**CSI OT 3D Platform Cyber Attack Demonstration Design Document**

**[Under Editing]**

Date: 15/12/2020

VERSION: CORPLAB-2019-T3.1-P1

**Contents**

**[Under Editing]**

**Introduction**

This project will implement an HMI for the OT-City simulator platform and power management simulator platform (generator and substation). Then we will demonstrate different cyber-attack situation on the OT-City simulator platform. The project contains four section:

**OT-City Simulator HMI Program**

In this section we will create a SCADA HMI system with Schneider Wonderware(R) program running on the SCADA PC for the user to control the OT-city simulator modules for the training or research purpose. The system control function is implemented by changing the output coils’ status of 3 PLC (Schneider M221 X2 + Siemens S7-1200 PLC X1).

**OT Platform Cyber Attack Simulation**

In this section we will demonstrate 3 different kinds of cyber-attack situation on the OT-platform and power management module: False data injection attack, Blackout attack and the Stealthy situation attack. The attack demonstration will be activated by the attack control website and launched from an attack control device (Raspberry PI) which connect to the system network. The influence of different attack situation will be introduced in the section “Cyber attack implementation”.

**OT-Cyber-Attack Control Website**

In this section we will create a website server running on the orchestrator PC to provide a web interface to let the user control different cyber-attack demos on OT-platform and show current system feedback/attack detail information during the presentation. The user will active/stop the attack demo by login the attack control webpage.

**Power Generator Manager**

We will provide a user interface running on the SCADA PC to remote control the OT-Power Generator Module. We will provide a module controller made by one Raspberry PI and an Arduino to receive the control request from the remove controller UI and change the state of the hardware components of Power Generator Module such as Pump, moto and LED display panel. The control program will also do the automatically adjustment of the generator's motor and pump speed based on the loads in the system.

**System hardware connection and IP address configuration**

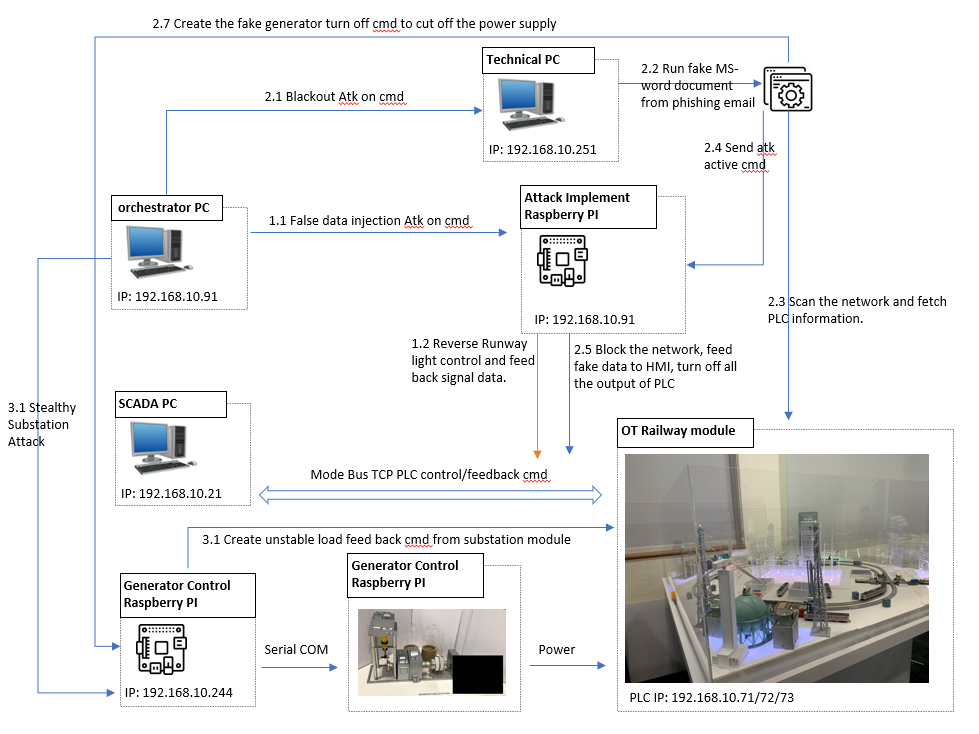
**Hardware connection diagram**:

**[Under Editing]**

Hardware list: **[Under Editing]**

|  |  |  |
| --- | --- | --- |
| Idx | Name | Function |
| 0 | SCADA PC |  |
| 1 | Technical PC |  |
| 2 | Orchestrator PC |  |
| 3 | Attack Rasp |  |
| 4 | Pwr Rasp |  |
| 5 | Pwr Arduino |  |
| 6 | Network switch |  |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |

**IP address configuration diagram**: **IP address configuration diagram**

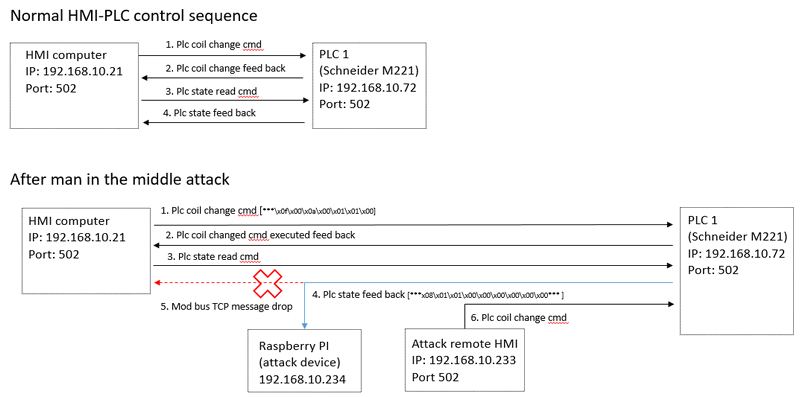


**Cyber Attack Simulation**

**False Data Injection Attack**

In this attack, we assume an additional foreign hardware (IoT/Raspberry PI) was plug in to the OT network. This attack will manipulate the SCADA command and feedback. When the attack device activated the attack, it will scan the Modbus TCP communication message between SCADA HMI and M221 PLC, find the related control bytes arrays and inert/replace the pre-saved fake data, which will causes the SCADA HMI to show the opposite feedback on the actual system. This demo will attack on airport light control, where the operator will see reverse PLC operation control and feedback on the actual system, e.g. When the operator tries to turn on the runway lights in the airport via HMI, the actual runway lights will be turn off. The attack control data flow is shown in the <IP address configuration diagram > section 1.x.

Attack Logic diagram:



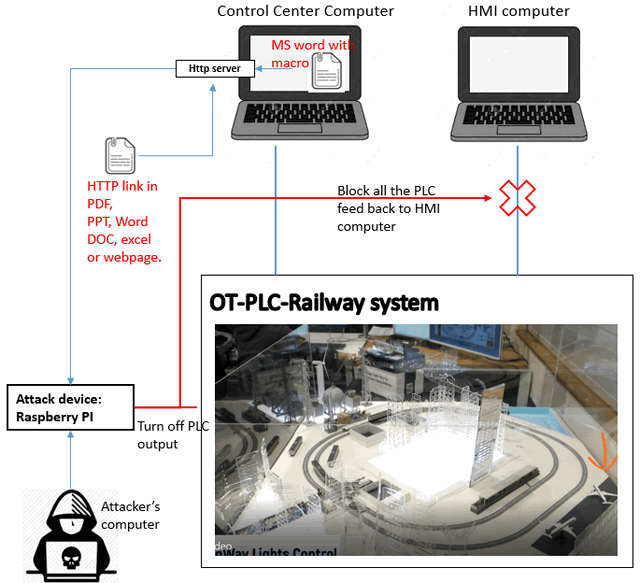
Attack scenario view:

**[Under Editing]**

**Blackout Attack**

This attack is model after 2015 Ukraine power grids cyber-attack. This attack will assume the system do not properly air-gapped, whereby the malware is entered to the system via spear phishing email. When the attack launched, the technical PC will simulate a careless user/worker who opened an un-authorized document from the phishing email on his technical PC which has connected in the OT-Platform network. The un-authorized document which contents the malware will scan and find the PLC, then block the PLC. control communication from SCADA PC(HMI), then take over the control of the PLC. Our attack device will also simulate the attack has takeover the control of All three plc and try to force turn off All the PLC output coils (energy output). **[Under Editing]**

