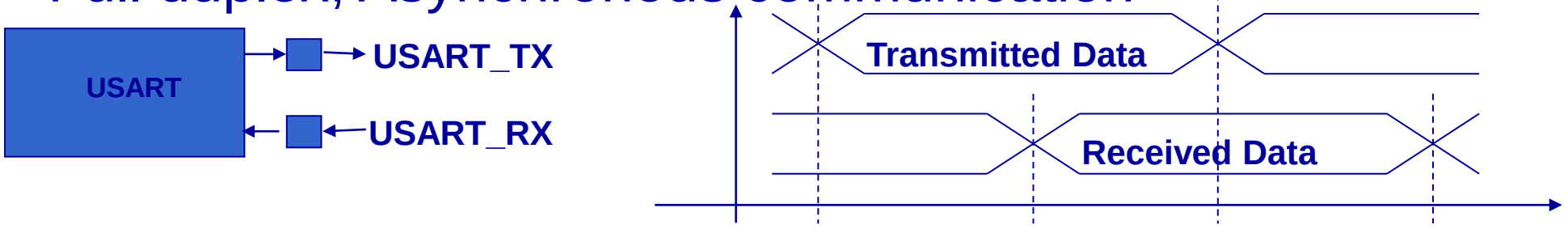
- High resolution baud rate generator systems
  - rates up to 2.5 M baud (up to 1.5 M baud for asynchronous communication)
- Programmable data word length
  - 8 or 9 bits
- Configurable stop bits
  - 1, 2, 1.5 (in Smart Card mode)
- Parity bit Management
  - even, odd
- 4 error detection flags
  - Noise, Frame (~Break), Parity, Overrun
- 6 interrupt sources with flags on two vectors
  - Transmit Data Register Empty, Transmission Complete
  - Received Data Ready, Parity Error, Idle Line, LIN Break & Delimiter
- Reduced power consumption mode
- Selectable clock & phase polarity
- Mute mode

STM8S USART block diagram

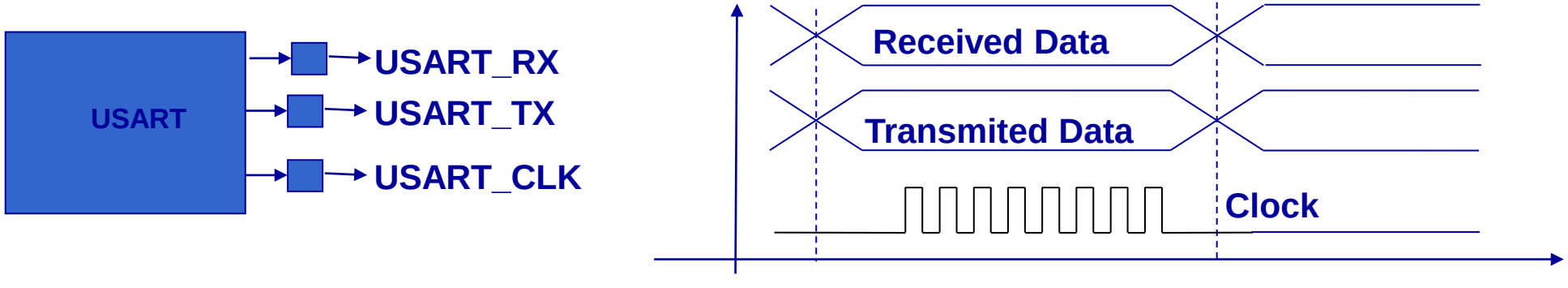


STM8S USART Communication modes

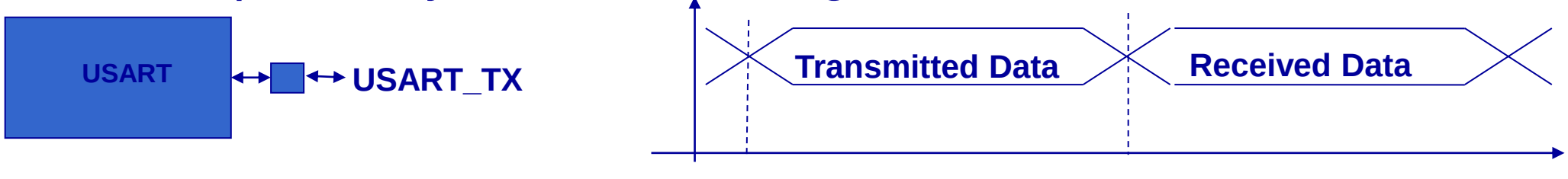
- Full duplex, Asynchronous communication



