



BACHELOR THESIS:
BO19E-35 Framo
Generator Simulation Tool

KAY THWE AYE

31. May. 2019

Document Control

<i>Report title</i> BO19E-35 Framo	<i>Date/Version</i> 30.May.2019 / Revision C
<i>Author(s):</i> KAY THWE AYE	<i>Report number:</i> B019E-35
	<i>Course:</i> 16HEEL
<i>Supervisor at Western Norway University of Applied Sciences</i> Per Eilif Thorvaldsen	<i>Number of pages including appendixes</i>
	<i>Security classification:</i> Open
<i>Comments:</i> We, the authors, allow publishing of the report.	

<i>Contracting entity:</i> FRAMO, FLATØY	<i>Contracting entity's reference:</i> Dan Robert Pedersen
<i>Contact(s) at contracting entity, including contact information:</i> Dan Robert Pedersen Direct: +47 55 99 94 46 E-Mail: drp@framo.no Odd Andre Boge Direct: +47 55 99 94 20 E-Mail: oabo@framo.no	

Revision	Date	Status	Performed by
A	28.01.2019	First Issue (Forstudie)	KAY THWE AYE
B	20.05.2019	Second Issue	KAY THWE AYE
C	30.05.2019	Final Issue	KAY THWE AYE

Preface

This project is performed under ELE150-1 19V course during the final semester as bachelor project for Bachelor of Electronics Engineering in Høgskulen på Vestlandet (Bergen). The contracting entity for this project is Framo, Flatøy. The internal project supervisor from Høgskulen på Vestlandet (Bergen) is Per Eilif Thorvaldsen. The external project supervisors from Framo Flatøy are Dan Robert Pedersen and Odd Andre Boge.

I would like to thank Framo, Flatøy for giving me the opportunity to work with this interesting and dynamic project. Although the project seems like a pure programming project by its title at first, it touches the other subjects such as hardware & software system construction, industrial communication protocol, basic knowledge of how generator works, and programmable logic controller.

I appreciate the great support and supervision given by Per Eilif Thorvaldsen who always give quick and precise instructions and feedbacks throughout the whole project. This project will not come this far without the time and supports given also by Dan Robert Pedersen and Odd Andre Boge, though they have tight and limited time schedules.

Finally, I would like to give a thank to Frode Natås (Project engineer, Software – Framo Flatøy) for providing me with all the necessary technical documentations, suggestions for improvement and testing of the simulation tool.

I am thankful for the patient and continuous advices from all my supervisors to be able go forward with the project and to have a smooth path along the whole project phases.

Summary

The project is assigned by Framo to develop a generator simulation tool which should act similarly as an actual generator. In simple term, a simulation tool which should be able to transmit generator signals to a Programmable Logic Controller (PLC) through a signal converter. A generator is generally a long lead item itself and adding on Framo customers' specific project specifications and technical requirements make the delivery time even longer. It might take average 6 to 8 months delivery time.

The main purpose of this project is to be able to test project software independent of the physical generator. Like heavy vehicles and diesel engines, generators normally use J1939 communication protocol to exchange data frames i.e. signal inputs and outputs.

The simulation tool is developed in visual studio 2017 and the programming language is C# windows form applications. J1939 network simulation card from Copperhill Technologies is used to be able to transmit the simulation data according to J1939 protocol.

It was identified two possible solution alternatives which are presented under chapter 3 of this report. These two solution alternatives are carefully analyzed in consideration of available time frame, complexity, previous knowledge to be applied and cost.

The generator simulation tool is designed and implemented in a simplest way so that the user will not have any doubt when using it. In addition, the simulation tool is made robust and flexible to make further improvement in the future. In the conclusion chapter of this report, it is also presented some thought about further improvement.

1 Index

- Document Control 2
- Preface..... 3
- Summary 4
- 1 Introduction..... 7
 - 1.1 Contracting entity..... 7
 - 1.2 Problem description 8
 - 1.3 Main idea of solution..... 9
- 2 Specification of requirements 9
- 3 Problem analysis..... 10
 - 3.1 Description of possible solutions 10
 - 3.1.1 Solution alternative 1 11
 - 3.1.2 Solution alternative 2 11
 - 3.1.3 Issues regarding tools and HW/SW components..... 12
 - 3.2 Problem analysis conclusion 12
- 4 Realization of selected solution 14
 - 4.1 Theory..... 14
 - 4.1.1 Generator 14
 - 4.1.2 J1939 Communication Protocol 17
 - 4.1.3 PLC 19
 - 4.1.4 HMI 19
 - 4.2 Programing the Application..... 20
 - 4.2.1 Code Layout & Structure 20
 - 4.2.2 Software Layout & Structure..... 24
- 5 Testing 33
- 6 Discussion 35
- 7 Conclusion 36
- Appendix A. Literature 37
- Appendix B. Abbreviations and dictionary explanations..... 38
- Appendix C. Project management 39
 - A.1 Project organization 39
 - A.2 Progress plan 39

A.3 Risk list..... 40

Appendix D. PGN & SPN Information..... 41

Appendix E. Source code, Bill of Materials etc..... 48

Appendix F. Midtveis rapport (for information)..... 49

1 Introduction

1.1 Contracting entity

Framo AS (formerly known as Frank Mohn AS) is based in Bergen, Norway, and is the parent company of Framo worldwide.

Framo became part of Alfa Laval in the middle of year 2014. Alfa Laval is a Swedish company and has been delivering heat exchangers , centrifugal separation and fluid handling. Alfa Laval has also been in marine industry for about 100 years now. ^[1]

Framo Flatøy has three main commercial areas which are Cargo pumping systems, Supply and recovery solutions and Oil & gas pumping systems. Framo Florvåg operates mainly with services, training and support for the delivered products.

Framo Flatøy manufacturing factory has advance manufacturing machinery and technologies. In addition, with the skillful engineers, it is known for delivering quality products and services.^[1]

Following are the list of Framo offices in Norway.

Framo Florvåg (Norway): (Head Office)

Framo Fusa (Norway): The manufacturing plant for marine pumping systems

Framo Flatøy (Norway): The manufacturing plant for oil&gas pumping systems

Framo Holsnøy (Norway): The manufacturing plant for piping systems

In addition to above, Framo has offices in Netherlands, USA, Brasil, Singapore, Japan, Korea, China and Dubai.

1.2 Problem description

This project related to the Framo diesel electric fire water pumps (FWP). Framo FWP system consists of Submerged electric firewater pump, Surge & Air release system and a fire water pump container.

The main important and essential parts inside the container are a diesel engine, a generator, an oil cooling unit, a sea water cooling system for engine and generator, a combustion air system, an exhaust system, a diesel fuel oil tank, HVAC system, Air cooling unit, Fire & gas detection and a unit control panel. [2]

The unit control panel handle the power distributions and control and monitoring of the system described above.

This project has mainly focus on control and monitoring relating to the Generator. PLC software for Fire water pumps packages are developed, designed and tested on Framo Flatøy. Generators used on FWP are specified and ordered from a sub-supplier. As generator is a long-lead item, the software testing process on generator is dependent upon the delivery time of the generator. Therefore, software testing is not possible before they are delivered and has done set-up on the workshop at Framo. Framo would like to improve this and make the software testing possible regardless of the delivery time of the generator.

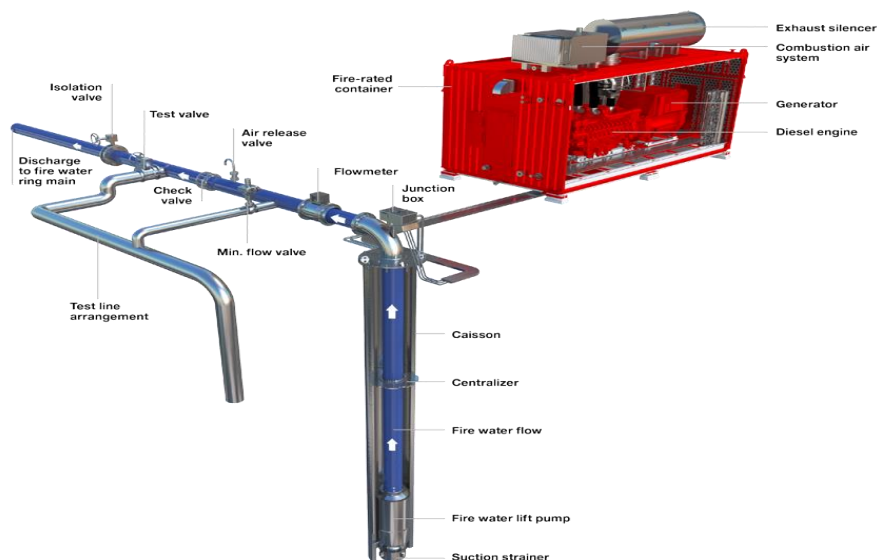
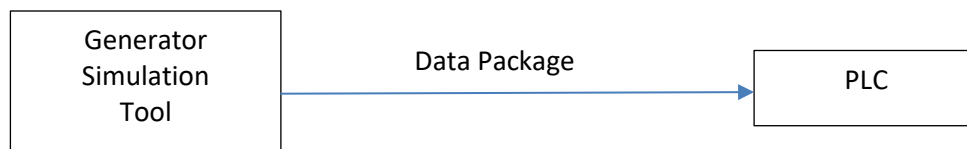


Figure 1.1: Framo diesel electric fire water pumps [3]

<https://www.framo.com/oil-and-gas-pumping-systems/products/fire-water-pumps-electric/>

1.3 Main idea of solution

The main idea of solution is to be able to test the PLC software without having the physical generator. It is to be achieved by using a simulation program which should be able to send the data value to the PLC software. Then Framo would be able to monitor and analyse if the PLC software respond as expected and programmed. The simulation tool programming language and communication protocol can be any which is suitable for this purpose.



2 Specification of requirements

This project is to develop a generator simulation program in visual studio or similar, so that Framo can test their software with PLC without the physical generator on workshop. As mentioned earlier in the previous chapter, the expected outcome of this project is a functioning generator simulation software tool, which is developed in Visual Studio or similar and which have possibilities for testing of generator short-circuit, over-voltage and some other scenarios.

The simulation program shall also made so that it is possible for further development in the future.

3 Problem analysis

The traditional test set-up of PLC software against the generator is shown in the figure 3.1 below. The set-up is simplified and left out the all the other detail which are not relevant for this project. As mention earlier in the report, the challenge with this test set-up is that this create delay on testing of the PLC software. Framo would like to perform their PLC software as early phase in a project as possible so that all the bugs and errors in the software can be detected and fixed.

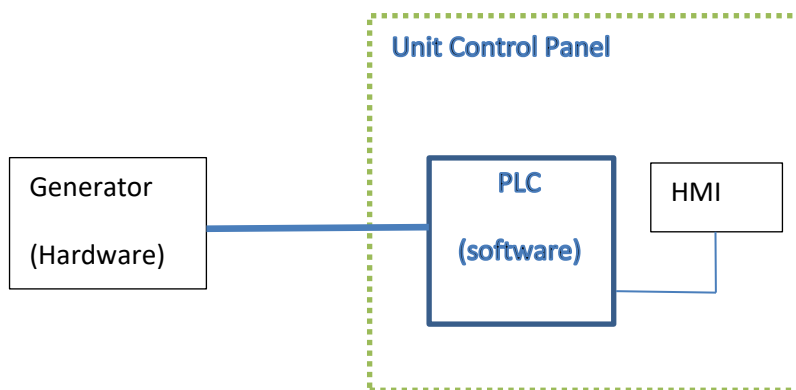


Figure:3.1 Current software testing option with physical Generator on workshop

3.1 Description of possible solutions

The possible solution is to design and develop a unit which generates generator output data to the PLC software. In other word, the unit should act like a virtual generator. It can either be solved in hardware basis, software basis or in combination of both. In the next session, two possible solutions will be discussed and made a comparison based on complexity, timeframe, previous knowledge and cost.

Generators are complicate machine and it would be difficult to create a complete simulation tool for the generator. After discussion with the contracting entity, we have limited the generator output values which the contracting entity would like to be tested with PLC software. The detail list of these signals will be mentioned further in the report.

In addition, the simulation program to be developed so that it shall be possible to develop further and add extra features in the future.

3.1.1 Solution alternative 1

The first possible solution is to design or build a physical control panel including electronic circuits with adjustable knobs. It will involve designing electrical circuits and signal conversion, conditioning and processing unit. The signals have then to be formatted into data package and using communication protocol to send into PLC software.

It will involve designing analog and digital circuits, testing and implementation of these circuits.

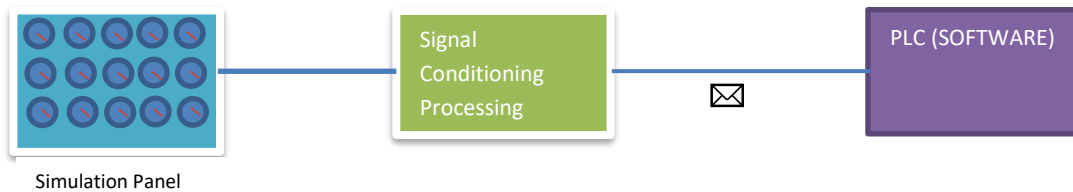


Figure 3.1.1.1: Possible Solution Alternative 1

3.1.2 Solution alternative 2

In contrast to the first possible solution, instead of having a physical panel, the simulation data to be generated using a software. In other word, a GUI where it is possible for user to type in the desired simulation data values and transmit toward the PLC software. For this solution, it might need for data simulation and processing unit. In the market, there are some available data simulation and processing cards which are suitable for this project task.

