Internship Report

Solarthermie-2000 Solarsimulation

University of Applied Sciences Offenburg

Author:	Dipl.-Ing. (FH) Tobias Kiefer
Matrikel Nr.:	164282
Working Period:	01.08.03 – 30.09.03
Date:	26. August 2003

Table of Content

Table of Figures

1. Introduction

The goal of this internship has been to set up a solar simulation in hardware and software in the research group "Solarthermie-2000" at the University of Applied Sciences Offenburg / Germany.
The main intention to set up such a project is to simulate the outdoor light conditions, not only with one single value at one specific moment but also of a whole day.

The project contains two main areas which have to be developed; hardware and software. The parts of the hardware are a halogen lamp, simulation box, dimmer for the lamp, control module, measuring board of National Instruments (PCI 6014 with 16 analoque input and 2 analoque output channels) and a connector block which is connected to the measuring board and the control module.
The programming task has been writing a program with the aid of LabView 6.1 which is a graphical programming languag in G. The program contains three different parts:

1. Reading values from an already existing log-file
2. Entering one single value to remote control the halogen lamp
3. Entering various values to remote control the halogen lamp

The following figure 1.1 illustrates the overall project with its all components:

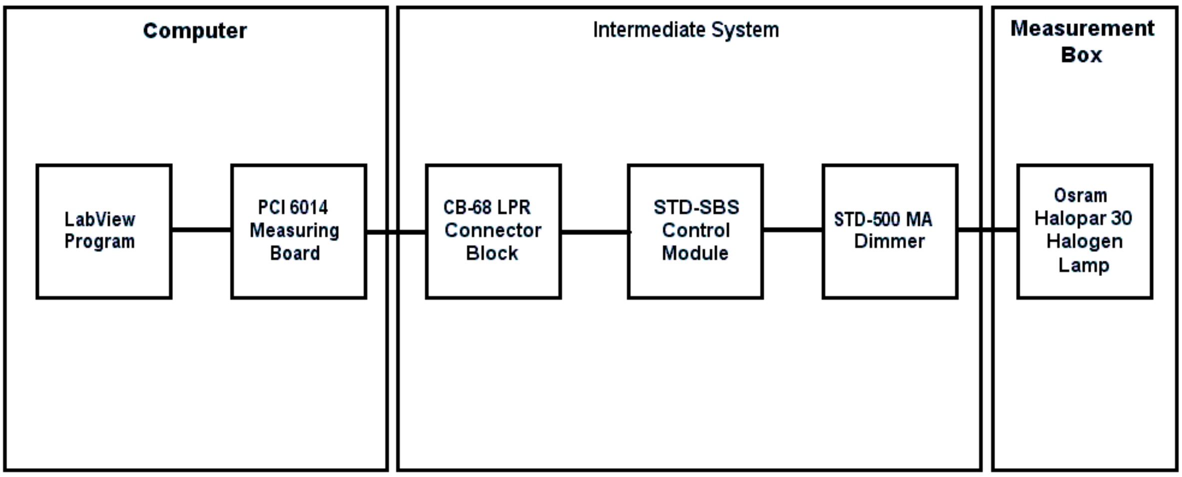


Figure 1.1: Block Diagram Overall Project

2. Hardware

Different options are possible to set up the hardware which are listed below. Two criterias have been desicive for the final result; the price and the availibilty of the various componets.

1. Option: (ca. 1260,- Euro)
- controllable power supply
- lighting devices
- A/D Measuring board of National Instruments

2. Option: (ca. 905,- Euro)
- normal power supply
- power transistor
- lighting devices
- A/D Wandler Measuring board von National Instruments

3. Option: (ca. 1720,- Euro)
- power supply with RS 232 interface
- lighting devices

4. Option: (ca. 1037,- Euro)
- 230 V lighting devices
- control module
- dimmer STD 500 MA
- A/D Measuring board of National Instruments

The last option and its components are taken, because a local distributer for electronic devices were able to deliver the componets very fast and also the price is quite acceptable.

2.1 Lighting Devices

The lighting device in option 4 requires a supply voltage of 230 V AC. For solar simulation, it would be better to use a lighting device which is operated by a DC voltage thus less ripples are caused on the supply voltage of the lighting device. As the sensors for measuring the light are too slow, the ripple does not effect any distortion on the supply voltage of the lighting device.
Furthermore, the lighting device should be very similiar to the real light conditions of the sun and the lighted area should be constant thus several sensors get the same amount of light. The so called AM-value (Air Mass) should be 1.5 which emits a light that is similar to the real sun beam.

Unfortunately, the producers and distributers of lamps do not mention anything about the irradiance, only about the light intensity which has a minor role for this project. Because of this missing information, several light devices are tested if they fulfill the requirements.

Requirements for the lighting device:

- 1000 watt / meter² at the sensor
- a big diameter of the light cone thus two sensors can be placed in it
- amount of light should be constant in this area
- equidistribution of the light --> no incandescent lamp, but halogen lamp
- possibility to dim

Different opportunities for a lighting device:

1. <u>Osram Halopar 30</u>:

- Base E27
- Halogen Lamp
- Rated volts 230 V
- Rated wattage in Watts 75 W
- Beam angle in degrees 30 °
- Luminous intensity in candela 2400 cd

Figure 2.1: Osram Halopar 30

- Average lamp life in hours 2000 h
- Operating without any transformator
- Opportunity to dim
- With cool-beam-reflector for less heat accumulation

2. Philips 13117:

- Halogen Lamp for special use in solar simulation
- Rated voltage of the lamp is 17 volt
- usually operated with 13 lamps in serious connection
- using a transformator is nessecary

3. Osram Concentra PAR 38 SP120:

- Base E27
- Rated volts 240 V
- Rated wattage in Watts 120 W
- Beam angle in degrees 12 °
- Luminous intensity in candela 9300 cd
- Operating without any transformator
- Opportunity to dim

Figure 2.2 Osram Concentra PAR 38

The both Osram lamps are bought but currently only the Osram Halopar 30 is in use because the other Osram halogen lamp is probably too strong. The most disadvantage of the Philips halogen lamp is, that the power supply is 17 V. For that reason, a transformator would have been bought with 17 V at the secondary side which is not very common.

2.2 Control Module STD-SBS

The control module STD-SBS is used in combination with the dimmer STD-500 MA of ABB. It allows the controlling of this dimmer with an analogue voltage between 0 V and 10 V. To use the analogue mode, the pin 6 has to be set to "ON" and the pin 1 has to be set to "OFF" for controlling the dimmer with a voltage and not with a current.

Below 10% of the input signal, the device is off; above this edge the device is increasing the brightness untill it has reached 100% of the brightness.

It is important that the length of the cable between STD-SBS and STD-500 MA is less than 2 meters. The analogue input of this device is connected to the measuring board PCI 6014 of National Instruments via the connector block. The control voltage is offered by the measuring board and the software.

2.3 Dimmer STD-500 MA

The universal power dimmer STD-500 MA are used to dim the light of 230 V halogen lamps, incandescent lamps and low-voltage halogen lamps. This dimmer can be controlled with a control module via a data line which is already mentioned above. This dimmer can be used for lamps with a power up to 500 W or for a current up to 2.17 A.

The following figure 2.3 shows the circuit diagram for the STD-SBS controle module and the STD-500 MA dimmer:

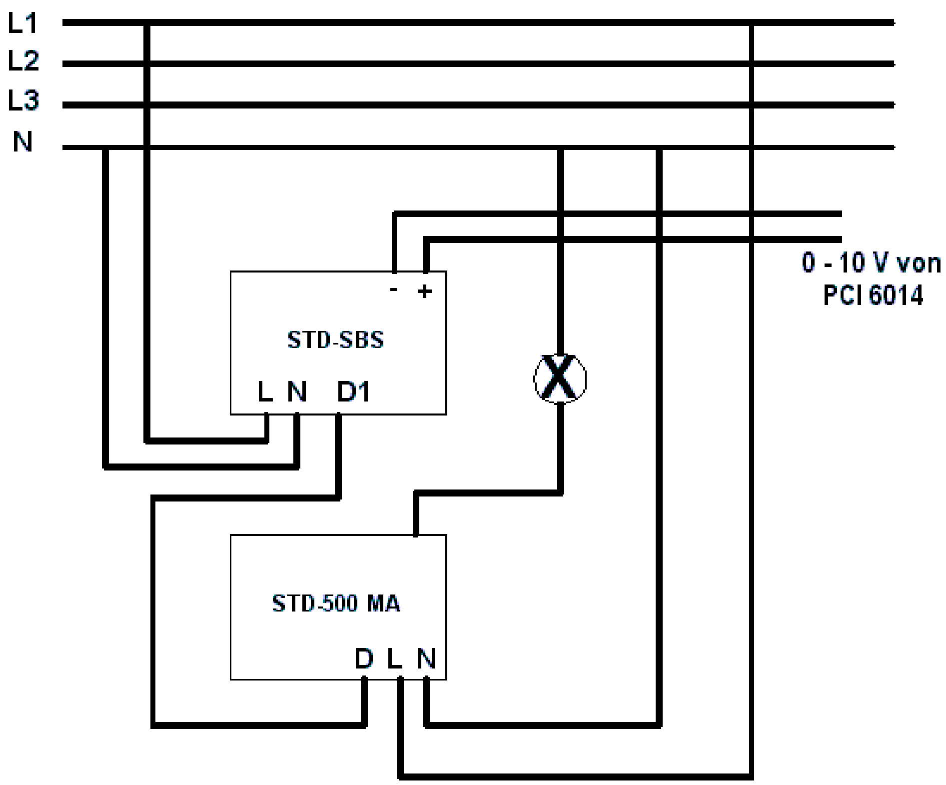


Figure 2.3: Circuit Diagram for STD-SBS and STD-500 MA

2.4 Measuring Board PCI 6014

As only the output channels of the measuring board are important for the use in this project, it was enough to buy one of the cheapest A/D converter measuring boards of National Instruments. This PCI card has 16 analogue input channels and only two analogue outputs which is still enough for the task of this project. Of course, as there are many other channels available, it is possible to use this card also for other applications in the near future.

Some problems appeared with the installation of this card. Usually the PCI 6014 is used in combination with the programming software LabView which is also produced by National Instruments. If the the user has installed this software before the measuring board is installed in the computer, problems can appear.

The software of the PCI 6014 contains not only the drivers for the card but also a so called "Meassurement and Automation Explorer". With this software, the user can check the funcionality of the measuring board very easily. The input and the output channels can be tested with a special test panel on which the user can choose a voltage to be given to the analogue output of the card.

The following figure 2.4 shows the PCI 6014 measuring board of National Instruments:

Figure 2.4: PCI 6014 Measuring Board of National Instruments

2.5 Connector Block CB-68 LPR

The connector block CB-68 LPR is used for connecting the PCI 6014 card with the STD-SBS controle module. As the connector block has not such a plug as the PCI 6014, it is possible that the cable between the measuring board and the connector block is attached upside down.

The figure 2.5 illustrates the connector block CB-68 LPR:

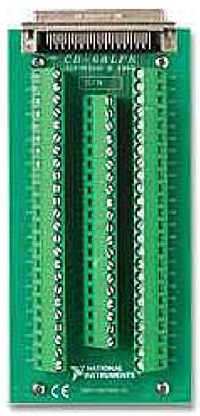


Figure 2.5: Connector Block CB-68 LPR of National Instruments

2.6 Measurement Box

This box has been constructed to perform measurements for the solar simulation. The most important reason to use such a box is, that no other light can irridiate to the sensors which are placed at the bottom of the box. Otherwise the measurements can get wrong and no reliable results are possible.

On top of the box, the halogen lamp Osram Halopar 30 is fixed to a socket of type E27. At the side panel of the box, two fans are mounted to blow fresh air into the space of the box thus the temperature will not get too high. The fans are protected by a guard that people cannot hurt their fingers while they are working with the measurement box. The two fans are operated with 230 V which means that no transformator has to be used. At the lower part of both side panels, borings with a diameter of 10 mm are placed in three rows to let the airflow enter the box. The front panel can be lifted up thus the user can change the place of the sensors or the distance between sensor and the lamp.

The complete measurement box will be mounted to the test stand of the research group "Solarthermie-2000".

The next figures 2.6 and 2.7 show the engineering drawings of the measurement box from different views:

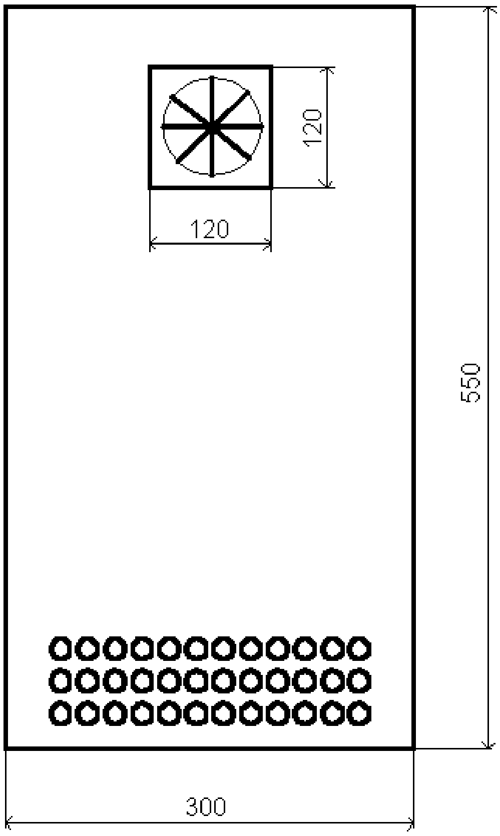


Figure 2.6: Side Panel

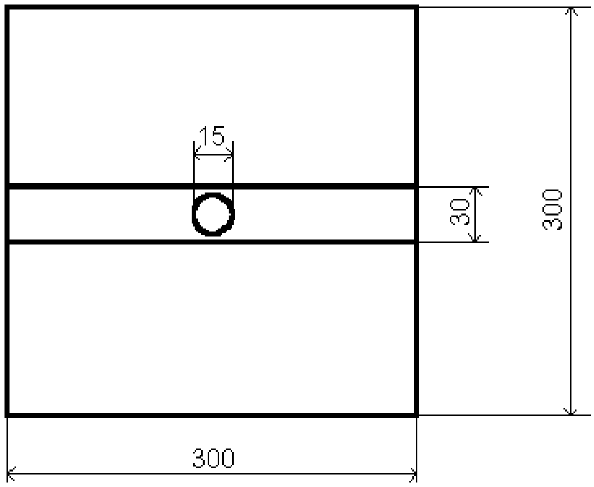


Figure 2.7: Top View

3. Software

The main part of this project has been to develop a software which fulfills the three different options which are already mentioned in chapter 1.