- Full duplex, Synchronous transmission



- Half Duplex, Asynchronous, Single Wire communication



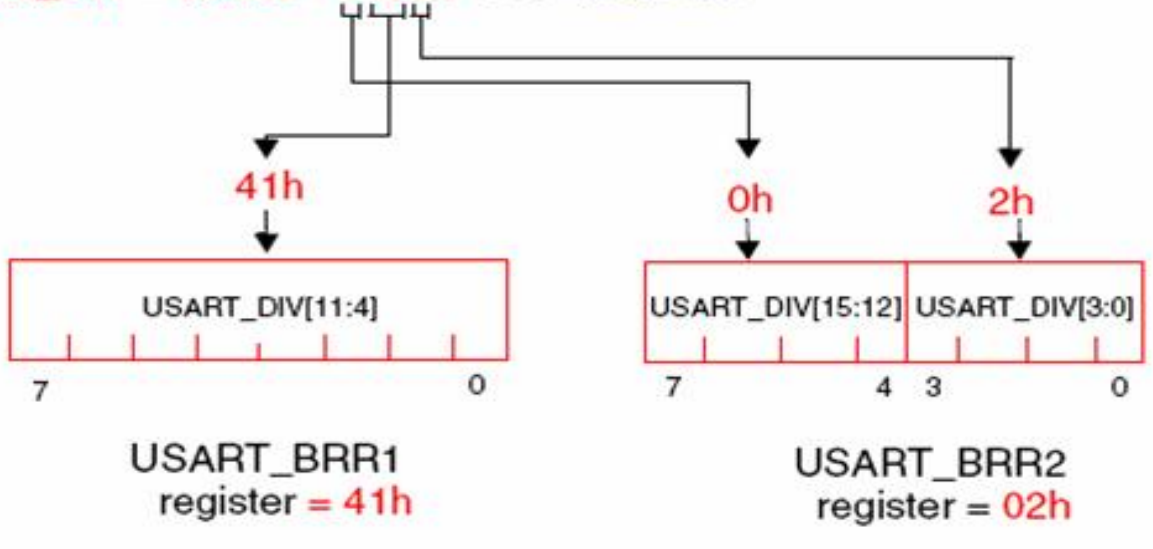
Baud Rate Setting

- $$USART\_DIV = \frac{f_{MASTER}}{BaudRate} \quad (>= 16)$$

Example: To obtain 9600 baud with f_{MASTER} = 10 MHz.

