30 de Noviembre 2018

YAZAKI TESTS

Semester i: Automotive Embedded Technology

Pedro Antonio Gámez

A00369538

Sharif Nasser Kadamani

A00820367

Digital System and Robotic Engineers

INSTITUTO TECNOLÓGICO DE ESTUDIOS SUPERIORES Y MONTERREY

MONTERREY, NUEVO LEON

	Printing established Datapool in Arduino 23-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez, Sharif Nasser			
Description	Test that with only the computer with Atmel installed I can print the datapool and extract a specific signal, in this case the Gear Postion, seeing the datapool in the putty			
Function	Establish a predefined datapool, establish addresses, read datapool, extract signal, display datapool.			
Components	Datapool data , control (main). Uartdriver			
Hardware	PC with Atmel and Serial reader, FTDI , Arduino Zero			
Procedure		Expected behavior	Actual behavior	Result
	Establish a datapool using structures and define the addresses of all the elements	We expect to create a correct union struct for the datapool with the elements information of the vehicle in it.	Figure 1. shows the defining of the addresses of the elements of the datapool Figure 2. Datapool structure	OK
	Step1: Start the program	Initialize all the components and elements in this case using the uart driver for the serial reading of the datapool.	Figure 3. Initializing the program in main	OK
	Step2: Start Putty with the right baud rate and read the buffer.	The program should continue execution and the rx_buffer should contain the datapool which shall be read at the using the putty interface	Figure 4.First try with DLC error marking 0 Figure 5.Putty reading	No, we had a minor where we corrected it by initializing correctly the length of the DLC
Comments	We had a minor problem which had to do with the DLC which was quickly resolved			

	Initializing TFT display with Static image 25-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez			
Description	Test that we can initialize the TFT display using the export file from the Storyboard, to just display the static image to the display			
Function	Display a static image to the TFT display			
Components	Graphics driver, Yocto Project, Ubuntu in virtual machine			
Hardware	PC with Linux, Sama, TFT display			
Procedure		Expected behavior	Actual behavior	Result
	Establish the correct connections between components, export Storyboard file and send it to the SAMA Board	Expect to have the correct export of the Storyboard	Figure 9. Storyboard Configuration, were we can observe the edited image that will be sent to the SAMA	OK
	Step1: Export Storyboard configuration to SAMA	Through scp transfer send the storyboard configuration to the SAMA	Succesful transfer form the Storyboard to the SAMA board	OK
	Step 2: Execute SAMA and kill the application launcher task	Once the program is executed in the display is the application launcher which we must kill before sending the updated graph from the storyboard.	Figure 10. Application launcher in display that is being killed next	OK
	Step 3: Declare environmental variables and Run the executable code to display graphics.	We expect the TFT to display the image in the correct resolution that we exported from the Storyboard, sending it through the SAMA.	Figure 11. Shows the excecutable instruccionens sent to the SAMA board. Figure 12. Shows the correct imaged displayed at the TFT display.	OK
Comments	At first we had an error in the aspect ratio, so we had the image cutout, for that we needed to make various changes in resolution to get the image correctly displayed			

	Can Reading through SPI 23-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez, Sharif Nasser			
Description	Test that we can read the CAN values established form the Canoe, send them thorugh SPI form Pican to the Arduino reading the values using the UART			
Function	Read the values from Canoe to Arduino correctly in real time			
Components	Canoe , control (main). Uartdriver, SPI driver, MCPCANdriver, Myprintf driver			
Hardware	PC with Atmel and Serial reader, FTDI , Arduino Zero, PICAN, Vector			
Procedure		Expected behavior	Actual behavior	Result
	Establish a datapool in Canoe with the respective addreses	We expect to create a correct datapool in the Canoewith the same addresses as above	Figure 6. shows the defining of the addresses of the elements of the datapool in Canoe	OK
	Step1: Initializa UART, SPI communication. Initialize Datapool	Initialize all the communication and elements in this case using the uart driver for the serial reading of the datapool. And the SPI communication between the Pican and Arduino	Figure 7. Initializing the program in main with the communication protocols correctly	OK
	Step2: Send data from Canoe to Pican and to Arduino	The data must travel in CAN form using the vector to the Pican and from there to the arduino in SPI form	It worked correctly and we can see the values once we print them using the UART	OK
	Step3: Print updated Can values	The program should continue execution and the rx_buffer should contain the datapool which shall be read at the using the putty interface	Figure 8. Datapool printed correctly al the way form the Canoe	OK
Comments	There were no problems once we could correctly send from the same arduino, there were just minimal changes to the previous unit test.			

