



مرکز آموزش نیرا سیستم

nirasystem.com

Controller Area Network (CAN)

Ali Mirghasemi



Introduction

- Controller Area Network (CAN) Bus is a robust, multi-master, and message-oriented communication protocol widely used in industrial and automotive systems for real-time data exchange.
- The STM32F407 microcontroller includes an integrated CAN peripheral that supports this protocol, enabling efficient communication between microcontrollers and other CAN-enabled devices.
- CAN Bus excels in environments where high reliability, real-time communication, and fault tolerance are critical, making it a popular choice in embedded systems.
- ISO-11898 is the CAN standard, CAN2.0A – 11 bit identifier and CAN2.0B - 29 bit identifier.
- High speed CAN - 1Mbps(suspension , engine control) and Low speed CAN – 125kbps(power window, wiper control).



Applications

- **Automotive Systems**
 - Widely used for in-vehicle communication between Electronic Control Units (ECUs), sensors, and actuators (e.g., engine control, airbag systems, ABS).
- **Industrial Automation**
 - Supports communication in factory automation, process control, and robotic systems.
- **Medical Equipment**
 - Enables reliable communication in devices like infusion pumps, ventilators, and patient monitoring systems.
- **Aerospace**
 - Used in avionics systems for real-time data exchange between flight control systems and sensors.
- **Home Automation**
 - Allows smart appliances and sensors to communicate efficiently in building management systems.
- **IoT Systems**
 - Facilitates connectivity in industrial IoT applications where reliable, decentralized communication is necessary.

