



LABORATORIO: RETI PER AUTOMAZIONE INDUSTRIALE STM32 NUCLEO

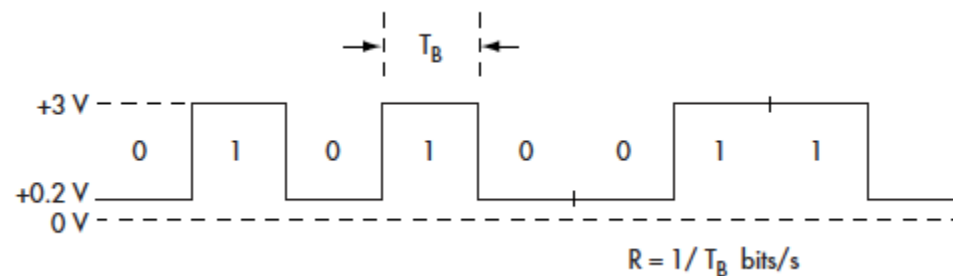
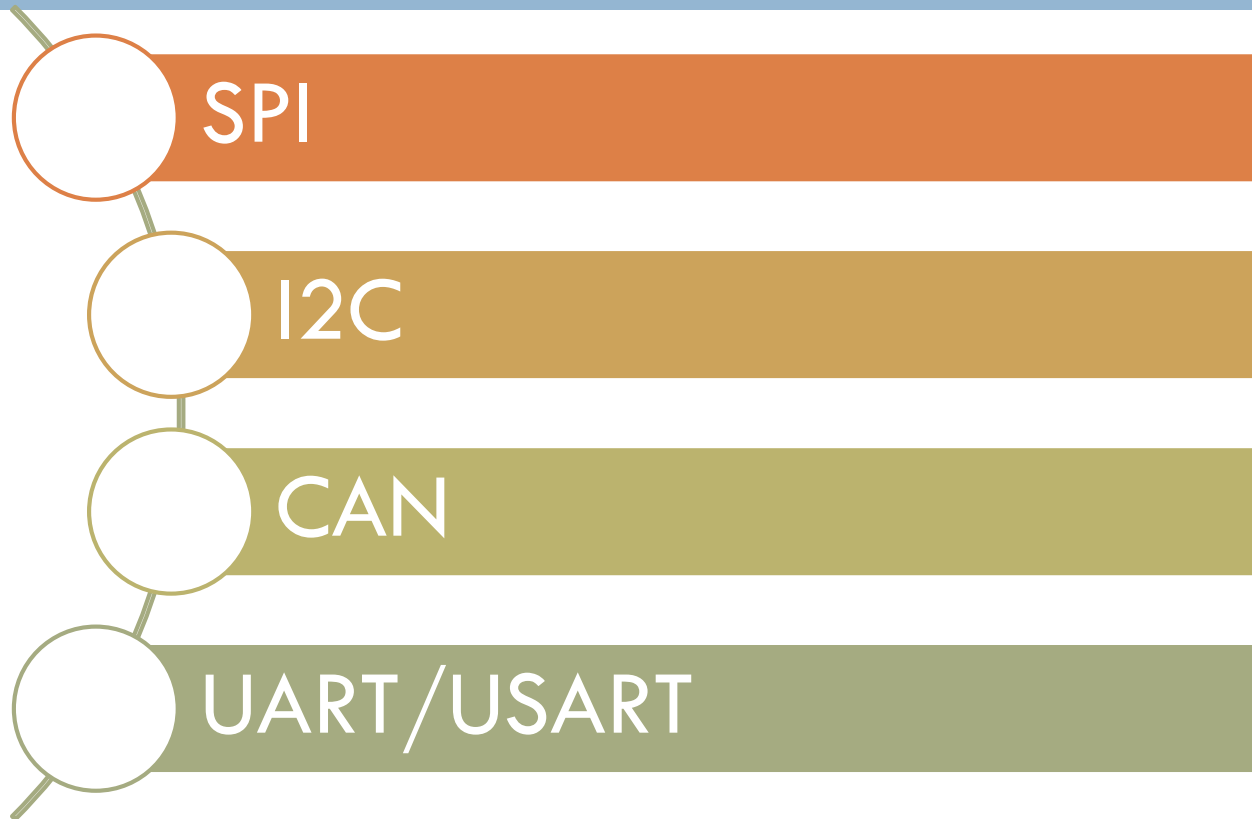
1