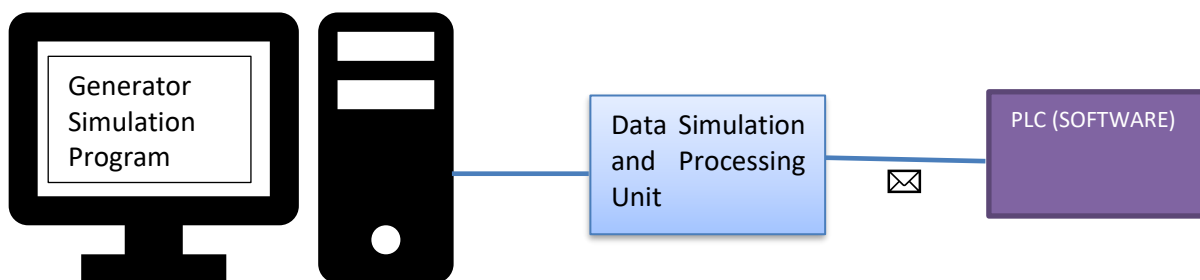


Figure 3.1.2.1: Possible Solution Alternative 2

3.1.3 Issues regarding tools and HW/SW components

If the possible solution alternative 1 is chosen, the extensive study must be done regarding designing of analog electronic circuits for control station panel and in addition for the signal conversion, conditioning and processing unit. The design can be done using Orcad which was used under ELE126 Analog circuit construction and design course. Making prototype of the electronics circuits and testing equipment such as multimeter, Oscilloscope must be considered.

Else if the possible solution alternative 2 is chosen, the software can be developed in Visual studio 2017. Visual Studio support many programming languages such as Python, C/C++, C#, Java, Visual Basic etc. C# has many pre-made GUI functionality which is ready to be implemented. I have had course in C++ programming and have a student license available for Visual studio software. Although it will be necessary to find the Data Simulation and Processing unit and has a quick study on J1939 data communication protocol. After discussion with the contracting entity relating to this possible solution2, we agreed to use J1939 communication protocol and J1939 simulation cards from Copperhill Technologies. ^[4]

3.2 Problem analysis conclusion

In theory both possible solution alternatives can be used here for this project. However, the available timeframe, complexity, project requirements & specifications, previous knowledge on the subject, cost and risks shall be considered. I will breakdown these criteria below and make a comparison.

- **Available timeframe - Approximately 6 months.**

Solution Alternative 1: Will take more than 6 months to finalize this solution

Solution Alternative 2: Timeframe is manageable and not pose as a challenge.

- **Complexity of the solution**

Solution Alternative 1: Very complex. Extensive circuit designs and testing.

Solution Alternative 2: Fair and manageable.

- **Project requirements & specifications**

Both solution alternative will fulfill the project requirements and specifications equally.

- **Previous knowledge on the subject**

Solution Alternative 1: Limited knowledge on advanced circuit design and development.

Solution Alternative 2: Fair and manageable. Had previous knowledge on C++ programming.

- **Cost and risks**

Solution Alternative 1: Higher cost and the risk of failure is higher

Solution Alternative 2: Fair cost and the risk of failure is minimal.

<i>Comparison between solutions</i>	Solution alternative 1	Solution alternative 2
Timeframe		
Complexity		
Project requirements & specifications		
Previous knowledge		
Cost & risks		



The above comparison clearly shows that I should go with the solution alternative 2 and the contracting entity also agreed.

4 Realization of selected solution

It will be focused mainly on the theory parts of the project in the first chapter, and then practical realization of the selected solution will be presented later.

4.1 Theory

4.1.1 Generator

A generator is a device that converts mechanical energy into electrical energy. Figure 4.1.1.1 is the basic structure of the generator. Generator is used mainly to generate voltage and power required by the firewater pump package.

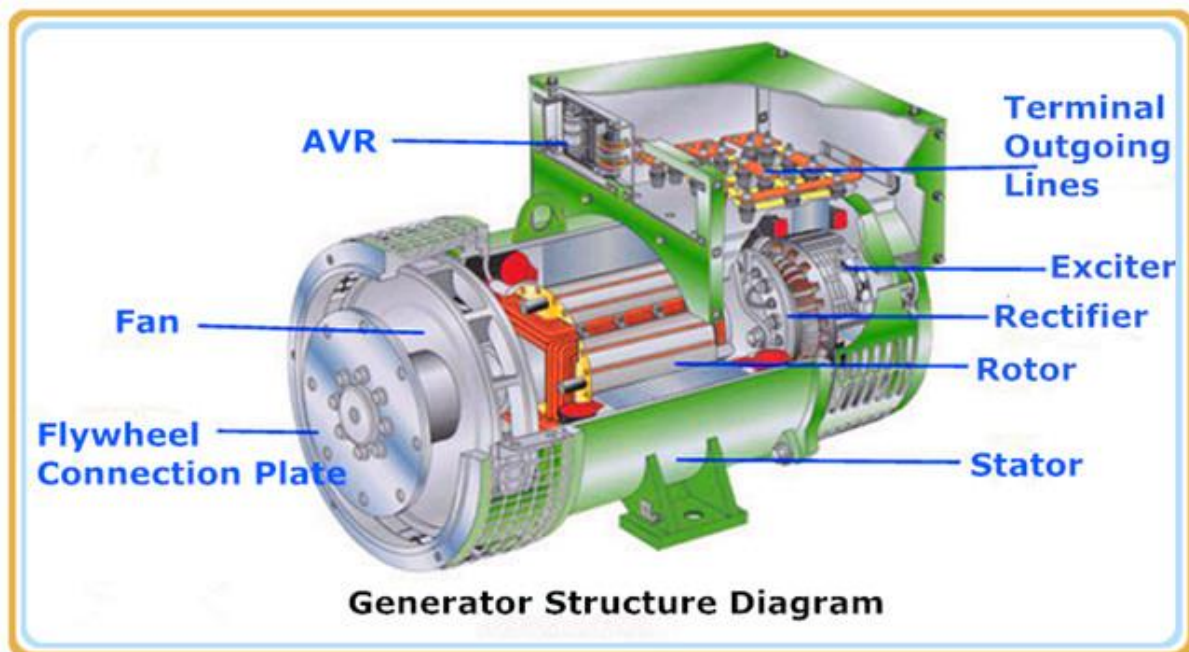


Figure 4.1.1.1 Generator Structure Diagram ^[6]

The contracting entity would like to simulate and test following signals (Table 1.1) with their PLC software. Generator simulation program will send these signals using J1939 protocol. Moxa signal converter will receive these data packages and perform signal conversion and I/O mapping before they are sent onward to PLS hardware.

The signals are identified by PGN number and SPN number with predefined data type, size and range by J1939 standard association (SAE). The desired data can be set in Generator simulation program and the simulation program will generate data packages based upon user input. The

detail descriptions on the action and respond from the software are not taken here as it is not part of this project interest. The contracting entity has made a hardware set-up which will be able to examine and check the software action upon receiving the data from simulation program. The main purpose is to check if the software acts as planned and programmed based on the set values from Generator simulation program.

Following is the list of the interested signal which the contracting entity would like to simulate and a short description of their purpose.

Parameters	Purpose
Generator Phase AB line	Monitoring of voltage between phase A-B (L1-L2)
Generator Phase A line neutral	Monitoring of voltage between phase A-Neutral Line (L1-N)
Generator phase A frequency	Monitoring frequency on phase A (L1)
Generator Phase A AC RMS current	Monitoring current on phase A (L1)
Generator Phase BC line	Monitoring of voltage between phase B-C (L2-L3)
Generator Phase B line neutral	Monitoring of voltage between phase B-Neutral Line (L2-N)
Generator phase B frequency	Monitoring of frequency on phase B (L2)
Generator Phase B AC RMS current	Monitoring of current on phase B (L2)
Generator Phase CA line	Monitoring of voltage between phase C-A (L3-L1)
Generator Phase C line neutral	Monitoring of voltage between phase C-Neutral Line (L3-N)
Generator phase C frequency	Monitoring frequency on phase C (L3)
Generator Phase C AC RMS current	Monitoring current on phase C (L3)
Generator Average Line-Line AC rms Voltage	Monitoring of average voltage between phases
Generator Average Line-Neutral AC rms Voltage	Monitoring average voltage between phases and Neutral-Line
Generator Average AC rms Frequency	Monitoring average frequency on all phases
Generator Average AC rms Current	Monitoring average current on all phases
Generator Excitation Field Current	Monitoring of the generator's excitation field current
Generator Total Reactive Power	Monitoring of the total reactive power
Generator Overall Power Factor	Monitoring of the overall power factor
Generator Total Real Power	Monitoring Calculation of the generator total real power
Generator total Apparent Power	Monitoring Calculation of the generator total apparent power

Automatic Voltage Regulator (AVR)

The most important function of the automatic voltage regulator is voltage regulation, conditioning and stabilizing. Automatics voltage regulator is an essential component for Alternating current (referred to as AC) brushless generators. The main purpose with the regulation is to stabilize the continuous supply within a predefined range so that the performance and stability for the connected loads i.e. motors, lights, etc. will be kept at safety state. It is achieved by keeping the terminal voltage magnitude of a generator at a constant level. [7]

A generator without an automatic voltage regulator may cause serious risks and losses. A scenario with sudden load reduction where the rotation speed of the generator changes and thus the generator will generate higher voltage. As the result, the electrical components will receive the higher voltage than which they withstand and might be burnt out. Another scenario is when the load increases suddenly where the generator will generate lower voltage. In this case, the electrical components will not receive the necessary voltage level which in term will cause malfunctioning or might perform as not intended. In both scenarios mentioned here, there are risks both for material and personal safety. Therefore, automatic voltage regulator is an essential control device for every power system. [7]

The contracting entity normally use synchronous brushless generator on their Firewater Pump Container packages from various generator suppliers. The signal from generator A.V.R. goes to the unit control panel with the programable logic controllers (PLC) are normally mounted. The differences A.V.R signal goes to I/O cards on the PLC and the PLC software either take necessary action or give warnings/alarms on the system. On this project the main task is to generate the values from Generator Simulation Tool and send it as J1939 package to PLC via Moxa converter, as the function of the Generator Simulation Tool is to check if the PLC software reacts to various signal level from A.V.R as programmed.

The following signal will be implemented in Generator Simulation Tool to be tested for A.V.R with PLC software.

Generator Voltage	-Generator Voltage measurement
Generator Frequency	-Generator Frequency measurement
Generator I U	- Generator Current measurement U-winding
Generator I V	- Generator Current measurement V-winding
Generator I W	- Generator Current measurement W-winding
Generator P	-Generator Active Power (kW)

Generator Q	-Generator Reactive Power (kVAR)
Generator S	- Generator Apparent Power (kVA)
Generator PF	-Generator Power Factor

4.1.2 J1939 Communication Protocol

J1939 is a further development of CAN bus protocol which provides serial communication between electronic microcontrollers in generators, diesel engines and heavy duty vehicles. The main reason of further improvement of CAN bus protocol is that CAN bus has limited flexibilities and performance relating network management and reliability. This is achieved by extending CAN bus protocol with addition software layer. ^[8]

“The 29 bit message identifier consists of the regular 11 bit base identifier and an 18 bit identifier extension. The distinction between CAN base frame format and CAN extended frame format is accomplished by using the IDE bit inside the Control Field. A low (dominant) IDE bit indicates an 11 bit message identifier, a high (recessive) IDE bit indicates a 29 bit identifier. An 11 bit identifier (standard format) allows a total of 2^{11} (= 2048) different messages. A 29 bit identifier (extended format) allows a total of 2^{29} (= 536+ million) messages.” ^[9]

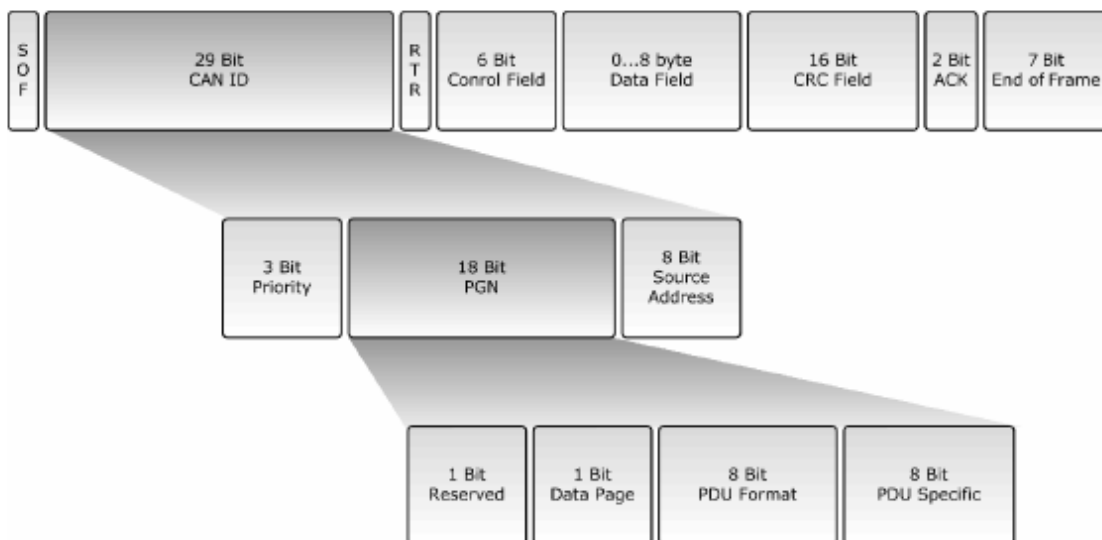


Figure 4.1.2.1 J1939 Message Format ^[8]

The main characteristic of J1939 is suspect parameter numbers (SPN) and parameter group number (PGN). SPN and PGN are embedded in the pre-defined location in J1939 data packages. The associated SPN are predefined for each PGN number by SAE association.

The first 3 bit of 29 bit message identifier defined the priority of the message. It decides which data package shall use the network first, in case when two or more nodes are trying to transmit the data at the same time. The value of '000' has the highest priority and the value of '111' has lowest priority. The most critical parameter group have the higher priority than the less critical one. For instance, if we take example in a car, the PGN relating status and warning of braking system has higher priority than the PGN for the light system. The 18bit after the priority bit is PGN information of the data package. The last 8 bit define the source address of the receiver. In case of a broadcasting message to the all the nodes, these 8 bits is set to global address of '255'.

