



* Learn more about our Client: MicroChip

Microchip, which is known for the innovative company and has the provision of embedded - and automobile - control systems requests us to implement the specific firmware on dsPIC33CSC. This inspires us to acknowledge the importance of efficient and reliable communications.

Microchip Technology provides an Integrated Development Environment (IDE) known as MPLAB-X, offering powerful features such as visualized I/O operations, code debugging, and validation of embedded designs for microcontrollers and digital signal controllers. This tool allows for the implementation and testing of CAN communication codes compatible with the dsPIC33C family

* Learn more about CAN -
  + Controller Area Network (CAN) is a data communication protocol utilized for broadcasting data and managing control systems.
  + The CAN protocol is a cost-effective communication option with real-time capabilities suitable for complex electrical environments. Various boards support the CAN protocol.
  + CAN is versatile and finds applications in a wide range of fields. While initially designed for interconnecting automotive devices, it has expanded into various industries. Crucial use-cases include agricultural equipment, navigational equipment, mechanical control, aircraft fueling systems, medical instruments, maritime applications such as ships, lighting control systems, and robotics.

For example, medical equipment manufacturers have integrated CAN into medical devices, and hospitals utilize CAN-based systems to control room components like lights, cameras, and X-ray machines.id

* More information about our group and our project:

Team members :

* Andrew DiFranco
* Bryan Marquez
* SeungJun Ryu
* Yeongwoong Choi
* Zhi Yong Li

Our Goals:

The main goal of this project is to gain a basic understanding of control area network (CAN) protocols and embedded programming via implementation onto dsPIC33CK1024MP710 Family microcontrollers. With hands-on experimentation and observation of the dsPIC33EV 5V CAN-LIN STARTER KIT capabilities, preparation and eventual implementation for CAN protocol onto the dsPIC33C Touch-CAN-LIN curiosity development board can be made.

* Implementation:
  + CAN-FD protocols on curiosity
  + CAN 2.0 protocols on curiosity
  + Sending and receiving capabilities for various message types between curiosity and starter boards
  + Collision detection and handling of arbitration
  + Can analyzer
  + Connection between boards to simulate a CAN bus network
  + Website database showcase
  1. Design

Firmware will include functionality that allows it to determine message arbitration in the event of a collision. The peripherals on both boards will be included to allow user interaction in some way. The current idea is to send message containing certain identifiers onto the bus when the user interacts with them. Each board will be able to be removed and added onto the bus at any time. When one board is disconnected, the other board should continue communication with the bus analyzer without interruption. When a board is added onto the bus, the board that was already there should pick it up in communication. Each board will have a unique set of masks and filters which specifies which messages they should accept and which message they should ignore.

1. How to Use

If you click "Menu" and "Start Now", you may see transmitting and receiving logs. Those consist of timestamps, communication type - TX, RX -, ID, and some data derived from the communication. Easy User Manual will be available soon from our site as a PDF file for better understanding.