Master’s Thesis 2012

Candidate: Babar Khan

Title: Wireless sensor networking using AADI Sensors with WSN Coverage

Telemark University College
Faculty of Technology
Kjølnes 3914 Porsgrunn Norway
Lower Degree Programmes – M.Sc. Programmes – Ph.D. Programmes

Telemark University College
Faculty of Technology
M.Sc. Programme

MASTER'S THESIS, COURSE CODE FMH606

Student: **Babar Khan**

Thesis title: **Wireless sensor networking using AADI Sensors with WSN Coverage**

Signature: . . . .

Number of pages: 73

Keywords: Automatic Weather Station, AADI, Wireless sensor network, SQL server

Supervisor: Saba Mylvaganam sign.: . . . .

2nd Supervisor: Hans-Petter Halvorsen sign.: . . . .

Censor: sign.: . . . .

External partner: (NI, Norway) sign.: . . . .

Availability: Open

Archive approval (supervisor signature)**:** sign.: . . . . **Date :** . . . .

Abstract:

This report provides detailed procedure of setting up Automatic Weather Station 2700 and implementing wireless sensor network. First the sensors were installed on mast, connected with Datalogger and then Datalogger was programmed. The LabVIEW programming was done to read and display real time data from AADI and WSN systems. Sensors data was logged into SQL server. A wireless sensor network system was configured and integrated into weather station. Again a LabVIEW program was made to retrieve and manipulate data. In order to check validity of system the resultant data was compared with Norwegian Meteorological Institute's data and weather station installed at Telemark University College, Porsgrunn. Finally, a test sensor was used to analyze and check validity of AWS and WSN sensors.

Table of Contents

Table of figures

Overview of Tables

Preface

This thesis project is mandatory part of master study in Systems and Control Engineering at Telemark University College. The entire work was carried out in Process Hall and equipment was provided by professor Saba Mylvaganam and senior engineer Hans-Petter Halvorsen and at Telemark University College. The main focus was to set up and run weather station, configure and run wireless sensor network system, integration of both systems, data acquisition and presentation of data from integrated system. LabVIEW 2011 together with NI WSN module was used in thesis project.

I would like to thank Prof. Saba Mylvaganam for his support and allowing me complete this project. My special thanks go to senior engineer Hans-Petter Halvorsen for his great interest and support. Without his help it would not be possible to complete this project and achieve the goals. I improved my problem solving and project management skills during this project and credit goes to Hans-Petter Halvorsen. Special thanks go to Mr. Tom-Arne Danielsen from National Instruments for his valuable suggestions.

Babar Khan

May 31, 2012

Nomenclature

AADI	Aanderaa Data Instruments
AWS	Automatic weather station
LabVIEW	Laboratory Virtual Instrumentation Engineering Workbench
NI	National Instruments
TUC	Telemark University College
WSN	Wireless sensor network

Part I Introduction and Theory

1 Introduction

A sensor network is a network of microcontroller integrated smart devices, called nodes, which are spatially distributed and sensors. A sensor is a primary component of network essential for monitoring real world physical condition or variables such as temperature, humidity, presence (absence), sound, intensity, vibration, pressure and motion. The important design and implementation requirements of a typical sensor network are energy efficiency, memory capacity, computational speed and bandwidth. The smart device has a microcontroller, a radio transmitter, and an energy source. Sometimes a central computer is integrated into the network in order to manage the entire networked system. A sensor network essentially performs three basic functions: sensing, communicating and computation by using the three fundamental components: hardware, software and algorithms, respectively.

A wireless sensor network (WSN) consists of nodes, sensors and a gateway. The smart device or node has some degree of intelligence for signal processing and management of network data. (Mahalik, 2007)

This device gathers data and transmits it via wireless links to user through a gateway. The nodes can be stationary or moving. They can be aware of their locations or not. They can be homogeneous or not. Monitoring and communication are performed cooperatively by the nodes. The main features of WSNs are: scalability with respect to the number of nodes in the network, self-organization, self-healing, energy efficiency, a sufficient degree of connectivity among nodes, low complexity, low cost, and small size of nodes. (Buratti, Martalò, Verdone, & Ferrari, 2011)

Automatic weather station AWS 2700 is an application with sensors, data logger and equipment for observing weather conditions to provide information to study weather and weather forecasting. The measurements taken include wind speed, wind gust, air temperature and pressure.

This project report is about wireless sensor networks, automatic weather station and integration of both systems. It starts with theoretical description of all the components of AWS and WSN in chapter 2. Also applications of wired and wireless sensor networks are discussed in chapter 2. Chapter 3 describes the practical work which was done in this project. First data logger programming is described in detail and then development of LabVIEW programming is mentioned. Data logging and WSN implantation is mentioned in chapter 3. Chapter 4 includes result and analysis and conclusion is given chapter 5.

1.1 Problem description

The aim of this project is to set up and run the weather station, data acquisition and data presentation. It is also required to configure existing wireless sensor network system and integrate it into AADI system. The weather sensors must be mounted on mast and connected to data logger. Data logger should be programmed before installation. Software for real time display is provided from vendor but it is desired in this project to use LabVIEW from National Instruments. LabVIEW is a graphical programming tool which stands for Laboratory Virtual Instrumentation Engineering Workbench. In order to study and analyze the whole system, real time data must be displayed from AADI and wireless sensors. Gathered data must be displayed and saved in a computer for analysis and study weather and environment.

2 System description

The following chapter includes the theory and description on which weather station and wireless senor network is based on. It includes among others definition of weather station and description of each component of automatic weather station 2700 which was used in this project and wireless sensor network components.

2.1 Automatic weather station

A weather station is a facility with instruments and equipment, either on land or sea used to monitor the atmospheric conditions to forecast the weather and to study the weather and climate. The measurements taken by such weather stations are temperature, barometric pressure, wind speed, wind gust, wind direction, humidity and precipitation. Wind measurements are taken as free of obstructions as possible i.e there should not be any obstacle when taking wind measurements. While temperature and humidity sensors are avoided by direct solar radiations. (Wikipedia, 2012)

An automatic weather station (AWS) is an automated version of the traditional weather station used for remote measurements. An AWS typically consist of a weather-proof enclosure containing the data logger, rechargeable battery and measuring sensors with an attached solar panel or wind turbine and mounted upon a mast. The data logger is configured before mounting the station. The data logger contains a memory which stores the weather data. In the past automatic weather stations were often placed where electricity and communication lines were available. Nowadays, the solar panel, wind turbine and mobile phone technology have made it possible to have wireless stations that are not connected to the electrical grid or telecommunications network. (Wikipedia, 2012)

Weather stations are used to monitor the weather situations in various industrial, governmental, commercial and military applications. The purpose of weather stations is to provide real time weather information to users. The monitoring and prediction of weather is very important when it comes to commercial uses for example it helps airplanes to land safely. The automatic weather station which was used in this project is known as AWS 2700 whose vendor is ‘Aanderaa Data Instruments AS’ and is shown in Figure 2-1

The Automatic Weather Station 2700 consists of:

- Air pressure sensor 2810
- Air temperature sensor 3455
- Wind speed sensor 2740
- Datalogger 3634
- Cable 3204 with 1.5m length using serial communication RS-232C
- Cable 3321 with 10m length

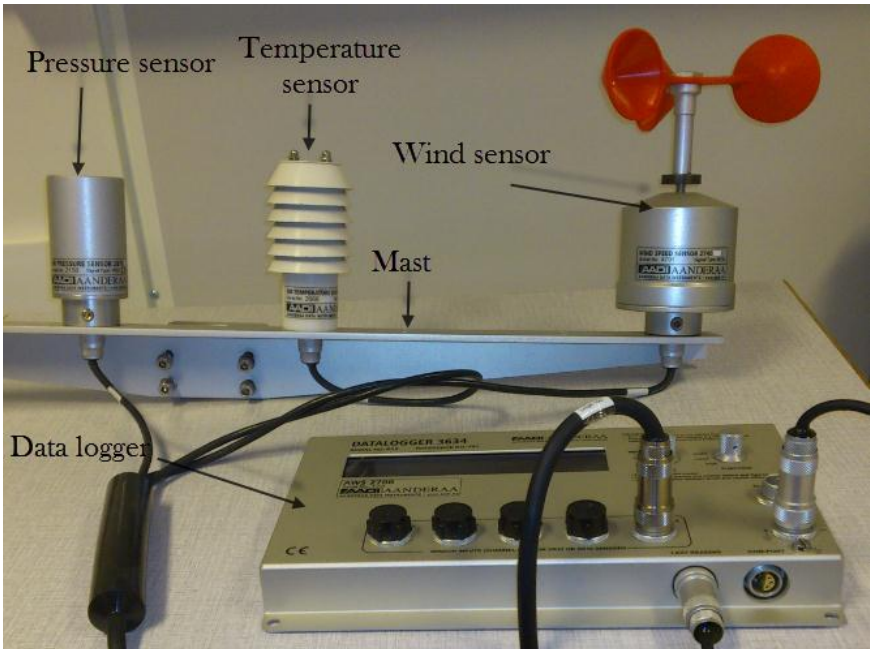


Figure 2-1 AWS 2700

2.1.1 Air pressure sensor

Air pressure sensor 2810 is a barometric pressure sensor which uses a small silicon chip as a sensing element. In the central area of this chip is a membrane that is exposed to atmospheric pressure at one side and to vacuum on other side. The membrane is equipped with 4 diffused resistors that form a Wheatstone bridge. The output signal is proportional to the atmospheric pressure. The chip thus acts as an absolute pressure-sensing device. Measuring range of sensor is 920 – 1080 hPa. hPa stands for hecto pascal where 1 hecto pascal is equal to 1 mbar. Its accuracy is +- 0.2hPa. (Air Pressure Sensor, 2006)

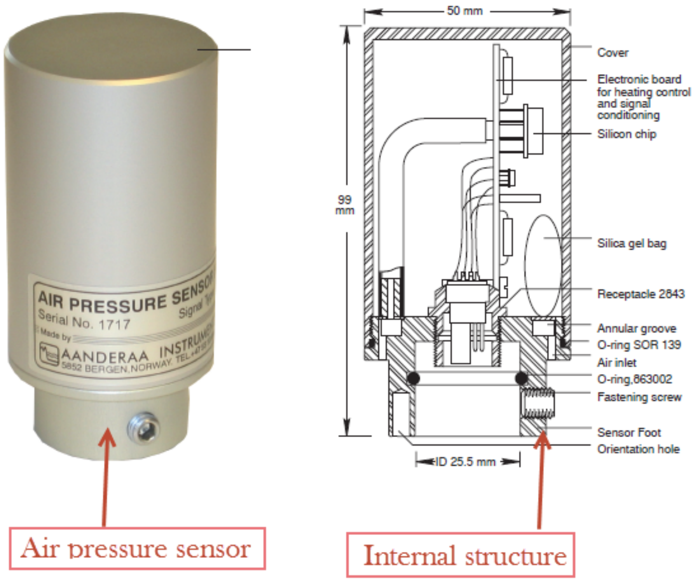


Figure 2-2 Air Pressure Sensor 2810 and its internal structure (Air Pressure Sensor, 2006)

This pressure sensor is calibrated when Datalogger is programmed. There are some coefficients which are prescribed in the specification sheet of the sensor. More details are given in Datalogger programming.

2.1.2 Air temperature sensor 3455

The air temperature sensor 3455 is designed for high accuracy air temperature measurements. The 80 mm long sensor is cylindrically shaped and built up on a 6-pin watertight receptacle. The sensor element is embedded in a small tube with cooling ribs. The wires and range resistors are molded in Durotong which forms the center part of the sensor. This construction ensures good thermal insulation between the sensor element, the receptacle and the connecting cable. The sensor is protected by radiation screens that make sure that sensor does not heat up in by direct sunshine. The sensor is based on ohmic half bridge principle (VR-22) and uses a 2000Ω film type platinum resistor as sensing element. The sensor and its internal structure are shown in Figure 2-3. (Air Temperature Sensor, 2006)

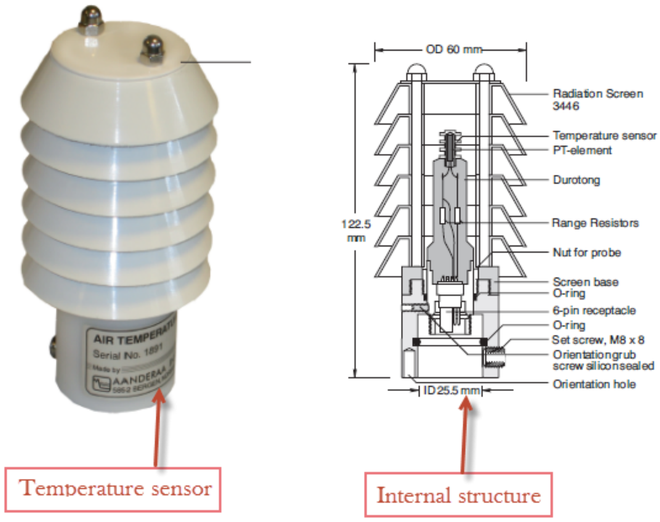


Figure 2-3 Air temperature sensor from AADI (Air Temperature Sensor, 2006)

Its data sheet is given in the table below:

Table 1 Datasheet for air temperature sensor (Air Temperature Sensor, 2006)

Measuring range	-43 to +48 °C
Sensing element	Pt 2000
Range resistors	R1: 4KΩ R2: Pt2000 + 2KΩ
Resolution	0.1% of range
Accuracy	+- 0.1% of range
Time constant	1 min 20 sec (at 5 m/s wind speed)
Electrical connection	Watertight plug 2828
Material and finish	Titinum and Durotong

Connecting cable 3321 is used to connect the sensor to the Datalogger. Circuit diagram for the sensing element is shown in Figure 2-4.

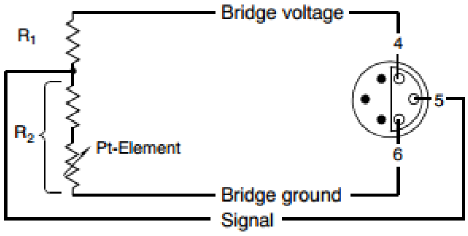


Figure 2-4 Circuit diagram (Air Temperature Sensor, 2006)

2.1.3 Wind speed sensor 2740

The wind speed sensor consists of three cup rotors on top of aluminum. The sensor is fitted on sensor arm of the Aanderaa Automatic Weather Station and cable 3321 is used to connect the sensor to the Datalogger. The rotor bearings consist of 2 stainless steel ball bearings protected by a surrounding skirt. A magnet is connected to the lower end of the skirt. The magnet's rotation is sensed by a magneto inductive switch located inside the housing. (Wind speed sensor, 2008)

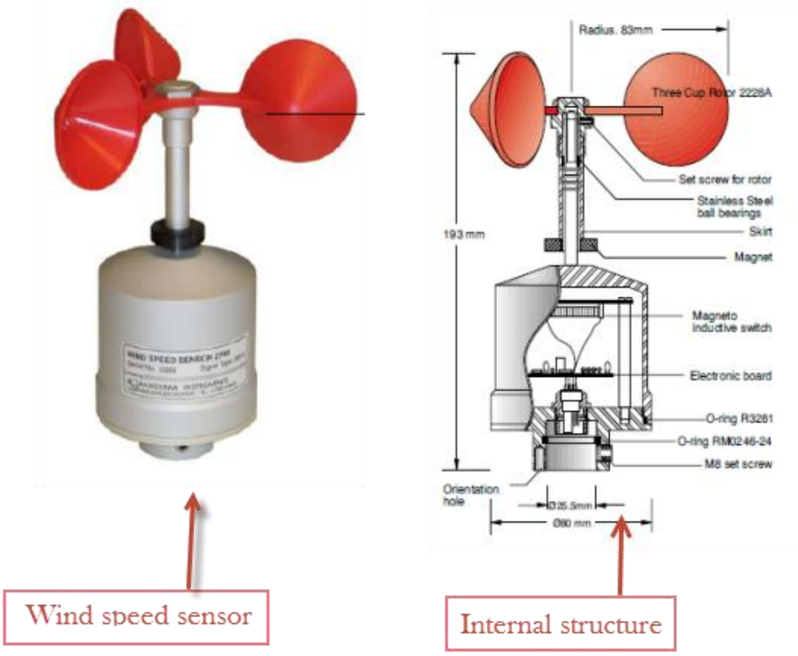


Figure 2-5 Wind speed sensor (Wind speed sensor, 2008)

The technical specifications of wind speed sensor are given below in Table 2.

Table 2 Specification for wind speed sensor is shown in the following table: (Wind speed sensor, 2008)

Range	Up to 79 m/s
Threshold speed	Less than 0.3 m/s
Distance constant	1.5 meters
Output signals	1. Average Wind, SR-10 2. Wind Gust, SR-10
Operation temperature	-40 to +65 deg. Celsius
Current consumption	250 µA
Operating voltage	7 to 14V DC
Electrical connection	Receptacle 2843 mating Watertight Plug 2828L
Gross weight	1.3 kg

2.1.4 Datalogger 3634

Datalogger is a device which stores data collected by sensing devices. Datalogger 3634 is a low power, light weight and watertight field operating device used to display data in engineering units. The Datalogger 3436 unit can scan up to 4 sensors. Data can be transmitted as raw data in 10 bit code by VHF or UHF-radio or as engineering units by modem. Data can be presented as a voice message by connecting a Voice Generator 3420. A PC can be used to read the real time data. A Display Program 3710 from Aaderaa or LabVIEW from National Instruments can be used to display real time data.

Datalogger contains a 4 line 40 character LCD, two control switches and a set of waterproof receptacles for electrical connections. It contains an internal battery and if power is lost, the Datalogger shall retain its programmed information due to this internal battery. A built-in quirts clock generates the trigger pulse for the unit. There are many selectable recording intervals in the Datalogger namely 2.5, 1, 2, 5, 10, 20, 30, 60, 120 and 180 minutes. The unit has a non-stop and a remote-start option. (Dataloggers 3634 and 3660, 2009)

2.2 Wireless sensor networking

The following chapter presents the theoretical foundation that wireless sensor networking is based on. It also includes the description of wired and wireless sensor networking and their applications.

2.2.1 Sensor network overview

A *sensor network* is an infrastructure comprised of sensing (measuring), computing, and communication elements that gives an administrator the ability to instrument, observe, and react to events and phenomena in a specified environment. The administrator typically is a civil, governmental, commercial, or industrial entity. The environment can be the physical world, a biological system, or an information technology (IT) framework. Typical applications of sensor network include, but are not limited to, data collection, monitoring, surveillance, and medical telemetry. In addition to sensing, one is often also interested in control and activation. (Sohraby, Minoli, & Znati, 2007)

There are four basic components in a sensor network: (1) an assembly of distributed or localized sensors; (2) an interconnecting network (usually, but not always, wireless-based); (3) data logging or information gathering; and (4) a set of computing resources for data analysis for example event trending, and data correlation. In this context, the sensing and computation nodes are considered part of the sensor network; in fact, some of the computing may be done in the network itself. Algorithmic methods for data management are very important since large amount of data is collected. (Sohraby, Minoli, & Znati, 2007)

A wireless sensor network (WSN) is a wireless network which consists of spatially distributed autonomous devices or nodes, sensors to monitor physical or environmental conditions, routers and a gateway. The autonomous devices, or nodes, combined with routers and a gateway form a typical WSN system. The distributed measurement nodes communicate wirelessly to a central gateway, which provides a connection to the wired world where data can be collected, processes, analyzed, and present the measurement data. To extend distance and reliability in a wireless sensor network, routers can be used to gain an additional communication link between end nodes and the gateway. (National Instruments, 2009)

The technology for sensing and control includes electric and magnetic field sensors; radio-wave frequency sensors; optical-, electro optic-, and infrared sensors; radars; lasers; location/navigation sensors; seismic and pressure-wave sensors; environmental parameter sensors (e.g., temperature, pressure, wind, humidity); and biochemical national security–oriented sensors. WSNs have unique characteristics, such as, but not limited to, power constraints and limited battery life for the WNs, redundant data acquisition, low duty cycle, and, many-to-one flows. (Sohraby, Minoli, & Znati, 2007)

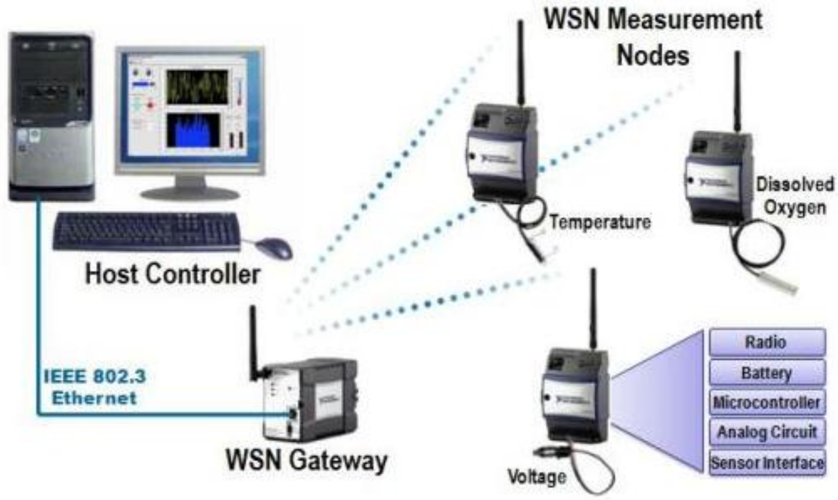


Figure 2-6 A WSN (National Instruments, 2009)

2.2.2 WSN benefits and Applications

“Some benefits of WSNs are highlighted as follows:

- *Anywhere and anytime*

The coverage of a traditional macrosensor node is narrowly limited to a certain physical area due to the constraints of cost and manual deployment. In contrast, WSNs may contain a great number of physically separated nodes that do not require human attention. Although the coverage of a single node is small, the densely distributed nodes can work simultaneously and collaboratively so that the coverage of the whole network is extended. Moreover, sensor nodes can be dropped in hazardous regions and can operate in all seasons; thus, their sensing task can be undertaken anytime.

- *Greater fault-tolerance*

This is achieved through the dense deployment of wireless sensor nodes. The correlated data from neighboring nodes in a given area makes WSNs more fault tolerant than single macrosensor systems. If the macrosensor node fails, the system will completely lose its functionality in the given area. On the contrary in a WSN, if a small portion of microsensor nodes fails, the WSN can continue to produce acceptable information because the extracted data are redundant enough. Furthermore, alternative communication routes can be used in case of route failure.

- *Improved accuracy*

Although a single macrosensor node generates more accurate measurement than one microsensor node does, the massively collected data by a large number of tiny nodes may actually reflect more of the real world. Furthermore, after processing by appropriate algorithms, the correlated and/or aggregated data enhance the common signal and reduce uncorrelated noise.

- *Lower cost*

WSNs are expected to be less expensive than their macrosensor system counterparts because of their reduced size and lower price, as well as the ease of their deployment." (Wang, Hassane, & Xu, 2005)

"Wireless sensors can be used where wireline systems cannot be deployed (e.g. a dangerous location or an area that might be contaminated with toxins or be subject to high temperatures). The rapid deployment, self-organization, and fault-tolerance characteristics of WSNs make them versatile for military command, control, communications, intelligence, surveillance, reconnaissance, and targeting systems. Many of these features also make them ideal for national security. Sensor networking is also seen in the context of pervasive computing.

Near-term commercial applications include, but are not limited to, industrial and building wireless sensor networks, appliance control [lighting, and heating, ventilation, and air conditioning (HVAC)], automotive sensors and actuators, home automation and networking, automatic meter reading/load management, consumer electronics/entertainment, and asset management. Commercial market segments include the following:

- Industrial monitoring and control
- Commercial building and control
- Process control
- Home automation
- Wireless automated meter reading (AMR) and load management (LM)
- Metropolitan operations (traffic, automatic tolls, fire, etc.)

- National security applications: chemical, biological, radiological, and nuclear wireless sensors
- Military sensors
- Environmental (land, air, sea) and agricultural wireless sensors.”

(Sohraby, Minoli, & Znati, 2007)

2.2.3 Wireless sensor network

This chapter explains the wireless sensor network which was established in this project. It consists of:

- Gateway NI WSN 9791

The NI WSN-9791 Ethernet gateway coordinates communication between distributed measurement nodes and the host controller in NI wireless sensor network (WSN). The gateway has a 2.4 GHz, IEEE 802.15.4 radio based on ZigBee technology to collect measurement data from the sensor network and a 10/100 Mbit/s Ethernet port to provide flexible connectivity to a Windows or LabVIEW Real-Time OS host controller. It requires 9 to 30V DC external power supply (National Instruments). The gateway is connected to host controller via a Ethernet cable.

- Node NI WSN 322

The NI WSN-3202 measurement node is a wireless device that provides four ±10 V analog input channels and four digital Input/Output channels. It is powered with four 1.5 V, AA alkaline cells. It can be externally powered with a 9 to 30 V supply. It transmits data to the WSN gateway wirelessly at 2.4 GHz radio based. (National Instruments)

The following Figure 2-7 shows the wireless sensor network.

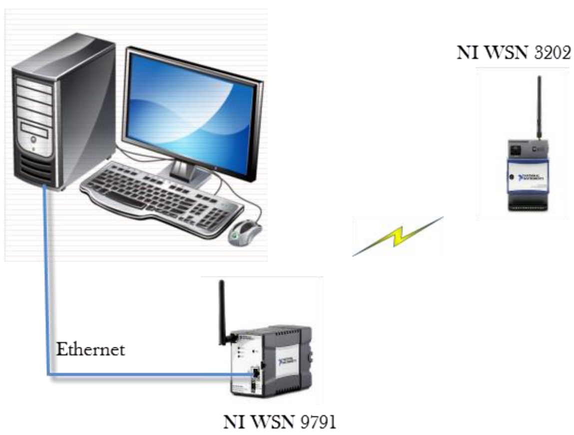


Figure 2-7 Wireless sensor network

Part II Methodology

3 Programming of Datalogger 3634

The following chapter refers to how the weather station was programmed and set up. First the Datalogger was programmed and then LabVIEW programming was done in order to show real time data and data logging in SQL server.

To convert the raw data into engineering units, the Datalogger must be programmed i.e.; the parameter names, units and calibration coefficients must be entered. The calibration coefficients for each sensor are given in specification sheet provided by the vendor.

Programming of Datalogger can be done in three different ways:

1. Programming by two control switches located on the Datalogger
2. Using a computer program “HyperTerminal”
3. Using a modem

3.1 Programming by control switches

To enter the programming mode, Mode Switch was turned to the MENU position. A menu appeared on the LCD. Figure 3-1 shows the LCD screen of Datalogger when entering the programming mode.

```
Main    [Prev]    _______Main_Menus________
Menu    [Enter]   >Display Last Data Set  <
        [Next]    |Buzzer Setting         |
                  |Channel Settings       |
```

Figure 3-1 LCD display of Datalogger (Operating manual Dataloggers , 2007)

The test in square brackets on the LCD is an operating key or field. The field **[Prev]** and **[Next]** is for scrolling through all menu items. The **[Enter]** field is for entering the menu item inside the “> <” brackets. To move the cursor to another field, Function Switch was set in the POS position and then Mode Switch was pressed towards the SET position. To enter a field or a menu item, Function Switch was set to CHAR position and Mode Switch was pressed towards the SET position. (Operating manual Dataloggers , 2007) After entering into Main Menus items, the parameters with calibration coefficients and units were entered which are shown in Figure 3-4. Date and time were entered by enter into ‘Set Date and Time’ menu. ‘No’ were chosen by entering into menu ‘Display Raw Data’. 5 were selected for number of channels in ‘Set Number of Channels’.

3.2 Programming by using HyperTerminal

The easiest way of programming is to use the computer and HyperTerminal program. First baud rate in Datalogger was read. Control switches were used to enter into “Set Baud Rate” in Datalogger. Noted the baud rate and turned the Datalogger OFF. (Operating manual Dataloggers , 2007)

Following steps were carried out to program Datalogger.

1. Started the HyperTerminal and choose the name for project ‘MS Project’.
2. Selected the COM1 port. Then used the following settings in Figure 3-2. Baud rate should correspond to the Datalogger’s baud rate.