The first part of the software should read data from an already existing log file of one location where a solar panel has been installed. This data should be simulated in a specific time length. The second part should allow the user to enter a value for the irradiance which should be simulated. This value is transformed to a voltage which controls the control module of the hardware part. The third part of the software is to enter multiple data, thus a ramp can be simulated. The data will be entered into a table by the user. Furthermore the user can determine in which period the ramp should have reached the corner points which has been entered into the table. The data of the first and the third part will be illustrated in a curve chart which should help the user to follow the running software. Furthermore the user is able to change between several measuring cards and analogue outputs if it is necessary. This can be done in the part which is called "Analogue Output Control".

The next figure 3.1 shows the front panel of the software with its three different parts, the curve charts and the analogue output control.

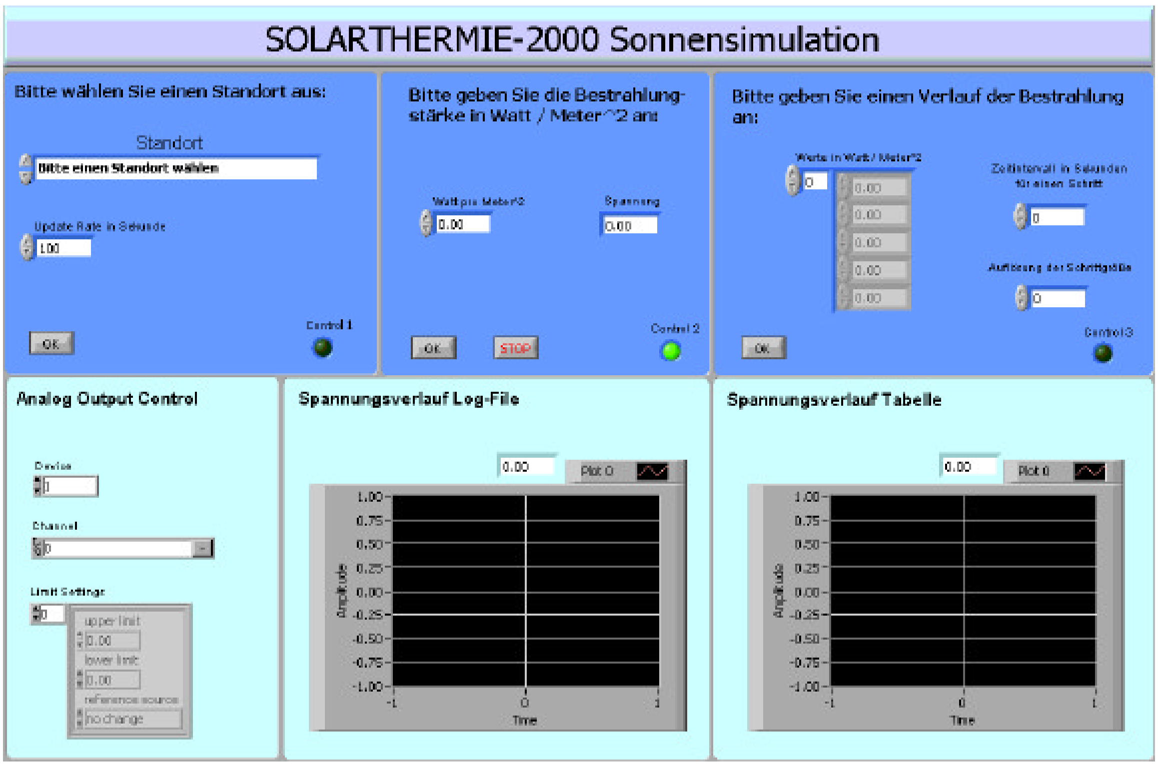


Figure 3.1: Front Panel of the Software

3.1 Main Program and SubVis

LabView is a graphical programing language but it has the same structures like textual programing languages such as C or C++.
In LabView, a main program is writen and it contains several SubVis (sub programs). Some of this SubVis are just taken from the installed software, other SubVis are written for the special use of this task.
The complete program contains the following SubVis which are written for the specific use of this task:

- Steuerung_OK _Buttons.vi
- Fehlermeldung.vi
- SearchAlgorithm.vi
- Referenztabelle.vi
- Calculation.vi
- Calculation_for_Table.vi

Other SubVis are used either. This Vis are taken from the software LabView which can be found in the section of examples:

- AO Harware Config.vi
- AO Group Confi.vi
- AO Single Update.vi
- AO Single Update (scaled array).vi
- General Error Handler.vi

For a better understanding of the program and that no errors occur, it is always good to use sequences for the programing. One sequence comes after each other in the main program thus the program can follow the right steps. If such sequences are not used for programing, it is possible that LabView interprets the source code wrong. LabView reads from left to right and from top to down.

The following figure 3.2 shows the hierachy of the complete program:

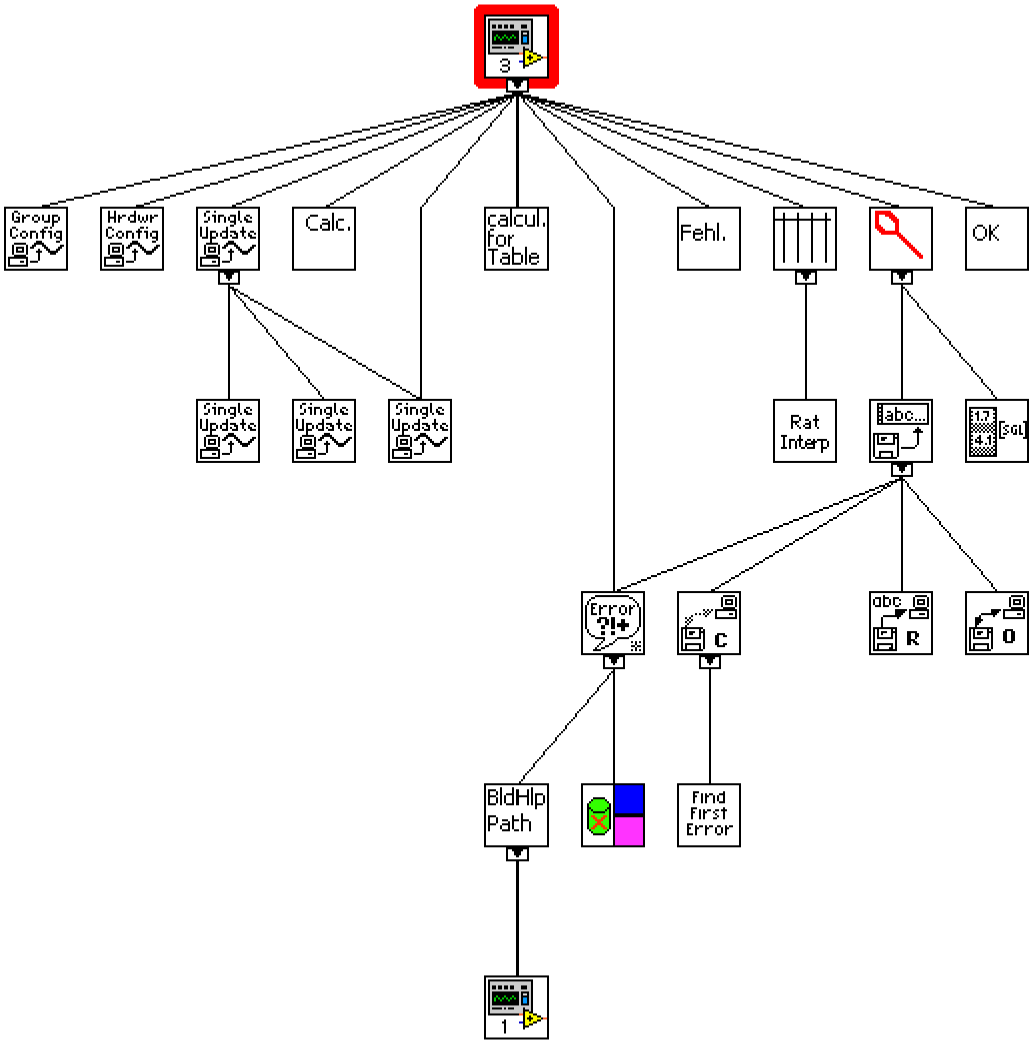


Figure 3.2: Hiearchy of the program

The uppermost square which is highlighted with red color is the main program. It contains the SubVis which are mentioned above. These SubVis contain other SubVis and so on thus a whole structure of sub programs are built.

3.2 Simulating from a Log File / Part 1

The difficultiy of this part is that there are 6 different locations, on which solar panels have been installed. The log files of this locations are different from each other which means, that the position in the log file where the important data is written, is different.
Furthermore, the SubVis "Search Algorithm" and "Calculation" have to be especially written for this part of the software.

3.2.1 Log Files

As mentioned above, six different location have to be considered. In the following paragraph, one log file is listed to show the structure of such a file.

Log file of Albtherme Waldbronn:

100,2003,183,5,0,0.0000,28.7912,29.7527,25.5454,29.2784,29.2467,29.6771,36.4259
101,24.7288,51.4930,53.9511,50.9644,51.1176,22.8151,22.7601,24.1940,25.1734,24.0445,22.9933,22.6942,21.7267,22.4740,22.4189,22.4962,23.4930,31.6696,12.1627,14.8521,0.0000,0.0000,0.0000,0.0000,51.8815,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000
102,0.0000
110,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,28.7435,29.7415,25.5300,29.2606,29.2373,29.6639,36.3414,24.6968,51.4614,53.7343,50.8977,51.1012,22.7929,22.7389,24.1727,25.1474,24.0236,22.9676,22.6747
111,21.7138,22.4589,22.3998,22.4846,23.4635,31.6624,12.1274,14.7893,-0.7719,-1.4950
120,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,28.8806,29.7777,25.5843,29.2890,29.2710,29.7079,36.4758,24.8073,51.5579,54.0240,50.9864,51.1587,22.8905,22.8006,24.2395,25.2637,24.0930,23.0601,22.7338
121,21.7652,22.5128,22.4383,22.5026,23.4995,31.7011,12.2251,14.8844,0.0000,0.0000
100,2003,183,10,0,0.0000,28.8638,29.7749,25.6064,29.2587,29.2670,29.7005,36.5015

All log files start with the same numbers at the beginning of the line and the data is stored comma seperated. But only the lines 100 and 101 have to be considered for this task.

In line 100, the year, the day of the year and a relative time is mentioned: 2003,183,5

In line 101 are different data available such as temperature and so on. At position 23, the irradiance can be taken for further calculations in the program. In the case above, the value is 0.0000 because the measurment started in the night.

In the next figure 3.3, the selection of the locations is shown. This part of the program is in sequence 3 out of 9 which can be seen on top of the frame.

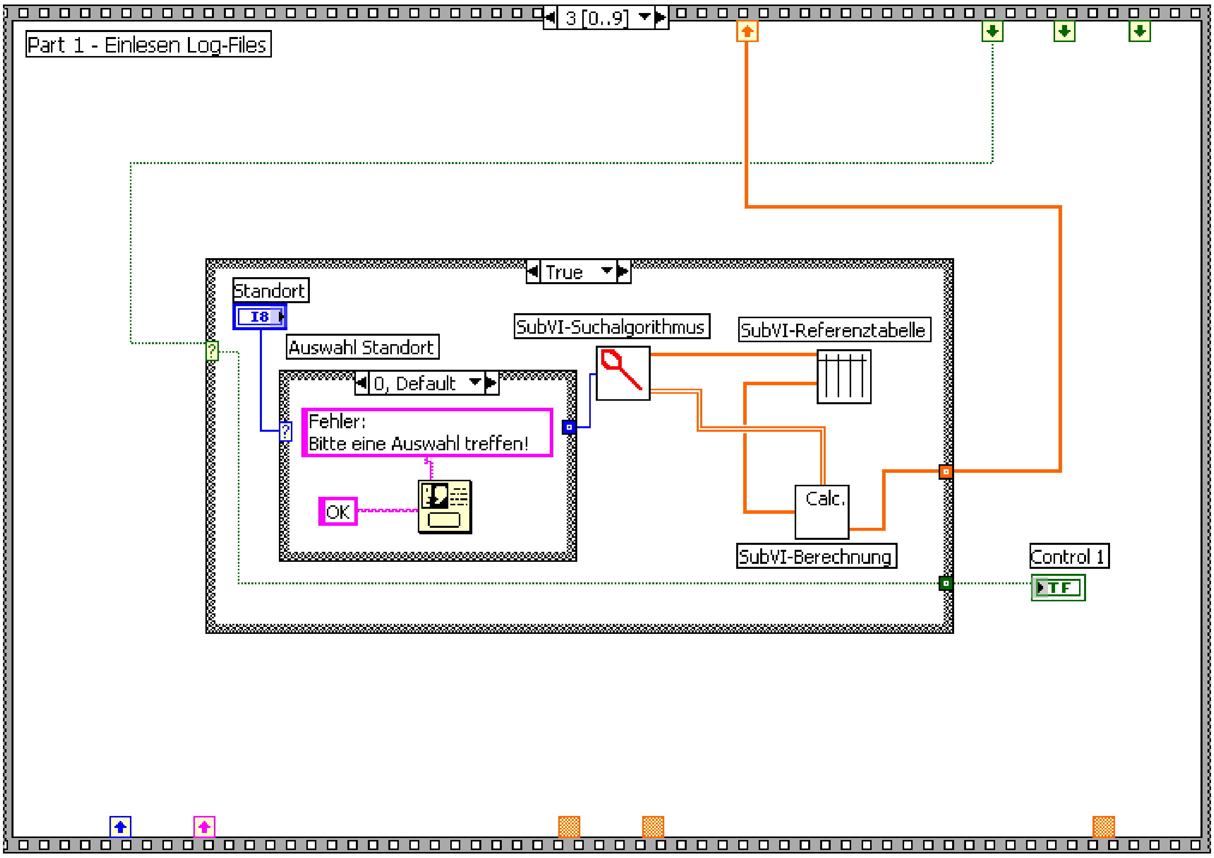


Figure 3.3: Block Diagram for the Selection of the Site

If the condition of the OK buttons has been right, a boolean value will be transmitted to the input of the case structure and the part 1 of the program will be performed. Another case structure inside of the outer case structure is responsible for the actual selection of the sites. By selecting a site from the front panel, which is shown in figure 3.1, a reference value will be transfered to the case structure. Every site has got its own value, starting from 0 to 6, to seperate from each other. The default value "0" is used to give an error feedback in case the user has not selected any location.
After selecting a location, the position of the important data will be transmitted to the SubVi "Search Algorithm" thus the data can be taken out of the log file.

