Development of a Management and Control System for Active and Reactive Power in the DeMoTec Laboratory

by: P.W. van de Sande and M. Au-yeung
EPS.08.S.376

De faculteit Elektrotechniek van de Technische Universiteit Eindhoven
aanvaardt geen verantwoordelijkheid voor de inhoud van stage- en afstudeerverslagen

Stagewerk verricht o.l.v.:
Dr.ir. J.M.A. Myrzik
19 mei 2008

/ faculteit elektrotechniek
Development of a Management and Control System for Active and Reactive Power in the DeMoTec Laboratory

by P.W. van de Sande

Internship Report

Name: P. W. van de Sande
Student number: 0537488

Project Period: December 2007 – April 2008
Submission date: April 17, 2008

Supervisor ISET: M. Braun
Supervisor TU/e: J.M.A. Myrzik
Abstract
In an electricity grid with a growing amount of distributed generators, the control of active and reactive power is becoming more important. Therefore a management and control system for active and reactive power has been developed. It consists of a server and several clients. The server can read PQ profiles for each device and set the target values for P and Q into a database. The client controls can read these target values from the database and set them to the device. Client controls for a load workshop, a 200 kVA biodiesel genset and a 20 kVA speed variable genset have been developed in Labview. On the server the measurements from every DG unit in the system can be read. An experiment shows that the SCADA system makes it possible to control the active and reactive power flow in a small grid.
Appendix G  List of Abbreviations ........................................ G-1
Appendix H  References ...................................................... H-1
1 Introduction

In the coming years, the share of distributed generation (DG) units in the electricity grid will continue to grow [1]. The implementation of this new kind of generators brings the need for a new kind of power control in the mains network.

One way of active and reactive power control in the electricity grid is with the use of PQ profiles. These profiles specify for each DG unit the active and reactive power that should be generated at a specific point in time. This means that the stability of the grid can be controlled and through cost-optimizing functions the efficiency of the grid can be improved.

At the Design Center for Modular Supply Technology (DeMoTec), several DG units are available. To test the behavior of the DG units and the grid under the control conditions, a control system for P and Q has to be developed. It should consist of a central server that can send target values for active and reactive power to the DG units and loads. The server should read the target values for active and reactive power from a PQ profile. Furthermore each generator and load needs a controller to receive the target values from the server and set them to the generator or load.

This report describes the development of such a system. It starts with an overview of the SCADA system in chapter 2. The infrastructure and controller of the system are explained, and information about the used PQ profiles is given.

The next chapter focuses on the load workshop that is available in the DeMoTec laboratory. This workshop consists of a controller and several relays that are connected to loads.

In chapter 4 the control for a 200 kVA Biodiesel genset is explained. Besides normal diesel, this generator can run on vegetable fuels such as biodiesel or vegetable oil.

The control of a 20 kVA Speed Variable genset is discussed in chapter 5. This small diesel generator can run variable speed to save fuel and costs.

In chapter 6 an experiment using the developed control system is explained. Chapter 7 deals with the development of a web-based user interface for a service and equipment database. This database contains information on DG devices from several European research institutes.

Finally, in chapter 8 a conclusion is given.

Appendix A shows an overview of the created functions in Labview for each of the controllers. An instruction on how the set-up and manage the database is given in Appendix B. How to operate the controls of the control systems can be found in Appendix C. Appendix D gives an overview of the first version of the workshop controller.
2 The SCADA system

This chapter focuses on the development of a SCADA system for the DeMoTec laboratory. The system can control the active and reactive power set points of the connected generators and loads. First a short introduction about the SCADA system is given. After that the created infrastructure is explained. The third section of this chapter will give more information about the PQ profiles. This section is followed by the explanation of the Labview control program that was developed for the SCADA system. The last two sections present the results of the developed system and the conclusion and recommendations.

2.1 Introduction

A SCADA (Supervisory Control And Data Acquisition) system can be seen as a distributed control and measurement system. SCADA systems are typically used in large industrial environments where many processes are controlled and monitored. It consists of a server and several clients. The first task of the server is to provide the clients with set points for the process they control. The second task is to retrieve the measurement values from the clients and store these for later analysis.

In the DeMoTec laboratory each client will control a generator or a load. The purpose of the SCADA system is to provide P and Q control for all the connected devices. The target values for a specific device are stored in PQ profiles. A PQ profile contains information about at which time which target value of active power P and reactive power Q has to be set.

2.2 Infrastructure

In Figure 2-1 the layout of the control system can be seen. At the top of the figure is the server. Aside from a MySQL database this computer runs the Labview SCADA controller, which is discussed in section 2.4. This controller can read target values for P and Q from PQ profiles and put these in the database.

Below the server are the different clients. The client computers are connected to the server via an Ethernet network. They can read the target values from and write the measurement values to the database on the server. The clients have different ways of connecting with the devices they control.
2.2.1 Server

The server is the controlling part of the SCADA system. It consists of two parts. First it runs a MySQL server. The MySQL server was installed using XAMPP [2]. This is a user-friendly package which includes an Apache web server with MySQL and PHP support. Administration of the database can be done with the PHPmyAdmin program which also runs on the server. See Appendix B for more information on this subject. The database that is set up consists of several tables, one for each device that needs to be controlled. Each table has the same amount of columns. The table for the 200 kVA biodiesel generator is shown in Table 2-1.

<table>
<thead>
<tr>
<th>pset</th>
<th>qset</th>
<th>pmeas</th>
<th>qmeas</th>
<th>smean</th>
<th>vmeas</th>
<th>fmeas</th>
<th>Pfmeas</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>500</td>
<td>1100</td>
<td>450</td>
<td>1188</td>
<td>399</td>
<td>49</td>
<td>0.88</td>
</tr>
</tbody>
</table>

Table 2-1: Example of table in database

The table has entries for the P and Q set values, and for the P, Q, S, V, frequency, and power factor measurement values. The set values are written to the database by the Labview control program while the measurement values are written to the database by the control programs running on the clients.

The second part of the server is the Labview Control program. This program is described in detail in section 2.4.
2.2.2 Clients

The client computers are normal windows workstations which can run the Labview control programs developed for the device they control. They are on the same Ethernet network as the server and have access to the database. Currently four client systems are set up. The first one is for controlling the 100 kVA Multi-PV via a custom protocol over a standard serial link. The second client controls the 15kVA genset. This client is connected to a USB Data Acquisition (DAQ) device which provides the analogue signals required for the control of the genset. For a detailed explanation of these two controls please see the report of J. Au-Yeung [3]. The third client is for controlling the 200 kVA biodiesel generator. This generator can be controlled by using the MODBUS protocol over a standard serial connection. Detailed information about the control for this unit is available in section 4.2. The next client is for controlling both the load workshop and the 20 kVA diesel generator. These devices can communicate via XML/RPC, but since Labview has no support for this protocol an OPC server is used. This OPC server translates the XML/RPC so the data can be accessed in Labview. Both the devices have their own OPC server. More information about the Labview control for the workshop can be found in chapter 3, for more information about the control for the 20 kVA diesel generator consult chapter 5.

2.3 PQ Profiles

To control the active and reactive power set points of the equipment in the laboratory, the target values for P and Q are stored on the server. This is done in a PQ Profile. A PQ profile is a chronological collection of target values for P and Q together with a time (in seconds) at which the values should be active.

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Active Power (W)</th>
<th>Reactive Power (VAR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1000</td>
<td>500</td>
</tr>
<tr>
<td>10</td>
<td>2000</td>
<td>400</td>
</tr>
<tr>
<td>20</td>
<td>3000</td>
<td>200</td>
</tr>
<tr>
<td>25</td>
<td>4000</td>
<td>300</td>
</tr>
<tr>
<td>35</td>
<td>5000</td>
<td>400</td>
</tr>
<tr>
<td>40</td>
<td>4000</td>
<td>200</td>
</tr>
<tr>
<td>60</td>
<td>3000</td>
<td>300</td>
</tr>
<tr>
<td>70</td>
<td>4000</td>
<td>200</td>
</tr>
<tr>
<td>80</td>
<td>4000</td>
<td>200</td>
</tr>
</tbody>
</table>

Table 2-2: Example of PQ profile

Table 2-2 shows an example of a load profile. The first column indicates the time in seconds, the second column the active power in Watt and the third column the reactive power in VAR. Therefore this example says that from zero to 10 seconds the active power is set to 1000 W and the reactive power to 500 VAR. From 10 seconds until 20 seconds the active power is 2000 W and the reactive power 400 VAR, etc.
The time in the last row of the profile denotes the end time and the P and Q set values belonging to it are only used when the controller runs longer than the maximum time specified in the PQ profile. To allow easy creation and modification of the load profiles they are stored as CSV (Comma Separated Value) files. The semi-colon (;) is used to discriminate between values.

2.4 Labview SCADA Controller

The Labview SCADA control running on the server has three main objectives. The first objective is to set the correct P and Q values in the database, at the time specified in the load profile. This needs to be done for each active device. Secondly it has to retrieve the measurement values that are written to the database by the clients and store these values. The last objective is to present a clear user interface where the information on the status of all the devices is presented.

Therefore the program consists of two main loops. One for setting the target values to the database and one for reading the measurement values from the database. These loops run in parallel and are described in the next two sections.

But before these loops are entered, an initialization is done. During this phase, the first phase of the program, the connection with the database is opened. Furthermore the specified load profiles are read and the names of the devices are extracted from the filenames of the load profiles. Also the target values for all the active devices are set to zero, so they can be started safely.

2.4.1 Setting P and Q target values to the database

When the initialization of the program is done, the ‘set values’ loop starts. It consists of a while loop that runs until the stop button is pressed. Figure 2-2 shows the flowchart of the set loop.

First for each device it is checked if it is active or not. If not active the program moves on to the next device. Otherwise a check is made whether the device is in automatic or manual mode.

If the device is in manual mode the entered value is looked up and compared with the current value. If they differ the new value is written to the database, if they are the same the loop moves on to the next device.

If the device is in automatic mode, then the elapsed time is checked and compared with the time at the next row of the PQ profile. A counter keeps track of which row of the PQ profile is currently active. If the elapsed time is bigger than the time specified at the next row, then a new value needs to be set. This new value is read from the load profile, the counter is moved to the next row and the new value is written to the database.
The implementation of this control loop in Labview is shown in Figure 2-3.

Figure 2-2: Flow chart of 'set values' loop

Figure 2-3: 'Set values' loop block diagram, database mode is active
The diagram consists of one ‘for’ loop and three case structures. This ‘for’ loop is the same loop as shown in the flow diagram. On the left-hand side the required data enters the loop. The first case structure encountered when moving from left to right looks whether the device is active. If the device isn’t active it enters the false case and moves on to the next device. If the device is active then the ‘true’ case is entered. Now a second case structure looks if the device is in manual mode or in database mode. In the figure is shown what happens when database mode is active. First the PQ profile is indexed at the current row. The time value in this row is then multiplied with the time step multiplier and compared with the elapsed time. If the elapsed time is bigger then the time from the PQ profile, and the counter position is not equal to the maximum counter value, then the new P and Q values are set to the database.

When the manual mode is selected, the second case structure from the outside will go to the ‘False’ case. This case is displayed in Figure 2-4. Now the target values are read from the front panel and set to the database.

When the end of all the load profiles is reached, two things can happen according to the choice of the user. The program can either stop or repeat the PQ profile from the beginning.

![Diagram](image)

**Figure 2-4:** ‘Set values’ loop block diagram, manual mode is active

### 2.4.2 Reading measured values from the database

The loop that is responsible for reading the measured values from the database and showing them on-screen runs in parallel with the ‘set values’ loop described above.

This part of the Labview control is depicted in Figure 2-5. Again the outer rectangle is a ‘for’ loop that executes for every device. Inputs on the left-hand side are the name of the device and the names of the measurement values, the reference link to the database and a Boolean value that indicates whether the device is active. If the device is active, the ‘Get Measurement Values’ function is called. One argument of this function is the table name from where the data should be read. It then returns the
data as an array. The first four values, which are P and Q target and P and Q measured, are then shown in a graph on the front panel. The rest of the data is shown numerically. The measured values are also put into an array together with the time the values were measured. The loop executes five times per second, so the values are saved with time steps of 200 milliseconds. The array can be written to a file at the end of the program. In Table 2-3 the layout of the columns in the measurement files is shown.

<table>
<thead>
<tr>
<th>time(s)</th>
<th>P target (W)</th>
<th>Q target (W)</th>
<th>P measured (W)</th>
<th>Q measured (W)</th>
<th>S measured (VA)</th>
<th>V measured (V)</th>
<th>f measured (Hz)</th>
<th>Power factor measured (cos(phi))</th>
</tr>
</thead>
</table>

Table 2-3: Layout of CSV measurement file

![Figure 2-5: Reading measurement values](image)

2.4.3 Graphical User Interface

The interface of the SCADA control gives the user a possibility to set load profiles to certain devices or enter set values manually. It also displays the current state of the system. The interface is shown in Figure 2-6. The top part of the interface consists of a control panel. Here the user can specify a link to the database and start or stop the program. To the right of these buttons are two radio buttons. Here the user can specify what needs to happen when the load profiles are completed. The top option stops the system when the PQ profiles are done, and the bottom option lets the PQ profile repeat from the beginning. Finally at the top-right corner a time step multiplier can be specified. This number specifies how long one time step in the PQ profile should take. When set to one, the time in the PQ profile will be in seconds. But when set to 60, the time of
the PQ profile will be in minutes. This makes it possible to use the same PQ profile for a short or long experiment. Below this control panel are the panels for all the devices in the system. At the left the name of the device and an on/off button is given. This button determines whether the devices should be activated or not.

Moving to the right is a section with controls for the device. Each device has three controls. At the top is a path control where a PQ profile can be specified. Below this is a switch that can be in database or manual mode. Manual means that the target values can be entered by the user while in automatic mode the target values are retrieved from the specified PQ profile. The manual values can be entered in the boxes to the right of the switch. Moving further to the right, the current measured values are displayed. These include the active, reactive and apparent power, frequency, voltage and power factor. On the rightmost block a graph with the current and set values for P and Q is displayed. At the top of the graphs a legend shows which parameter has which color.

2.5 Results
The SCADA system described above is capable of controlling P and Q for several kinds of equipment. It is possible to specify PQ profile files that
contain multiple target values for P and Q and the time at which they should be active. The values in the database are updated at a rate of maximum five times per second. This means that there can be a 200 ms difference between the time in the PQ profile and the time the new variables are set.

Seven different generators and loads can be controlled. The measurement values of each device are visible on the display and can be saved as a CSV file for later analysis.

2.6 Conclusion and recommendations

A SCADA system has been developed that is able to control and monitor the active and reactive power flow in a grid. It consists of a server and several clients. The server runs a MySQL database and a Labview controller. The controller can read PQ profiles which contain target values for active and reactive power for all the DG units in the network. The P and Q target values are then set to the database at the time specified in the PQ profile. On the front panel the measurement values of the different generators and loads in the system are shown. It is also possible to save the data for later analysis.

The system can be improved by implementing event structures instead of 'while loops' for updating the target values in the database. It would then require less system resources and be more accurate.
3 The Load Workshop

In this chapter the controller for the Load Workshop is explained. Two versions of PQ controllers for the workshop are created. The first version is based on the load profile storage function of the Netmaster and reads the load profile directly from the database. An explanation of this controller is available in Appendix D. Because of problems with the speed of the controller (see section Appendix D.6 for more details) a new version is created. This version does not use the load profile storage and can read P values directly from the database and set the relays accordingly. First an introduction about the load workshop is given. After that the Labview controller is explained. Finally a conclusion and recommendations are given.

3.1 Introduction

To test the distributed generators in DeMoTec, and to see how they would behave in the real world they have to be connected to some kind of load. Therefore a workbench with different kinds of loads has been set up in DeMoTec. This workbench consists of a Remote Terminal Unit (RTU), switchable electrical outlets and several loads. A picture of the workshop is shown in Figure 3-1. The connected loads and their required power are given in Table 3-1.

<table>
<thead>
<tr>
<th>Relay</th>
<th>Load</th>
<th>Power [W]</th>
<th>Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relay 1</td>
<td>Lamp 1</td>
<td>500</td>
<td>I</td>
</tr>
<tr>
<td>Relay 2</td>
<td>Lamp 2</td>
<td>500</td>
<td>II</td>
</tr>
<tr>
<td>Relay 3</td>
<td>Lamp 3</td>
<td>500</td>
<td>III</td>
</tr>
<tr>
<td>Relay 4</td>
<td>Lamp 4</td>
<td>500</td>
<td>I</td>
</tr>
<tr>
<td>Relay 5</td>
<td>Lamp 5</td>
<td>500</td>
<td>II</td>
</tr>
<tr>
<td>Relay 6</td>
<td>Lamp 6</td>
<td>500</td>
<td>III</td>
</tr>
<tr>
<td>Relay 7</td>
<td>Not connected</td>
<td>0</td>
<td>I or II or III</td>
</tr>
<tr>
<td>Relay 8</td>
<td>Not connected</td>
<td>0</td>
<td>3-phase</td>
</tr>
</tbody>
</table>

Table 3-1: Available loads in the workshop

The RTU consists of a Elsist Netmaster programmable controller based on a Dallas Tini module. It has an Ethernet interface that allows it to communicate in a network. Possible protocols are FTP, Telnet and XML-RPC. The Netmaster has eight digital outputs. Seven of these are connected to a 1 phase 16 A power relay, and one is connected to a 3 phase 16 A power relay. The relays are connected to the different loads. For each relay there are three different options. It can be in on mode, off mode or in auto mode. When in auto mode, the relays are controlled by the load profile. The load profile specifies when which relay has to switch on or off. An example of a load profile can be seen in Table 3-2. Each row in this table can be regarded as a state. The leftmost column determines at which time the transition to the next state will be made.
Table 3-2: example load profile

The relay columns decide whether the relay should be on (1) or off (0). When the last row in the table is reached, the load profile will start again from the beginning. The maximum number of rows is limited by 100. The load profile is saved as a comma separated value (CSV) file. This file can be transferred to the Netmaster via ftp or telnet.

The global control method for the load workshop is depicted in Figure 3-2.

The connection to the workshop is made with the XML-RPC protocol. An OPC server running on the controlling computer translates the XML-RPC
into OPC. The controller runs in Labview, and can connect to the OPC server and a SQL database.

