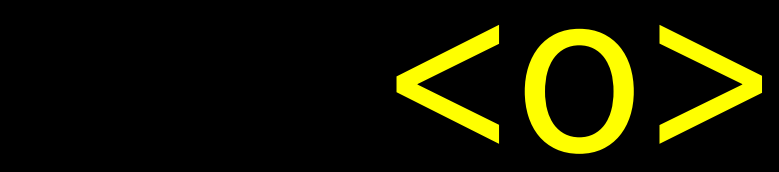
Controller Area Network (CAN bus)

Team Zero

Daniel Caballero (Reviewer, Presenter)
Eric Hansen (Researcher, Reviewer, Presenter)
William McKinnon (Reviewer, Presenter)
Richard Hemphill (Demo, Content, Figures, Presenter)
Joel Zamora (Reviewer, Presenter)



Course: ECE5595
Term: Fall 2019
Project Presentation
Submitted:12/3/2019

Executive Summary

Controller Area Network is prevalent in many industries, especially automotive. In order to educate the class on this network architecture, some of the fundamentals are presented and then a demonstration of sending various types of messages over the bus is shown. Finally, an analysis that maps the logic values to packet field value is performed.

Controller Area Network Fundamentals

History

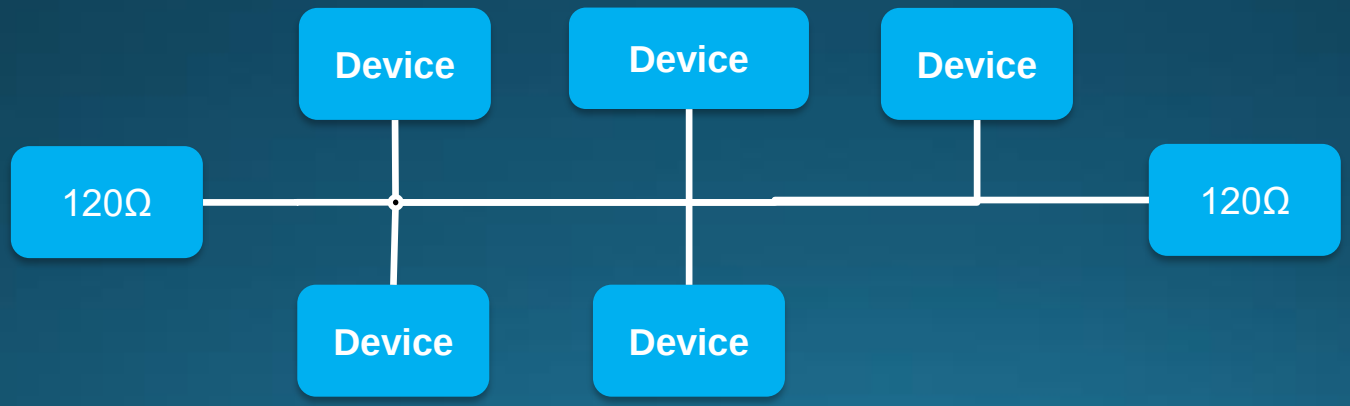
- Origin
  - Development started Bosch Global in 1983
  - Protocol released by SAE (Society of Automotive Engineers) in 1986
  - Standardized by ISO (International Organization for Standardization) in 1993
- What problem did it solve?
  - Wiring-reduction

Standards

- ISO 11898: Road vehicles — Controller area network (CAN)
  - Part 1: Data link layer and physical signalling
  - Part 2: High-speed medium access unit
  - Part 3: Part 1: Low-speed, fault-tolerant, medium-dependent interface
  - Part 4: Time-triggered communication
  - Part 5: High-speed medium access unit with low-power mode
  - Part 6: High-speed medium access unit with selective wake-up functionality
- SAE J1939: Recommended Practice for a Serial Control and Communications Vehicle Network