	Changing the values in the TFT display 25-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez			
Description	Test that we can read the CAN values established form the Canoe, send them thorugh SPI form Pican to the Arduino reading the values using the UART			
Function	Read the values from Canoe to Arduino correctly in real time			
Components	Graphics driver, Yocto Project, Ubuntu in virtual machine			
Hardware	PC with Linux, Sama, Piican, Arduino Zero,			
Procedure		Expected behavior	Actual behavior	Result
	Establish the correct connections between components, have Storyboard export ready and loaded to Sama, and Datapool established in Canoe program.	Expect to have the correct export of the Storyboard, Correct datapool established in canoe and the hardware correctly connected	Figure 9. Storyboard Configuration, were we can observe the edited image that will be sent to the SAMA Figure 12. Shows the physical connections made between hardware	OK
	Step1: Export Storyboard configuration, Initializa SPI communication in arduino.	Succesful Storyboard transfer and arduino initialization	Succesful transfer form the Storyboard to the SAMA board and arduino ready from Canoe and sending to SAMA	OK
	Step 2: Initialize I2C communication in SAMA and execute SAMA and kill the application launcher task.	Succesful initialization of Arduino and Sama as well as killing launcher app to begin sending the new graphics	Figure 10. shows initial application launcher	OK
	Step 3: Update at Datapool values in Canoe to be send to Can.	We expect the TFT to display the image and update the information displayed accordingly to the datapool values changed in the Canoe program	Video 1: Shows the Display updating its values when changing at the Canoe Video 2: Shows an error were we could refresh the value	OK
Comments	We made various test like sending a signal much higher than the normally permitted, for example sending a 500 mph, were 200 is the maximum, for this we limit the storyboard to only display its maximum which is 200. we also had some issues were it was quickly fixed because we were sending the values pointer insted of the value.			

	Storyboard Gear Position implementation 25-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez, Sharif Nasser			
Description	Test that we can implent a new element in the Crank Story board to add the gear position			
Function	Add gear position in Crank StoryBoard			
Components	Storyboard .gapp, graph_link			
Hardware	PC with StoryBoard Software			
Procedure		Expected behavior	Actual behavior	Result
	Create New Visuals in external application, or use text in Storyboard as new visual	Expect to create the new text for the gear position P R N D L and overlap them all	We now have the GearPostion in the CrankSoftware	OK
	Step1: Add an event to each letter of the PRND	Expect to create an event for each letter having a byte of information 0 being invisible an 1 visible	Correct creation of each Letter overlapped	OK
	Step 2: Create LUA file, and simulation	Create the Lua file to implement the information accordingly to the information that is being given and run the simulation	Correct simulation of the PRND texts.	OK
	Step 3: Testing in changing values	We expect the TFT to display the image and update the information displayed accordingly to the datapool values changed in the Canoe program	Video 2:Changing values and seeing the distinct forms	OK
Comments	In Canoe we send 0 1 2 3 4 5 6 to change the PRND,D1,D2,L and sending a value that is not in between 0 and 6 will leave us in the last established gear position.			

	Changing PWM 29-Nov-2018			
Team	RoboDuo			
Responsible	Pedro Gámez, Sharif Nasser			
Description	Changing the PWM for the LED Driver Dimming			
Function	Sending a specific PWM cycle for the dimming			
Components	Port.c, Pm.c, gclk.c, timer. c, main			
Hardware	PCwith Atmel Studio			
Procedure		Expected behavior	Actual behavior	Result
	Create the port for the PWM	Assign a correct port this case PA14 (PIN 2) to the PWM outout	Port assigned as output with PWM	OK
	Step1: Set Power manager for port	Assign the power that will flow through the PIN	Correct creation PORT	OK
	Step 2: Set GCLK an Timer	Assign the gclk and set the timer for the high to low change. And establish the correct frequency and duty cycle to the output.	Get the correct PWM frequency and duty cycle according to whats established.	OK
	Step 3: Testing in changing the PWM	Get the correct duty cycles at the output	Figure 14. Shows PWMs outputs at different duty cycles.	OK
Comments				

NAME	ID	DATA [0]	DATA [1]	DATA [2]	DATA [3]	DATA [4]				
ENG_SPEED	0x15	SPEED	0				0-200			
ENG_RPM	0x16	RPM	RPM				0-10,000			
ENG_OILPRS	0x21	PRESSURE					0-55			
ENG_OILLEVEL	0x20	LEVEL								
ENG_TEMP	0x25	TEMP	0				0-255	85-95		
ENG_GEARPOS	0x26	POS					0-6	P;R;N;D;D1;D2;L		
BD	0x30			BREAK		ARROW	0-1	0-3	NONE;LEFT;RIGTH;BOTH	
SAF	0x32			SEAT		DOOR	0-3	NONE;DRIVER;PASSENGER;BOTH		
INF_OT	0x64	TEMP_OUT					-128 - +128			
ENG_FUEL	0x22	FUEL					0 - 100			
INF_TIME	0x65	HOUR		MINUTE			0 - 24	0 - 60		
INF_GEN	0x63	BATTERY		HIGHBEAM	SWERVE	ABS	8 - 17	0-1	0-1	0-1
INF_ODO	0x60	ODO	ODO	ODO	ODO	ODO	0 - 9,999,999,999			
INF_TRIP	0x58	TRIP	TRIP	TRIP	TRIP	TRIP	0 - 999,999,999			

Figu

Figure 1.