The following figure 3.4 illustrates which numbers are transmitted to the SubVi "Search Algorithm":

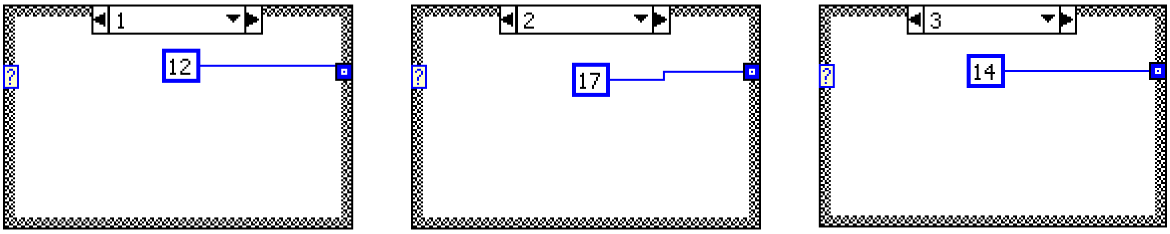


Figure 3.4: Decission of Location

Only the first three cases are shown in the figure above. The position of the data in the several log files are listed below:

1. Studentendorf Vauban Freiburg (12)
2. Kreiskrankenhaus Mindelheim (17)
3. Hegau Klinikum Singen (14)
4. Stadtklinik Baden-Baden (13)
5. Wilmersdorferstr. Freiburg (17)
6. Albtherme Waldbronn (21)

3.2.2 Search Algorithm

After a location has been selected from the front panel, the user has to start the program. A window opens to select a the corresponding log file from the location on the computer. This log file will be read into a for-loop which ends when the log file has reached its end. In the first for-loop, the program is looking for all the lines which start with the number "101". As the complete line is in one string, the elements of the string has to be seperated from each other thus a further working with the data is possible. The seperated data is written to an array with rows and columns. With the position index, which is shown in figure 3.4, the program takes only the necessary data out of the array.

The data are taken one by one out of the array into a for-loop which compares the current value of the data with a contant value of "0". As the data can get below "0", for example during the night, it is important to change all negative values to "0". The revised array of data is available at the end of the upper part.

In the lower part, the same procedure is done for the time. The year, day of year and the relative time are taken out of the log file and stored into a seperated array from the actual data (irradiance).

The figure 3.5 below shows the complete SubVi "Search Algorithm" with all its elements:

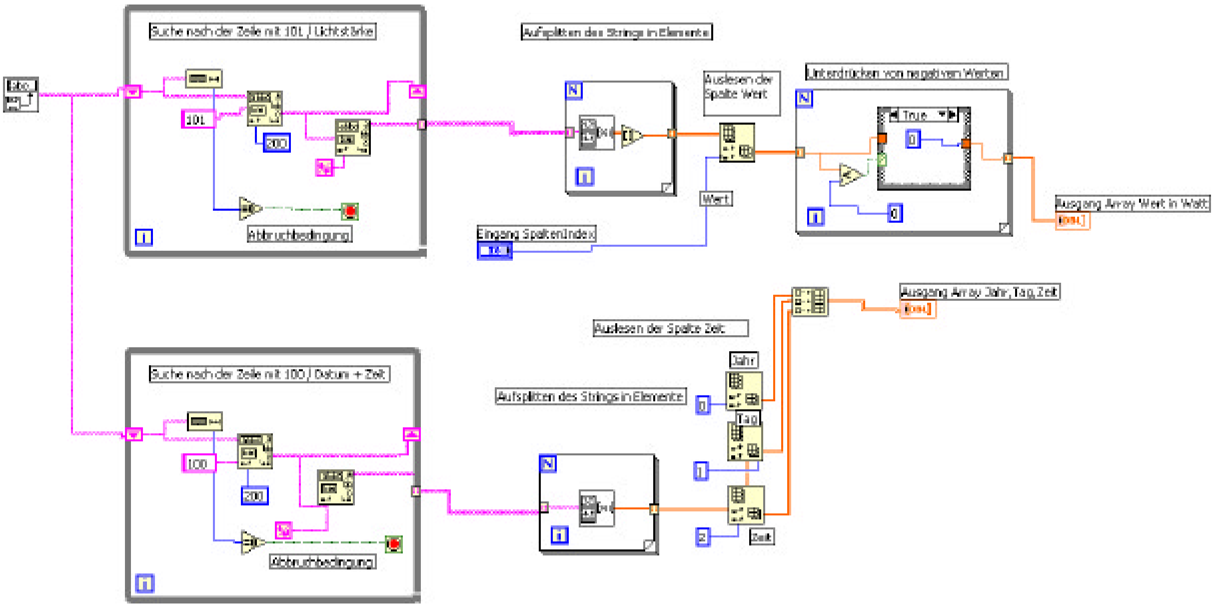


Figure 3.5: Block Diagram of Search Algorithm

3.2.3 Reference Table

The SubVI "Reference Table" is necessary because the data which is taken out of the log file is still in the unit of Watt / Meter². As the halogen lamp is controled by a reference voltage between 0 V and 10 V, the data of the log file has to be changed to voltages. This is done by the reference table.

To enter data into the reference table, measurements have to be done with the halogen lamp. But the maximimun value for the irradiance should not be more than 1300 Watt / Meter². For that reason the 10 V of the control voltage should cause an irradiance of about 1300 Watt / Meter². This can be done by varying the distance between the sensor and the halogen lamp. If other halogen lamps are used than the Osram Halopar 30, the reference table has to be renewed.

The reference table should be entered once by the user, afterwards it should be defined as constant values thus no changings can be done anymore. Of course, the more corner points are taken, the less is the error which is caused by the interpolation and the more exact gets the complete measurement.

The next figure 3.6 illustrates the block diagram of the SubVi Reference Table:

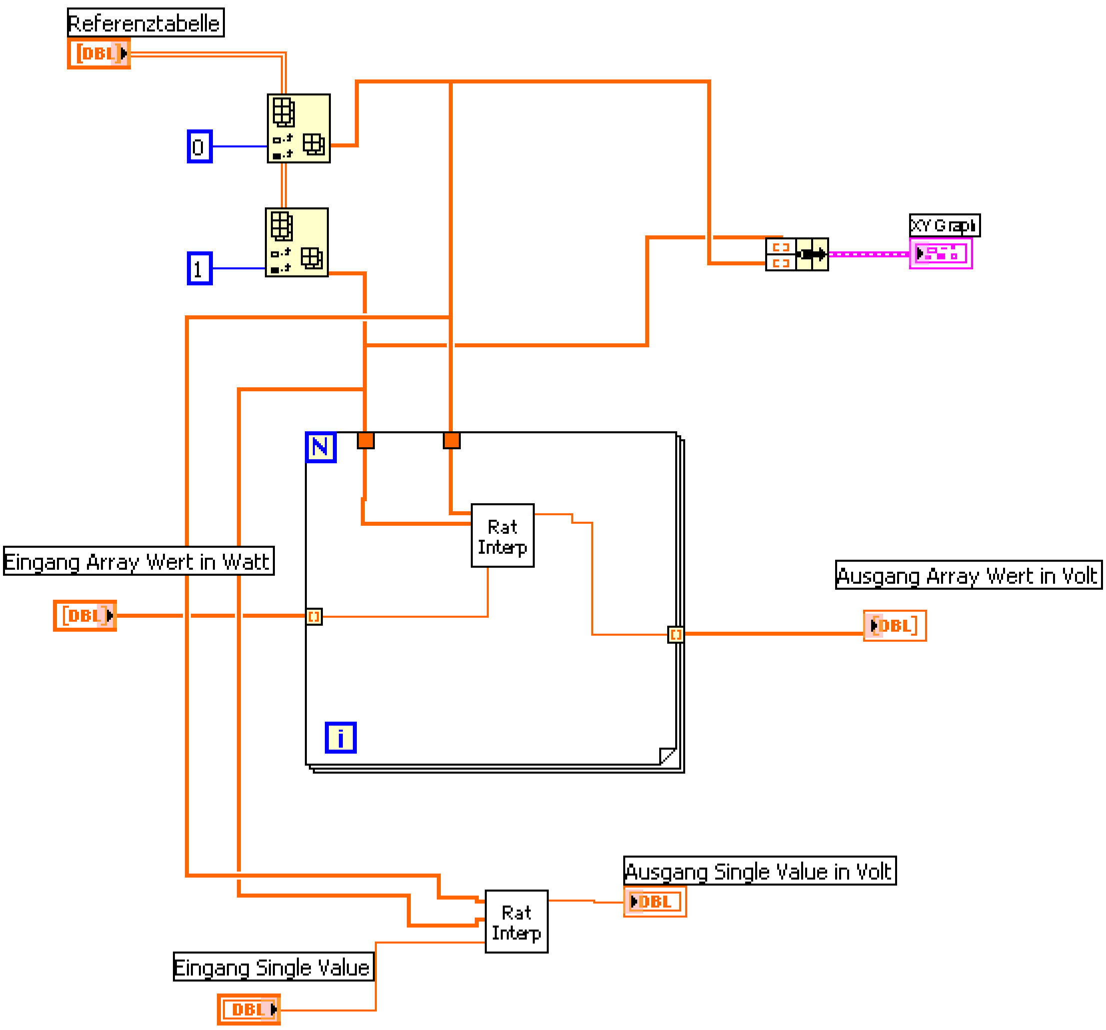


Figure 3.6: Block Diagram of Reference Table

The reference table is not only used for arrays of data but also for a single value, as it is used in part 2 of the software. As only corner points for the reference table are used, an interpolation has to be done between those points thus an exact changing from Watt / Meter² to Volt can be done. Both values of the reference table, Watt / Meter² and Volt, are given to the SubVi "Rat. Interp." via a for-loop. Furthermore, the data of the array is also given to this SubVi. The SubVi "Rat. Interp." calculates an approximate value for the control voltage. After all data is changed to voltages, an array at the right side of the figure 3.6 is available for further use in the program.

The same procedure is done for a single value, just without a for-loop. This changed value is also available for furher usage.

3.2.4 Calculation

Two different kinds of log files have to be considered as well. Some log files have a recording time of 5 minutes and some of 30 minutes. This means that either every 5 minute or every 30 minute a measurement data is stored into the log file.

The first two paragraphs show a log file with a measuring time of every 5 minutes. The position of the time is high lighted by green color.

100,2003,183,5,0,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.000
0,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,
0.0000,0.0000,0.0000,0.0000,0.0000,28.7912,29.7527
....

100,2003,183,10,0,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.00
00,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.000
0,0.0000,0.0000,0.0000,0.0000,0.0000,28.8638

The next paragraph shows a log file with a measuring time of every 30 minutes.

In this case an irregularity can be seen. Half hours are represented by thirties and full hours are represented by hundreds.

100,2003,3,30,0.25,0,0,0,0,0.5,0.5,0,0.5
....

100,2003,3,100,0.25,0,0,0,0,0.5,0.171,0.31,0.168
....

100,2003,3,130,0.125,0,0,0,0,0.501,0,0.501,0

This fact has to be considered for the further calculations. The following figure 3.7 shows the block diagram of the SubVi "Calculation":

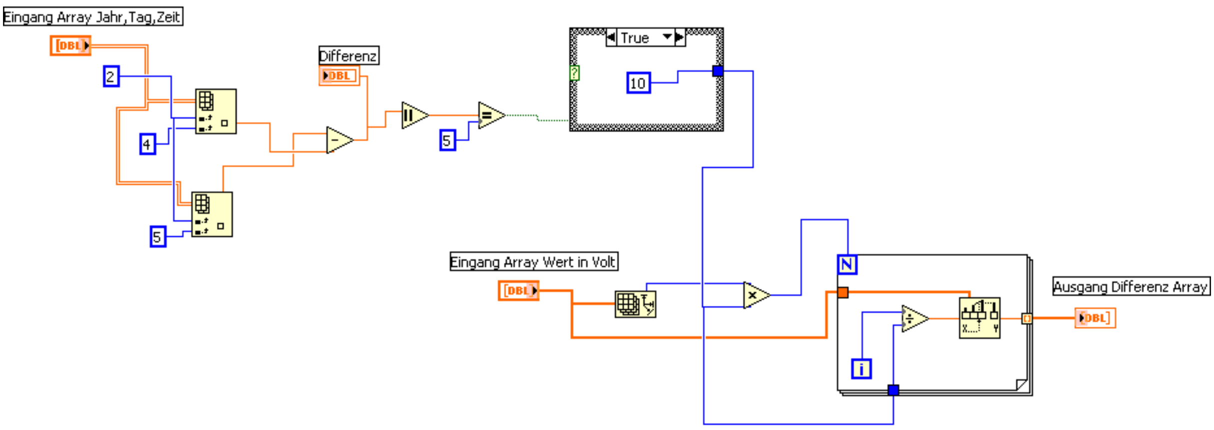


Figure 3.7: Block Diagram of the Calculation

The SubVi takes the time from the 4th and the 5th row of the array. Both values are subtracted. After this step, the absolute value is taken and compared with the constant number of "5". If the comparison is "true" the number "10" will be given to the for-loop. If "false" the number "60" will be given to the for-loop.
The following figure 3.8 illustrates the case "false":

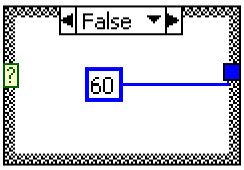


Figure 3.8: Case "False"

This two factors are used to generate an array of the same size after the interpolation between log files of 5 minutes recording time and log files of 30 minutes recording time. This interpolation is done to minimize jumps from one corner point to another one. The shape of the curve of the data is more smooth than without any interpolation.
After the new array is created the array can be used for the analogue output where the voltages are available to be given to the connector block.