Attack Logic diagram:



Attack scenario view:

**[Under Editing]**

**Stealthy situation attack:**

Technically, known as False Command Injection attack. [ **[Under Editing]** Attack introduction will follow Shantanu ‘s paper]. In this attack we will simulate how the attack try to break control system of the substation to generate the stealthy PWR load changes which will make influence of the power generator and make parts of the OT system paralysis. (Railway track-A, Train station and airport.)

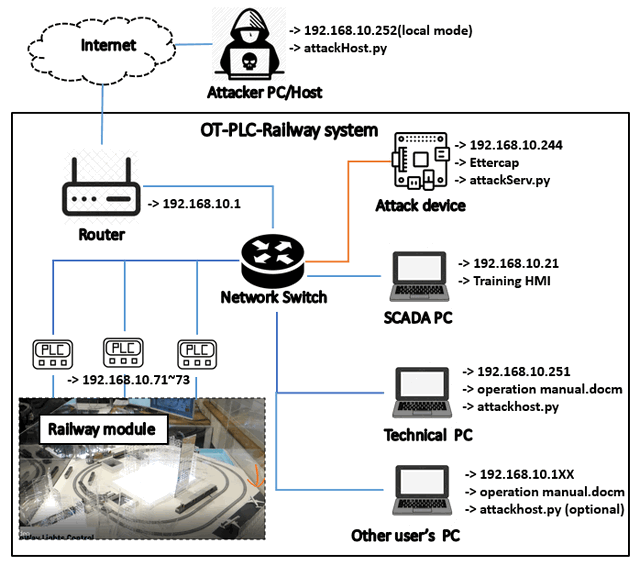
Attack Logic diagram: **[Under Editing]**

Attack scenario view:

**[Under Editing]**

**Steps to Implement The Attack Demo:**

1. Connect the attack device (A Raspberry PI with the attack server program) to the OT-PLC-Railway network system by a CAT-5 cable.
2. Attack situation I : Attacker can use a remote attack control panel to turn off all the PLC output coils without detectable by the train supervisory control and data acquisition (SCADA) system.
3. Attack situation II : When a normal user open a MS-Word document, a document edit enable message box will pop up. If the confirm button was clicked, the attack server will turn off all the PLC output coils.
4. Attack situation III : when a normal user click the "PLC detail menu" hyper-link in a MS-Word document the attack start.
5. Attack situation IV: Connected the attack PC to the system from internet/local Ethernet. Start the attack from the attack control Web interface.

**Attack file Running diagram:** 

**[Under Editing]**

#### **Attack Program Setup**

###### Development Environment: Python 2.7 & python 3.7 + HTML5

###### Additional Lib/Software Need:

1. snap7 + python-snap7 (need to install for S71200 PLC control)

Install instruction:   
http://simplyautomationized.blogspot.com/2014/12/raspberry-pi-getting-data-from-s7-1200.html

1. Ettercap-graphical (need to install on Raspberry PI to do the attack)

$ sudo apt-get update -y  
$ sudo apt-get install -y ettercap-graphical

1. Flask (need to install on the attakerPC to show the attack control web)

pip install Flask

###### Hardware Needed:

Raspberry PI Mode 3 B+ with Ettercap installed and IP set:

$ ifconfig eth0 192.168.10.244 netmask 255.255.255.0 up

###### Program File List:

| **Program File** | **Execution Env** | **Description** |
| --- | --- | --- |
| attckBlackE3.py | python2.7/python3 | This module will be called from the macro in doc "operation manual.docm" to simulation the black out attack on the OT-PLC-platform. |
| attackhost.py | python2.7/python3 | This module is used to create a http server on port 8080 to handle the attack get request. |
| attackServ.py | python3 | This module will create a attack service program to run the Ettercap false data injection attack. |
| controlPanel.py | python2.7/python3 | This module will create attack control panel to start and stop the man in the middle attack. |
| M2PLC221.py | python2.7/python3 | This module is used to connect the Schneider M2xx PLC. |
| S7PLC1200.py | python3 | This module is used to connect the siemens s7-1200 PLC |
| m221\_1 filter/m221\_1.ef | C | This filter is used do reverse all the PLC communication command between HMI and the PLC1. ( 192.168.10.21<=> 192.168.10.72) |
| m221\_3 filter/m221\_2.ef | C | This filter is used to do block all the PLC feedback data to the HMI computer.(192.168.10.21) |
| operation manual.docm | MS word/VBA | MS-Word document with Macro to active the attack. |
| attackWeb/attackHost.py | python3 | flask server to create a attack control web for the people who does the presentation. |

#### **Program Usage/ Demo Implement**

All the program has to mode: test mode and demo mode. The demo is used to do the demo on the real PLC-Railway System and under test mode all the programs can run on one computer without do any setting or connect to any hardware.

To active test mode set the test flag TEST\_MODE = True # Test mode flag - True: test on local computer

###### Setup Attack Device

* + Set the IP address and copy the file base on the section "Program file deployment for the demo".
  + Install Ettercap in Raspberry PI and compile the filter to executable file for the Ettercap :

$ etterfilter m221\_3.filter -o m221\_3.ef

* + Plug the Raspberry Pi in the OT-PLC-Railway system and setup the IP address:

$ sudo ifconfig eth0 192.168.10.244 netmask 255.255.255.0 up

* + Copy the file **m221\_1.ef**, **m221\_3.ef** and **attackServ.py** in to the same folder and run the attack server:

$ sudo python3 attackServ.py

Better use sudo to avoid permission deney error.

###### Start/Stop Black Out Attack From Attack Control Panel

* + Make should the client computer is also connected in the OT-PLC-Railway system and its IP address is set to same subnet(192.168.10.XXX), run/double click the **controlPanel.py** :
  + Click the "**Connect Serv**" button to test the server connection, the text field will show "**C;1**" to identify the server response server connection successful.
  + Click the "**Active attack**" button to start the attack, the server will response "**A;1**" when it started the attack. Click the "**Stop attack**" button and the when sever response "**A;0**" the attack will be stopped.

###### Start Black Out Attack From the MS-Word Document's Macro

* + Open the word document and press "Alt+F11" to edit the macro execution path. Replace the attackBlackE3.py path in the follow line which shown in the VBA editor:

Shell "cmd /c C:\Python27\python.exe C:\Users\dcslyc\Documents\attackBlackE3.py " & sDir, vbNormalFocus

If there is a space in the path need to put "" at the front and end of the path, for example :

Shell "cmd /c C:\Python27\python.exe ""C:\Users\Liu Yuancheng\Documents\attackBlackE3.py"" " & sDir, vbNormalFocus

* + Run the **attackhost.py** on the client computer with the word document.
  + Double click the MS-Word **operation manual.docm** : when the document is opened, an editing enable message box will be show. Click the "**OK**" button.

The scan command window will pop-up and show the computer scan information.

An executable file will be created with the same folder of the MS-Word document as shown below. Then the attack will be start automatically.

If the file was put in the 'Desktop', the file may not be created as the "exe" file creation permission is not high enough.

###### Start Black Out Attack From the Document's Hyper-link

* + Run the **attackhost.py** on the client computer with the word/PDF document.
  + Click the hyper-link and the attack will be start when the bowser pop-up and shows "405 file not found error".

###### Start/Stop Black Out Attack/ False Data Injection Attack From Web Control Panel

* + Run the attackHost.py in the attackWeb folder:

python attackHost.py

* + Type in url <http://127.0.0.1:5000/> (local) <http://xxx.xxx.xx.xx:5000/> (remote) or in the browser. the web will show:
  + Press the "Start/Stop attack" button in the first line to active/de-active Black Out attack.
  + Press the "Start/Stop attack" button in the second line to active/de-active the False Data Injection Attack.