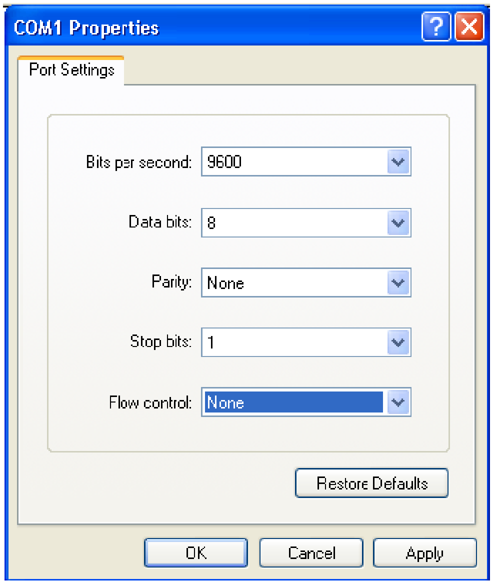


Figure 3-2 COM port settings

3. Connected the cable 3204 to computer and Datalogger’s Com-Port. Set the mode switch in “Menu” position then the following “Setup” menu was appeared. See Figure 3-3.

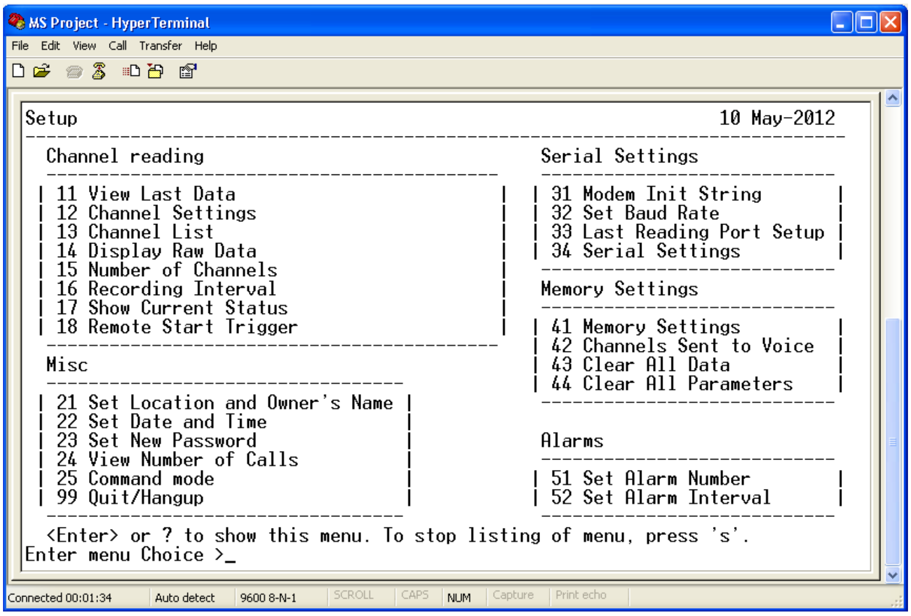


Figure 3-3 Setup menu of the Datalogger

In this menu all the sensors are setup with their corresponding calibration coefficients and other essential settings are made.

4. Weather station 2700 includes 3 sensors. The wind speed sensor gives out two values: wind speed and wind gust. There is a reference channel in the Weather Station 2700 which specifies the weather station. Also there is a battery voltage channel. Now the total number of channels with corresponding coefficients become: See Figure 3-4.

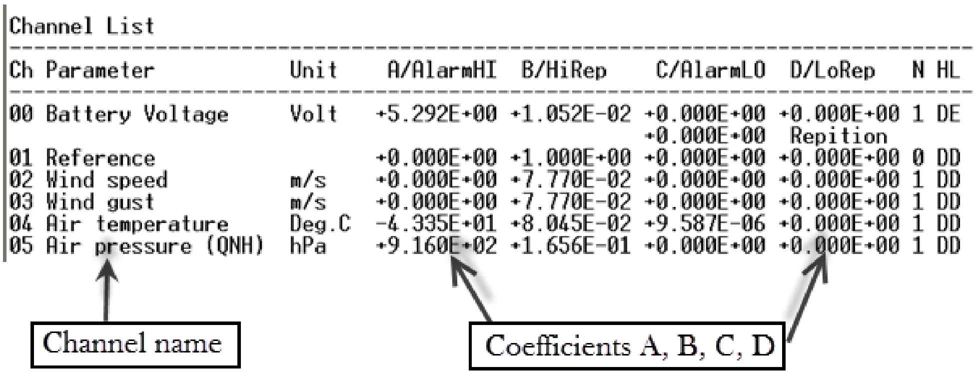


Figure 3-4 Channels list

These parameters have to be configured into Datalogger using HyperTerminal program.

5. To enter the channels, pressed 12 then the following screen was shown. Then corresponding unit and coefficients A, B, C and D for Battery Voltage channel were entered.

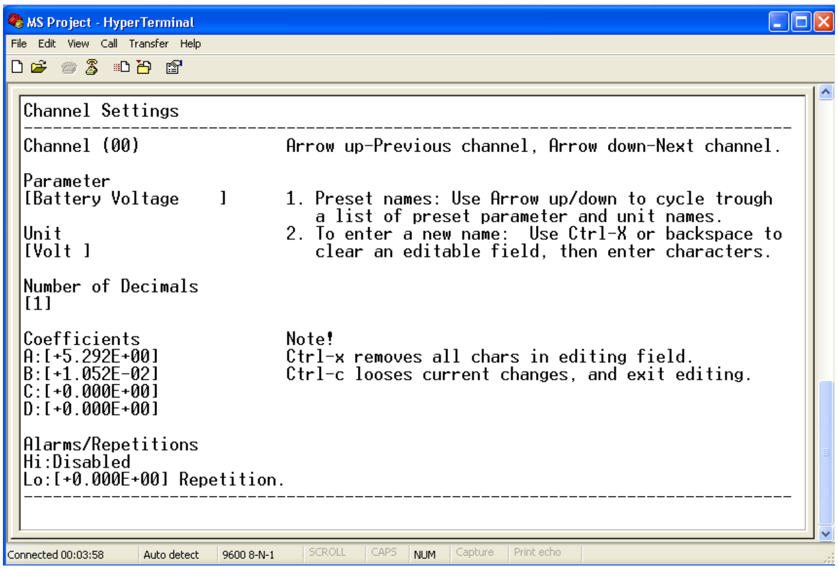


Figure 3-5 Settings for battery voltage channel

6. To setup the Reference channel the corresponding unit and coefficients were entered. Selected the channel and looked through the channel list by pressing Arrow up or Arrow down key.

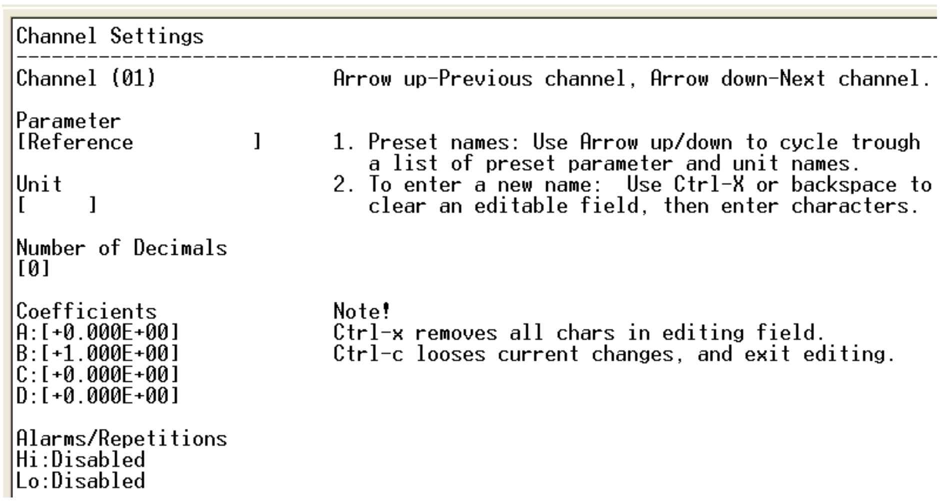
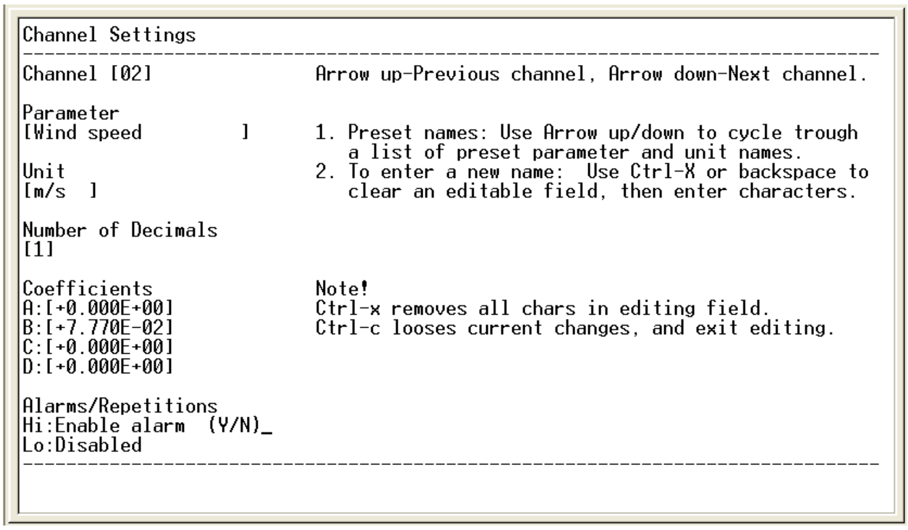


Figure 3-6 Settings for Reference channel

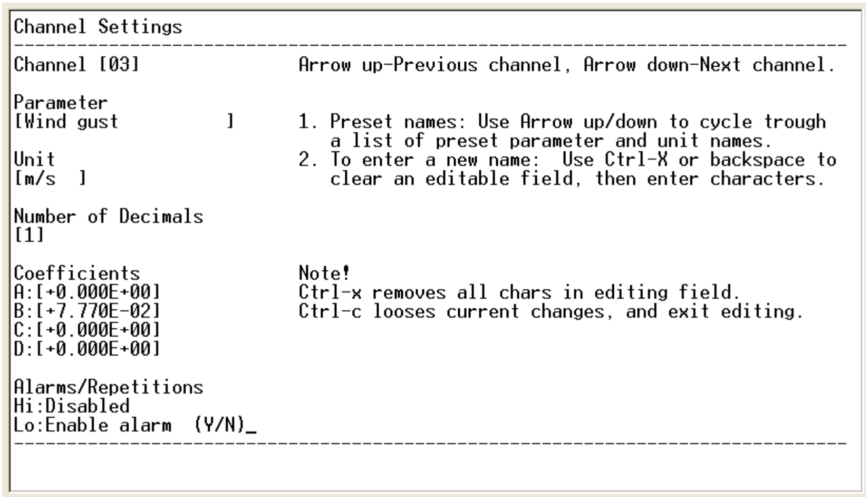
7. To setup the Reference channel the corresponding unit and coefficients were entered.
8. Wind speed channel were setup using following settings.



```
Channel Settings
--------------------------------------------------------------------------------
Channel [02]                Arrow up-Previous channel, Arrow down-Next channel.

Parameter
[Wind speed          ]      1. Preset names: Use Arrow up/down to cycle trough
                               a list of preset parameter and unit names.
Unit                        2. To enter a new name:  Use Ctrl-X or backspace to
[m/s  ]                        clear an editable field, then enter characters.

Number of Decimals
[1]

Coefficients                Note!
A:[+0.000E+00]              Ctrl-x removes all chars in editing field.
B:[+7.770E-02]              Ctrl-c looses current changes, and exit editing.
C:[+0.000E+00]
D:[+0.000E+00]

Alarms/Repetitions
Hi:Enable alarm  (Y/N)_
Lo:Disabled
--------------------------------------------------------------------------------
```

Figure 3-7 Settings for Wind speed channel

9. Wind gust channel were setup using following settings. See Figure 3-8.



```
Channel Settings
--------------------------------------------------------------------------------
Channel [03]                Arrow up-Previous channel, Arrow down-Next channel.

Parameter
[Wind gust           ]      1. Preset names: Use Arrow up/down to cycle trough
                               a list of preset parameter and unit names.
Unit                        2. To enter a new name:  Use Ctrl-X or backspace to
[m/s  ]                        clear an editable field, then enter characters.

Number of Decimals
[1]

Coefficients                Note!
A:[+0.000E+00]              Ctrl-x removes all chars in editing field.
B:[+7.770E-02]              Ctrl-c looses current changes, and exit editing.
C:[+0.000E+00]
D:[+0.000E+00]

Alarms/Repetitions
Hi:Disabled
Lo:Enable alarm  (Y/N)_
--------------------------------------------------------------------------------
```

Figure 3-8 Settings for Wind gust

10. Air temperature channel was configured as shown in the following Figure 3-9.

```
Channel Settings
-------------------------------------------------------------------------------
Channel [04]                  Arrow up-Previous channel, Arrow down-Next channel.

Parameter
[Air temperature    ]         1. Preset names: Use Arrow up/down to cycle trough
                                 a list of preset parameter and unit names.
Unit                          2. To enter a new name:  Use Ctrl-X or backspace to
[Deg.C]                          clear an editable field, then enter characters.

Number of Decimals
[1]

Coefficients                  Note!
A:[-4.335E+01]                Ctrl-x removes all chars in editing field.
B:[+8.045E-02]                Ctrl-c looses current changes, and exit editing.
C:[+9.587E-06]
D:[+0.000E+00]

Alarms/Repetitions
Hi:Enable alarm  (Y/N)_
Lo:Disabled
-------------------------------------------------------------------------------
```

Figure 3-9 Settings for air temperature channel

11. Channel for air pressure were configured using the following settings. See Figure 3-10.

```
Channel Settings
-------------------------------------------------------------------------------
Channel (05)                  Arrow up-Previous channel, Arrow down-Next channel.

Parameter
[Air pressure (QNH) ]         1. Preset names: Use Arrow up/down to cycle trough
                                 a list of preset parameter and unit names.
Unit                          2. To enter a new name:  Use Ctrl-X or backspace to
[hPa  ]                          clear an editable field, then enter characters.

Number of Decimals
[1]

Coefficients                  Note!
A:[+9.160E+02]                Ctrl-x removes all chars in editing field.
B:[+1.656E-01]                Ctrl-c looses current changes, and exit editing.
C:[+0.000E+00]
D:[+0.000E+00]

Alarms/Repetitions
Hi:Disabled
Lo:Disabled
-------------------------------------------------------------------------------
```

Figure 3-10 Settings for air pressure channel

Menu choice 13 displays all the configured channels on the HyperTerminal screen.

```
Enter menu Choice >13

Channel List
-------------------------------------------------------------------------------
Ch Parameter            Unit    A/AlarmHI  B/HiRep    C/AlarmLO  D/LoRep    N HL
-------------------------------------------------------------------------------
00 Battery Voltage      Volt    +5.292E+00 +1.052E-02 +0.000E+00 +0.000E+00 1 DE
                                                      +0.000E+00  Repition
01 Reference                    +0.000E+00 +1.000E+00 +0.000E+00 +0.000E+00 0 DD
02 Wind speed           m/s     +0.000E+00 +7.770E-02 +0.000E+00 +0.000E+00 1 DD
03 Wind gust            m/s     +0.000E+00 +7.770E-02 +0.000E+00 +0.000E+00 1 DD
04 Air temperature      Deg.C   -4.335E+01 +8.045E-02 +9.587E-06 +0.000E+00 1 DD
05 Air pressure (QNH)   hPa     +9.160E+02 +1.656E-01 +0.000E+00 +0.000E+00 1 DD
```

Figure 3-11 Configured channels

12. Menu choice 12 is for raw data settings. Following settings were used.
    Displaying raw data: NO

13. Number of channels or sensors is selected in menu choice 15. Hence 5 were selected. See Figure 3-12.

```
  ---------------------------------
  <Enter> or ? to show this menu. To stop lis
Enter menu Choice >15

Set number of channels
--------------------------------------------
Select number of channels (Min:2 Max:05): 5_
```

Figure 3-12 Number of channels

14. To enter the recording interval of data, menu choice number 16 were pressed. Recording interval means here that each sensor value will be read in the Datalogger according to this time interval. See Figure 3-13.

```
Set Interval
-----------------------------------------------------------
 0 Remote start     4     2.0 min.     8    30.0 min.
 1 Nonstop          5     5.0 min.     9    60.0 min.
 2     0.5 min.     6    10.0 min.    10   120.0 min.
 3     1.0 min.     7    20.0 min.    11   180.0 min.
-----------------------------------------------------------

Select interval:_
```

Figure 3-13 Set the time interval between sensor readings

Menu choice 17 displays the current running values shown by sensors. When programming was completed and selected 17 in HyperTerminal then following sensor values were shown. See Figure 3-14.

```
Enter menu Choice >17

Show Elapsed Sequence
---------------------------------------------



Show Channels
---------------------------------------------
To stop list- press 's' or Ctrl-c
---------------------------------------------

Interval: Nonstop      No.of channels:  5

01 Reference              780
02 Wind speed               1.2  m/s
03 Wind gust                2.5  m/s
04 Air temperature         10.6  Deg.C
05 Air pressure (QNH)    1013.2  hPa

01 Reference              780
02 Wind speed               2.7  m/s
03 Wind gust                4.7  m/s
04 Air temperature         10.6  Deg.C
05 Air pressure (QNH)    1013.0  hPa
```

Figure 3-14 Current sensor values

15. Location and owner name of the Weather Station can be set by pressing 21. See Figure 3-15.

```
  <Enter> or ? to show this menu. To stop listing of menu, press 's'.
Enter menu Choice >21
Owner's name: (Telemark University College    )
Location    : (Porsgrunn                      )

Enter menu Choice >_
```

Figure 3-15 Owner's name and location of AWS 2700

16. Date and time were set using menu choice 22.
17. The baud rate of Datalogger was changed by entering 32 in the menu choice in hyper terminal. The baud rate 9600 was selected and is shown in Figure 3-16 Baud rate options menu Figure 3-16

```
Enter menu Choice >32

Set new Baud Rate
-----------------------------------------------
Current Baud Rate is: 9600 baud
Next Baud Rate is   :  Same as current.

0 - 1200   1 - 2400   2 - 4800   3 - 9600
Enter choice :3
Next Baud Rate is   : 9600 baud
Enter menu Choice >_
```

Figure 3-16 Baud rate options menu

18. Last reading port setup was done by selecting menu choice 33. See Figure 3-17.

```
Last Reading Port Setup
------------------------------------------------------
Is the Last Reading port connected to a VOICE (Y/N)NO
Last Reading port is set to: PC

Enter menu Choice >_
```

Figure 3-17 Last reading port setting

4 LabVIEW programming

The following chapter describes the data acquisition and presentation from AADI sensors and integration of WSN system into AADI sensors. The AADI sensors data is logged in Datalogger that must be collected in a computer and manipulated. The Datalogger has an interface of RS-232C serial communication. LabVIEW has VIs which supports serial communication.

A LabVIEW program was made to display real time data of AADI and WSN sensors. The Figure 4-1 shows flowchart of program which was built in LabVIEW. The front panel and block diagram is given in Appendix C. Note that each case structure in programming code is given in appendix. The sensor data is represented by indicators and trend graphs in front panel.

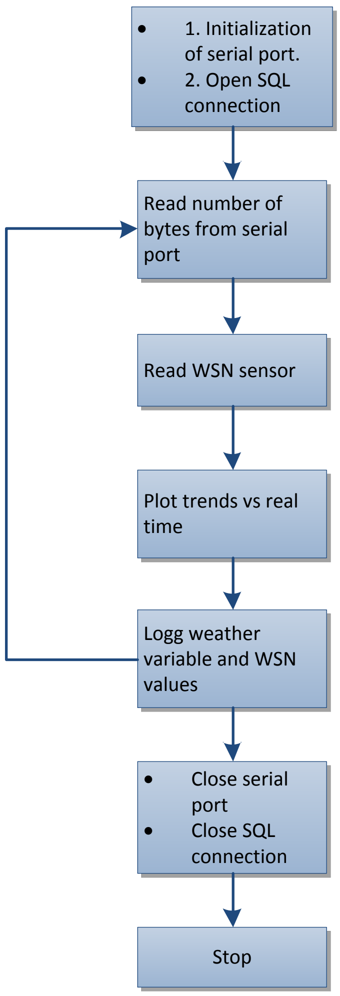


Figure 4-1 Flowchart diagram of LabVIEW program

The block diagram was made as state machine. A state machine consists a case structure inside a while loop. Case structure consists of many cases which contain a program code or a part of program code. In the block diagram first case ‘Init’ inside while loop contains the code which intilializes the serial communication. Second case ‘VisaRead’ contains the code which reads the specified number of bytes from com port connected to computer. Also, the output string is manipulated and sensor values are extracted from string and indicated on front panel. The third case ‘Read WSN sensor’ displayes WSN senor values. The next case ‘trending’ plots real senors values vs real time and date. In the next case ‘Datalogging’ values of each sensor are logged into a server called SQL server. In the next case ‘Exit’ serial communication is closed by a VI called VISA close. Each case in case structure specifies the differenct states in stare machine. Shift register carries values from one state to another. The following figure shows that when the program is run, it starts from state ‘Init’ and continues to the next state specified in case structure.

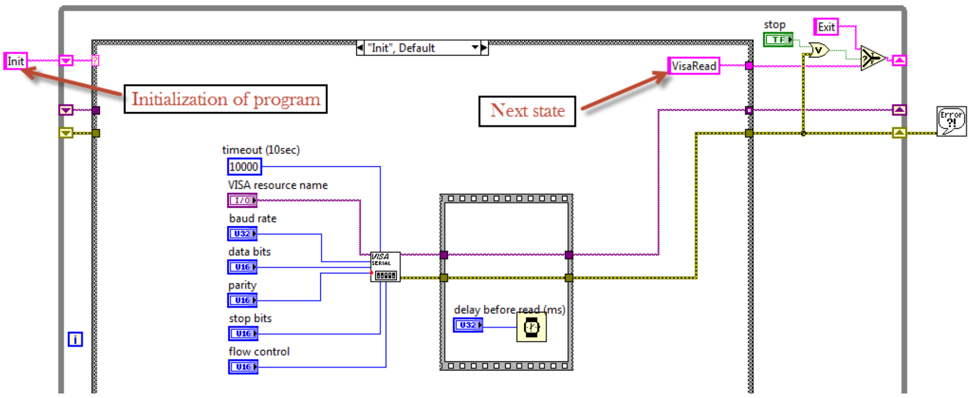


Figure 4-2 State machine (NI, 2006)

The following Figure 4-3 shows ‘VISA Configure Serial Port VI’ in the right while configuration settings in the left. This VI initializes serial port communication. ‘Bytes read’ indicator shows bytes read by the VISA read VI. ‘VISA resource name’ specifies the source which will be opened which is serial communication port RS-232. Baud rate is the rate at which data transmission is occurred. ‘Data bits’ shows the number of bits in incoming data.

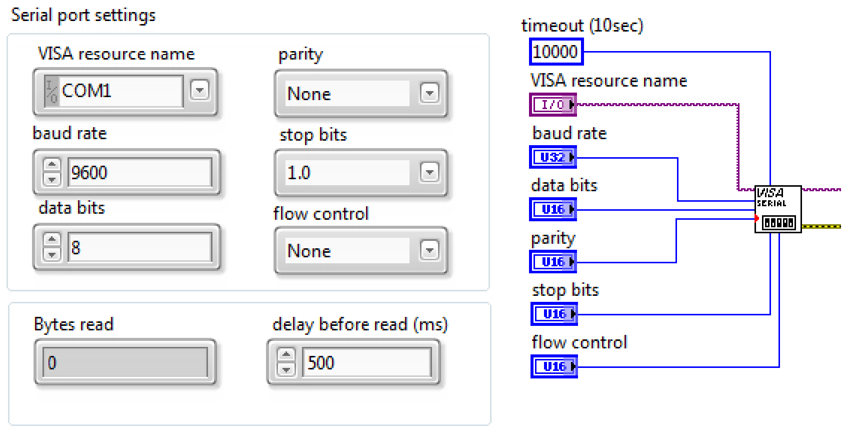


Figure 4-3 VISA configure serial port VI on right and settings on left (NI, 2006)

The following VI plots the data vs date and time at which data is being displayed.

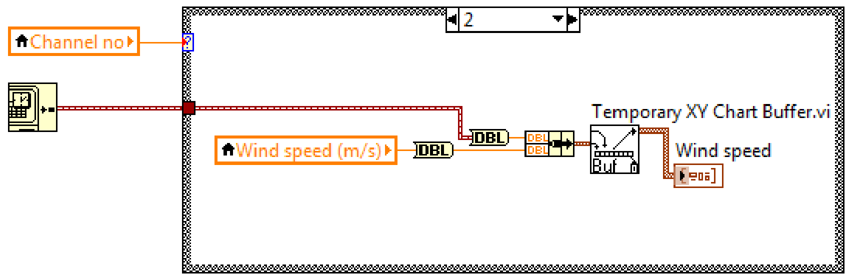


Figure 4-4 Block diagram for trend plot vs real time

The WSN system was integrated into weather station using following code.

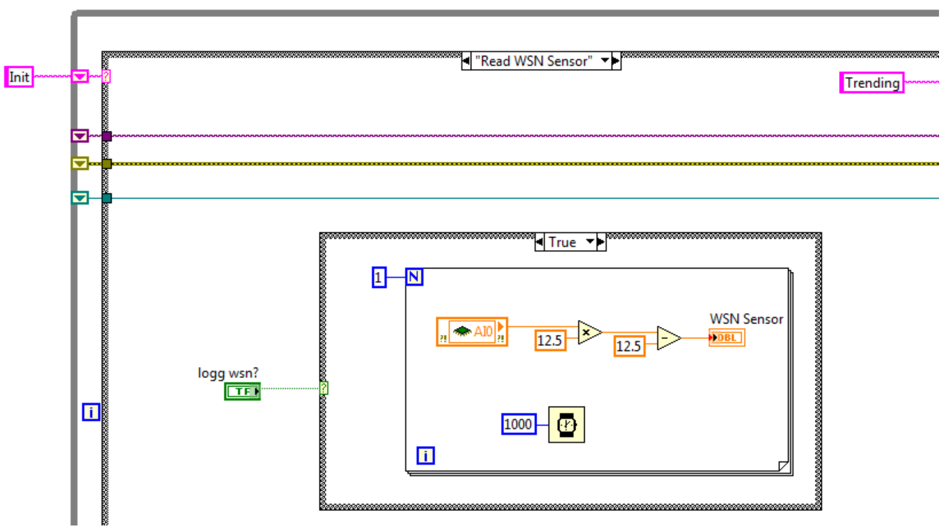


Figure 4-5 Block diagram for WSN

5 Data logging

Data logging is the process where information from sensing devices is stored in a device called Datalogger. This information is used to study the systems which are monitored. Data logging is used in many systems for example a black box is used to collect flight information which may be used latter. Data logging in weather station is important in the sense that weather condition is predicted on the basis of collected data.

Datalogger was used to store information from weather sensors. Now the data must be downloaded and analyzed in order to know about the information which was captured by sensors. The data can be downloaded in text files or in other format and stored in computer. These files can be opened in spreadsheet. One solution is that data can be saved in software like SQL server. SQL is a relational database management system (RDBMS) from Microsoft. LabVIEW is a powerful programming tool that provides among others database communication. One must establish a connection to a database before accessing data in a table or executing a SQL query. There are different methods in order to connect to Database. In computing, Open Database Connectivity (ODBC) provides a standard software API method for using database management systems (DBMS). (Halvorsan, 2011) Here open database connectivity (ODBC) Database Source Name (DSN) was used. ODBC connection named 'MSProject' was created in 'ODBC Data Source Administrator'. The following Figure 5-1 shows the created database connection in ODBC Data Source Administrator.