The next figure 3.9 shows the output for the part 1of the software. At this output the user is able to change the speed of updating the voltages via the front panel.

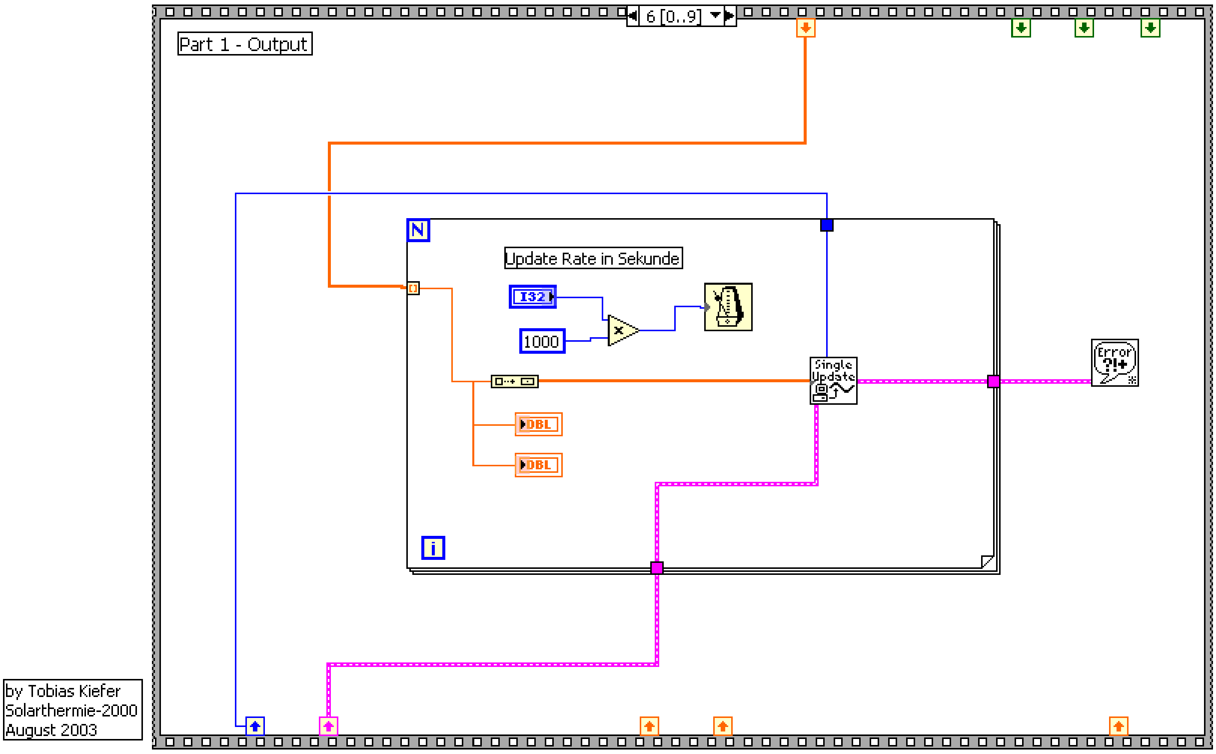


Figure 3.9: Block Diagram of the Output of Part 1

The array is applied to a for-loop which contains a time module and two indicators as an user feedback which are placed at the front panel. The two indicators are numerical display and a curve chart. The factor "1000" is used to change the users input for the update time from seconds to milliseconds which is required for this time module. The SubVi "Single Update" applies the voltages from the array to the analogue output of the measuring board, under the consideration of the time.

3.3 Simulating one specific Value / Part 2

In sequence 4 out of 9, the part 2 of the software is shown. This part is more easy than the first part because only one single value is changed to a voltage and applied to the analogue output afterwards.

The following figure 3.10 shows the block diagram for the part 2 of the software:

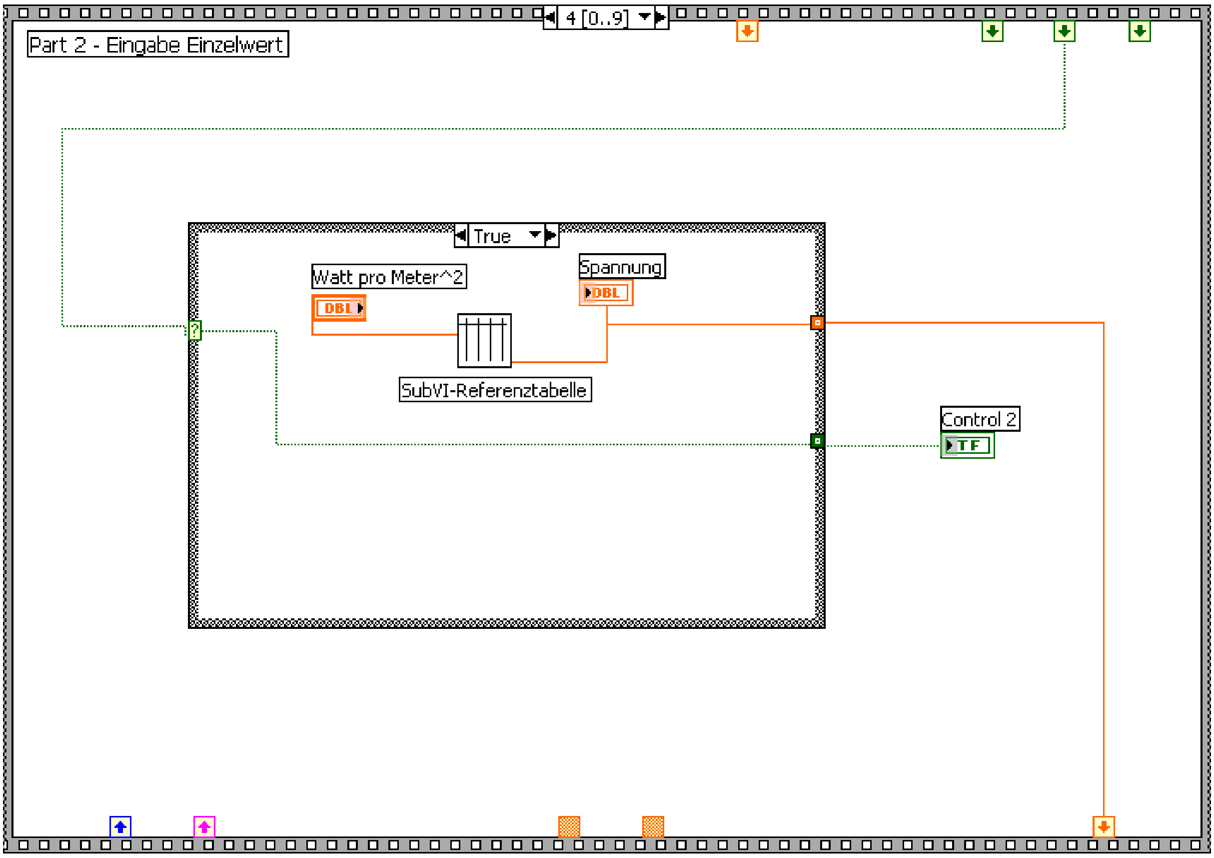


Figure 3.10: Block Diagram of part 2

If the condition of the OK buttons is right and the boolean value "true" is applied to the case structure, the part 2 will be executed. One single value is entered by the user via the front panel. This value is changed to a voltage by the reference table, which is explained in chapter 3.2.3. The entered value (irradiation) and the corresponding voltage are shown via numeric indicators at the front panel.
At the output of the case structure, the current voltage will be applied to the sequence 7 out of 9.

This sequence is illustrated in the figure 3.11 below:

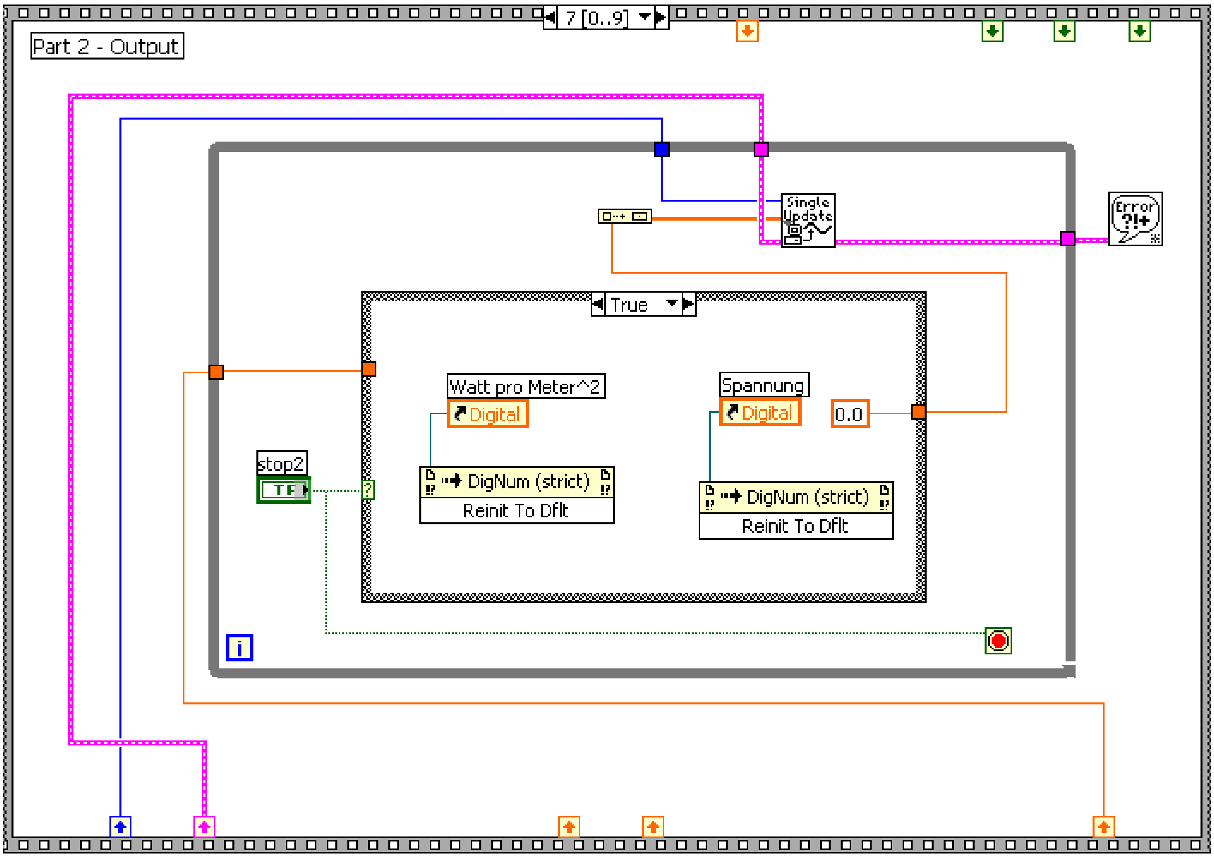


Figure 3.11: Block Diagram of the Output of Part 2

The value of the voltage is given to the for-loop, as well to the inner case structure. If the stop button at the front panel is pushed by the user, the numeric displays are reset to the default value "0" and no voltage is applied to the analogue output.

In case, that the stop button is not pushed by the user, the applied value of the voltage is straight given to the SubVi "Single Update" which is responsible for the voltage output at the measuring board.

The case "False" is illustrated in the next figure 3.12:

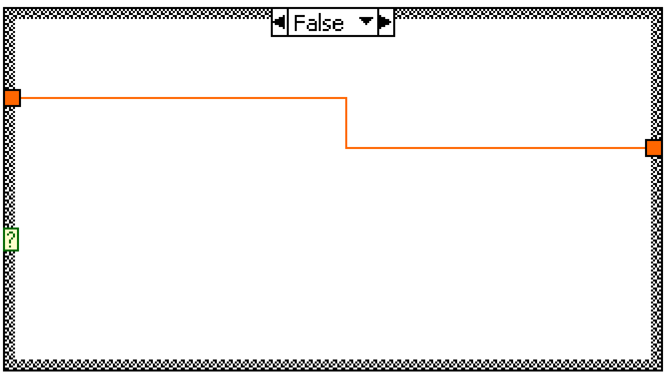


Figure 3.12: Case "False" for Stop Button

3.4 Simulating a Ramp / Part 3

In the 3. part of the software, the user is able to enter several data of irridation into a table. This can be used for simulating a specific ramp which is not measured by a log file.

After the check of the OK buttons, the typed data of the table will be changed to voltages by the reference table and applied to the SubVi "Calculation for Table". The output of this SubVi is applied to the sequence 8 out of 9.

