CAN COMMUNICATION AND MOTOR CONTROL USING STM32F407/F103 MICROCONTROLLERS

Personal Embedded Systems Project by Bui Phuoc Vinh

I. INTRODUCTION & OBJECTIVE

Controller Area Network (CAN) is widely used in automotive industry for reliable multi-node communication. This project builds a multi-node CAN network for real-time motor control and monitoring with STM32 MCUs.

Technical requirements:

- Robust communication via CAN bus at 500 kbps.
- Deterministic response for motor speed/gas pedal changes (< 50 ms).
- Accurate feedback acquisition: encoder (RPM) and current (A).

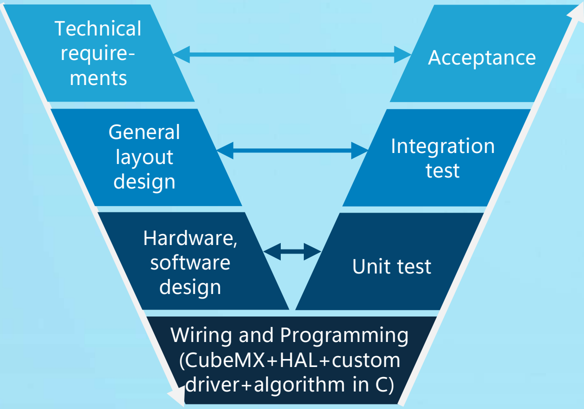

Technical require-ments
Acceptance
General layout design
Integration test
Hardware, software design
Unit test
Wiring and Programming (CubeMX+HAL+custom driver+algorithm in C)

Fig. 1. Project V-model

II. SYSTEM DESIGN & IMPLEMENTATION

Architecture: three nodes connected via CAN bus with 120Ω termination.

- **STM32F407** (Main Controller): motor control, encoder & current sensing.
- **STM32F103** (Input Node): gear selection via push buttons, gear status LEDs, gas pedal posittion via potentiometer.
- **STM32F103** (Display Node): status monitoring on LCD1602 via PCF8574.

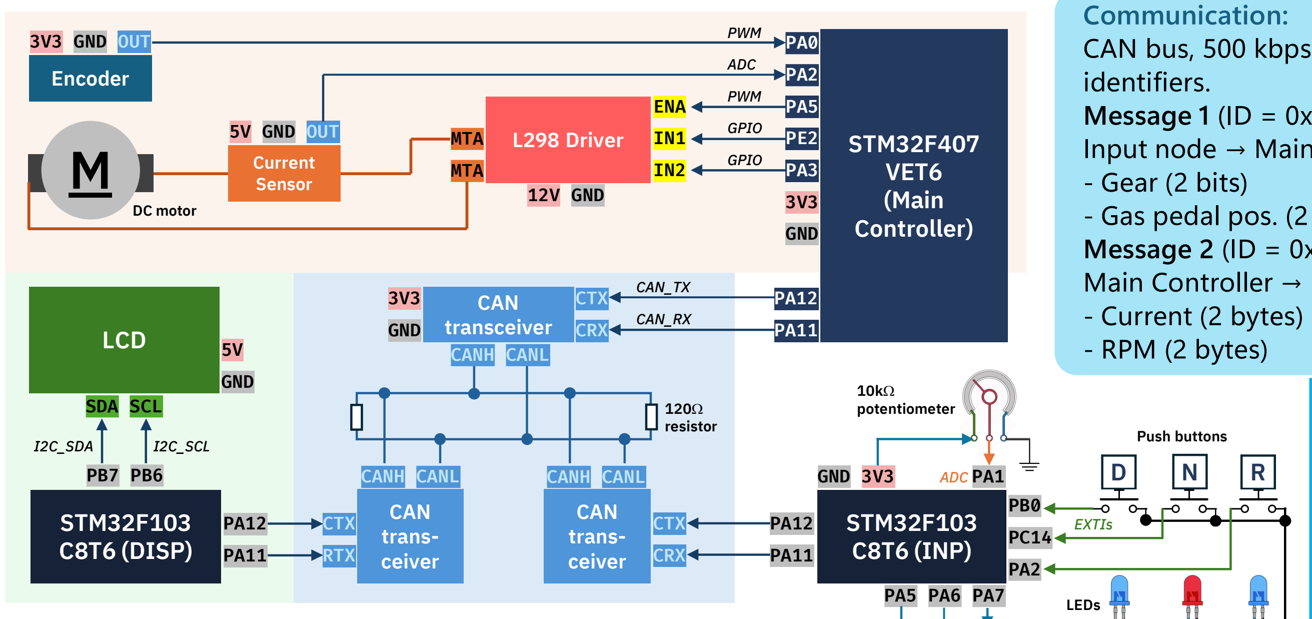


Fig. 2. Hardware Block Diagram and CAN Bus Topology

Display <- CAN <- Feedback <- Motor <- Controller <- CAN <- Input

Communication:

CAN bus, 500 kbps, standard 11-bit identifiers.