3.2 Control

The first controller designed for the Load Workshop was too slow. To set a relay to a new value takes up to 30 seconds and this is unacceptable. Therefore a new version of the workshop control is developed. This version does not need the load profile to be stored on the Netmaster. Thus the load profile analyzing function can be removed from the Netmaster and this makes the reaction time up to 20 seconds per relay faster.

The main function of the new program consists of three phases. First the connection with the database and de OPC server is made. In the second phase the reading of the target values and the writing to the OPC variables is done. This phase is entered as soon as the start button is pressed, and stops when the stop button is pressed. The third and last phase closes the connection to the database and the OPC variables.

3.2.1 Retrieving and processing the set values

The target value for the active power of the load workshop is stored in the database. This value is translated to an array of Boolean values that indicate whether a relay should be on or off. The extraction of the set point for P from the database is shown in Figure 3-3.

![Figure 3-3: Reading P Q set values from database](image)

The conversion is done in the 'PQ to Boolean' function. This function creates the Boolean array associated with the required active power set point. In Figure 3-4 this function is shown.
The first six relays connect to loads of 500 W. Therefore the target active power is divided by 500 and rounded to the nearest lower integer. This number tells how many relays are switched to the on position. A for loop with as many iterations as relays that need to be switched then creates the Boolean array.

### 3.2.2 Setting the array with OPC

The relays can be controlled via an OPC server. The connection of Labview with the OPC server is implemented by using data sockets. At the start of the controller for each relay a connection with the OPC server is opened. These connections are kept open throughout the running of the controller and closed when the controller is closed. When a relay has to be switched its connection with the OPC server is used and there is no need to open or close a connection at that time.

This fact keeps the function for setting the relays simple, because the opening and closing functions for the connections can be omitted. The function is depicted in Figure 3-5. On the left side the connections to the relays and the data is shown. A 'for' loop iterates trough all the relays and compares the current setting with the new setting. If they differ then the new value is written to the OPC server.

The set relays function is called from the main program, as is visible in Figure 3-6. This implemented as an 'Event Structure'. Such a structure can be used to monitor variables and act if they change. The structure is set up to look at the 'Pset' variable. If the value of this variable changes,
be it from a database or manual action, the new value is converted to an array of Boolean and the relays are set accordingly.

![Figure 3-6: Calling the set relay function](image)

### 3.2.3 Graphical User Interface

The interface of the control is shown in Figure 3-7. On the top a database UDL file can be specified and the selection between manual and database set values can be made. When manual mode is selected they can be entered below, in database mode they are shown there. The lower right part of the interface shows all the relays and their current status.

![Figure 3-7: User Interface of workshop control](image)

Instructions on how to operate the control are available in Appendix C.2.
3.3 Results

The controller described in the previous section is tested in the DeMoTec Laboratory and works according to the specifications. It is possible to set target values for P in the database and the controller will then change the appropriate relays to control the loads. Because six of the relays are connected to 500 W lamps, the active power can be controlled between 0 and 3000 W with steps of 500 W. The time lag between setting a new target value and the actual switching of the relays varies between 5 and 10 seconds per relay that has to be switched. This means that switching from 0 to 3000 W can take up to 60 seconds.

3.4 Conclusion and recommendations

For a load workshop in the DeMoTec laboratory a Labview controller for active power is developed. The workshop consists of a RTU and several loads that can be switched on or off with a relay. The Labview controller can read a target value for active power from a database and translate this value into the corresponding configuration of the relays. Changing the state of a relay can take up to 10 seconds. The load can be varied from 0 to 3000 W with 500 W steps.

Recommendations for this control are to make the conversion program described in section 3.2.1 more flexible so it can handle all kinds of loads and to further reduce the response times of the relays. Another valuable addition is the support for Q control so that loads which produce or consume reactive power can be connected.
4 200 kVA Biodiesel Genset

In the DeMoTec laboratory a 200 kVA Biodiesel Generator is available. This generator can be a useful addition to various experiments because it is possible to set values for the active and reactive power remotely. In this chapter first an introduction of the generator will be given. In the second section the design of the Labview control system for this generator is described in detail. This is followed by an instruction on how to operate the control. After that the results of the control system are discussed. The chapter ends with a conclusion and some recommendations.

4.1 Introduction

The 200 kVA Biodiesel Generator set is build by Polyma. It consists of a 183 kW diesel engine from KHD which can run on diesel, biodiesel or vegetable oil. It has a rated speed of 1500 rpm. This diesel engine drives a Marelli Motori generator with a rated voltage of 400 V and a rated current of 303 A at 50 Hz. At a power factor of 0.8 this means that the maximum apparent power is 210 kVA.

This diesel generator set is controlled by a Symap power management system from Stucke elektronik GmbH. This system provides power protection, monitoring, diesel control and power management. It can be accessed for remote control in several ways. On the front panel a standard serial RS232 interface is available. This port supports the Modbus and PC-link protocols. On the rear of the main panel three more connectors exist. First there is a serial RS485 or RS422 connector which supports the Modbus and Profibus protocols. Then there is an optical fibre interface that can also work with the Profibus protocol. And finally a Canbus port is
provided on the back side. This port can be used for the communication between several Symap diesel generator systems. With the use of these interfaces the system can be controlled remotely.

4.2 Control

An overview of the control system for the 200 kVA biodiesel genset is shown in figure Figure 4-1.

For the control of the 200 kVA biodiesel generator the RS232 port in combination with the Modbus protocol is used. The RS232 runs over a serial connection. The settings of the parameters for the serial connection are shown in Table 4-1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Com-Port</td>
<td>COM 2</td>
</tr>
<tr>
<td>Baudrate</td>
<td>57600</td>
</tr>
<tr>
<td>Parity</td>
<td>None</td>
</tr>
<tr>
<td>Flow Control</td>
<td>None</td>
</tr>
<tr>
<td>Mode</td>
<td>RTU</td>
</tr>
</tbody>
</table>

Table 4-1: Serial connection settings

The Labview control function consists of four parts. First there is a main function that starts when the program starts. This main function uses two sub functions. One is for reading the measurement values of the diesel generator and one for setting new target values for the active and reactive power to the genset. Finally there is a function for reading and updating the database. Before these functions are explained a brief introduction about Modbus is given.

4.2.1 The Modbus protocol

Modbus is a simple protocol designed for data transfer from and to measuring devices. A Modbus communication system consists of one master and one or more slaves. Each slave has a set of registers. In these registers the slave can store measuring values and parameters. The master can send messages to request the contents of a register or to write a new value to a register. Each Modbus message consists of the four parts visible in Table 4-2.
### Table 4-2: Structure of Modbus Message

<table>
<thead>
<tr>
<th>Address</th>
<th>Function Code</th>
<th>Data</th>
<th>Error Check</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 bits</td>
<td>8 bits</td>
<td>N*8 bits</td>
<td>16 bits</td>
</tr>
</tbody>
</table>

At the beginning of a message the address of the slave is specified. When no address i.e. zero is specified the message will be send to all slaves. Only the slave that is addressed will respond to the message.

The second part of the message is the function code. This code defines the operation that has to be performed by the slave. The diesel generator supports function code 03, which is the reading of a register and function code 06 which is writing to a register.

The third part in the Modbus message is the address of the register on which the operation specified in the function code has to be performed. The implementation in the diesel generator also allows to specify a starting address and a total number of addresses. This makes it possible to read multiple registers with one message and decreases overhead.

The last part of the message consists of a 16 bit error check. The slave will only respond to the message if the error check is correct.

At the National Instruments website a Labview library with Modbus functions is available [4]. The NI Modbus add-on provides a Labview library with Virtual Instruments, or VI’s, created for Modbus. The VI’s needed for the diesel control are the ‘Modbus serial init’ VI and the ‘Modbus serial master query’ VI. The Modbus messages are constructed in Labview by using type definitions from the library. The bundle-by-name function of Labview is used to set the parameters of a Modbus cluster to the correct values. The cluster can then be connected to the ‘Modbus master query’ VI. This VI will send the query to the slave and return the response.

#### 4.2.2 Main Controller Function

The main controlling program consists of four steps. The first step is the initialization part. Here the serial Modbus and database connections are opened and the charts are reset to zero. This part can be seen in Figure 4-2.

When the initialization step is ended, the program enters the main loop. This loop will execute until the user presses the stop button. In this loop two sub functions are called in a sequential way.

First the measurement values from the generator are read, as seen in Figure 4-3. The actual reading is done with the ‘Modbus read’ function. This function, when supplied with the correct registers and a link to the serial connection, returns the current values of the requested registers. More information on this function is available in section 4.2.3.
The values are returned as an array. This array is converted to a cluster and the content is shown on the front panel. The measured power parameters are also written to a graph on the front panel. Finally, the measurement values are written to the database with the 'Set measurement values to database' function, this function is explained in section 4.2.5.
The next step of the main loop is depicted in Figure 4-4. This step is responsible for transferring the new target values to the biodiesel genset. The target values for active and reactive power can be entered manually or read from a database. Because the biodiesel genset can only control the active power and power factor parameters, the Q value is converted to a power factor. Information on this function is available in section 4.2.5. The 'set Modbus' function, explained in section 4.2.4, transfers the apparent power and power factor set values to the diesel genset. The new values are only set to the generator if they differ with the previous values. This is done to prevent unnecessary write actions.

In the last step of the main control the connections with the serial port and the database are closed.

4.2.3 Reading measurement values

The measurement values stored in the biodiesel generator can be retrieved via the Modbus Read Output Registers function. Every parameter that can be read has its own register address. Therefore, for each register that needs to be read, a Modbus command has to be sent. This happens in the 'read Modbus' function shown in Figure 4-5. On the left-hand side the input arguments are displayed. At the top is an array with the addresses of the registers that are read. Below that is a VISA resource that indicates on which COM port the Modbus connection is opened. The last input is an error handler.
The next part of the function consists of a 'for' loop that executes for each address. It creates a Modbus command (large pink rectangle in Figure 4-5) and sets the address to the address specified in the input argument. Next the 'Modbus write command' function is called. This function sends the command to the generator and returns the content of the register that was addressed in the request.

### 4.2.4 Writing new set values

The function to write the new set values for P and Q to the generator is shown in Figure 4-6. The same Modbus write command function is used as in the previous section, but now the 'Write Single Register' command is used. Inputs are again a VISA resource, an array with the addresses of the registers where the data should be written to and an array with the data that has to be written. The 'for' loop has two iterations. The first time it will send the P target value and on its second iteration the Q target value is send. On a successful write the function returns nothing, but if the write failed an error is returned.
4.2.5 Reading and updating the database

The controller for the 200 kVA biodiesel genset uses two functions to access the database. One function is for reading the target values and one for writing the measurement values. Reading the target values is done in the same way as explained in Appendix 8D.4. When the target values for active and reactive power are read, they have to be converted to the corresponding values of active power and power factor. For the conversion the formulas shown in equation (1) are used. The ‘PQ to S Phi’ function takes care of this. It is shown in Figure 4-7.

\[
S = \sqrt{P^2 + Q^2} \\
PF = \cos(\varphi) = \frac{P}{S}
\]  

(1)

Should \(P\) and \(Q\) both be zero, then the power factor is automatically set to one. The conversion function also checks if the target values are within the limits of the generator. The maximum active power is limited to 160 kVA, and the power factor must not be lower than 0.8. If the target values are not within these bounds, the user is warned and the target values are set to their maximum allowed values.

4.2.6 Graphical User Interface

The interface of the control for the 200 kVA Biodiesel genset is depicted in Figure 4-8. In the top left part a database link can be specified and the correct COM port can be selected. A start and stop button can be used to start and stop the controller. Below of these controls are the controls for the target values of \(P\) and \(Q\). At the left side a switch lets the user choose between manual or database control. Manual set values can be entered in the boxes on the right from the switch. A warning light will blink when the set values are out of range.
The middle upper part of the interface shows the measured values for the active, reactive and apparent power, voltage, frequency and power factor. At the bottom the screen a graph of the active and reactive target and measured values is available.

4.3 Results
The controller described above is tested in the DeMoTec Laboratory. It can read the target values for active and reactive power from the database and transfer them to the biodiesel generator via the Modbus protocol. Figure 4-9 shows the user interface of the control when the system is running.

The white and green graph represent the active power and the yellow and green graphs the reactive power. The active power is controlled reasonably well, although there is some delay, and the diesel generator cannot cope with fast changes in the profile.
The control of the reactive power however, is not good. The measurements show that it does not follow the load profile. This is due to the power factor controller used in the Symap control system. It is very sensitive and small variations in the power factor can lead to huge variations in the reactive power.

4.4 Conclusions and recommendations

A control system for the 200 kVA biodiesel is designed and realized. It can handle target values for active and reactive power from a database or from user input. A user interface provides control of the system and an overview of measured values from the generator. The target values are send to the genset with the Modbus protocol. The control of active power works but has a limited speed. The reactive power control is very sensitive and does not follow the target values. This control cannot be improved from the outside. Changes to the power factor controller in the symap system are necessary to improve the controller.

A good addition to the system would be to automatically start the diesel engine when needed, and shut down when possible. This could save fuel as there is no need to let the generator run at idle speed.
5 20 kVA Speed Variable Generator

This chapter explains the control that is developed for the 20 kVA speed variable diesel generator set. First an introduction about the generator is given. After that the Labview control system is explained in detail. The third section presents the results of the control and in the last section a conclusion and some recommendations are given.

5.1 Introduction

Most diesel generators have an engine that runs at a constant speed, independent of the electrical power that has to be produced. This means extra fuel consumption and more wear on the motor. The costs of the power from the generator will thus be higher. To overcome this problem diesel generators that can adapt the speed of the engine have been developed. One of these generators is available in the DeMoTec laboratory. The generator set has been developed by SMA and Kirsch and consists of a Deutz 30kW diesel engine, a Kirsch permanently excited synchronous generator and a SMA 3-phase static converter. A photograph of the genset can be seen in Figure 5-1. A speed-controlled generator has numerous advantages over a traditional diesel generator. These include lower fuel consumption and fewer emissions, longer lifetime and less sound production.

Figure 5-1: Picture of 20 kVA diesel generator
The generator set is connected to a Remote Terminal Unit or RTU. This is a small PC with a network interface and it allows the generator to be remotely controlled. Target values for the P and Q can be set and measurement values can be read from the RTU with the OPC protocol.

5.2 Control
To control the active and reactive power of the generator a control program was written in Labview. The global control method is depicted in Figure 5-2.

![Figure 5-2: 20 kVA diesel generator control method](image)

The Labview program runs on a standard Windows PC. It can connect to a database to retrieve the target values for the active and reactive power of the generator. When the set values are known, they are send to the RTU by means of a OPC server also running on the windows PC. This OPC server can communicate with the RTU through the XML/RPC protocol. The RTU handles the incoming data and controls the diesel generator with the use of the SMA data protocol.

The Labview control of the generator can be divided in four distinctive components. First there is the main section. This is the part that is launched when the program is started. It will call three functions, a function to get the target values from the database, a function to write the new target values to the OPC server and finally a function to write the measurement values to the database. These four parts are explained in detail in the coming sections.

For an overview of VI’s and functions used please consult the Table of used VI’s in Appendix A.

5.2.1 Graphical User Interface
The Labview control program has a graphical user interface to provide easy access to the control parameters and measurement values. This user interface, shown in Figure 5-3, consists of three parts. The top left section gives the user the possibility to enter the database link and the OPC URL. When these are entered the program can be started with the start button and stopped with the stop button.

To the right of this section the actual and target values for P and Q are shown. The switch chooses between manual control and automatic control. In manual mode the user can enter target values for P and Q
directly in this screen, while in automatic mode the target values are read from a database.
In the bottom of the interface, a graph is displayed. This graph shows the actual and target values for P and Q.

5.2.2 Main Control
The main control of the Labview program can be divided in three phases. The first phase opens the connection with the database and waits for the user to press the start button. The second phase is the most important phase and contains the main body of the program. It is shown in Figure 5-4.
It consists of a while loop that keeps running until the user presses the stop button. First the 'Get Pset and Qset' function is called. This function returns the target values for active and reactive power from the database. The target values are also connected to a graph so the user has information about what the current target values are.
The target values are then given to the 'SetPQ' function. This function opens a connection to the OPC server, sets the new target values in the generator and returns the current measurement values. The current values are then written to the database with the 'PQ to DB' function and they are also shown in the graph.
The last phase of the main function is entered as soon as the user presses the stop button. In this phase the connection with the database is closed and should errors have occurred during the program they are shown on the screen.

5.2.3 Getting the P and Q target values

The function explained in this section is responsible for getting the active and reactive power target values from the database. The block diagram of this function is shown in Figure 5-5.

The function uses a connection reference to the database to select the appropriate information from the database. The results of the query are converted to integers and then returned to the main function.

5.2.4 Setting and Reading P and Q

An OPC server is used to communicate with the RTU that controls the generator. To set and read P and Q values, Labview has to connect to this
server. This is done in the same way as explained in section D.3 on page 13. This function is divided in two parts. First the target values are written to the OPC server. The block diagram of this action can be seen in Figure 5-6.

To prevent unnecessary write actions the current set value is compared with the new set value. Only if they differ, the new value is written to the OPC server.

After the P and Q values are set the current measurement values are read from the OPC server. This is the second part of the function. This part is shown in Figure 5-7.
5.2.5 Updating the database

After the setting and reading of the active and reactive power, the measurement values are put in the database. This does the 'Set PQ to Database' function. This function updates the measurement values in the database. To update the values the function uses a SQL query. The query is constructed using strings and the measured values that were retrieved from the OPC server. The final query looks like the following:

'UPDATE 20kva SET pmeas = (measured P value), qmeas = (measured Q value)'

The query is then executed and when it does not return an error, it was a success.

The block diagram of this function is displayed in Figure 5-8

![Figure 5-8: Updating database](image)

5.3 Results

Due to an absent exhaust pipe the 20 kVA diesel generator could only be tested in simulation. The controller is able to read set values from the database or the front panel. The connection with the OPC server is not tested.

5.4 Conclusion and recommendations

A control for the 20 kVA diesel generator was developed. It can read the target values from the database, set the values to the generator and store the measurement values in the database. The simulation of the control was successful but the control was not tested with the real generator. Recommendations for future changes include adding more measurement values such as apparent power and power factor. Also a there is no check whether the target values are with the limits of the generator.
6 P Q Controller test experiment

This chapter explains the experiment that was conducted using the developed P Q controller system. First a description off the experiment is given and the participating generators are described. After that the result of the experiment is shown.

6.1 Experiment description

To test the developed P Q control system an experiment is conducted. The experiment simulates three different generators and one load over the course of one day. The connection diagram of the test setup is shown in Figure 6-1.