Suspect parameter numbers define the data type and data length of each parameter in a parameter group. [8]

The following picture demonstrates the use of Suspect Parameter Numbers, Parameter Groups and Parameter Group Numbers. [10]

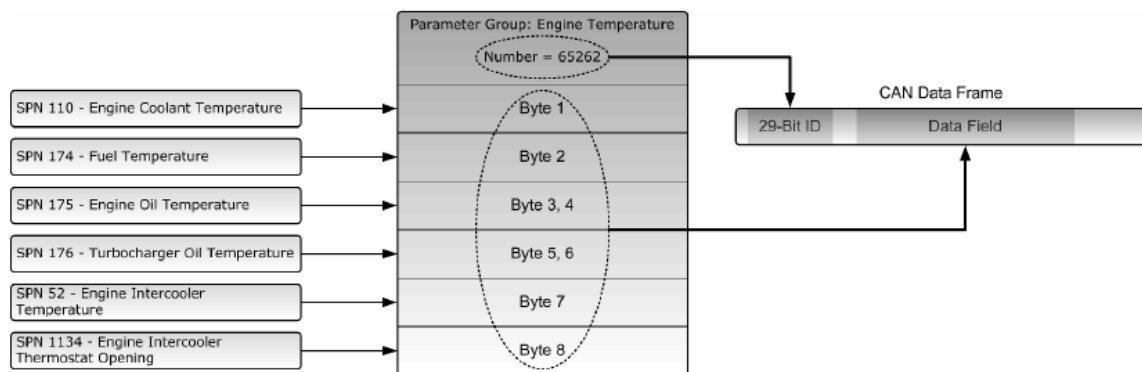


Figure 4.1.2.2 SPNs and PGN Example [10]

4.1.3 PLC

PLC stands for programmable logic controller. The main components in PLC are power module, processor, input assembly and output assembly. Power module is for power management for the PLC system. Processor which is also known as CPU is the brain of the PLC. Inside the processor the software programming is done. Input assembly reads all the input data from sensors and equipments. The processor then handles the logic based on the programmed software and give an action on the output assembly.

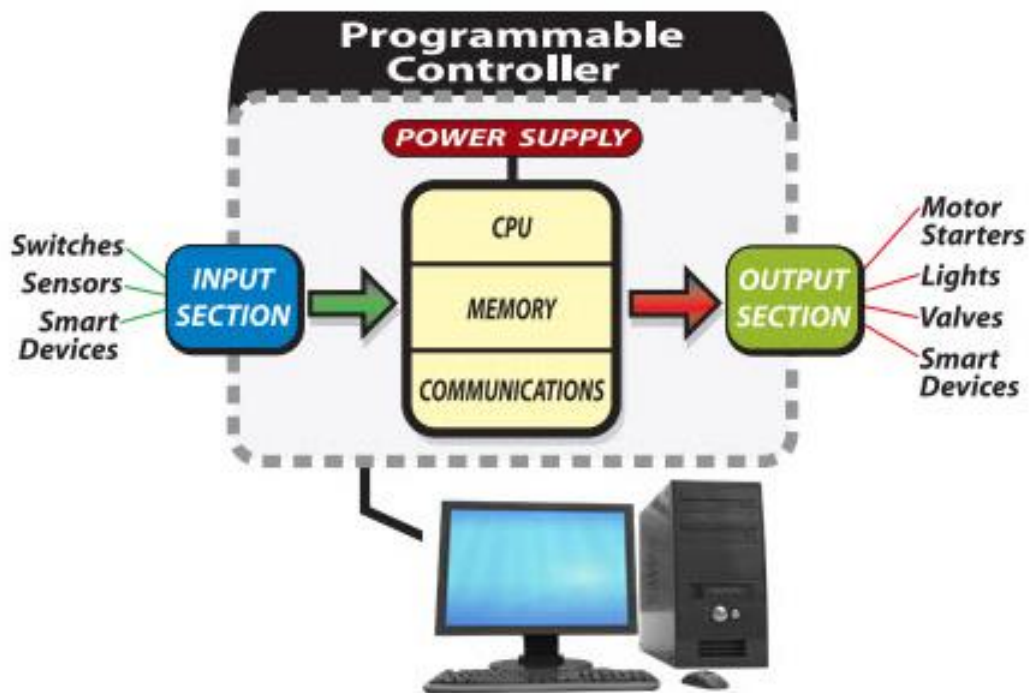


Figure 4.1.3.1 Basic PLC structure ^[11]

4.1.4 HMI

HMI stands for Human-Machine Interface. HMI are generally connected with a PLC. HMI is a graphical representation of a process in which the user gets the better overview of what is happening in the process. The user can see for instance the level of a tank or status of a valve with a graphical presentation. The user can control the process instruments and equipments via the HMI. For example, giving a command to shut-off a valve or to start a motor. Normally, the user can also see the alarms and warnings from the process instruments and equipments.

[12]

4.2 Programming the Application

4.2.1 Code Layout & Structure

In this chapter, the structure of the coding of Generator Simulation Program will be discussed in detail. As mentioned earlier in the previous chapter, the libraries provided together with the J1939 simulation cards made it flexible and faster coding for this project. In general, the coding structure can be demonstrated as shown in the drawing below. In the coming sessions, the description about their purposes and functions will be provided.

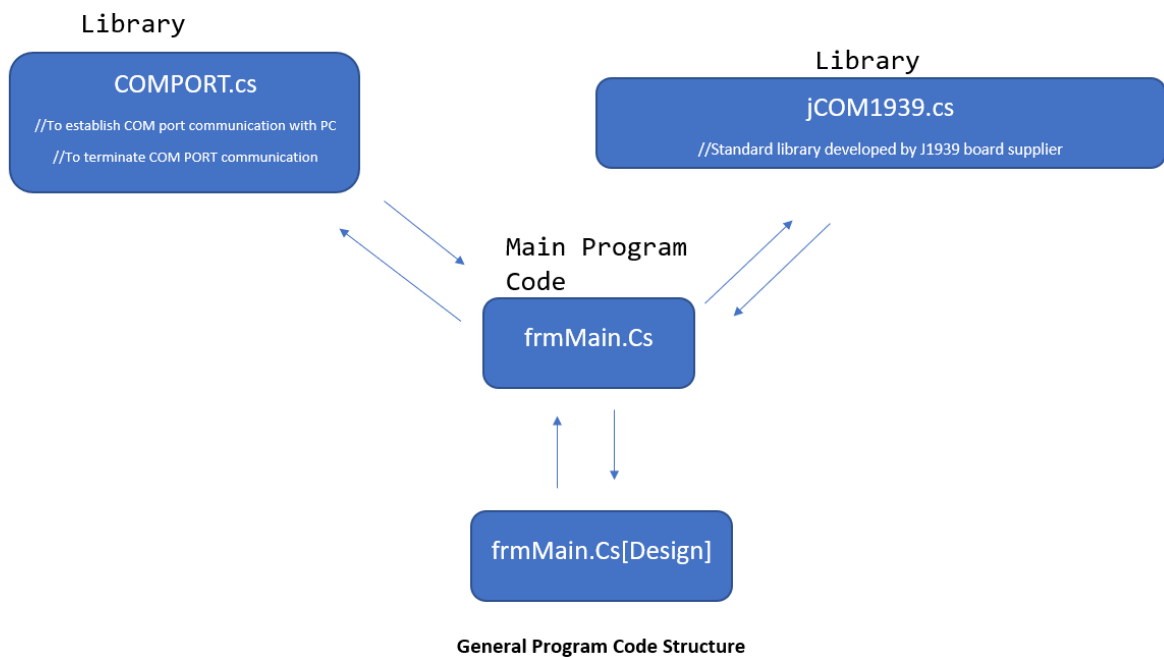


Figure 4.2.1.1 Generator Simulation Program Code Structure

COMPORT.cs

This library is included with the JCOM J1939 simulation cards. The main purpose of this library is to handle the connection between simulation cards and Windows COM Port (USB).

The main functions included in this library will be focused in this chapter.

The first function is to initialize the COM interface. It will scan and return which COM port number the J1939 cards are connected. If there is no COM port with J1939 cards connected, it will return an error message (-1) "No COM Port available".

```
public static int Initialize (int nComPort)
```

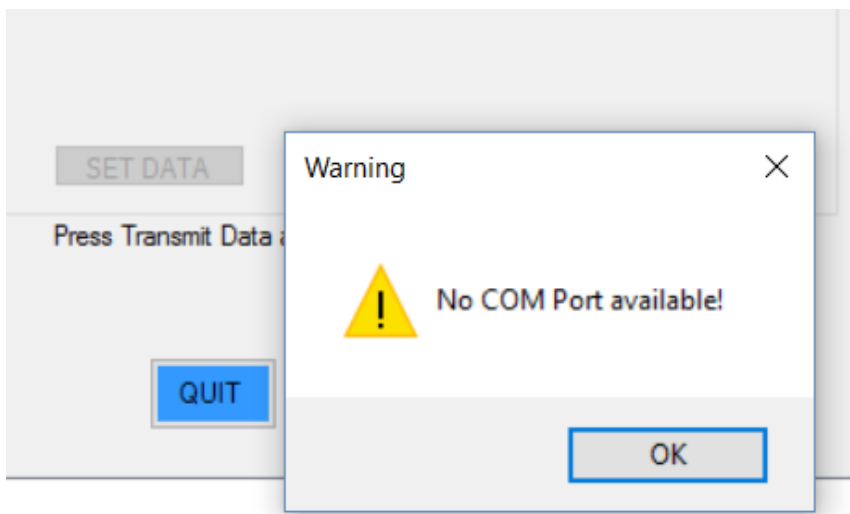


Figure 4.2.1.2 Example -No COM Port connected with J1939 card

The second function is to terminate an active COM port serial communication upon user's request.

The third function is to transmit the data to the selected Com port.

Other functions included are storing received messages into the global buffer & clearing the buffer.

jCOM1939.CS

This library is included with the JCOM J1939 simulation cards. The main purpose of this library is to create data frames according to J1939 protocol, to receive the data frames and to transmit the data frames to another J1939 node. In this session, only the main functions which are used for Generator simulation program will be focused.

The first function is to reset the gateway when user requested to terminate an active communication section.

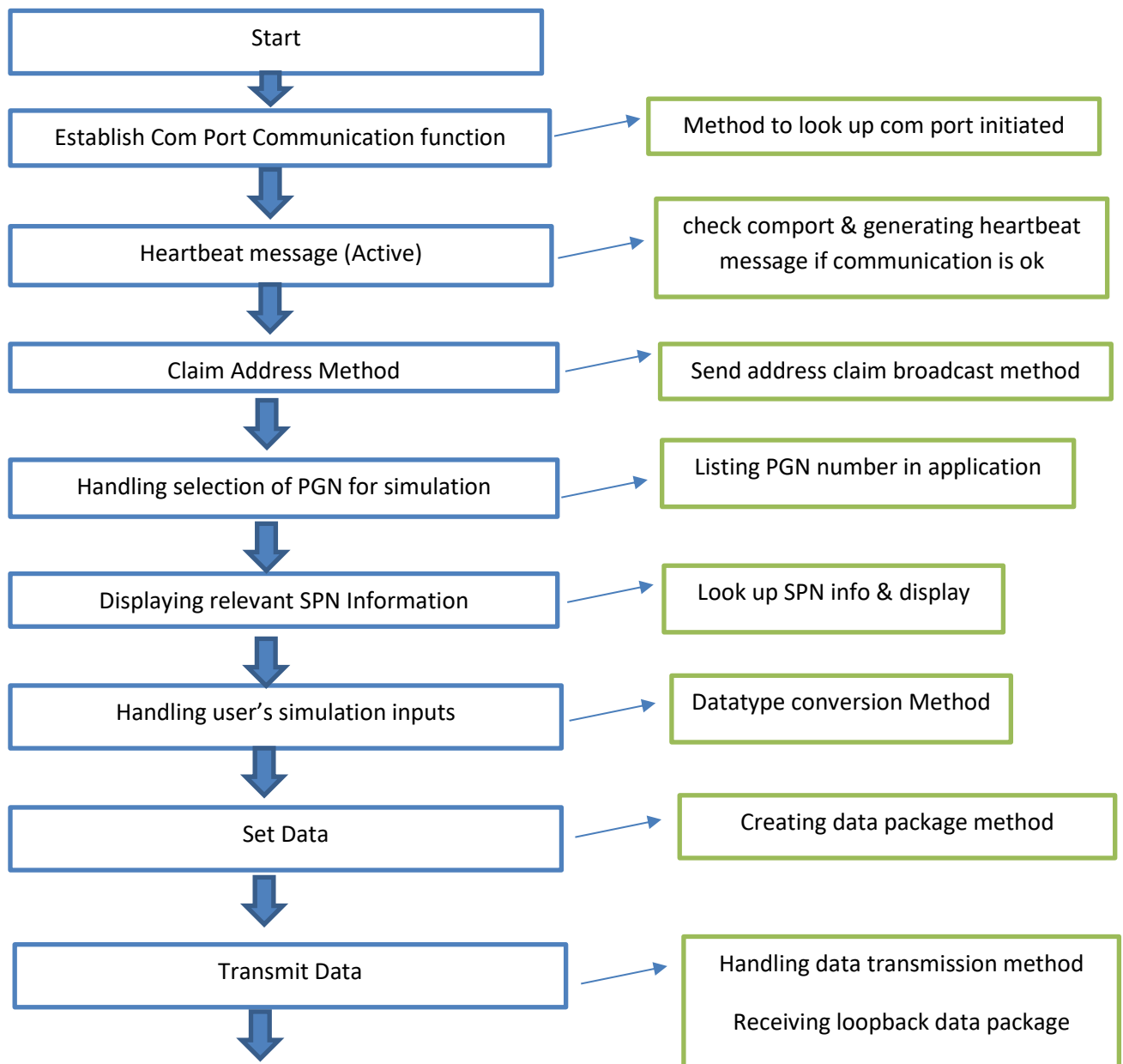
The second function is to handle the received data packages. Data packages can be of different types such as address claiming frames, acknowledgment frames, error frames, status frames, data frames, etc. jCOM1939.cs has all the necessary function implemented to handle the different type of data packages.

The third important function is to handle the transmit data packages. The transmitting packages, as similar to receiving data packages, have different. In the Generator Simulation Program, the address claiming process is a type of transmitting frame. Data entered by user to simulate in the program will also be arranged and created as a standard J1939 data frame by this function.