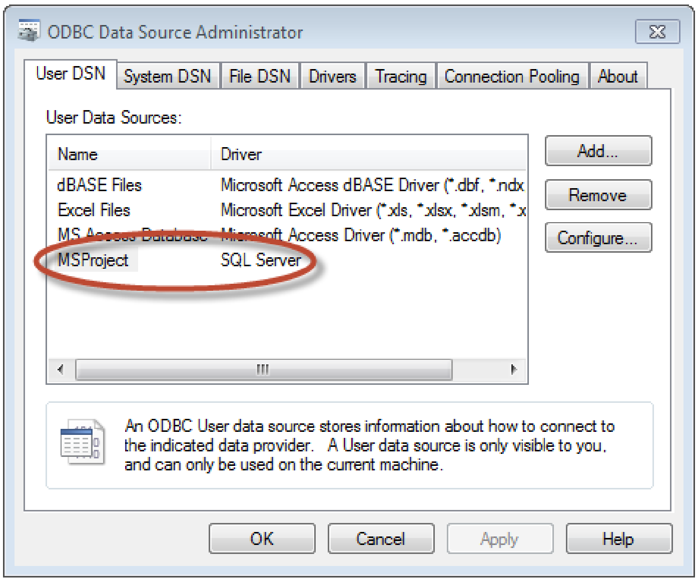


Figure 5-1 ODBC data source administrator with created database connection

Two tables were created in SQL server to store the data. First table 'TAG' contains parameters while other table 'TAGDATA' contains records. The figure shows the overview of complete database where two tables with their columns, database diagram and views are shown. ‘View’ in SQL server is a table which is formed by selecting columns from other tables in the database. A ‘View’ named ‘GetWeatherData’ was created as shown in Figure 5-3.

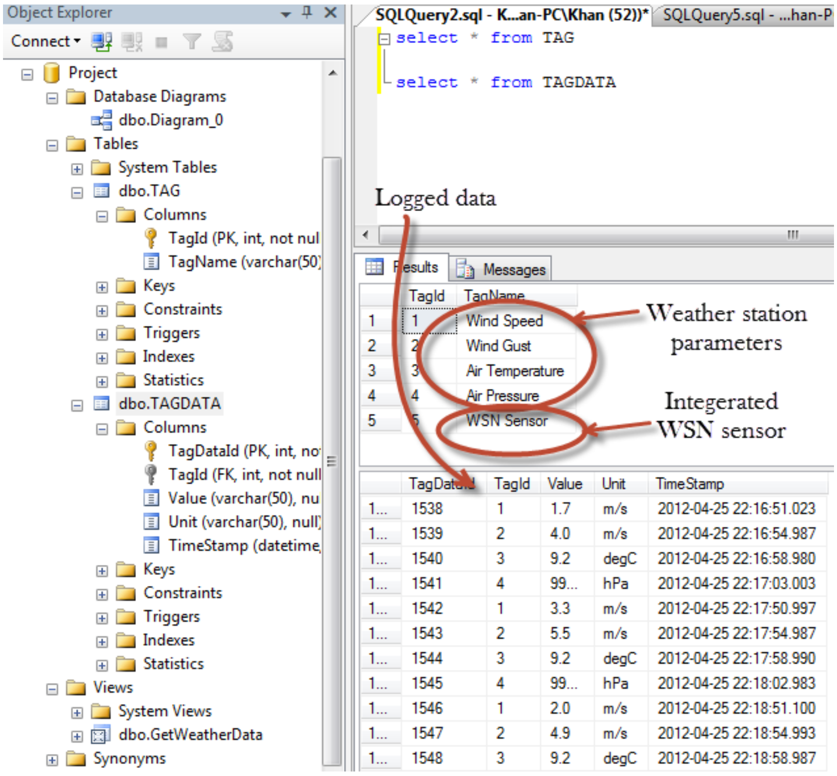


Figure 5-2 Database created in SQL server

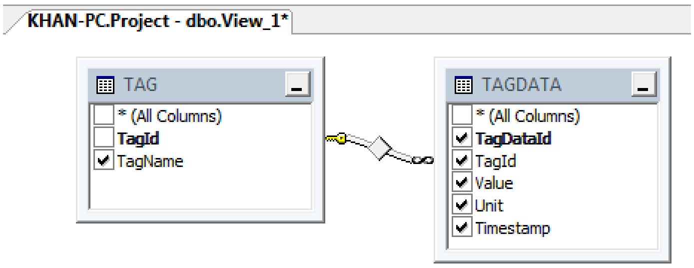


Figure 5-3 VIEWS created in SQL server

The LabVIEW code which inserts data into SQL server is given in Appendix C. The outermost case structure has two cases i.e; whether one may want data logging or not. The second case has five cases where in each case one weather variable stores data in SQL server by executing SQL query. ‘Connection In’ is the reference of connection which was initialized in state machine. Also this connection was closed in state machine. ‘Format into string’ VI was used which gives a string output which is SQL query. This SQL query is executed by ‘execute VI’ and inserts data into a table in SQL server.

6 Wireless sensor network implementation

To study the WSN, an experiment was done in which a node NI WSN 3202, a gateway NI WSN 9791 and a temperature sensor PT-100 were used. The gateway was connected to computer via Ethernet cable and power was given via a power supply with output in the range of 12V and1.25A. The battery powered node was connected to the temperature transmitter. The connection diagram of node NI WSN 3202, temperature transmitter and temperature sensor PT-100 is shown in Figure 6-1 and diagram of overall WSN is shown in the Figure 6-3.

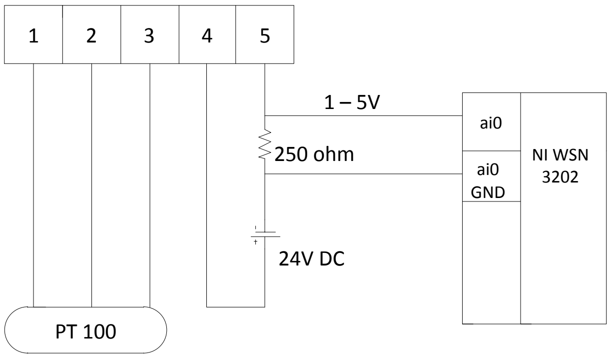


Figure 6-1 Temperature sensor, transmitter, resistor and NI WSN node connection

Temperature transmitter has an output of 0 – 50°C and voltage drop across resistor is 1 – 5V. So 1 – 5V must be scaled to 0 – 50°C. Temperature of less than 0°C and greater than 5°C cannot be measured by this system. Scaling was accomplished in LabVIEW. The scaling diagram is shown in following Figure 6-2.

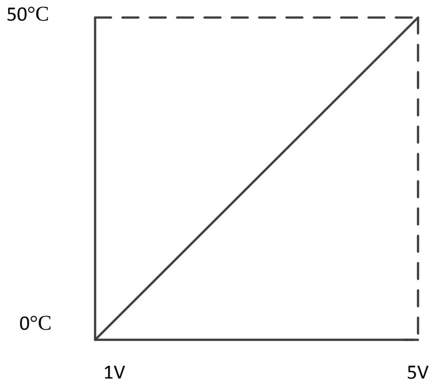


Figure 6-2 Scaling diagram

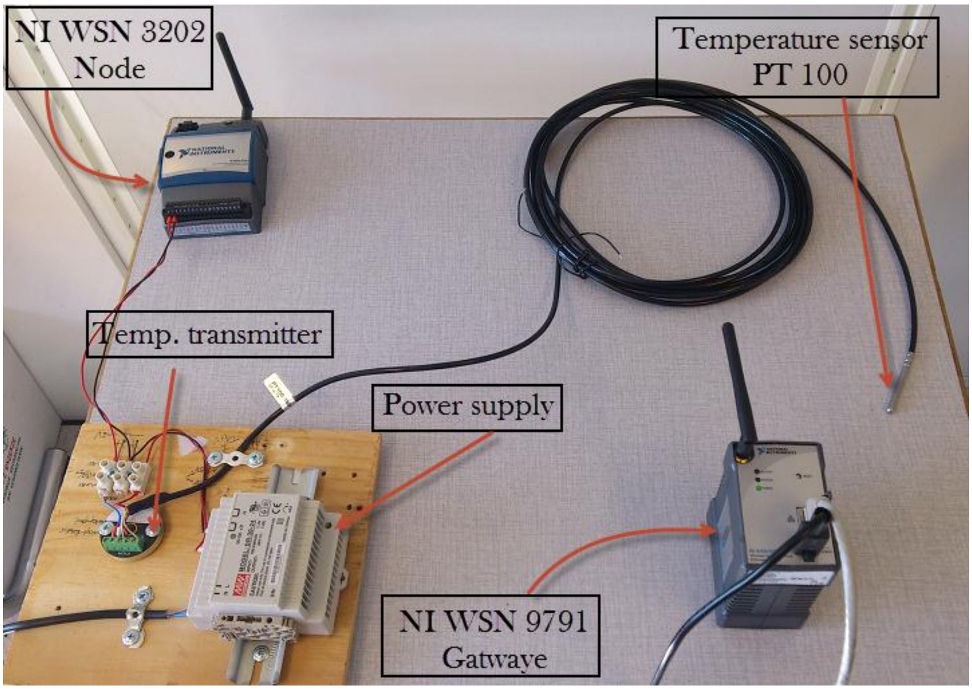


Figure 6-3 Wireless sensor network

The following steps were carried out to configure the WSN devices in LabVIEW.

1. LabVIEW 2011 was installed on a computer.
2. NI WSN module was installed along with LabVIEW 2011.
3. The gateway was connected to computer via Ethernet cable and power cable was connected to it.
4. Battery powered NI WSN node was connected to temperature transmitter.
5. When ‘Measurement and Automation Explorer’ were opened and ‘Remote Systems’ was expanded, the gateway was already detected by MAX. See Figure 6-4
6. When gateway was selected the node was already detected by the MAX, was connected to gateway and was able to communicate with gateway.

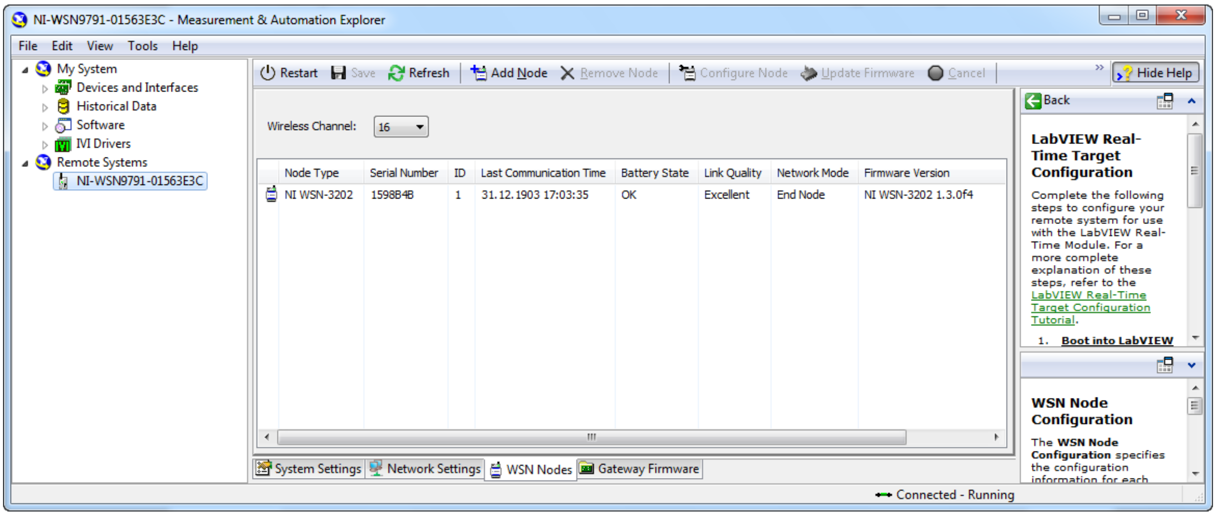


Figure 6-4 MAX showing the gateway and node

7. Thus WSN devices were configured in MAX.

In order to be able to detect or add a node in MAX one must make sure that the subnet mask address in MAX must be same as given in 'Internet Protocol version 4(TCP/IP) Properties' in the computer. If not, it will not be possible to add a new node in MAX because IP settings will be inconsistent. The following Figure 6-5 shows that subnet mask settings were same in both MAX and adapter settings.

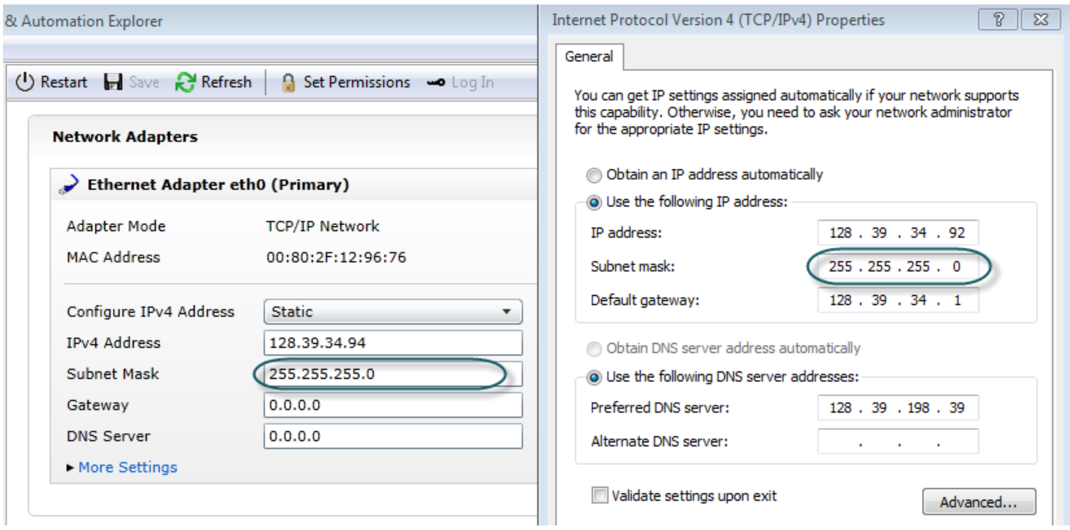


Figure 6-5 Ethernet adapter settings

8. Now a new project was created in LabVIEW with the name of 'Master theses'.
9. The project name was right clicked and New >> Targets and Devices was selected.

10. The existing gateway was selected under 'WSN Gateway' in 'Existing target or device'. Figure 6-6 shows project explorer window with gateway.

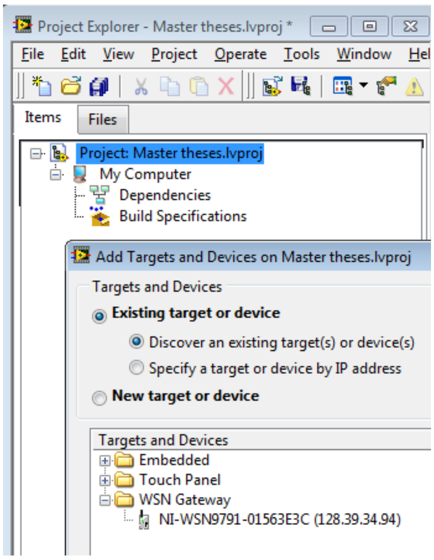


Figure 6-6 Project explorer window showing gateway

11. After selection of gateway and pressing OK button the gateway was added to the project explorer window.
12. Then NI-WSN9791 was expanded to see the node and its I/O variables. Figure 6-7.

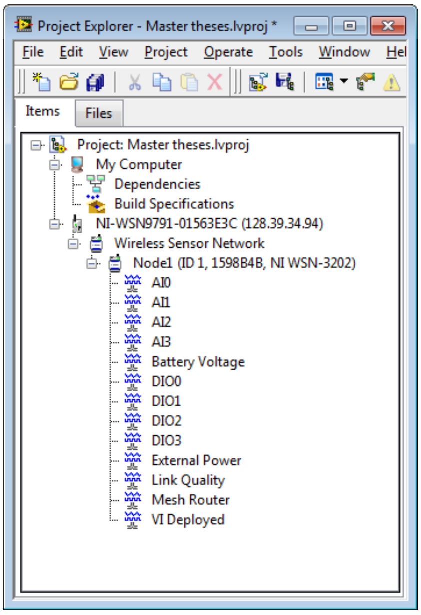


Figure 6-7 Project window explorer with I/O variables of the node

13. Now a new VI was created to see the WSN devices response. The screen shots are shown in the following Figure 6-8.

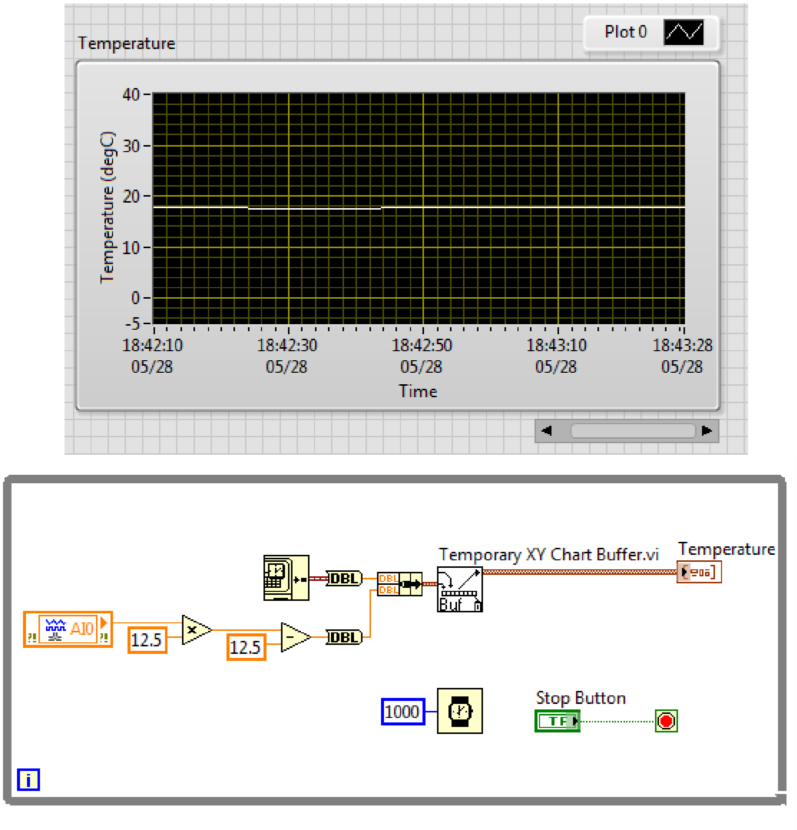


Figure 6-8 Front panel and block diagram of WSN

Part III Result, analysis and conclusion

7 Result and analysis

When the LabVIEW programming and practical implementation of AWS and WSN were completed, the system was run. It gave the excellent results and it worked like it should. The results of AWS and WSN were compared with the weather station installed in Telemark University College. Figure 7-1 shows real time display of AWS and Figure 7-2 shows WSN display while real time display of weather station installed at Telemark University College Porsgrunn is given in Figure 7-3. It can be seen that air temperature in Telemark University College, Porsgrunn is 17°C registered by AWS and 17.57°C WSN while 15.6°C shown by weather station installed at TUC. So AWS 2700 gave realistic values.

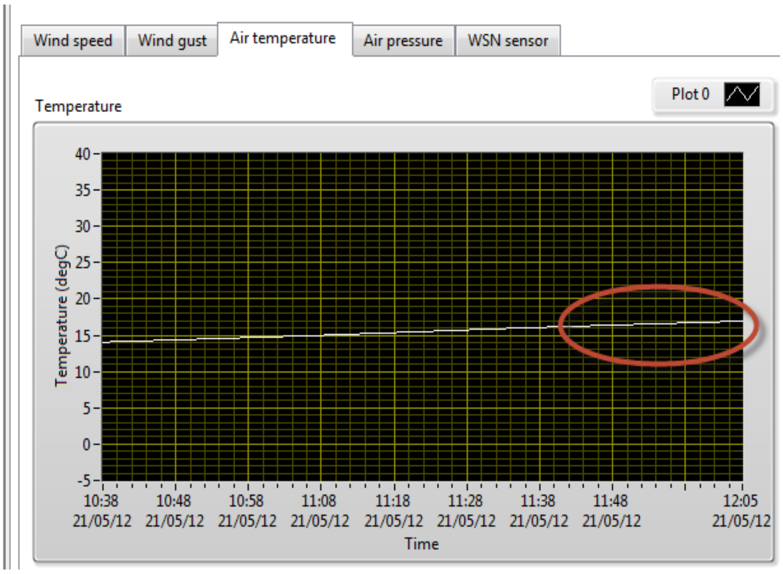


Figure 7-1 Real time display of AWS and weather station at TUC

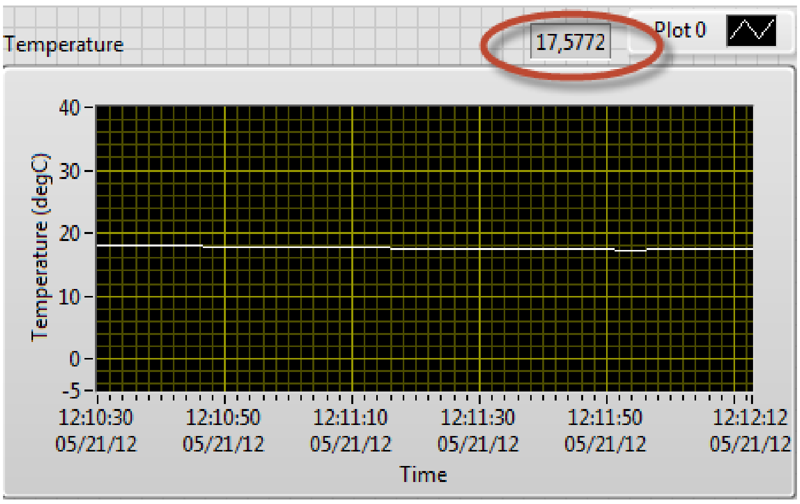
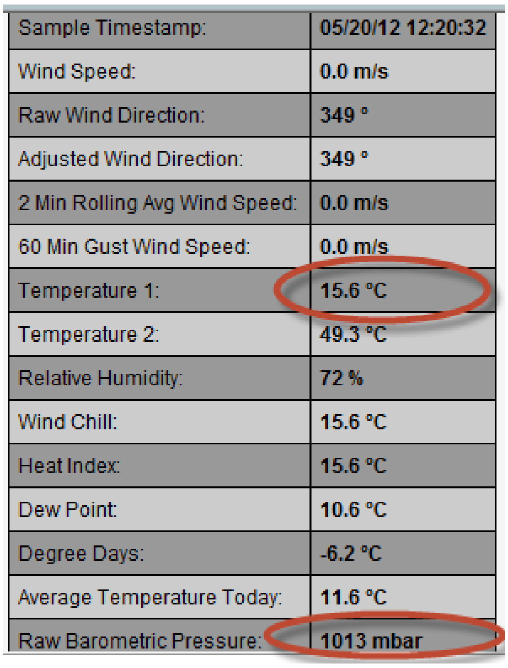


Figure 7-2 Real time display of WSN sensor

When air pressure represented by AWS 2700 given in Figure 7-4 was compared with those given in Figure7-3, it can be seen that there is 1hPa difference and resultant values of AWS are quit realistic.



Sample Timestamp:	05/20/12 12:20:32
Wind Speed:	0.0 m/s
Raw Wind Direction:	349 °
Adjusted Wind Direction:	349 °
2 Min Rolling Avg Wind Speed:	0.0 m/s
60 Min Gust Wind Speed:	0.0 m/s
Temperature 1:	15.6 °C
Temperature 2:	49.3 °C
Relative Humidity:	72 %
Wind Chill:	15.6 °C
Heat Index:	15.6 °C
Dew Point:	10.6 °C
Degree Days:	-6.2 °C
Average Temperature Today:	11.6 °C
Raw Barometric Pressure:	1013 mbar

Figure 7-3 Real time display of weather station installed at TUC, Porsgrunn (Halvorsen)

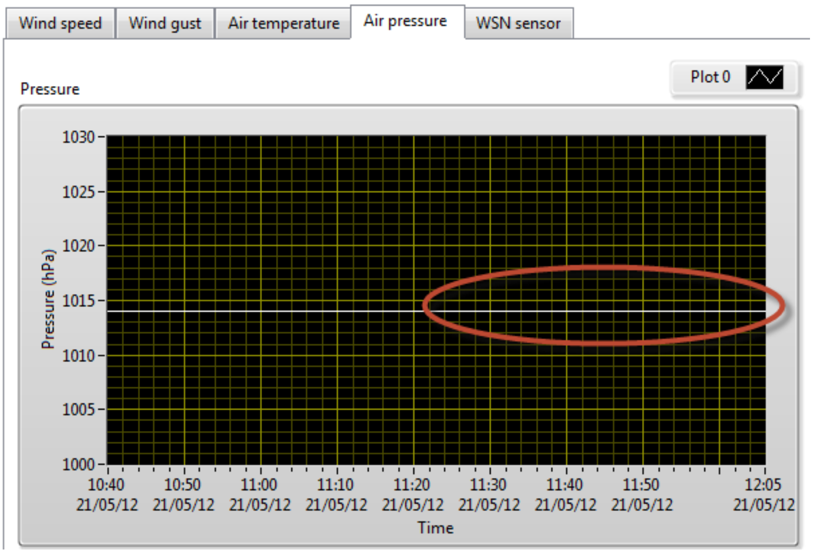


Figure 7-4 Real time display of AWS where air pressure is shown

In order to compare the results with Norwegian Meteorological Institute, the data was downloaded from www.eklima.no and plotted. The data is given in Appendix D.

It can be seen in Figure 7-5 that the pressure was about 1008.5 hPa on average at Telemark University College Porsgrunn on 16th May 2012 while average pressure for same day was 992 hPa in Skien which is about 12 km from TUC. Skein weather station of Norwegian Metrological Institute was selected because it was nearest station to the station used in this project. It can be seen in the Figure 7-5 that trend curve is almost same but values are different. Possible reason is the difference in elevation as pressure decreases with increasing elevation. (Wikipedia, 2012)

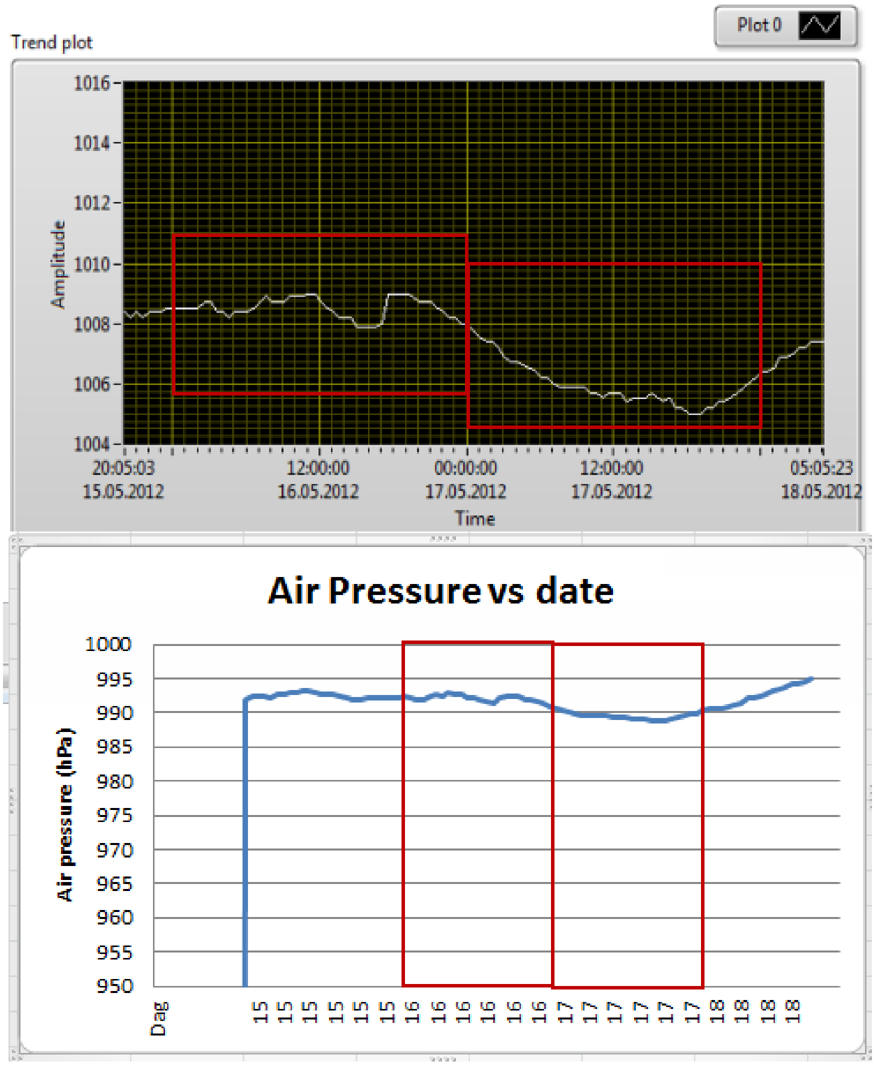


Figure 7-5 Comparison of air pressure (Norwegian Meteorological Institute)

When air temperature registered by AWS was compared with Norwegian Metrological Institute's data, the resultant graph is shown in Figure 7-6. The trend graphs show the same behavior but different values. Reason is that both the stations are about 12 km away from each other and temperature cannot be the same over an area. Wind speed can also affect the temperature. AWS should be mounted in open air and there should not be any obstacle which could affect wind sensor values. If there is any obstacle near sensors then wind speed values will not be true. AWS was not mounted at roof of any building or in any other open place but rather placed in an open window and measurements were taken. Because of this, wind speed and wind gust measurements were not taken into results.