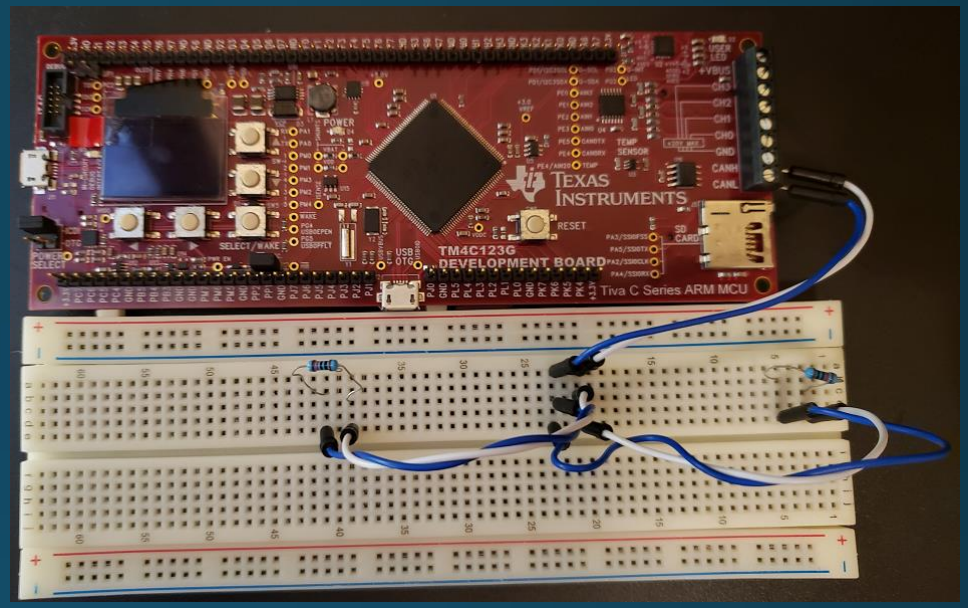
Topology

- Controller Area Network uses a bus network topology
  - All nodes connect to High/Low of serial bus
  - 120Ω termination at each end to dampen signal refection