There are many other useful functions included in this library for other purposes. As these were not used for this project, I would not go further detail into them.

frmMain.cs

At the start of the program, the first thing the code does is to find which ports the J1939 nodes are connected. It will basically try to establish a communication section. If it was not successful achieved, then the user will be notified in a window message. In the message box, the watchdog function will start run and display Com port status every second. Before the user able to transmit the data, the address of other nodes shall be claimed. Upon clicking the "Claim Address", the program will create an address claim data package and broadcast it into the network. Then the program will listen the respond messages from another node. Upon receiving the address, the program will store this address in the network address list. Then it is ready to be transmit the simulation data. The user can now be able to punch in the simulation data and the data can be set when done. By clicking Transmit icon now, the data will be transmitted to the receiver.



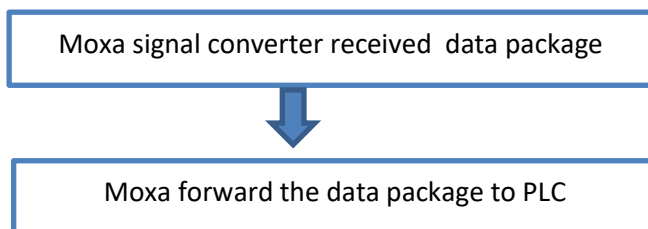
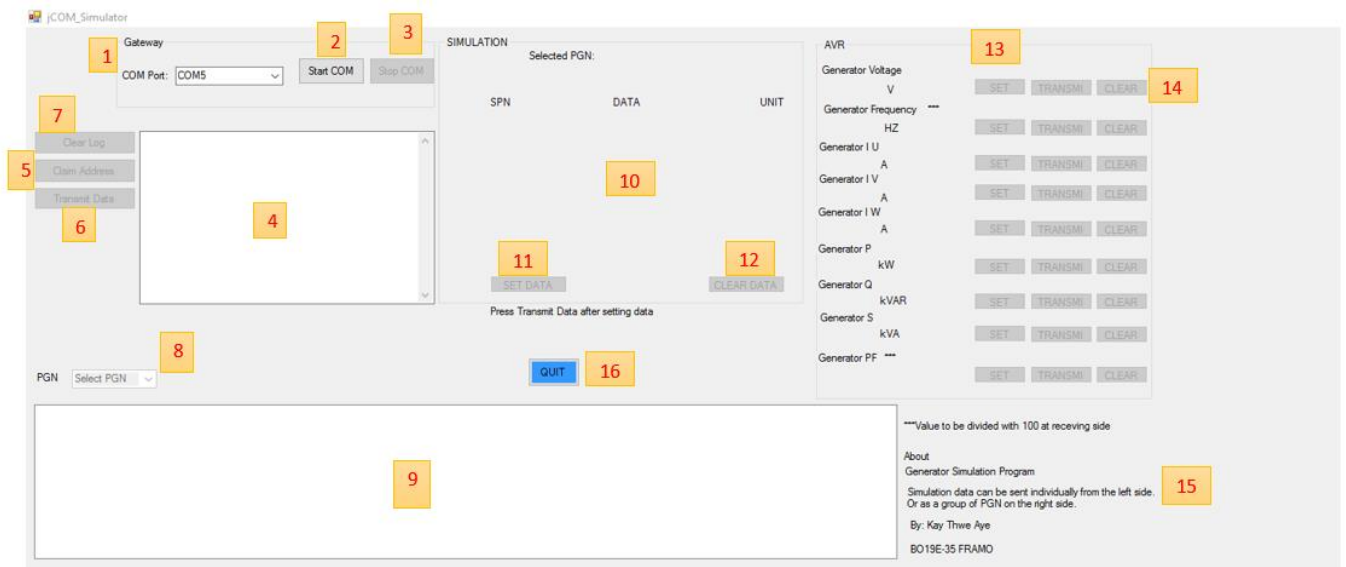
Framo


Figure 4.2.1.3 frmMain.cs code cycle

frmMain.Designer.cs

The code inside this library are generated base on the layout set-up made in the main frame design. It is basically included such as: If a label or button or textbox shall be as default active or inactive. The location of them in the design layout. etc.

4.2.2 Software Layout & Structure



Generator Simulation Program layout

1. Setup a communication section
2. Start the communication section with Com port on PC
3. Stop an active communication section
4. Message box for communication information
5. Claim address from connection J1939 node
6. Transmit data/package to another J1939 node
7. Clear the message box with communication information
8. Selection of the desired PGN to simulated
9. Display information on the selected PGN, regarding SPN numbers assigned for this PGN
10. Setting the simulation value to transmit
11. To set the data to transmit to another J1939 node
12. Clear the value entered for simulation
13. Transmit a SPN number individually
14. Enter the desired value to transmit/simulate
15. Short description about this software.
16. To quit the program.

1. Setting up a communication section

jCOM.J1939 Starter Kit and Network Simulator which is used for on this project has two J1939 nodes so that it is possible to test the generator simulation program without connecting to the actual hardware setup prepared by the contracting entity. The generator simulation program will detect the which USB communication port are connected on the computer and list them in the “Com Port” selection box. The user then able to select the Com Port. In this case, if both USB cables from the J1939 simulator are connected, there will be listed two communication port numbers here.

In the actual set-up, only one of the card will be used to transmit the simulation data onward to the PLC.

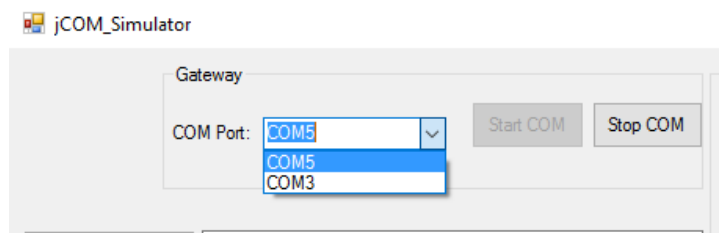


Figure 4.2.2.1 Two J1939 nodes connected

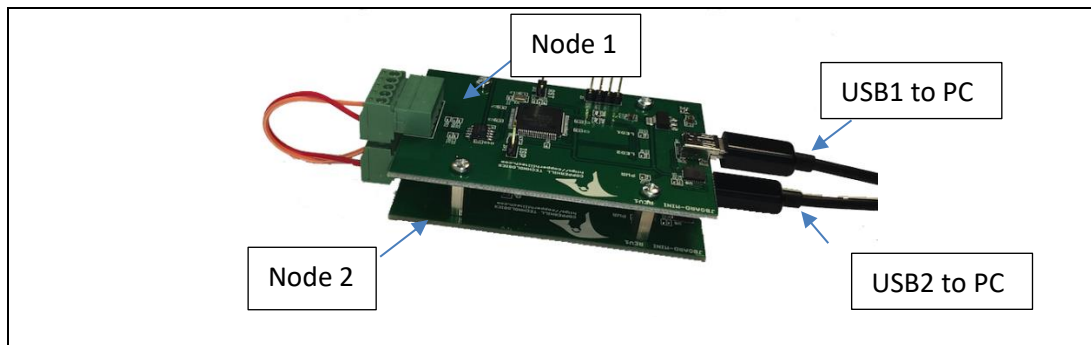
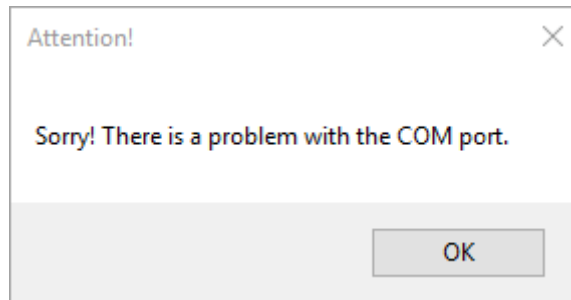


Figure 4.2.2.2 jCOM.J1939 Starter Kit and Network Simulator from Copperhill Technologies

2. Start the communication section with Com port on PC

After selecting the desired Com port from the list, by clicking the Start COM button, a communication section will be established.

If the program encounters a communication error, the message will be displayed. The user should then check if the USB cables are properly connected.



3. Stop an active communication section

This button is simply to stop the active communication section. Then the user needs to perform the address claim process to be able to transmit new set of simulation data packages.

4. Message box for communication information

During an active communication section, the message such as watchdog notification, in this case "Com port ok" will be displayed once in every second. This is to notify the user the connection with the USB communication port is active and working great. Acknowledgement statement after successful address claim process will also be displayed. Upon restarting a new communication section, all of the previous stored data in the arrays for instance will be reset. The reset acknowledgement will also be shown here. This message box is fully flexible to modify so that any other desired messages can be displayed in the future modification.

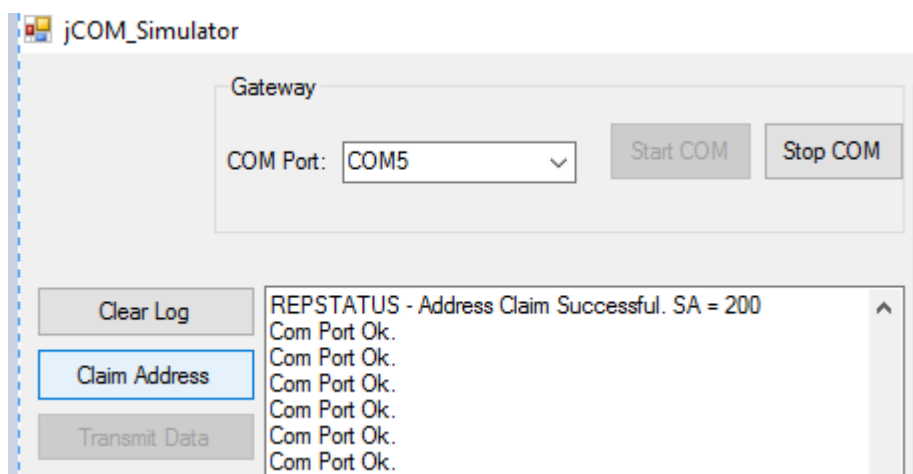
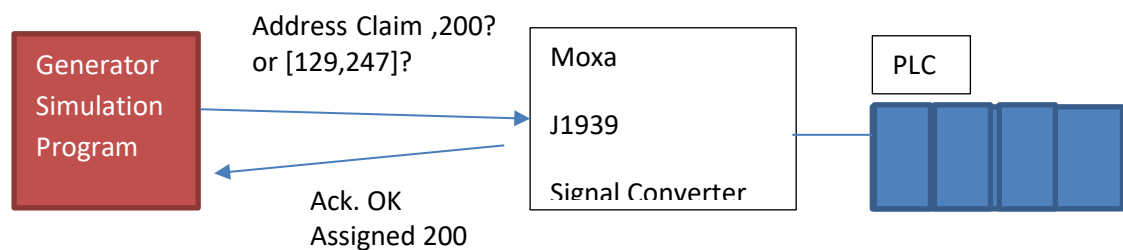


Figure 4.2.2.3 Address claim successful acknowledge in message box (Source address=200)

5. Claim address from connection J1939 node

Before a J1939 node transmits data package, it needs to know the address of the connected nodes in the network. At first, the node which would like to transmit data will send an address claim broadcast message to the all other nodes in the network. The address requesting node will suggest an address and a range. The other node will pick the first available address from this range. The advantage of using jCOM.J1939 Starter Kit and Network Simulator from Copperhill Technologies is that it comes with C# library which includes function to handle address claiming process. As described in the previous section, the node will respond with acknowledgment and the address which have been assigned. In this project, there will be two J1939 nodes, Generator simulation program and the data receiver on the contracting entity hardware set-up.



Library function used in Generator Simulation Program:

```
// Initiate the address claim process; request status feedback
```

```
jCOM1939.SET_J1939(SRCADDR, ADDRBOTTOM, ADDRTOP, jCOM1939.OPMODE_Event, pNAME, true);
```

SRCADDR: Suggested source address, 200 here

ADDRBOTTOM: Address 129 here

ADDRTOP: Address 247 here

6. Transmit data package to another J1939 node

This button is to transmit a data package to the another J1939 node, in this project to Moxa J1939 signal converter. After successful data transmission, the receiving node is asked to reflect back (loop back) the same package back to sender. According to J1939 standard protocol, unlike other communication protocol, no acknowledgement is sent to confirm the reception of a data.

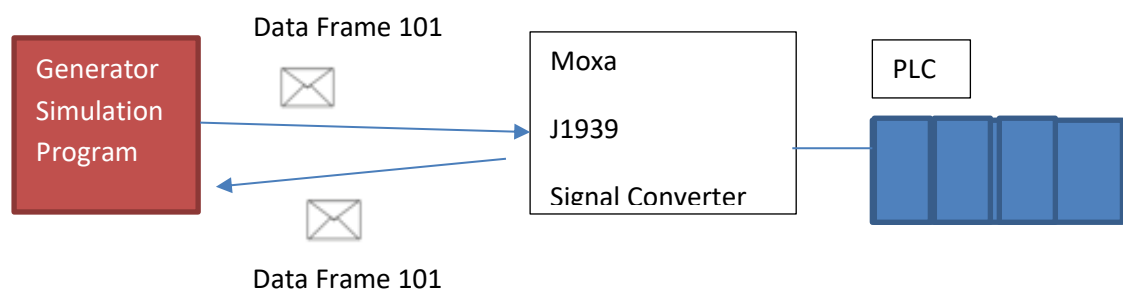


Figure 4.2.2.4 Data Frame Loopback illustration

7. Clear the message box with communication information

This button is to clear the messages in the message box .