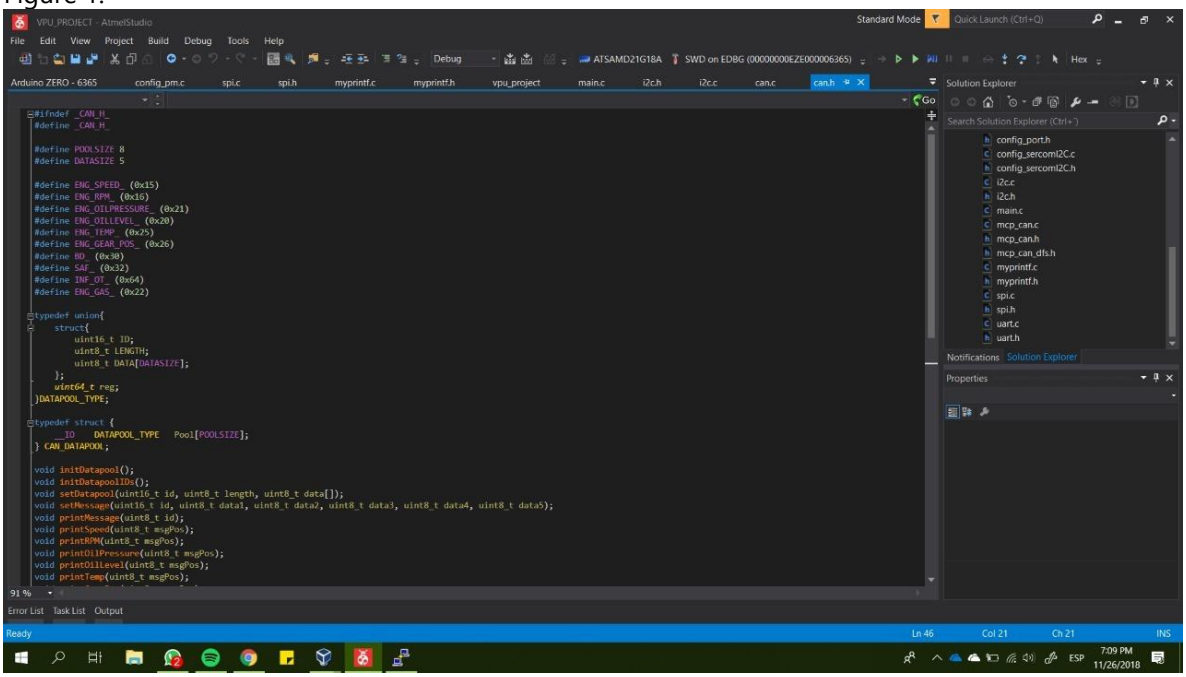


Figure 2.

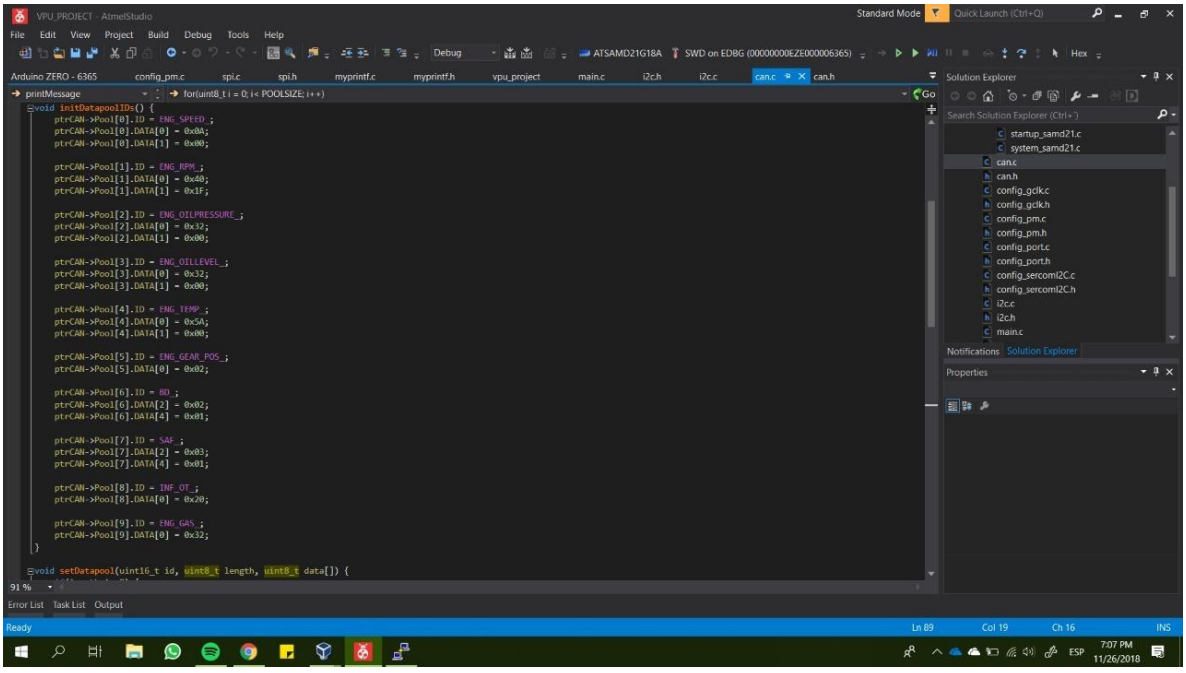


Figure 3.