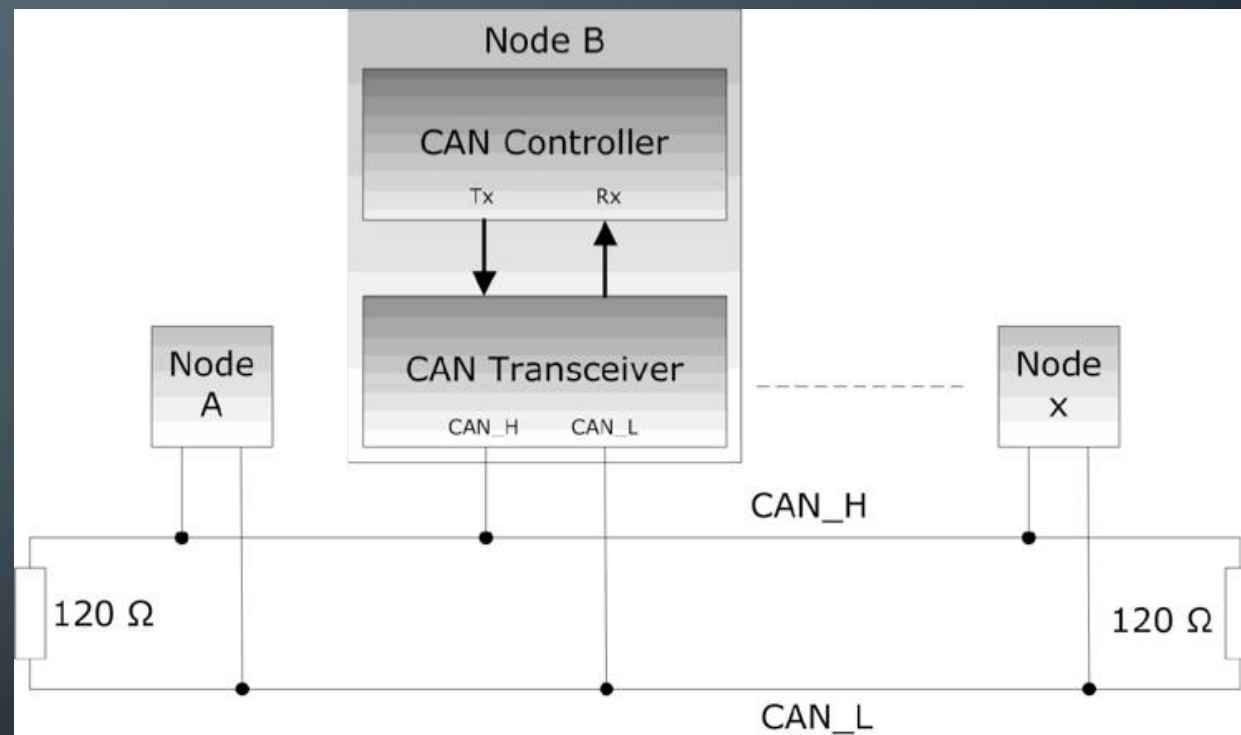


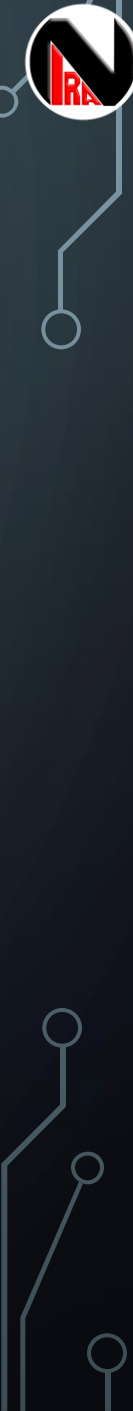
Features

- **Support for Standard and Extended IDs**
 - Handles both the 11-bit standard and 29-bit extended CAN identifiers, making it compatible with various CAN devices.
- **Bit Rates up to 1 Mbps**
 - Supports high-speed data transmission, essential for real-time communication in embedded systems.
- **Error Detection and Handling**
 - Includes mechanisms such as CRC checks, bit stuffing, and frame check to detect and recover from errors.
- **Message Filtering**
 - Allows filtering of incoming messages to reduce CPU load and focus on relevant data.
- **Multiple Mailboxes**
 - Provides dedicated transmit and receive mailboxes to handle multiple messages simultaneously, improving throughput.
- **Low Power Mode**
 - Supports low-power modes for energy-efficient operation in battery-powered systems.
- **Flexible Interrupts**
 - Offers a range of interrupt options for handling message transmission, reception, and errors.

Properties

- Connection Type: Serial
- Communication Type: Half-Duplex
- Data Type: Packet
- Synchronize: Async
- Channel Type: Copper Wire
- Voltage State: TTL (Differential)





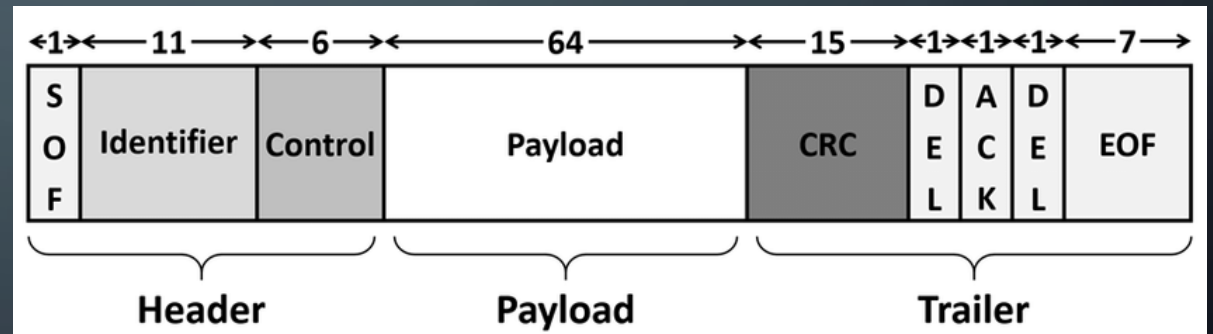
Frame

- **Data Frame**
 - Carries data between devices.
- **Remote Frame**
 - Requests data from another node.
- **Error Frame**
 - Signals an error in communication.
- **Overload Frame**
 - Introduces a delay when the bus is overloaded.