The three generators and the load workshop are connected to the 400V 50Hz grid of the DeMoTec laboratory. This grid is also connected with the mains grid.

Each generator is given a specific PQ profile. These profiles are based on a typical load profile for a specific type of generator. The profiles are displayed in Figure 6-2. There are profiles available for a PV system, a wind generator, a hydro generator and the load.

In the experiment the 15 kVA genset takes the profile of the Hydro generator, the 100 kVA Multi-PV the profile of the PV generator and the 200 kVA biodiesel genset gets the profile of the wind generator. The time of the profiles was adjusted to make a 15 minute step every five seconds. This means that the whole profile takes 480 seconds to complete. Not all profiles can be used directly. The profile for the biodiesel genset is filtered with a low pass filter because it is unable to follow the rapid changes that occur in the wind generator profile. The converted load profiles can be seen in Figure 6-3.
6.2 Results
The PQ profiles described above are loaded into the SCADA controller. The result is visible in Figure 6-4. The workshop is not shown here because it does not return any measurement values. The generators generate or consume active and reactive power according to the PQ profiles. Only the
reactive power of the Biodiesel generator does not correspond. The reason for this is given in section 4.3.
Also visible is that the Q control for the 15kVA genset stops after 230 seconds. This is not an error in the control system but is the result of human interference.

![Graph of P Q Measurements of generators](image)

**Figure 6-4: P Q Measurements of generators**

### 6.3 Conclusion
An experiment was designed to test the P Q control system. Three generators and one load were part of the test. Each device was given a PQ profile based on typical generator profile. The result of the experiment shows that the devices follow the target values from the PQ profile. Thus it is possible to use the system for testing of DG units with active and reactive power control in a network.
7 DERlab esd User Interface

This chapter covers the development of a user interface for the DERlab esd database. After an introduction about DERlab and the database, the structure of the website is explained. The next section deals with the search function of the database. In the last section the security of the site is explained.

7.1 Introduction

The Distributed Energy Resources Laboratory (DERlab) is a European Network of Excellence of several acknowledged research institutes. They focus their research on the integration of distributed energy resources in the electricity grid. Their mean goals are:

- Setup an independent, distributed DER laboratory in Europe
- Support for the development of standards in the field of DER
- Create and improve relationships between the DER research institutes

To realize these targets certain activities are planned. One of these activities is the creation of a database that contains testing facilities and equipment. Every institute has lots of testing facilities and equipment and they all have their own area of expertise. However because of a lack of standardization and compatibility, the organization of experiments that involve different laboratories is tedious. This situation is undesirable because it leads to a waste of resources and limits the research potential of the DERlab network. Therefore the DERlab Equipment and Service Database (esd) was founded. This database contains information about the equipment and facilities from all the partners. With such a database it is possible for the researchers in the cooperating institutes to have easy access to information from other institutes. The quality of experiments would improve due to the easier exchange of information and complex testing would be much easier to organize.

The DERlab esd is implemented as a MySQL database on a web server. This ensures easy access via the internet from laboratories all over Europe.

Data in the database is ordered in three main categories. First there are the facilities. In this category each partner can enter the research facilities that are available at their institutes. Next there is a category for the stationary equipment. Here stationary equipment can be listed. Finally there is a list for the mobile equipment. This includes equipment that can easily be transported.
The categories mentioned above are implemented in the database as tables. Each institute has his set of these tables and is responsible for maintaining them.

This database was implemented as described above, but it lacks a simple user interface. Searching the database for specific information is not possible and adding or removing entries requires knowledge of a difficult interface. Therefore an easy to use interface is required. The interface should allow the following actions:

- Search the database
- Add entries
- Delete entries

In this chapter the interface that is created will be explained. First the global structure and layout of the website will be discussed. After that the search function and the login procedure of the website are explained.

7.2 Global structure of the website

7.2.1 Overview

An overview of the structure of the website can be found in Figure 4-1. Each block represents a PHP file and the connections between the boxes represent the normal flow through the website. Of course alternate paths are possible but these are not shown as this would make the figure unnecessarily complex. When a client arrives at the website an index page is displayed. Here the user is presented six options. These options are:

- Search for equipment or facilities
- Search for interconnection requirements
- Add new equipment or facilities
- Add new interconnection requirements
- Delete equipment or facilities
- Delete interconnection requirements
When a user wants to search for specific entries in the database he is automatically forwarded to the 'search.php' or 'searchinterconnect.php' pages. A log in is not necessary because every institute has reading rights for the other institute's tables. However when a user wants to add or delete data he is send to a ‘login.php’ page first. At this page the user needs to enter his username and password. If this is correct, which is checked in ‘checklogin.php’, then the user is send to the page he originally wanted to visit. For the add and delete functions it is required to log in because the user needs to be identified and he can only add or delete items in tables of the institute he is from.

7.2.2 Layout

For the user it is important that the interface has a consistent layout with easy to understand navigation options. Two php files make sure that every page on the website has the same header and footer, one file for the top part (htmltop.php) and one for the bottom part (htmlbottom.php). These files are based on the layout of the rest of the DERlab website and use same style sheet. This guarantees a consistent look throughout the whole website. The result of ‘htmltop.php’ can be seen in Figure 7-2.
The `htmltop.php` file also includes a login check function. This means that at the top of every page the user can see whether he is logged in. When he is not logged in the user is presented a possibility to log in. After he is logged in he can see the status of his login and he is presented a possibility to log out.

The `htmlbottom` function merely shows the bar displayed in Figure 7-3.

### 7.2.3 Database connection

The data is stored in a mySQL database. To gain access to this database a connection has to be made. For this the function shown in Code Example 7-1 was created.

```php
<?php
function con2sql($myusername,$mypassword)
{
    $con = mysql_connect("abta-14.iset.uni-kassel.de","$myusername","$mypassword");
    if (!$con)
    {
        die('Could not connect: ' . mysql_error());
    }
    if (!mysql_select_db("DERlabESD",$con))
    {
        die('database does not exist ' . mysql_error());
    }
}
?>
```

**Code Example 7-1**

A lot of the php files of the website need to have access to data from the database. Therefore this function was put in a separate file and included in the other files.

### 7.3 The Search function

An important part of the DERlab esd is the possibility to search for specific kinds of facilities and equipment. Researchers should be able to find the equipment they need for their experiments. The database stores various
kinds of information about the equipment such as to which category it belongs, what the rated power is and information about the calibration certificates. According to these options specific queries should be created that gives the user all the information he is looking for.

The search is implemented in a top down manner. This means the user can further refine the results until he finds the information he needs. At first the user arrives at the ‘search.php’ page. Here information is gathered about what the user wants to search for e.g. category, keywords etc. This information is then sent to the ‘results.php’ page which creates a query for the SQL database based on the information from the user. It then shows the results in a table which can be sorted by clicking on the appropriate column name. More information about a certain entry is available when the user clicks on the name of an entry. This will take the user to the ‘info.php’ page. On this page a table presents the user with all the information about a specific entry including contact person details and a link to the datasheet. Should the user want to print this information he can click on a ‘create PDF’ button which takes him to the ‘pdf.php’ page. This isn’t a page as such, but it gives the user with a dynamically generated pdf file that contains all the information about the entry and can be printed or saved on a computer for later use. In the coming sections each of these pages will be described in more detail.

7.3.1 search.php

This page is created to get all the information from the user about what he wants to search for and is divided into two phases. First he is presented a form with a dropdown box with four search options: Facilities, Stationary equipment, Mobile equipment or search the whole database. This page can be seen in Figure 7-4.

When the user has selected a category, he arrives in phase two of the search page. This page shows further refinement options. These options...
are dependent on the category that was selected in phase one. For instance when ‘Stationary equipment’ is selected in phase one the user can refine his search in phase two by selecting a general category and a stationary equipment category. The layout of phase two can be seen in Figure 7-5.

Welcome to the DERlab database search engine
To search the database please select your options below:

You chose for: Facility
Now select a General Category

All Categories

Now select a Facility Category

All Facility

You can enter some optional keywords below:

Search

Figure 7-5: Layout of phase two of the search page

When those selections are made it is possible to supply one or more keywords. The results page will only show entries in which these keywords appear.

When the user hits the search button the data will be send to `results.php` page by use of the http GET method. This method was chosen because it allows the user to bookmark a specific search query. This would not be possible with the http POST method. However extra care had to be taken because the GET method can be vulnerable to SQL injection attacks. Therefore the input has to be checked that it does not contain malicious code. The function used for checking and cleaning the input can be seen in Code Example 7-2.

```php
function check_input($value)
{
    // Stripslashes
    if (get_magic_quotes_gpc())
    {
        $value = stripslashes($value);
    }
}
```
7.3.2 results.php

In the 'results.php' page the actual search query is build. First this page looks what kind of category was selected. For each category a specific 'querybuilder' function was created. This function has as inputs the category that has to be searched for, the column names which have to be returned and a list with all the partner names. The query is created in the following way. First the appropriate columns are selected with the SELECT statement. Then the FROM statement is used to specify to which tables the selected columns belong. Finally a WHERE statement is used to select only the relevant information. Because each partner has a separate table and all the tables have to be included in the query, a UNION command is used to bind all the tables together.

Now that the query has been created it is passed on to the SQL server. The results are checked and when they are all right the 'createtable' function is called on them.

To make the results from the query easy to read they are put in a table. This table has to be dynamically created according to the number and type of results that have been found. Therefore first the results are analyzed. When the amount of columns is known the column headings are created. These headings include a link to the same results page, but add an extra sort by argument. This means that when the user clicks on the column name, the same table will be displayed, but the results are now sorted by the column the user clicked on.

The entry names are also displayed as hyperlinks. This link leads the user to the info.php page. The information about which entry the user wishes to see is send along with the http GET method. In Figure 7-6 an example of the results page can be seen.
7.3.3 info.php

The ‘info.php’ page is showed when a user wants to see information about a certain entry in the database. It shows all the information that is stored for that entry. The category and ID number of the entry is received from the previous page and with that information a query is made. This query looks in the correct table and selects all the columns. The result is displayed in a table and looks like Figure 7-7. At the bottom of the page is a create pdf button. This button leads the user to the ‘pdf.php’ page.

---

**Figure 7-6: Example of results page**
Not Logged in  Back  Home

Equipment Data:

<table>
<thead>
<tr>
<th>Name/Model:</th>
<th>Grid impedance RLC circuit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer:</td>
<td>arsenal research</td>
</tr>
<tr>
<td>Description:</td>
<td>1-Phase RLC resonance circuit and grid impedance simulation</td>
</tr>
<tr>
<td>General Category:</td>
<td>DER</td>
</tr>
<tr>
<td>Subcategory:</td>
<td>Artificial Mains Network</td>
</tr>
<tr>
<td>Calibration Certificate:</td>
<td></td>
</tr>
<tr>
<td>Datasheet:</td>
<td></td>
</tr>
<tr>
<td>Located at:</td>
<td>Inverter Testing Laboratory, Vienna</td>
</tr>
<tr>
<td>Partner:</td>
<td>ARS</td>
</tr>
<tr>
<td>Country:</td>
<td>Austria</td>
</tr>
<tr>
<td>Contact Name:</td>
<td>Roland Bründlinger</td>
</tr>
<tr>
<td>Contact Email:</td>
<td><a href="mailto:roland.bruendlinger@arsenal.ac.at">roland.bruendlinger@arsenal.ac.at</a></td>
</tr>
</tbody>
</table>

Figure 7-7: Example of info page

7.3.4 pdf.php

For the user it is convenient when he has the possibility to save or print the information he acquired about a certain entry in the database. To accommodate this, a pdf generator has been added to the website. To create the pdf file, the FPDF function was used. This is a non-commercial pdf function for php. Because there are a lot of entries in the database, and pdf files require a lot of space, the pdf files are created dynamically at the moment the user requests them. The information which is needed is passed on from the `info.php` and parsed into the pdf file. The user has the choice to open the pdf file or save it to disk for later access.
Mobile Equipment Data:

Name: PM3000A
Manufacturer: Voltech
Description: 3-Phase Power Analyser
General Category: DER
Subcategory: Power Quality Analyser
Calibration Certificate: 
Datasheet: 
Located at: Inverter Testing Laboratory, Vienna
Partner: ARS
Country: Austria
Contact Name: Roland Bründlinger
Contact Email: roland.bruendlinger@arsenal.ac.at

Figure 7-8: Example of pdf document

7.3.5 Conclusion and recommendations
The search function which was created allows the user to search for equipment and facilities stored in the database. First the user can specify whether to search for mobile equipment, stationery equipment, facilities, or the whole database. After that selection it is possible to select a general and an equipment category and add keyword. The results are displayed in a table which can be sorted by clicking on the appropriate column name. More information on a specific entry is available when the user clicks on the name of an entry. The information can be saved or printed. Thus it provides a user-friendly search environment. Recommendations for future extensions incorporate the possibility to search for specific power requirements (i.e. show all the PV inverters with a rated power above 5kW) or search for equipment at a specific location.

7.4 Login
The information in the database is organized in such a way that each institute has reading rights for the whole database and read/write rights
for the tables with their own information. Thus when adding or deleting entries from the database it is important to know which user is accessing the database, and whether he has the correct privileges for the kind of operation he wants to perform.

Therefore each page that requires the user to be logged in has a function which checks whether the user is logged in and if he has the correct rights to perform the operation he wants to perform. This was realized by using the php session variables.

Session variables are global variables which are stored for a complete session. A session starts when the page is loaded, and ends when the browser is closed. During this session each php file can access the global variables. These possibilities make the session variables suited to store the login information. The user can perform various actions while logged in, and does not need to log in separate for each page.

There are three variables that are saved in the session; one for the username, one for the password and one that indicates the privileges of the user.

The setting and reading of these variables is done in the following way.

When the user wants to access a page that requires the user to be logged in, the session is started and the variables are checked as seen Code Example 7-3.

```php
<?php
session_start();

if(!isset($_SESSION['myusername']) && !isset($_SESSION['mypassword']))
{
    header("location:login.php?action=5");
    exit;
}

Code Example 7-3: check if logged in
```

There are two possibilities, either the variables are set or they are empty. When the variables are not set it means that the user has not yet logged in and he is forwarded to the login.php page. This page will ask him to fill in the username and password. The checklogin.php page will then check whether the information entered does correspond with the usernames and passwords stored in a separate php file. The code used can be found in Code Example 7-4.

```php
function checkusername($myusername,$mypassword)
{
    $access = 0;
    include ("pwform.php");

    if($myusername == $SpecialUS && $mypassword == $SpecialPW)
```

49
{$access = 2;
}
else
{
  for ($i=1; $i<=count($Users)+1; $i++)
  {
    if (($myusername == $Users[$i]) && ($mypassword ==
      $Password[$i]))
    {
      $access = 1;
    }
  }
  return $access;
}
$access = checkusername($myusername,$mypassword);
if($access>=1)
{
  // Register $myusername, $mypassword
  $_SESSION['myusername'] = $myusername;
  $_SESSION['mypassword'] = $mypassword;
  $_SESSION['myaccess'] = $access;
} else //if access<1 no valid username was given
{
  require('htmltop.php');
  echo "<p>Wrong Username or Password<br> Please go back and try again</p>";
  session_destroy();
  require('htmlbottom.php');
}

Code Example 7-4: Checking the username and password

Should this be the case then the session variables are set with the appropriate information and the user is returned to page he intended to visit. For administrative purposes there is also a super user or admin account available. This account has full access to all the tables in the database. A status bar is incorporated in the header of each page. When the user is not logged in this is shown here together with a link to the login page. After the user is logged in this status bar shows the name as which he is logged in and a logout link. The logout link redirects to the `logout.php` page which simply destroys the session and sends the user to the index page of the site.

7.4.1 Conclusion and recommendations
The login part ensures that only people with the correct username and password have access to the adding and deleting part of the database. It
is only possible to enter a page which is protected with a password when the correct sessions variables are set. The username and password are stored in a separate file in plain text. A higher level of security could be reached if the passwords would be saved in an encrypted file.
8 Conclusion

A SCADA system for the control of active and reactive power is developed and realized in the DeMoTec laboratory. A central server, running a Labview controller and a MySQL server, can read target values for active and reactive power from PQ profiles and set these values in a database. In addition it can read the measurement values from the generators and loads and present them on a front panel. The measurement values can also be saved to a file.

Labview P and Q controls for three other devices are developed. First there is a control for a load workshop. It can turn several loads on and off, according to the target value for active power in the database.

Secondly there is a control for a 200 kVA biodiesel genset. This control can read the target values for active and reactive power from a database or the front panel. It uses the Modbus protocol to transfer the target values to the biodiesel generator.

The third control is for a 20 kVA speed variable genset. It also can read the target values for P and Q from a database or the front panel and set them to the generator. Communication with the generator is done via OPC.

An experiment with three generators and one load has shown that the control system can be used for active and reactive power control in the DeMoTec laboratory.

It possible to add other devices to the control system and this makes it a good platform for future experiments with P and Q controllable generators and loads. Another valuable expansion would be to give the central server more control. Instead of just setting target values for P and Q, it then could also start up, configure and shutdown generators.