The next figure 3.13 shows the block diagram of the 3. part of the software:

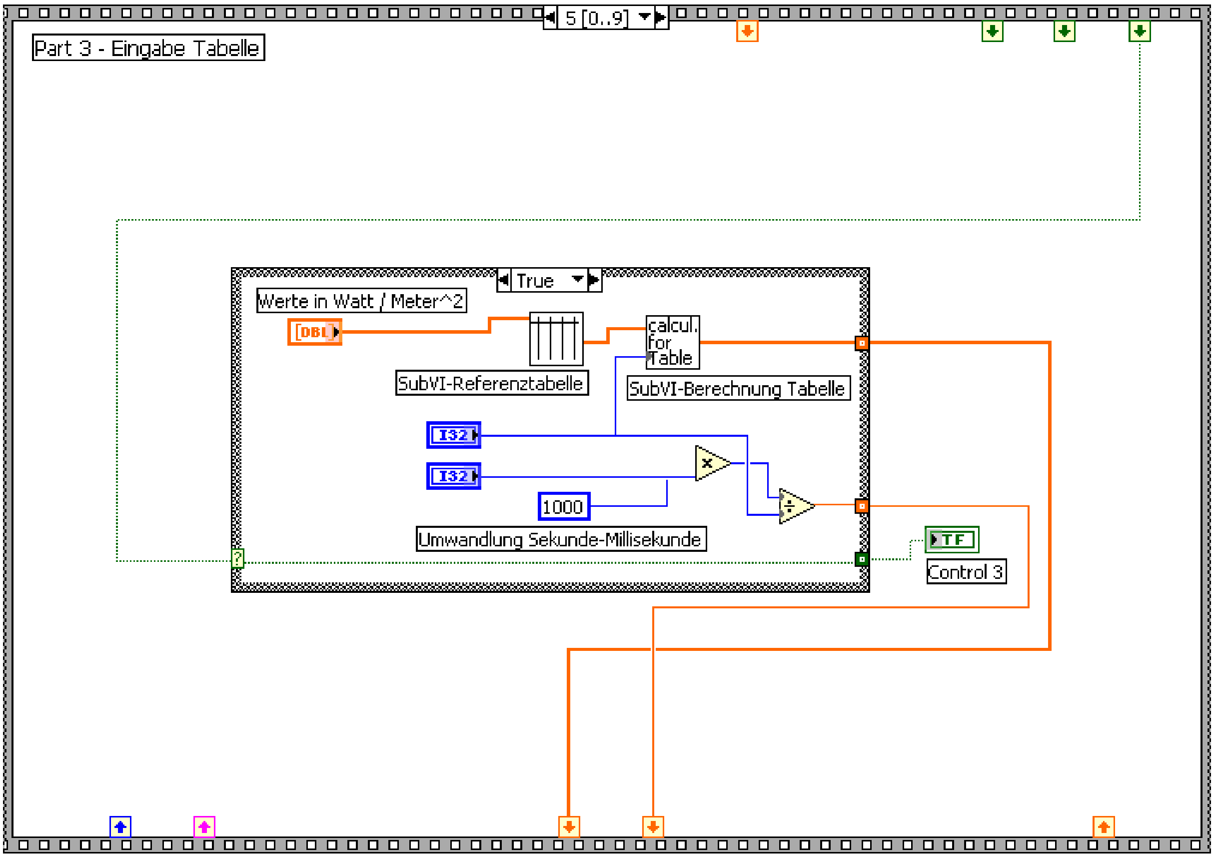


Figure 3.13: Block Diagram of Part 3

Furthermore, the user has to define a time period in which the program should have reached each corner point of the table. Beside this the user has to enter the amount of steps between two corner points. The result of the division of these two factors is used to control the time module of the sequence 8 out of 9.

At the output of part 3, there are also used two indicators for the respresentation of the current values at the analogue output.; numeric indicator and curve chart.

Again, the for-loop is used to read the single values step by step out of the array. The step length is depending on the ratio of the two factors which are entered by the user. The x-scale multiplier is used for the curve chart thus the x-axis is scaled to seconds.

The output of part 3 is illustrated in the block diagram of figure 3.14:

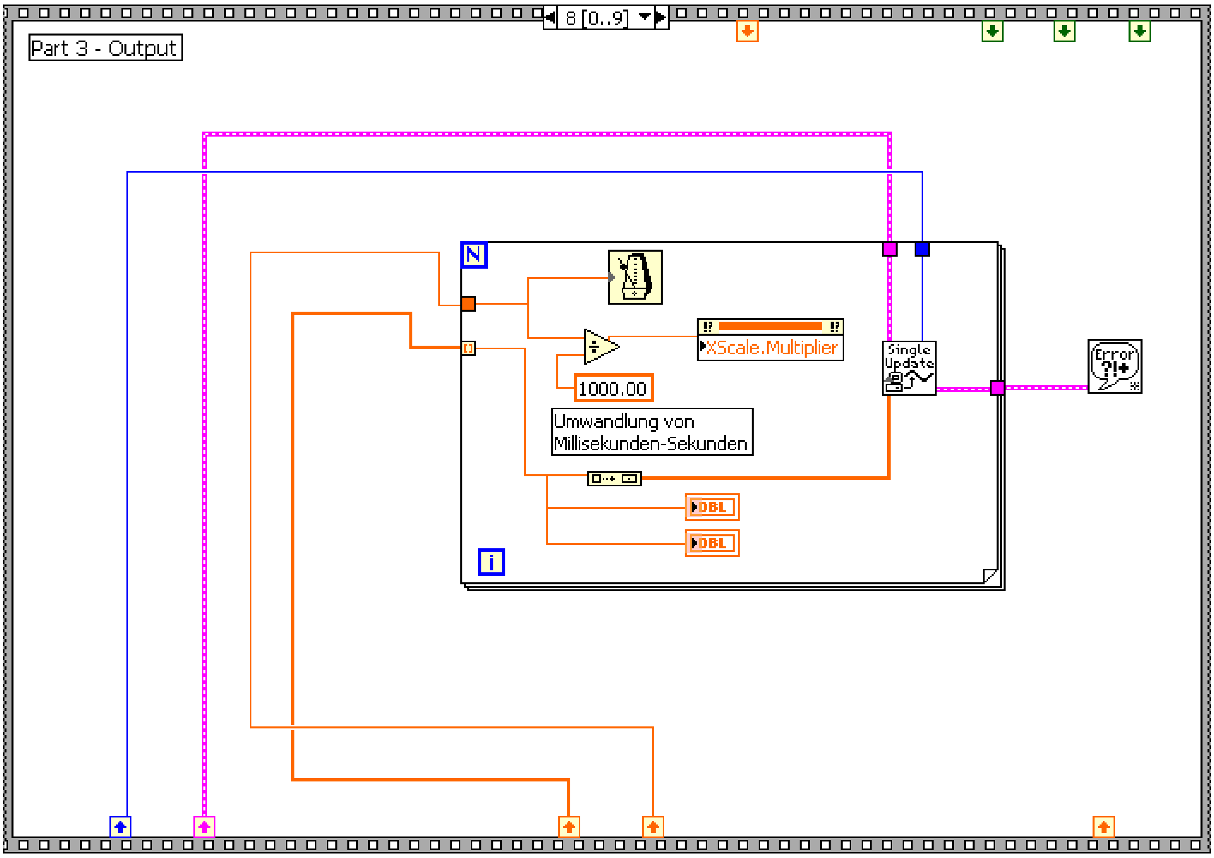


Figure 3.14: Block Diagram of the Output of Part 3

3.4.1 Calculation for Table

This calculation for the table is very similar two the calculation which is mentioned in chapter 3.2.4. The only difference is, that no different locations has to be considered. The amount of steps between two corner points which has been entered by the user is applied to the multiplier and to the for-loop. While the index of the for-loop is increasing one by one, the interpolation SubVi within the for-loop calculates the steps between a value x and a value y. These two values are taken from the table which the user has entered. x and y are corner points. The for-loop takes the data out of the array untill it has reached the end which is the length of the array multiplied by the intermediate steps. At the right side of the for-loop, the new array with the intermediate steps is available for further usage.

The figure 3.15 shows the SubVi "Calculation for Table":

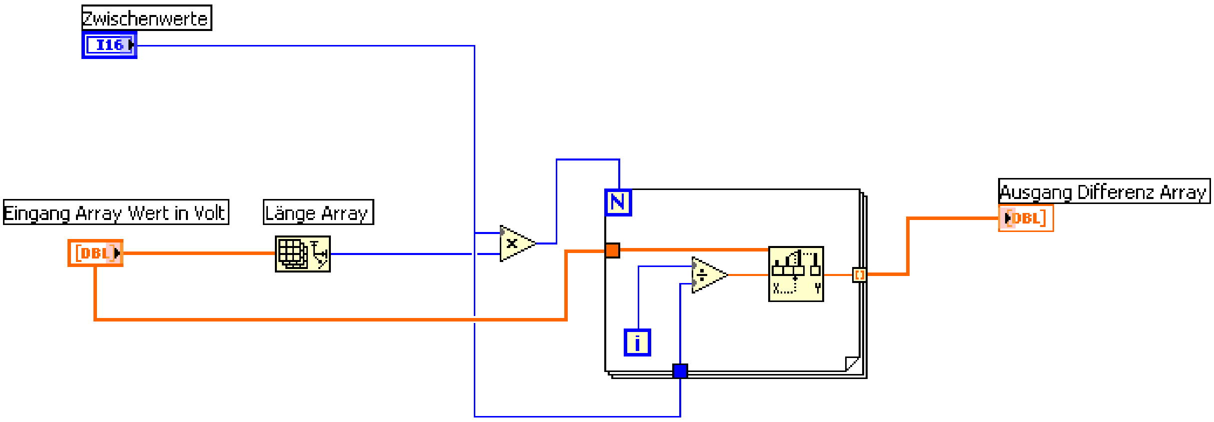


Figure 3.15: Calculation for Table

3.5 Status Request for OK Buttons

This SubVi is created to check if only one OK button is pushed by the user. The program should only start if only one OK button is pushed. Furthermore, each part of the software contains an OK button, placed at the front panel. If the OK button of part 1 is pushed, part 1 should be executed, and so on.

The sequence 2 out of 9 contains the connections between the three OK buttons and two SubVis "Fehlermeldung" and "OK-Buttons".

In addition to that, if the part 3 of the software wants to be executed, the stop button of part 2 has to be set to "True". Otherwise, an error occurs and the part 3 of the software will not be executed at all.

For that reason, a case-structure is used behind the OK button 3. The boolean value will be applied to the structure if the OK button 3 is pushed. The stop button is set to "True". If the OK button 3 is not pushed which means the boolean value is "False", nothing will happen, as it can be seen in the figure 3.16 below:

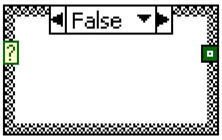


Figure 3.16: Setting the Stop Button

The next figure 3.17 shows the status request of the OK buttons.:

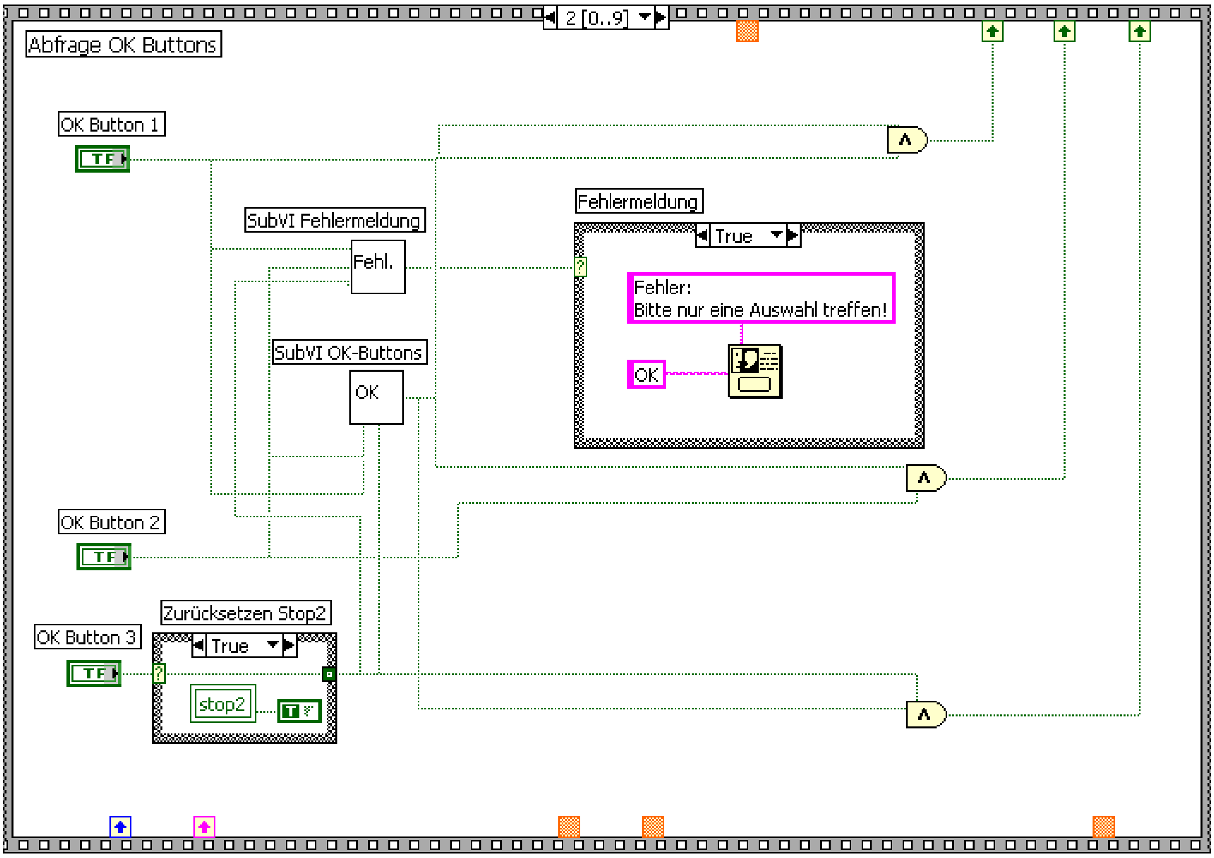


Figure 3.17: Status Request of OK buttons

3.5.1 SubVi OK Buttons

A truth table shows the three OK buttons, represented as A,B and C and the result, which indicates what should happen if the buttons are pushed in all different combination.

A	B	C	Result
0	0	0	0
0	0	1	1

0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Table 1: Truth Table for OK buttons

The value "1" means that the button is pushed and the value "0" means not pushed. In the column "Result" is shown, when the output of the SubVi "OK-Buttons" has the boolean values "1" or "0". If the value is "1", the "1" is applied to an AND gate of each OK button. This AND gate decides which OK button is pushed and executes the corresponding part of the software.