**Attack Control Website**

Hardware Needed: Orchestrator PC

Development Environment: NodeJs(v12.18.4)/JavaScript HTML5

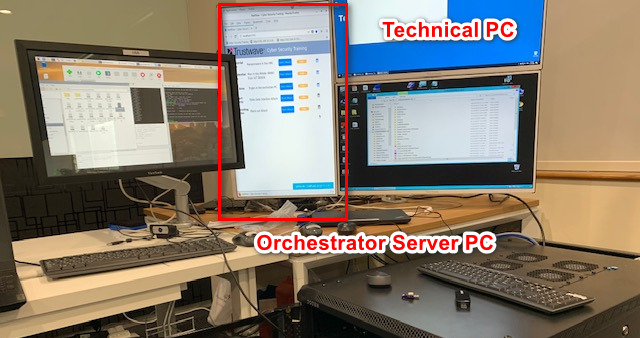
Additional Lib/Software Need: NodeJs(v12.18.4)

**Webpage View**

Open web browser and type in URL: <http://localhost:8080> or <http://127.0.0.1:8080>.



**Login the orchestrator PC**



**Attack Control Logic**

When the user pressed the “Start” attack button, the web host will send the attack activation command to the related agent running on technical PC or the attack device. (Communication use the asynchronous UDP)

**[Under Editing]**

False Data Injection Attack:

The attack server sends the attack active cmd ‘Atk:A:2’ to the attack device to start the attack or ‘Atk:A:0’ to stop the attack. The attack control data flow is shown in the <IP address configuration diagram 1.x > section

Black Out Attack:

The attack server send the attack active cmd ‘Atk:A:1’ to the technical PC agent(actionHost.py) to open the MS-Word document with macro setup, then the attack program will run and block the network communication between PLC and the HMI program. Then the attack program on the technical PC will send attack active cmd ‘Atk:A:1’ to the attack device to do the PLC take over control. The web will send the cmd ‘Atk:A:0’ to attack device stop the attack and release the network ModBus TCP communication block.

The attack control data flow is shown in the <IP address configuration diagram 2.x > section

Stealthy situation attack

The attack server will send the the attack active cmd ‘Atk:A:3’ to the power generator manager controller(raspberry PI) to active the attack. **[Under Editing]**

The attack control data flow is shown in the <IP address configuration diagram 3.x > section

**Program File List and Execution**

Execution: In the "server" folder, run cmd:

$ sudo Node app.js

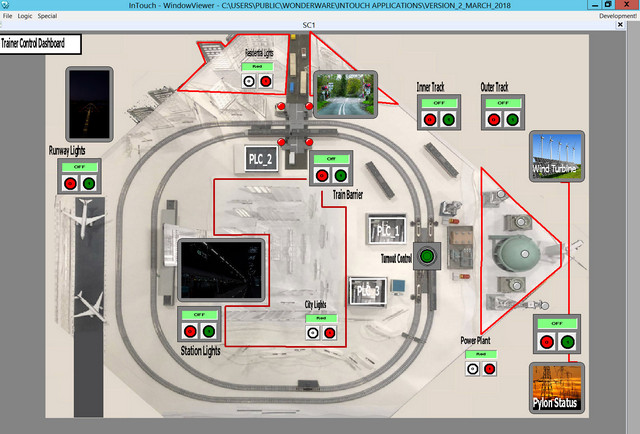
Open a web browser and type in link: http://localhost:8080/

| **Program File** | **Execution Env** | **Description** |
| --- | --- | --- |
| node\_modules | Node.js(JavaScript) | All the additional module which need to import in the app.js |
| public | HTML, CSS, JavaScript | The main web page interface. |
| app.js | Node.js(JavaScript) | Main server program. |
| httpserver.service | sh | Auto run setup when running on Linux Platform. |
| playAlert.sh | sh | Play the alert sound when the attack happens. |
| runServer.sh | sh | Run the app.js and removed the duplicate running if found. |