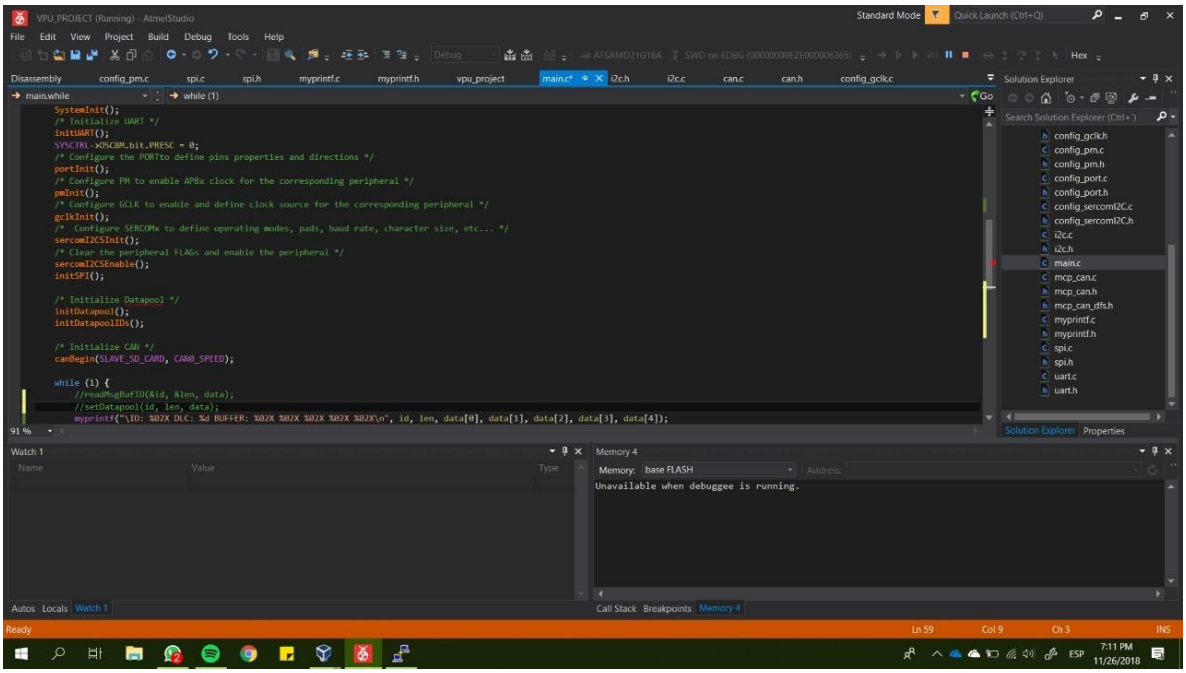


Figure 4.

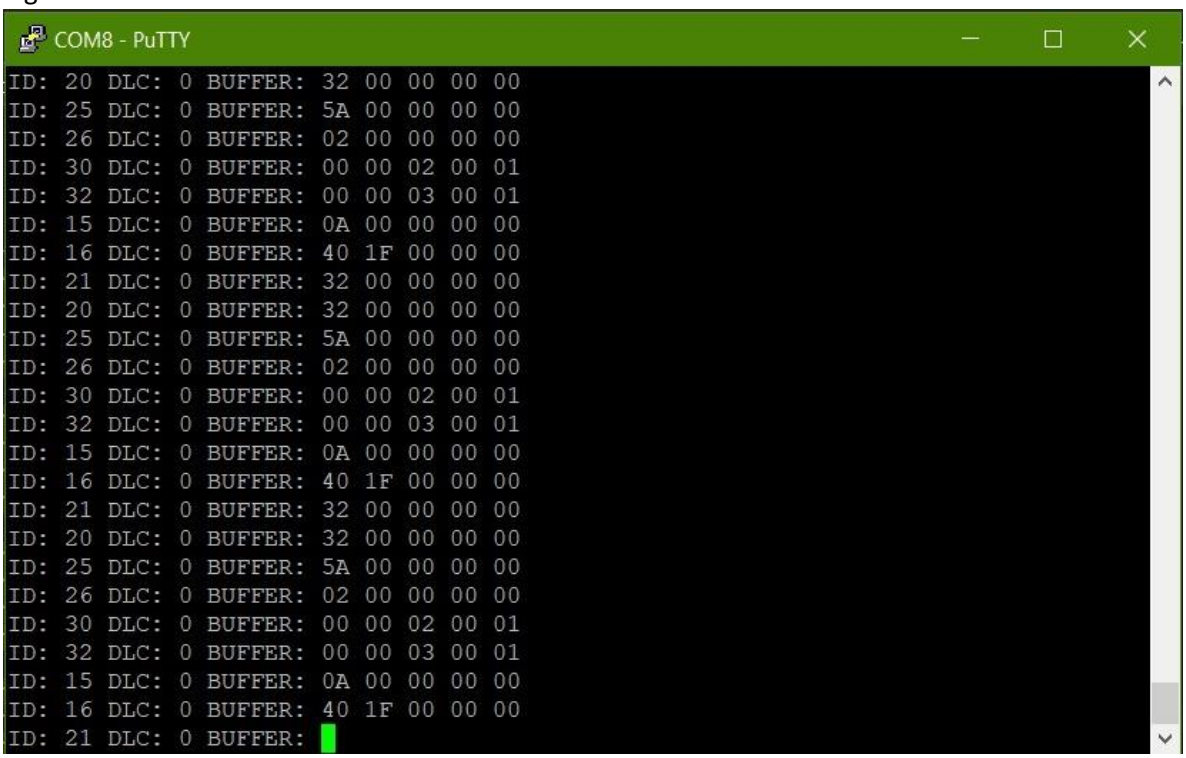


Figure 5.