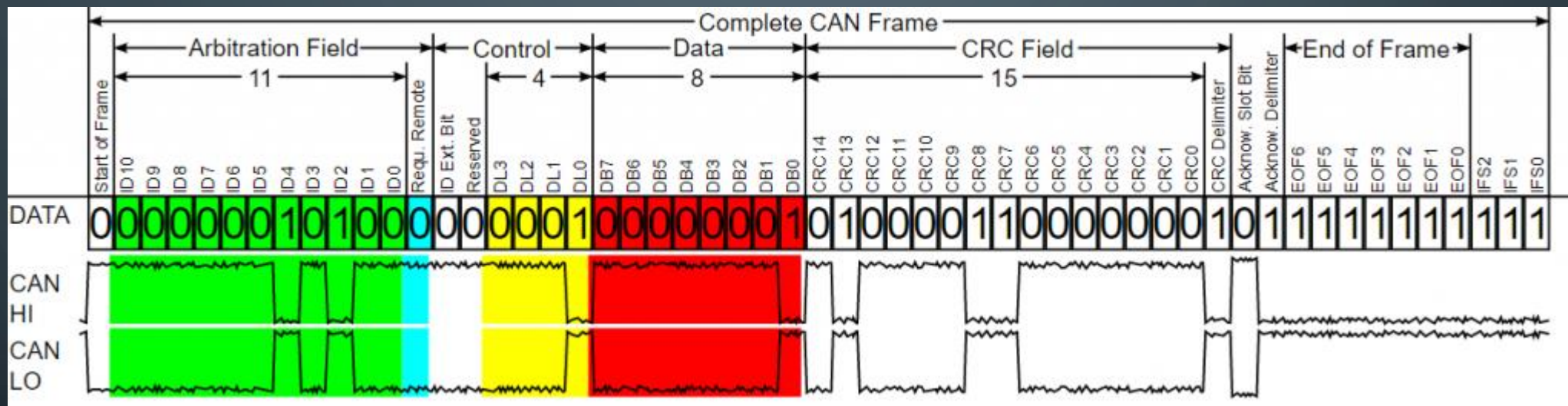
Frame - Fields

- **Start of Frame (SOF)**
 - Signals the beginning of a new frame.
- **Identifier**
 - Contains the message priority and the sender's address (11-bit for standard frames, 29-bit for extended frames).
- **Control Field**
 - Specifies the data length and type of frame.
- **Data Field**
 - Contains the actual payload, up to 8 bytes of data.
- **Cyclic Redundancy Check (CRC)**
 - Ensures data integrity by checking for errors during transmission.
- **Acknowledge (ACK) Field**
 - Confirms that the frame was received correctly by at least one receiver.
- **End of Frame (EOF)**
 - Indicates the end of the message frame.





Frame - Fields





CSMA/CD

- **Carrier Sense (CS)**

- Before transmitting data, a node checks if the bus is idle (i.e., no other device is transmitting).