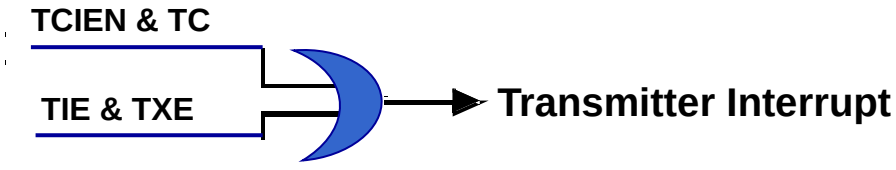
USART_DIV = 10 000 000/9600

USART_DIV = 1042d = 0412h See *Table 40.*

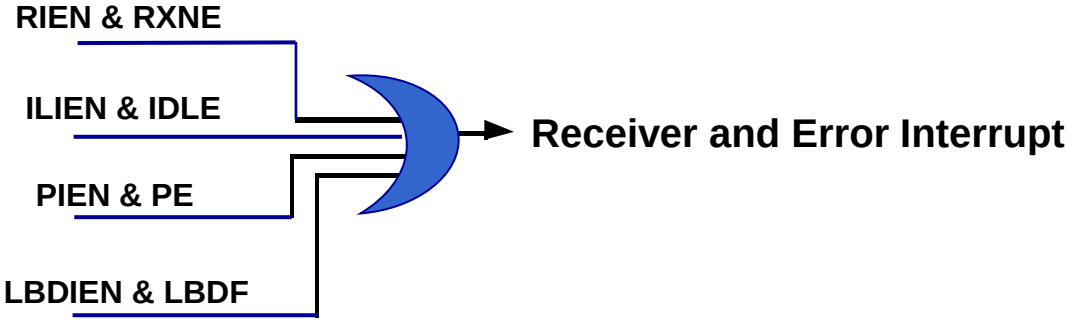


Interrupts

- Transmitter
  - Transmit Data Register Empty
  - Transmission Complete



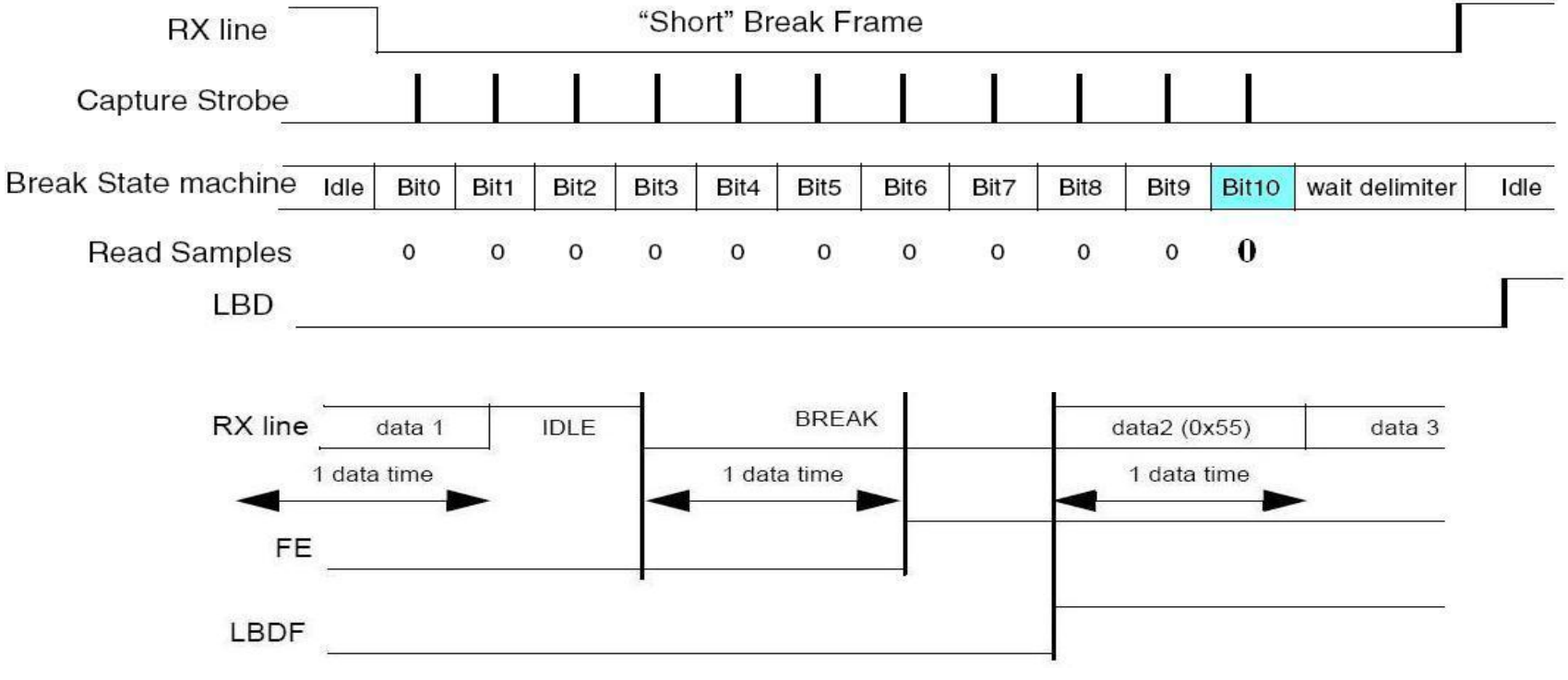
- Receiver
  - Read Data Register Not Empty
    - Noise error
    - Framing error (~break)
    - Overrun error
  - Idle Line
  - Parity error
  - LIN Break & Delimiter



- **Wait:** No effect on USART (exit on interrupt)
- **Halt:** USART stop transmitting/receiving