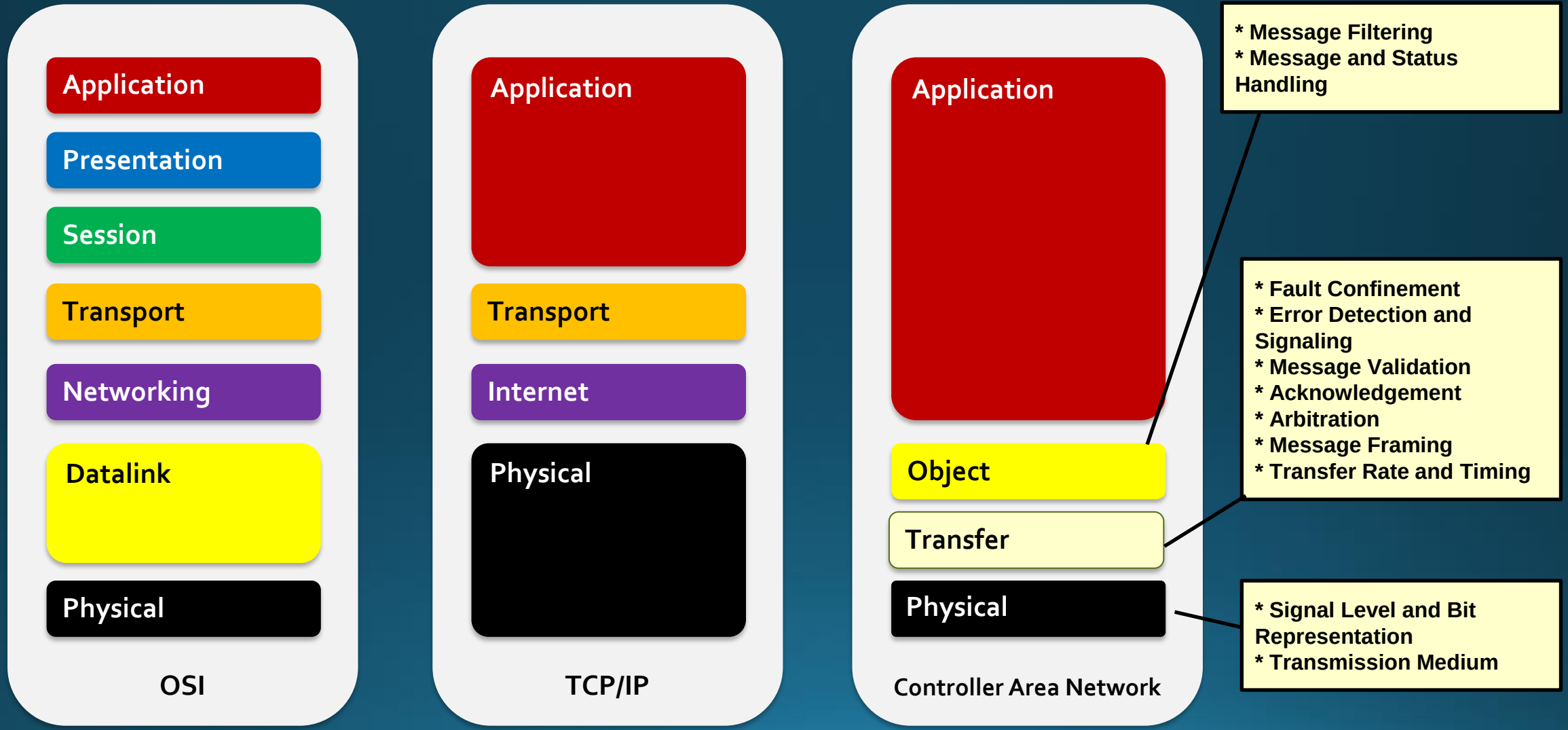
Physical Layer

- Twisted-Pair Cable
  - 120 Ω termination
- Bit-Wise Arbitration
  - Logic 0 ($V_{diff} > 2.3$) is dominant
  - Logic 1 ($V_{diff} < 0.6$) is recessive
  - Bits are ANDed
  - Lower binary value for identifier, higher priority



Device	ID (Binary)	ID (Dec)	Priority
1	0b01100110010	818	Medium
2	0b01100010000	748	Highest
3	0b11111111111	2047	Lowest
CAN Bus	0b01100001000	748	