**Schneider Wonderware HMI**

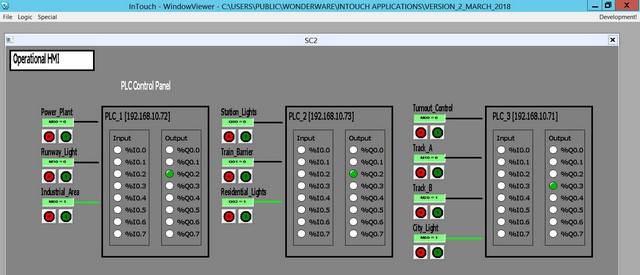
The main HMI User interface contains 3 pages:

1. Main Control HMI



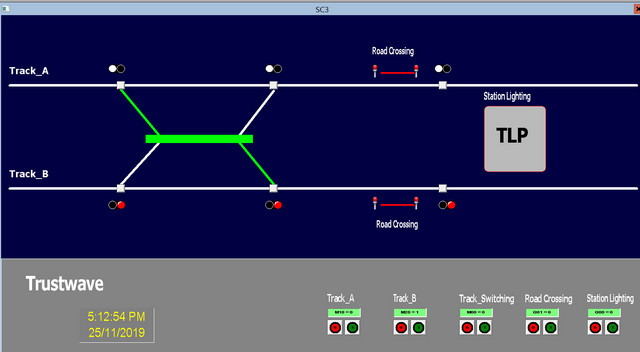
This page is used to control all the components in the OT-Platform. **[Under Editing]**

PLC Control Window



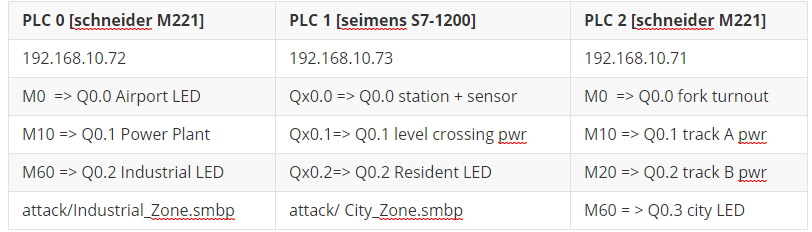
This page is used to show the 3 PLC states and the relation ship of the components with the PLC’s output coils.

RailWay State Window



This Page is used to simulate the Real Railway Control HMI system for training the railway operator.

PLC Memory mapping table:



**Program File List:**

| **Program File** | **Execution Env** | **Description** |
| --- | --- | --- |
| src/M2PLC221.py | python 2.7/3 | This module is used to connect the Schneider M2xx PLC. |
| src/railwayAgentPLC.py | python 3 | This module is the agent module to init different items in the railway system or create the interface to connect to the hardware. |
| src/railwayGlobal.py | python 3 | This module is used as the local config file to set constants, global parameters which will be used in the other modules. |
| src/railwayHub.py | python 3 | This function is used to create a rail control hub to show the different situation of the cyber-security attack's influence for the railway HMI and PLC system. |
| src/railwayMgr.py | python 3 | This function is the railway function manager to connect the agent element with their control panel. |
| src/railWayPanel.py | python 3 | This module is used to provide different function panels for the rail way hub function. |
| src/railWayPanelMap.py | python 3 | This module is used to show the top view of the main city map in the railway system. |
| src/ S7PLC1200.py | python 3 | This module is used to connect the siemens s7-1200 PLC |
| attack/ City\_Zone.smbp | Schneider Wonderware IDE | City Zone PLC ladder diagram. |
| attack/Industrial\_Zone.smbp | Schneider Wonderware IDE | Industrial Zome PLC ladder diagram. |