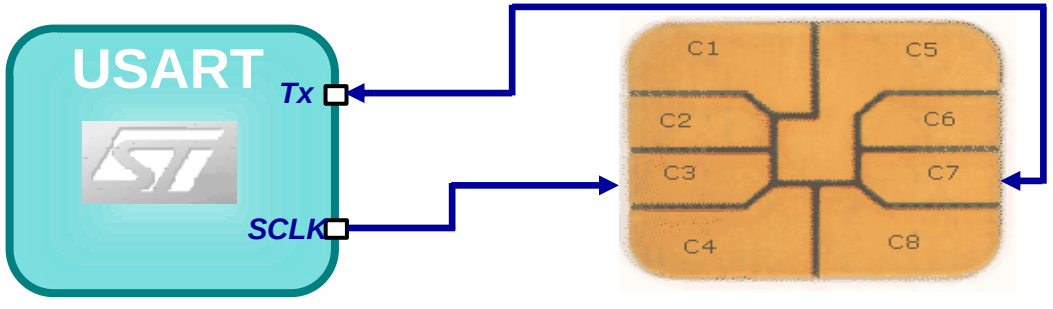
LIN Master mode

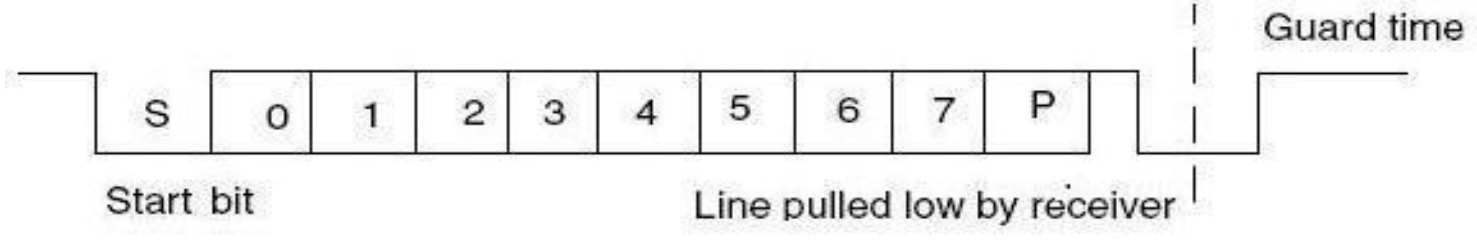
- LIN 13-bit Break transmission
- LIN Break and Delimiter detection for readback checking
- Refer to LINUART section for more details



Smart Card mode

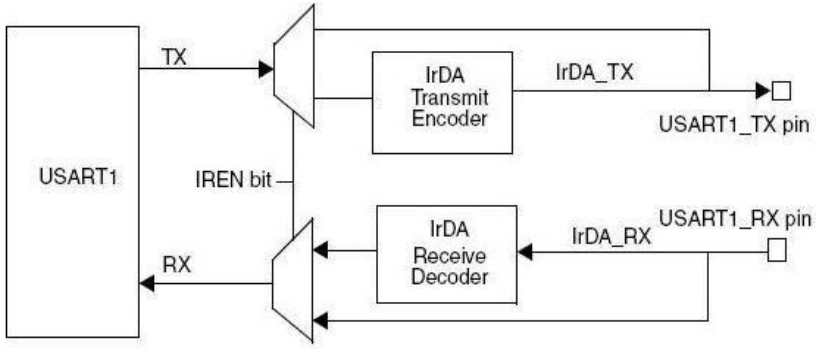
- USART supports Smart Card Emulation ISO 7816-3
  - Single wire Half-Duplex, Asynchronous communication, Clock Output (SCLK)
  - 8 Bits data, 1 parity, 1.5 Stop Bits, TE and RE should be set
  - Transmit and receive managing is automatic
  - Parity Error Generation with NACK transmission
  - Programmable Guard Time – delay assertion of Transmission Complete flag
  - Programmable Clock Prescaler to guarantee a wide range clock input for Smart Card

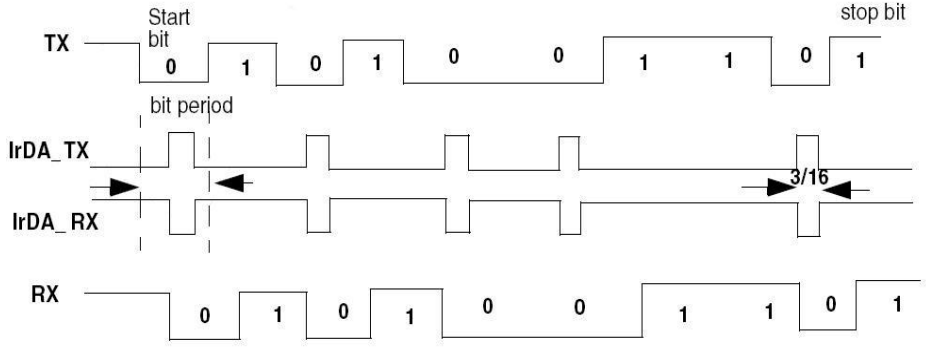




IrDA SIR Encoder Decoder

- USART supports the IrDA Specifications
  - Half-duplex, NRZ modulation to RZI (logical 0 represent Infrared light pulse)
  - Max bit rate 115 200 bps
  - 1 stop bit should be set
  - Adjustable glitch detection
  - Modulation:
    - Normal Mode - centered pulses with width 3/16 of current bit period
    - Low power mode: fixed pulse width at IrDA acceptable minimum (>1.42 us) (prescaler setting: 1.42MHz< fmaster/prescaler <2.12MHz)





Quiz

- Which communication modes the USARTs in the STM8S device support?
- In which mode the USART should be configured to send and receive data on the same pin?
- In synchronous mode, can we transmit and receive in the same time?
- How many stop bits you should have to set for Smartcard interface and how you can indicate parity error?