Out of the truth table, you get the result for the truth function and therefore the SubVi "OK Buttons" which is shown as a block diagram in the below figure 3.18:

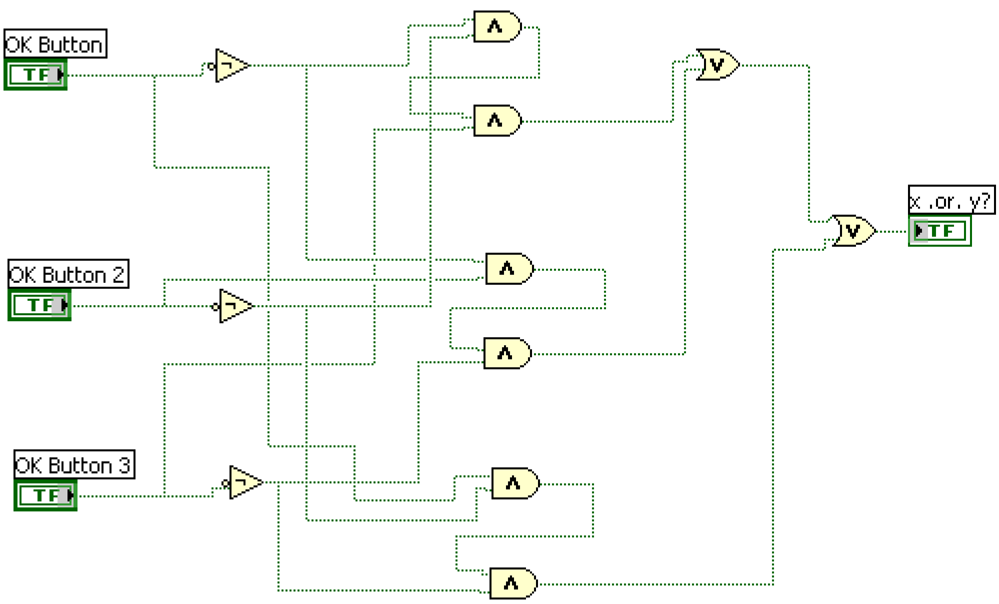


Figure 3.18: Block Diagram of OK Buttons

3.5.2 SubVI Error Message

The base to create this SubVi is as well a truth table as in the previous chapter.

A	B	C	Result
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 2: Truth Table for Error Message

As well as in the Table 1, the three buttons are mentioned as A,B and C. The boolean value "1" appears in the column "Result" if at least two OK buttons are pressed at the same time. This case causes an error message in the sequence 2 out of 9, which is shown in the figure 3.17. The message requests the user only to press one OK button at all.

The block diagram of the SubVi for the error message is illustrated in the following figure 3.19:

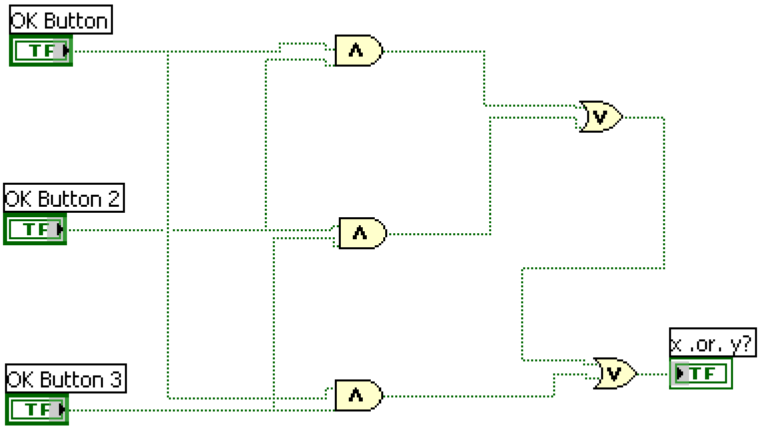
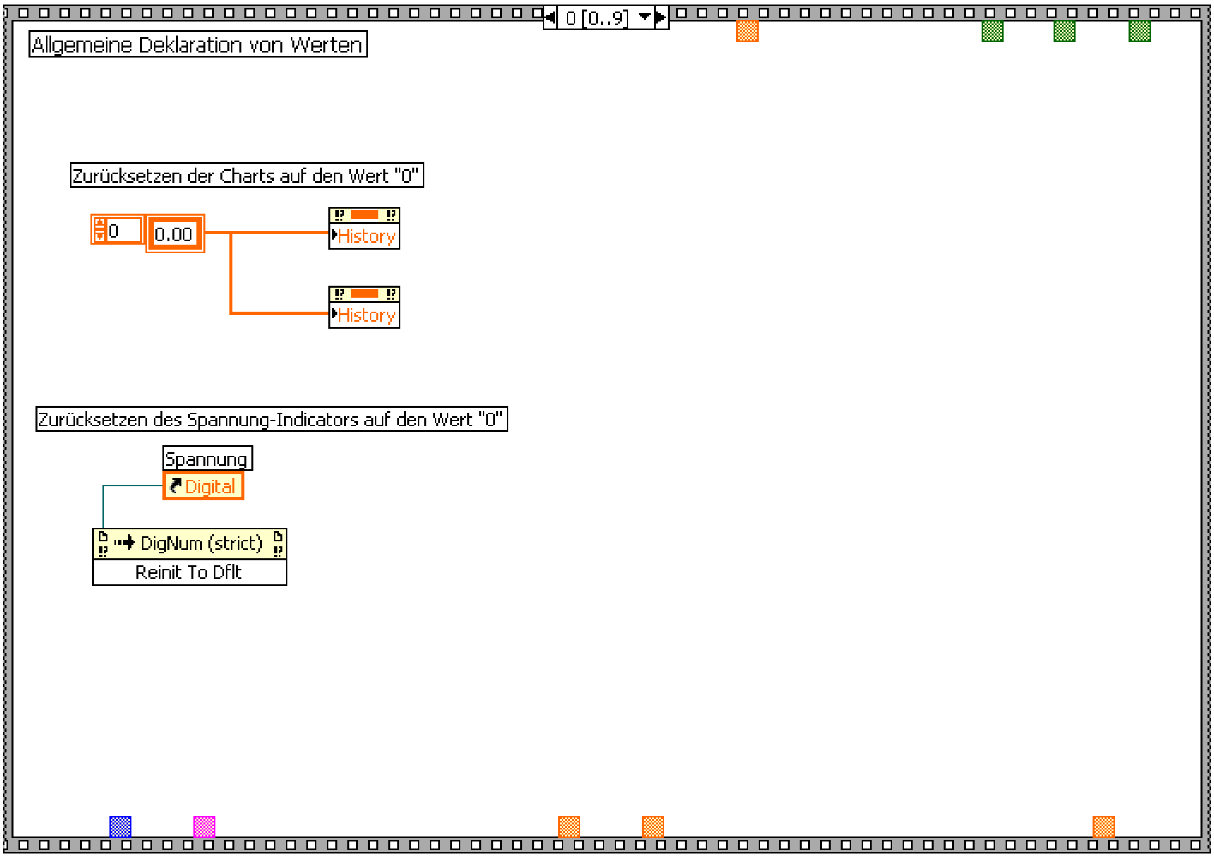
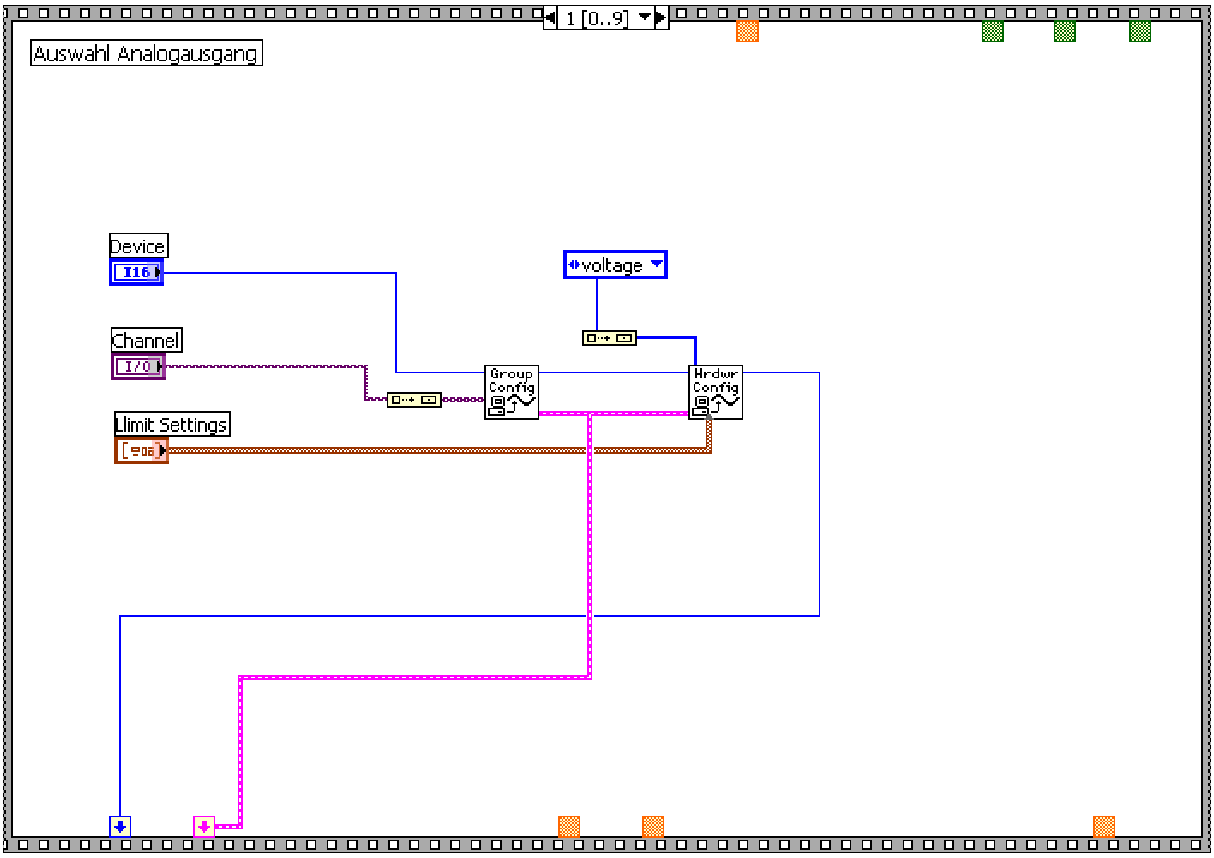
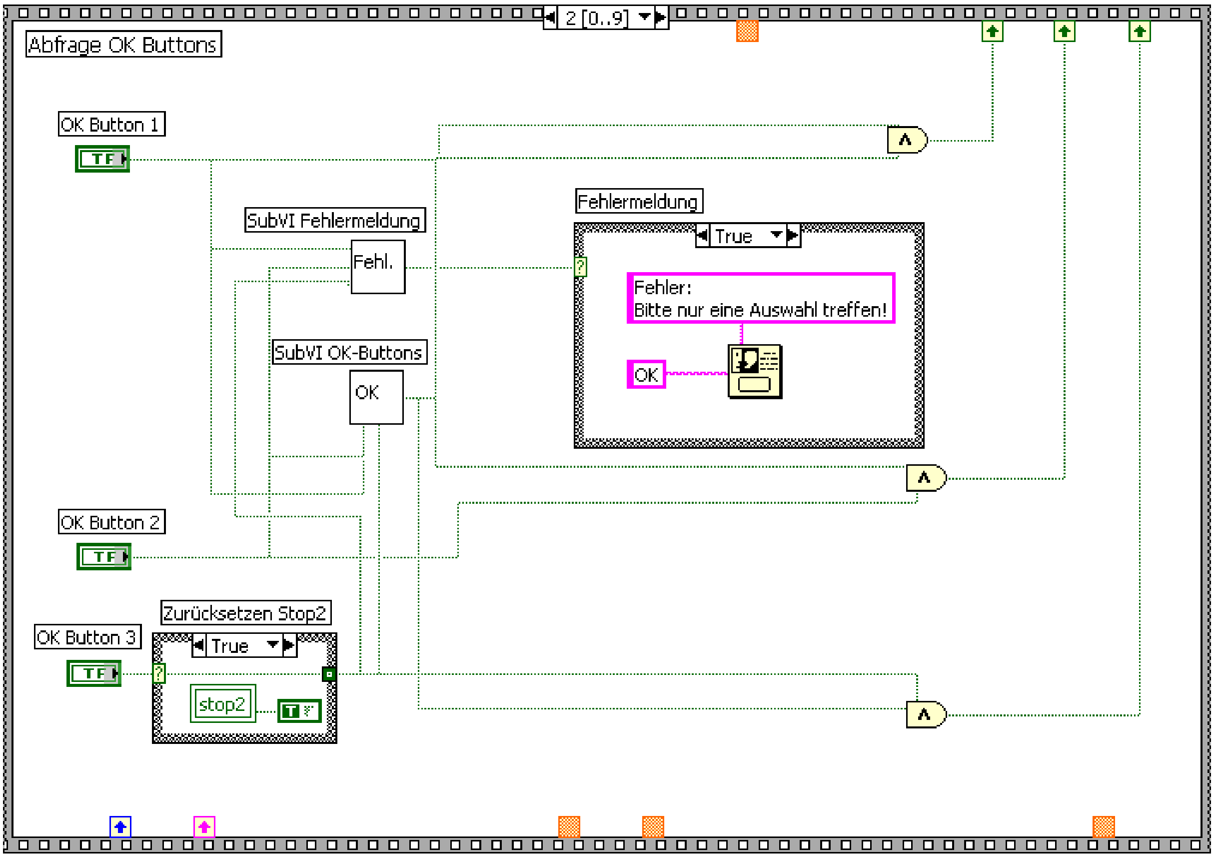


