Data Acquisition Architecture for HVDC Grids

Erik Björk
Viktor Åkerberg

Handledare:
Davood Babazadeh
Yiming Wu

AL125x Examensarbete i Energi och miljö, grundnivå
Stockholm 2016
Data Acquisition Architecture for HVDC Grids

Erik Björk and Viktor Åkerberg

Abstract—The climate crisis has caused many countries around the world to invest in large amounts of renewable energy. To be able to handle the intrinsic unreliability and geographic dependency of many renewable energy sources, HVDC technology is considered due to its low cost when transferring electricity across great distances. Traditional AC grids are controlled with 15-minute intervals at control centers, but HVDC grids require a faster control due to more power fluctuations within the grid. The aim of this project was to propose an architecture for a gateway in a control center for an HVDC grid. The gateway was programmed in C and C++ and the data was sent using UDP packets. Testing of the gateway was done using a real-time simulation of an HVDC grid. The data was sent with intervals smaller than a second which satisfied the speed requirements for this project. A gateway like the one developed in this project can be implemented at control centers to display and process data and to improve the overall reliability of an HVDC grid.

I. INTRODUCTION

THE European Union has set a goal of having 27% renewable energy consumption in the year 2030 [1]. In 2014 the European union had reached 15.3% [2], this indicates that there will be an increased expansion of renewable energy in the coming years. An increasing amount of renewable energy connected to the grid requires a more flexible and interconnected grid. Most renewable energy sources are dependent on the weather to produce electricity. This causes more power fluctuations within the grid, and power shortage and abundance will be more common in the future [3]. To cope with this issue the electric infrastructure needs to be updated, both the grid and the control centers [4].

An electric grid that extends across different weather systems ensures a stable flow of power from renewables, but this requires a grid that can transport electricity over greater distances than they are doing today. One solution to this is the so-called “supergrid”. The supergrid enables transport of electricity over greater distances [5]. The supergrid is a grid which uses High Voltage Direct Current (HVDC) technology.

When transporting energy across long distances, usually more than 400 km, HVDC cables has less power losses than the commonly used Alternating Current (AC) cables [6]. HVDC lines also costs less than high voltage AC lines across long distances. At longer distances the reduced power losses in HVDC lines makes up for the cost of the AC-DC converters that connect HVDC to AC [7]. HVDC grids can use two different technologies, Line Commutated Converter (LCC) and Voltage Source Converter (VSC). VSC is the most promising technology according to [8]. Further on in the report HVDC lines and grids are assumed to use VSC technology.

The control center is constantly monitoring the power flows within the grid and regulates the generation to ensure a steady power supply for the customers [9]. To control the balance between load and generation in an AC grid one of the fundamental units to measure is frequency. The frequency indicates if the grid is in balance or imbalance. In the HVDC grid the corresponding unit to measure is the voltage because HVDC grids do not have any frequency. However the HVDC grid is more sensitive to changes between the generation and load which makes the voltage change more rapidly than the frequency [4]. Currently the AC grids are controlled at the substation-level with protective functions that can respond within milliseconds. But at the control station level the load and generation control are executed with 15-minute intervals. The main application of HVDC today is to transfer power from offshore wind farms to AC grids. This poses a problem as the power fluctuations in wind power can be quite high, the generation can sometimes vary with as much as 1.6% per minute [8]. The control center in an HVDC grid therefore requires more accurate measurements of voltage and a faster than 15-minute update speed compared to control centers in AC grids. The HVDC control center is placed between the slower AC grid control and the faster substation control as seen in Fig. 1.

The goal of this project was to develop an architecture for a gateway at an HVDC control center. The gateway should transfer obtained measurements from substations in an HVDC grid to a database where they are stored. The purpose of the model was to transfer data fast and reliably, this was tested using a real time platform. The required update speed was less than one minute according to the requirements set in the article [8]. The front end was created in the programming languages
C and C++ and the data was transferred using UDP packages. UDP packages are further described in section Background Theory.