Finally a web-based user interface for a database containing DER units and services is implemented. A search engine allows user to seek out DER units according to several requirements and a login systems ensures that only people with the correct rights have access to the database.
### Appendix A  Tables of created VI's

#### A.1  SCADA Controller

<table>
<thead>
<tr>
<th>Name</th>
<th>Filename</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>Main.vi</td>
<td>None</td>
<td>None</td>
<td>Main VI of the SCADA controller</td>
</tr>
<tr>
<td>Get Column names</td>
<td>Getcolumnnames.vi</td>
<td>Database con. reference Path to csv file</td>
<td>Columnnames Tablename</td>
<td>Gets the column names and table name from the database</td>
</tr>
<tr>
<td>Get measured values</td>
<td>Getmeasval.vi</td>
<td>DB con. Ref. tablename</td>
<td>values stored in table with tablename</td>
<td>Gets the measured values stored in table with tablename</td>
</tr>
<tr>
<td>Set to DB</td>
<td>Todb.vi</td>
<td>DB con. Ref. Tablename Pset Qset</td>
<td>None</td>
<td>Updates the Pset and Qset values in table specified in tablename</td>
</tr>
</tbody>
</table>

**Table A-1: Used VI's for SCADA Controller**

#### A.2  20 kVA Variable Speed genset

<table>
<thead>
<tr>
<th>Name</th>
<th>Filename</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>20kvamain.vi</td>
<td>None</td>
<td>None</td>
<td>Main program for the 20 kVA controller</td>
</tr>
<tr>
<td>Get Set Values</td>
<td>Getsetval.vi</td>
<td>DB con. Ref.</td>
<td>Pset Qset</td>
<td>Retrieves Pset and Qset from database</td>
</tr>
<tr>
<td>Set P Q</td>
<td>Setpq.vi</td>
<td>Array with [P,Q] OPC URL</td>
<td>Array with Current status</td>
<td>Set the new target values to the generator and retrieves the measurement values</td>
</tr>
<tr>
<td>Write values to database</td>
<td>Writeval2db.vi</td>
<td>DB con. Ref. with current status</td>
<td>None</td>
<td>Updates the database with the measured values</td>
</tr>
</tbody>
</table>

**Table A-2: Used VI’s for 20 kVA Variable Speed genset Controller**

#### A.3  Workshop controller

<table>
<thead>
<tr>
<th>Name</th>
<th>Filename</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>Workshopmain.vi</td>
<td>None</td>
<td>None</td>
<td>Main program for the workshop controller</td>
</tr>
<tr>
<td>Open OPC connection</td>
<td>Openopc.vi</td>
<td>None</td>
<td>Array with OPC connections Name for each connection</td>
<td>This VI opens for each relay an OPC connection</td>
</tr>
</tbody>
</table>

A-1
| Convert PQ to bool | int2bool.vi | Active power boolean array | Boolean array | Converts the P value to a Boolean array for the relays |
| Set Relay | Setrelaysimple.vi | Array with OPC connections Array with strings | None | Set the relays to the new position |
| Compare values | Compvalues.vi | Read value Set value | Boolean | Compares the read value with the set value, returns true if they differ |

**Table A-3: Used VI’s for Workshop Controller**

### A.4 200 kVA Biodiesel genset

<table>
<thead>
<tr>
<th>Name</th>
<th>Filename</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main</td>
<td>Mainsymap.vi</td>
<td>None</td>
<td>None</td>
<td>Main program for 200 kVA control</td>
</tr>
<tr>
<td>Get Column names</td>
<td>Getcolnames.vi</td>
<td>DB con. Ref.</td>
<td>Array with columnnames</td>
<td>Returns the columnnames from the 200 kVA table in the database</td>
</tr>
<tr>
<td>Modbus Read</td>
<td>Modbusread.vi</td>
<td>VISA resource Array with addresses</td>
<td>Array with data at addresses</td>
<td>Reads the data at the location of the input addresses</td>
</tr>
<tr>
<td>Read set points</td>
<td>Readsphi.vi</td>
<td>DB con. Ref.</td>
<td>Power Factor setpoint Load sharing setpoint</td>
<td>Reads the P and Q targets from the database and converts them to S and phi</td>
</tr>
<tr>
<td>Modbus Write</td>
<td>Modbuswrite.vi</td>
<td>VISA resource Array with addresses</td>
<td>none</td>
<td>Writes the data to the addresses</td>
</tr>
<tr>
<td>P Q to S phi</td>
<td>Pq2sphi.vi</td>
<td>P Q Array with data</td>
<td>S Phi</td>
<td>Converts P and Q to S and phi</td>
</tr>
</tbody>
</table>

**Table A-4: Used VI’s for 200 kVA Biodiesel genset Controller**
Appendix B  Managing and connecting to the database

This appendix describes how to operate the MySQL database server and how to connect to it by using UDL files.

B.1 Starting the database server

The MySQL database is installed via the XAMPP package [2]. To start the database, double-click the ‘XAMPP control panel’ icon on the desktop. The control panel will now open, see Figure B-1. To start the MySQL server, click the start button next to it. When changes to the database have to be made, the Apache server must also be started.

When the MySQL server is started, the control panel can be closed. To verify whether the server is running, check if there is an orange XAMPP icon in the taskbar. Click on it to open the control panel. If a green box with the word ‘running’ appears next to the server, the server is running.

B.2 Managing the database

Changes to the structure and content of the database can be made by using the ‘PHPmyAdmin’ utility. To access this first make sure that the MySQL and Apache servers are running (see Appendix B.1). Now open a web-browser and type in the address bar: ‘localhost/phpmyadmin’. The administrator utility for the database server is now opened. To log-in use the account displayed in Table B-1. An overview of the server is given on the next screen. To manage the user
accounts of the server, click on the ‘privileges’ link. Aside from the root user, there is one other account. The details for this account are displayed in Table B-2. This account is used by the Labview controllers of each device, and it has only rights for reading and writing data.

<table>
<thead>
<tr>
<th>Username</th>
<th>*****</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>*****</td>
</tr>
</tbody>
</table>

Table B-1: Login details for database admin utility

<table>
<thead>
<tr>
<th>Username</th>
<th>*****</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>*****</td>
</tr>
</tbody>
</table>

Table B-2: Login details for normal database user

To manage the database used by the SCADA system, select the ‘demotec_control’ database in the drop-down box on the left hand side of the screen. In the current screen it is possible to add, remove or alter tables in the database. For more information on how to use the ‘PHPmyAdmin’ utility please refer to its manual [6]. Note that this utility is only accessible from the computer the server runs on.

B.3 Creating a UDL file to connect to the database

To connect to the database on a remote computer, an UDL file has to be created. This section will explain how to create such a file.

First the MySQL ODBC driver has to be installed on the remote computer. This driver can be found on the MySQL website [7]. Download the driver and install it by following the instructions on screen. When the driver is successfully installed, it has to be added to the ODBC manager. Go to the control panel and open the ‘administrative tools’ folder. Now double-click on the 'Data Sources (ODBC)' icon. In the window that is opened go to the ‘System DSN’ tab and click ‘Add’. Scroll down to select the MySQL OBDC driver from the list and click ‘Finish’. The window shown in Figure B-2 will now open. As seen in the figure, enter the following information. On the login tab, enter a name and description for the database link. The server ip address should be set to ‘172.16.3.32’, and for the username and password refer to Table B-2. At the ‘database’ option select the ‘demotec_control’ database. Click ‘OK’ to close the window. Note that for this step to work, the computer needs to be on the same network as the server.

Now the UDL file can be created. To do this create a new text file and rename it to ‘mysql.udl’. Note that for this to work the ‘hide common file extensions’ option in the windows explorer folder options has to be off. When the file is successfully renamed, double-click it. A configuration window for the UDL file will open. Choose the ‘use data source name’ option and select the MySQL DSN that was created earlier. Click ‘OK’ to close the window. The UDL file is now ready to use in the Labview control programs.
Figure B-2: Creating the system DSN

Data Source Name: demotec control
Description: database for demotec control
Server: 172.16.3.32
User: demotec
Password: ********
Database: demotec_control

Password
The password for the server user combination.
Optional: Yes
Default: [empty]
Appendix C  Operation of Controls

C.1 Operating the SCADA Controller
This section provides an overview on how the SCADA controller is operated. Before opening the controller please make sure that the database is running. For information on how to set up and manage the database please refer to Appendix B.

The SCADA controller can be opened by clicking on the start SCADA controller icon on the desktop. The interface shown in Figure 2-6 will appear. First select the database UDL file by clicking on the yellow folder icon.

For each device that needs to be controlled the following must be done:

- Activate it by clicking the button below its name.
- Set a PQ profile CSV file by entering the path or browsing for it by clicking on the yellow folder icon. For the name of the file please refer to Table C-1. It is of vital importance that the file names are exactly as described.
- Select whether the device should operate in manual or PQ profile mode.
- When in manual mode, set the P and Q target values in the box next to it.

Next select to either stop or repeat the PQ profile when the longest PQ profile is finished by selecting the appropriate radio button and the time step multiplier can be set. The time step multiplier specifies how long one time unit in the PQ profile takes. Default is 1; this means that the time is regarded as a second. But when set to 60, one time unit of the PQ profile corresponds with one minute.

When the above actions are completed, press the ‘ready’ button. If everything is right the green light below the button lights up. For all the devices that were activated, the target values are set to zero. Thus at this time the client systems can be started.

When all the clients have been started and are ready, press the start button.
The controller will now go through the PQ profiles. The measurement values are shown and the active and reactive power values are displayed in a graph. When the value 999999 is displayed, it means that the corresponding parameter is not monitored. During operation a device can be changed from manual mode to PQ profile mode or vice versa. Note that when changing to PQ profile mode at a certain point in time, the controller will set the target values specified in the PQ profile at that point in time i.e. the PQ profile does not pause when the device is in manual mode!

When the end of a PQ profile is reached for a certain device, the controller takes the last value in the profile for that device. When all the active profiles have ended, the profile is either repeated or the program is stopped.

When the program is stopped the set values for the active devices are set to zero. A dialog window then asks whether the measured values should be saved. The measurement values are saved for each device in a separate CSV file. The first column of this file contains the time, and the other columns contain the measurements values in the same order as shown in Table 2-1. The files are saved in the C:\Measurements\ folder. The measurements are saved approximately five times per second, but because this is not constant the time column in the file specifies at which time a measurement is taken.

C.2 Operating the Workshop Control

Before starting the controller the Workshop has to be activated. The Netmaster automatically starts when the workshop is connected to a grid. Next make sure that the OPC server is running. When it is not, start it by double-clicking on the 'Start Workshop OPC server’ icon. Now the controller can be started by double-clicking the 'start Workshop Control’ icon.

Next a database UDL file must be specified. Click on the little yellow folder icon to browse for the file or enter the path to the file manually. Information on how to create this file is available in Appendix B.

When the file is selected, set the manual/database switch to the desired position.

Now press the start button to start the program. If the controller is in manual mode, the set values for P can be entered in the Pset box. If the controller is in database mode then this box displays the current set value that was retrieved from the database.

The controller can be stopped by pressing the stop button. Please note that when the controller is stopped it needs to be restarted by pressing the white arrow on the top left of the screen.

C.3 Operating the 200 kVA Biodiesel genset Control

Before starting the control make sure that the serial cable is connected to the 200 kVA genset. For instructions on how to set up the 200 kVA biodiesel genset please refer to the user manual [5]. Power up the Symap system by inserting the key in the ‘batterieschalter’ switch. An error will
display, acknowledge it with the ‘ack’ button. Now press the auto button once and set the ‘lastprobe’ switch to the ‘1’ position. The generator will now start up and it can be controlled externally.

The Labview PQ control can be opened by clicking on the Start Control icon on the desktop. This launches the user interface shown in Figure 4-8. First a link to the database needs to be selected. Do this by clicking on the yellow folder and selecting the appropriate UDL file. Now the COM port has to be select, default this is COM 2.

The control can now be started by pressing the ‘start’ button. When the program is running a selection between database mode and manual mode can be made. When in manual mode the set values for P and Q can be entered in the appropriate boxes. In database mode the set values will be read automatically from the database.

To exit the controller, use the stop button and close the window. Shut down the generator by setting the ‘lastprobe’ switch to zero.

C.4 Operating the 20 kVA Variable Speed genset

Before running the control for the 20 kVA genset, make sure that the OPC server is running. If it is not running, start it by clicking on the ‘Start 20 kVA genset OPC server’ icon.

Now run the control by clicking on the ‘Start 20 kVA genset Control’ icon. The window shown in Figure 5-3 will now open. First select a UDL file for the access to the database. Next enter the OPC URL. This URL can be found by opening the Matrikon OPC Explorer and browsing to the correct OPC server.

Now the controller can be started by clicking on the start button. When in manual mode the target values can be entered by hand in the appropriate boxes. When database mode is selected the target values are retrieved from the database and shown in the same boxes.

To stop the control, press the stop button.
Appendix D  First version of workshop control

The first version has two modes of control: Manual and Automatic. Manual control allows the setting of each relay by hand. Automatic control reads the load profile from the database. This section will first explain how the manual control is implemented in Labview. Secondly the implementation of the automatic control will be discussed. After that the setting and reading of the relays is explained. The last section gives some information about the connection between Labview and the SQL database.

D.1 Manual control

When the program is started and the user has chosen the manual control option he is shown the front panel in Figure D-1. Here the eight relays are shown together with their current status. The dropdown boxes on the right give the user the possibility to change the status of a relay.

The Block diagram of the manual mode can be found in Figure D-2. It takes the set values from the front panel and sends them to the ‘Set Relays’ function. This function sets the relays to the appropriate values. Its inner workings are explained in detail in section D.3. This function returns the names of the relays and their current value. These are also displayed on the front panel. A green led shows the user which mode is currently active.
D.2 Automatic control

The automatic control mode makes use of a load profile stored in a SQL server. At specified time intervals it reads the load profile from the database, puts it in a CSV file and uploads the file to the Netmaster via ftp. The front panel for the automatic control is shown in Figure D-3.
The automatic control function starts with opening a connection to the database. The user has to specify an UDL file which stores information about how to connect to the database. Once the connection is opened, the workshop controller function is started. This function, the green block in Figure D-4, gets the load profile from the database with the 'get load profile' function (see section D.4). When the values are retrieved, they are compared with the values from the last time a new load profile was set. Only if the load profile differs from the previous, the load profile is uploaded to the Netmaster with the 'set load profile' function (see section D.5). The block diagram of the workshop controller function is shown in Figure D-5.

Because the default state for the relays is off, the first time the automatic control is run the relays have to change to the Auto mode. Therefore the 'set relays' function is used to set all the relays to Automatic mode. The check whether the relays have to be set to auto is done by looking at the Load Profile variable. If this variable is empty it means that the load profile was not yet loaded and the relays have to be set to the automatic mode.

![Figure D-4: Block diagram of automatic control](image-url)
D.3 Setting the relays

Labview offers support for communication with OPC server. This works by using datasockets. Communicating with an OPC server consists of three steps and a basic Labview implementation can be found in Figure D-6. In the first step a connection to the server has to be opened. This is done with the 'DataSocket Open' function. Input arguments for this function are the URL of the variable you want to establish a connection with and whether the connection should be read, write or read/write. If the provided URL is valid the function returns a connection with the variable.

In the second step the actual reading and/or writing will take place. The 'DataSocket Read' function will return the value of the variable. The 'DataSocket Write' function can write a value to the variable. The third and final step ensures that the connection with the variable is closed again. This is done with the 'DataSocket Close' function.

In the workshop there are eight controllable relays available. This means that there are also eight variables which have to be controlled. To control...
all the variables at once the 'set relays' function was created. The only argument of this function is an array with the set values for the relays. The function returns the current values of the relay and the names of the relays for display purposes. The function is displayed in Figure D-7.

The main part of the function consists of a 'for' loop. The loop count is determined by the size of the input array, and thus the amount of relays that have to be set. At first the connection is opened, the URL of the OPC variable is created from the general OPC URL and the channel name. When the connection is opened the current value of the relay is read. This value is compared with the set value by the 'compare values' function. The reason for this is that writing a new value to the Netmaster can take up to 30 seconds to complete. Therefore writing only occurs when the set value differs from the current value. This will significantly speed up the controller because the Netmaster has to handle fewer requests per second.

The 'Compare values' function will produce either true or false. In the true case, which is shown in Figure D-7, the new set value will be written. In the false case the set value matches the current value and no writing will be done. A separate function to compare the values is created because when reading the current value the OPC server returns either On, Off or Auto. When setting the values however, a value of 0, 1 or 2 should be used.

D.4 Get load profile from database
To centralize the control of the workshop, the load profile is stored in a SQL database. To retrieve the information from the database Labview
offers a database connectivity toolkit. This toolkit introduces standard VI's, or functions, that allow connecting to and reading and writing in the database. The first step of retrieving the load profile is opening a connection to the database. This part is shown in Figure D-8.

To open the connection three input arguments are required. A username and password and the database UDL or Unified Data Link file. For information on how to create this file please refer to Appendix B. When all the parameters are valid, the function returns a connection reference to the database. This reference is then connected to the get load profile function, the green 'GET LP from SQL' block in Figure D-5. The block diagram of this function is shown in Figure D-9.

This function starts at the left side with the 'database tools Select Data' function. When given a connection reference and a table name, this function returns the contents of the table. Because Labview does not know what the type of the data is, it is returned as a Variant. The variant values then are converted to integers. This is done with the two 'for' loops. They
iterate through the two dimensional variant matrix and convert each value to integer. The integer values are then stored in a table. This table is returned to the workshop controller function.

D.5 Set load profile to FTP
When the load profile is received from the SQL database and compared with the previous values it is uploaded to Netmaster via an ftp connection. This happens in the 'Set load profile' function (Figure D-10).

The array received from the database does not have the correct format to send it over immediately. The first column needs to be omitted because it contains ID values which are not necessary in the csv file. Furthermore the integer values have to be converted to strings, and at the top of the profile a row with the column names is added. The array can now be written to a temporary CSV file. In the next step the file is send to the Netmaster by using the 'ftp put file' function. This is a Labview function available in the internet toolkit. When the correct ip address, username and password are supplied, it will send the file to the Netmaster.

D.6 Results of first version
The Labview implementation described above was tested in DeMoTec. With the manual mode selected it is possible to change the relays to any desired state. However the relays take a very long time to respond, typically around 30 seconds. This is however a fundamental limit of the Netmaster, as it has only limited processing power.