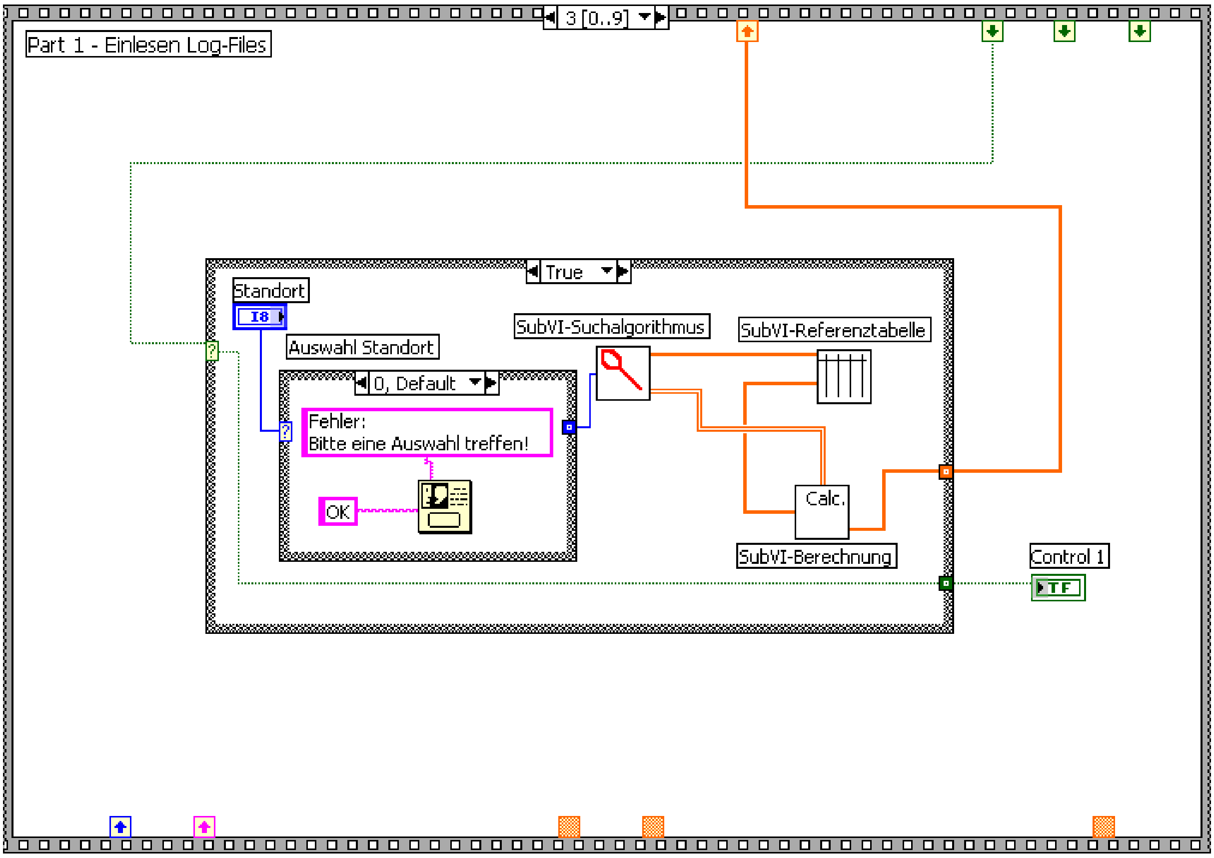
Figure 3.19: Block Diagram of Error Message

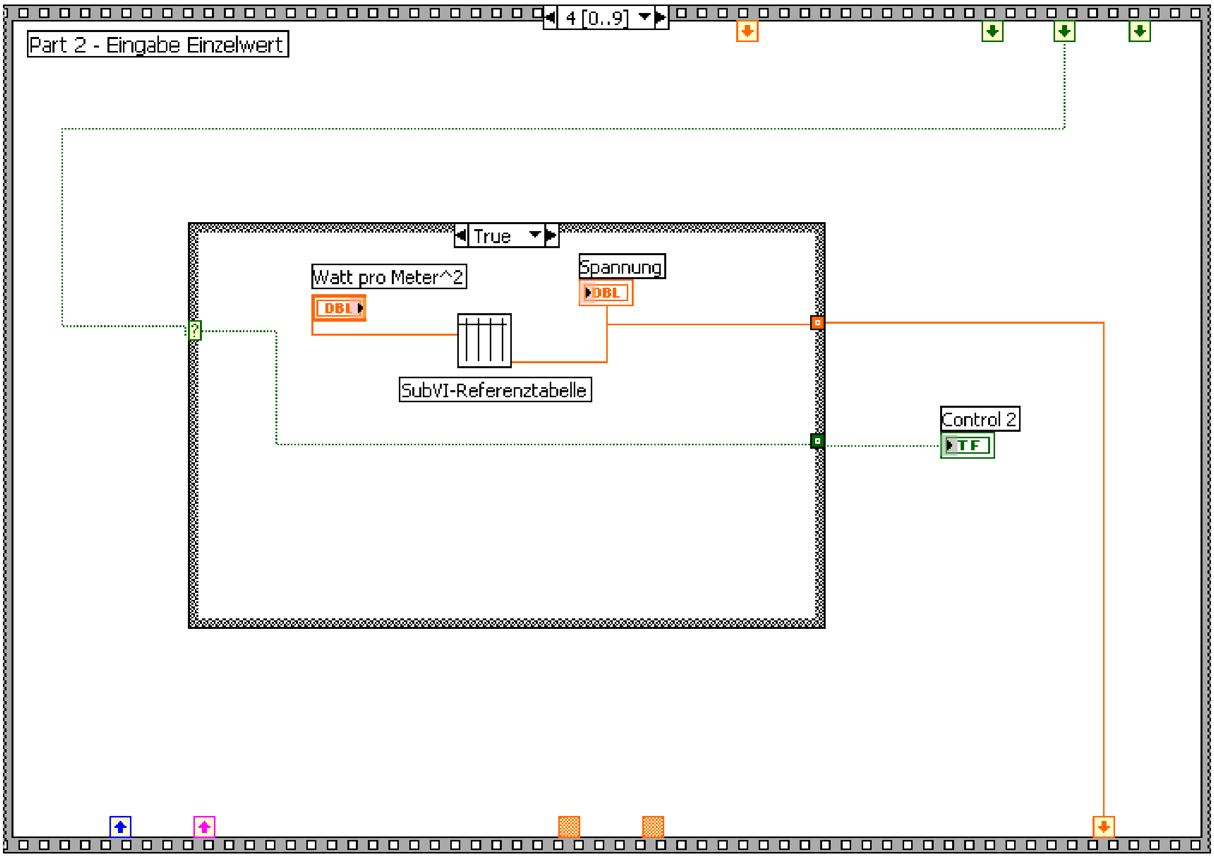
Appendix A / Main Program - Sequences

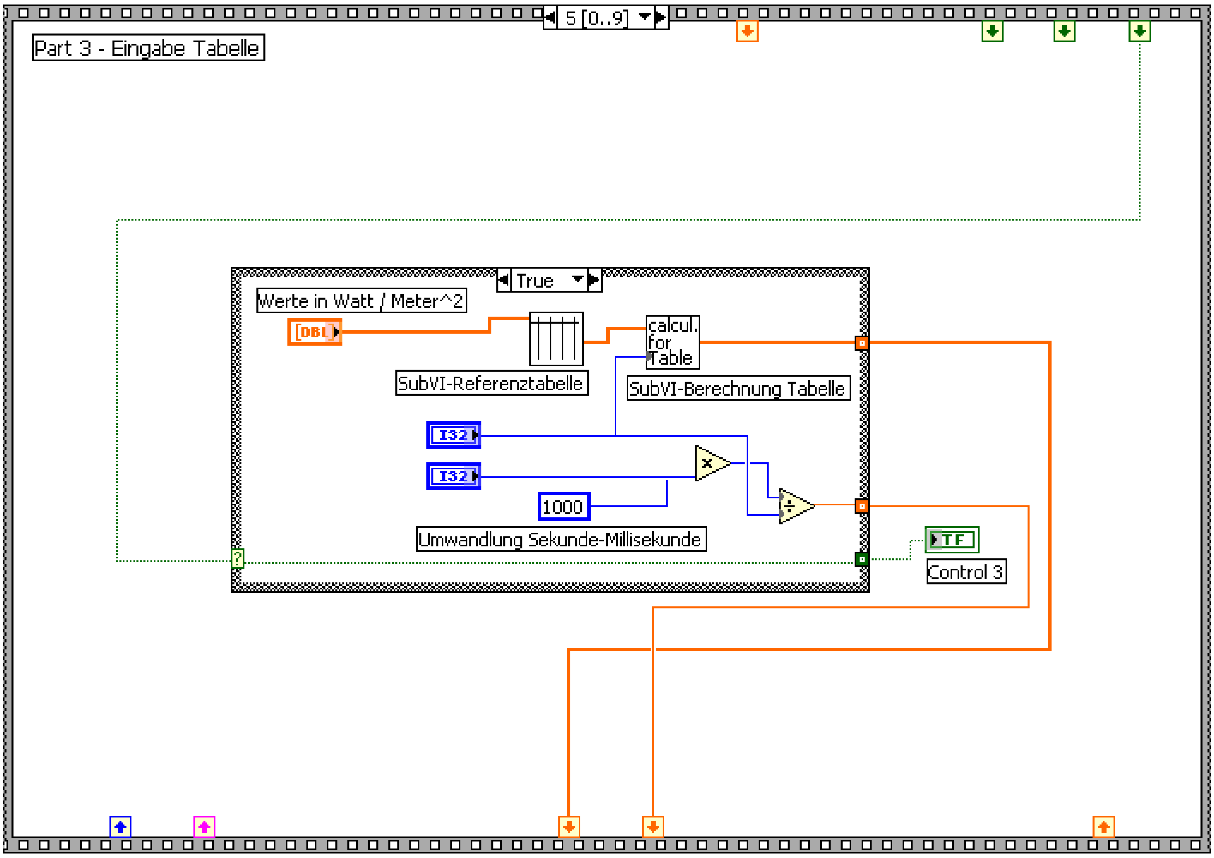




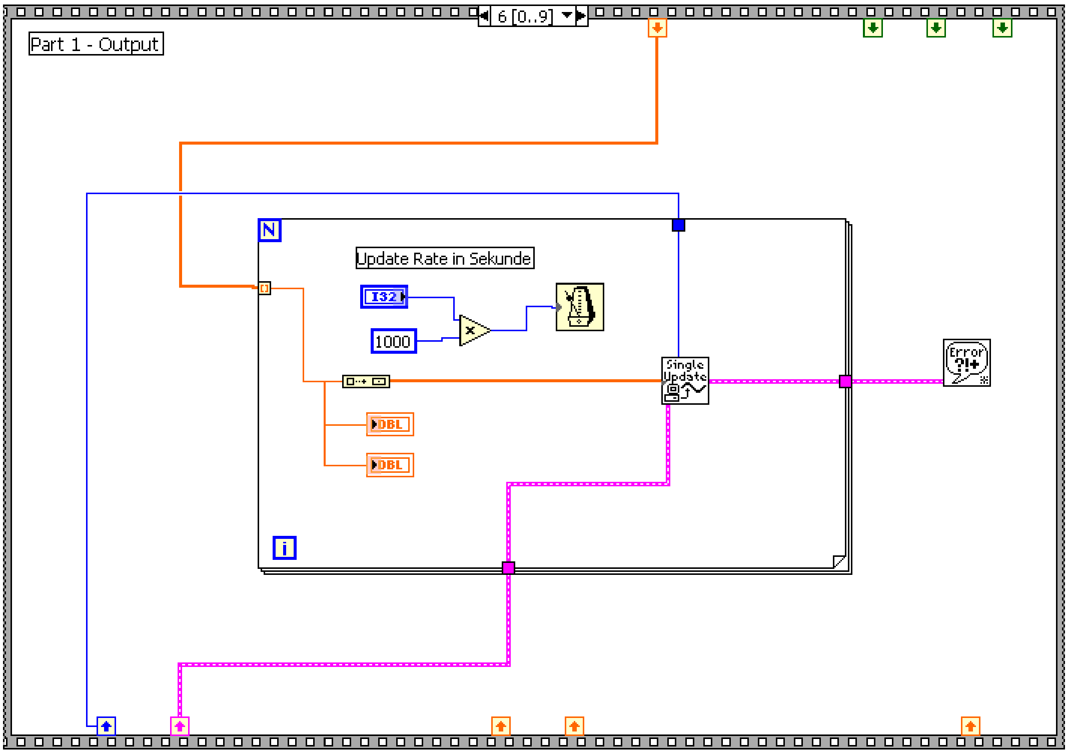

2 [0..9]
Abfrage OK Buttons
OK Button 1
SubVI Fehlermeldung
Fehl.
Fehlermeldung
True
Fehler:
Bitte nur eine Auswahl treffen!
OK
SubVI OK-Buttons
OK
OK Button 2
Zurücksetzen Stop2
OK Button 3
True
stop2

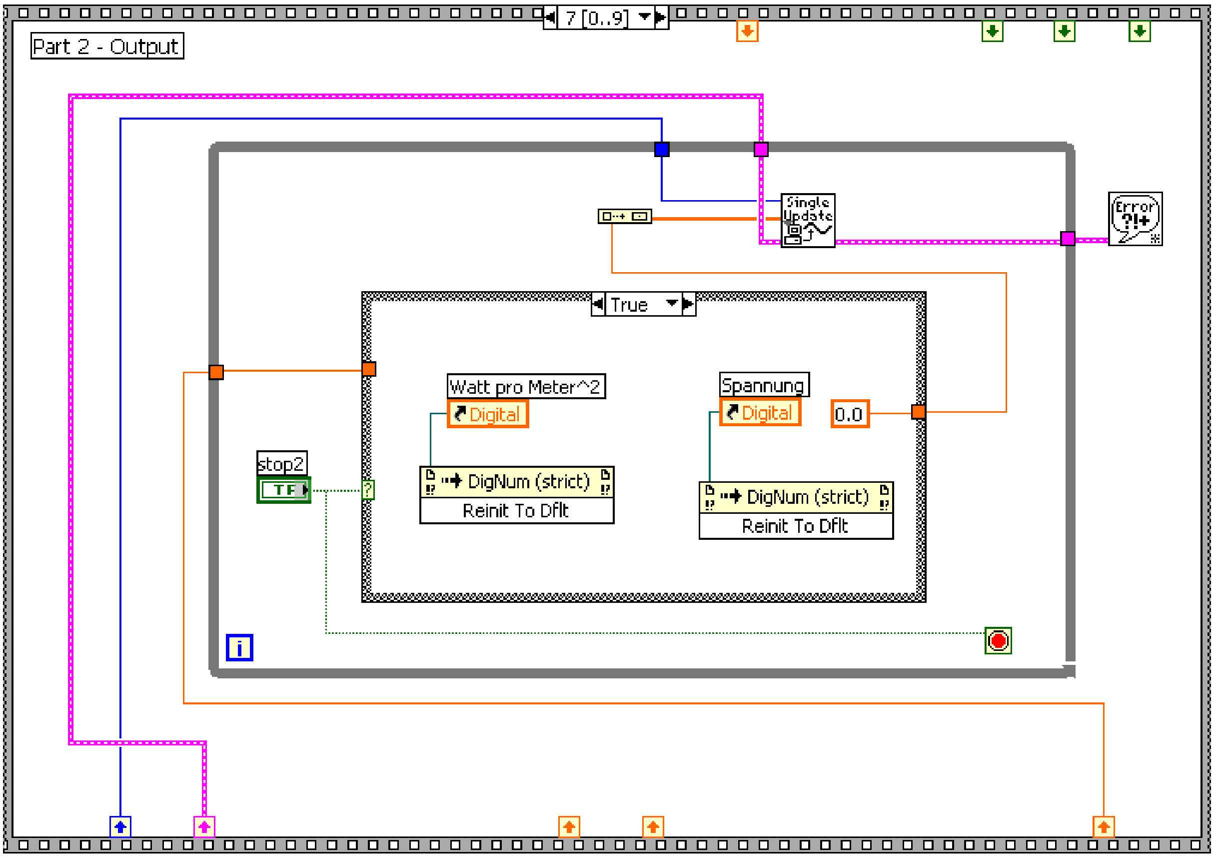

3 [0..9]
Part 1 - Einlesen Log-Files
True
Standort
I8
Auswahl Standort
0, Default
Fehler:
Bitte eine Auswahl treffen!
OK
SubVI-Suchalgorithmus
SubVI-Referenztabelle
Calc.
SubVI-Berechnung
Control 1