- **Multiple Access (MA)**

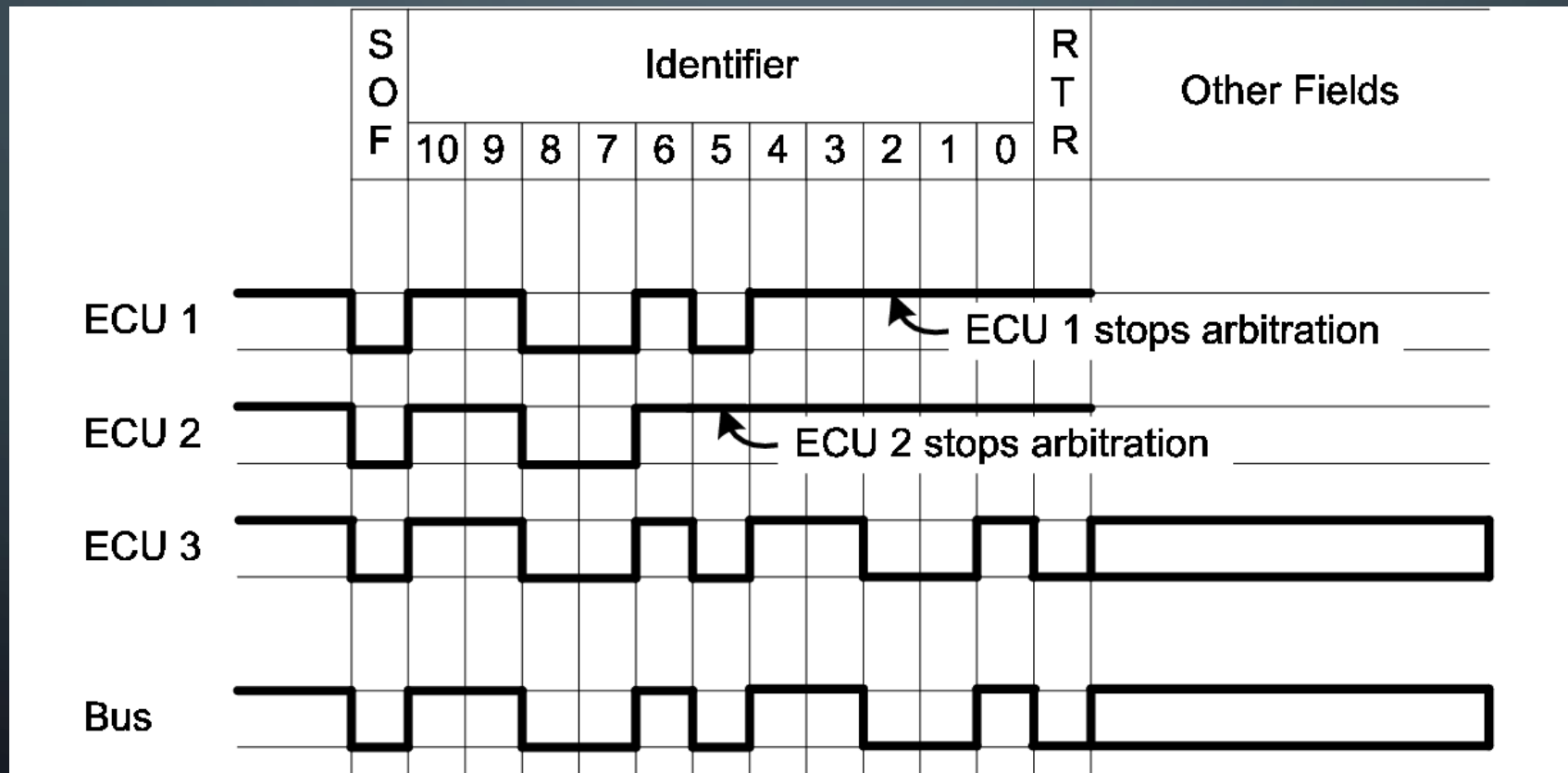
- All nodes on the network have equal access to the bus and can attempt to transmit data when the bus is idle.

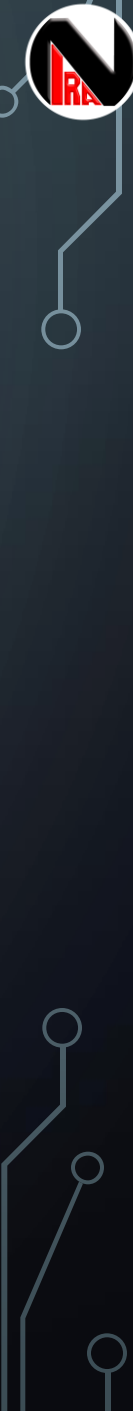
- **Collision Detection (CD)**

- If two nodes attempt to transmit at the same time, the node with the lower message identifier wins the arbitration process and continues to transmit, while the other node waits. This ensures that higher-priority messages are sent first.