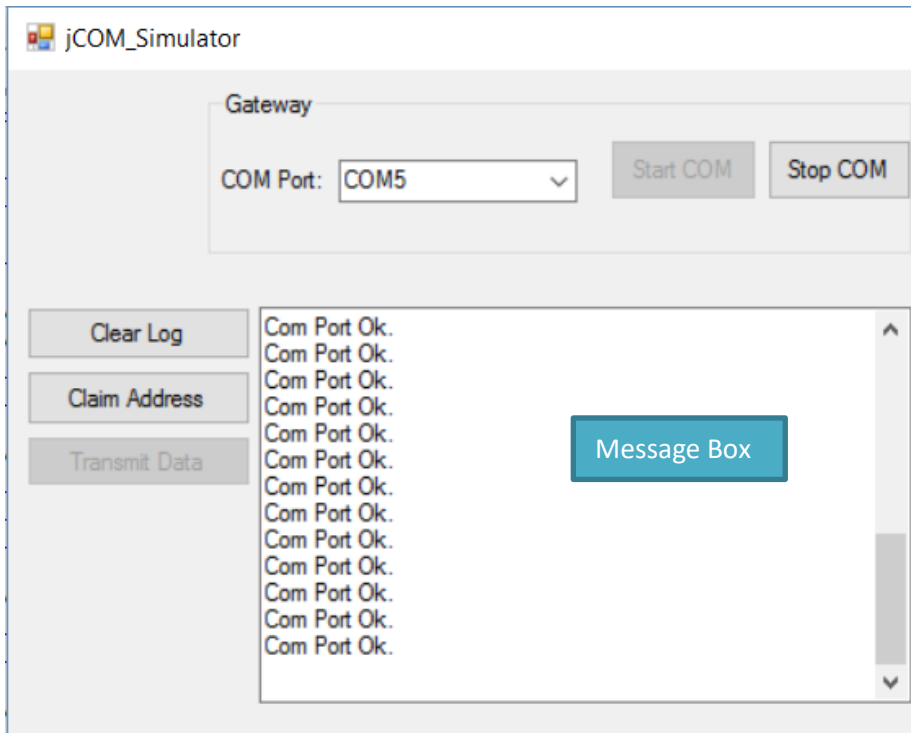


Figure 4.2.2.5 Message Box showing watchdog message popping up every second

8. Selection of the desired PGN to simulated

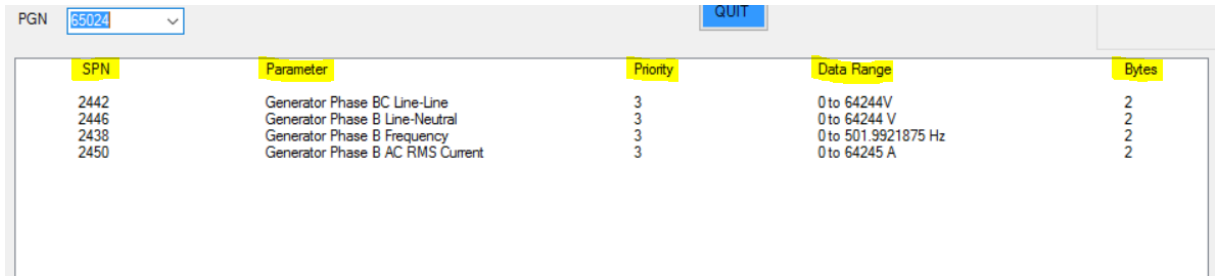
This list box contains all the desired PGN (Parameter Group Number) which the contracting entity would like to perform simulation of data. Upon selecting a PGN number, all the predefined SPN (Suspect Parameter Number), including their data range , description ,package priority and data size, will be displayed under .



Figure 4.2.2.6 Selection of a PGN (Parameter Group Number)

9. Display information on the selected PGN, regarding SPN numbers assigned for this PGN

On this area all the pre-defined SPN shown based on the selection of PGN number, as described in the above section.

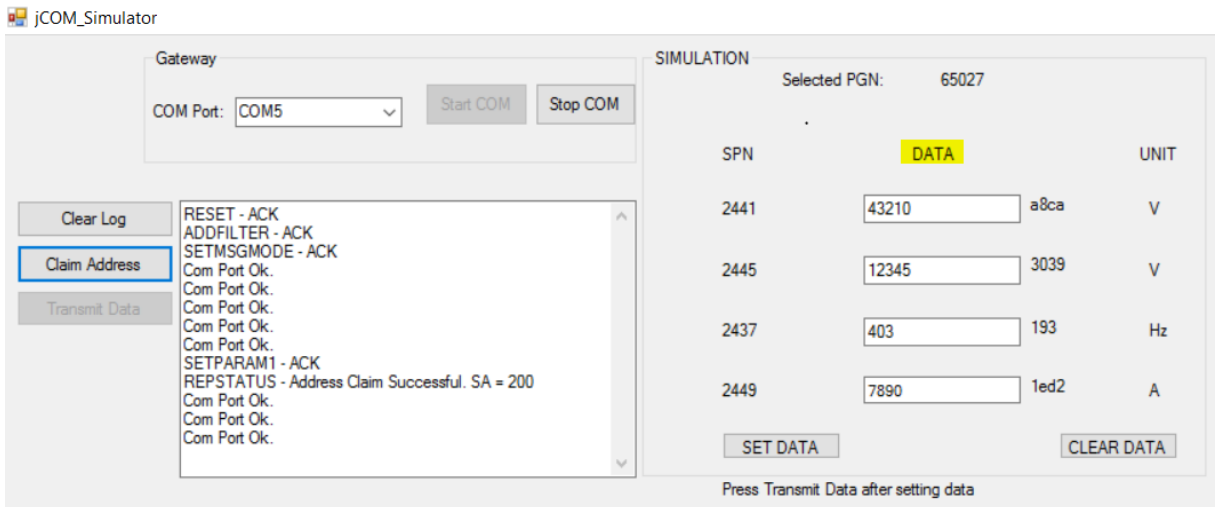


SPN	Parameter	Priority	Data Range	Bytes
2442	Generator Phase BC Line-Line	3	0 to 64244V	2
2446	Generator Phase B Line-Neutral	3	0 to 64244 V	2
2438	Generator Phase B Frequency	3	0 to 501.9921875 Hz	2
2450	Generator Phase B AC RMS Current	3	0 to 64245 A	2

Figure 4.2.2.7 Listing of a PGN (Parameter Group Number)

10. Setting the simulation value to transmit

After the user has selected the desired PGN number for simulation, this section will become active and it is ready to type in the simulation data. It is programmed so that the user can only type in the number, simply to prevent the typo errors. As the user type in the data, hex equivalent will be shown beside the textbox for information.



Gateway

COM Port:

Clear Log

Claim Address

Transmit Data

RESET - ACK
 ADDFILTER - ACK
 SETMSGMODE - ACK
 Com Port Ok.
 Com Port Ok.
 Com Port Ok.
 Com Port Ok.
 SETPARAM1 - ACK
 REPSTATUS - Address Claim Successful. SA = 200
 Com Port Ok.
 Com Port Ok.
 Com Port Ok.

SIMULATION

Selected PGN: 65027

SPN	DATA	UNIT
2441	<input type="text" value="43210"/> a8ca	V
2445	<input type="text" value="12345"/> 3039	V
2437	<input type="text" value="403"/> 193	Hz
2449	<input type="text" value="7890"/> 1ed2	A

Press Transmit Data after setting data

Figure 4.2.2.8 Entering Data for Simulation

11. To set the data to transmit to another J1939 node

After entering the desired data value, the user can set the data by clicking SET DATA button. Then the program will build a data frame based upon the set data and make it ready to transmit to receiver.

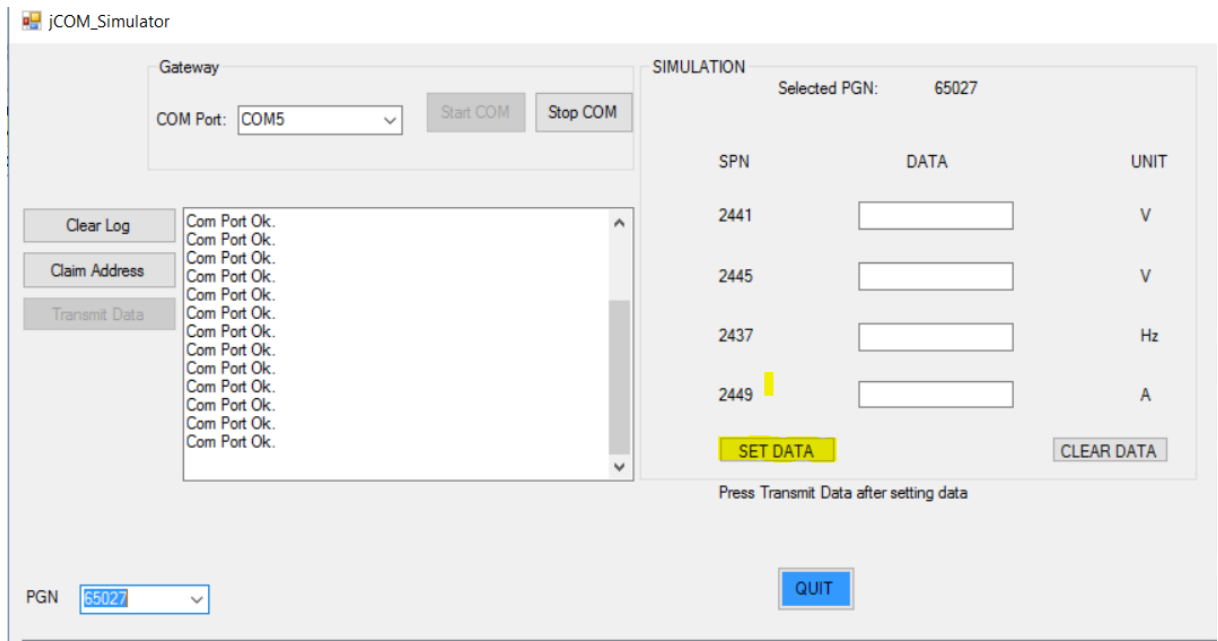


Figure 4.2.2.9 To set the data by clicking SET DATA

12. Clear the value entered for simulation

This button can be used to clear out the entered data in the Data field.

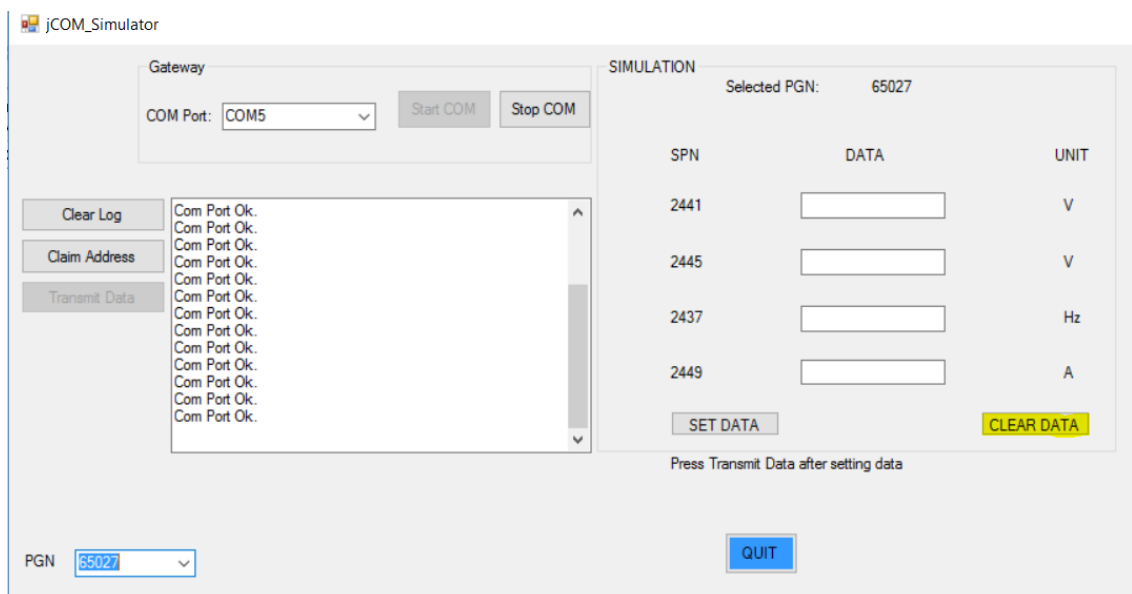
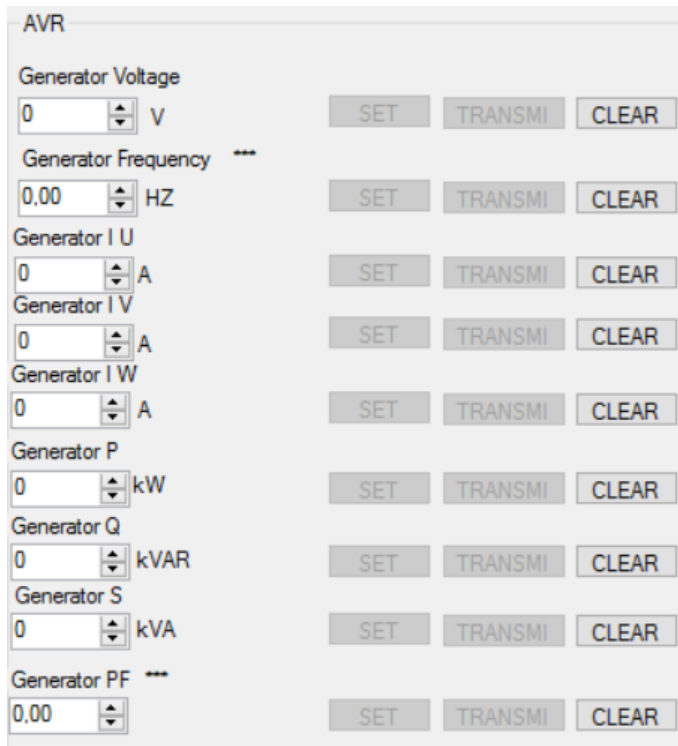


Figure 4.2.2.10 To clear the data by clicking CLEAR DATA

13. Transmit a SPN (Suspect Parameter Number) individually

In this section, it is programmed to be able to transmit SPN individually. The relevant PGN number is pre-programmed in the program. The user only needs to type in the desired data value to simulate.