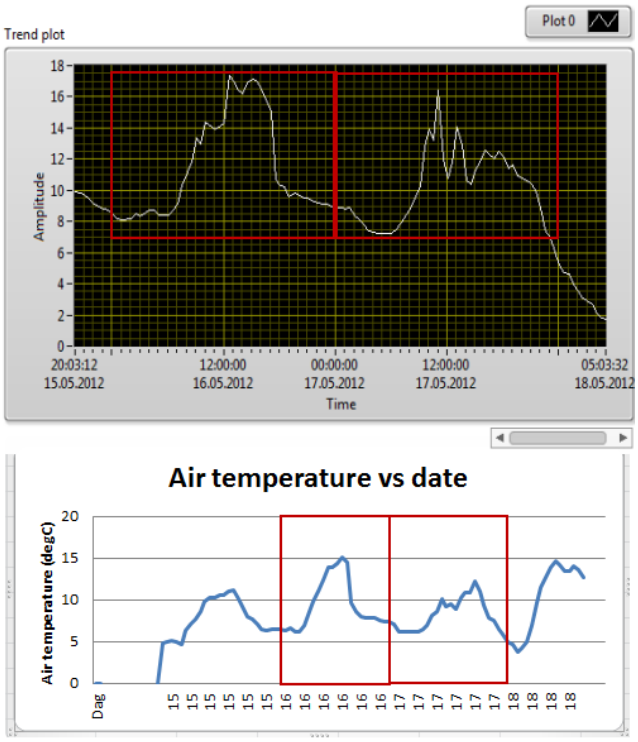


Figure 7-6 Comparison of air pressure (Norwegian Meteorological Institute)

The statistical plot of data is shown in Figure 7-7. The histogram was drawn using 10 bins. Standard deviation, variance arithmetic mean and median are also calculated.

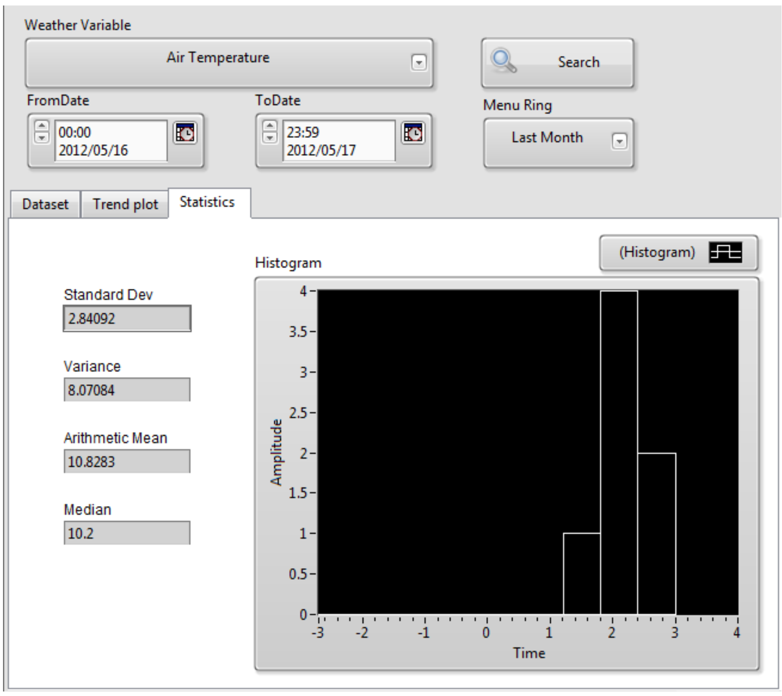


Figure 7-7 Statistical analysis

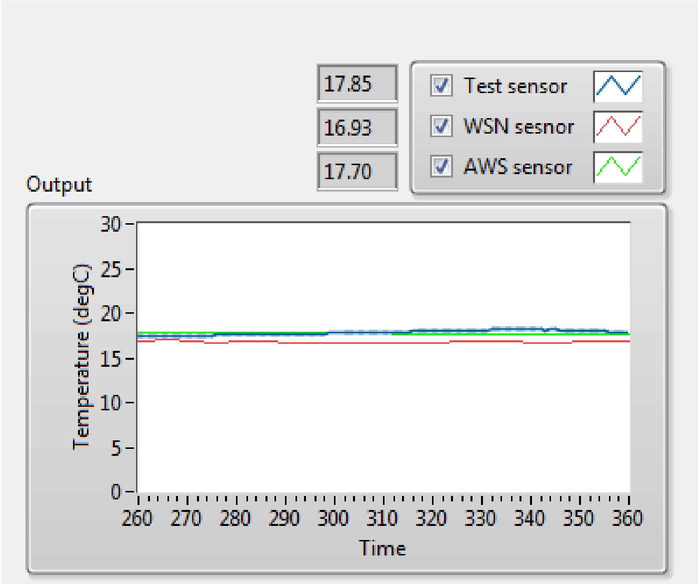


Figure 7-8 Comparison of air pressure

To verify the results, the temperature sensor values from AWS and WSN were compared with another test sensor. Test sensor was a NI USB – TC01, J – type thermocouple device. When the three temperature sensors were run they gave the result which is shown in Figure 7-8. As it can be seen in the Figure 7-7 that AWS sensor gave the value of 17.70°C while test sensor had a value of 17.85°C so temperature sensor of AWS gave values as expected. There was little variation in test sensor graph. There is a difference of 0.92°C in values of WSN and test sensor.

8 Conclusion and future work

The weather station setup and implemented LabVIEW program gives a satisfactory solution to the problems described in the beginning. The Datalogger was programmed by entering into it the parameter names, units and calibration coefficients. The coefficients for individual sensors were given in datasheet for each sensor from vendor. The sensors were mounted on a mast and connected to Datalogger. To retrieve data from Datalogger, LabVIEW coding was done which displayed real weather condition and log data into Microsoft SQL server. A wireless sensor networked system was configured comprising of a node, a gateway and a temperature sensor. This sensor network was integrated into automatic weather station and output of which was shown together with automatic weather station. Also another program was made using LabVIEW in order to read and analyze historical data. In LabVIEW coding of historical data analysis it was made possible to retrieve data for a desired period of time. Some intervals were selected for retrieving the data. For example it was possible to retrieve data for last 24 and 48 hours and last 30 days even a last year. Some statistics were used in data analysis part.

After completion of thesis the system worked as it should. The resultant output of weather variables was compared the weather station installed in Telemark University College, Porsgrunn and data from Norwegian Meteorological Institute.

There was no vision hardware used in this project therefore vision module was not used. Problem number two given in task description was not discussed because of lack of vision hardware.

Some suggestions for tasks that can be performed on subsequent projects are as follows:

- Improve visualization of weather variables output.
- Make mathematical models for future prediction of air temperature, pressure and wind speed. Make long term alerts for seven days ahead.
- Make the prediction whether there will be sun, rain, snow or fog.

9 Bibliography

Air Pressure Sensor. (2006, November). *Air Pressure Sensor 2810/2810 EX D161*. Bergen, Norway: Aanderaa Data Instruments AS.

Air Temperature Sensor. (2006, December). *Air Temperature Sensor 3455/3455 EX D276*. Bergern, Norway: Aanderaa Data Instruments AS.

Operating manual Dataloggers . (2007, Juli). *Operating manual Dataloggers 3660/3634*. Bergen, Norway: Aanderaa Data Instruments AS.

Wind speed sensor. (2008, April). *Wind speed sensor 2740/2740 EX D151*. Bergen, Norway: Aanderaa Data Instruments AS.

Dataloggers 3634 and 3660. (2009, April). Bergen, Norway: Aaderaa Data Instruments AS.

National Instruments. (2009, July 23). Retrieved April 2012, from www.ni.com: http://zone.ni.com/devzone/cda/tut/p/id/8707

Wikipedia. (2012). Retrieved april 2012, from wikipedia.org: http://en.wikipedia.org/wiki/Automatic_weather_station

Wikipedia. (2012, May 22). Retrieved May 24, 2012, from wikipedia.org: http://en.wikipedia.org/wiki/Atmospheric_pressure

Wikipedia. (2012). Retrieved April 2012, from www.wikepedia.org: http://en.wikipedia.org/wiki/Weather_station

Buratti, C., Martalò, M., Verdone, R., & Ferrari, G. (2011). *Sensor Networks with IEEE 802.15.4 Systems: Distributed Processing, MAC, and Connectivity (Signals and Communication Technology).* Springer.

Ferguson, S. S. (2007, December 27). *wikipedia.org*. Retrieved April 2012, from wikipedia.org: http://en.wikipedia.org/wiki/File:MQ-9_Reaper_in_flight_(2007).jpg

Halvorsan, H. P. (2011). *Database Communication in LabVIEW.*

Halvorsen, H. P. (n.d.). Weather Station at Telemark University College. http://home.hit.no/~hansha/.

Mahalik, N. P. (2007). Introduction. In *Sensor Networks and Configuration Fundamentals, Standards, Platforms,and Applications* (pp. 1-4).

National Instruments. (n.d.). Retrieved May 2012, from National Instruments: http://sine.ni.com/nips/cds/view/p/lang/en/nid/206919

National Instruments . (n.d.). Retrieved May 2012, from National Instuments: http://sine.ni.com/nips/cds/view/p/lang/en/nid/206921

NI. (2006, Septempber 6). *National Instruments Corporation*. Retrieved February 1, 2012, from http://zone.ni.com/devzone/cda/epd/p/id/2669

Norwegian Meteorological Institute. (n.d.). Retrieved May 2012, from www.eklima.no

Sohraby, K., Minoli, D., & Znati, T. (2007). *Wireless sensor networks, Technology, Protocols, and Applications.* John Wiley & Sons, Inc.

Wang, Q., Hassane, H., & Xu, K. (2005). *Handbook of sesnor networks: Compact wireless and wired sensing system.* CRC Press.

Appendices

Appendix A Problem description

Telemark University College
Faculty of Technology

FMH606 Master's Thesis

Title: Wireless Sensor Networking using AADI Sensors with WSN Coverage

Student: Babar Khan

TUC supervisor: Saba Mylvaganam, Hans-Petter Halvorsen,

External partner: Tom-Arne Danielsen, NI, AADI/ Norway

Task description:

This study will have focus on wireless sensor network (WSN) using existing NI modules and AADI modules using techniques for increasing WSN coverage. The activities planned for this project are

(1) Setting up and using the AADI sensors
(2) Usage of vision Modules for activation of certain types of sensors
(3) Integrating the data acquisition systems of AADI and NI-modules, possibly configuring the system architecture to accommodate wireless NI modules
(4) Configuring and running existing WSN system consisting of temperature sensors
(5) Modifying and developing existing LabVIEW programs to acquire and present the data from NI and AADI modules.
(6) Continuous presentation of data from the integrated system
(7) Delivery of written thesis following the guidelines and using the template from TUC

.

Task background:

Wireless networking has already penetrated the industries in various forms, for data transfer, communication and sensor networking. With the increased interest for wireless networking in general in the industry, naturally there is a need for looking into strategies of enhancing performance of wireless networking with respect to the possibility of integrating wireless networking with existing wired networks, extending coverage distance of wireless networks etc. Industrial users are interested predominantly in the following areas of study in wireless and integrating sensor systems from vendors other than NI

Student category:

For SCE students with experience in using LabVIEW.

Adress: Kjølnes ring 56, NO-3918 Porsgrunn, Norway. **Phone:** 35 57 50 00. **Fax:** 35 55 75 47.

Practical arrangements:

SysCon / TUC has the latest NI Wireless Sensor Networking Modules in addition to AADI sensors. SysCon / TUC collaborates with National Instruments, Norway and AADI. Necessary hardware and software will be provided by HiT. Work will be performed in Sensor Lab and Flow Lab. Possible interaction with organisations and research groups working with similar problems in Norway and abroad.

Filename Saba_Mylvaganam_master_thesis_proposal_24_11_2011_3.rtf

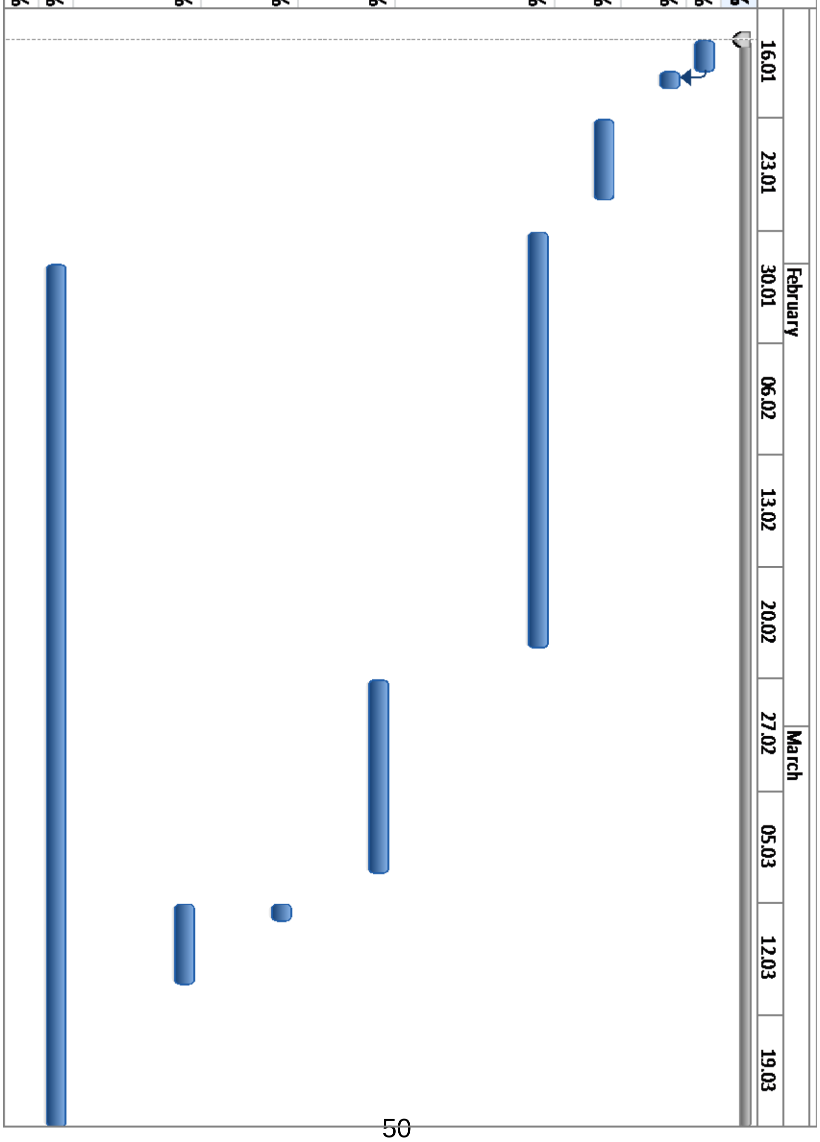
Signatures:

Student (date and signature): 23.02.2012,

Supervisor (date and signature): 23022012

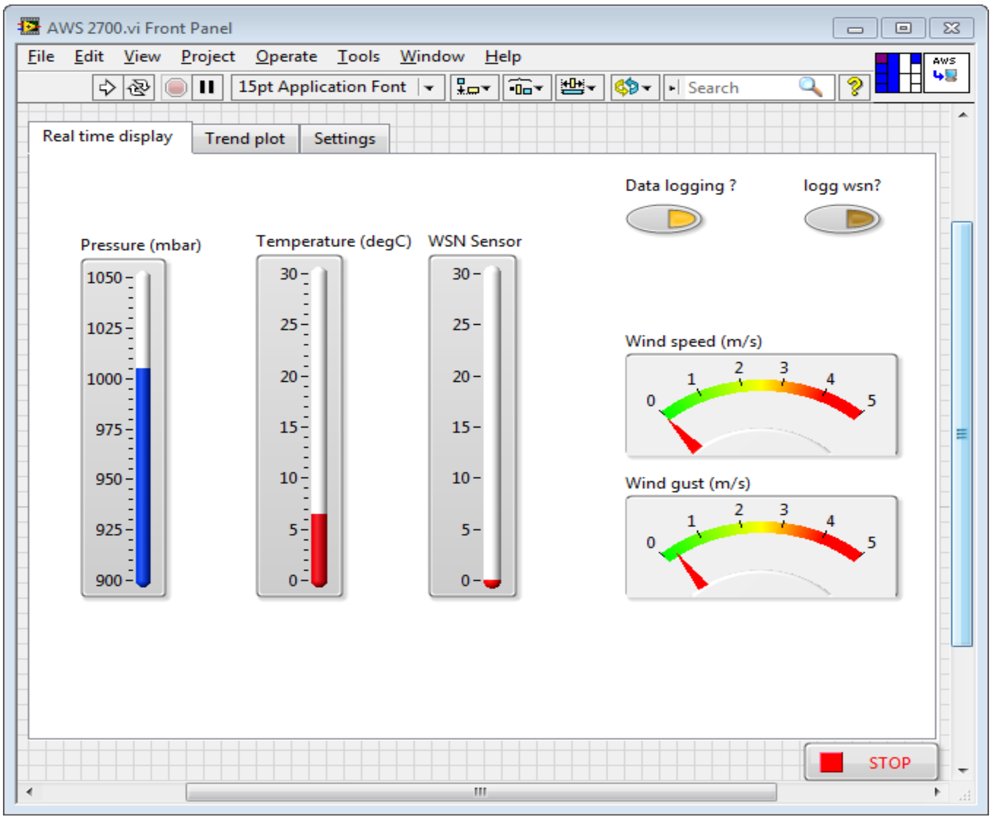
Appendix B Gant chart

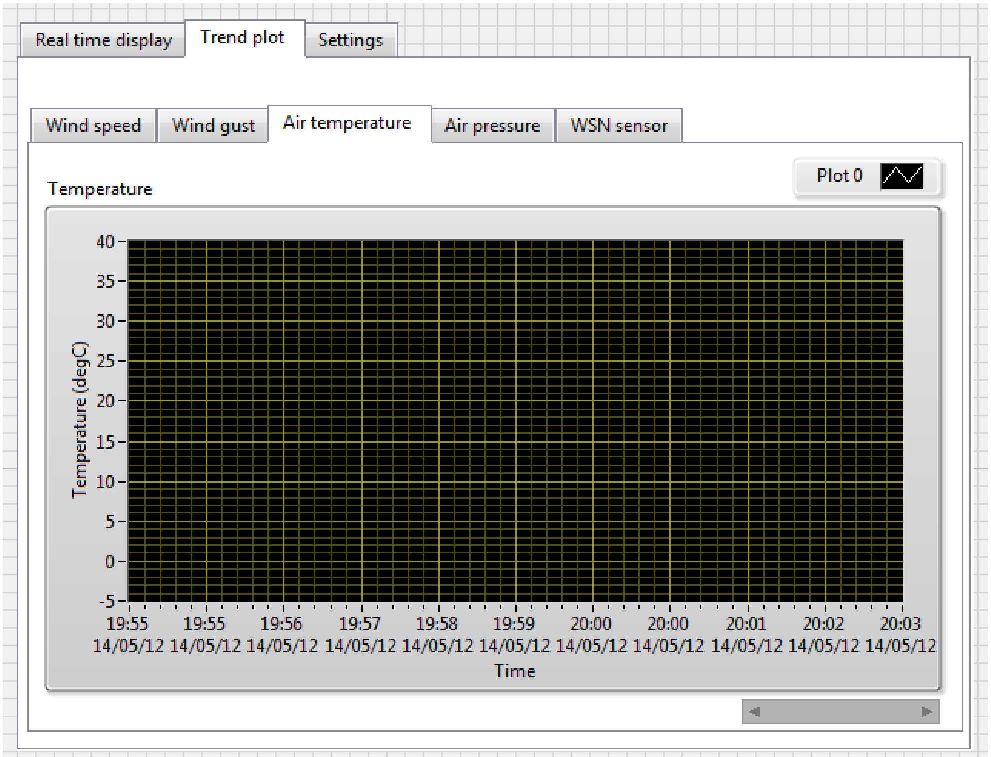
ID	Task Name	Duration	Start	Finish	% Work Complete
0	**Master thesis project**	**97 days**	**Wed 18.01.12**	**Thu 31.05.1**	**100%**
1	Planning	2 days	Wed 18.01.12	Thu 19.01.12	100%
2	Reading AWS documents	1 day	Fri 20.01.12	Fri 20.01.12	100%
3	Configuration and running up AWS	5 days	Mon 23.01.12	Fri 27.01.12	100%
4	Intergrating AADI and NI's LabVIEW for datalogging. Making a statemachine in LabVIEW	20 days	Mon 30.01.12	Fri 24.02.12	100%
5	Making a LabVIEW program for data analysis.	10 days	Mon 27.02.12	Fri 09.03.12	100%
6	Presentation of logged data in LabVIEW	1 day?	Mon 12.03.12	Mon 12.03.12	100%
7	Integration of WSN in current AWS. Making a LabVIEW program for WSN.	5 days	Mon 12.03.12	Fri 16.03.12	100%
8	Report	75 days	Wed 01.02.12	Tue 15.05.12	100%
9	Review of report	13 days?	Tue 15.05.12	Thu 31.05.12	100%