The automatic mode worked. However the setting of the relays to the auto position would also take around 30 seconds per relay to complete.
Appendix E List of Figures

Figure 2-1: Layout of SCADA system ................................................................. 4
Figure 2-2: Flow chart of ‘set values’ loop .......................................................... 7
Figure 2-3: ‘Set values’ loop block diagram, database mode is active ................. 7
Figure 2-4: ‘Set values’ loop block diagram, manual mode is active ................. 8
Figure 2-5: Reading measurement values ............................................................ 9
Figure 2-6: Interface of SCADA controller ......................................................... 10
Figure 3-1: Load Workshop ................................................................................. 14
Figure 3-2: Workshop control overview ............................................................... 14
Figure 3-3: Reading P Q set values from database ............................................. 15
Figure 3-4: Converting P to boolean array ......................................................... 16
Figure 3-5: New function for setting the relays ................................................... 16
Figure 3-6: Calling the set relay function ............................................................ 17
Figure 3-7: User Interface of workshop control .................................................. 17
Figure 4-1: Overview of Biodiesel control .......................................................... 20
Figure 4-2: Initialization of controller ................................................................. 22
Figure 4-3: First section of main loop ................................................................. 22
Figure 4-4: Second section of main loop ............................................................. 23
Figure 4-5: Reading data with Modbus .............................................................. 24
Figure 4-6: Setting new data with Modbus ......................................................... 24
Figure 4-7: Conversion function of P and Q to S and cos(\phi) .............................. 25
Figure 4-8: User interface of 200 kVA Biodiesel genset control ......................... 26
Figure 4-9: P and Q measurements of 200 kVA Biodiesel Genset ..................... 27
Figure 5-1: Picture of 20 kVA diesel generator .................................................. 29
Figure 5-2: 20 kVA diesel generator control method .......................................... 30
Figure 5-3: User interface for generator control ................................................. 31
Figure 5-4: Phase two of the main program ....................................................... 32
Figure 5-5: Block diagram of ‘get set values’ function ....................................... 32
Figure 5-6: Block diagram of set function ......................................................... 33
Figure 5-7: Block diagram of read function ....................................................... 33
Figure 5-8: Updating database ........................................................................... 34
Figure 6-1: Connection of devices during the experiment ............................... 35
Figure 6-2: Profiles over the course of a day ..................................................... 36
Figure 6-3: PQ profiles used in experiment ....................................................... 36
Figure 6-4: P Q Measurements of generators .................................................... 37
Figure 7-1: Overview of the website structure ................................................... 41
Figure 7-2: Picture of ‘htmltop.php’ when not logged in ..................................... 42
Figure 7-3: Picture of ‘htmlbottom.php’ .............................................................. 42
Figure 7-4: Phase one of the search page .......................................................... 43
Figure 7-5: Layout of phase two of the search page ......................................... 44
Figure 7-6: Example of results page ................................................................. 46
Figure 7-7: Example of info page ..................................................................... 47
Figure 7-8: Example of pdf document ............................................................... 48

Figure B-1: XAMPP Control Panel ................................................................. B-1
Figure B-2: Creating the system DSN ............................................................... B-3
Figure D-1: Manual Control ...................................................... D-1
Figure D-2: Block diagram of manual control .......................... D-2
Figure D-3: Front panel of automatic control ............................ D-2
Figure D-4: Block diagram of automatic control ....................... D-3
Figure D-5: Workshop main control block diagram .................... D-4
Figure D-6: Connecting to OPC variables ................................. D-4
Figure D-7: Set relays function .............................................. D-5
Figure D-8: Opening database connection ............................... D-6
Figure D-9: Read load profile function block diagram ............... D-6
Figure D-10: Set load profile function block diagram ............... D-7
Appendix F  List of Tables

Table 2-1: Example of table in database .................................................. 4
Table 2-2: Example of PQ profile ................................................................. 5
Table 2-3: Layout of CSV measurement file ............................................... 9
Table 3-1: Available loads in the workshop .............................................. 13
Table 3-2: Example load profile ................................................................. 14
Table 4-1: Serial connection settings ......................................................... 20
Table 4-2: Structure of Modbus message ................................................... 21

Table A-1: Used VI’s for SCADA Controller ............................................. A-1
Table A-2: Used VI’s for 20 kVA Variable Speed genset Controller ............ A-1
Table A-3: Used VI’s for Workshop Controller ......................................... A-2
Table A-4: Used VI’s for 200 kVA Biodiesel genset Controller .................. A-2
Table B-1: Login details for database admin utility .................................... B-2
Table B-2: Login details for normal database user .................................... B-2
Table C-1: Specifications of CSV file names ............................................. C-1
Appendix G    List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Ampere</td>
</tr>
<tr>
<td>CSV</td>
<td>Comma Separated Value</td>
</tr>
<tr>
<td>DAQ</td>
<td>Data Acquisition Device</td>
</tr>
<tr>
<td>DeMoTec</td>
<td>Design-Centre for Modular Supply Technology</td>
</tr>
<tr>
<td>DG</td>
<td>Distributed Generation</td>
</tr>
<tr>
<td>DS</td>
<td>Data Socket</td>
</tr>
<tr>
<td>OPC</td>
<td>OLE for Process Control</td>
</tr>
<tr>
<td>P</td>
<td>Active Power</td>
</tr>
<tr>
<td>PF</td>
<td>Power Factor</td>
</tr>
<tr>
<td>PHP</td>
<td>PHP: Hypertext Preprocessor</td>
</tr>
<tr>
<td>Q</td>
<td>Reactive Power</td>
</tr>
<tr>
<td>S</td>
<td>Apparent Power</td>
</tr>
<tr>
<td>SCADA</td>
<td>Supervisory Control and Data Acquisition</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>UDL</td>
<td>Unified Data Link</td>
</tr>
<tr>
<td>URL</td>
<td>Unified Resource Locator</td>
</tr>
<tr>
<td>V</td>
<td>Volt</td>
</tr>
<tr>
<td>VI</td>
<td>Virtual Instrument</td>
</tr>
</tbody>
</table>
Appendix H References


Development of Active and Reactive Power Control Systems in DeMoTec

Justin Au-yeung

Internship Report:

Work Term: 1 December 2007 – 30 April 2008

Supervisor in ISET:
M. Braun

Supervisor in TU/e:
J. Myrzik

Submission Date: 17 April 2008
# Table of contents

ABSTRACT ......................................................................................................................... 4  

1. INTRODUCTION ........................................................................................................... 5  

2. CENTRAL CONTROL SYSTEM .................................................................................... 6  

3. 100 KVA MULTI-PV .................................................................................................. 8  
   3.1. INTRODUCTION ........................................................................................................ 8  
   3.2. 100 KVA MULTI-PV LABVIEW CONTROL APPLICATION ................................ 9  
      3.2.1. GRAPHICAL USER INTERFACE ........................................................................ 10  
      3.2.2. 100 KVA MULTI-PV P AND Q CONTROLLER ................................................ 11  
      3.2.3. FORMAT 100 KVA MULTI-PV PROTOCOL ..................................................... 13  
      3.2.4. SERIAL READ AND SERIAL WRITE ................................................................. 15  
      3.2.5. SPLIT RECEIVED STRING .............................................................................. 17  
   3.3. MEASUREMENTS AND RESULTS ........................................................................... 17  
   3.4. CONCLUSIONS AND RECOMMENDATIONS ......................................................... 22  

4. 15 KVA GENSET .......................................................................................................... 23  
   4.1. INTRODUCTION ........................................................................................................ 23  
   4.2. 15 KVA GENSET LABVIEW CONTROL APPLICATION .................................... 26  
      4.2.1. GRAPHICAL USER INTERFACE ........................................................................ 26  
      4.2.2. SYNCHRONIZATION ....................................................................................... 27  
      4.2.3. 15 KVA GENSET P AND Q CONTROLLER ..................................................... 30  
   4.3. MEASUREMENTS AND RESULTS ........................................................................... 37  
   4.4. CONCLUSIONS AND RECOMMENDATIONS ......................................................... 39  

5. WEBSITE INTERFACE DRLAB DATABASE .................................................................. 40  
   5.1. INTRODUCTION ........................................................................................................ 40  
   5.2. OVERVIEW WEBSITE STRUCTURE .................................................................... 40  
   5.3. EQUIPMENT AND FACILITIES UNITS ................................................................. 42  
      5.3.1. ADDING EQUIPMENT OR FACILITIES ............................................................ 42  
      5.3.2. DELETING EQUIPMENT OR FACILITIES ....................................................... 46  
      5.3.3. CONCLUSIONS AND RECOMMENDATIONS .................................................. 47  
   5.4. INTERCONNECTIONS REQUIREMENT UNITS ....................................................... 48  
      5.4.1. ADDING INTERCONNECTION REQUIREMENTS .............................................. 48  
      5.4.2. DELETING INTERCONNECTION REQUIREMENTS .......................................... 49  
      5.4.3. SEARCH INTERCONNECTION REQUIREMENTS ................................ .......... 49  
      5.4.4. CONCLUSIONS AND RECOMMENDATIONS .................................................. 51  
   5.5. CONCLUSIONS AND RECOMMENDATIONS ......................................................... 51
6. CONCLUSIONS AND RECOMMENDATIONS

Appendix A  Configuration A2000 power meter for 15 kVA Genset measurements ............... 53
Appendix B  Building a running application for the 15 kVA Genset and 100 kVA Multi-PV .... 58
Appendix C  List of variables .................................................................................. 67
Appendix D  Load cabinets ....................................................................................... 68
Appendix E  15 kVA Genset hardware setup ............................................................... 74
Appendix F  List of Abbreviations ............................................................................. 82
Appendix G  List of tables ......................................................................................... 83
Appendix H  List of figures ....................................................................................... 84

REFERENCES ........................................................................................................... 86
Abstract

The FENIX project is aiming to enable distributed energy resources based systems to become the solution for the future European electricity supply. For the FENIX project to enable distributed energy resources into the European electricity supply system, the development of intelligent interfaces for the distributed energy resources is required. Therefore one of the FENIX project partners “ISET” is developing intelligent interfaces for the distributed energy resources in the DeMoTec laboratory using the graphical measurement program Labview. The Labview control applications for two distributed energy resources located in DeMoTec have been developed during the internship term in “ISET”. The Labview control applications control the “15 kVA generator set” and the “100 kVA multifunctional photovoltaic inverter” towards the active and reactive power set values which can be received from a central SQL database or set manually. The central SQL database comprises the set and measurement values which are used for analysis.

The resulting Labview control applications are:

- **15 kVA generator set**: The Labview control application sends with a data acquisition device set voltages to the generator excitation current input and the driving motor torque input. These set voltages influence the active and reactive power. Besides sending set voltages, it receives active and reactive power measurement values from an A2000 power measurement device. By using these measurement values a “proportional integrative control” is implemented to control the “15 kVA generator set” towards the set values.

- **100 kVA multifunctional photovoltaic inverter**: The Labview control application serially sends set values using a user defined protocol to the “100 kVA multifunctional photovoltaic inverter”. Dspace control system controls the “100 kVA Multifunctional Photovoltaic inverter” towards those set values. After sending the set values, the control application serially receives the measurement values from the Dspace control system.

Both control systems use a user-friendly graphical user interface to display in a chart and via indicators the set and measurement values.

In the resulting system, the Labview control applications can control the “15 kVA generator set” and the “100 kVA multifunctional photovoltaic inverter” with respect to active and reactive power.

Beside the Labview control applications, a user-friendly website was built to add, delete or search within the DERlab database which contains the information about equipment, facilities and services from different partners. DERlab is the European Network of Excellence of independent laboratories, working in the area of the integration of distributed energy resources into electricity grids and the preparation of related standards and test procedures.
1. Introduction

Presently, the end user still consumes electrical energy from the main power supply and not yet discovered all the benefits by using distributed energy resources. Controlling reactive power, grid voltage control, reducing transportation losses and reducing environmental pollution are some of the benefits using distributed energy resources instead of the main power supply. FENIX project is aiming to enable distributed energy resources based systems to become the solution for the future European electricity supply. Therefore a technical architecture and commercial framework have to be designed and demonstrated. One of the FENIX partners “ISET” is developing intelligent interfaces for the distributed energy resources in the DeMoTec laboratory using the graphical measurement program Labview. The Labview control applications for two distributed energy resources have been developed during the internship term.

One of the Labview control application should control the 15 kVA Genset in terms of active and reactive power using a data acquisition unit and an A2000 measurement device. The Labview control application for the “100 kVA Multi-PV” should control the active and reactive power via a serial port. And each separate control system should have a user friendly graphical user interface with the possibility to control the active and reactive power directly or via the central SQL database comprising the active and reactive power set values. Furthermore the active and reactive power measurements should be presented using charts also known as real-time graphs.

Besides the integrated Labview control system a user friendly website should also be implemented which can add, remove and search inside the DERlab database which contains all the equipment, facilities and services from all partners. DERlab is the European Network of Excellence of independent laboratories, working in the area of the integration of distributed energy resources into electricity grids and the preparation of related standards and test procedures.

In chapter 2, “the central control system” is described. The implementation of the “100 kVA Multi-PV” is presented in chapter 3. Chapter 4 describes the 15 kVA Genset control system. The website interface for the DERlab database is explained in chapter 5. Finally, chapter 6 presents the conclusions and recommendations for the Labview control applications and the website interface.

Preliminary investigations are done for the load cabinets and presented in Appendix D.
2. Central control system

In this section the central control system for the developed active and reactive power control systems is given and shown in Figure 2-1.

As shown in Figure 2-1, the central server is receiving active and reactive power set values from a user defined load profile. The set values are stored in the central SQL database where it possible is to visualize and monitor the measurement results from each subsystem using a central server. Refer to [2] for more information about the central server and the central SQL database.

The Labview applications, the \textit{workshop}, the \textit{20 kVA Variable speed Genset} and the \textit{Biodiesel Genset} on the left side of Figure 2-1, are also explained in [2]. The Labview applications, the \textit{100kVA Multi-PV} and the the \textit{15 kVA Genset} which are explained in this report are on the right side of Figure 2-1.

The \textit{100kVA Multi-PV} is controlled by a Labview application which retrieves the set values from the central SQL database or manually. These set values are sent over via a serial port to the \textit{100 kVA Multi-PV} and the Dspace controller controls...
the 100 kVA Multi-PV towards these set values. In the meantime active and reactive power measurement values are serially received and displayed in a chart or via indicators.

The 15 kVA Genset is controlled by a Labview control application which drives a data acquisition unit. The control application retrieves the active and reactive power set values from the central SQL database or manually. Then the control application controls the 15 kVA Genset towards the set values by adjusting the 15 kVA Genset "driving motor torque voltage" and "synchronous generator excitation current" using the data acquisition unit.

The A2000 measurement device is used to retrieve active and reactive power measurement values to control the 15kVA Genset towards the set values.
3. 100 kVA Multi-PV

The control of the 100 kVA Multi-PV using a serial connection is described in this section. First of all an overview of the 100 kVA Multi-PV controller is given. The 100 kVA Multi-PV Labview control application is described in section 3.2. Section 3.3 presents the measurements and results. The last section contains the conclusions and recommendations for the 100 kVA Multi-PV.

3.1. Introduction

An overview of the control system for the 100 kVA Multi-PV is given in Figure 3-1. Figure 3-2 shows the internal connections for the 100 kVA Multi-PV.

First of all Labview will retrieve the set values for active power "P" and reactive power "Q" from the central SQL database or from the manual input. The Labview application sent the set values towards the 100 kVA Multi-PV via a RS232 serial
communication, the control system (Dspace model) inside the 100 kVA Multi-PV regulate the P and Q towards the set values. Between the periods of sending new set values, Labview will retrieve from the 100 kVA Multi-PV the current P and Q measurement values. These values are forwarded to the Labview application via the same RS232 serial communication port. In the Labview control application, the measurement values are retrieved and displayed in a chart or via indicators. Finally the P and Q measurements values are saved in the central SQL database.

Information about the 100kVA Multi-PV project can be found on the website of Multi-PV project [3].

3.2. 100 kVA Multi-PV Labview control application

The 100 kVA Multi-PV Labview control application is based on stacked sequence implementation; sequences are programmed frames which follows each other sequential. The implemented sequences are initialization, write Multi-PV and read Multi-PV and described in this subsection.
3.2.1. **Graphical user interface**

The "100 kVA Multi-PV Labview control application" graphical user interface is shown in Figure 3-3.

The graphical user interface consists of "Initialization" on the left side and on the right side "Measurements 100 kVA Multi-PV". In the initialization part, the data link to the central SQL database is selected using the "Path UDL file". And serial communication is used to control the 100 kVA Multi-PV and using the front panel option "VISA Resource Name" the connected serial port can be selected. The user can select with the switch "Manual or Automatic" the method to retrieve set values, by selecting the switch "Automatic" the application retrieves active and reactive power set values from the central SQL database. The led "Value database out of range" turns on when the set values from the central SQL database is out of control range. The other option "Manual" enables the possibility to manual filling in the set values via the two numeric control displays, "Psetman (kW)" and "Qsetman (kVAR)". Stopping the 100 kVA Multi-PV without closing the program is possible using the "Start/Stop machine" switch. After finishing the initialization the application can run using the "Start Program" switch. The "Close Program" switch stops the 100 kVA Multi-PV and closes the 100 kVA Multi-PV application. Below the "Initialization" is located the active and reactive power chart. This chart displays the active and reactive power measurements together with the set values. The indicators on the right side shows the active power ("Pset") and reactive power ("Qset") set values next to the active power ("Pmeas"), reactive power ("Qmeas"), apparent power, power factor and the three phase RMS terminal voltage ("Vmeas") measurement values.
3.2.2. 100 kVA Multi-PV P and Q controller

The Labview implementation for the 100 kVA Multi-PV is shown in Figure 3-4.
Only the first sequence (On the right side of the figure) is visible in this figure, which contains the initialization of the 100 kVA Multi-PV. First of all, all the set values are retrieved from the central SQL database and from the manual filled in set values. When the user selects “Manual” on the front panel option “Manual or Automatic”, the manual filled in set values are forwarded. The selection “Automatic” tends to forward the set values from the central SQL database. These values are forwarded to the first sequence “Initialization”, this sequence checks whether the set values are within the allowed limits. The allowed limits are for active power between 0 kW - 40 kW and for reactive power between 40 kVAr injecting inductive reactive power and -40 kVAr injecting capacitive reactive power specified by D. Geibel. When the set values exceed the allowed limits, the Labview control application changes the set values into the maximum or minimum set values. Then the set values are saved as global variables, global variables can be retrieved and changed in the whole Labview application.

After the first sequence is completed, the second sequence “Write Multi-PV” shown in Figure 3-5 starts.

This sequence forwards the set values and the selected option “start/stop machine” to the Format Multi-PV function block; this function block encapsulates the set values into the user defined string protocol. The Format Multi-PV function block is explained in section 3.2.3

The user defined string protocol is forwarded towards the Serial Write function block; this block sends it serially towards the 100 kVA Multi-PV, which is described in detail in section 3.2.4. When the user enables “Close program”, the “Multi-PV control application” resets the “100 kVA Multi-PV” by setting the set values back to zero.
The last sequence “Read Multi-PV” is shown in Figure 3-6, this sequence receives serially the measurement values from the 100 kVA Multi-PV. Firstly the user defined protocol string is received using the function block Serial Read; described in 3.2.4. Then the user defined protocol string splits into smaller parts containing the measurement values; “active power”, “reactive power” and “three phase RMS voltage” using the function block Split Receive String. Section 3.2.5 deals with the function block Split Receive String.

The measurement values are saved as global variables, and used for the calculation of apparent power and power factor. The measurement values and the calculated apparent power and power factor are saved in the central SOL database using one SOL update sentence.

