Recitation #11

18-649 Distributed Embedded Systems

13th November 2015

Electrical & Computer ENGINEERING

Carnegie Mellon

Note: Course slides shamelessly stolen from lecture

Announcements and Administrative Stuff

- **Project 11 due Thursday Nov 19th**
- **Project 12 due Monday December 7th**
- **Presentation slides due Sunday December 6th at 5 PM**
- **Presentations week: December 7th 2015 onwards**
- **Final project due Tues, Dec. 15th.**
- **15th December 2015 is hard deadline.**

A Few Words on Traceability

- **Don't slack on traceability**
  - The final project handing are graded more thoroughly than others, and mistakes you may have gotten away with before might be found later
  - Do it right the first time

- **Just as a heads-up, the final project grading criteria requires complete end-to-end traceability**
  - Avoid taking shortcuts with process
  - Introduces errors in design traceability and makes bugs harder to track down
  - End up generating extra work for yourselves

- **You should NOT be using the Future Expansion column anymore to complete your traceability tables.**

Project 11

- **Complete Network Schedule**
- **Meet Bandwidth constraints**
  - Modify payload translators
- **All unit tests must pass**
- **Create integration tests**
  - Integration tests must run, but do not have to pass
- **Run acceptance tests**
  - Acceptance tests must run (kind of), but do not have to pass.
  - It's okay for your design to E-Brake, for this project.
- **Update traceability**

Message Dictionary

Message Dictionary					
Sender Node Name	**Message Name**	**Deadline (ms)**	**Message ID**	**Sender Node Type**	**Replica**
Safety Sensor	Emergency Brake	50	1000	20	none
At Floor Sensor	AtFloor				
	Hoistway Limit				
	Hall_Light				
	Hall_Call				
	Drive_Speed				
	Drive_Command				
	Door_Motor_Command				
	Door Reversal				
	Door Opened				
	Door Closed				
	Desired_Floor				
	Desired_Dwell				
	Car_Position				
	Car_Light				
	Car_Lantern				
	Car_Call				
	Car Weight Alarm				
	Car Weight				
	Car Level Position				

Message Dictionary

Message Dictionary					
Message Name	**Deadline (ms)**	**Message ID**	**Sender Node Type**	**Replication Type**	**Base CAN ID**
Emergency Brake	50	1000	20	none	0x0BE81400
AtFloor					
Hoistway Limit					
Hall_Light					
Hall_Call					
Drive_Speed					
Drive_Command					
Door_Motor_Command					
Door Reversal					
Door Opened					
Door Closed					
Desired_Floor					
Desired_Dwell					
Car_Position					
Car_Light					
Car_Lantern					
Car_Call					
Car Weight Alarm					
Car Weight					
Car Level Position					

Network Analysis

Sender Node Name	Message Name	Base CAN ID	Replication Count	Deadline (ms)	Desc
Safety Sensor	Emergency Brake	0x0BE81400	1	50	Ebrake
At Floor Sensor	AtFloor				
Level Sensor	Level				
	Hoistway Limit				
	Hall_Light				
	Hall_Call				
	Drive_Speed				
	Drive_Command				
	Door_Motor_Command				
	Door Reversal				
	Door Opened				
	Door Closed				
	Desired_Floor				
	Desired_Dwell				
	Car_Position				
	Car_Light				
	Car_Lantern				
	Car_Call				
	Car Weight Alarm				
	Car Weight				
	Car Level Position				

Network Analysis Continued

	Field 3			Total Payload Bit Len	Total Payload Byte Len	Best Case Msg Len	Worst Case Msg Len	Best Case BW (bits/sec)	Worst Case BW (bits/sec)
Bit Len	Desc	Type	Bit Len						
0	n/a	n/a	0	1	1				
							Total bw		

Reducing bandwidth consumption

- **Up till now, simulator has run with unlimited bandwidth**
  - Previously it was OK to send a boolean value in a 64-bit payload
- **Now bandwidth is limited to 200,000 bits per second (this value will not be changed in the future)**
  - Need to modify data payloads to reduce bandwidth consumption
- **Use "-b 200" for acceptance tests**
  - You do not need to use this for integration and unit tests
- **Computing best case and worst case bandwidth**
  - Best case is no stuff bits
  - Worst case is one stuff bit per every four data bits
  - For a refresher check out the formula in Lecture 12 - CAN Performance
    - Use this formula for the most conservative upper bound

Ways to Reduce Bandwidth Consumption (1/2)

- **Ensure that you have a single message dictionary**
- **Remove unused message types**
  - E.g., mHallLight (remember to remove it from all other documentation too! For example - message dictionary in Requirements I, updating removed messages and the input and output interfaces in the Requirements II)
- **Combine message types**
  - Must have same period
  - Must originate from same instance of a node
    - You can combine multiple messages originating from the same dispatcher
    - But you can't combine messages from four different door controllers
- **Use the minimum number of bits to send values**
  - CAN specification requires payloads to be in bytes
  - Takes 1 byte to send an 'on' or 'off'
    - 7 unused bits, so you could send other messages in these
  - How many bits do you need to send the drive speed? Hint: Its not 64

    (Keep an eye the number of bits being used for it)

Ways to Reduce Bandwidth Consumption (2/2)

- **Issue log update**
  - If you end up removing a message, ensure that the issue log is updated.
- **If no module actually reads a given message, its generally safe to remove it.**
- **You may NOT add new messages to the schedule unless you obtain approval from the course staff.**
- **You may NOT change message periods unless you get prior approval from one of the TA's (Preferably in an email copying the group & staff mail list)**
- **You may NOT use bits of the message ID to transmit data values (limitation of the simulator architecture)**
- **You may NOT remove any of the pre-defined constants. Doing so may cause your code to no longer be compile compatible with the simulation framework.**

Final Presentation

1. **Showcase design aspects of your *elevator***
   - You spent the whole semester working on it
   - Tell us about the coolest parts or biggest challenges!

2. **Lessons learned about process**
   - Now that you've had a chance to do a relatively large design project using process, tell us about it
   - Good vs. bad
   - What bugs you found in various phases of review and testing

- **We want to emphasize that there is much more flexibility for content in the design explanation portion than previous presentation**
  - If you're unsure whether what you want to present is appropriate in content or scope, ask us!
  - But, required elements need to all be there (especially the metrics)

Course Project Exit Criteria

- **Run Time Monitor Must Be Implemented**
  - Pass all unit tests with zero failed assertions
  - Pass all integration tests with zero failed assertions
- **Pass all acceptance tests**
  - Using -b 200 and -fs 5.0
  - Zero failed assertions (after startup)
- **Must have a working elevator to complete the course**
  - "Working" means passes the set of tests listed on the final project web page
  - Non-working results in Incomplete if you don't get it working by grade deadline
- **+1% final grade for best elevator (one group only)**
  - Rank groups by average performance and satisfaction across acceptance tests
- **+2% final grade for complete and consistent design portfolio**
  - All groups are eligible for this

Questions?