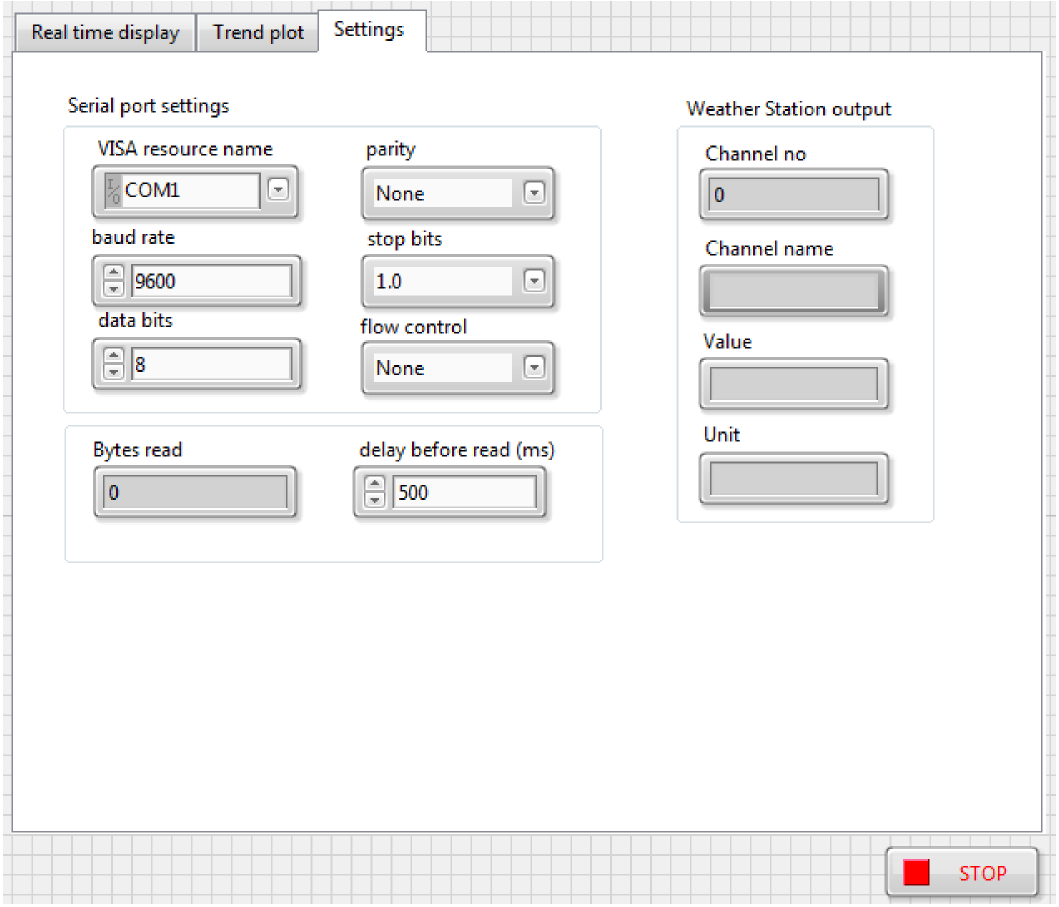


Appendix C LabVIEW code

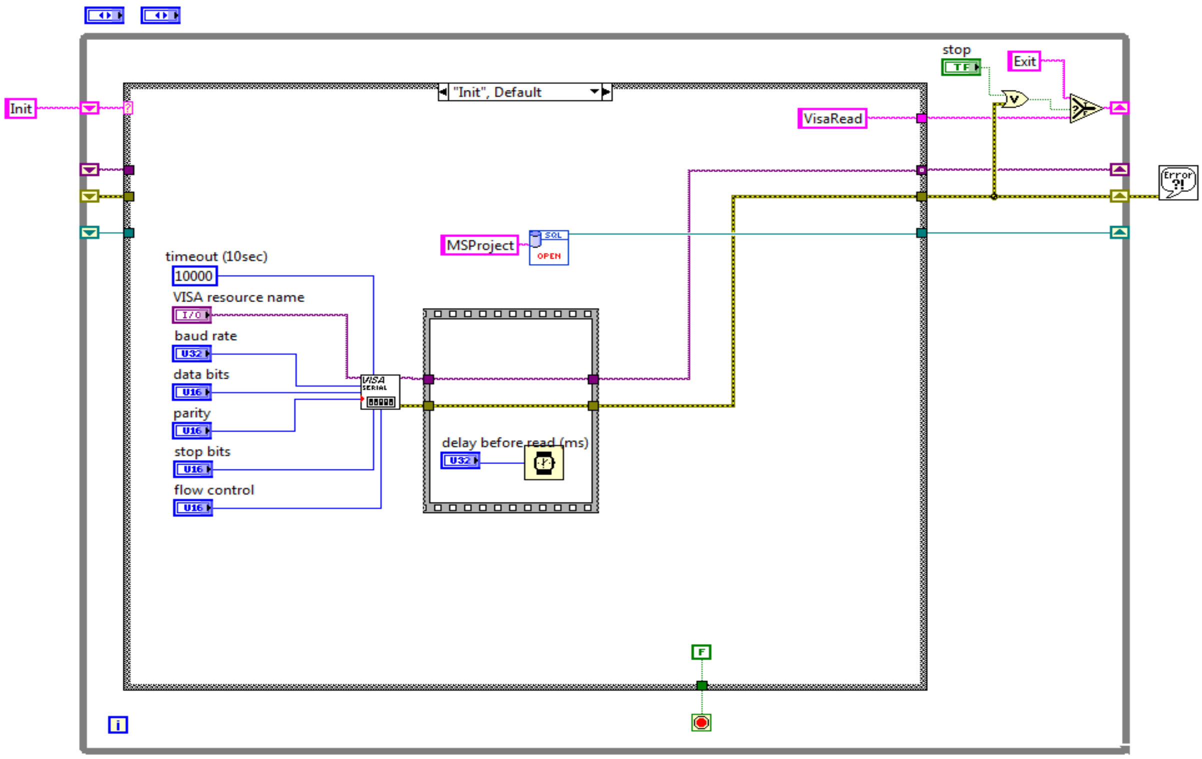
Front panel of AWS 2700.vi

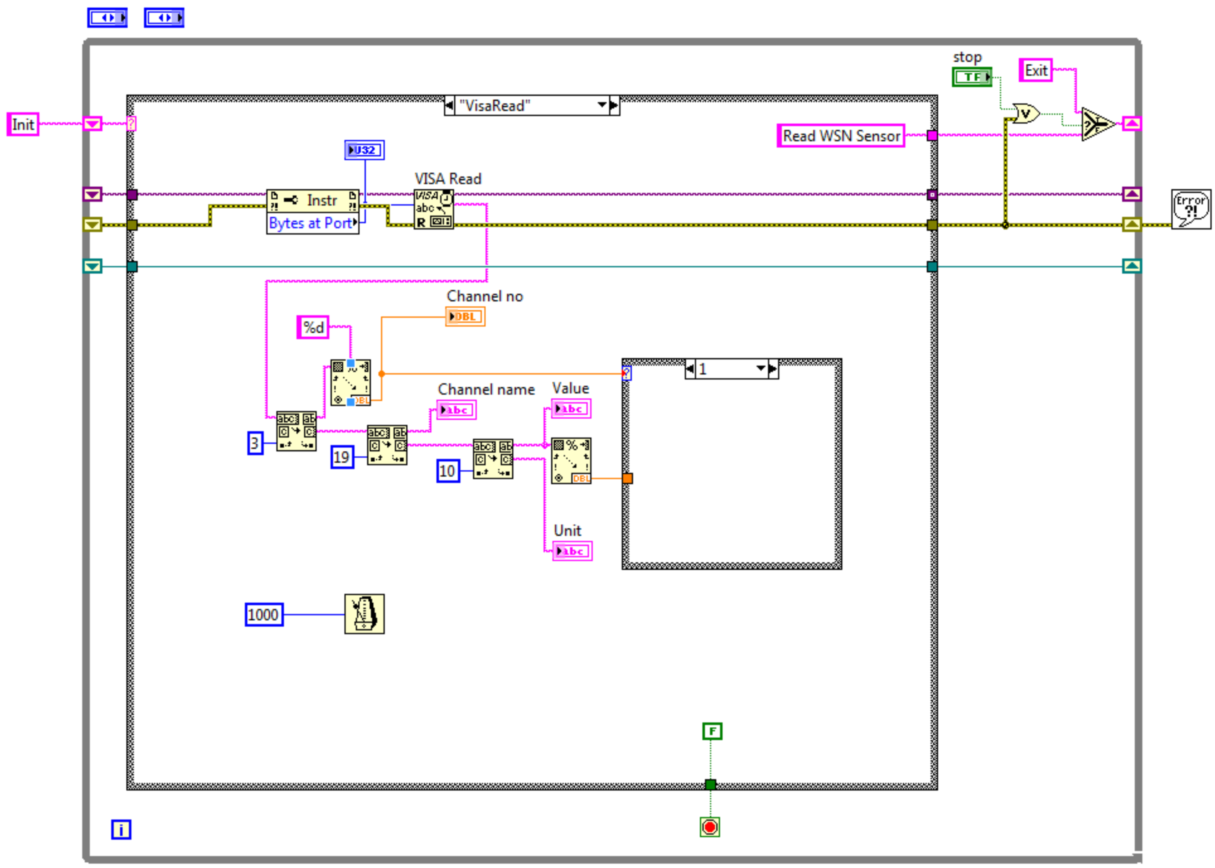


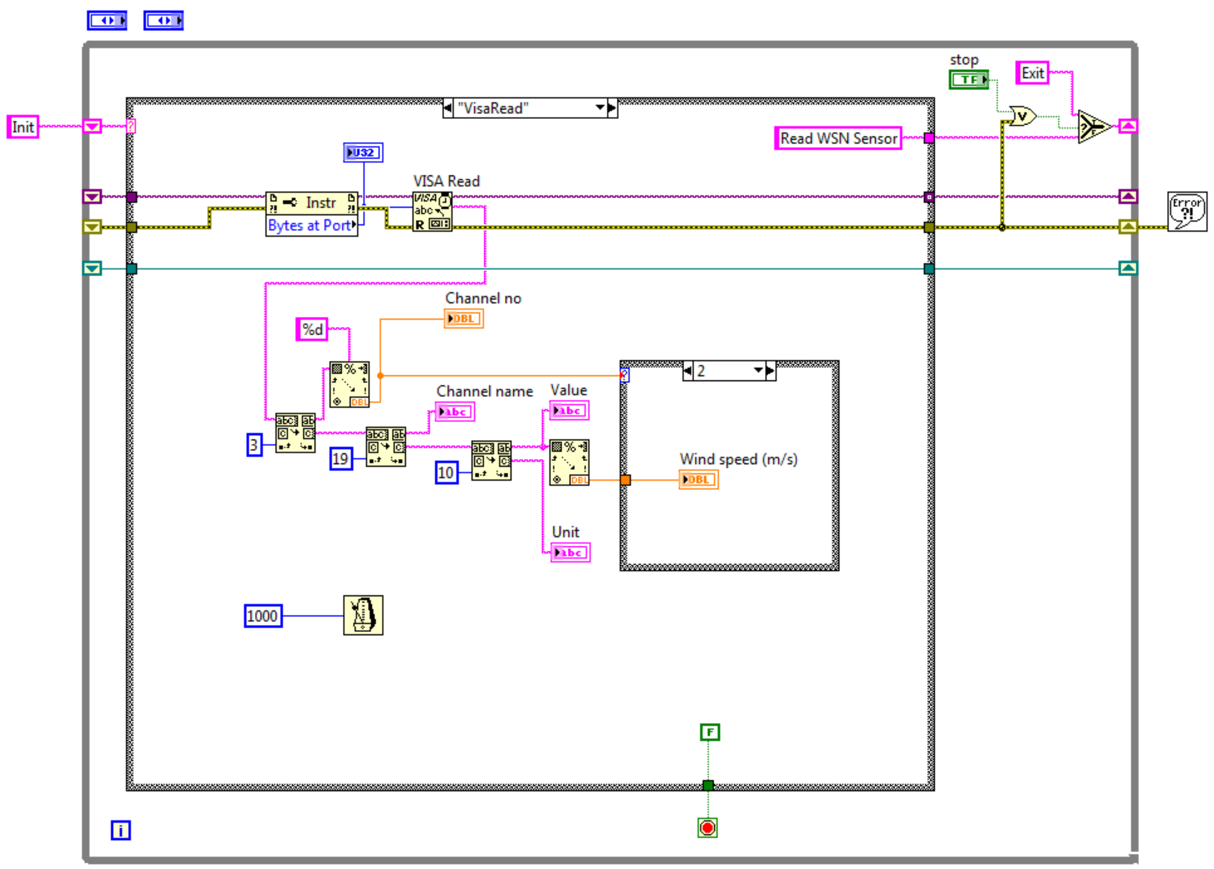


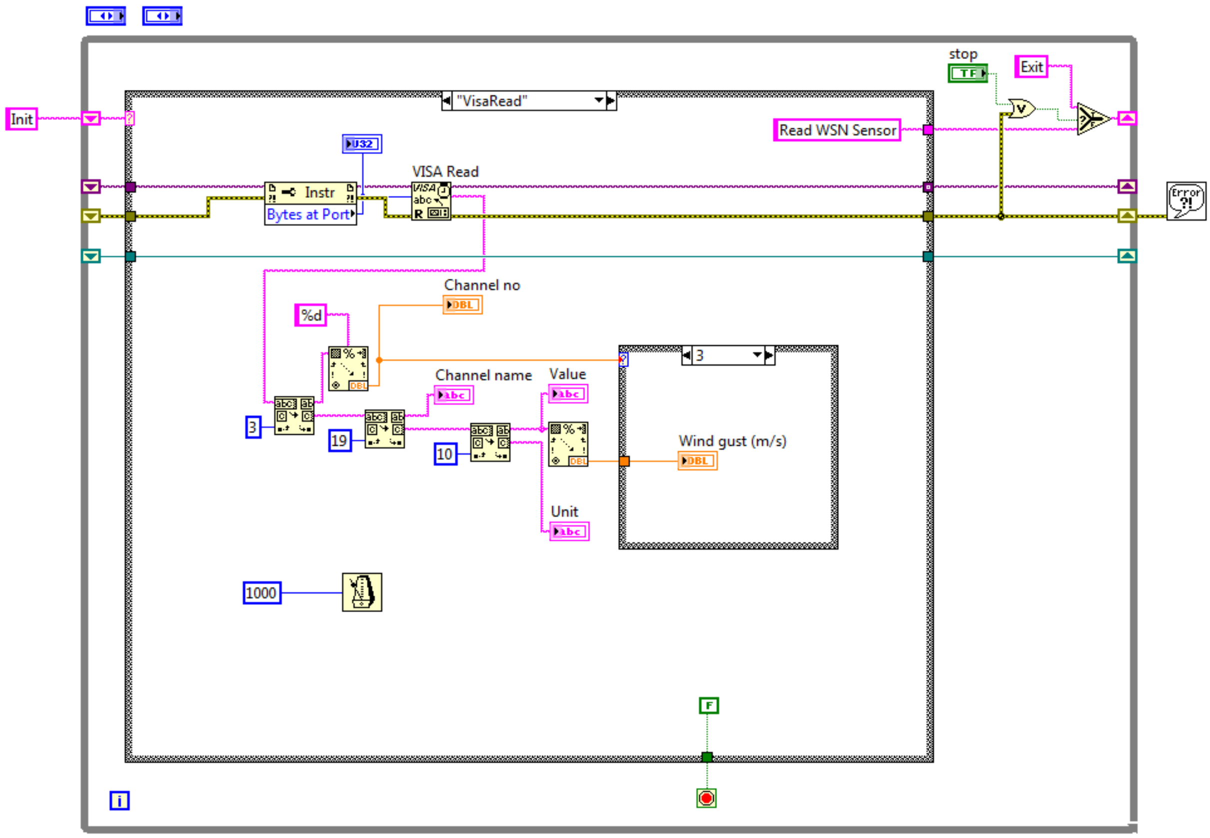


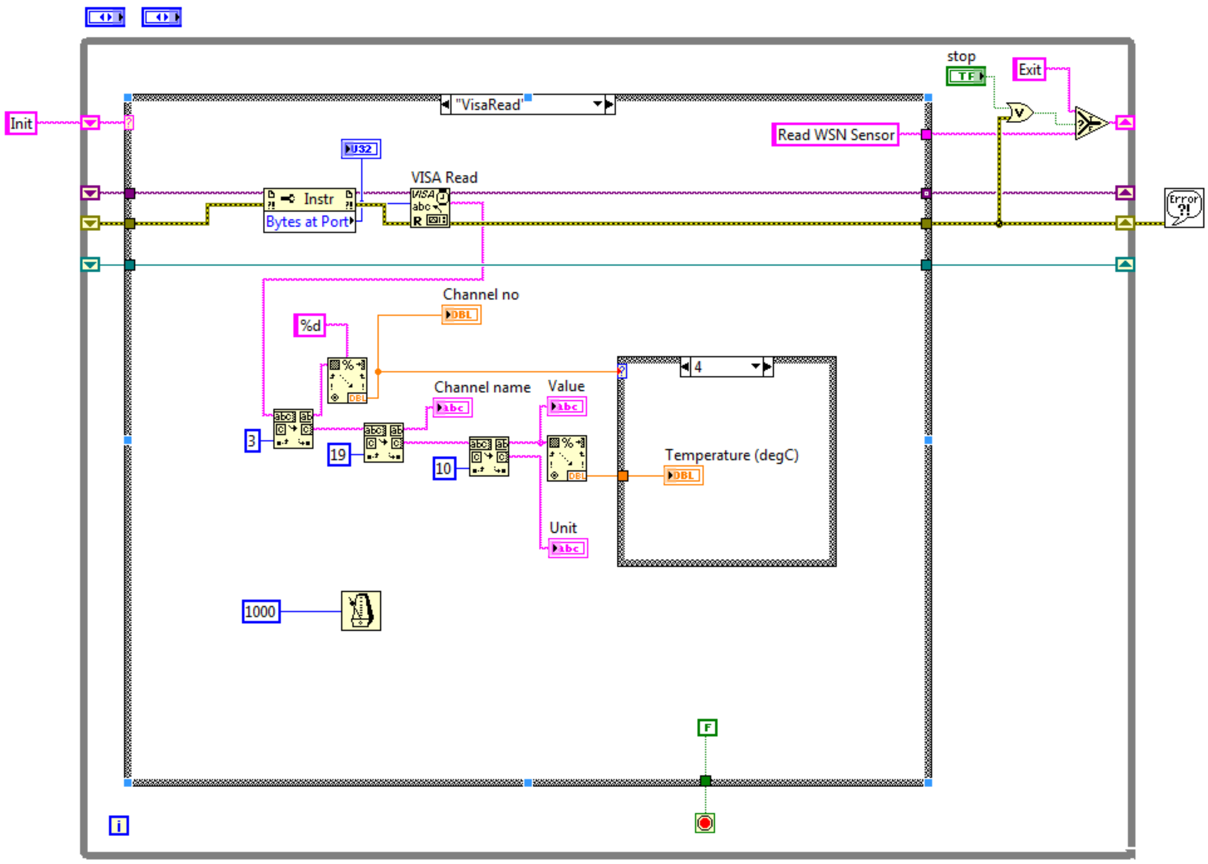
Block diagram of AWS 2700.vi

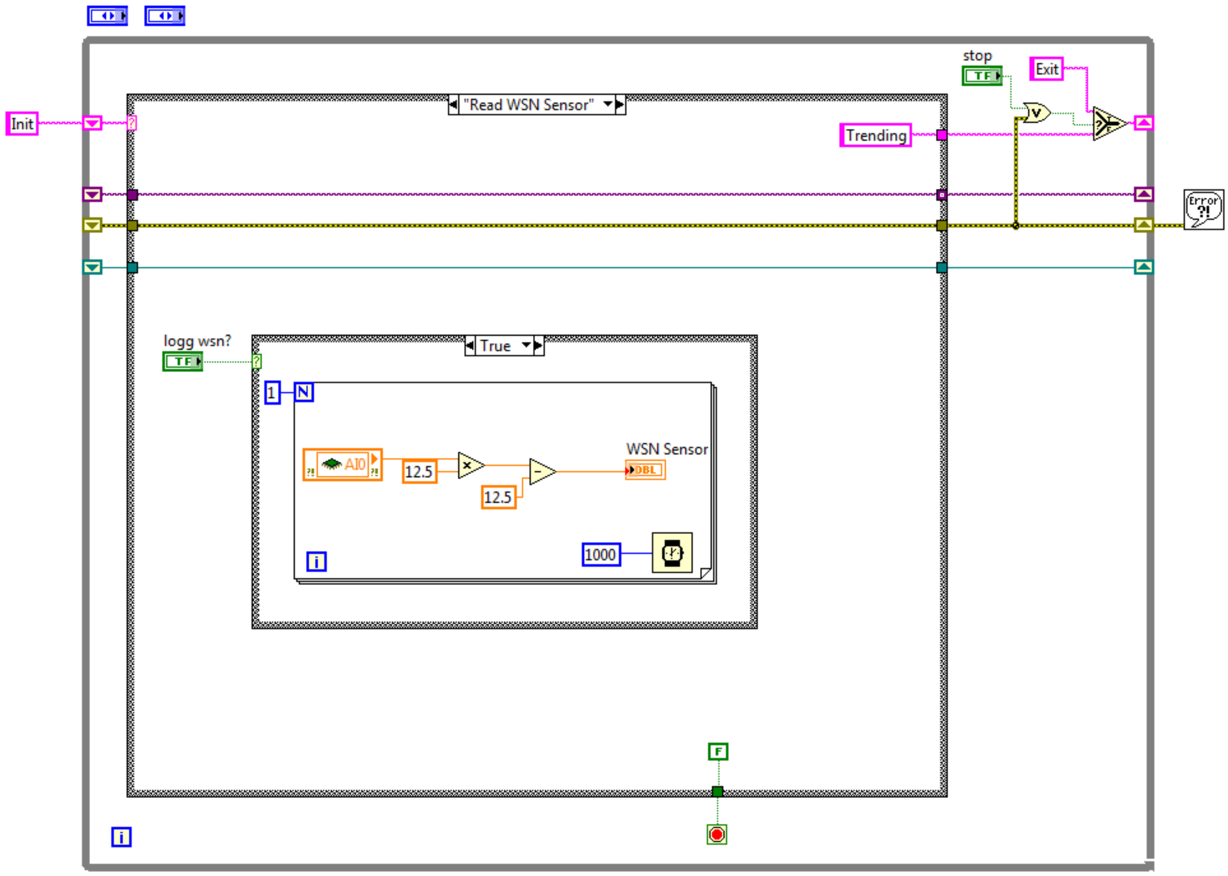



"VisaRead"
Init
stop
Exit
Read WSN Sensor
U32
VISA Read
Instr
Bytes at Port
Channel no
%d
Channel name
Value
Unit
3
19
10
1
1000

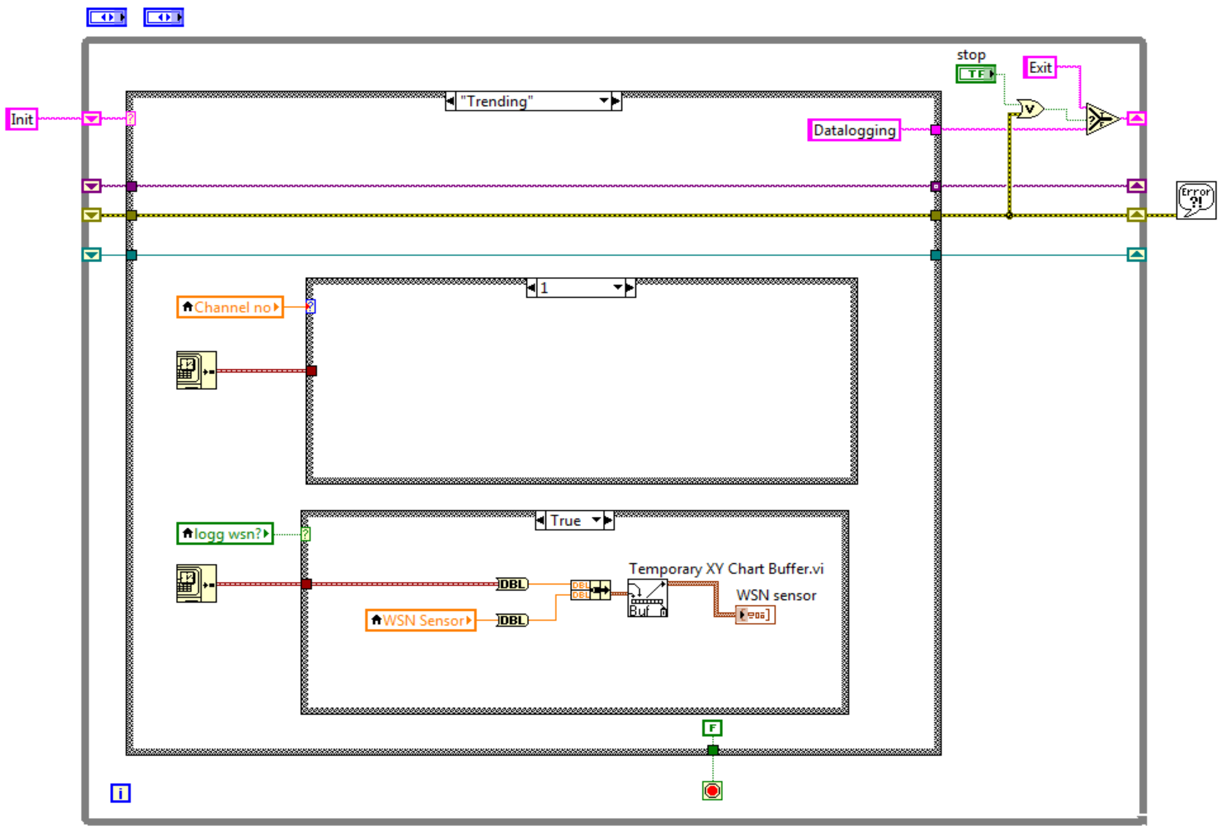

"VisaRead"
Init
stop
Exit
Read WSN Sensor
U32
VISA Read
Instr
Bytes at Port
Channel no
%d
Channel name
Value
Unit
3
19
10
2
Wind speed (m/s)
1000


"VisaRead"
Init
stop
Exit
Read WSN Sensor
U32
Instr
Bytes at Port
VISA Read
Channel no
%d
Channel name
Value
3
19
10
Wind gust (m/s)
Unit
1000
F

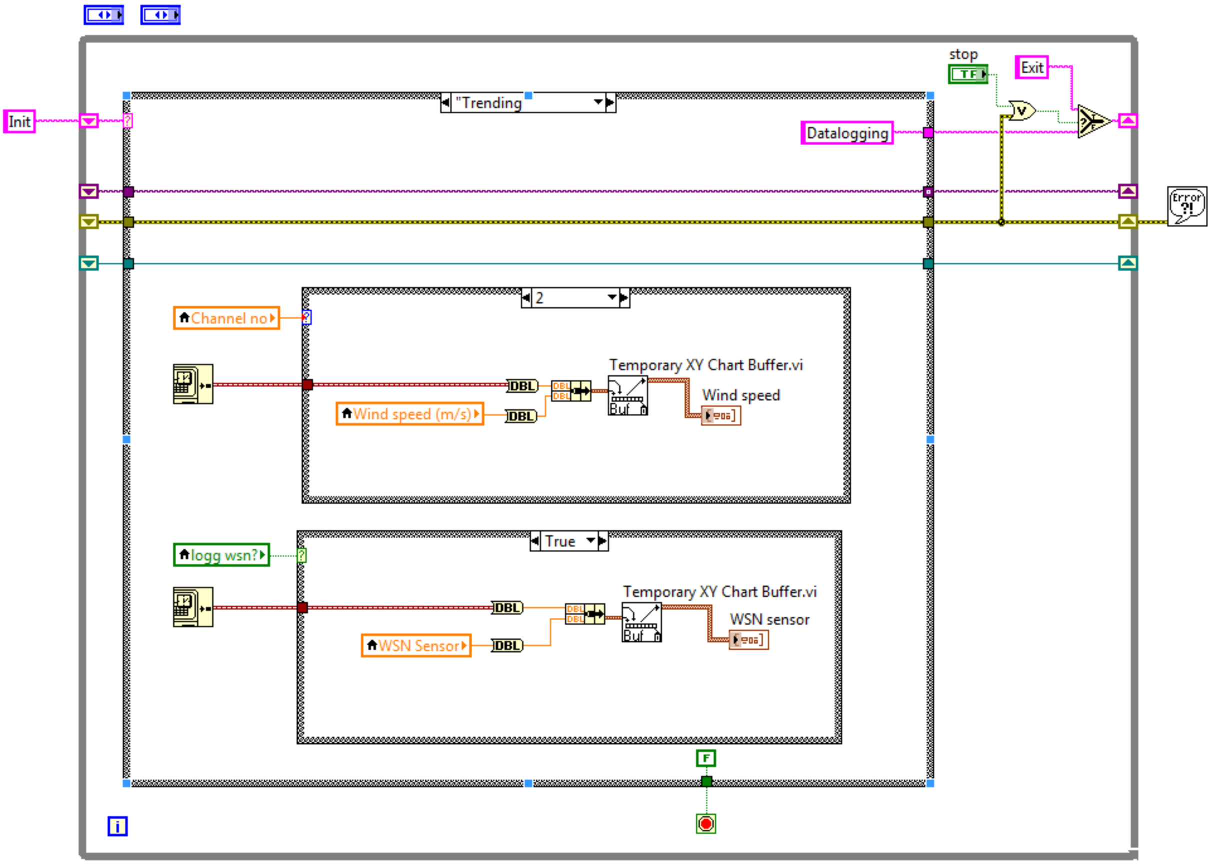

"VisaRead"
Init
stop
Exit
Read WSN Sensor
U32
Instr
Bytes at Port
VISA Read
Channel no
%d
Channel name
Value
4
3
19
10
Temperature (degC)
Unit
1000
F

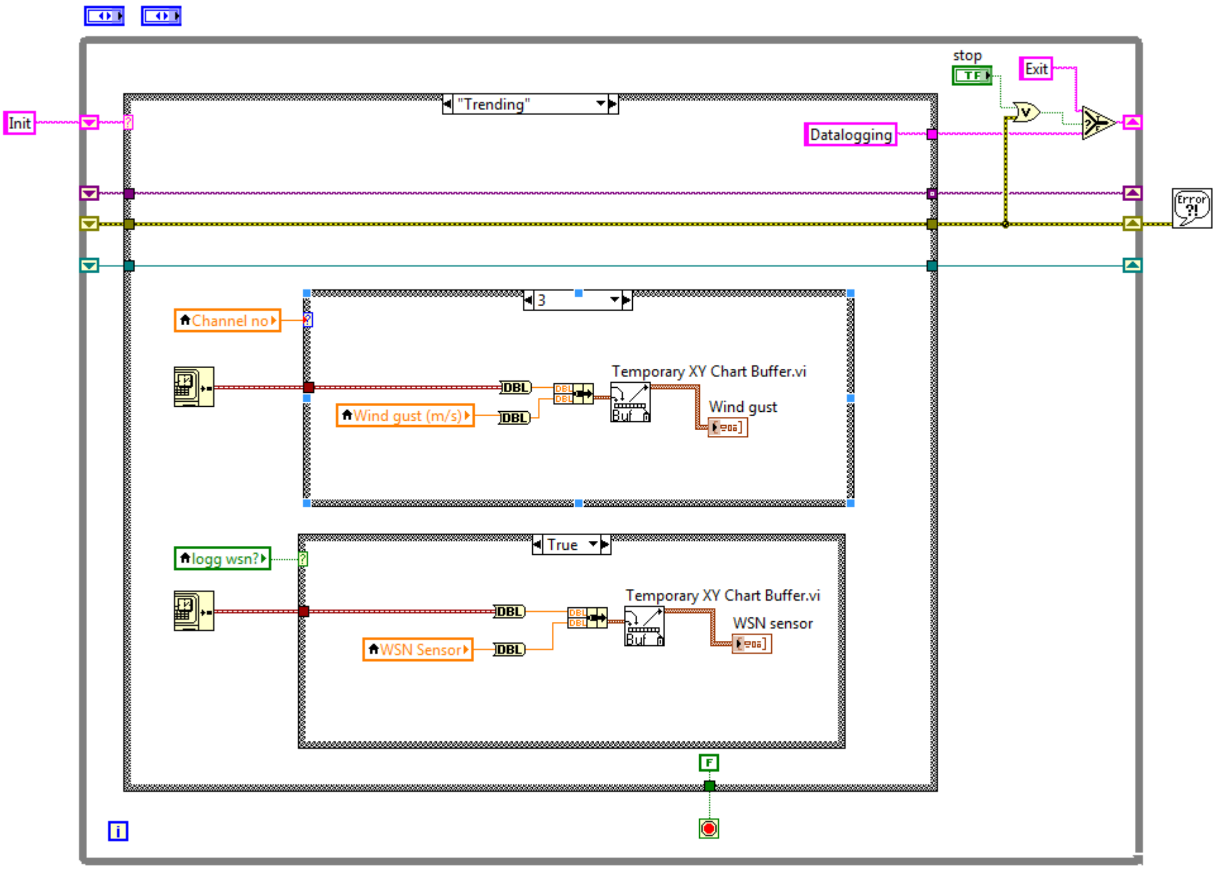


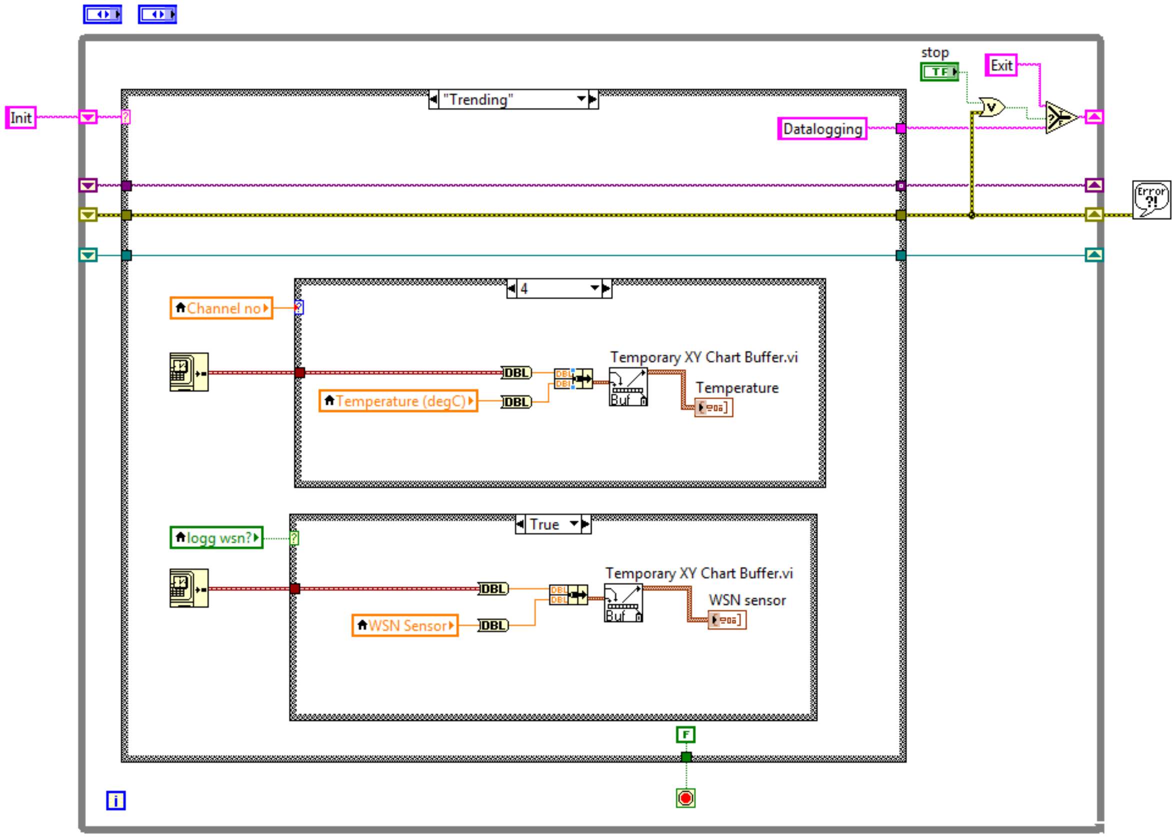
"Read WSN Sensor"
Init
stop
Exit
Trending
logg wsn?
True
1
N
AI0
12.5
12.5
WSN Sensor
1000
i
F
i