II. Background Theory

A. Energy Management Systems

Energy Management System (EMS) controls and monitors the energy flows within the grids. The EMS is stationed within system control centers. The EMS controls the supply and demand of electricity and keeps it balanced continuously because the electrical grid can not store power. The EMS has to be able to control load, generation or both in order to ensure a stable grid. The EMS also has to be able to prevent the damage caused by unexpected events such as lightning strikes, short circuits, faulty equipment or other accidents. This is done by automatic relays that disconnect parts of the grid that are prone to damage. The computer systems that control and monitor the electrical grid are called Supervisory Control And Data Acquisition (SCADA) systems. These SCADA systems often consist of two levels, a control center-level and a substation-level. Information about the electrical grid, such as measurements and device states, are stored in substations. Substations contain measuring equipment and devices that collect and send the information. The control center communicates with substations via different units, one of them being Remote Terminal Units (RTUs). This is explained in Fig. 2. The substations are located at different locations within the grid [11]. Most of these substations follow the IEC 61850 standard, a standard for communication between substations in electrical grids [12].

The control center displays select measurements and states in such a way that the operator can understand them. Some measurements are stored in a database for future use. The control center also performs automatic generation control (AGC) that regulates the load and generation in the grid. The main purposes of this control is to keep the frequency for AC grids within the desired interval, keep the load and generation balanced and also doing so with minimum cost [11].

Today the RTUs are typically connected directly to a SCADA system via dedicated communication channels. This causes problems when other control centers want access to the substations real time data as these control centers are not connected to the dedicated communication channels. A solution to this is network based communication. This enables different users to have access to the substations simultaneously. Although it might be more convenient for the grid owners, network based communication causes problems with integrity and security as the data might be intercepted by unknown entities [9].

B. Network Communication

User Datagram Protocol (UDP) is a protocol for sending packets of data across the internet and is mostly used for sending messages that are required to arrive fast. When sending a UDP packet, an IP-address and port number needs to be specified in order for the packet to arrive at the correct location. An IP-address specifies which computer the packet should be sent to, if the IP-address is compared to a house address, the port number is the mailbox where the packet should be put. A computer can have many different port numbers ranging from one to 65535. Unlike other protocols, UDP does not require the sender to establish a connection with the receiver before messages are sent. Because of this connectionless communication, sending UDP packets is faster than most other protocols, but the packets are more prone to getting lost along the way. Another common protocol is Transmission Control Protocol (TCP), which is slower but more reliable. TCP packets are more likely to arrive as the protocol establishes a connection between sender and receiver before messages are sent [13].

III. Data Acquisition Architecture

A. Purpose and limitations

The purpose of the gateway developed in this project was to receive UDP packets containing timestamps and measurements from substations. The gateway should then unpack the packets and insert the timestamps and measurements into a database. The existing architecture that the gateway needs to collaborate with was the substations and the database. Seven computers of model Raspberry PI (RPI) simulated substations which timestamps received measurements from an electrical grid. The substation communication devices that were implemented on the RPIs mimic the industrial grade RTUs. The program on the RPIs can only send one measurement per packet.

B. Architecture

The gateway needs to send a request packet to the substations to receive their measurements. This request packet contains the IP-address and port number that the RPI should start sending information back to, in other words, a request to subscribe to a stream of information. After the request packet is sent, the program needs to get ready to receive the packets. This was done by opening the port specified in the request packet. The program then continuously received UDP packets from the RPIs. Only one port could be open at any time.

The gateway contained an array with all port numbers that have a corresponding measurement in the grid, and another array of equal size with all bus names. By matching the array index of the port number array and the bus name array, the
K3B. HVDC DATA ACQUISITION

Fig. 3. A flowchart visualizing the gateway

The gateway knew from which bus the measurement received in a port comes from. This was necessary to make sure that the gateway can insert a received measurement into the correct table and row in the database.

Once the message arrives, it contained a timestamp and a measurement. The timestamp sent from the RPIs was in UTC milliseconds, a measurement of how many milliseconds has passed since 00:00 on the first of January 1970. The gateway translated this 13-digit number into a readable date. Then the measurements were inserted into the database and the gateway moves onto a new port number and starts requesting more measurements. Fig. 3 explains the gateway in a flowchart.

C. Classification of measurements using port numbers

Due to limitations of gateway implementation in the RPIs, only one measurement can be sent in one packet. The amount of packets received quickly increased once measurements from all lines and buses in the grid were sent. The grid later mentioned in proof of concept (IV) contains over 30 buses and over 15 lines. If all relevant measurements such as current, voltage, power and reactive power were sent, close to one hundred unique packets will be sent. To make sure that the packets were sent systematically and with as few errors as possible, a system was developed where all measurements sent have a unique port number. All port numbers in the range of 50000 to 59999 are free to use, so all measurements sent were assigned a port number within this range. The classification of port numbers is explained in Fig. 4 with the example port number 52123. All port numbers start with a five, the second digit specified the RPI used. The third digit specified the bus within the RPI, starting from zero. The fourth digit specified what line number the measurement comes from and the fifth digit specified the physical quantity of the measurement.