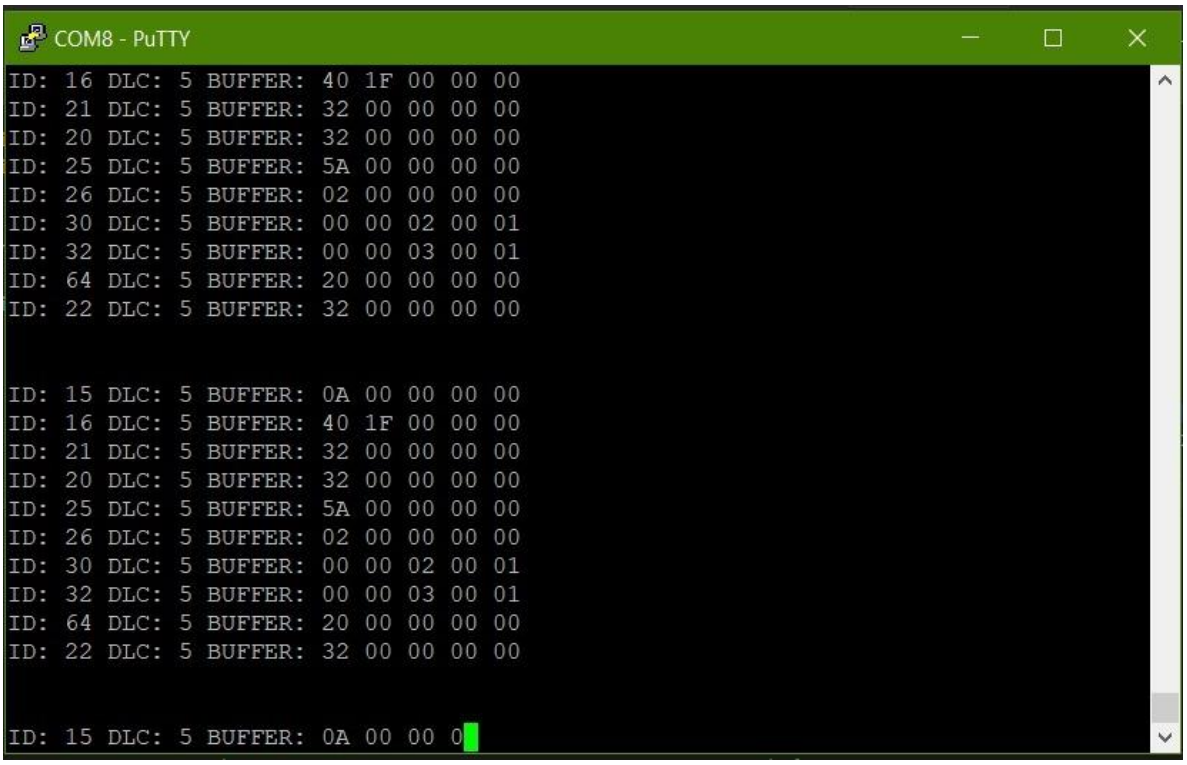


Figure 6.

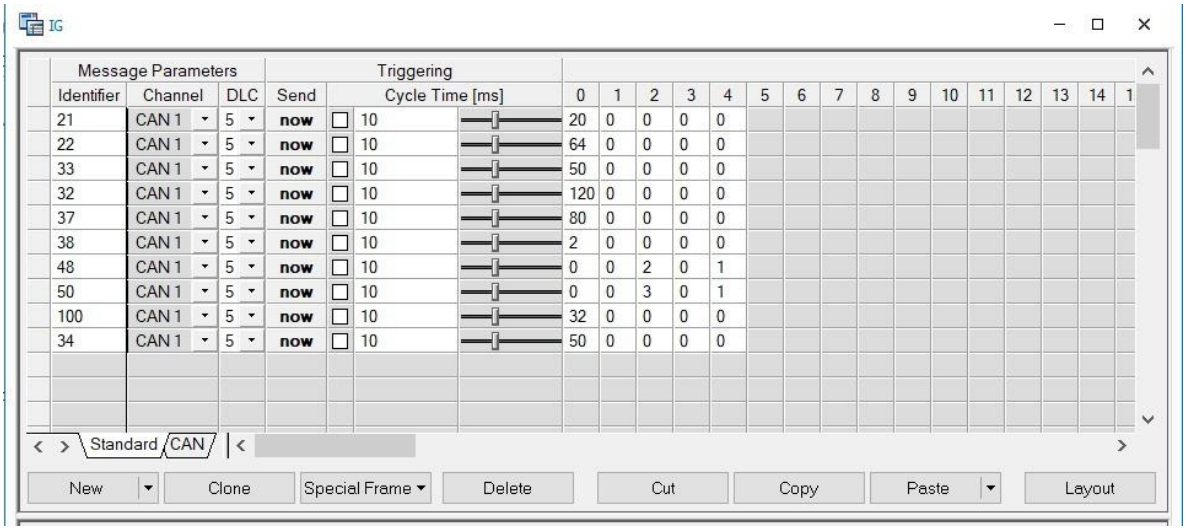


Figure 7.