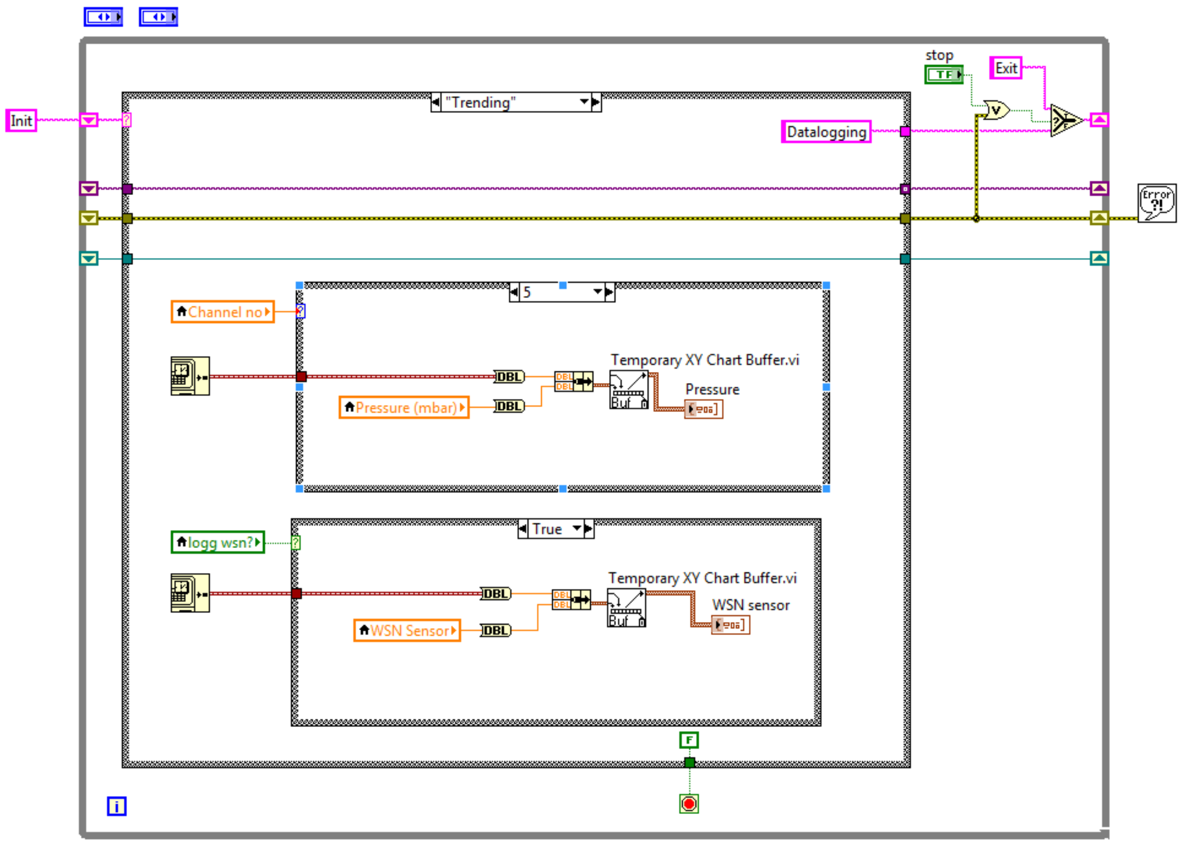
"Trending"
Init
stop
Exit
Datalogging
Channel no
1
logg wsn?
True
DBL
DBL
WSN Sensor
Temporary XY Chart Buffer.vi
WSN sensor
F
i


stop
Exit
Init
"Trending"
Datalogging
Error
Channel no
2
Temporary XY Chart Buffer.vi
Wind speed
Wind speed (m/s)
logg wsn?
True
Temporary XY Chart Buffer.vi
WSN sensor
WSN Sensor

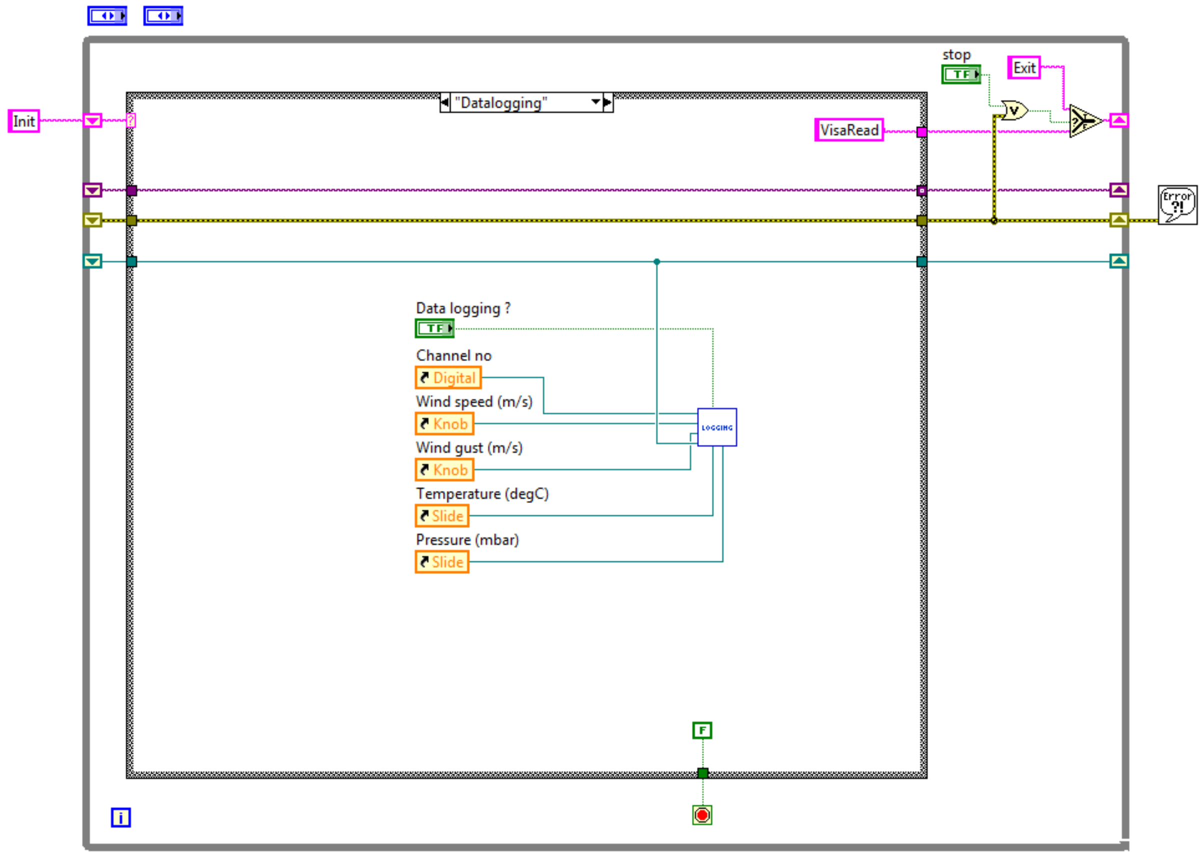

stop
Exit
Init
"Trending"
Datalogging
Error
Channel no
3
Temporary XY Chart Buffer.vi
Wind gust
Wind gust (m/s)
logg wsn?
True
Temporary XY Chart Buffer.vi
WSN sensor
WSN Sensor

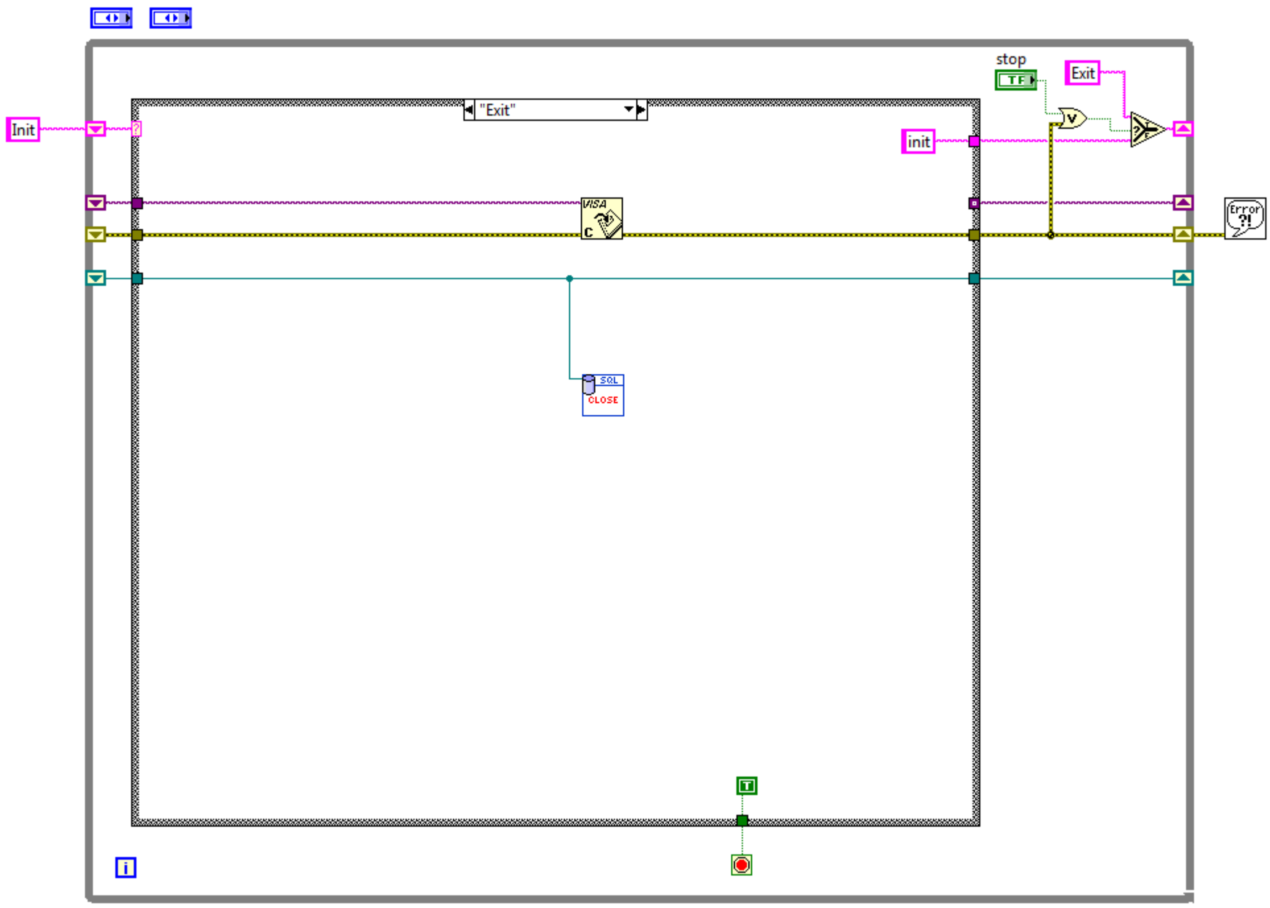


Init
"Trending"
stop
Exit
Datalogging
Channel no
4
Temporary XY Chart Buffer.vi
Temperature
Temperature (degC)
logg wsn?
True
Temporary XY Chart Buffer.vi
WSN sensor
WSN Sensor



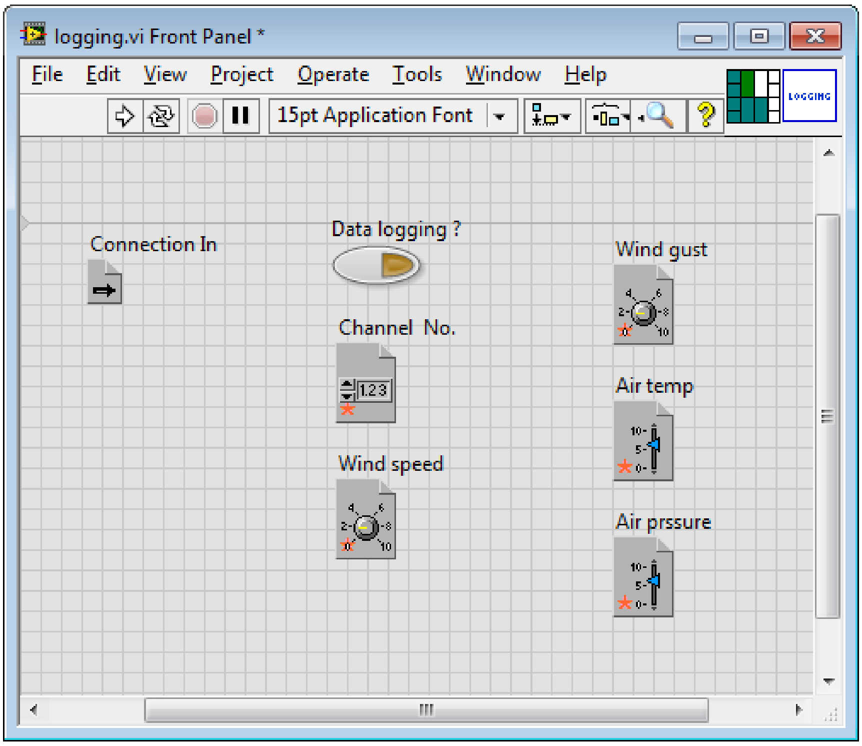
Init
"Trending"
stop
Exit
Datalogging
Channel no
5
Temporary XY Chart Buffer.vi
Pressure
Pressure (mbar)
logg wsn?
True
Temporary XY Chart Buffer.vi
WSN sensor
WSN Sensor


Init
"Datalogging"
stop
Exit
VisaRead
Data logging ?
Channel no
Digital
Wind speed (m/s)
Knob
Wind gust (m/s)
Knob
Temperature (degC)
Slide
Pressure (mbar)
Slide
LOGGING

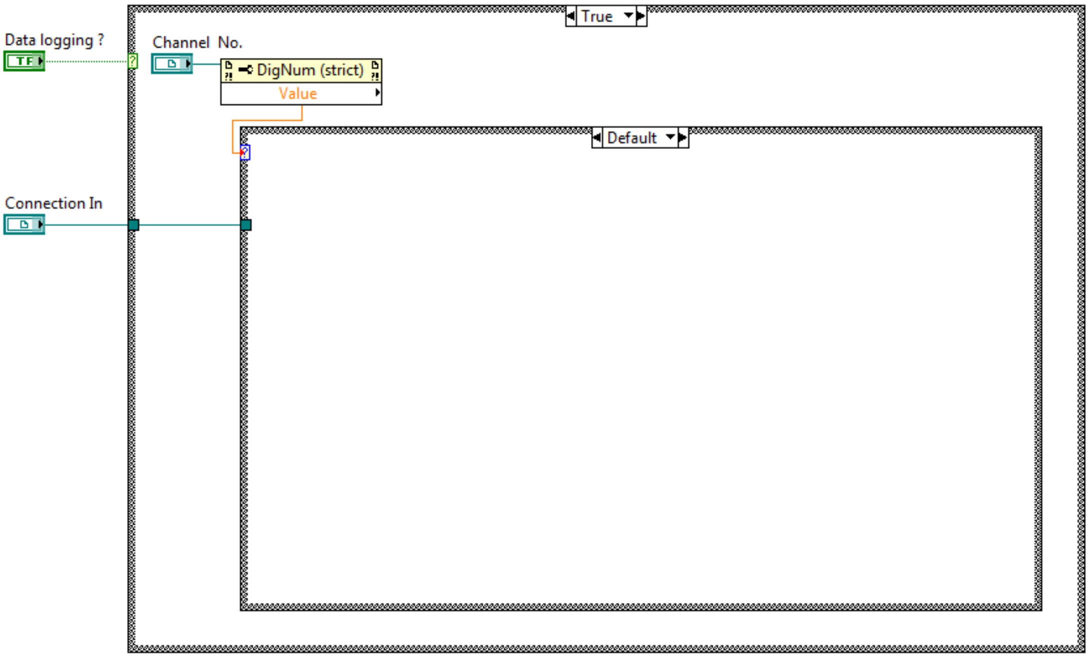

Init
"Exit"
stop
Exit
init
VISA
SQL
CLOSE

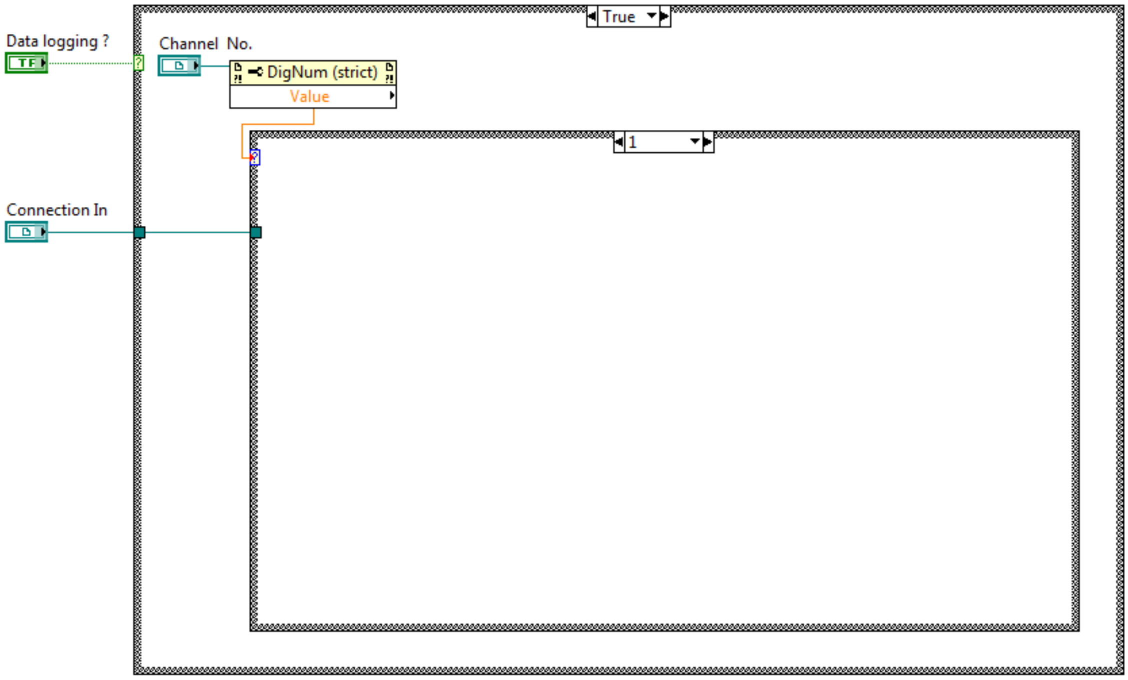
LabVIEW code for sub VI logging.vi

Front Panel

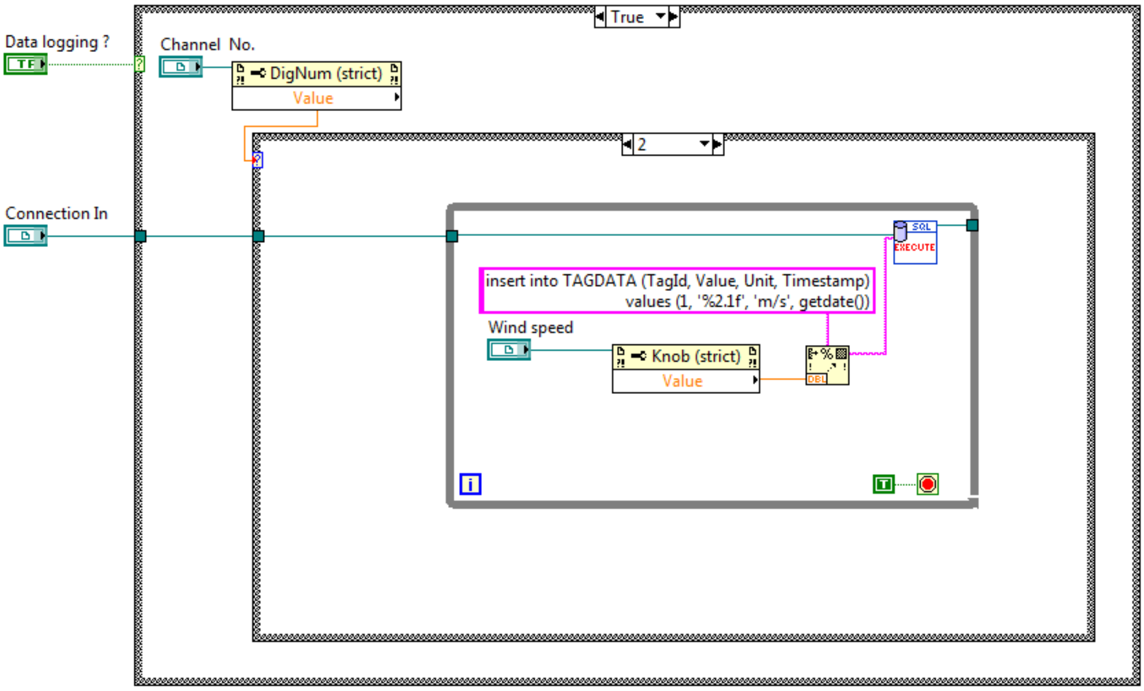


Block diagram

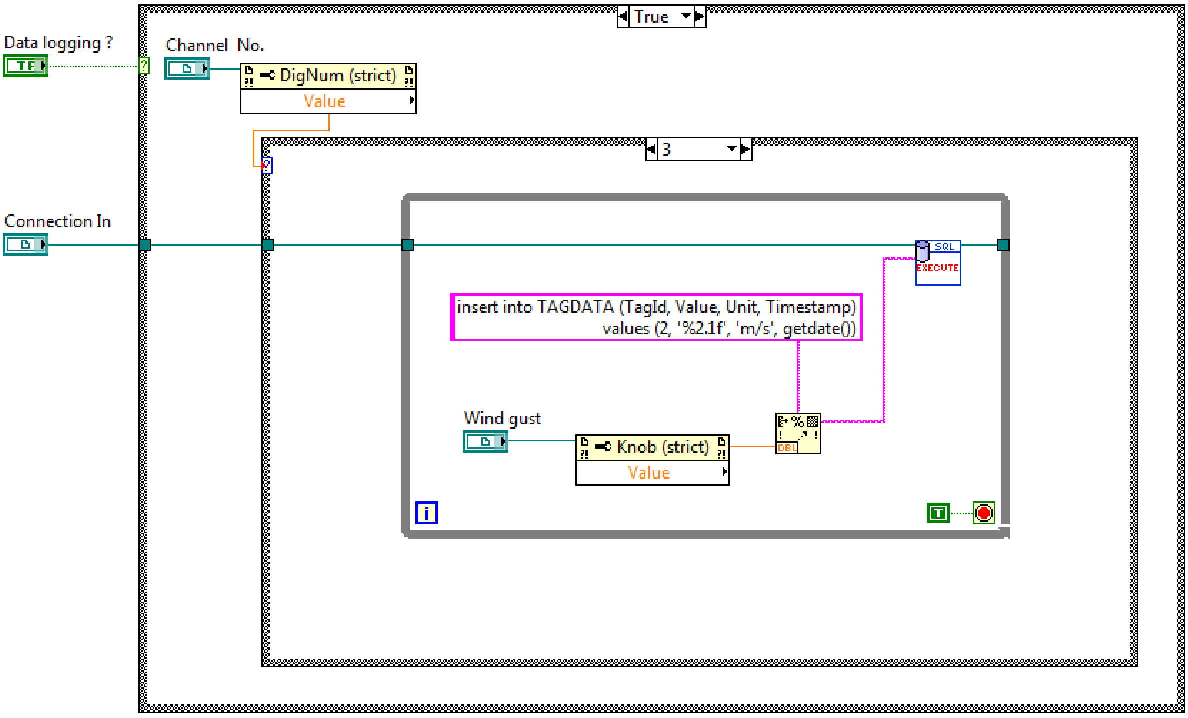


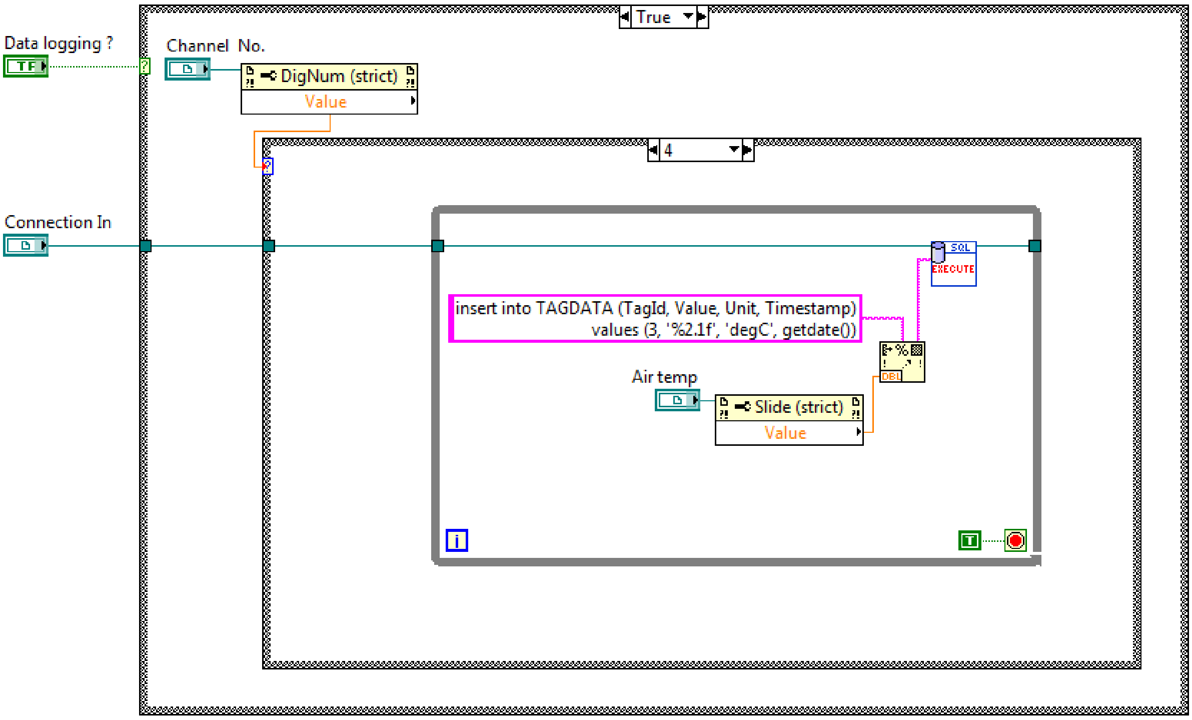


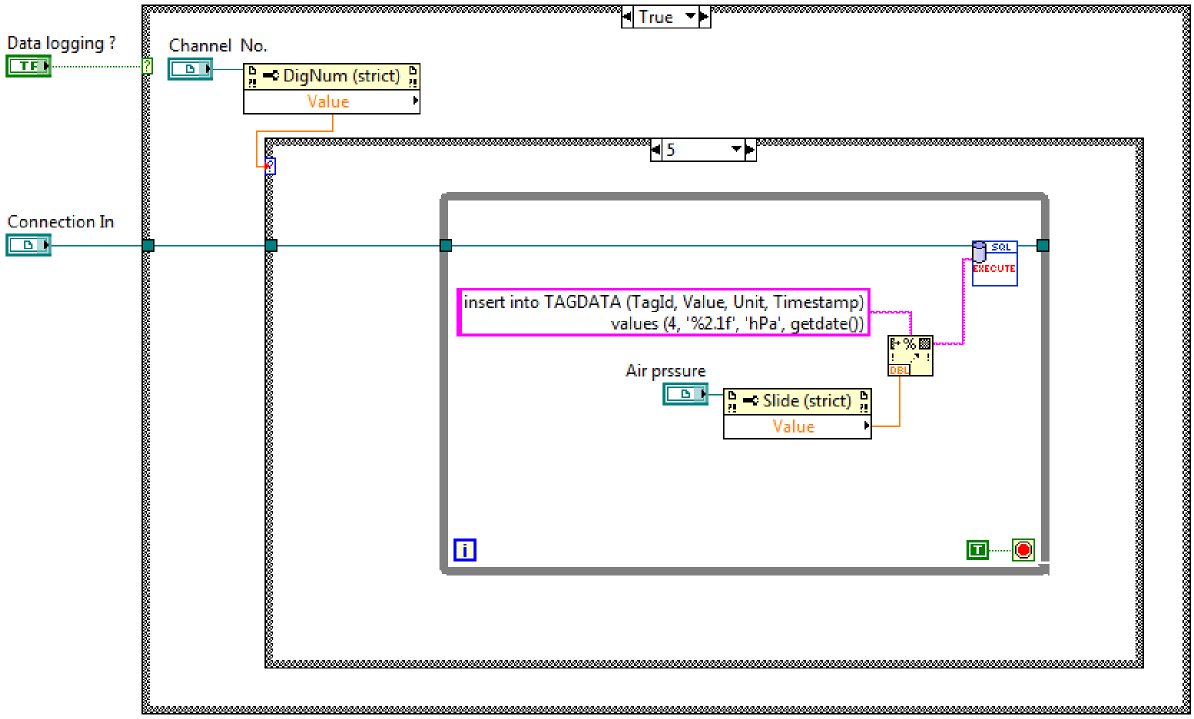
True
Data logging ?
Channel No.
DigNum (strict)
Value
1
Connection In