STM8S - LINUART PERIPHERAL
LIN Universal Asynchronous Receiver Transmitter

LINUART Features overview

- Full-duplex two wires asynchronous communication
- LIN Master and Slave communication
- Multi-Processor communication using node address

LINUART Features / UART mode

- High resolution baud rate generator systems
  rates up to 2.5M baud (up to 1.5 M baud for asynchronous communication)
- Programmable data word length
  8 or 9 bits
- Configurable stop bits
  1, 2
- Parity bit Management
  even, odd
- 5 error detection flags
  Noise, Frame, Parity, Overrun, LIN header
- 7 interrupt sources with flags on two vectors
  - Transmit Data Register Empty, Transmission Complete
  - Received Data Ready, Parity Error, Idle Line, LIN header receive, LIN Break & Delimiter
- Reduced power consumption mode
- Mute mode

LINUART Features / LIN mode

- Master mode
  - LIN Break and Delimiter generation
  - LIN Break and Delimiter detection with separate flag and interrupt source for readback checking
- Slave mode
  - Autonomous header handling and mute mode to filter responses
  - Identifier parity error checking
  - LIN automatic resynchronization, allowing operation with internal RC oscillator (HSI)
  - LIN B&D detection at any time, even during a byte reception
  - Header errors detection
    - Delimiter too short
    - Synch field error
    - Deviation error (if automatic resynchronization is enabled)
    - Framing error in Synch Field or Identifier Field
    - Header timeout

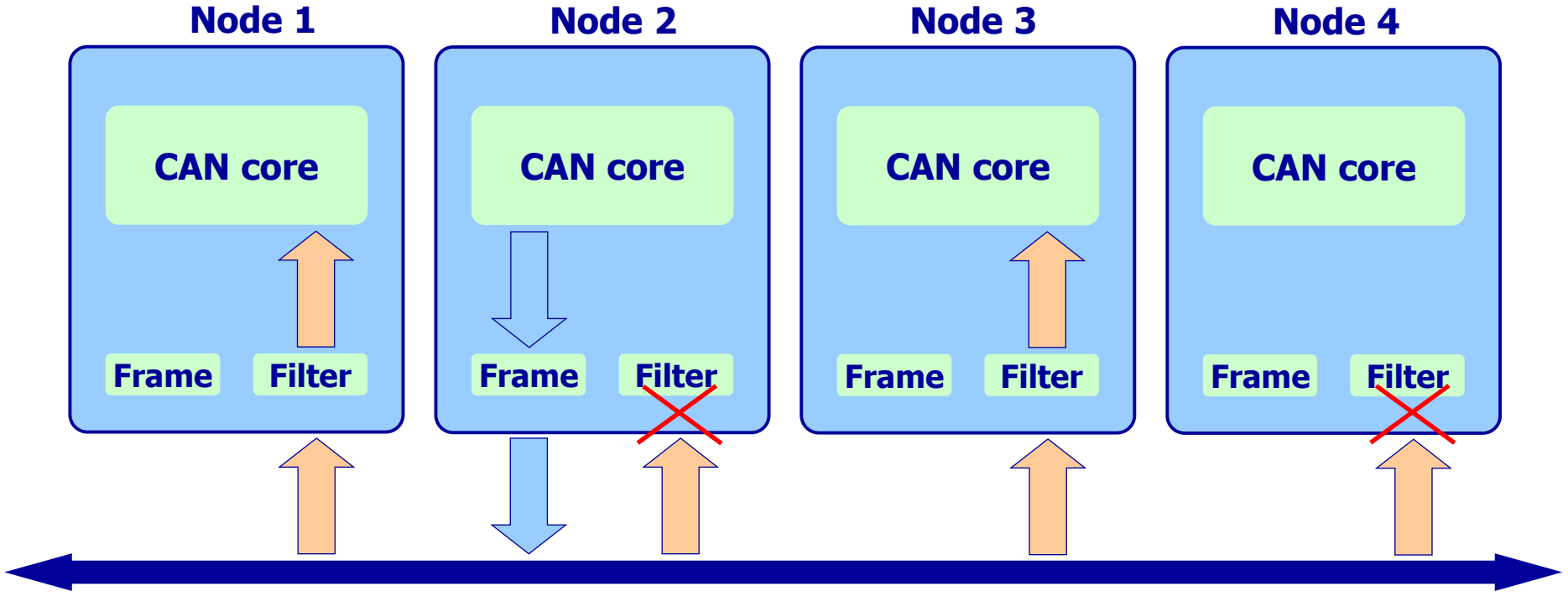
Quiz

- What is the main feature of LINUART or USART to implement a LIN master node?
- How many interrupts are generated during a header transmission in LIN master mode?
- How many interrupts are generated during a normal header reception in LIN slave mode?
- Is it possible to implement a LIN slave node, using a non precise clock?