The screenshot displays the AVR control interface with the following parameters and controls:

Parameter	Value	Unit	SET	TRANSMI	CLEAR
Generator Voltage	0	V	SET	TRANSMI	CLEAR
Generator Frequency ***	0.00	HZ	SET	TRANSMI	CLEAR
Generator I U	0	A	SET	TRANSMI	CLEAR
Generator I V	0	A	SET	TRANSMI	CLEAR
Generator I W	0	A	SET	TRANSMI	CLEAR
Generator P	0	kW	SET	TRANSMI	CLEAR
Generator Q	0	kVAR	SET	TRANSMI	CLEAR
Generator S	0	kVA	SET	TRANSMI	CLEAR
Generator PF ***	0.00		SET	TRANSMI	CLEAR

Figure 4.2.2.11 Transmit a SPN (Suspect Parameter Number) individually

14. Enter the desired value to transmit/simulate

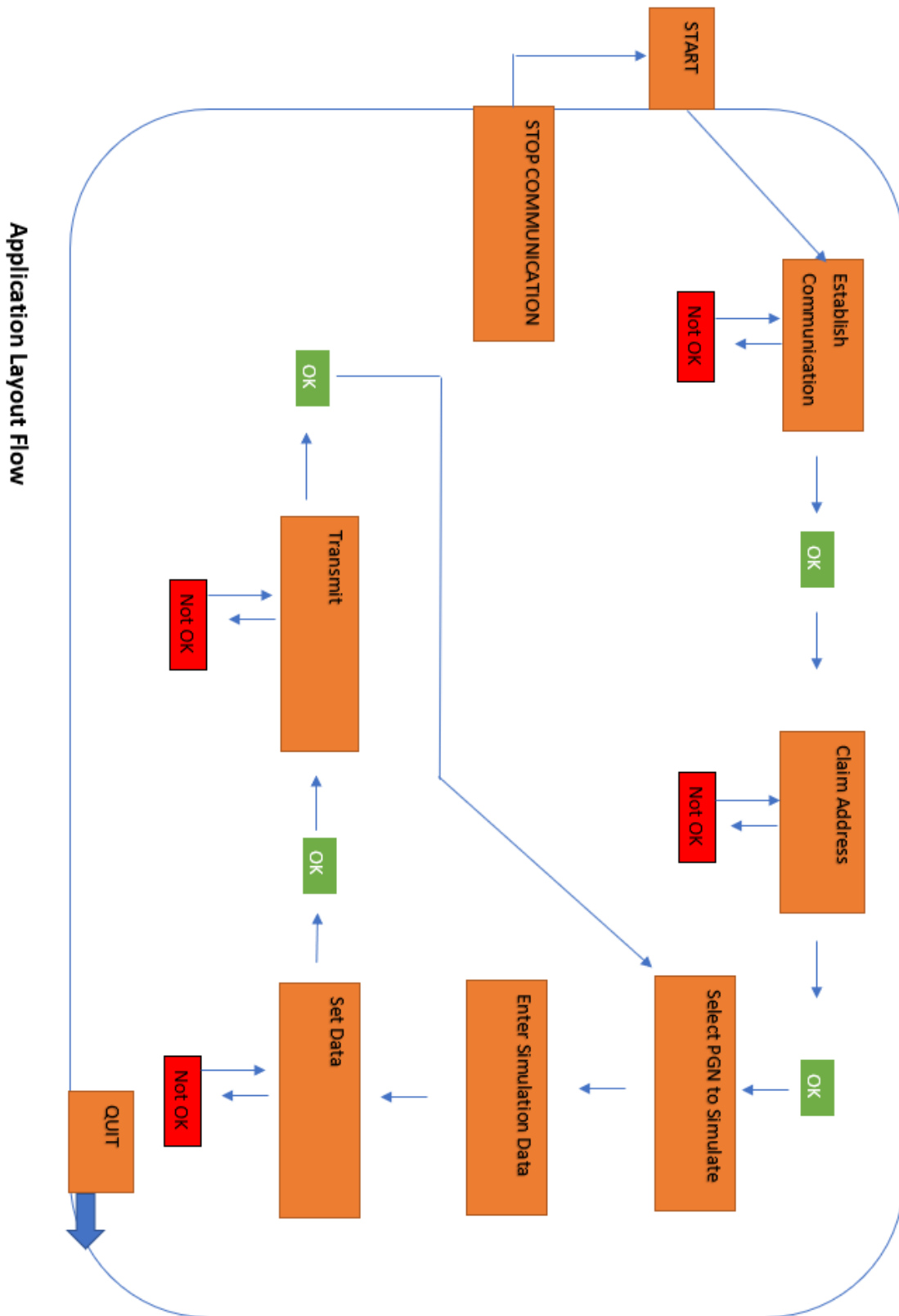
Desired data can be entered here for simulation.

15. Short description about this software.

This is a short description about this application.

16. To quit the program.

By clicking this button, the application will be shut down.



5 Testing

The testing of the generator simulation program was done periodically at the different phases of the project as this is an application design and development project. The jCOM.J1939 Starter Kit and Network Simulator from Copperhill Technologies makes the testing process a lot flexible. This kit includes two J1939 network nodes so that it is possible to test if the simulation data is correctly transferred from the sender and received at the receiver side. Otherwise it will be dependent to connect to the actual hardware setup with Moxa signal converter & PLC for testing purpose.

Generator simulation program is designed so that it is easy to identify that the transmitted data packages are correctly received by the receiver. It is established by requesting the receiver to transmit back the exact same package which it received to the transmitter. In this way, the transmitter will be able to check if the data packages are correctly received by the receiver.

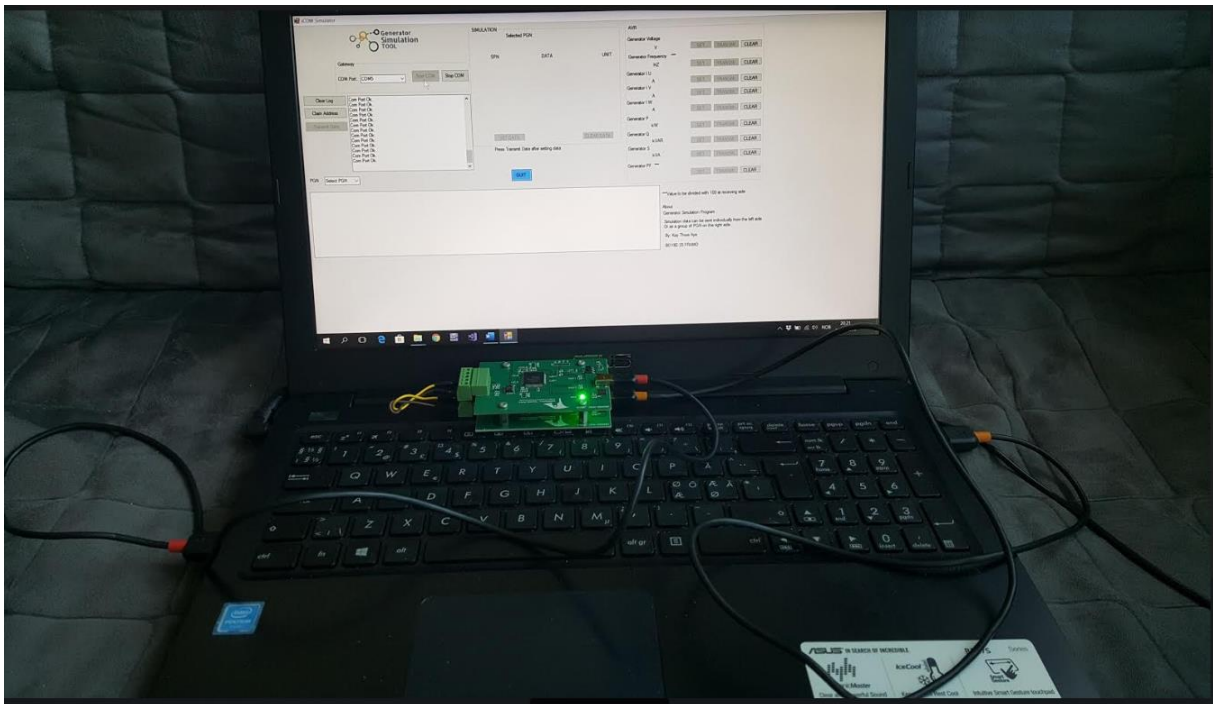


Figure 5.1 Test set up with jCOM.J1939 Starter Kit and Network Simulator

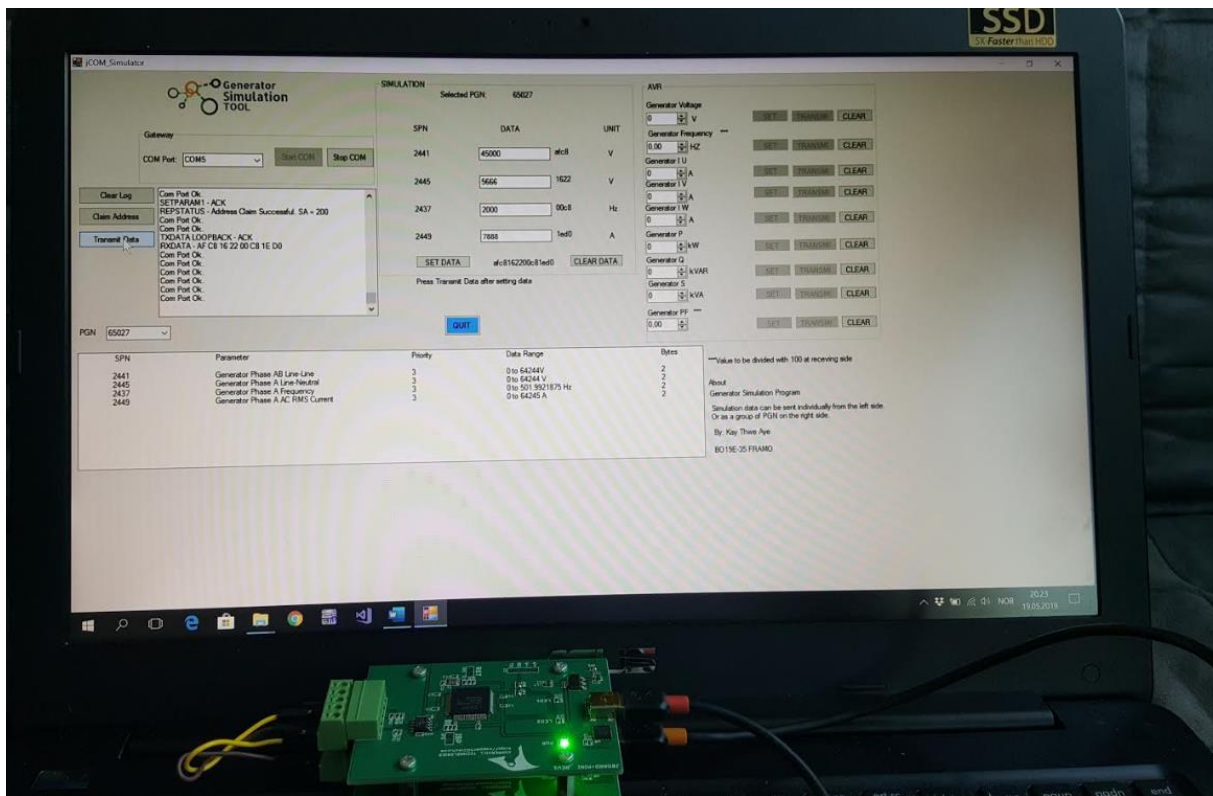


Figure 5.2 Test set up with jCOM.J1939 Starter Kit and Network Simulator

The contracting entity has also performed some tests on simulation program with Moxa signal converter and confirmed that the data packages are successfully received by the Moxa signal converter. Further implementation with PLC and HMI will be done in the near future by the contracting entity.

The purpose of this project is to be able to send Generator simulation data packages using J1939 protocol and it can conclude that the project specifications and requirements are fully achieved.

6 Discussion

The project went smoothly according to the original project plan which was made at the beginning of the project. It might be due to the great amount of effort which was put to split down the project activities at the beginning. All the activities (refer to the Gant diagram in Appendix) has been nearly accomplished according to the estimated hours and date.

As mentioned in chapter 3 of this report, two possible solution alternatives were identified. The choice of which solution alternative to be realized for this project is correctly made. Another solution alternative has uncertainties, risks and rate of failure is high.

After choosing the solution alternative, the great amount of time and effort were put on choosing the correct J1939 network simulation card to be used. In the market, there are different suppliers who supply this kind of simulation card with different features and set-up. The choice of picking the simulation card is correctly done and it made the design and developing of the simulation tool a lot smoother. The chosen simulation card comes along with the C# libraries with many function blocks which was relevant for this project. It saved a lot of time for writing the code for the simulation tool. If chosen the wrong simulation card from another supplier which does not delivered C# libraries, it will be needed a great amount of time and challenge to write the code for all the function.

There was still uncertainty at the beginning, if the Moxa signal converter on the Framo side would be able to receive the transmitted simulation data. After finishing the first revision of the simulation tool, a test was performed to check if it can establish communication section between simulation tool and Moxa signal converter. The test went smoothly and the road ahead with further developing of the simulation became clearer.

It was estimated major risks regarding failure of the simulation card and malfunctioning of the laptop which is used for developing of the simulation tool. The occurrence of these major risks were null. As these risks are identified and necessary action are already sorted out, there would not be major impact on the project schedule.

During the developing of the simulation tool, it was encountered 2/3 times when the program would not compile and throw an exception or error messages from Visual Studio. This also was not a challenge, as this is already identified in the risk list, the program was compiled often, and it made easier to find and fix the errors.

In short, correct project plan with activities, right component choice and the identified risks were the factors which made this project ran smoothly.

7 Conclusion

The main task of this project is to develop a functioning generator simulation which shall be able to transmit simulation data frames to PLC where the Framo project software for the firewater pump is located. The data frames shall be transmitted using J1939 communication protocol which generators normally use to exchange input and output data frames. A list of desired generator simulation data for simulation was provided by Framo, refer to Appendix D PGN & SPN Information, to be implemented in the generator simulation tool. In addition, the simulation tool shall be made so that it is flexible for further development.

The project is accomplished according to the requirements and specifications which is discussed and settled at the beginning of the project. A functioning simulation tool is developed using C# programming, J1939 communication protocol is used to transmit data frames and all the desired simulation data are implemented in the simulation tool.

The project activities, refer to Appendix C, is followed accordingly throughout the project. No major changes were made to project plan throughout the project.

The generator simulation tool can be developed further in the future for simulation of a diesel engine. The basic features for J1939 communication methods and functions used for generator can be re-used for diesel engine simulation tool which will certainly make the further development easier.