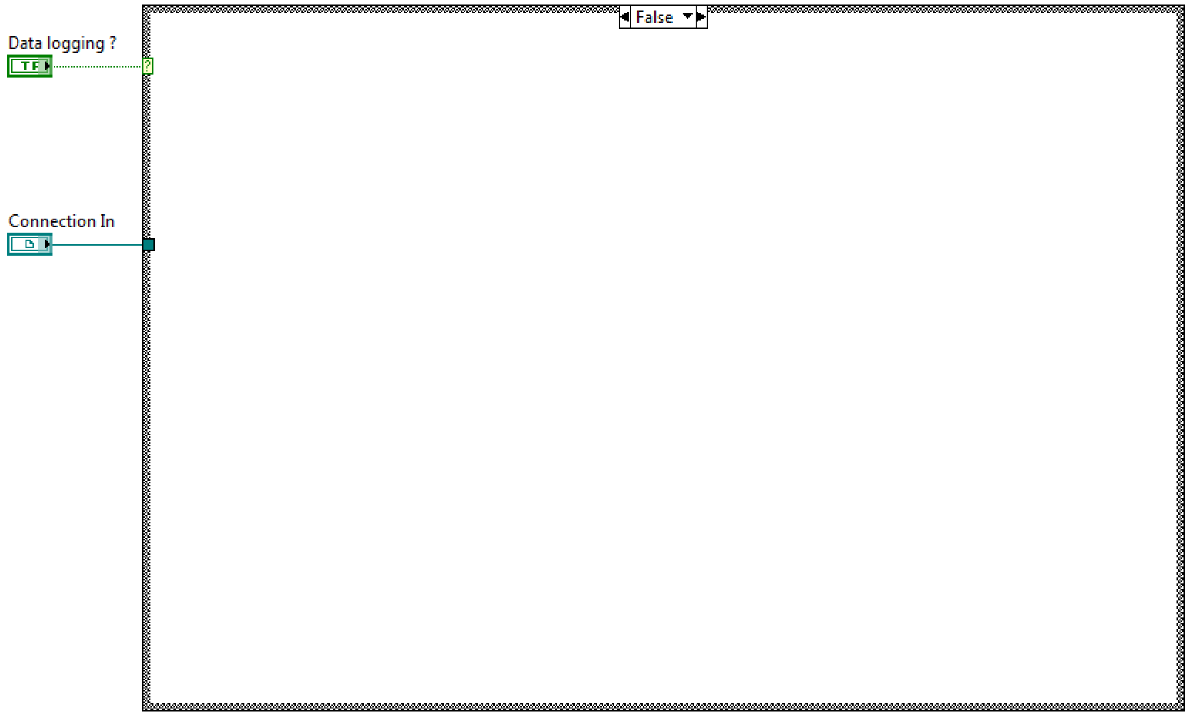


True
Data logging ?
Channel No.
DigNum (strict)
Value
2
Connection In
insert into TAGDATA (TagId, Value, Unit, Timestamp)
values (1, '%2.1f', 'm/s', getdate())
Wind speed
Knob (strict)
Value
SQL
EXECUTE

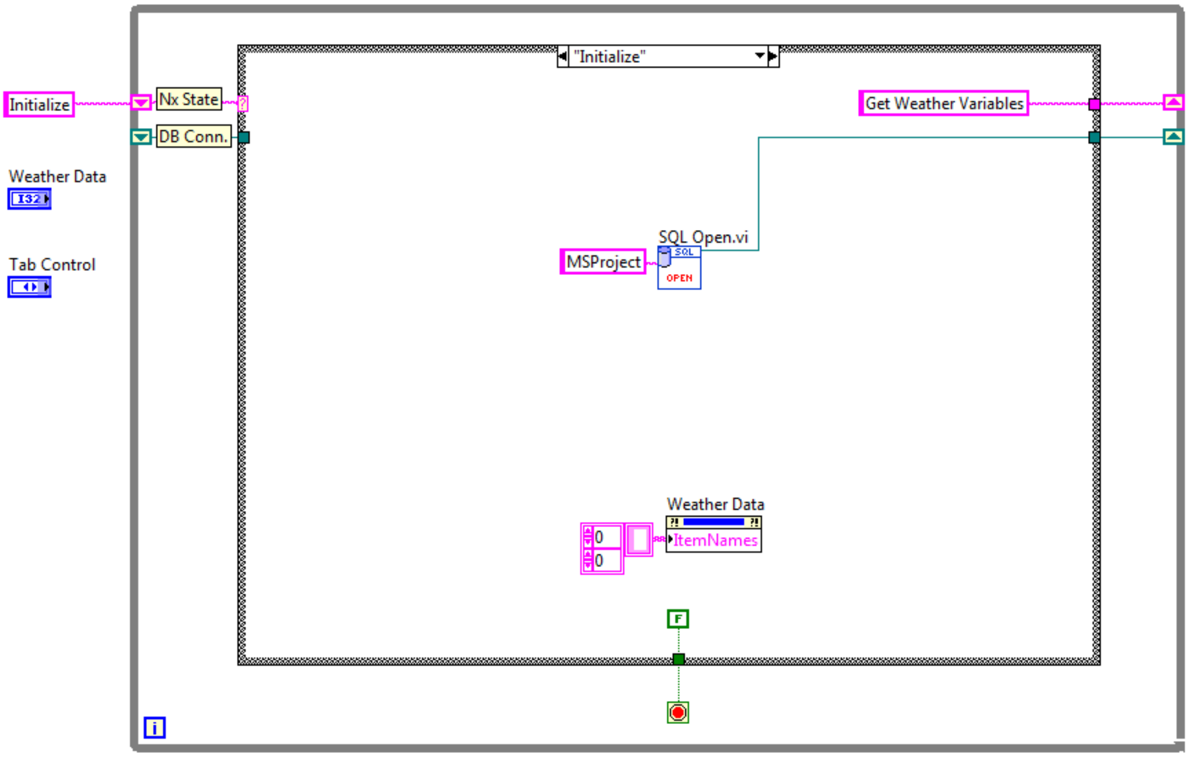

True
Data logging ?
Channel No.
DigNum (strict)
Value
3
Connection In
SQL
EXECUTE
insert into TAGDATA (TagId, Value, Unit, Timestamp)
values (2, '%2.1f', 'm/s', getdate())
Wind gust
Knob (strict)
Value
DBL

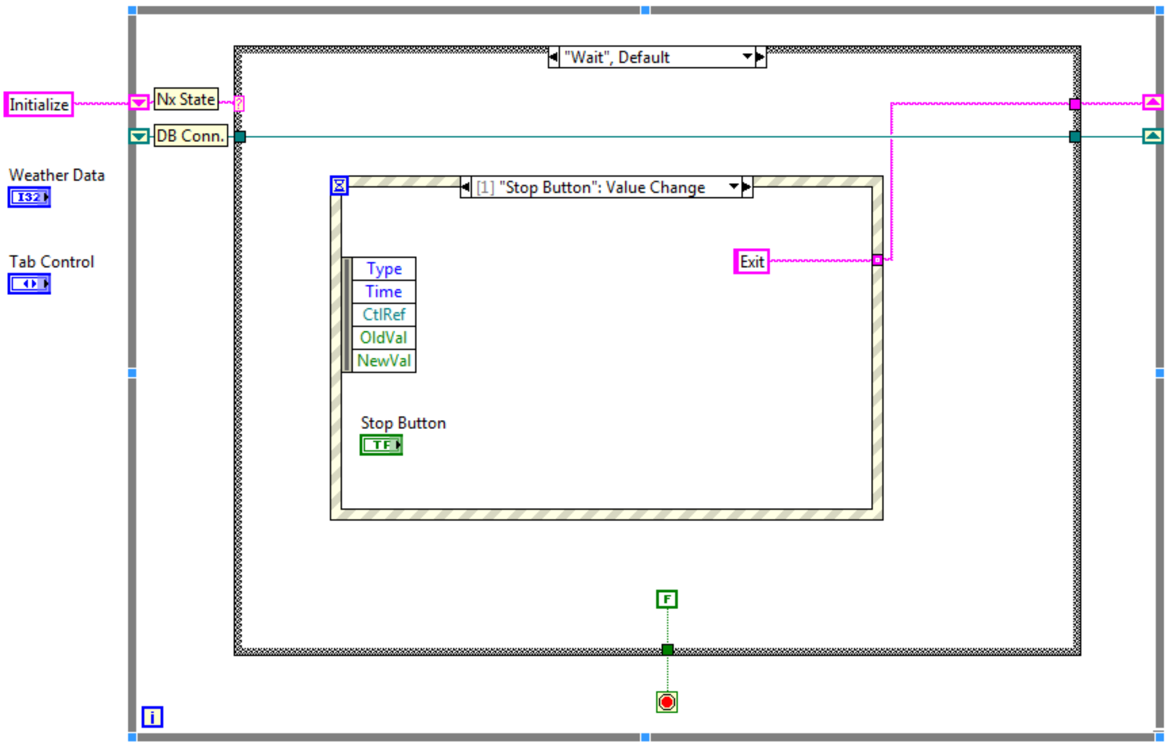

True
Data logging ?
Channel No.
DigNum (strict)
Value
4
Connection In
SQL
EXECUTE
insert into TAGDATA (TagId, Value, Unit, Timestamp)
values (3, '%2.1f', 'degC', getdate())
Air temp
Slide (strict)
Value
DBL

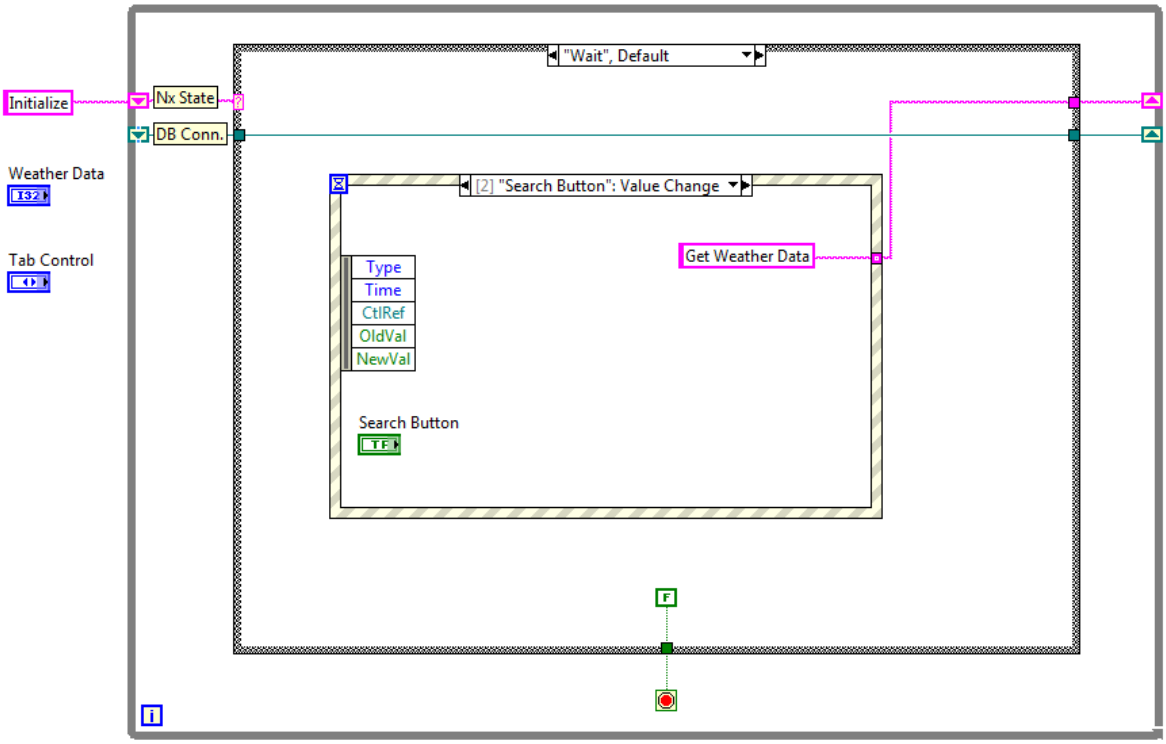

Data logging ?
True
Channel No.
DigNum (strict)
Value
5
Connection In
SQL
EXECUTE
insert into TAGDATA (TagId, Value, Unit, Timestamp)
values (4, '%2.1f', 'hPa', getdate())
Air prssure
Slide (strict)
Value

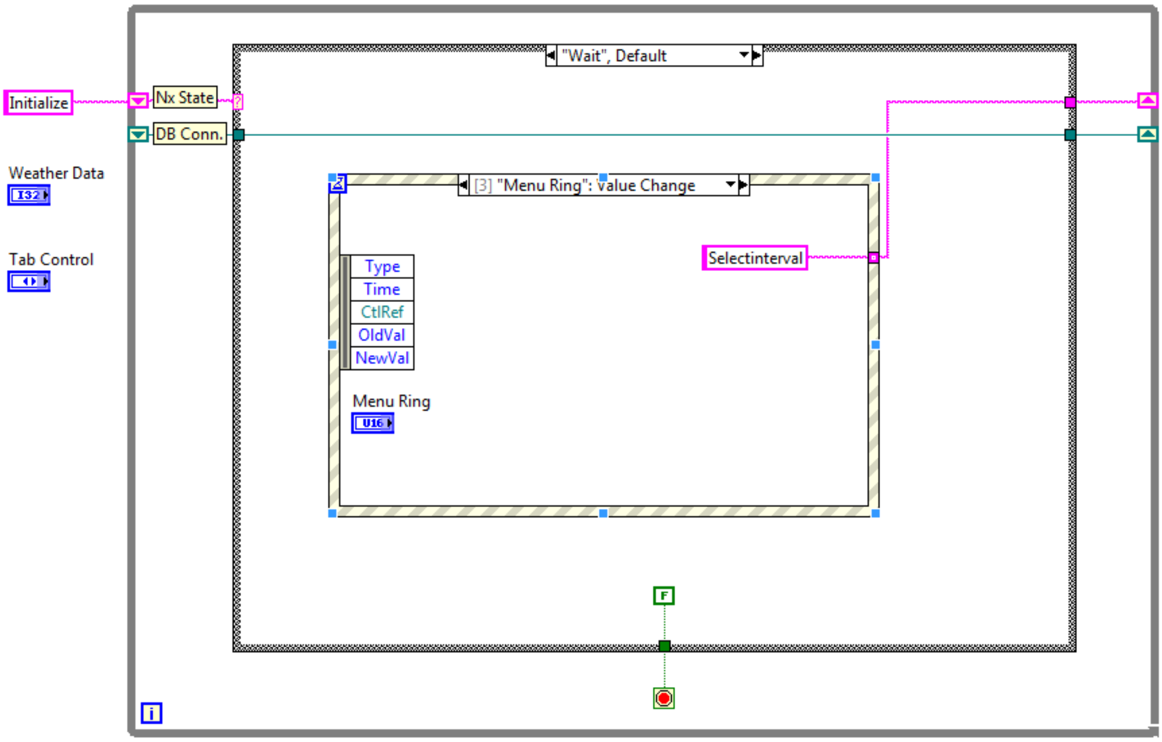

Data logging ?
False
Connection In

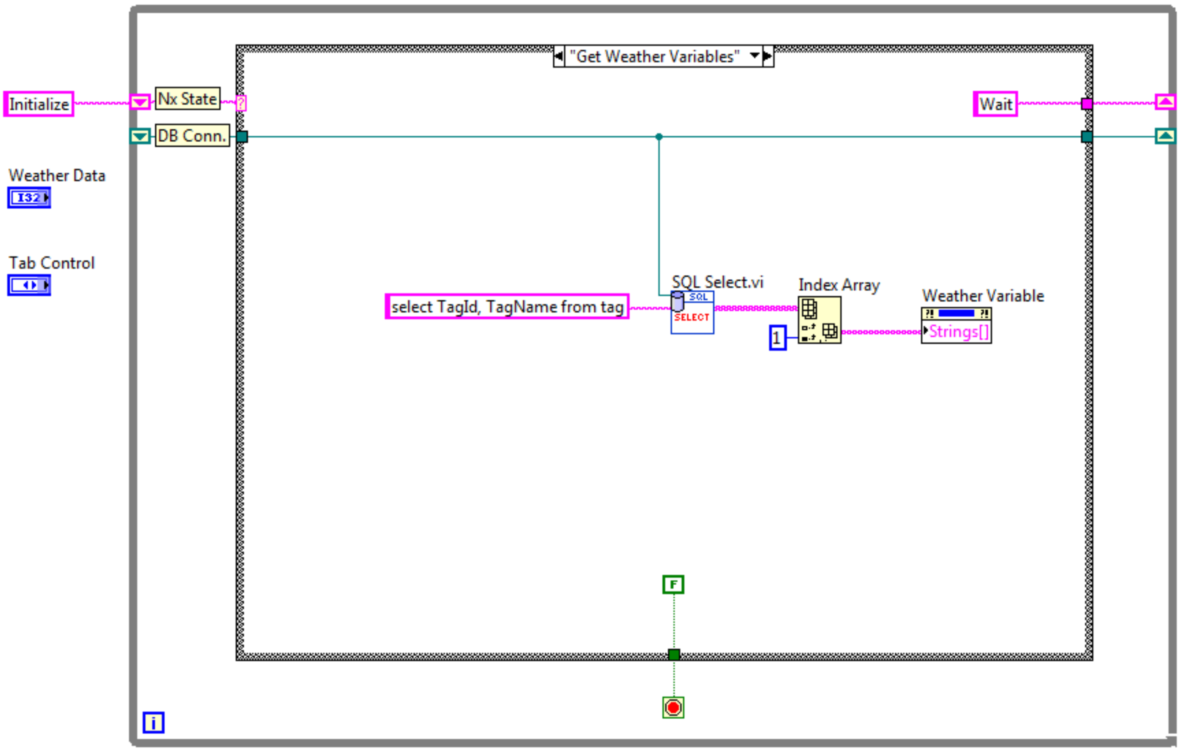
LabVIEW code for Historical data.vi

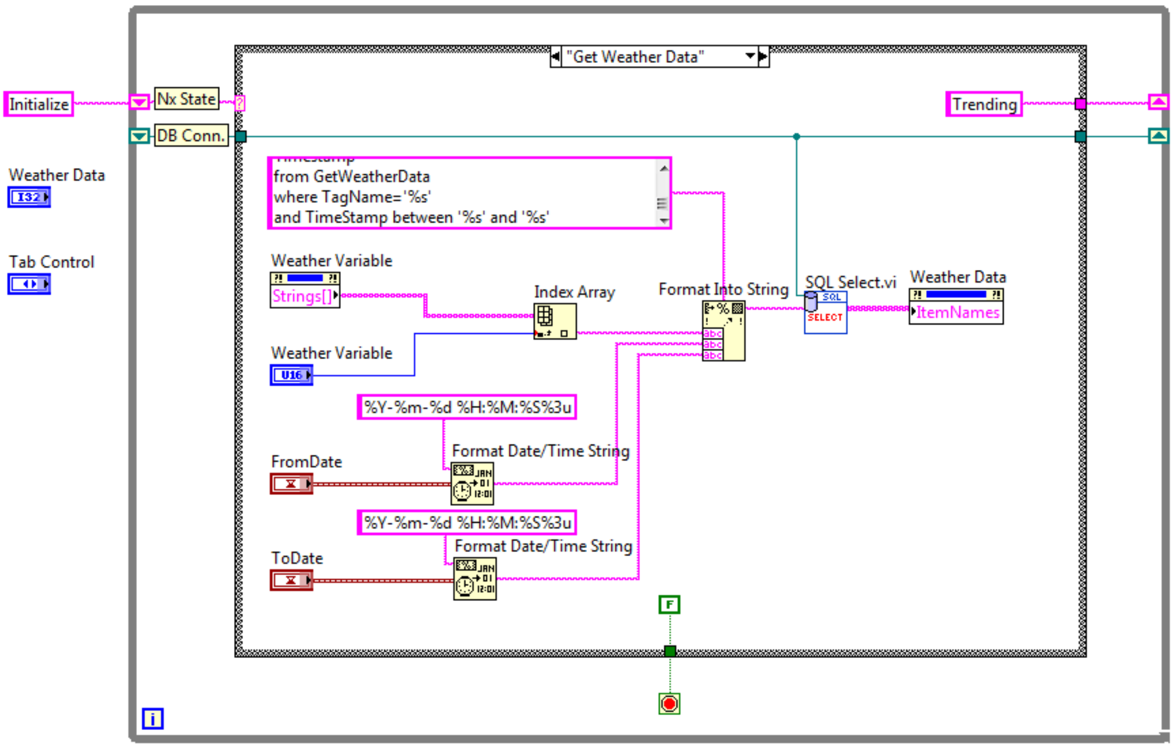


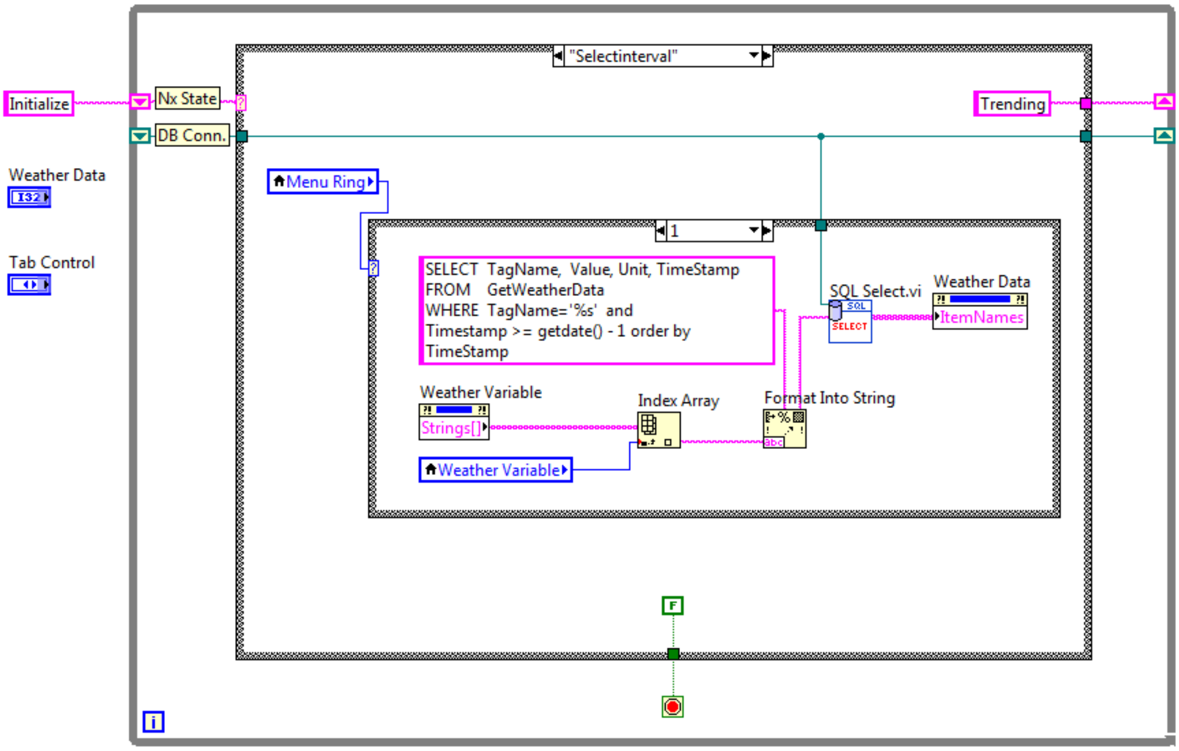


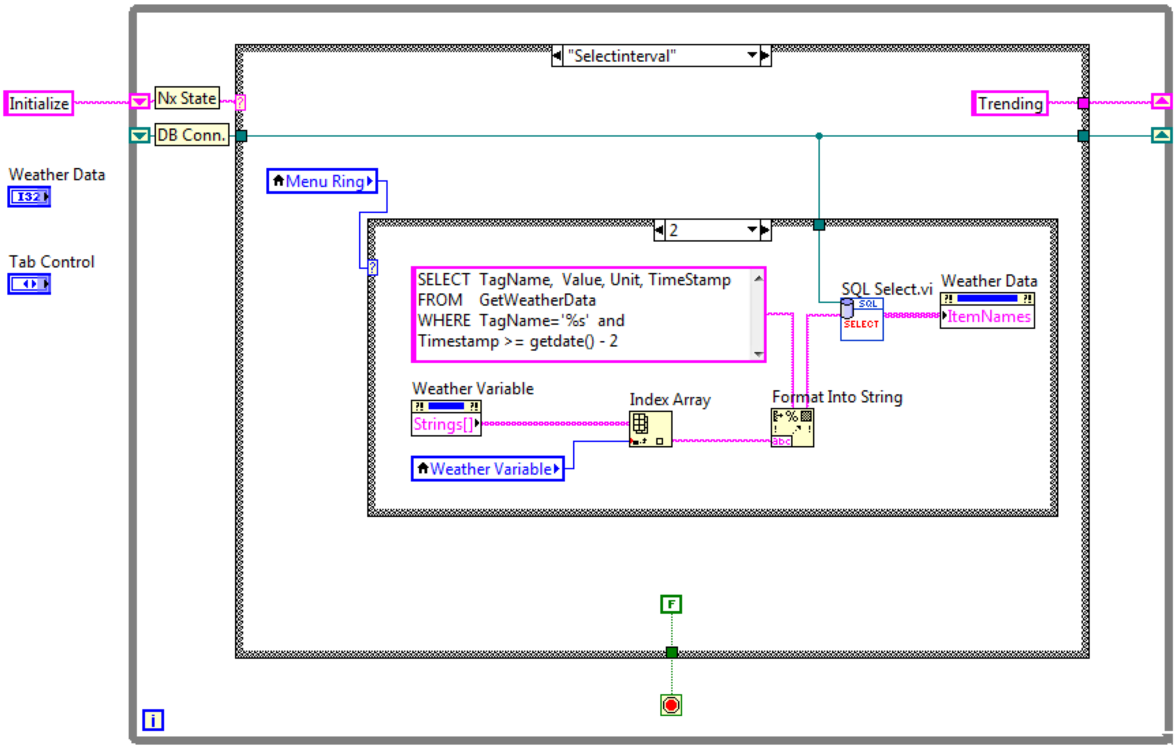

"Wait", Default
Initialize
Nx State
DB Conn.
Weather Data
I32
Tab Control
[2] "Search Button": Value Change
Type
Time
CtlRef
OldVal
NewVal
Get Weather Data
Search Button
TF
F
i