Bitwise Arbitration





Data Protection - Bit Monitoring

- Sender Task
- Compares every bit placed on the CAN bus with the actual bus level
- Discrepancy indicates a bit monitoring error and results in error handling



Data Protection - Stuff Check

- Receiver Task
- Compares arriving bit stream for a sequence of six homogeneous bits.
- Detection of a sixth homogeneous bit indicates bit stuffing error and results in error handling



Data Protection - Form Check

- Receiver Task
- Comparison of the arriving bit stream with the message format
- Detection of a dominant delimiter bit (CRC delimiter, ACK delimiter) or a dominant bit within EOF indicates a format error and results in error handling



Data Protection - Cyclic Redundancy Check

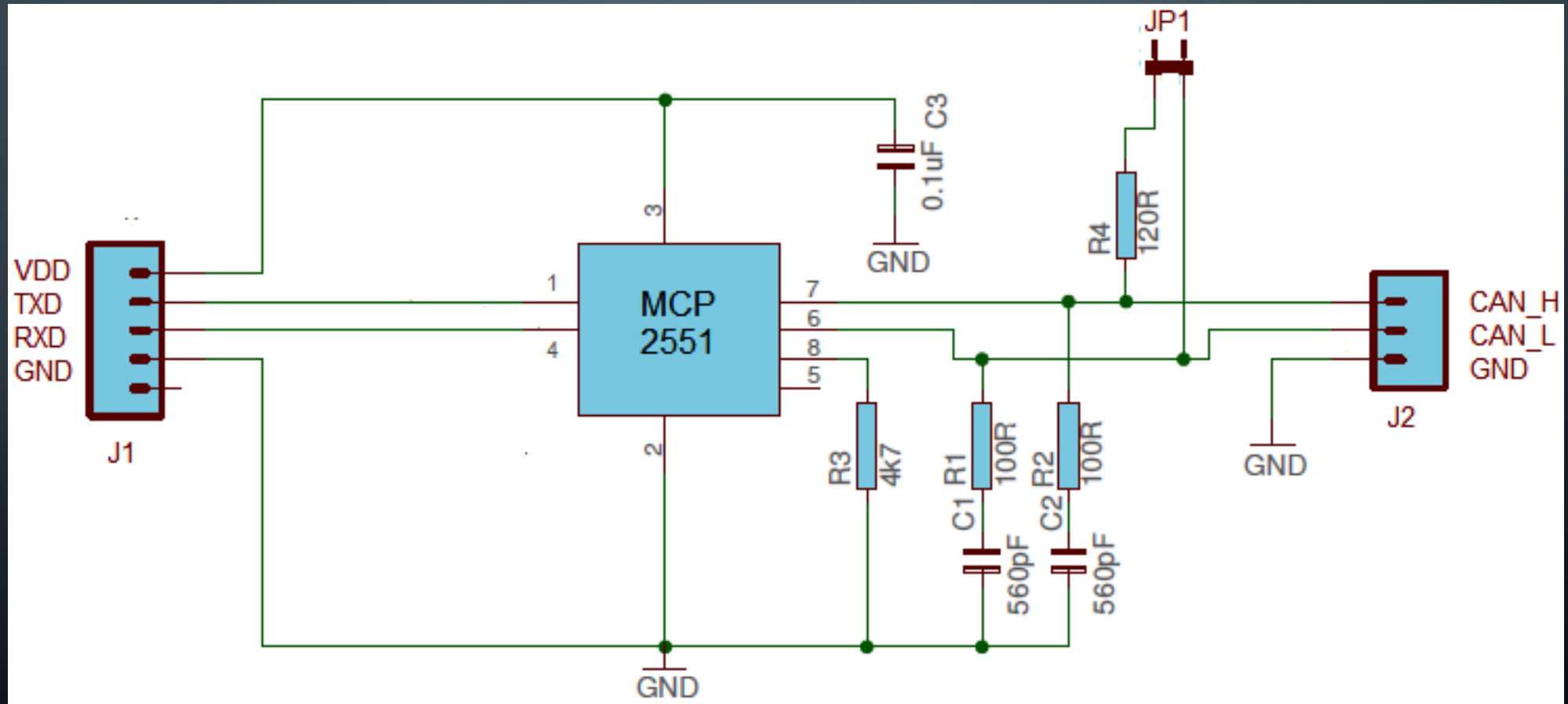
- Receiver Task
- Utilizes the arriving bit stream and generator polynomial for the Cyclic Redundancy Check defined in ISO 11898-1
- Detection of a CRC error results in error handling