These three sequences are inside an infinite while loop and can only be stopped using the switch “Close Program”

### 3.2.3. Format 100 kVA Multi-PV protocol

The 100 kVA Multi-PV user defined protocol is described in this subsection. The sent protocol is based on 25 bytes, the protocol starts with STX and ends with ETX. The protocol layout and an example is shown below.

<table>
<thead>
<tr>
<th>STX</th>
<th>7 bytes Pset</th>
<th>7 bytes Qset</th>
<th>Start/Stop Machine</th>
<th>ETX</th>
</tr>
</thead>
<tbody>
<tr>
<td>STX</td>
<td>+006003</td>
<td>-020013</td>
<td>1</td>
<td>ETX</td>
</tr>
</tbody>
</table>

Table 1 100 kVA Multi-PV protocol layout
Semicolons are used in this protocol to distinct between different parameters. Stopping the 100 kVA Multi-PV is possible using the start/stop machine bit (1->start and 0->stop). Positive active power values refers to producing active power and consuming active power is displayed with a negative sign. The positive reactive power values refers to injecting capacitive reactive power and the negative reactive power values means injecting inductive reactive power. The second row of Table 1 shows an example whereby $P_{\text{set}} = 6.003$ kW, $Q_{\text{set}} = -20.013$ kVA and option start machine are transferred to the 100 kVA Multi-PV. The Labview implementation for this protocol is shown in Figure 3-7.

Unfortunately the $P_{\text{set}}$ and $Q_{\text{set}}$ are not always 7 bytes due to the format of the set values from the SQL database. That's why it is required to adjust the size of the information; this is implemented in Figure 3-8. Firstly the sign of the information is determined using the bigger or equal with zero statement. And then depending on the size of the active and reactive power, zeros are placed in front of the numbers to achieve in total of 7 bytes. This is realized using a case construction conditioned by the size of the active and reactive power set values. For example, when $P_{\text{set}}$ is smaller then 10 this function puts 2 zeros in front of the number. After achieving 7 bytes for $P_{\text{set}}$ and $Q_{\text{set}}$, the information is used in the composition of the user defined protocol string.
3.2.4. Serial read and Serial write

The user defined protocol is sent towards the 100 kVA Multi-PV via a RS232 serial port. The physical connection between the 100 kVA Multi-PV and the destination computer consists of a serial cable and an USB to serial converter. The Labview implementation for sending serial data towards the 100 kVA Multi-PV is presented in Figure 3-9. The serial settings baud rate, data bits, parity bits, stop bits, flow control and enable termination char are configured as shown in Figure 3-9. The VISA Resource name defines the connected serial com port and can be selected in the Front Panel.

After completed the serial settings, the write visa function block performs a write operation towards the 100 kVA Multi-PV using the user defined protocol named as “Serial Input”. At the end of the writing the serial connection is closed. Besides the normal serial write functionality, this function block consists also error control. The error control checks whether the serial com port is connected and displays an error if this is not the case.
Configure baud rate, data bits, parity, stop bits and flow control for serial port.

Figure 3-9 Serial write configuration

The serial read configuration has almost the same serial settings as the serial write configuration as shown in Figure 3-10. Only extra setting is that the user needs to define the total bytes to read (33 bytes) per serial read cycle.

Figure 3-10 Serial read configuration
3.2.5. **Split received string**

As shown in Figure 3-11, the received string from the serial connection is split into several parts. This function block splits the "active power" (Pmeas), "reactive power" (Qmeas), and the "three phase terminal RMS voltage (Vmeas)" measurements from the input received string. Besides these measurement values the status of the machine (On/off machine) is retrieved from the received string.

![Figure 3-11 Split received string](image)

3.3. **Measurements and results**

First of all the serial communication can be established, the 100 kVA Multi-PV can be set into certain active and reactive power range. This range is currently for active power between 0 and 40 kW and for reactive power between -40 kVAR injecting capacitive reactive power and 40 kVAR injecting inductive reactive power.

The active power measurements are made according to a daily load profile from a real photovoltaic situation. In a real photovoltaic situation, active power is only produced when it is still light as shown in Figure 3-12. In the afternoon, there is a shadow around the photovoltaic which caused the active power dip. The reactive power measurements are setup according to the total amount of loads connected to the grid over 24 hours. Normally, mostly inductive loads (lighting, heating, motors, transformers, etc.) are connected with the grid in the afternoon (175 to 400 seconds) as shown in Figure 3-14. The generators inject capacitive reactive power to compensate the inductive loads. But around the night (0 to 175 seconds and 400 to 500 seconds), there are less inductive loads connected, thus the required capacitive reactive power decreases towards inductive reactive power.
Figure 3-12 Active power measurement 100 kVA Multi-PV
Figure 3-13 Zoom in active power measurement 100 kVA Multi-PV
Figure 3-14 Reactive power measurement 100 kVA Multi-PV
The measurements show that it is possible to control the active and reactive power in a certain range of values. Unfortunately currently the maximum range is limited to 57 kVA.
The 100 kVA Multi-PV Dspace controller requires one second time to rise/decrease 10 kW/kVAr active and reactive power. The time it takes to increase active and reactive power towards a certain set value is shown in Figure 3-13 and Figure 3-15. The bottleneck in the Dspace controller delay is the control range of reactive power which is larger than the control range of active power, causing a maximum delay of 8 seconds corresponding to 80 kVAr rise/decrease. The Labview serial read has a delay of 2 seconds because the serial write has one second delay and the serial read has also one second delay. The maximum delay is ten seconds before the 100 kVA Multi-PV is controlled to the set value.

3.4. Conclusions and recommendations
Starting with the main conclusion; the Labview control applications works according to all specifications. The 100 kVA Multi-PV can be controlled in terms of active and reactive power. It can also measure active and reactive power values, besides measuring it can display the values on a user-friendly chart. And in the application it is possible to choose the option manually retrieving set values or automatically retrieving set values. The 100 kVA Multi-PV application runs also on a separate computer in DeMoTec. Summarized: the remote control of the 100 kVA Multi-PV works and can be used for the laboratory activities in the FENIX project.
4. 15 kVA Genset

Section 4 describes the Labview control system for the 15 kVA Genset. Starting with an overview of the control system and followed by some background information about the driving motor and synchronous generator. The other sections explain the different parts of the Labview control application including monitor and visualize active and reactive power measurements. In Appendix A information is shown about the connections and configuration of the A2000 box. Creating the 15kVA Genset application (.exe) is explained in Appendix B. The 15 kVA Genset hardware setup containing the configuration and the wiring is presented in Appendix E.

4.1. Introduction

An overview of the 15kVA Genset control system is shown in Figure 4-1, whereby a connection is made with the central SQL database to retrieve P and Q set values. Another option is to import set values manually via the graphical user interface. The Labview application converts these values into controllable digital torque voltage and excitation current to control the uDAQ lite (Data acquisition unit) component.

This uDAQ lite component converts the digital values into analog signals. The analog signals control the excitation current and torque voltage of the 15 kVA Genset which influence the active and reactive power. The A2000 measurement device measures active and reactive power values which are used to control the 15kVA Genset. Besides the active and reactive power measurement, the A2000 measurement device receives also the three phase RMS voltage.
Figure 4-2 shows the practical setup of the 15kVA Genset.

The driving motor can control the active power by adjusting the torque. The equivalent per-phase representation for the driving motor is shown in Figure 4-3. On the left side of Figure 4-3 is shown the per phase representation of the driving motor. Rs is the winding resistance and Ls is the leakage current of the winding. Eag is the air gap voltage which is connected with the air gap flux. This flux is created by the magnetizing component Im which origin is from Is. To produce torque in the driving motor there is interaction needed for the air gap flux and the rotor currents. In the 15 kVA Genset it is possible to control the voltage Vs as shown in Figure 4-3. This voltage can be expressed inside motor equations. Refer to [4] for more information about these motor equations.

\[
V_s = E_{ag} + (R_s + j \cdot 2 \cdot \pi \cdot f \cdot L_m) \cdot I_s \tag{4-1}
\]

\[
\frac{d \phi_{ag}}{dt} = -E_{ag} \tag{4-2}
\]

\[
T_{em} = \text{const} \cdot \phi_{ag} \cdot I_r \tag{4-3}
\]

\[
P_{em} = T_{em} \cdot \omega \tag{4-4}
\]

As mentioned in equation 4-2 the flux \( \phi_{ag} \) and the voltage \( E_{ag} \) are interlinked. The torque can be adjusted using the input torque voltage \( V_s \) which is linked with the flux \( \phi_{ag} \).
As shown in equation 4 the active power $P_{em}$ is related with the torque $T_{em}$. Based on these assumptions it could be concluded that by increasing the torque voltage $V_s$ the active power will increase and vice versa.

In the 15 kVA Genset it is possible to control the synchronous generator excitation current which links with the reactive power. This link is better explained using Figure 4-4 which shows the phasor diagram of the synchronous generator.

Following from Figure 4-4 the following equations can be derived

\[ S = VI \quad (4-5) \]

\[ P_e = VI \cos \phi = \frac{E_q V}{X_d} \sin \delta \quad (4-6) \]

\[ Q_e = VI \sin \phi = \frac{E_q V}{X_d} \cos \delta - \frac{V^2}{X_d} \quad (4-7) \]

As shown in Figure 4-4 the excitation current $I$ and the terminal voltage $V$ determine the angle $\phi$. This angle controls the synchronous generator active and reactive power as described with the equations 4-6 and 4-7. Changing excitation
current I and keeping Active power $P_e$ and terminal voltage constant changes the Reactive power $Q_e$ in terms of sign and amplitude. The excitation current can be controlled by applying a voltage to the excitation current input. For more information about the synchronous generator principle refer to [5].

4.2. 15 kVA Genset Labview control application

The 15 kVA Genset Labview control application is based on three parts initialization and synchronization, controlling and monitor the measurements results. Those three parts are described in this subsection.

4.2.1. Graphical user interface

Figure 4-5 shows the graphical user interface of the 15 kVA Genset control application. On the left side of the front panel, the initialization is shown whereby the possibility to select the path of the udl (database) file can be found. Also some other values need to be selected like choosing the parameters for the active and reactive power when the manual configuration is selected. The automatic configuration shall automatically retrieve the active and reactive power set values from the central SQL database. When the values exceed the limits then the values will be limited towards the maximum or minimum set value and the led value database out of range turns on.

After filling in all parameters the Start program vertical slide switch can be enabled. This will start the synchronization and after synchronization is achieved the Stop synchronization vertical slide switch can be enabled. This will allow a popup to appear whereby the option $P$ and $Q$ measurement values should be selected. This allows the control systems to control the 15 kVA Genset to the desired active and reactive power values. Those values are shown via indicators and also via an active and reactive power chart.
4.2.2. Synchronization

The 15 kVA Genset can only control active and reactive power when the system is synchronized to the grid. That's why inside the 15 kVA Genset there is a synchronizer device available as shown in Figure 4-6. This device detects the grid voltage and frequency and connects the 15 kVA Genset if the deviations of voltage, frequency and phase are within allowed limits.

Before synchronization can start, the 15 kVA Genset front panel settings (Figure 4-7) have to be configured. Firstly, Auto is enabled to automatically synchronize the 15 kVA Genset with the grid. Then the SG Betrieb setting is enabled because in the setup the synchronous generator is used instead of the asynchronous generator.
After configuring correctly these settings, the *start program* in the graphical user interface can be enabled whereby a popup is generated as shown in Figure 4-8. The user can select one of the three options; by *selecting stop everything* the user can abruptly end the program.

In this section the *synchronization* is selected, this allows the Labview application as shown in Figure 4-9 to run.
Figure 4-9 Synchronization 15 kVA Genset
The intention is to synchronize the 15 kVA Genset with the 400V and 50Hz grid, whereby the values to control the torque voltage and excitation current towards synchronization are known. Thus implementing a small synchronization control, automatic synchronization can be achieved.

Starting from the left side, the DAQ board is selected using the function Sel board. As explained in Appendix E, the excitation current is dependent on the excitation voltage. Because the excitation current needs a certain excitation voltage before the excitation current can be controlled (current source), that’s why the 5V excitation voltage digital output is enabled in the top of the figure.

By making the excitation current constant and risen slightly the torque voltage around the desired values, the voltage and frequency can arise towards 400V and 50 Hz. Unfortunately if the system is not synchronized after reaching 50 Hz, the 15 kVA Genset will get unstable. That’s why in the middle of the figure, a security is built to set the Torque voltage and Excitation current back to lower values. This tends to swing the voltage and frequency around 400V and 50Hz and after a specific time, synchronization is achieved. The next subsection describes the “15 kVA Genset P and Q controller” which controls the active and reactive power towards the active and reactive power set values.

4.2.3. 15 kVA Genset P and Q controller

In this subsection the “15 kVA Genset P and Q controller” Labview implementation is explained. The “15 kVA Genset P and Q controller” consists of “retrieving the specific active and reactive power set values”, “controlling the 15 kVA towards these set values”, “displaying these values in a user friendly chart” and “saving the values into the central SQL database”.

Starting with the explanation of retrieving set values from the SQL database or retrieving manually filled in set values. In Figure 4-10, a small part of the controller is shown where the set values are retrieved. First of all the program can get the values from the central SQL database via a simple SQL select statement.

• Select 'pset', 'qset' from 15kva.

Before the set values can be forwarded, it is checked whether the values are exceeding the limit values. When this is the case, the values are changed into the maximum or minimum values and the led value database out of range turns on.

The checked values are forwarded to a case construction, which save the SQL database values or the manually filled in values (Psetman, Qsetman) depending on the option “set values settings” (Manual or automatic) into local variables. The “SQL or Manual getting set values” is inside an infinite while loop and can only be stopped by pressing stop 15 kVA.
The second step is to control the 15 kVA Genset towards these set values. This will be realized by two independent parallel PI controllers which control the active power and the reactive power. The following example gives an insight of the chosen PI control design.
The "PI control system" in Figure 4-11 is controlling the active power (Torque voltage) towards the active power set value. Because the control system has an integrative effect it will go to the final active power set value (Pset). The integrator gain and the proportional gain for the active and reactive power control are determined and optimized by experiments and shown in Figure 4-12.
The output limits are placed to restrict the active and reactive power rising to uncontrollable excitation current or torque voltage values. This prevents the PI control becoming unstable.

Negative values of reactive power correspond to leading power factor operation or "inductive reactive power operation".

The synchronous generator injects more capacitive reactive power (positive sign) when the applied excitation current rise and injects less capacitive reactive power when the excitation current decrease until it reaches zero capacitive reactive power. When the excitation current still decreases, the synchronous generator starts to inject inductive reactive power (negative sign). Decreasing more the excitation current caused the inductive reactive power to increase.

Considering this relationship, the PI control tries to decrease the capacitive reactive power or increasing the inductive reactive power by applying a lower excitation current and vice versa.

The parallel control loop is infinite and can only be stopped by pressing the stop 15 kVA button.

In parallel to this control system, the measured values are saved in the SQL database and also shown via an active and reactive power chart. The Labview
implementation for saving the measurement values in the central SQL database is shown in Figure 4-13. By using one update sentence the measurement values active power, reactive power, three phase RMS voltage, speed, apparent power and power factor are saved in the central SQL database.

Figure 4-13 Updating the measurement values in SQL database
The chosen set values (SQL or Manual) are plotted together with the measurement values into a chart as shown in Figure 4-14. The displaying of the measurement values and retrieving measurement values are inside an infinite loop and can only be stopped by pressing the stop 15kVA button.

There is a function block called A2000 P and Q and the Labview implementation is shown in Figure 4-15. The uDAQ retrieves the A2000 analog output signals but the range of the analog outputs are not the same as the input of the uDAQ, so some conversion are required as shown in Figure 4-16, Figure 4-17 and Figure 4-18.
Figure 4-15 Receiving A2000 measurement values

Figure 4-16 Conversion to Reactive power values

Figure 4-17 Conversion to Active power values
The range for the uDAQ is between -5V and 5V and the range for the A2000 is between 0 and 10V. Further the maximum range for the terminal voltage is between 0 and 500V and the range for the P and Q is between -20000 and 20000. This implies that an offset of 5V is used to define negative values and a factor 2000 and 135 is used to gain the correct active power, reactive power and voltage values.

4.3. Measurements and results

The 15kVA measurements results are shown in Figure 4-19 and Figure 4-20. Time is measured in seconds but it simulates the load profile which is 24 hours, implicates that 250 seconds refer to 24 hours. The chosen active and reactive power set values are according to the real situation, the 15 kVA generator set simulates a power plant. Normally, a power plant generates constantly the same amount of active power as shown in Figure 4-19. Positive values of active power correspond to producing active power. The reactive power measurements are made according to the total amount of loads connected to the grid. Mostly inductive loads (lighting, heating, motors, transformers, etc.) are connected with the grid between 120 and 250 seconds which simulates the afternoon as shown in Figure 4-20. The generator injects capacitive reactive power to compensate the inductive loads. But around the night or between 0 and 120 seconds, there are less inductive loads connected, thus the required capacitive reactive power decreases.
Figure 4-19 15 kVA active power measurement

Figure 4-20 15 kVA reactive power measurement
The charts show that the active and reactive power are coupled. When the active power changes from 0 kW to 6 kW around time is 5 seconds, the reactive power reacts by changing from injecting 2 kVAr "inductive reactive power" into injecting 5 kVAr "inductive reactive power" even when the set value is still injecting 2 kVAR inductive reactive power. The delay for controlling the active and reactive power towards the set value varies between 1 and 4 seconds.

The maximum consuming active power is around -2.5 kW and maximum producing active power is around 10 kW. The reactive power range is between injecting 4 kVAr capacitive reactive power and injecting 4 kVAr inductive reactive power. The combination of active and reactive power is limited to the maximum apparent power, 15 kVA.

4.4. Conclusions and recommendations

The 15 kVA Genset can be controlled in terms of active and reactive power. The 15 kVA Genset functions according to the specifications and can be controlled by a normal computer via a 15kVA application (.exe). This application works also together with the central SQL database or by manual filling in the values. A possible extension to the active and reactive power control is to implement a control system to stabilize the grid in terms of voltage and frequency.
5. Website interface DERlab database

5.1. Introduction

As a result of the difficulty to add, delete and search equipment, facilities or interconnection requirements inside the SQL (phpMyAdmin) database, there is demand for a user-friendly web interface.

The DERlab website consists of an interconnection- and equipment, facilities-part. These units are linked with a DERlab SQL database and controlled by the DERlab website. DERlab is the European Network of Excellence of independent laboratories, working in the area of the integration of distributed energy resources into electricity grids and the preparation of related standards and test procedures.