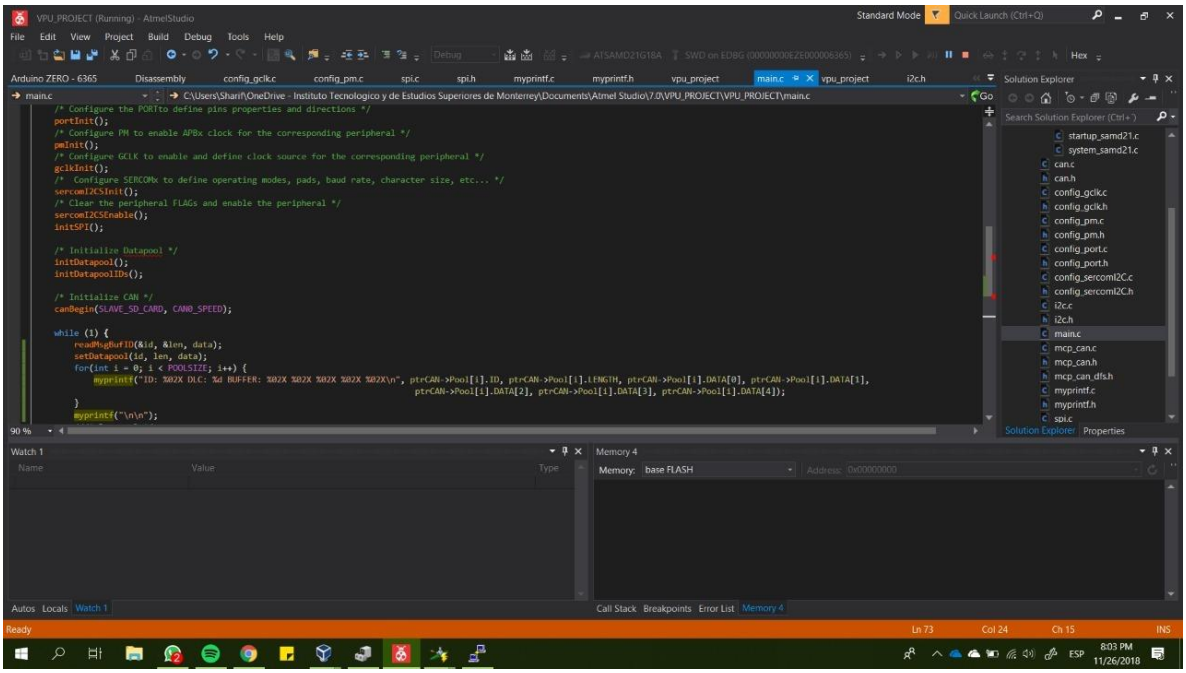


Figure 8.

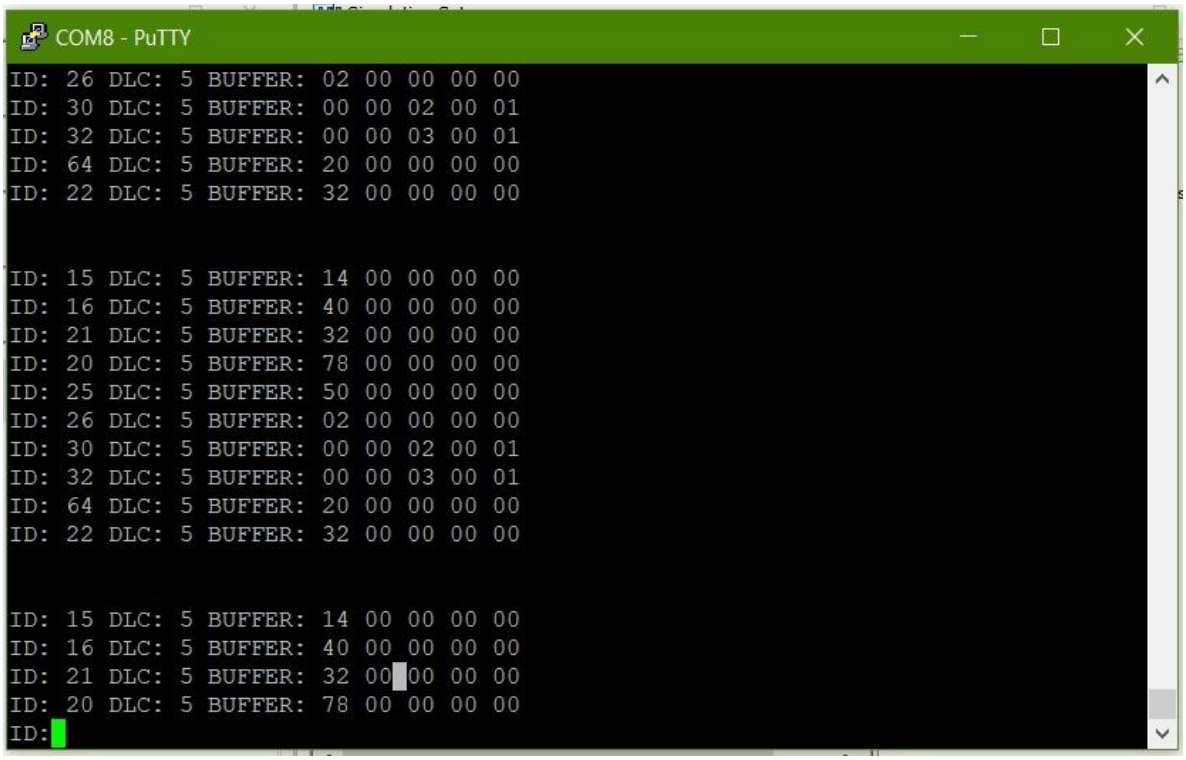


Figure 9.