Appendix A. Literature

- [1] Source: <https://www.framo.com/About-Framo/>
Internet, last checked 15.05.2019
- [2] Fire Pump Arrangements at Industrial Facilities, Third Edition by Dennis P. Nolan, 2017
- [3] Source: <https://www.framo.com/oil-and-gas-pumping-systems/products/fire-water-pumps-electric/>
Internet, last checked 18.05.2019
- [4] Source : <https://copperhilltech.com/sae-j1939-starter-kit-and-network-simulator/>
Internet, last checked 10.05.2019
- [5] Authoritative Dictionary of Standards Terms (7th ed.), *IEEE*, 2000, *ISBN 978-0-7381-2601-2*, Std. 100.
- [6] Source: <https://www.dieselgeneratortech.com/generators/what-is-synchronous-generator.html>
Internet, last checked 10.05.2019
- [7] Introduction to electrical power and power electronics , ISBN 978-1-4665-5661-4 by Mukund R. Patel , 2013
- [8] A Comprehensible Guide to J1939 by Copperhill Media Corporation By Wilfried Voss ISBN: 978-0-9765116-3-2 , 2008
- [9] A Comprehensible Guide to J1939, *page 5* by Copperhill Media Corporation By Wilfried Voss ISBN: 978-0-9765116-3-2 , 2008
- [10] A Comprehensible Guide to J1939, *page 20* by Copperhill Media Corporation By Wilfried Voss ISBN: 978-0-9765116-3-2 , 2008
- [11] Source: <https://instrumentationforum.com/t/major-components-of-plc/5715>
Internet, last checked 10.05.2019
- [12] Source: <https://www.inductiveautomation.com/resources/article/what-is-hmi>
Internet, last checked 10.05.2019

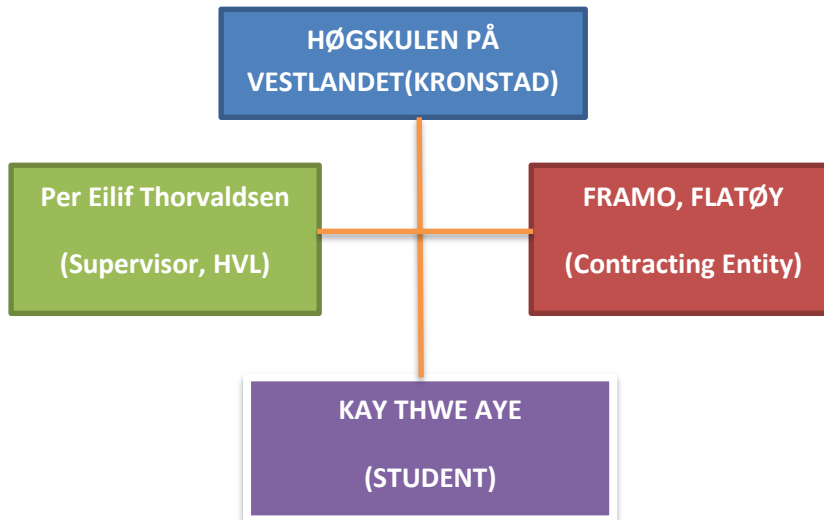
Appendix B. Abbreviations and dictionary explanations

PLC:	Programmable Logic Controller
HMI:	Human Machine Interface
FWP:	Firewater Pump
HVAC:	Heating, ventilation, and air conditioning
GUI:	Graphical User Interface
AVR:	Automatic Voltage Regulator
I/O:	Input Output
AC:	Alternative Current
kVAR:	Reactive Power (unit)
kW:	Active Power (Unit)
SAE:	Society of Automotive Engineers
PGN:	Parameter Group Number
SPN:	Suspect Parameter Number

Appendix C. Project management

This project is an individual project. All the project management and project realization is done only by Kay Thwe Aye.

A.1 Project organization



Project organization consists of one supervisor from Høgskulen på Vestlandet and two supervisors from Framo Flatøy.

A.2 Progress plan

The project was accomplished according to the planned schedule set at the beginning. One activity “Testing of the simulation tool on previous delivered project” was removed ,as the contracting entity does not have possibility to perform this.

BO19E-35 Framo

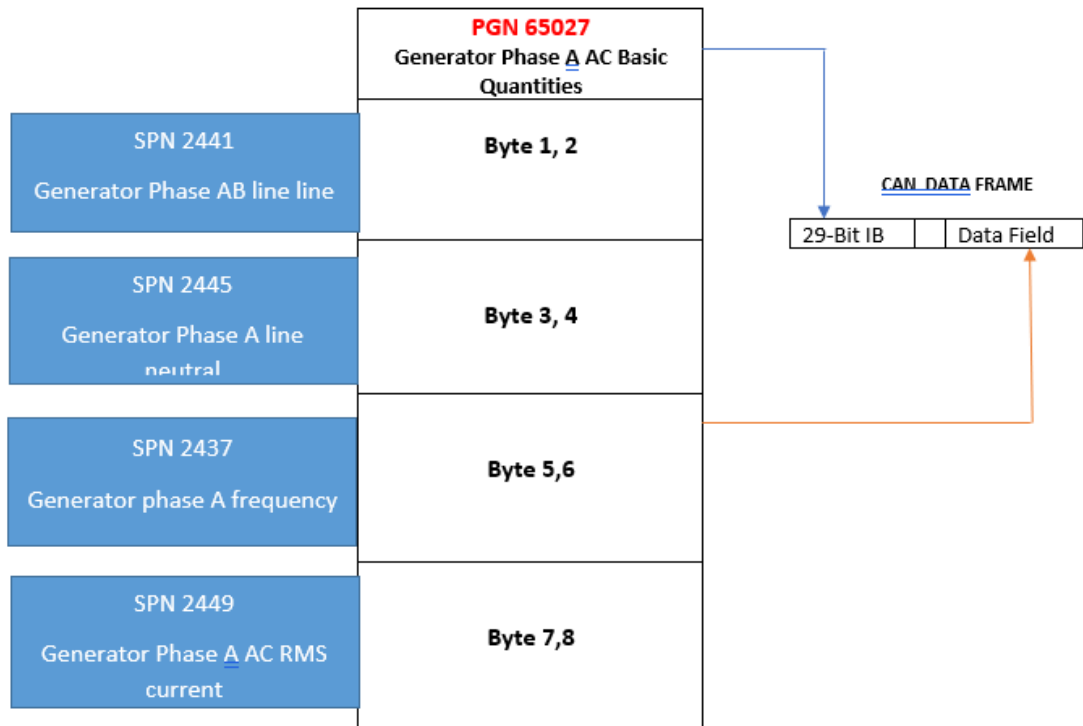
#	Aktivitet	Planl. Arbeid [timer]	Utført Arbeid [timer]	Start Dato	Slutt Dato	Fram drift	Ansvarlig	Uke	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12	w13	w14	w15	w16	w17	w18	w19	w20	w21	w22	w23	w24	
								Man	7/1	14/1	21/1	28/1	4/2	11/2	18/2	25/2	4/3	11/3	18/3	25/3	1/4	8/4	15/4	22/4	29/4	6/5	13/5	20/5	27/5	3/6	10/6	
								Fre	11/1	18/1	25/1	1/2	8/2	15/2	22/2	1/3	8/3	15/3	22/3	29/3	5/4	12/4	19/4	26/4	3/5	10/5	17/5	24/5	31/5	7/6	14/6	
																						Påske										
1	Arbeid med forstudie	100	75	1/1	31/1	100 %	KAY					◆																				
2	Innlevering av forstudie	0			31/1	100 %	KAY					◆																				
3	Studere/Set up Hardware	50	65	1/2	28/2	100 %	KAY								◆																	
4	Studere Software Utviklingspråk	50	35	10/2	28/2	100 %	KAY									◆																
5	Definere/Analysere scope (Detaljert)	30	35	20/2	15/3	100 %	KAY										◆															
6	Utvikling av Software	120	135	1/3	10/4	100 %	KAY											◆														
7	Testing av Software hos Framo	40	25	15/3	30/4	100 %	KAY												◆													
8	Forbredelse av Midtveis presentasjon	15	10	15/3	30/3	100 %	KAY																									
9	Midtveis presentasjon	0		1/4	5/4	100 %	KAY																									
10	Ferdigstilling av software	30	35	15/4	30/4	100 %	KAY																									
12	Bachelor rapport	120	115	1/3	30/5	100 %	KAY																									
13	Bachelor oppgave innlevering	0		31/5		100 %	KAY																									
14	Bachelor oppgave presentasjon	0		3/6	12/6	100 %	KAY																									
15	EXPO	0		13/6		100 %	KAY																									
		555	530																													

A.3 Risk list

Following is the identified list of the risks.

Risk	Chance of occurrence	Criticality	Consequence	Action
Longterm sickness	Low	Medium	Delay of project	Perform the most critical part of the project early
Malfunction of J1939 card	Low	Low	Extra cost	Follow safety procedure when handling the card
Malfunction of developing laptop	Low	High	Extra cost	Have daily backup of the project files
Unexpected bugs in the simulation tool	Low	High	Delay of project	Test & Debug the simulation tool often

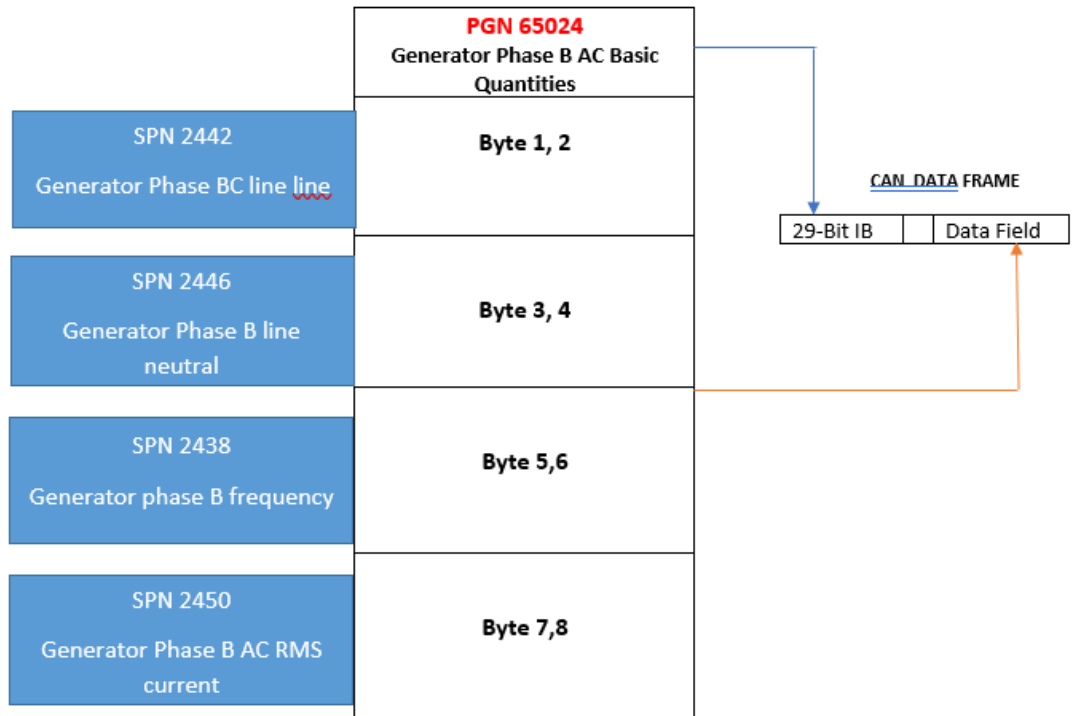
Appendix D. PGN & SPN Information



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Phase AB line line	3	FE03	65027	2441	2	1V/bit	0	0 to 64244V
Generator Phase A line neutral	3	FE03	65027	2445	2	1V/bit	0	0 to 64244V
Generator phase A frequency	3	FE03	65027	2437	2	1/128Hz/bit	0	0 to 501.9921875 Hz
Generator Phase A AC RMS current	3	FE03	65027	2449	2	1A/bit	0	0 to 64255A

PGN 65027 & SPN information

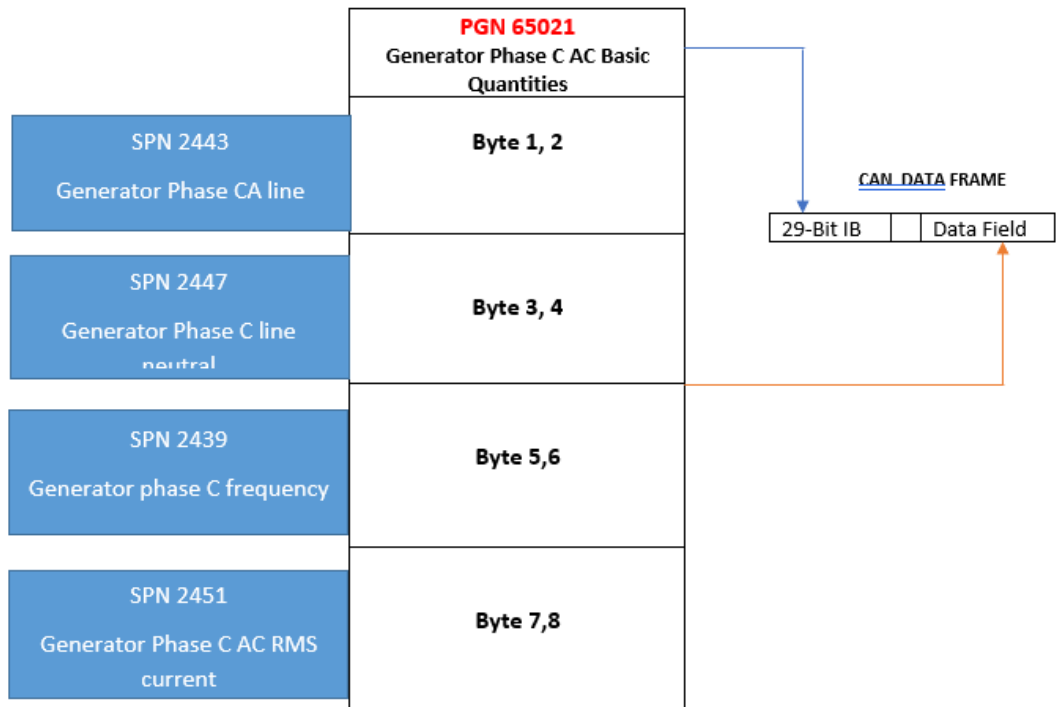
PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Phase BC line line	3	FE00	65024	2442	2	1V/bit	0	0 to 64244V
Generator Phase B line neutral	3	FE00	65024	2446	2	1V/bit	0	0 to 64244V
Generator phase B frequency	3	FE00	65024	2438	2	1/128Hz/bit	0	0 to 501.9921875 Hz
Generator Phase B AC RMS current	3	FE00	65024	2450	2	1A/bit	0	0 to 64255A

