

NODE 2

```
graph TD; CANBUS[CAN-BUS] --- IOW[I/O-board]; IOW -- "TWI-bus 2-bit" --> Arduino; Arduino <--> "Data-bus 8-bit" --- IOW; Arduino -- "analog 1-ch" --> MotorBox[MOTOR BOX]; MotorBox -- "MJ1 8-bit" --> Arduino; MotorBox -- "MJ2 8-bit" --> Arduino; MotorBox -- "ENCODER 6-bit" --> Motor[MOTOR]; Motor -- "1 ch analog" --> IR[IR]; IR -- "TRIGGER" --> Solenoid[SOLENOID]; Solenoid -- "TRIGGER" --> PongBoard[PING PONG BOARD]; Arduino -- "BUZZER SIGNAL" --> Buzzer[BUZZER]; Arduino -- "servo" --> PongBoard; PongBoard -- "1 ch" --> IR;
```

The diagram illustrates the hardware architecture of Node 2, centered around an Arduino microcontroller. The Arduino is connected to an I/O-board via a TWI-bus (2-bit) and a Data-bus (8-bit). The I/O-board is also connected to a CAN-BUS. The Arduino controls a MOTOR BOX via an analog signal (1-ch) and receives feedback from the MOTOR BOX through two 8-bit channels (MJ1, MJ2) and a 6-bit ENCODER. The MOTOR BOX is connected to a MOTOR, which provides an analog signal (1 ch) to an IR sensor. The IR sensor sends a TRIGGER signal to a SOLENOID, which in turn sends a TRIGGER signal to the PING PONG BOARD. The Arduino also controls a BUZZER via a BUZZER SIGNAL and a servo motor via a servo signal, both connected to the PING PONG BOARD. The PING PONG BOARD also provides an analog signal (1 ch) to the IR sensor.

The diagram illustrates the hardware architecture of NODE 1, centered around the ATmega162 microcontroller. The microcontroller is connected to several peripheral components:

- CAN-BUS:** Connected to the MCP2551 CAN-transceiver and the MCP2515 CAN-controller via RX/TX lines.
- DVI-POR:** Connected to the MAX233 component via a COM line.
- MAX233:** Connected to the ATmega162 via an in/out line.
- MCP2515:** Connected to the ATmega162 via an SPI-bus (4-bit).
- ADC:** Connected to the ATmega162 via a Data-bus (8-bit) and a CS line.
- GAL:** Connected to the ATmega162 via a CS line and a Data-bus (8-bit).
- D-Latch:** Connected to the ATmega162 via a Data-bus (8-bit) and an Adr-bus (LSB, 8-bit).
- SRAM:** Connected to the ATmega162 via a Data-bus (8-bit) and an Adr-bus (MSB, 4-bit).
- USB MULTIFUNCTION CARD:** Connected to the ATmega162 via an analog line (4-ch), a CS line, and an OLED-CMD line.

The ATmega162 microcontroller is the central processing unit, managing data flow between these components. The diagram uses color-coded boxes to represent different functional blocks: red for CAN-related components, green for the microcontroller, cyan for the ADC, blue for the GAL, pink for the D-Latch and SRAM, and white for the DVI-POR and USB MULTIFUNCTION CARD.