Message 1 (ID = 0x123, DLC = 2)

Input node → Main Controller:

- Gear (2 bits)
- Gas pedal pos. (2 byte)

Message 2 (ID = 0x125, DLC = 4)

Main Controller → Display node:

- Current (2 bytes)
- RPM (2 bytes)

Hardware detail:

- CAN transceiver (SN65HVD230)
- DC motor 12V
- Encoder TTL
- Current sensor (ACS712 5A)
- LCD16x02 + PCF8574 module

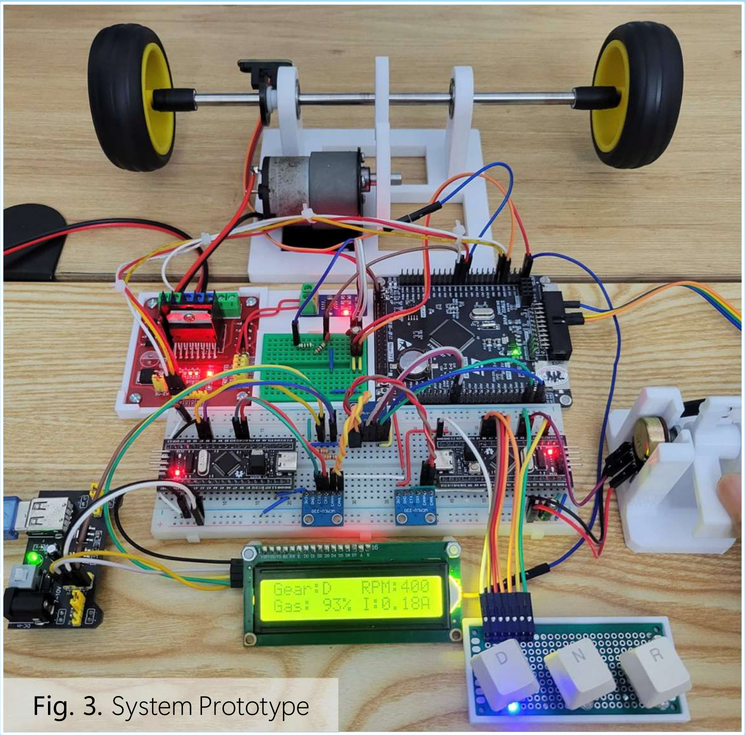
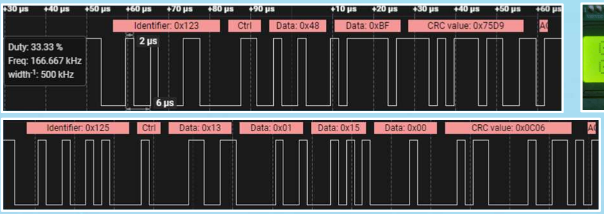


Fig. 3. System Prototype

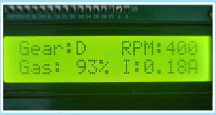
III. RESULTS

- CAN bus communication successfully implemented at 500 kbps.
- Motor responds correctly to gear selection (D/N/R) and gas pedal input (potentiometer).
- LCD updates in real time with Gear, Gas, RPM, and Current.



Duty: 33.33 %
Freq: 166.667 kHz
width⁻¹: 500 kHz

Fig. 4. CAN frames captured by Logic Analyzer



Gear:D RPM:400
Gas: 93% I:0.18A

Fig. 5. Real-time Data Display on LCD

IV. CONCLUSION

- Implemented a distributed CAN-based system using STM32.
- Achieved real-time motor control with reliable feedback monitoring.
- Demonstrated successful communication across MCUs.

V. FUTURE WORKS

- Integrate RTOS for better task scheduling and scalability.
- Implement regenerative braking for energy recovery.