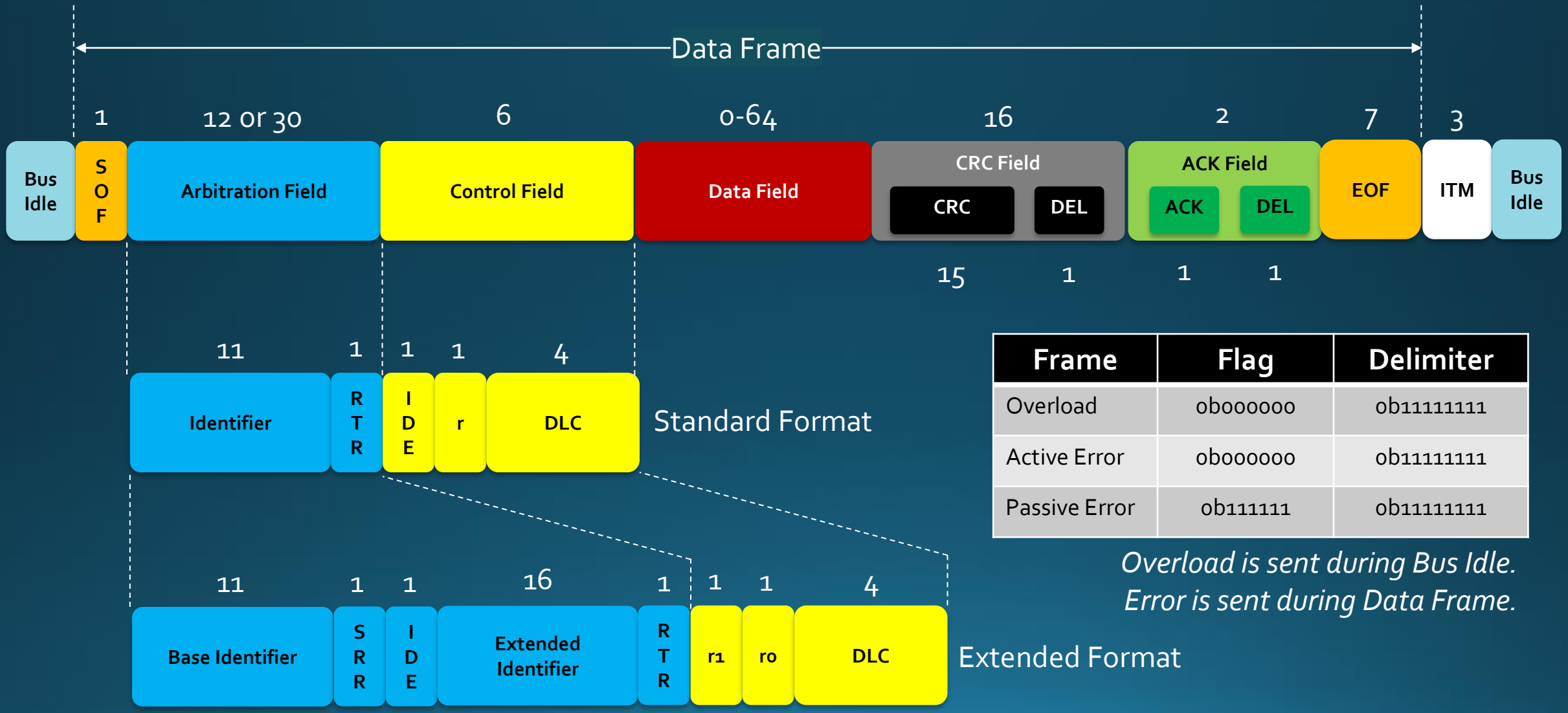
Protocol Stack



Distributed Application Structure

- Peer-to-peer
- Any node can be bus master
  - Message with lowest ID value wins
- All messages are broadcast
- Message oriented transfer
  - Transfer is in chunks (i.e. packets)
  - No continuous flow of data by default

Packet Format



Frame	Flag	Delimiter
Overload	ob0000000	ob11111111
Active Error	ob0000000	ob11111111
Passive Error	ob111111	ob11111111

Overload is sent during Bus Idle.
Error is sent during Data Frame.

CAN Chat

Demonstration of CAN Bus Messages

CAN Chat: Concept of Operation

- Team Zero member identifies themselves
- Zero member select destination members for chat
- Any 8-character message sent gets transmitted to selected destination members
- Upon receipt of message, the message and who sent it is shown to the member
- Upon button press, a command is sent to activate an LED connected to another subsystem (e.g. Arduino Uno or Mega)
- Upon button release, a command is sent to deactivate an LED connected to another subsystem.
- The primary subsystem (i.e. Arduino Mega) logs CAN message fields (i.e. ID, length, and data) over the bus
- An oscilloscope monitors the voltages on the CAN bus Hi/Lo lines