STM8S CAN PERIPHERAL
Controller Area Network

Principle of data exchange using CAN

- Broadcast communication mechanism



CAN Features (1/2)

- Main features:
  - Supports CAN protocol version 2.0B Active
  - Bit rates up to 1Mbit/s
  - Support the Time Triggered Communication option (TTCAN)
  - Provision for external clock to avoid problem due to clock deviation at high baud rates
- Transmission
  - Three transmit mailboxes
  - Configurable transmit priority
  - Time Stamp on SOF transmission
- Reception
  - One receive FIFO with three stages
  - Six scalable filter banks
  - Identifier list features
  - Configurable FIFO overrun
  - Time Stamp on SOF reception

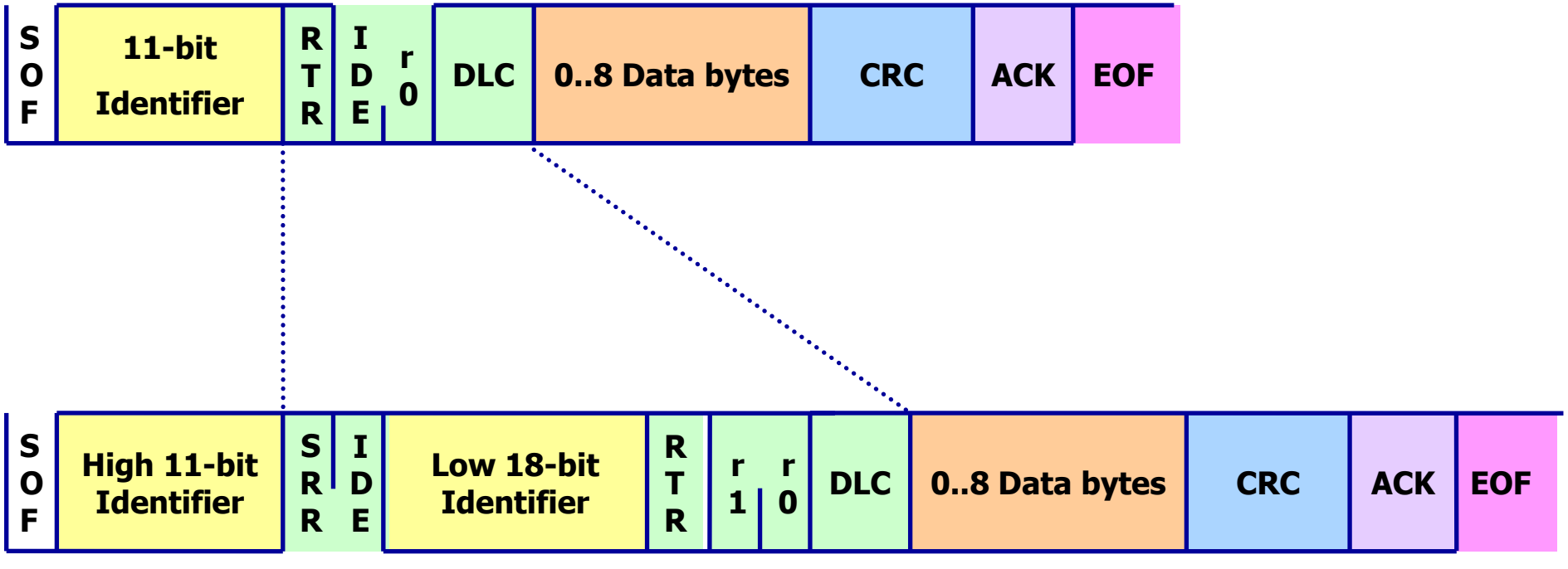
One more mailbox vs ST7

CAN Features (2/2)

- Time Triggered Communication option
  - Disable automatic retransmission mode
  - 16-bit free running timer
  - Time Stamp sent in last two data bytes
- Management
  - Maskable interrupts
  - Software-efficient mailbox mapping at a unique address space
  - 2 dedicated interrupt vectors: transmit/Error/Status change interrupt and FIFO interrupt