COMMUNICATION PERIPHERALS



Communication

2



SPI – Serial Peripheral Interface

3

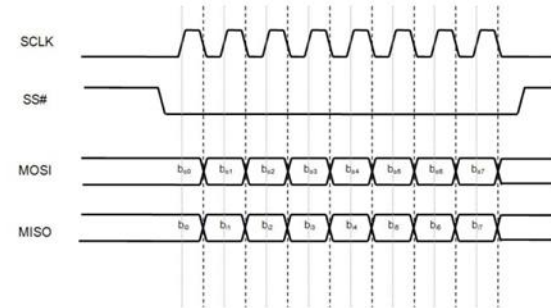
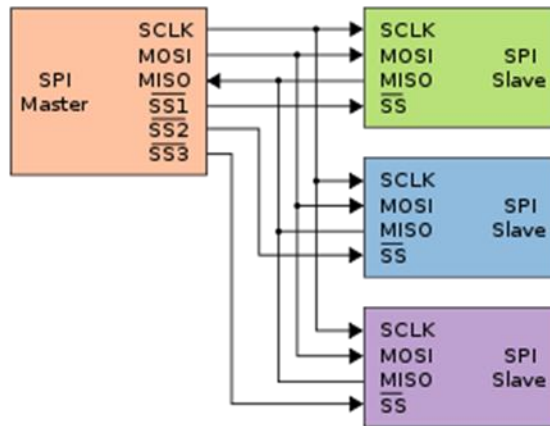
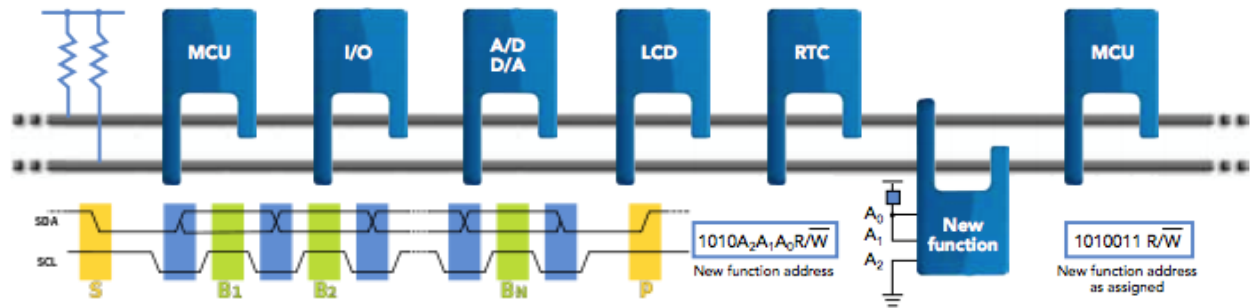
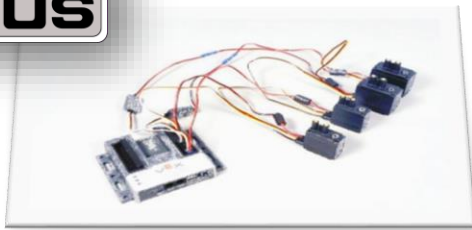


Figure 2 : A simple SPI communication. Data bits on MOSI and MISO toggle on the SCLK falling edge and are sampled on the SCLK rising edge. The SPI mode defines which SCLK edge is used for toggling data and which SCLK edge is used for sampling data.

I²C – Inter Integrated Circuit

10 o 100 kb/s

4



The I2C bus is used in a wide range of applications because it is simple and quick to use. It consists of a two-wire communication bus that supports bidirectional data transfer between a master and several slaves. The master or processor controls the bus – in particular, the serial clock (SCL) line.

Data is transferred between the master and slave through a serial data (SDA) line. This data can be transferred in four speeds or modes: standard (0 to 100 Kbps), fast (0 to 400 Kbps), fast-mode plus (0 to 1 Mbps) and high-speed (0 to 3.4 Mbps). The most common speeds are the standard and fast modes. See block diagram below for a generic system.

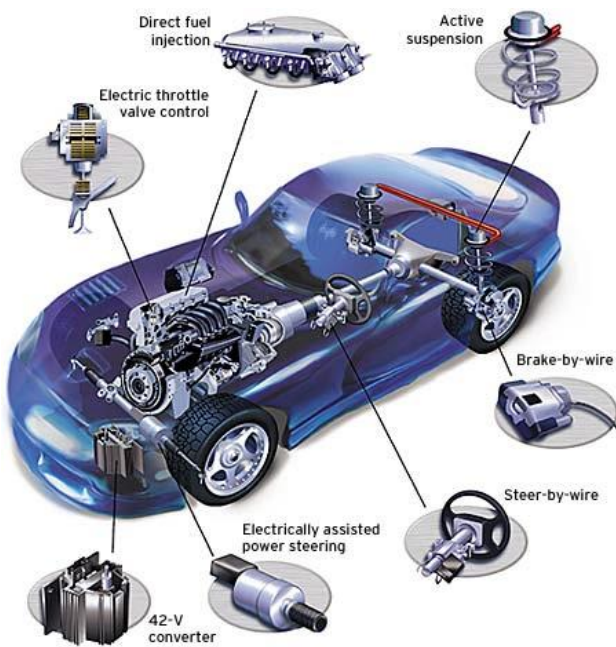
FEATURES

- **Requires one master (processor) and one or more slave devices**
- **Each device on the bus has a unique address**
- **Bus capacitive load: 400 pF max**
- **Rise time: 1000 ns (standard mode) and 300 ns (fast mode)**