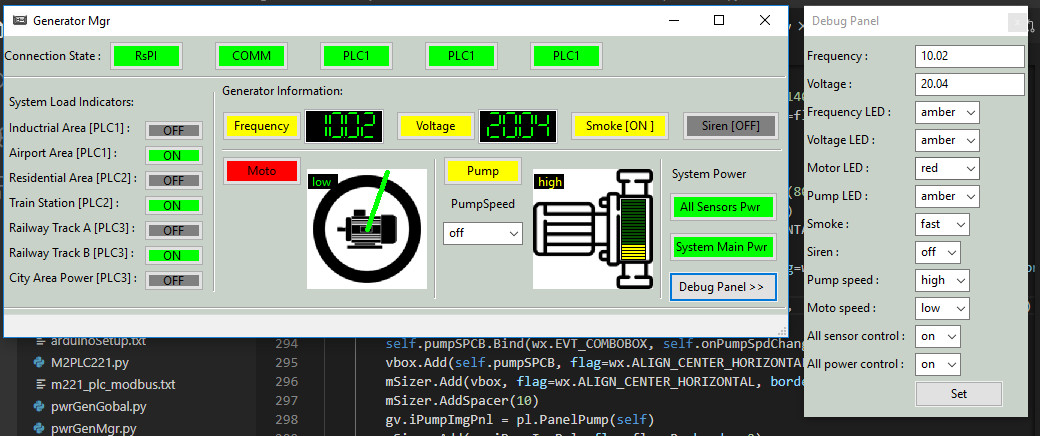
**Power Generator Manager**

This project contains two sections:

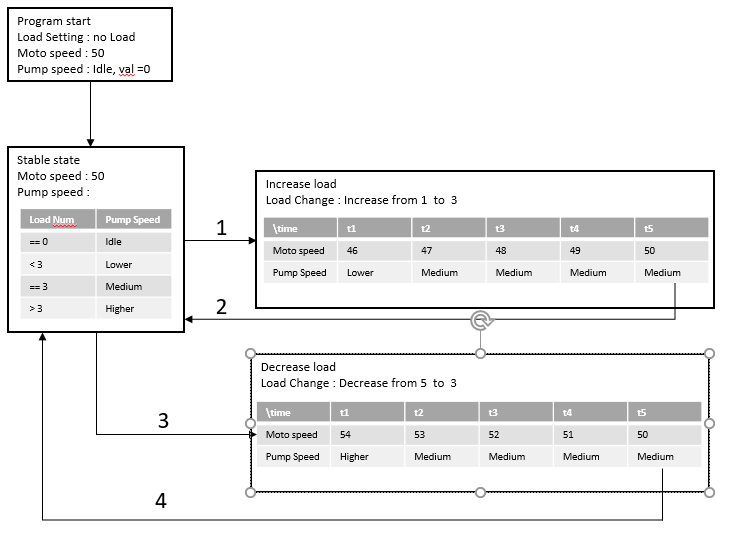
**Power generator auto-control program**: This module will be running in the Raspberry PI to control the OT-Power Generator Module's hardware. It will send command to PLC and Arduino and receive the control request from the remove controller. The control program will do the adjust of the generator's motor and pump speed based on the loads in the system.

**Remote generator controller:** This module will provide a UI to connect to the power generator auto control program by UDP and display the generator states.

Power generator UI view:



Load automatically adjustment match table [ Airport, Tran station, Track 1. Track 2 ]



#### **Program Setup.**

1. Set the test mode flag TEST\_MD to False, set the IP address in pwGenRun.py to the Raspberry PI's IP.
2. In Raspberry PI run cmd:

python3 pwrGenMgr.py

1. In the control computer, run cmd:

python pwrGenRun.py

###### Program File List:

| **Program File** | **Execution Env** | **Description** |
| --- | --- | --- |
| M2PLC221.py | python 2/3 | This module is used to connect to the Schneider M2xx PLC. |
| pwrGenGlobal.py | python 3 | This module is used as a local config file to set constants, global parameters which will be used in the other modules. |
| pwrGenMgr.py | python 3 | Power generator auto-control manager. |
| pwrGenPanel.py | python 3 | This module is used to create different function panels. |
| pwGenRun.py | python 3 | This module is used to create the control panel to connect to the Raspberry PI generator control by UDP. |
| S1new\_192\_168\_10\_72.smbp | PLC | PLC-1 ladder diagram. |
| S3new\_192\_168\_10\_71.smbp | PLC | PLC-3 ladder diagram. |
| S7PLC1200.py | python 3 | his module is used to connect to the siemens s7-1200 PLC. |
| serialCom.py | python 3 | This module will inheritance the python built-in serial module with automatically serial port serach and connection function. |
| udpCom.py | python 3 | This module will provide a UDP client and server communication API. |