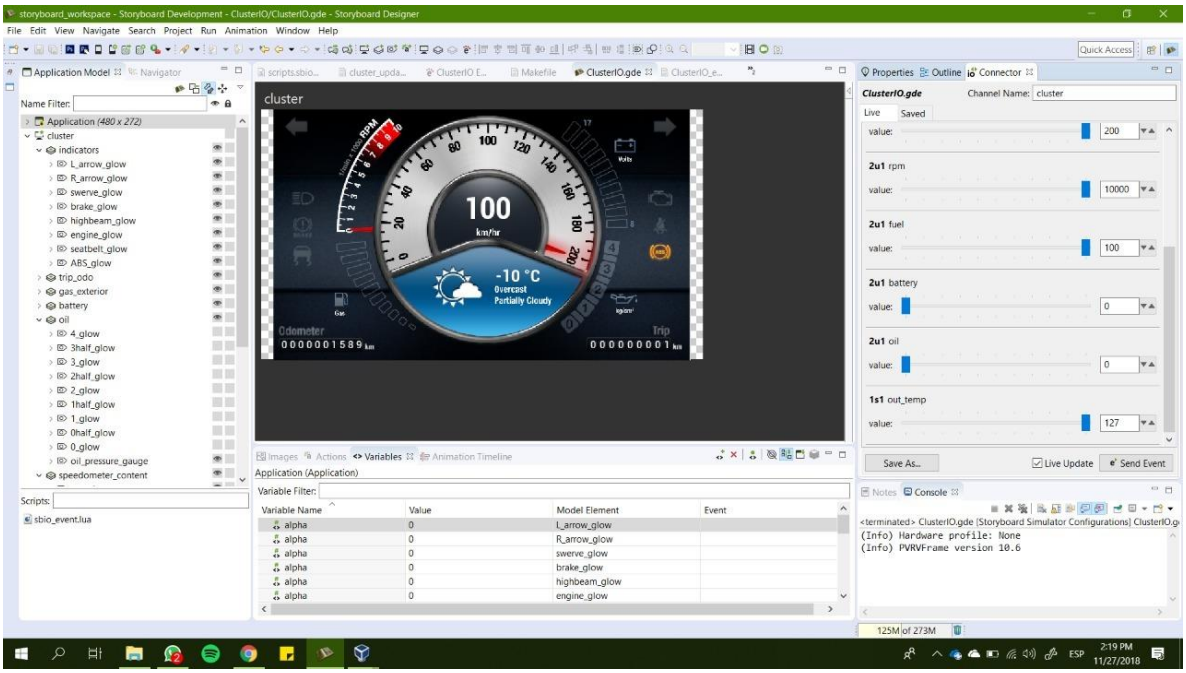


Figure 10.