PGN 65024 & SPN information

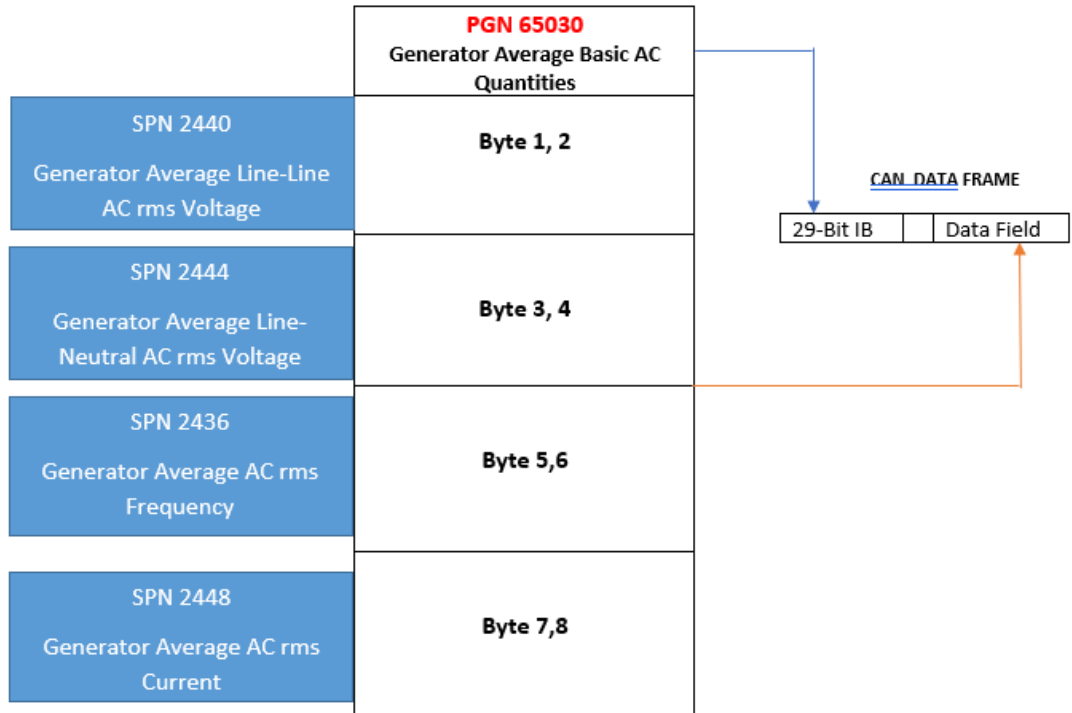
PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Phase CA line line	3	FDFD	65021	2443	2	1V/bit	0	0 to 64244V
Generator Phase C line neutral	3	FDFD	65021	2447	2	1V/bit	0	0 to 64244V
Generator phase C frequency	3	FDFD	65021	2439	2	1/128Hz/bit	0	0 to 501.9921875 Hz
Generator Phase C AC RMS current	3	FDFD	65021	2451	2	1A/bit	0	0 to 64255A

PGN 65021 & SPN information

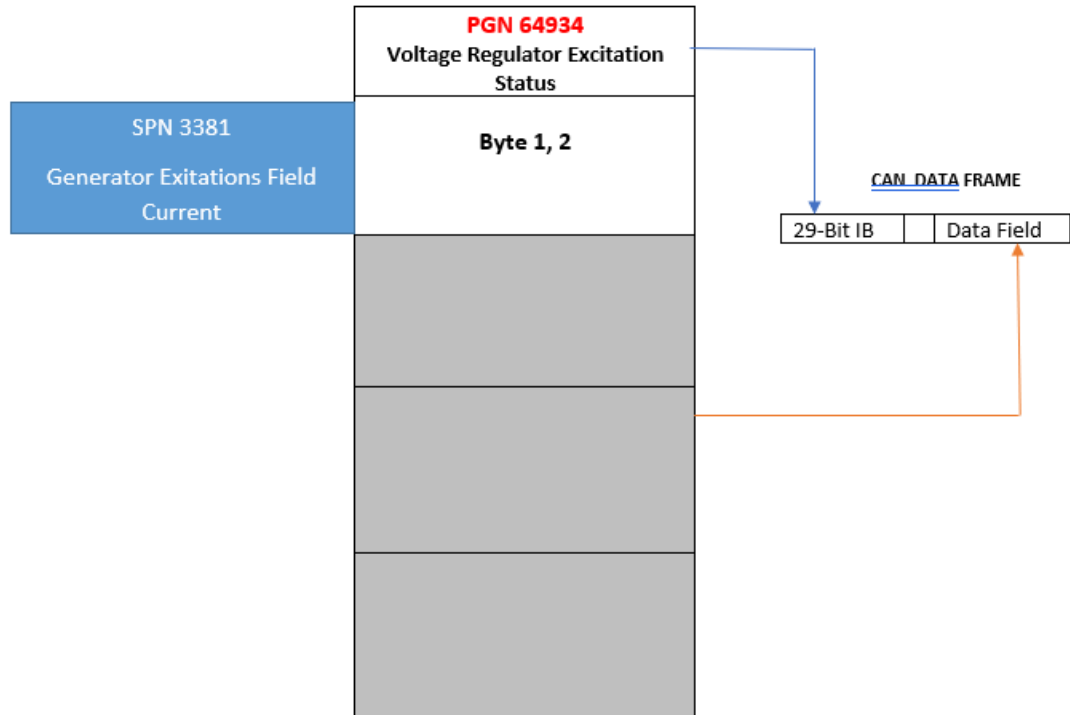
PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Average Line-Line AC rms Voltage	3	FE06	65030	2440	2	1V/bit	0	0 to 64244V
Generator Average Line-Neutral AC rms Voltage	3	FE06	65030	2444	2	1V/bit	0	0 to 64244V
Generator Average AC rms Frequency	3	FE06	65030	2436	2	1/128Hz/bit	0	0 to 501.9921875 Hz
Generator Average AC rms Current	3	FE06	65030	2448	2	1A/bit	0	0 to 64255A

PGN 65030 & SPN information

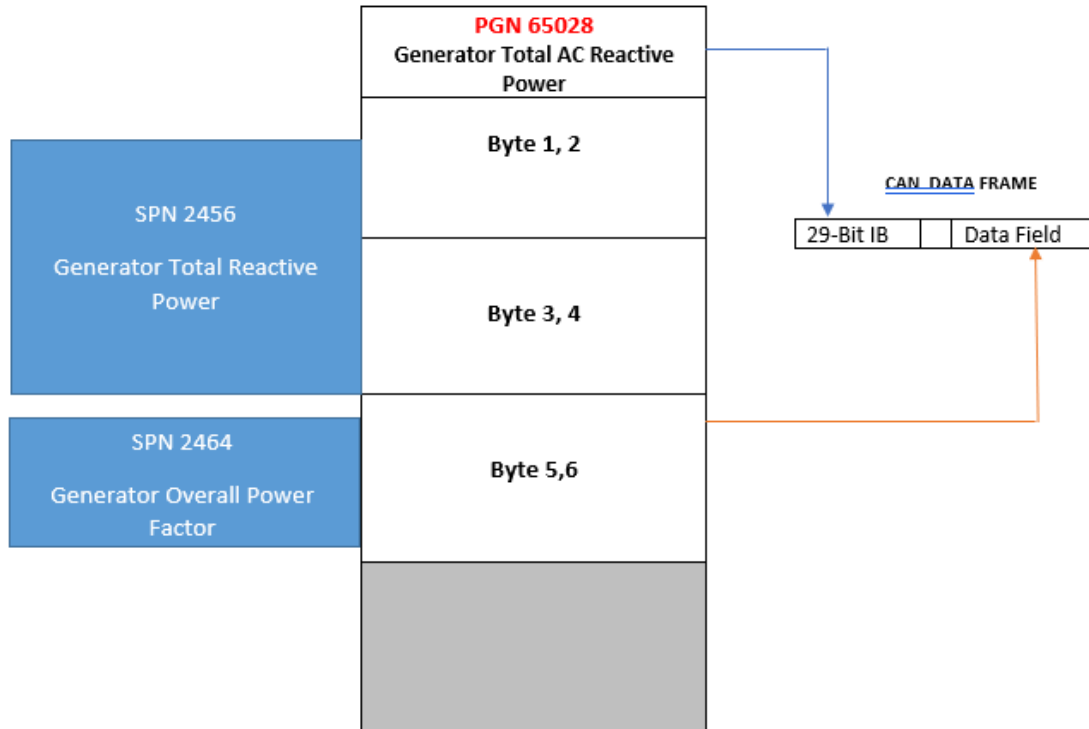
PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Exitations Field Current	3	FDA6	64934	3381	2	0,05A/bit	0	0 to 3212.75A

PGN 64934 & SPN information

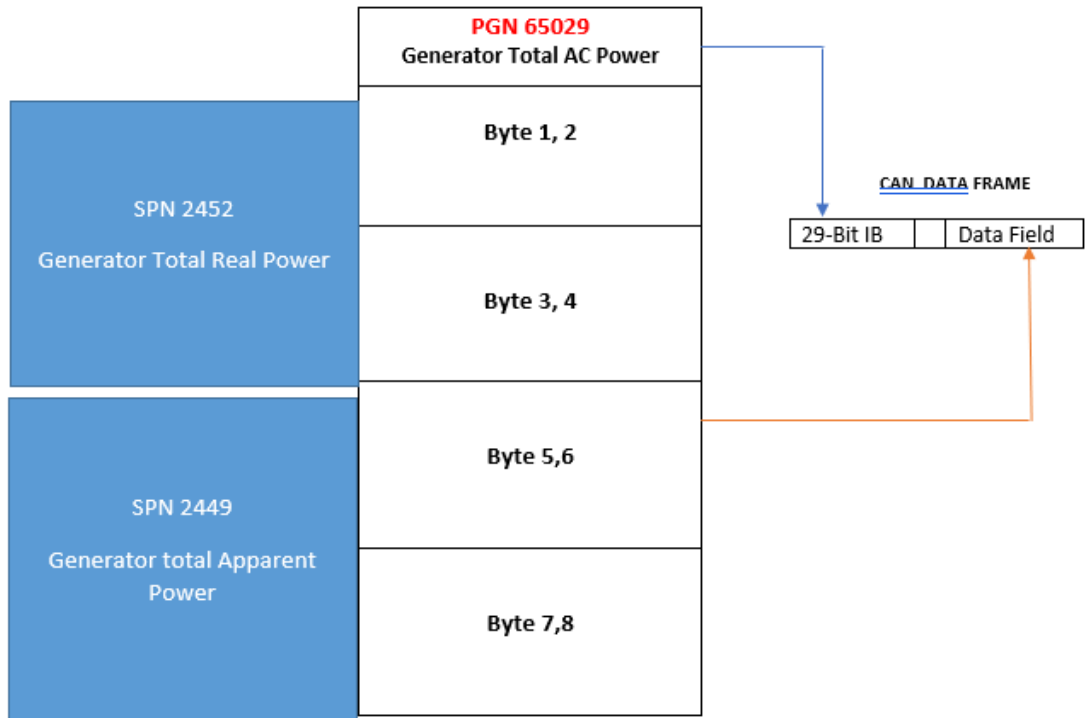
PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Total Reactive Power	3	FE04	65028	2456	4	1Var/bit	-2 000 000 000	-2000 000 000 to 2211 081 215 Var
Generator Overall Power Factor	3	FE04	65028	2464	2	1/16384 / bit	-1	-1.00000 to 2.921814

PGN 65028 & SPN information

PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers



Parameters	Priority	PGN Hex	PGN	SPN	Bytes	Resolution	Offset	Data Range
Generator Total Real Power	3	FE05	65029	2452	4	1W/bit	-2 000 000 000	-2000 000 000 to 2211 081 215 W
Generator total Apparent Power	3	FE05	65029	2460	4	1VA/bit	-2 000 000 000	-2000 000 000 to 2211 081 215 W

PGN 65029 & SPN information

PGN = Parameter Group Numbers SPN = Suspect Parameter Numbers

Appendix E. Source code, Bill of Materials etc.

Source Code

The source code is delivered as zipped file.

Bill of Material

Type	Manufacturer	Model Code/Title	Cost
Software	Microsoft	Visual Studio 2017	Student License
Hardware	Copperhill Technologies	jCOM.J1939 Starter Kit and Network Simulator	USD 129,95
Literature	Copperhill Media Corporation	A Comprehensible Guide to J1939 by Wilfried Voss ISBN-13: 978-0976511632	USD 15.57

Appendix F. Midtveis rapport (for information)

STATUS REPORT 03/04/2019

Kay Thwe Aye

Project Vision

To create a generator simulation tool so that software testing can be performed independent of the delivery time of generator

Project Health

GOOD. Excellent feedback from Framo on testing of the current version of simulation program. Set milestones achieved so far, as agreed with Framo.

Arbeid med forstudie	100%	Innlevering av forstudie	100%
Studere/Set up Hardware	100%	Studere Software Utviklingsspråk	100%
Definere/Analysere scope (Detaljert)	100%	Utvikling av Software	55%
Testing av Software hos Framo	35%	Forberedelsen av Midtveis presentasjon	100%
Midtveis presentasjon	100%	Ferdigstilling av software	0%
Testing mot tidligere leverte utstyr hos Framo	0%	Bachelor rapport	15%

Completed so far

About 60% of the generator signals which Framo would like to be tested with their PLC software, are implemented and tested fine.

What to achieve before delivery?

- Implement AVR on the simulation program (next milestone). To be discussed in detail with Framo in the next meeting
- Fix the bugs in the program
- Final project report

Risks /Issues

- The number of signals to be implement for A.V.R function in simulation program and complexity unknown yet