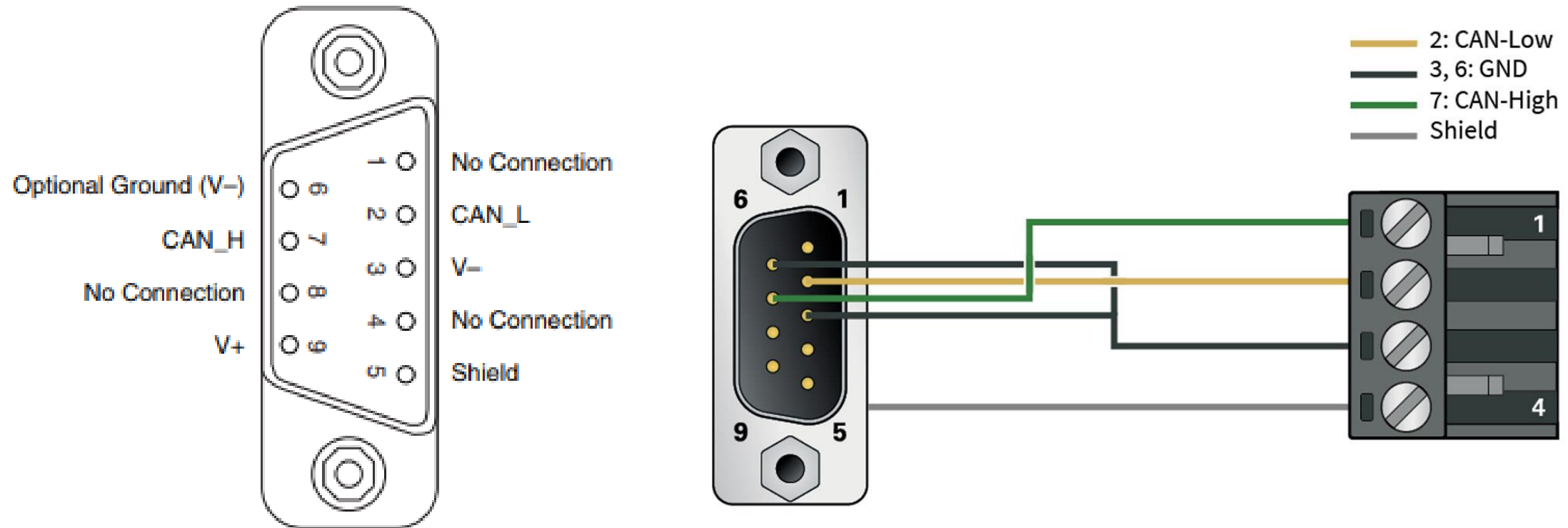
Data Protection - ACK Check

- Sender Task
- Acknowledge error (ACK error) is detected if the recessive level placed by the sender is not overwritten
- Detection of an ACK error results in error handling

Phy (MCP2551)



Connector





Cable Length

- Recommended to have maximum 30 nodes with 120 ohm termination
- According to the 'rule of thumb', $BitRate(Mbps) \times Length(Meter) \leq 50$
- Twisted pair cable

Bit Rate	Bus Length (Meters)	Bus Length (Feet)
1 Mbps	40	131
500 kbps	100	328
250 kbps	200	656
100 kbps	500	1640
50 kbps	1000	3280



Registers

- **CAN_MCR (Master Control Register)**
 - Configures the CAN peripheral, enabling or disabling it, managing initialization mode, and controlling sleep mode.
- **CAN_MSR (Master Status Register)**
 - Reflects the status of the CAN peripheral, including initialization and error status.
- **CAN_TSR (Transmit Status Register)**
 - Manages the status of the transmit mailboxes, providing information on transmission completion and errors.
- **CAN_RF0R and CAN_RF1R (Receive FIFO Registers)**
 - Handle the reception of messages, indicating the number of messages received and managing FIFO status.
- **CAN_IER (Interrupt Enable Register)**
 - Configures the interrupt sources for CAN, such as transmit, receive, error, and wake-up interrupts.