First of all it is possible to add new equipment, facilities or interconnection requirements inside the DERlab database using two different ways.

1. Online form
2. Excel datasheet (Using the given excel sheet on the website not available for interconnection requirement)

The filled in information will be checked on errors before saving in the DERlab website.

And it is also possible to remove the equipment, facilities or interconnection requirements from the DERlab database using the DERlab website form.

Thirdly, there is also a possibility to search for equipment, facilities or interconnection requirements within the DERlab database. The search function contains searching with keyword or searching on subject and the found information could be ordered on category.

The entries found in equipment, facilities can be saved in a PDF format for usage (Not available for interconnection requirements).

In section 5.2, the website structure will be explained. And in chapter 5.3, the equipment and facilities part will be explained. The interconnection requirements are presented in section 5.4

Finally in chapter 5.5 the results and recommendations for this website are discussed.

5.2. Overview website structure

The global website structure is shown in Figure 5-1.
The *index.php* is the main module for the website interface. As shown on the left side of the figure the DERlab website is divided into two sub trees one included login module and one without login module, the search modules don't require a login module.

Starting with the modules without login on the left side of the figure, the *search.php* is the form to search inside the equipment and facilities database. And after filling in the required search information inside the *search.php*, the result will be printed using the *result.php* module. For more information about specific equipment or facility the *info.php* is used and the *pdf.php* will convert this information into a PDF file.

Secondly, in the DERlab website there exists also a search function for interconnection requirement. The form for the interconnection requirement is produced by the *searchinterconnect.php* and the results are printed out by *findinterconnect.php*.

On the right side of Figure 5-1 the modules included login function are shown. It is important to generate a global login module to check whether the user has the rights to add or delete information from the database. The *login.php* will produce a form to login and *checklogin.php* will check the login information. After login into the DERlab website there are possibilities to add new equipment and facilities into the database. Firstly a form (*insertchoose.php*) will appear where you can choose between manual adding new equipment or facilities or using a
CSV file to upload new equipment and facilities. The form to add manual new equipment or facilities is generated by formderlab.php. In insertdatabase.php the information filled in the form will be added into the DERlab database. Further it is possible using the exceltosql.php to add new equipment or facilities by uploading a CSV file with all the information.

In the second sub tree the user can add new interconnection requirements into the database. In the maininter.php a form is generated for interconnection requirements. And in interconnection.php the information retrieved from the form will be inserted into the DERlab database.

The third three consists of modules to remove equipment or facilities information from the database. In delform.php a form will be produced to gain information about which equipment and facilities need to be deleted. In confirmdel.php a confirmation form is shown prohibiting some mistakes are made when someone is deleting information. At the end the delfunc.php removes the filled information from the DERlab database. In the last sub tree, information from the interconnection requirements can be removed. The form for deleting interconnection requirements is shown by delinform.php. Following is a confirmation form to ask again if you really are intending to delete the information from the interconnection requirements database. At the end the delinterfunc.php deletes the interconnection requirement from the database.

5.3. Equipment and Facilities units

In this chapter the equipment and facilities units are discussed. First of all the add functionality using a manual fill in form or a CSV file is explained in section 5.3.1. And the explanation of the delete functionality is shown in section 5.3.2. The search functionality for equipment and facilities is not explained in this chapter, refer to [2] for more information. In section 5.3.3 there are some conclusions and recommendations for the equipment and facilities part described.

5.3.1. Adding equipment or facilities

The manual fill in form to add equipment or facilities is explained in this subsection. First of all, the menu is different between a super user and a normal user. The difference is that the super user can add equipment everywhere inside the DERlab database and the normal user may only add equipment or facilities inside his table. That's why the menu for the super user has an extra option to choose between companies where he would like to add equipment or facility. The form for the specific category which is selected: Stationary equipment, Mobile equipment or Facilities is shown in Figure 5-2, Figure 5-3 and Figure 5-4. Note that in Figure 5-2, there are drop down boxes inside the form for question 7,8,9 and 10.
The following form is used to add information into the stationary equipment list. The required fields are marked with *.

1) Enter the name of the manufacturer:

2) Enter the name of the stationary equipment:

3) Enter the rated power of the stationary equipment:

4) Enter a Description of the stationary equipment:

5) Enter the datasheet link of the stationary equipment e.g. http://www.example.com, preferably pdf files

6) Enter the calibration certificate link of the stationary equipment e.g. http://www.example.com, preferably pdf files

7) Make a choice of the following general categories:
   - High Voltage/High Power
   - Climate Simulation
   - DeMoTec

8) Make a choice of the following stationary equipment categories:

9) Make a choice of the following facilities (if the facility box is empty please go back and add the facility first):

10) Make a choice of an existing contact or fill in a new contact person

   Choose one of the contact persons from this list

   Or make a new contact person

   Enter the name of the new contact person

   Enter the email of the new contact person

Figure 5-2 Stationary equipment form
The following form is used to add information into the mobile equipment list.
The required fields are marked with *.

1) Enter the name of the manufacturer:

2) Enter the type or version of the mobile equipment:

3) Enter a description of the mobile equipment:

4) Enter the datasheet link of the mobile equipment e.g. http://www.example.com, preferably pdf files

5) Enter the calibration certificate link of the mobile equipment e.g. http://www.example.com, preferably pdf files

6) Make a choice of the following general categories:

7) Make a choice of the following mobile equipment category:

8) Make a choice of the following facilities, e.g. where the equipment is located (if the facility is empty please make a new one):

9) Make a choice of an existing contact or fill in a new contact person

Choose one of the contact persons from this list:

Or make a new contact person:

Enter the name of the new contact person:

Enter the email of the new contact person:

---

Figure 5-3 Mobile equipment form
The following form is used to add information into the facilities list. The required fields are marked with *.

1) Enter the name of the facility:

2) Enter the location(city) of the facility:

3) Enter the dimension(m2) of the facility:

4) Enter the supply of the facility (kVA) e.g. mains, nominal power of transformers:

5) Enter a description of the facility:

6) Enter the link of the website if available e.g. http://www.example.com

7) Make a choice of the following general categories:

High Voltage/High Power

8) Make a choice of the following facilities category:

High Voltage/High Power

10) Make a choice of an existing Contact or fill in a new Contact person

Choose one of the Contact persons from this list

Or make a new Contact person

Enter the name of the new Contact person

Enter the email of the new contact person

Figure 5-4 Facilities form
In the process to generate the form it is important to know the company name, this can be retrieved from the username for normal users or is retrieved from form data for super users. Further it is very important when programming in PHP to do everything in case sensitive way because in Windows is case insensitive but Linux is case sensitive. After the user filled in the form, the form will be sent to the insertdatabase.php.

Another option to add multiple equipment or facilities is to make use of the given Excel file and save the file as CSV file. In PHP the CSV file lines can be saved as an array and used to insert into the DERlab database. Before the CSV information can be saved into the database the CSV format and database format need to be matched. When the user is using a CSV file it is possible that the information is duplicated in the CSV file and database. That's why there is a duplicate check for database and CSV file

5.3.2. Deleting equipment or facilities

As mentioned on the start of this section, there is also a possibility to delete equipment or facilities inside the DERlab database. The interface for deleting equipment or facilities is shown in Figure 5-5.
Figure 5-5 Interface “delete equipment or facilities”

This delete form shows only the equipment or facility for the specific company. And using the dellID number (interconnected with the checkboxes), the information will be deleted.

5.3.3. Conclusions and recommendations

At the end, the web interface for equipment and facilities is easy to use and functions according to the specifications. But there were some improvements possible, especially for the database and the user login. Because the user login is used to add and delete information from the database, it is very inconvenient that the names are not always the same as inside the database. Because of these small name differences, when searching, adding or deleting from the database, these differences need to be nullified. The implementation of the database would be better if the usernames are the same as the prefix names and
acronyms. In addition some implemented functions didn't work on the server because the server is still running on an old PHP server. These are some email, website check functions which are only available for PHP 5.0 versions. At the end the adding functionality and deleting functionality works correctly.

5.4. Interconnections Requirement units

The interconnection requirements units are explained in this section. First of all the add functionality is explained in section 5.4.1. The delete functionality will follow in section 5.4.2 and the search functionality in section 5.4.3. Finally some conclusions and recommendations are discussed in section 5.4.4.

5.4.1. Adding interconnection requirements

In this subsection, the adding interconnections requirement is described. There is only one possibility to add new interconnection requirements and that's via the manual way.

The following figure shows the interconnection requirements form.

![Interconnection requirements form](image)

Figure 5-6 Interconnection requirements form
5.4.2. Deleting interconnection requirements

This part describes the delete functionality inside the interconnection requirements. The delete functionality is similar to the delete functionality of equipment and facility. As shown in Figure 5-7, only the private interconnection requirements are visible. The delete selection will be created by checkboxes. After clicking delete the selected entries, the screen for confirmation will appear. By confirming the deletion the selected entries will be removed from the DERlab database.

5.4.3. Search interconnection requirements

As explained at the beginning, there is also a search function inside the interconnection requirements. This search function allows the user to search within categories or to search within some keywords. Figure 5-8 shows the form for searching inside the interconnection requirement.
To search inside the interconnection requirement database, please fill in the required information:

Please select the partner: [All partner]

Please select a section: [All Section]

Make a choice of the following general categories: [All Category]

Please select a document type: [All Type]

Enter some keywords to search inside interconnection requirement: 

Search in database

Figure 5-8 Form search interconnection requirements
5.4.4. **Conclusions and recommendations**

In the end the interconnection requirements part functions correctly. An extra option can be added such as an add function via Excel (CSV file). Another extension would be to save specific information about a requirement into a PDF file. But the interconnection requirement part has a user friendly web interface and functions according to the specification.

5.5. **Conclusions and recommendations**

Finally, to conclude the website interface, it is possible to add, delete and search inside the DERlab database using a user friendly website interface. But the DERlab database is implemented with inconsistent names compared with the login information. Secondly, please do not use capitals when creating the table names in the DERlab database. Because the programmer has to carefully program all the capitals in the PHP code which can cause mistakes in the code. There are still possibilities to improve the web interface design. One of these improvements is to add a function which makes it possible to add interconnection requirements using CSV files. Another possible improvement is to add the possibility to save interconnection requirements information into a PDF file.
6. Conclusions and recommendations

The implementation of the Labview control applications have been successful finished. It comprises the following developments:

- Controlling the 15 kVA Genset towards active and reactive power set values. These set values are retrieved from the central SQL database or via manual filled in set values.
- Controlling the 100 kVA Multi-PV towards active and reactive power set values. These set values are retrieved from the central SQL database or via manual filled in set values.
- Running the control systems in an DeMoTec PC using an application (.exe)
- Providing a user-friendly graphical user interface with the corresponding active power measurement chart and reactive power measurement chart.
- Build a mobile A2000 box to measure parameter values from a certain device.
- Implementation of a user-friendly website to add, delete and search within the equipment, facilities and services from the DERlab database.

A recommendation is to test the 100 kVA Genset for cases above the limited apparent power of 57 kVA. The next step for the 15 kVA Genset can be an extension towards a control for grid voltage and grid frequency.
Appendix A  

Configuration A2000 power meter for 15 kVA Genset measurements

In this subsection the measurement with “the A2000 Multifunctional Power Meter for 3-Phase Systems” (A2000) is explained. For more information about the A2000, refer to [6]

This mobile equipment is build to measure the active power, reactive power, voltages and other electrical parameters of the 15 kVA Genset connected to the grid. The A2000 front panel is shown in Figure A-1.

Figure A-1 Front Panel A2000 measurement box

And in Figure A-2 is the internal environment presented whereby the current is measured using a three phase current shunt labeled as 1. The A2000 labeled as number 3 receives the current measurement values from the three phase current shunt outputs. The voltages are retrieved by connecting the three phase voltages via a fuse labeled with number 2 to the A2000.
After connecting the cables for voltage and current measurement to the A2000, there are configurations which need to be set. Before the A2000 can retrieve all the measurement values, the user needs to define the inputs which are connected to the A2000, the configuration steps are shown in Figure A-3. By changing the reactive power setting, the reactive power value can be shown with a positive sign and negative sign otherwise the A2000 shows only positive values. The reactive power negative value setting is shown in Figure A-4.
4.3 Measurement Inputs, Configuring the Synchronizing Input

Menu Display

ProG

Input 5

Analog Output Menu...

3-Wire Connection

Input transformer primary voltage: 400 V phase-to-phase
Input transformer secondary voltage: 100 V phase-to-phase
Input transformer primary current: 60.0 kA

- Value Settings

4L: 100 V ... 700 V in 1 V steps
5L: 500 V ... 750 kV in 100 V steps
5L-1: 5 A ... 150 kA in 5 A steps for I < 5 kA
50 A steps for I > 5 kA
500 A steps for I > 50 kA

Figure A-3 Configuration input parameters A2000
After these configurations are done, it is possible to display the measurements parameters (active power, reactive power, power factor and etc.) on the A2000 display.

There are two possibilities to import those measurements values into the 15 kVA Genset Labview application. First of all it is possible to do it via Lon interface, an Ethernet network interface. The A2000 will store via the Lon interface the measurement values in an OPC server. Via an OPC client, the user can retrieve the measurement values and use it to control the active and reactive power.

Another way is to retrieve the measurement values directly from the A2000 via the analog outputs. This is implemented in the 15kVA Genset control application. By connecting wires to the four A2000 analog outputs it is possible to retrieve four measurement values. The analog outputs can be configured using the A2000 Front Panel; the analog outputs configuration is shown in Figure A-5.
4.4 Configuring the Analog Outputs (not with Proflbus-DP)

Menu
Display

Time

Pro
Analog out

Pulse Menu...

Analog Outputs 1 and 2

Source

Analog Output 1
Voltage or current output, depends on DIP switch setting

Source for analog output 1

Analog Output 1
Output Quantities

Collective Values

Neutral conductor current

0-20
0-10
0-20

Neutral conductor current

Output Quantities

Display

Volt

mA

DIP A1: U=on I=on

I=off U=off

The appropriate DIP switches must be set correctly!

The same windows and values apply to analog output 2. Analog outputs 3 and 4 may also be optionally included.

The source is relative to the latest completed interval value (P1, Q1, S1) for internal power

Figure A-5 Configuration analog outputs

The 15 kVA Genset uses three analog outputs to measure the active power, reactive power, and the RMS value of the three phase terminal voltage. Finally to activate the analog outputs, dip switches are set to the option voltage output, the dip switches are located on the backside of the A2000. The dip switch setting is also explained in Figure A-5.
Appendix B   Building a running application for the 15 kVA Genset and 100 kVA Multi-PV

The 15 kVA Genset and 100 kVA Multi-PV Labview applications are converted into a running application (.exe) for each control PC. The conversion into an application (.exe) is explained in this appendix.

15 kVA Genset application settings
Firstly, the conversion from the 15kVA Genset Labview application to a running application is explained. Before building the running application, a project for the 15 kVA Genset has to be made consisting of all involved functions and global variables. The 15 kVA Genset project is shown in Figure B-1, whereby the main program is named as Final15kVA.vi. The other vi files supports the main program.

![Figure B-1 Project 15kVA Genset](image)

After finishing adding all vi files then a run-time engine shown in Figure B-2 has to be created. This Run-time engine contains libraries and other files necessary to execute the LabVIEW application (.exe)
After pressing *build* the installer is created as shown in Figure B-1 as “My Installer”. Creating the application (.exe) is presented in Figure B-3. Firstly the “source files” for the 15 kVA Genset project is defined; the startup Vis and all the support files need to be added as shown in Figure B-3.
After all settings are configured the build can be pressed and the application is created as shown in Figure B-1 as “My Application”. Then the created application files and the installer are transferred to the destination computer. At the destination computer the file “My Installer” has to be installed before the application (.exe) can run. After installation is finished the application (.exe) can run and has the same graphical user interface as in the normal Labview application.

Besides the Labview application it is important to install all the required uDAQ drivers. There are two different drivers which need to be installed, the uDAQ lite drivers and the uDAQ lite Labview drivers. Please refer to [7] which drivers need to be installed.

100 kVA Multi-PV application settings
Creating the “100 kVA Multi-PV” application is almost the same as the 15 kVA Genset application. The only two differences are the 100 kVA Multi-PV requires VISA (Serial software driver) to be installed and doesn’t need uDAQ drivers. Like in the 15 kVA Genset case a project is created with a main program (MultiPVcontrolunit.vi) and the support files (functions and globals) as shown in Figure B-4.
Creating “My Installer” needs besides the “Run-time Engine” an extra installer item. The VISA Runtime is the extra item as shown in Figure B-5. This item is added into “My Installer” because the 100 kVA Multi-PV requires Labview drivers for serial communication.
At the end the application (.exe) is created as shown in Figure B-6.
After transferring the created files into the destination computer, firstly “My Installer” needs to be installed and then the application (.exe) can run. This works similar as in the case of the 15kVA Genset.

Both applications make use of the central SQL database and this requires a UDL file to be created. This UDL file is a link to the central SQL database.

The first step is to install an ODBC driver in the destination computer. Currently ODBC driver version 3.51 is installed in the DeMoTec laboratory computers.

Secondly, creating an ODBC data source via the ODBC Data Source Administrator (Windows XP: Control panel -> Administrative Tools -> ODBC Data Source Administrator) as shown in Figure B-7.
The third step is to add a new data source by clicking on the “Add” button. Then in Figure B-8, the Data Source can be created by pressing the finish button after choosing the option “MySQL ODBC 3.51 Driver”.

Then a configuration screen for the new Data Source appears as shown in Figure B-9. In this configuration, only the Data Source Name, description and
the database has to be filled in. Note that central SQL database is labeled as “demotec_control”.

After the Data Source is created, the last step is to create the UDL file. By pressing the right mouse in the desktop screen and choosing New -> Microsoft Data Link an empty UDL file is created. Open and configure this UDL file by selecting the created Data Source as shown in Figure B-10. Finally, by pressing the “OK” button the UDL file is created.
Specify the following to connect to ODBC data:

1. Specify the source of data:
   - Use data source name
     - MySQL Het SQL Leerboek
   - Use connection string
     - Connection string:
2. Enter information to log on to the server
   - User name:
   - Password:
     - Blank password
     - Allow saving password
3. Enter the initial catalog to use:
   - Initial catalog:

Figure B-10 Configuring UDL file
Appendix C  List of variables

The following list shows the used global or local variables in the 15 kVA Genset and the 100 kVA Multi-PV.