CAN Chat: Laboratory Setup

Arduino Serial Monitor

Push Button

LED

16x2 LCD

Computer

USB

Arduino Mega

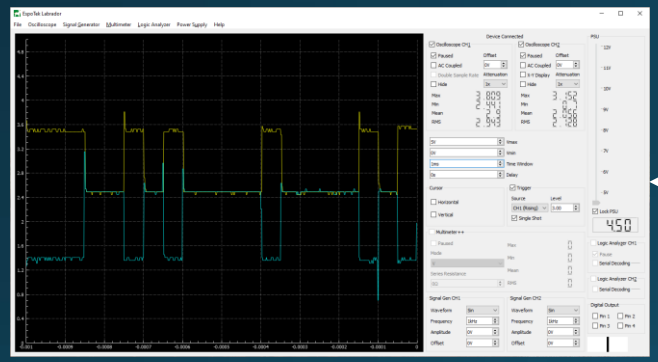
CAN Bus Transceiver

H

L

120 Ω Resistor

CAN Bus

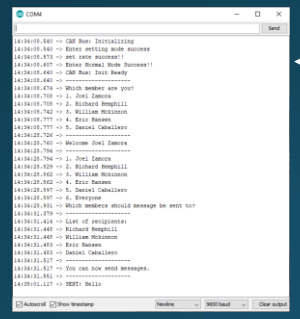


Espotek Labrador

Computer

USB

Oscilloscope



Arduino Serial Monitor

Computer

USB

Arduino Uno

CAN Bus Transceiver

Push Button

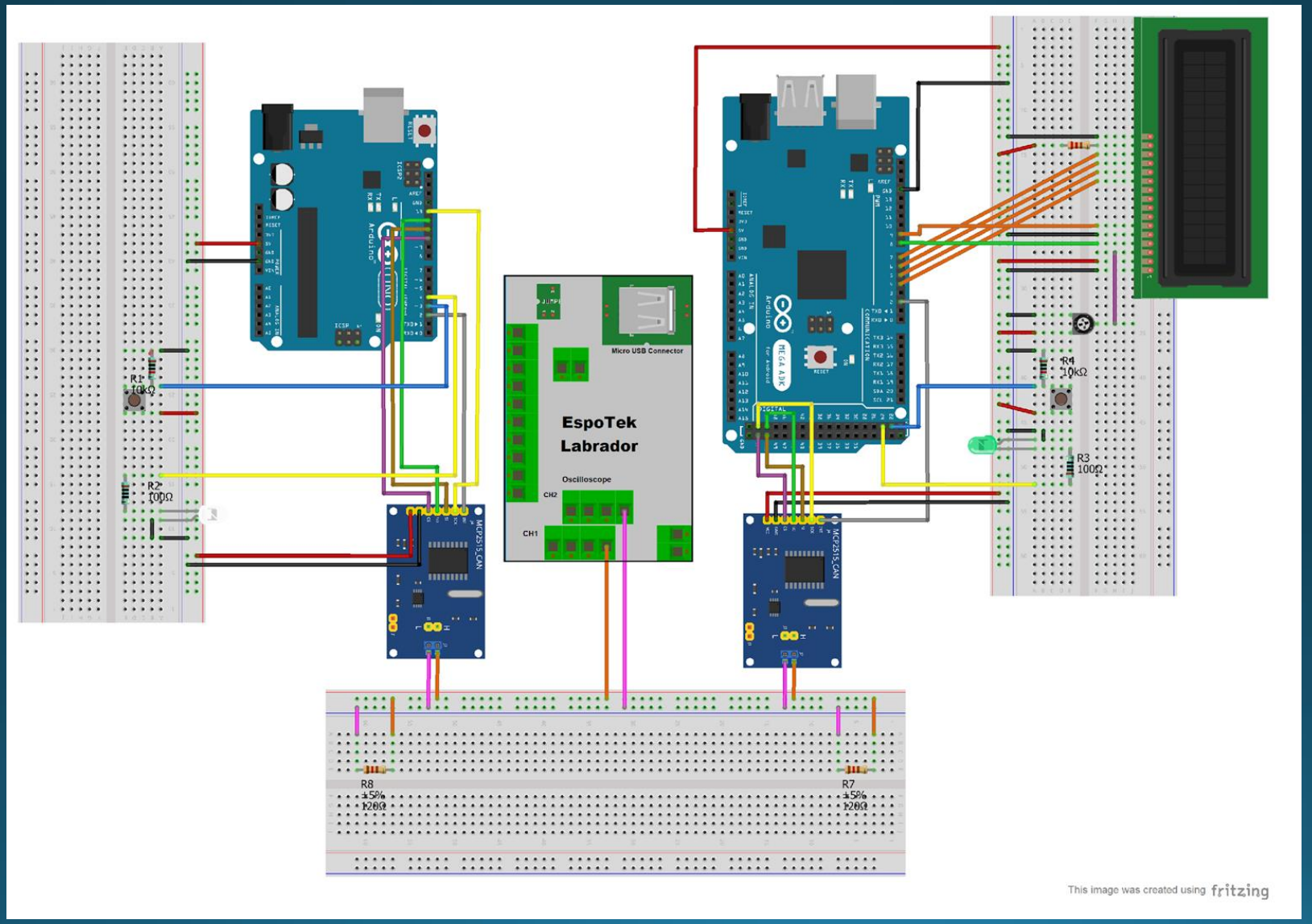
LED

H

L

120 Ω Resistor

CAN Chat: Wiring Diagram



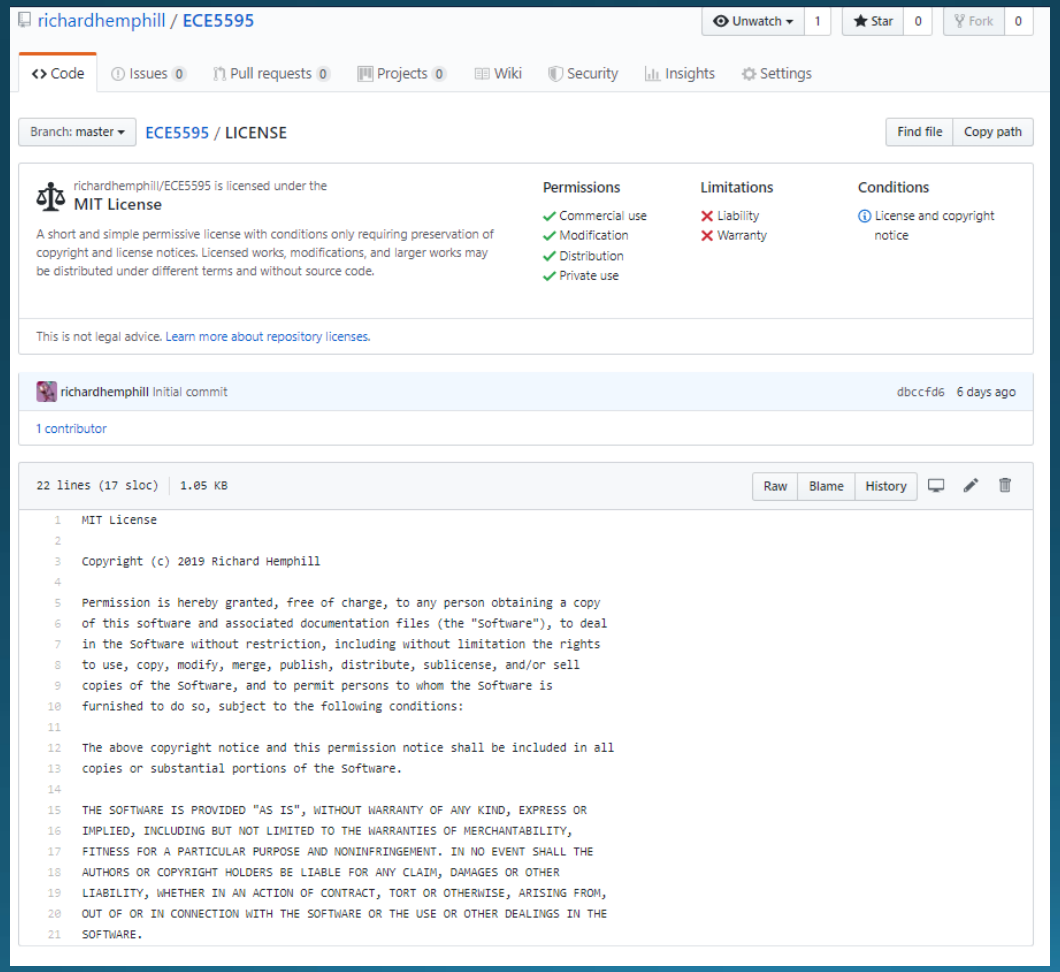
CAN Chat: Bill of Materials

#	Description	Unit	Qty	Cost	Total
1	Laptop	EA	3	479.99	1439.97