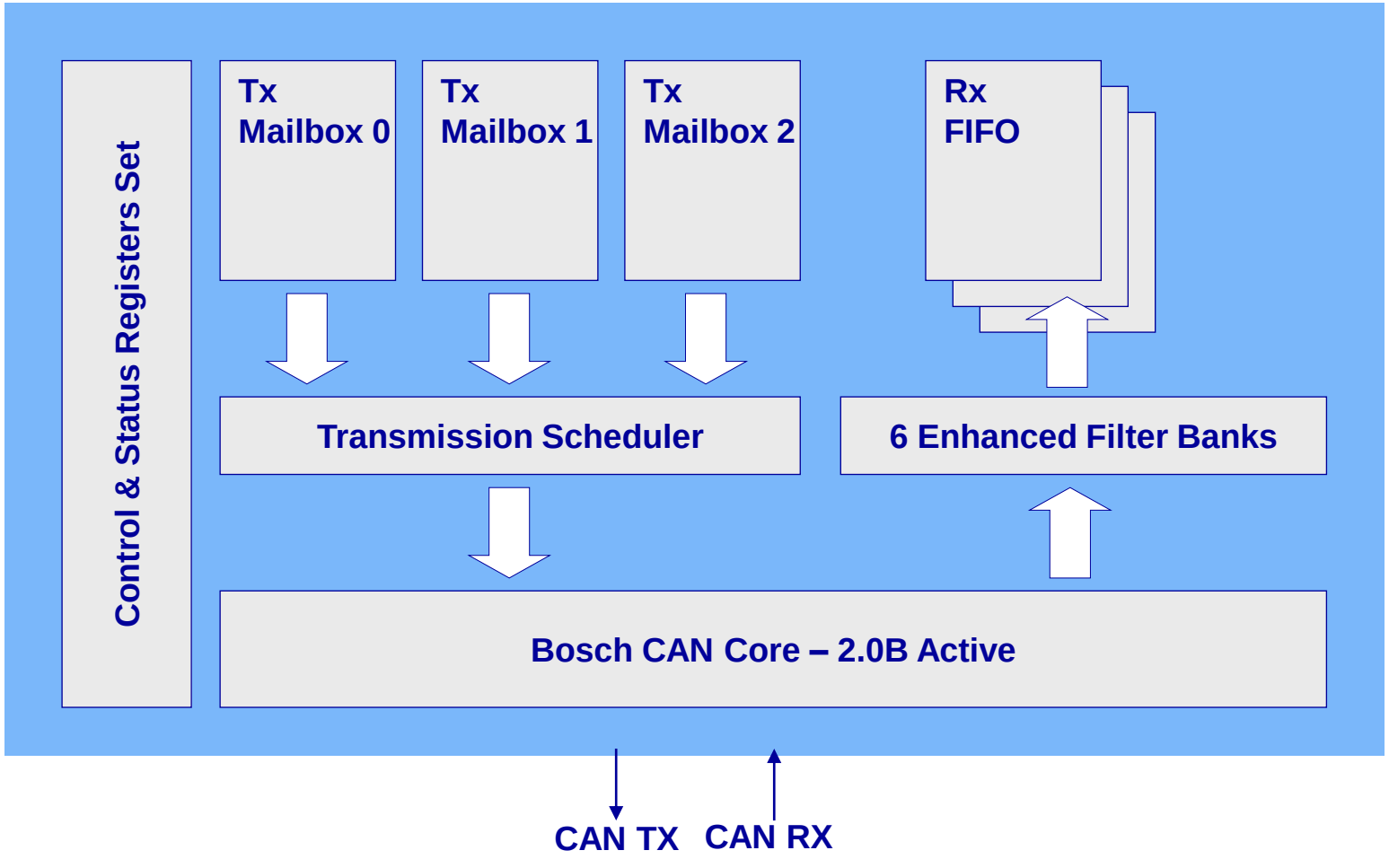
CAN data frames

- Standard data frame ☞ 11-bit identifier



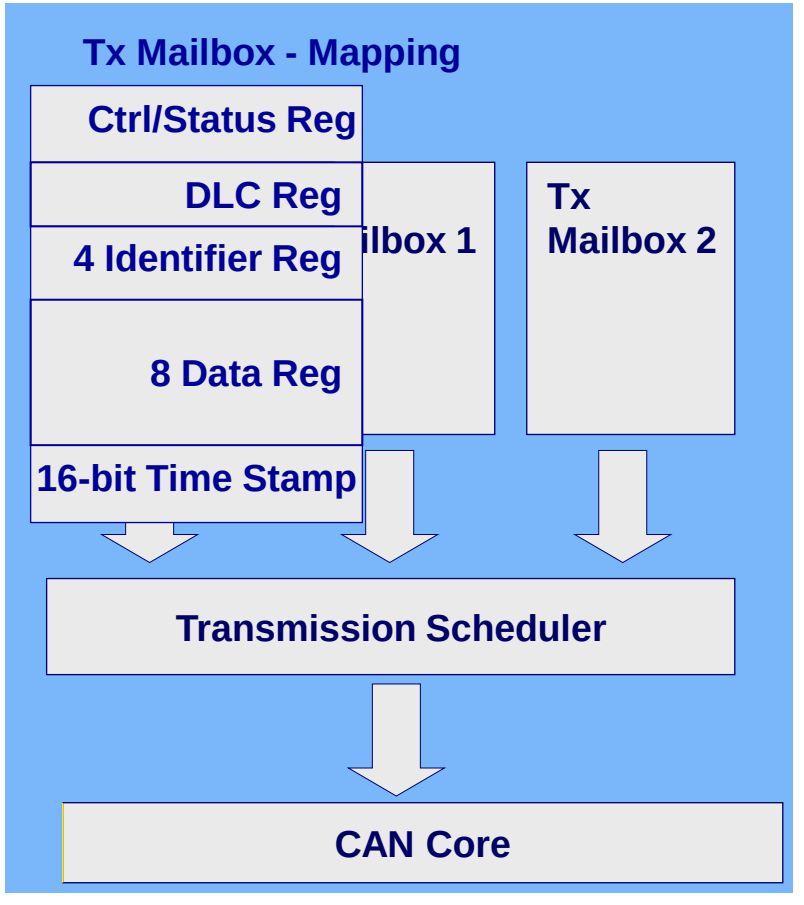
- Extended data frame ☞ 29-bit identifier