100 kVA Multi-PV
Local variables:
- \( P_{set_{man}}, Q_{set_{man}} \) are the manual set values for active power or reactive power

Global variables:
- \( \text{GlobalPO.vi} \) consists of \( p_{set} \) (Set value SQL database active power) and \( q_{set} \) (Set value SQL database reactive power) as global values. Further it consists of start/stop machine and manual or automatic buttons as global values.
- \( \text{GlobalPOmeas.vi} \) consist of \( p_{meas} \) (active power measurement), \( q_{meas} \) (reactive power measurement) and \( v_{meas} \) (Three phase RMS terminal voltage) as global values. Also an indicator for on/off machine.
- \( \text{Globalstop.vi} \) consists of one \textit{stop button} used for the whole 100kVA multi PV inverter program.
- \( \text{Databasepath.vi} \) consists of the \textit{UDL path} select button.
- \( \text{Startglobal.vi} \) consists of the start program button
- \( \text{Visaresource.vi} \) consists of the \textit{serial port control} select button

15 kVA Genset
Local variables:
- \textit{Stop 15kVA} is used to stop the 15 kVA Genset Labview application
- \textit{Stop synchronization} stops the synchronization control cycle
- \( P_{set_{man}}, Q_{set_{man}}, P_{set_{sql}} \) and \( Q_{set_{sql}} \) defines the set values for active power and reactive power via man (manual) or sql (central SQL database)
- \textit{Speed} and \textit{Vmeasured} are the rotation speed (Hz) of the 15 kVA Genset and the three phase RMS terminal voltage.
- \( P_{measured} \) and \( Q_{measured} \) are the measured active power and reactive power values.
- \textit{Torque voltage} and \textit{Excitation current} are influencing the active power and reactive power.

Global variables:
- \( \text{Fmeasandstop.vi} \) consists of \textit{fmeas} (speed measurement value in Hz) and \textit{stop button}.
- \( \text{Databasepath.vi} \) consists of the \textit{UDL path} select button
Appendix D  Load cabinets

This section describes possible ways to integrate the load cabinets inside the FENIX project. Firstly the load cabinets are explained and followed by the control application. Then the solution to integrate this application into the FENIX project is given and at the end conclusions and recommendations are given.

Introduction

The physical appearance of the reactive power load and active power load is shown in Figure D-1 and Figure D-2. The loads are controlled using an Ethernet bus terminal, this terminal is currently driven by a Microsoft Visual C application. This application can control all the resistances, inductances and capacitance from both loads via an IP address. The control application is based on digital input bit pattern and is shown in Figure D-3.

Figure D-1 Load reactive power
Figure D-2 Load active power
In the current situation, the control can only run via manually setting the input and it is not known which value belongs to which resistance ($R$), inductance ($L$) or capacitance ($C$). Integrating this application into the FENIX project, it is important to know the conversion between the “Actual Value” as shown in Figure D-3 and the parameter values $R$, $L$ and $C$ which are linked with the active and reactive power.

Thus measurements need to be made to determine the conversion table. This can be measured by using a voltage meter at the three phase power supply. The results from the measurement are:

**Load Active Power**

The active power values options 100 VA, 200 VA, 500 VA, 1000 VA and 2000 VA are available, there are six 200 VA resistances available and the other resistances are three times available. This is also shown in Figure D-1 Error! Reference source not found. whereby the input connections are divided into 3 equal rows. The conversion between active power and input bits is shown in Table 2.

**Load Reactive power**

The load reactive power consists of 6 rows and like the load active power it has values like 100 VAr/Vac, 200 VAr/Vac, 500 VAr/Vac, 1000 VAr/Vac, 2000 VAr/Vac. There are twelve 200 VAr/Vac inductances or capacitances available and the other inductances and capacitances are available for six times. The conversion between reactive power and input bit is shown in Table 3.
<table>
<thead>
<tr>
<th></th>
<th>Row1 decimal</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000VA</td>
<td>32</td>
<td>6th bit 1</td>
</tr>
<tr>
<td>1000VA</td>
<td>16</td>
<td>5th bit 1</td>
</tr>
<tr>
<td>500 VA</td>
<td>8</td>
<td>4th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>4</td>
<td>3th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>2</td>
<td>2th bit 1</td>
</tr>
<tr>
<td>100 VA</td>
<td>1</td>
<td>1th bit 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Row2 decimal</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2000VA</td>
<td>2048</td>
<td>12th bit 1</td>
</tr>
<tr>
<td>1000VA</td>
<td>1024</td>
<td>11th bit 1</td>
</tr>
<tr>
<td>500 VA</td>
<td>512</td>
<td>10th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>256</td>
<td>9th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>128</td>
<td>8th bit 1</td>
</tr>
<tr>
<td>100 VA</td>
<td>64</td>
<td>7th bit 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Row3 decimal</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2000VA</td>
<td>2</td>
<td>2th bit 1 output 2.0</td>
</tr>
<tr>
<td>1000VA</td>
<td>1</td>
<td>1th bit 1 output 2.0</td>
</tr>
<tr>
<td>500 VA</td>
<td>32768</td>
<td>16th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>16384</td>
<td>15th bit 1</td>
</tr>
<tr>
<td>200 VA</td>
<td>8192</td>
<td>14th bit 1</td>
</tr>
<tr>
<td>100 VA</td>
<td>4096</td>
<td>13th bit 1</td>
</tr>
<tr>
<td>L2 connect met L3 &amp; 4</td>
<td>3th bit 1 output 2.0</td>
<td></td>
</tr>
<tr>
<td>L1 connect met L2 &amp; 8</td>
<td>4th bit 1 output 2.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 2 Bit configuration active power values
<table>
<thead>
<tr>
<th>Left part reactive power load</th>
<th>4th from left reactive power load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>Decimal</td>
</tr>
<tr>
<td>2000VAr</td>
<td>32</td>
</tr>
<tr>
<td>1000VAr</td>
<td>16</td>
</tr>
<tr>
<td>500VAr</td>
<td>8</td>
</tr>
<tr>
<td>200VAr</td>
<td>4</td>
</tr>
<tr>
<td>200VAr</td>
<td>2</td>
</tr>
<tr>
<td>100VAr</td>
<td>1</td>
</tr>
<tr>
<td>2nd from left reactive power load</td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td>Decimal</td>
</tr>
<tr>
<td>2000VAr</td>
<td>2048</td>
</tr>
<tr>
<td>1000VAr</td>
<td>1024</td>
</tr>
<tr>
<td>500VAr</td>
<td>512</td>
</tr>
<tr>
<td>200VAr</td>
<td>256</td>
</tr>
<tr>
<td>200VAr</td>
<td>128</td>
</tr>
<tr>
<td>100VAr</td>
<td>64</td>
</tr>
<tr>
<td>5th from left reactive power load</td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td>Decimal</td>
</tr>
<tr>
<td>3th from left reactive power load</td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td>Decimal</td>
</tr>
<tr>
<td>2000VAr</td>
<td>2</td>
</tr>
<tr>
<td>1000VAr</td>
<td>1</td>
</tr>
<tr>
<td>500VAr</td>
<td>32768</td>
</tr>
<tr>
<td>200VAr</td>
<td>16384</td>
</tr>
<tr>
<td>200VAr</td>
<td>8192</td>
</tr>
<tr>
<td>100VAr</td>
<td>4096</td>
</tr>
<tr>
<td>100VAr</td>
<td>1</td>
</tr>
<tr>
<td>1 to 2</td>
<td>4</td>
</tr>
<tr>
<td>2 to 3</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 3 Bit configuration reactive power values
**Control approach load cabinets**

The final purpose is to control the load cabinets using active power set values and reactive power set values from the central SQL database or manual filled in set values. Those values needs to be rounded to the closest possible load cabinet configuration and then converted into the correct bit pattern to drive the load cabinets. So basically there are some tasks which need to be programmed, before these two loads can be integrated into the FENIX project.

- Retrieving from SQL database the set values or retrieving manual filled in set values
- Search for closest possible load value
- Conversion load value to bit pattern
- Load the bit pattern in the Ethernet terminal
- Make a easier graphical user interface

This programming pattern should be done for both load cabinets.

**Conclusions and recommendations**

This section gives an introduction and the possibility to control the load cabinets. There are two possible ways to implement the load cabinets inside the FENIX project. One way is to program everything in Microsoft Visual C whereby a sample source code available is. Another solution is to install an IBS OPC server, which means controlling the load cabinets using a Labview application which can control OPC items.
Appendix E 15 kVA Genset hardware setup

This section describes the 15 kVA Genset hardware setup. An overview of the “15 kVA Genset” and the “15 kVA Genset SMA control cabinet” are shown in Figure E-1.

Figure E-1 15kVA Genset (front) and 15kVA Genset SMA control cabinet (backside)
Inside the "15kVA Genset SMA Control cabinet" is located the 15kVA Genset power supply as shown in Figure E-2.

In the middle of Figure E-2 is a green and red button which activates and stops the 15kVA Genset power supply. When the 15kVA Genset is not properly closed, a display error is shown.

Figure E-2 15kVA Genset power supply

Figure E-3 shows the control display of the "15 kVA Genset control cabinet" and is located in front of the "15kVA Genset SMA Control cabinet". The status of the 15 kVA Genset has to be run before the 15 kVA Genset can be activated. When the 15kVA Genset is not properly closed, an error is shown on the display.
After properly configuring the "control display" and "power display" the 15kVA Genset can be activated using an "AEG switch" inside the "15kVA Genset SMA Control cabinet" and this "AEG" switch is shown in Figure E-4 as the most left one. The "AEG switch" on the right side is indifferent for the control.
Besides these 15 kVA Genset startup instructions, new cables are attached to the current transformers B13018 F1 to control directly the synchronous generator excitation current. The current transformers “B13018 F1” are shown in Figure E-5. “Hellas 200-40-5” or the synchronous generator power supply which controls manually the excitation current and excitation voltage is also shown in Figure E-5.
Figure E-5 Analog inputs/outputs synchronous generator
Figure E-6 shows the “synchronous generator power supply” front panel. Firstly the two meters in Figure E-6 display the synchronous generator “excitation current value” and “excitation voltage value”. The LED’s CV and CC are indicators for constant current and constant voltage. These indicators indicate also the status of the power supply; voltage source (CV) or current source (CC). The toggle switch on the left side enables the “manual control” (INT) or “externally control” (EXT). When the “manual control” is enabled, the user can directly control the excitation current via the knobs “Voltage” and “Current”. These knobs determine the status of the power supply. The highest absolute value for current or voltage determines if the power supply works as voltage source or current source. The inputs “−S(Gnd) and +S” measures the excitation current and the “−V(Gnd) and +V” measures the excitation voltage.

In Figure E-7 the connections between the excitation voltage and excitation current input port with the outside environment is shown. The “ground” is the green/yellow cable (163). And the red cable (162) is to control the “excitation voltage” and the most left brown cable (165) is to control the “excitation current”. In DeMoTec these cables are labeled as “excitation current”, “excitation voltage” and “ground”.

Figure E-6 Synchronous generator power supply
Apart from the excitation current/voltage cables, there are also cables connected to the 15 kVA Genset to set the torque voltage. This cable is labeled in DeMoTec as "Torque voltage".

Besides the input cables there are also four output cables to measure specific parameters, one of the output cable is the induction motor speed (Hz) measurement cable. This cable is labeled in DeMoTec as "Motor speed".

The three other output cables, "Active power measurement", "Reactive power measurement" and the "Three phase RMS terminal voltage measurement" are coming from the A2000 measurement device. These cables are labeled in DeMoTec as "Pmeas", "Qmeas" and "Vmeas".

These inputs and outputs are integrated into one data acquisition device (uDaq Lite) which is shown in Figure E-8 Error! Reference source not found.
The labels for the inputs and outputs to the 15kVA Genset uDAQ are shown in Figure E-8, the definition for the input and output is summarized below.

- Ch0 digital output: Excitation voltage (V)
- Ch0 analog output: Torque (V)
- Ch1 analog output: Excitation current (V)
- Ch1 analog input: Motor speed (Hz)
- Ch2 analog input: Pmeas (Watt)
- Ch3 analog input: Qmeas (VAR)
- Ch5 analog input: Vmeas (V)

This device sends and receives information to or from the implemented Labview application which is described in section 4.2.
Appendix F  List of Abbreviations

A: Ampere
C: Capacitance
DERlab: Distributed energy resources laboratory
Fmeas: Motor frequency
FENIX: Flexible Electricity Networks to Integrate the eXpected energy evolution
Hz: Hertz
L: Inductance
P: Active Power
Pfmeas: Power factor measurement value
Pmeas: Active Power measurement value
Pset: Active Power set value
Q: Reactive Power
Qmeas: Reactive Power measurement value
Qset: Reactive Power set value
R: Resistance
UDL: Microsoft data link
V: Volt
VAR: Volt-Ampere
Vmeas: Three phase RMS voltage measurement value
W: Watt
15 kVA Genset: 15 kVA generator set
100 kVA Multi-PV: 100 kVA multifunctional photovoltaic inverter
Appendix G List of tables

Table 1 100 kVA Multi-PV protocol layout ................................................................. 13
Table 2 Bit configuration active power values ............................................................. 71
Table 3 Bit configuration reactive power values ......................................................... 72
Appendix H List of figures

Figure 2-1 Overview developed active and reactive power control systems........6
Figure 3-1 Overview 100 kVA Multi-PV control...........................................8
Figure 3-2 100 kVA Multi-PV connections ..................................................8
Figure 3-3 Front panel 100 kVA Multi-PV ....................................................10
Figure 3-4 100 kVA Multi-PV Labview Control implementation......................11
Figure 3-5 Write Multi-PV. ............................................................................12
Figure 3-6 Read Multi-PV ..............................................................................13
Figure 3-7 Format 100 kVA Multi-PV protocol ............................................14
Figure 3-8 Make P or Q 7 bytes .....................................................................15
Figure 3-9 Serial write configuration .............................................................16
Figure 3-10 Serial read configuration .............................................................16
Figure 3-11 Split received string .....................................................................17
Figure 3-12 Active power measurement 100 kVA Multi-PV ..........................18
Figure 3-13 Zoom in active power measurement 100 kVA Multi-PV .............19
Figure 3-14 Reactive power measurement 100 kVA Multi-PV ......................20
Figure 3-15 Zoom in reactive power measurement 100 kVA Multi-PV ...........21
Figure 4-1 Overview 15 kVA Genset..............................................................23
Figure 4-2 15 kVA Genset (left: induction generator; centre: induction motor; right: synchronous generator) [1]..........................................................24
Figure 4-3 Per-phase representation driving motor [4].....................................25
Figure 4-4 Phasor diagram synchronous generator [5].....................................25
Figure 4-5 Graphical user interface 15 kVA Genset control system ...............27
Figure 4-6 Synchronizer device 15 kVA Genset ...........................................27
Figure 4-7 Synchronization panel 15 kVA Genset ..........................................28
Figure 4-8 Popup 15 kVA Genset .................................................................28
Figure 4-9 Synchronization 15 kVA Genset ...................................................29
Figure 4-10 SQL or Manual getting set values ..............................................31
Figure 4-11 PI control design for active power P ..........................................32
Figure 4-12 PI control active and reactive power .........................................33
Figure 4-13 Updating the measurement values in SQL database ....................34
Figure 4-14 Creating chart to display measurement values ............................35
Figure 4-15 Receiving A2000 measurement values .......................................36
Figure 4-16 Conversion to Reactive power values .......................................36
Figure 4-17 Conversion to Active power values ..........................................36
Figure 4-18 Conversion to Three phase terminal Voltage values ....................37
Figure 4-19 15 kVA active power measurement .........................................38
Figure 4-20 15 kVA reactive power measurement ........................................38
Figure 5-1 Global website structure .............................................................41
Figure 5-2 Stationary equipment form ............................................................43
Figure 5-3 Mobile equipment form .................................................................44
Figure 5-4 Facilities form ..............................................................................45
Figure 5-5 Interface “delete equipment or facilities” ......................................47
Figure 5-6 Interconnection requirements form .............................................48
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-7</td>
<td>Form delete interconnection requirements</td>
</tr>
<tr>
<td>5-8</td>
<td>Form search interconnection requirements</td>
</tr>
<tr>
<td>A-1</td>
<td>Front Panel A2000 measurement box</td>
</tr>
<tr>
<td>A-3</td>
<td>Configuration input parameters A2000</td>
</tr>
<tr>
<td>A-4</td>
<td>Reactive power negative value setting</td>
</tr>
<tr>
<td>A-5</td>
<td>Configuration analog outputs</td>
</tr>
<tr>
<td>B-1</td>
<td>Project 15kVA Genset</td>
</tr>
<tr>
<td>B-2</td>
<td>Run-Time Engine installer</td>
</tr>
<tr>
<td>B-3</td>
<td>Creating 15kVA application (.exe)</td>
</tr>
<tr>
<td>B-4</td>
<td>Project 100 kVA Multi-PV inverter</td>
</tr>
<tr>
<td>B-5</td>
<td>Run-time Engine installer</td>
</tr>
<tr>
<td>B-6</td>
<td>Creating 100kVA application (.exe)</td>
</tr>
<tr>
<td>B-7</td>
<td>ODBC Data Source Administrator</td>
</tr>
<tr>
<td>B-8</td>
<td>Create New Data Source menu</td>
</tr>
<tr>
<td>B-9</td>
<td>Configuration central SQL database Data Source</td>
</tr>
<tr>
<td>B-10</td>
<td>Configuring UDL file</td>
</tr>
<tr>
<td>D-1</td>
<td>Load reactive power</td>
</tr>
<tr>
<td>D-2</td>
<td>Load active power</td>
</tr>
<tr>
<td>D-3</td>
<td>Application to set and read input</td>
</tr>
<tr>
<td>E-1</td>
<td>15kVA Genset (front) and 15kVA Genset SMA control cabinet (backside)</td>
</tr>
<tr>
<td>E-2</td>
<td>15kVA Genset power supply</td>
</tr>
<tr>
<td>E-3</td>
<td>15kVA Genset control display</td>
</tr>
<tr>
<td>E-4</td>
<td>Switch 15kVA Genset</td>
</tr>
<tr>
<td>E-5</td>
<td>Analog inputs/outputs synchronous generator</td>
</tr>
<tr>
<td>E-6</td>
<td>Synchronous generator power supply</td>
</tr>
<tr>
<td>E-7</td>
<td>Input ports excitation current and excitation voltage</td>
</tr>
<tr>
<td>E-8</td>
<td>Configuration uDAQ</td>
</tr>
</tbody>
</table>
References