2	Breadboard	EA	3	7.95	23.85
3	A-Male to B-Male USB Cord	EA	3	4.98	14.94
4	Arduino Uno	EA	1	6.99	6.99
5	Arduino Mega	EA	1	13.99	13.99
6	EspoTek Labrador (Oscilloscope)	EA	1	29.00	29.00
7	XCSOURCE MPC2515 (CAN Bus Transceiver)	EA	3	3.33	9.99
8	16x2 LCD	EA	1	5.99	5.99
9	Push Button	EA	2	0.36	0.72
10	LED	EA	2	0.06	0.12
11	10kΩ Potentiometer	EA	1	0.77	0.77
12	10kΩ Resistors	EA	2	0.06	0.12

#	Description	Unit	Qty	Cost	Total
13	100Ω Resistors	EA	4	0.06	0.24
14	10Ω Resistors	EA	4	0.06	0.24
15	Male Jumper Wires	EA	52	0.11	5.72
16	Male to Female Jumper Wires	EA	2	0.10	0.20
		Grand Total			1552.85

CAN Chat: Source Code

- https://github.com/richardhemphill/ECE5595/tree/master/canChat



richardhemphill / ECE5595

Unwatch 1 | Star 0 | Fork 0

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Security | Insights | Settings

Branch: master ▾ ECE5595 / LICENSE

Find file | Copy path

richardhemphill/ECE5595 is licensed under the **MIT License**

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions
- ✓ Commercial use
- ✓ Modification
- ✓ Distribution
- ✓ Private use

Limitations
- ✗ Liability
- ✗ Warranty

Conditions
- ⓘ License and copyright notice

This is not legal advice. Learn more about repository licenses.