CAN block diagram



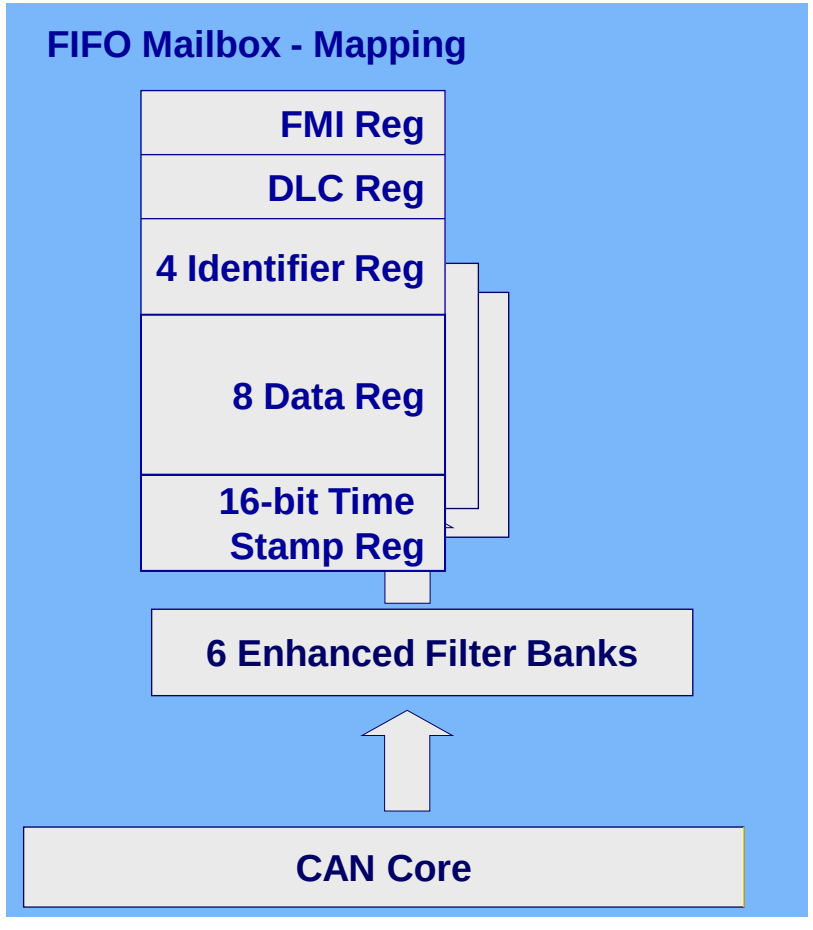
Transmission handling

- 3 Transmit Mailboxes allow to:
  - sidetrack of low priority messages by high priority messages
  - reduce software queuing
- Transmission priority by
  - Identifier
  - Transmit request order (FIFO)
- Abort Management
  - Transmit Priority Status Register
  - No retransmission mode



Reception handling

- FIFO
  - Reduce real-time constraint
- FIFO Features
  - Completely managed by hardware
  - Three messages deep
  - FMI stored with associated message
  - A message, Full and Overrun signaling



Quiz

- How many transmit mailboxes are in the STM8 beCAN?
- What are the differences between a standard and an extended identifier frame?
- How many CAN messages the STM8 can receive without any processing?

References

- RM0016 STM8S Reference Manual
- AN3281 STM8 I2C optimized examples
- AN2646 Smart Card Interface with STM8
- AN3055 STM8A LIN Package Getting Start Manual(NDA needed)

CONFIDENTIALITY OBLIGATIONS:

THIS DOCUMENT CONTAINS SENSITIVE INFORMATION.

IT IS CLASSIFIED “MICROCONTROLLERS, MEMORIES & SECURE MCUs (MMS) RESTRICTED AND ITS DISTRIBUTION IS SUBMITTED TO ST/MMS AUTHORIZATION

AT ALL TIMES YOU SHOULD COMPLY WITH THE FOLLOWING SECURITY RULES:

- DO NOT COPY OR REPRODUCE ALL OR PART OF THIS DOCUMENT
- KEEP THIS DOCUMENT LOCKED AWAY
- FURTHER COPIES CAN BE PROVIDED ON A “NEED TO KNOW BASIS”, PLEASE CONTACT YOUR LOCAL ST SALES OFFICE OR DOCUMENT WRITTER.