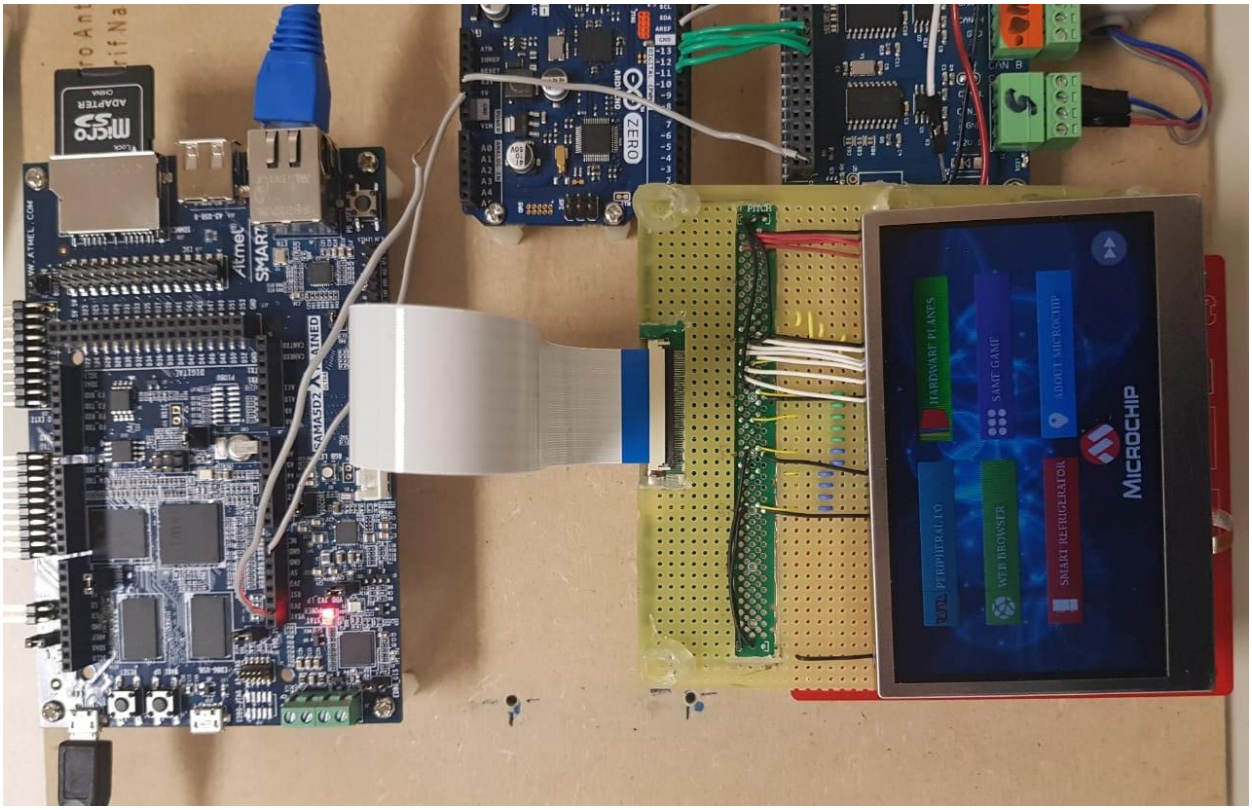


Figure 11.

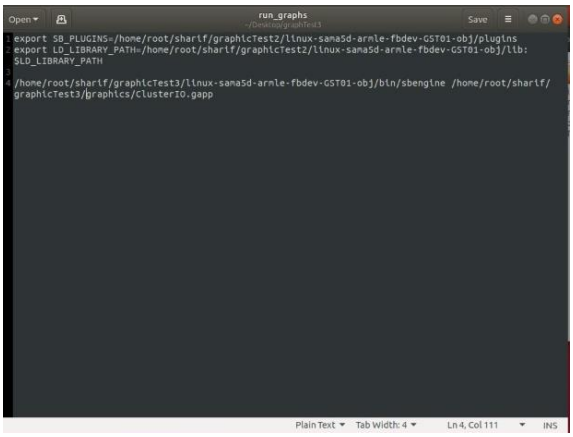
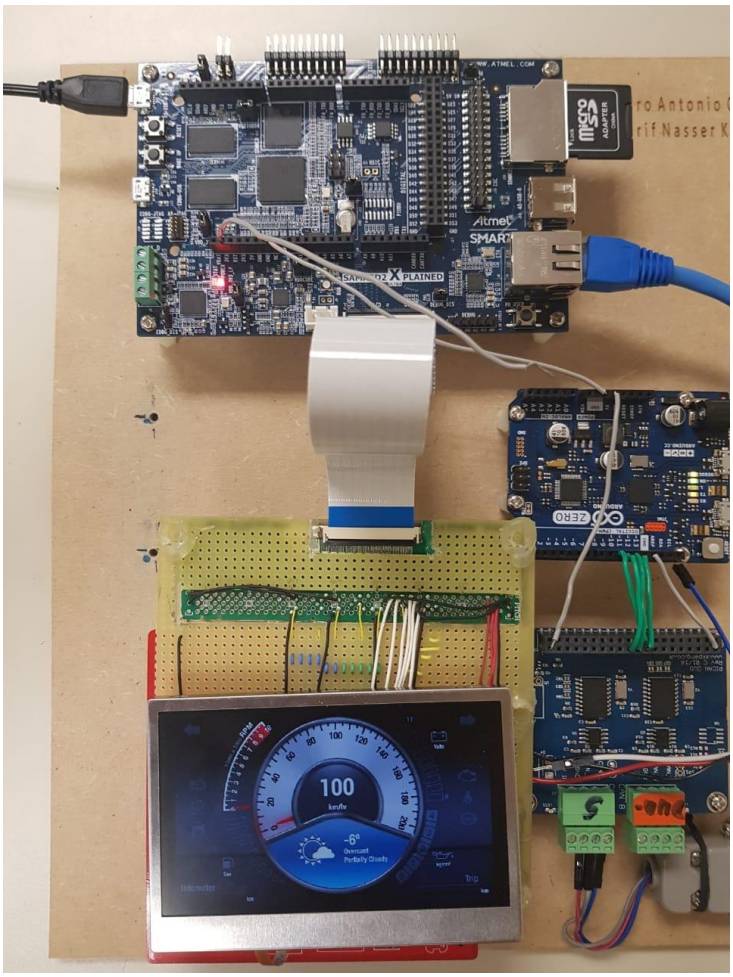


Figure 12.



Video 1.

https://drive.google.com/file/d/1S4Rp3hOnc8uHFqs4TseQQlZOoOFwMPG9/view?usp=drivesdk

Video 2.

https://drive.google.com/file/d/1U0rOlEeuFqbLii0HBMK4XentWDKHt8ID/view?usp=drivesdk

Video 3.

https://drive.google.com/file/d/1jEmB9WClN75Y6lGYY3MuP1v3Y80_HdyC/view?usp=drivesdk