richardhemphill Initial commit dbccfd6 6 days ago

1 contributor

22 lines (17 sloc) | 1.05 KB | Raw | Blame | History

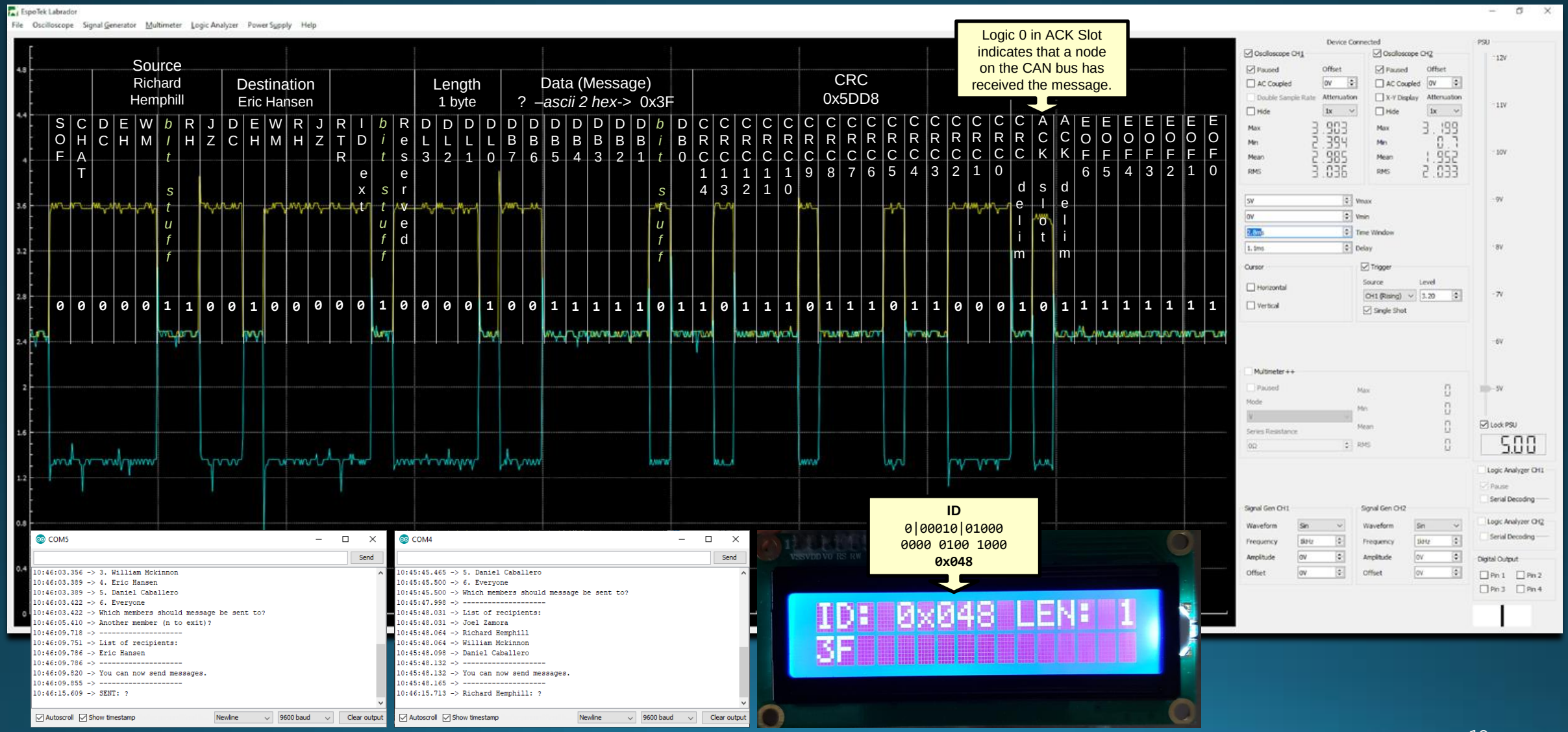
```
MIT License

Copyright (c) 2019 Richard Hemphill

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

CAN Chat: Test Procedure

1. Connect both Arduinos and Oscilloscope to computer
2. Start Arduino IDE
    1. Start Arduino Serial Monitor for each Arduino
    2. Log in as self
    3. Select Zero members to send messages to
3. Start EspoTek Labrador
    1. Adjust source trigger and delay to capture message
4. Type message in serial monitor and send
    1. Message should display on recipients' serial monitor
    2. LCD should show ID, length, and message in hex
5. Press button for one board
    1. LED should light up for other board
    2. LCD should show ID (0x400), length (2), and message in hex (0x0101=on & 0x0100=off)

Demonstration

CAN Chat: Signal Analysis



CAN Chat: Problems Encountered

- Every time the message identification field changed, the first transmission thereafter would use the old identification value
  - Suspect that this the root cause is the MPC2515 (CAN Bus Shield)
  - Workaround is to send dummy message every ID change
- At first glance of the oscilloscope results, thought that there was results were wrongs. Upon further analysis, determine that it was due to bit stuffing.
  - CAN Bus Bit Stuffing - Bit of opposite logic is applied after every 5 consecutive bits of same logic

Questions?

References

- Wikipedia contributors, "CAN bus," *Wikipedia, The Free Encyclopedia*, https://en.wikipedia.org/w/index.php?title=CAN_bus&oldid=915800522 (accessed October 10, 2019).
- H. F. Othman, Y. R. Aji, F. T. Fakhreddin and A. R. Al-Ali, "Controller Area Networks: Evolution and Applications," *2006 2nd International Conference on Information & Communication Technologies*, Damascus, 2006, pp. 3088-3093.
- Corrigan, S. (2016, May). Introduction to the Controller Area Network (CAN). Retrieved from http://www.ti.com/lit/an/sloa101b/sloa101b.pdf.
- Comer, D. (2015). *Computer networks and Internets*. Boston, MA: Pearson.