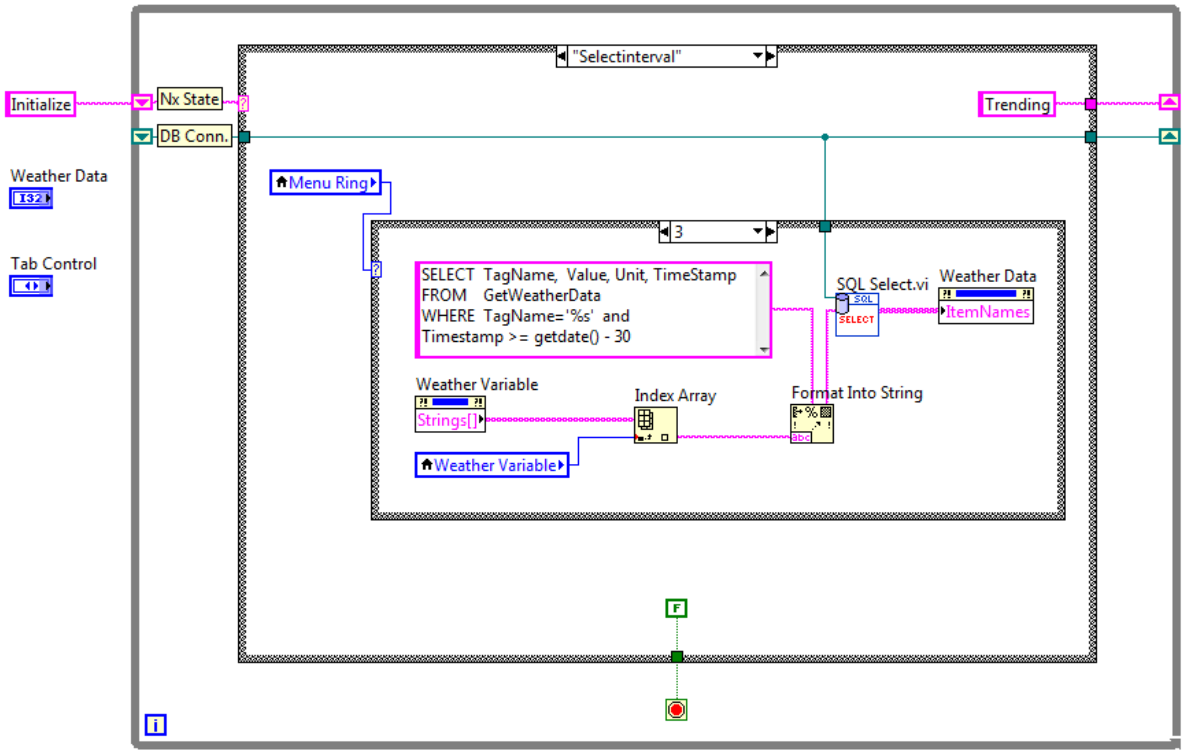

"Wait", Default
Initialize
Nx State
DB Conn.
Weather Data
I32
Tab Control
[3] "Menu Ring": Value Change
Type
Time
CtlRef
OldVal
NewVal
SelectInterval
Menu Ring
U16
F
i

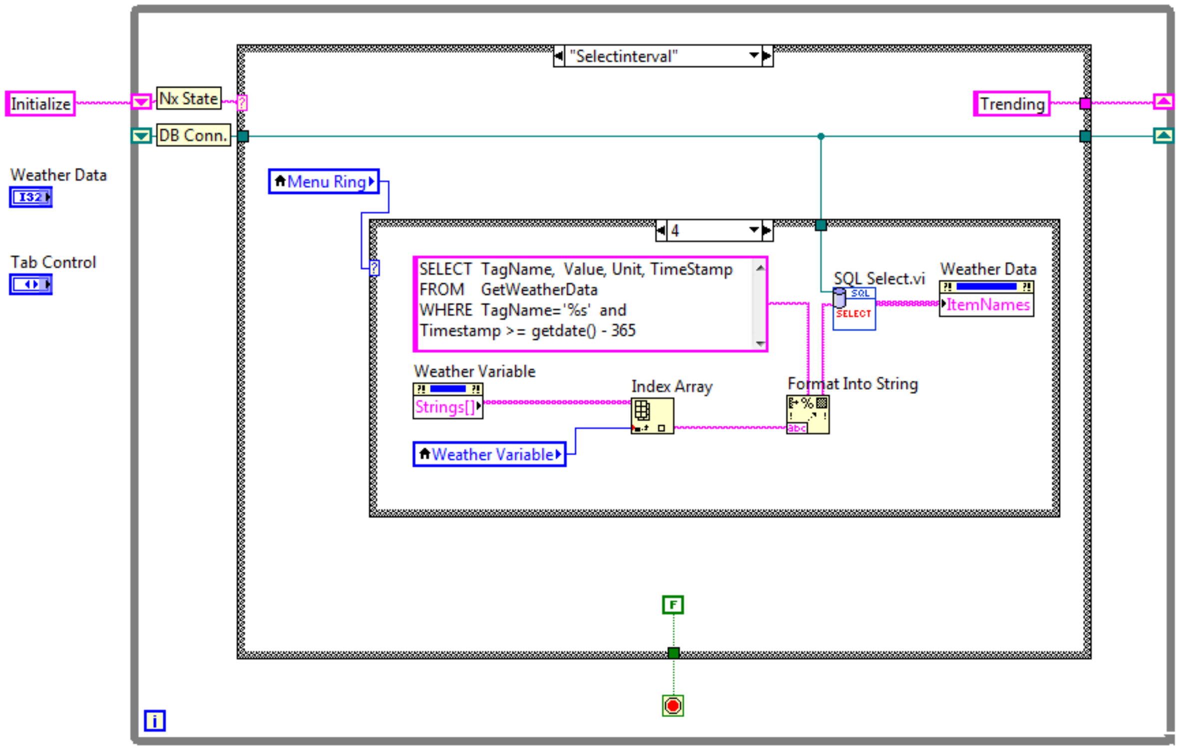

"Get Weather Variables"
Initialize
Nx State
DB Conn.
Weather Data
Tab Control
Wait
SQL Select.vi
Index Array
Weather Variable
select TagId, TagName from tag
1
Strings[]

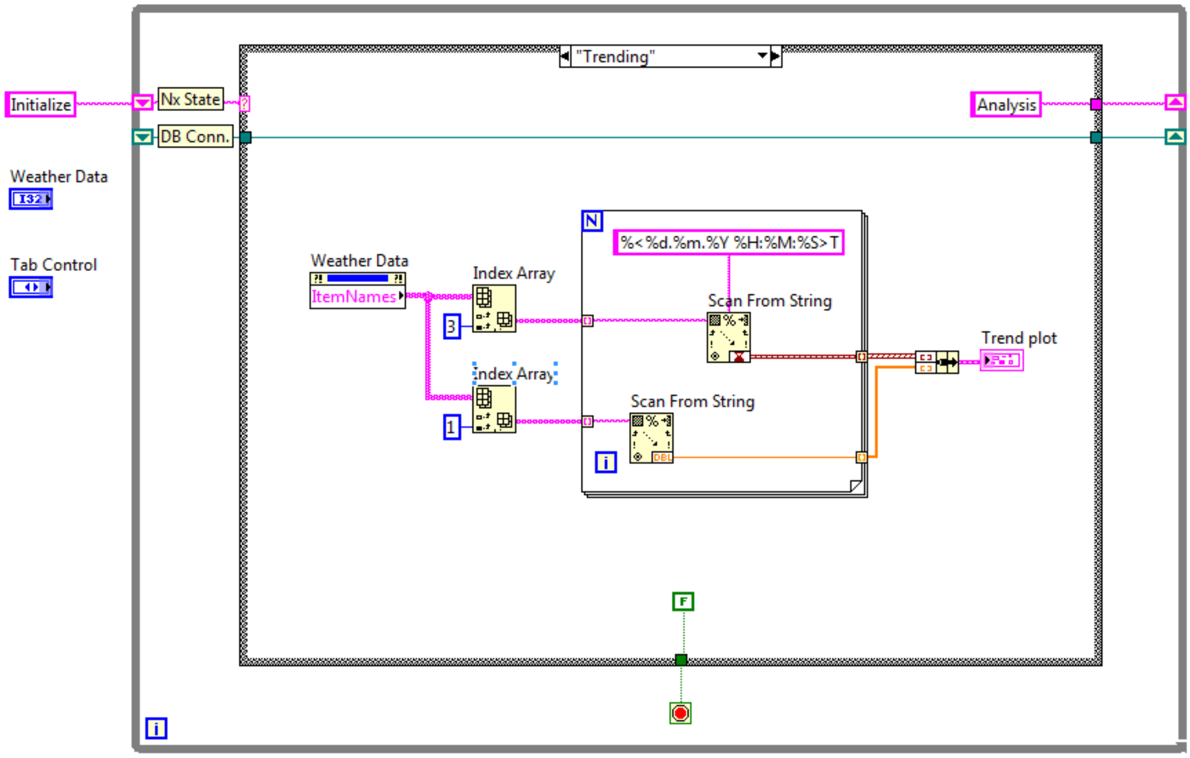

"Get Weather Data"
Initialize
Nx State
DB Conn.
Weather Data
Tab Control
Trending
from GetWeatherData
where TagName='%s'
and TimeStamp between '%s' and '%s'
Weather Variable
Strings[]
Index Array
Format Into String
SQL Select.vi
Weather Data
ItemNames
Weather Variable
%Y-%m-%d %H:%M:%S%3u
FromDate
Format Date/Time String
%Y-%m-%d %H:%M:%S%3u
Format Date/Time String
ToDate

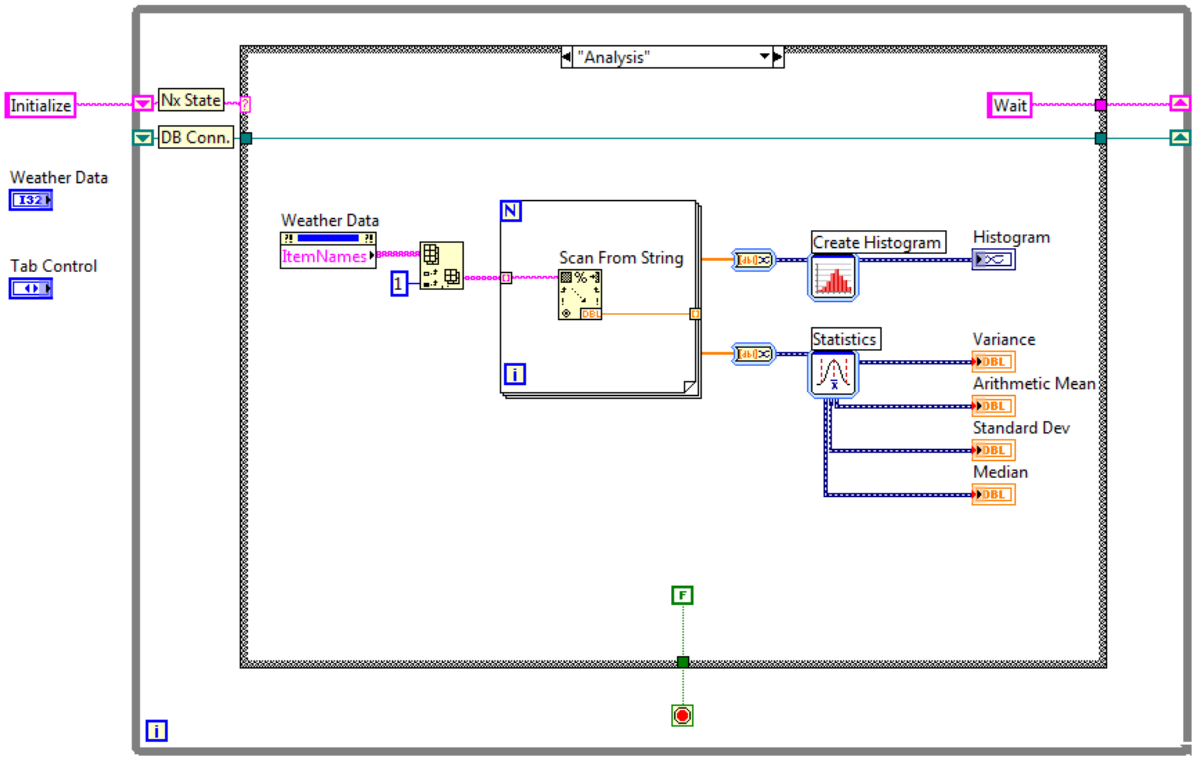

"Selectinterval"
Initialize
Nx State
DB Conn.
Trending
Weather Data
I32
Tab Control
Menu Ring
1
SELECT TagName, Value, Unit, TimeStamp
FROM GetWeatherData
WHERE TagName='%s' and
Timestamp >= getdate() - 1 order by
TimeStamp
SQL Select.vi
Weather Data
ItemNames
Weather Variable
Strings[]
Index Array
Format Into String
Weather Variable
F

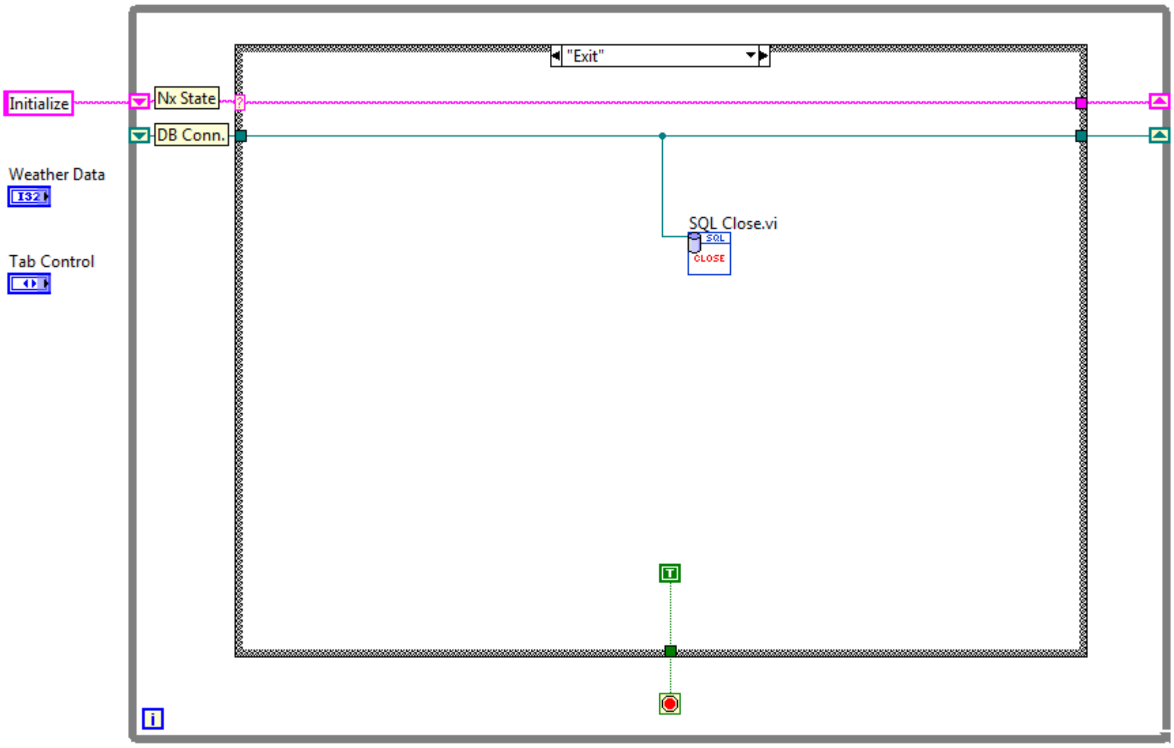

"Selectinterval"
Initialize
Nx State
DB Conn.
Trending
Weather Data
I32
Tab Control
Menu Ring
2
SELECT TagName, Value, Unit, TimeStamp
FROM GetWeatherData
WHERE TagName='%s' and
Timestamp >= getdate() - 2
SQL Select.vi
Weather Data
ItemNames
Weather Variable
Strings[]
Index Array
Format Into String
Weather Variable
F


"Selectinterval"
Initialize
Nx State
DB Conn.
Trending
Weather Data
Tab Control
Menu Ring
3
SELECT TagName, Value, Unit, TimeStamp
FROM GetWeatherData
WHERE TagName='%s' and
Timestamp >= getdate() - 30
SQL Select.vi
Weather Data
ItemNames
Weather Variable
Strings[]
Index Array
Format Into String
Weather Variable
F


"Selectinterval"
Initialize
Nx State
DB Conn.
Trending
Weather Data
Tab Control
Menu Ring
4
SELECT TagName, Value, Unit, TimeStamp
FROM GetWeatherData
WHERE TagName='%s' and
Timestamp >= getdate() - 365
SQL Select.vi
Weather Data
ItemNames
Weather Variable
Strings[]
Index Array
Format Into String
Weather Variable
F

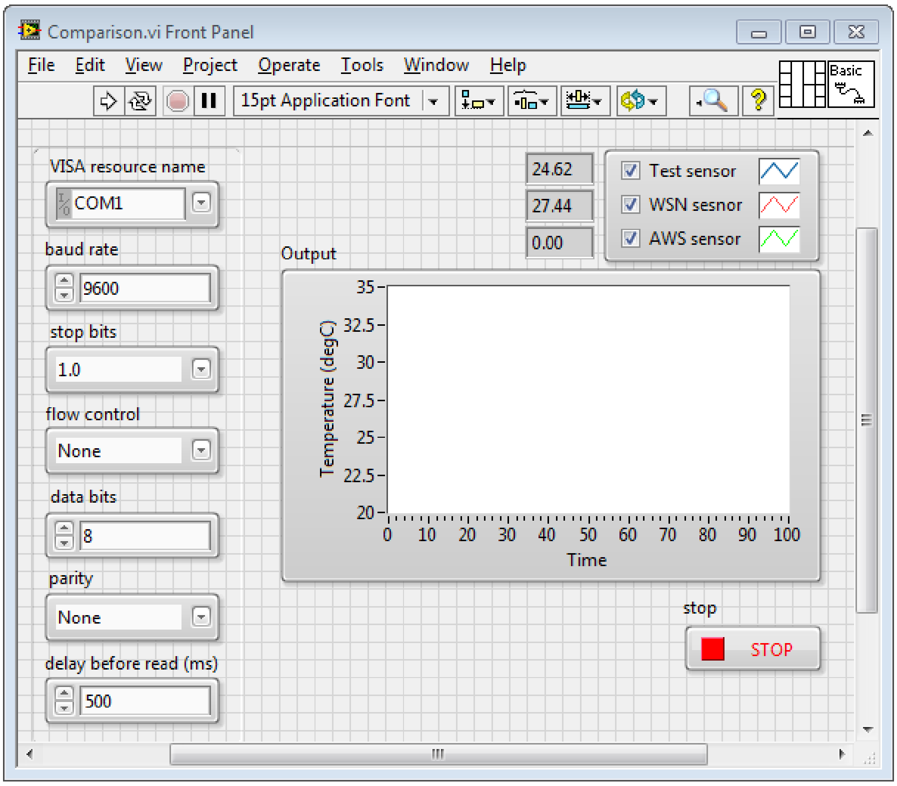

"Trending"
Initialize
Nx State
DB Conn.
Analysis
Weather Data
Tab Control
Weather Data
ItemNames
Index Array
3
Index Array
1
%<%d.%m.%Y %H:%M:%S>T
Scan From String
Scan From String
Trend plot


"Analysis"
Initialize
Nx State
DB Conn.
Wait
Weather Data
Tab Control
Weather Data
ItemNames
1
Scan From String
Create Histogram
Histogram
Statistics
Variance
Arithmetic Mean
Standard Dev
Median

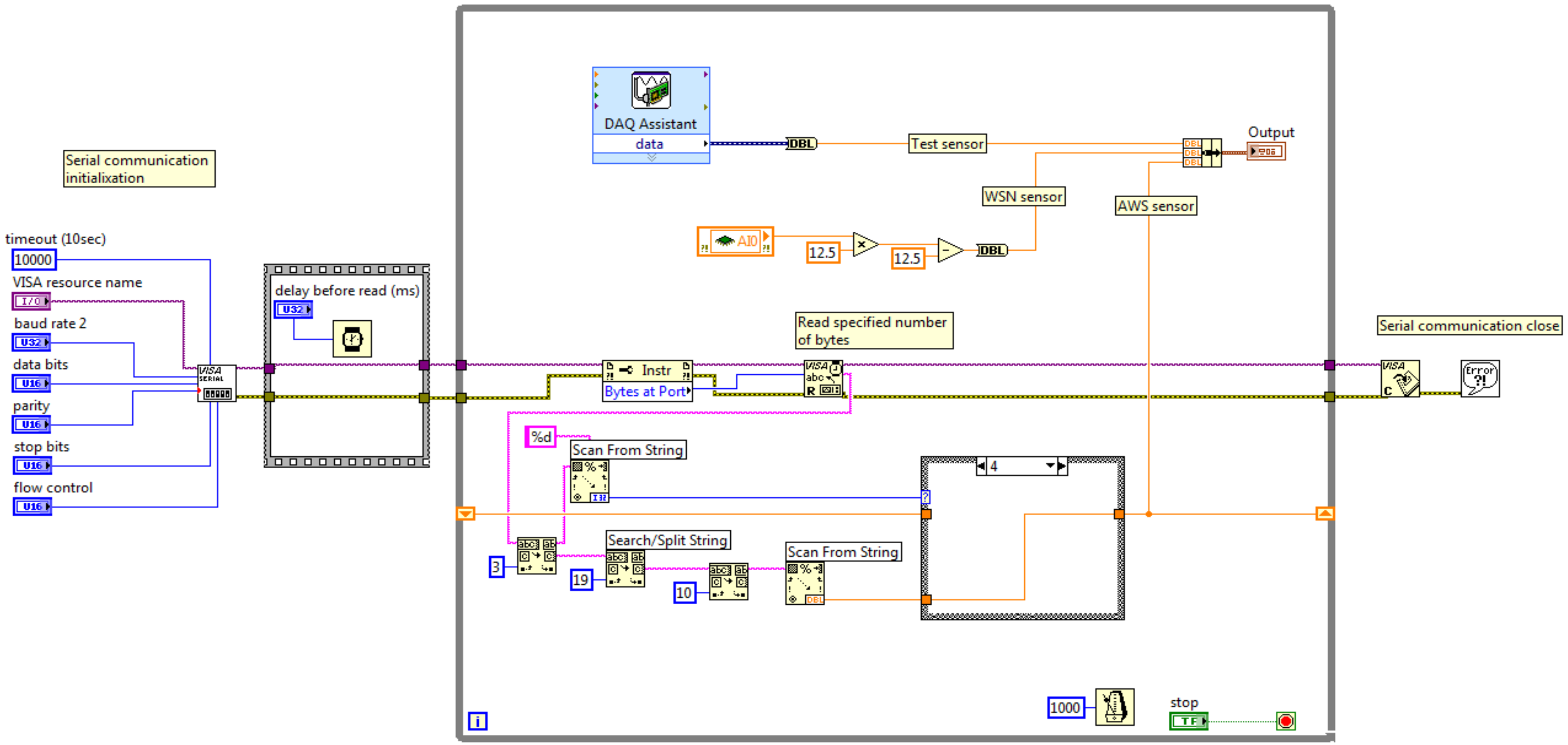


LabVIEW code for comparison.vi

Front panel



Block diagram for Comparison.vi



Appendix D Meteorological data

OBSERVASJONER

Stasjoner

Stnr	Navn	Måned	Dag	Hoh	Kommune	Fylke	Region
30420	SKIEN - GEITERYGGEN	okt.62	136		SKIEN	TELEMARK	Ã˜STLANDET

Elementer

Kode	Navn	Enhet
PO	Lufttrykk i stasjonsnivÃ¥	hPa
TA	Lufttemperatur	ÂºC

****

Stnr	Ã…r	Mnd	Dag	Time(NMT)	TA	PO
30420	2012	5	15	1	4.8	992
30420	2012	5	15	2	5	992.3
30420	2012	5	15	3	5.1	992.3
30420	2012	5	15	4	4.9	992.4
30420	2012	5	15	5	4.6	992.2
30420	2012	5	15	6	6.3	992.6
30420	2012	5	15	7	7.1	992.6
30420	2012	5	15	8	7.6	993
30420	2012	5	15	9	8.5	993
30420	2012	5	15	10	9.8	993.2
30420	2012	5	15	11	10.2	993.1
30420	2012	5	15	12	10.2	993
30420	2012	5	15	13	10.6	992.7
30420	2012	5	15	14	10.6	992.6
30420	2012	5	15	15	11	992.6
30420	2012	5	15	16	11.2	992.3
30420	2012	5	15	17	10.1	992.2
30420	2012	5	15	18	9.1	992
30420	2012	5	15	19	8	992
30420	2012	5	15	20	7.6	992.1
30420	2012	5	15	21	7	992.1
30420	2012	5	15	22	6.5	992.2
30420	2012	5	15	23	6.3	992.2
30420	2012	5	15	24	6.4	992.2
30420	2012	5	16	1	6.4	992.2
30420	2012	5	16	2	6.5	992.4
30420	2012	5	16	3	6.3	992.2
30420	2012	5	16	4	6.6	992
30420	2012	5	16	5	6.1	992
30420	2012	5	16	6	6.1	992.3
30420	2012	5	16	7	6.9	992.6
30420	2012	5	16	8	8.6	992.5
30420	2012	5	16	9	9.8	993

30420	2012	5	16	10	11	992.6
30420	2012	5	16	11	12.3	992.7
30420	2012	5	16	12	13.8	992.2
30420	2012	5	16	13	13.8	992.2
30420	2012	5	16	14	14.3	991.8
30420	2012	5	16	15	15.1	991.6
30420	2012	5	16	16	14.5	991.3
30420	2012	5	16	17	9.7	992.1
30420	2012	5	16	18	8.5	992.5
30420	2012	5	16	19	7.9	992.5
30420	2012	5	16	20	7.8	992.3
30420	2012	5	16	21	7.8	992
30420	2012	5	16	22	7.8	991.9
30420	2012	5	16	23	7.5	991.7
30420	2012	5	16	24	7.3	991.3
30420	2012	5	17	1	7.4	990.9
30420	2012	5	17	2	7	990.7
30420	2012	5	17	3	6.2	990.3
30420	2012	5	17	4	6.2	990.2
30420	2012	5	17	5	6.2	989.9
30420	2012	5	17	6	6.1	989.7
30420	2012	5	17	7	6.2	989.5
30420	2012	5	17	8	6.5	989.7
30420	2012	5	17	9	6.9	989.7
30420	2012	5	17	10	8.1	989.6
30420	2012	5	17	11	8.6	989.4
30420	2012	5	17	12	10.1	989.2
30420	2012	5	17	13	9.1	989.2
30420	2012	5	17	14	9.5	989.1
30420	2012	5	17	15	8.8	989.1
30420	2012	5	17	16	10.3	989
30420	2012	5	17	17	10.8	988.8
30420	2012	5	17	18	10.8	988.8
30420	2012	5	17	19	12.2	988.9
30420	2012	5	17	20	11	989.1
30420	2012	5	17	21	9.4	989.4
30420	2012	5	17	22	7.8	989.6
30420	2012	5	17	23	7.5	989.9
30420	2012	5	17	24	6.4	989.9
30420	2012	5	18	1	5.9	990.3
30420	2012	5	18	2	4.9	990.6
30420	2012	5	18	3	4.7	990.6
30420	2012	5	18	4	3.7	990.7
30420	2012	5	18	5	4.2	990.9
30420	2012	5	18	6	5	991.1
30420	2012	5	18	7	6.9	991.4
30420	2012	5	18	8	9.6	992.2
30420	2012	5	18	9	11.4	992.2
30420	2012	5	18	10	12.7	992.5
30420	2012	5	18	11	13.8	992.8

30420	2012	5	18	12	14.6	993.1
30420	2012	5	18	13	14.2	993.5
30420	2012	5	18	14	13.4	993.6
30420	2012	5	18	15	13.4	994.1
30420	2012	5	18	16	14.1	994.3
30420	2012	5	18	17	13.5	994.6
30420	2012	5	18	18	12.6	995