CAN – Control Area Network

Up to 1 Mb/s

5



A controller area network (CAN) is ideally suited to the many high-level industrial protocols embracing **CAN** and **ISO-11898:2003** as their physical layer. Its cost, performance, and upgradeability provide for tremendous flexibility in system design

CAN
CUW

The CAN bus was developed by BOSCH(1) as a multi-master, message broadcast system that specifies a maximum signaling rate of 1 megabit per second (bps). Unlike a traditional network such as USB or Ethernet, CAN does not send large blocks of data point-to-point from node A to node B under the supervision of a central bus master. In a CAN network, many short messages like temperature or RPM are broadcast to the entire network, which provides for data consistency in every node of the system.

Once CAN basics such as message format, message identifiers, and bit-wise arbitration -- a major benefit of the CAN signaling scheme are explained, a CAN bus implementation is examined, typical waveforms presented, and transceiver features examined.

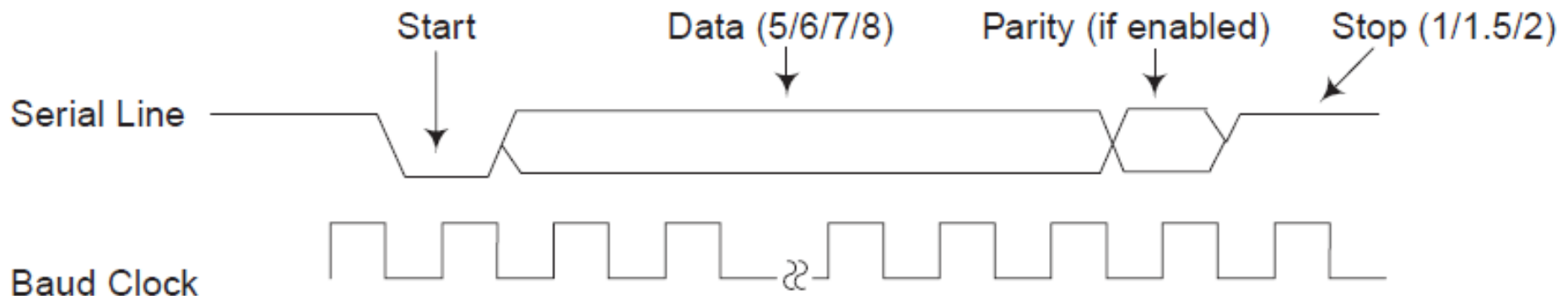
Identifier-The Standard CAN 11-bit identifier establishes the priority of the message. The lower the binary value, the higher its priority

S O F	11-bit Identifier	R T R	I D E	r0	DLC	0...8 Bytes Data	CRC	ACK	E O F	I F S
-------------	----------------------	-------------	-------------	----	-----	------------------	-----	-----	-------------	-------------

USART Serial Data Format

6

- Standard UART characters are contained within frames consisting of start/stop indicators, data, and parity information. The serial bitstream uses the following sequence:
 - Start bit
 - Data bits
 - Parity bit (if enabled)
 - Stop bit(s)
- When there is nothing to be sent, the serial line is high. The first low bit is the start bit, which indicates the beginning of a new frame. The next five to eight bits are data bits, which convey the actual information to be sent, least significant bit first. The number of data bits per frame is configurable.
- If enabled, a parity bit will be sent on the serial line after the data bits. Finally, the serial line is held high again for at least one bit to indicate the end of a frame. This is called the stop bit, which also returns the serial line to the idle state. The type of parity and length of stop bits are also configurable. Figure below shows the waveform of a frame.



USART Features

7

Through these pins, serial data is transmitted and received in normal USART mode as frames comprising:

- ❑ An Idle Line prior to transmission or reception
- ❑ A start bit
- ❑ A data word (8 or 9 bits) least significant bit first
- ❑ 1, 1.5, 2 Stop bits indicating that the frame is complete
- ❑ This interface uses a baud rate generator
- ❑ A status register (USART_ISR)
- ❑ Receive and transmit data registers (USART_RDR, USART_TDR)
- ❑ A baud rate register (USART_BRR)

USART Parameters

8

Baudrate

-
- 115200
- 57600
- 38400
- 19200
- 9600
- 4800
- 2400
- 1200
-

Datasize

- 8
- 9

Parity

- None
- Even
- Odd

Stop bit

- 1
- 1.5
- 2

8n1

USART Bridge USB

9