What physical quantity was represented by which number is specified in Table I. The HVDC lines has one positive and one negative cable, therefore each line had two current measurements, one on the positive line and one on the negative.

<table>
<thead>
<tr>
<th>Physical Quantity</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage</td>
<td>1</td>
</tr>
<tr>
<td>Current</td>
<td>2</td>
</tr>
<tr>
<td>Power</td>
<td>3</td>
</tr>
<tr>
<td>Reactive Power</td>
<td>4</td>
</tr>
</tbody>
</table>

IV. PROOF OF CONCEPT

The gateway was comprised of several different C and C++ programs according to the architecture described above. The gateway was made using the integrated development environment Code::Blocks on Ubuntu and tested using a real time simulated HVDC grid. The model for the HVDC grid was developed by the International Council on Large Electrical Systems (CIGRE). The grid was comprised of 15 HVDC buses, six AC buses that simulates a connection to an AC grid and five AC buses that simulate offshore nodes [14]. Several AC-HVDC converters connected the AC and HVDC grid together. The CIGRE electrical grid was simulated in real time using the software OPAL-RT. Using a real time simulator means that the gateway can be tested in an environment resembling the real world. Several measuring devices that measured current, voltage and power were implemented into the simulation and sent to the Raspberry PIs. The data sent from the simulator goes through seven RPIs. The RPIs gathered measurements from several buses within the grid and sent the data to the gateway. Once the UDP packets were received and unpacked, the measurements were stored in a database written in SQL by project group K3A. An application program interface that inserts measurements from C++ into the database was required to connect the gateway and the database. Once the simulation, the RPIs and the gateway were online, the gateway started receiving measurements. The measurements were sent and unpacked one by one. By looking at the timestamps, the interval between each measurement sent was less than one second.

Once all measuring devices in the simulation were sending measurements, they can be used to visually represent the grid in a Graphical User Interface (GUI). This GUI can display the grid to a control center operator and change dynamically when measurements within the grid are updated. Power flows and voltages or currents exceeding certain limits can be displayed.
in this GUI. More information about the GUI and database can be found in project group K3A’s report EMS FOR HVDC. A snapshot of the GUI made by project group K3A is displayed in Fig. 5.

V. CONCLUSION & FURTHER RESEARCH

This article has presented an architecture for a network gateway suitable for HVDC. The architecture fulfilled the requirement of sending measurements reliably and had an update speed of less than one minute. The gateway has been tested in real-time and is ready to be implemented into a real HVDC grid. It suited the needs for the GUI and for other applications such as bad data detection.

The gateway was sufficient for the purpose specified in this project but there were still many areas in which it should be improved in order for it to be relied upon in a real-world control center.

1) Error elimination: Currently the gateway is a single program running on one computer that runs through all bus measurements one by one. This makes the gateway vulnerable to many types of errors, as it requires each measurement to be sent in order for the next one to be processed. Further programming can be made to make the gateway skip requesting measurements from RPIs that are not sending any. Running two or more programs in parallel may eliminate errors as one program can cease sending measurements without causing the entire control system to fail. Several independent computers running the same program may also reduce the risk of the control system failing due to errors caused by the operating system or hardware.

2) Secure transfer: The data in the UDP packets were sent without any encryption. Anyone can send measurements to the gateway as long as they know the correct port number and UDP packet structure. This is unfeasible in a real world application. An entity with malicious intent may send incorrect measurements to the gateway and cause the control system to turn off buses or disconnect cables. The information has to be encrypted before it is sent and a system which ensures that the measurements are sent from the real bus needs to the implemented in a real world application.

3) Reliability: The program made in this project was sending and receiving UDP packets that are faster but less reliable than TCP packets. A program similar to the one made in this project but using TCP packets running in parallel with a program using UDP packets might be the best way to design a gateway for reliability and speed. The UDP program gives the control system fast updates of the state of the grid while the TCP program can give the control system slower but more reliable updates.

REFERENCES