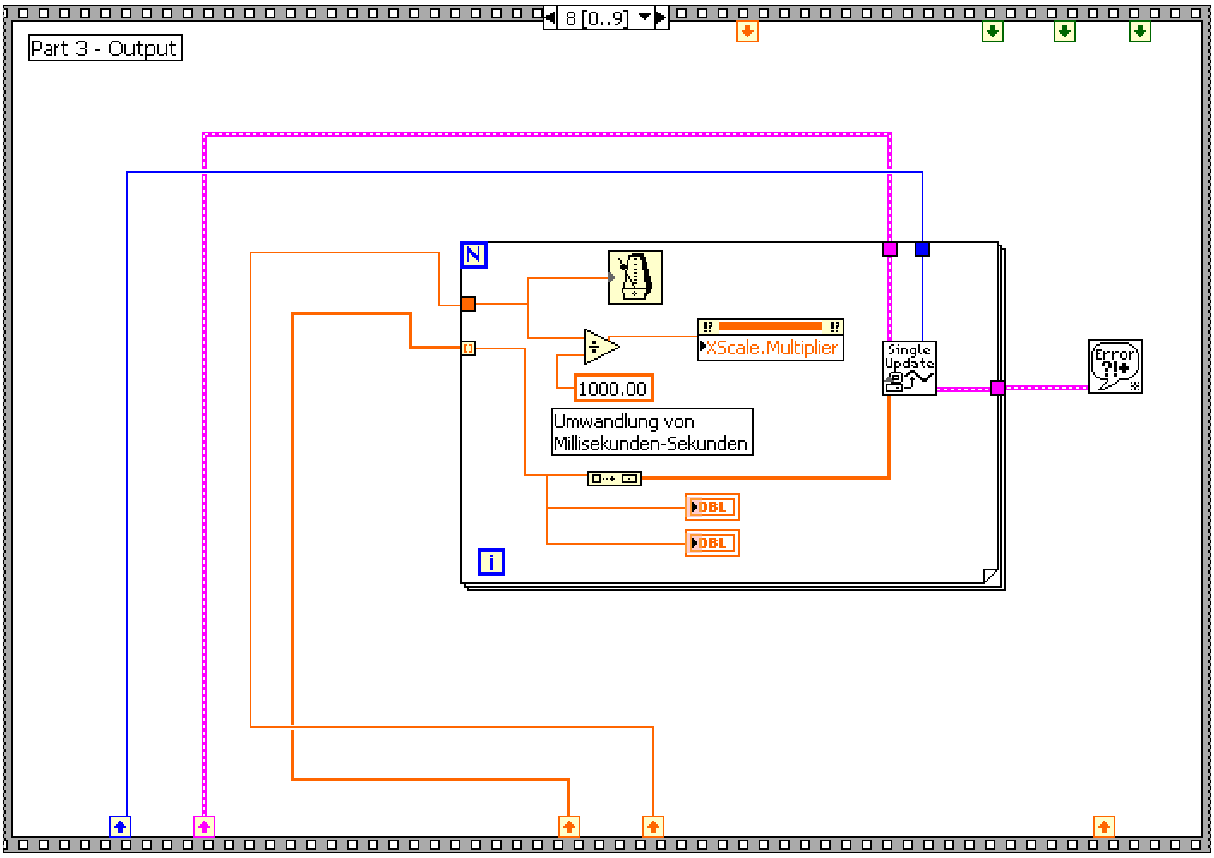

4 [0..9]
Part 2 - Eingabe Einzelwert
True
Watt pro Meter^2
DBL
Spannung
DBL
SubVI-Referenztabelle
Control 2
TF


5 [0..9]
Part 3 - Eingabe Tabelle
True
Werte in Watt / Meter^2
DBL
SubVI-Referenztabelle
calcul. for Table
SubVI-Berechnung Tabelle
I32
I32
1000
Umwandlung Sekunde-Millisekunde
TF
Control 3

by Tobias Kiefer
Solarthermie-2000
August 2003

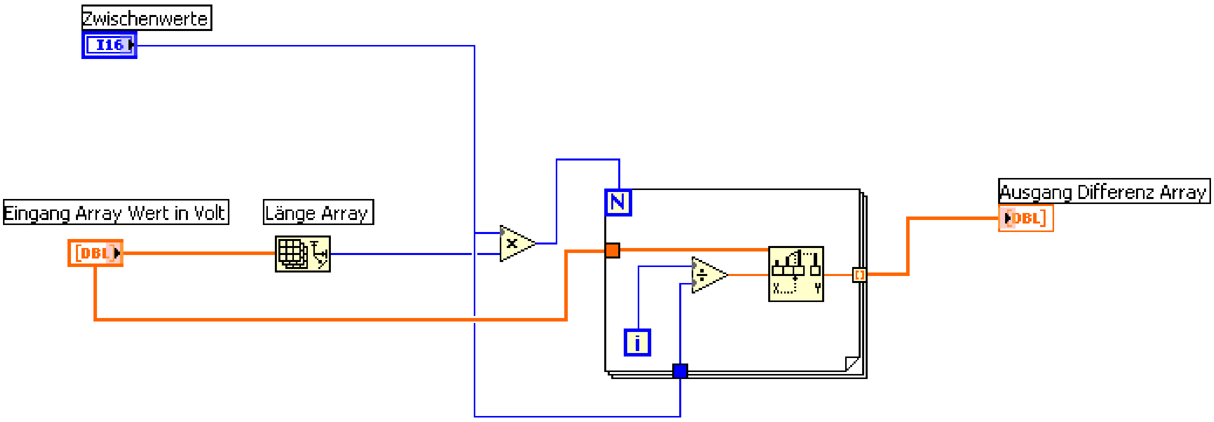




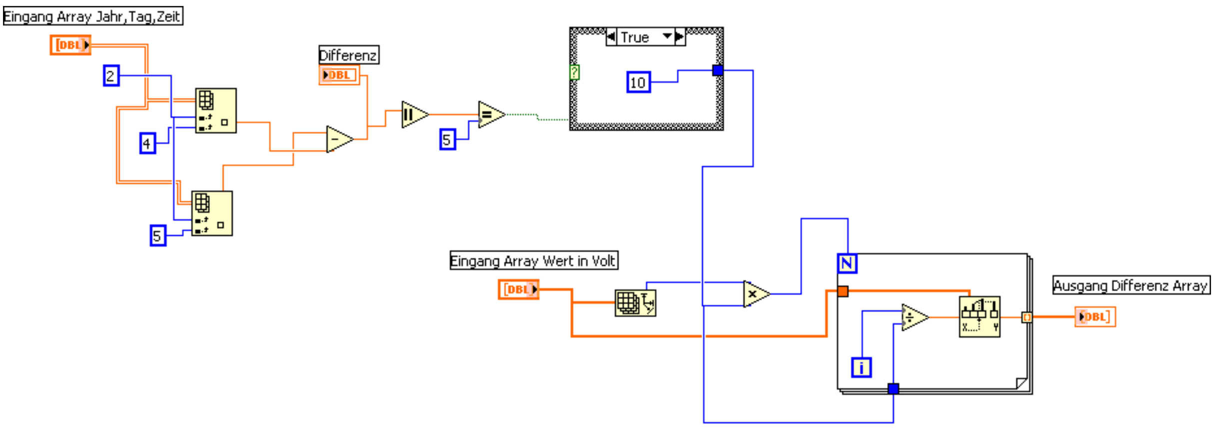

8 [0..9]
Part 3 - Output
N
XScale.Multiplier
1000.00
Umwandlung von
Millisekunden-Sekunden
Single
Update
DBL
DBL
i
Error
?!+

Appendix B / SubVis

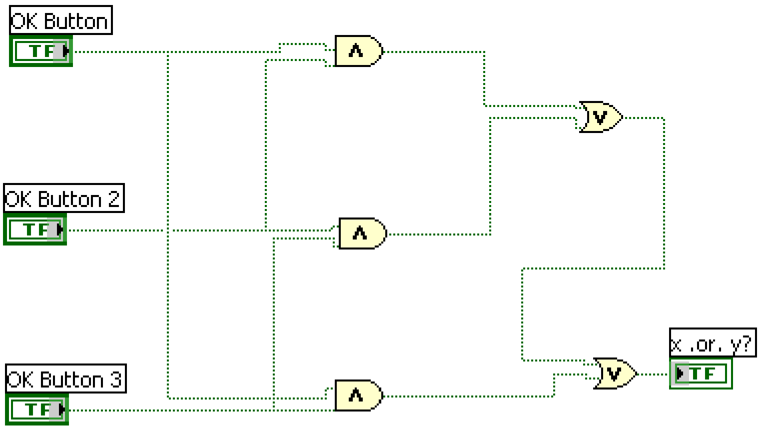
Calculation for the Table:



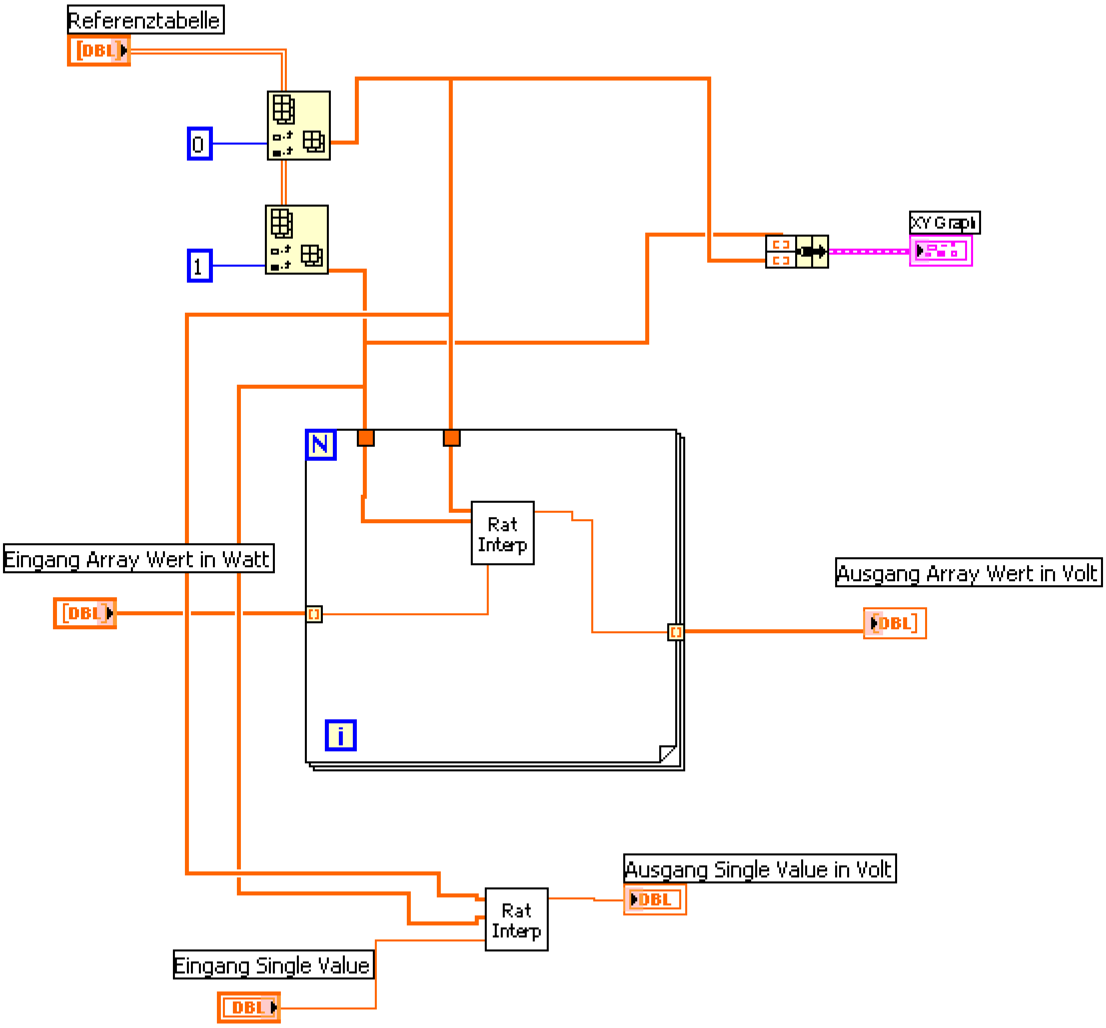
Calculation:



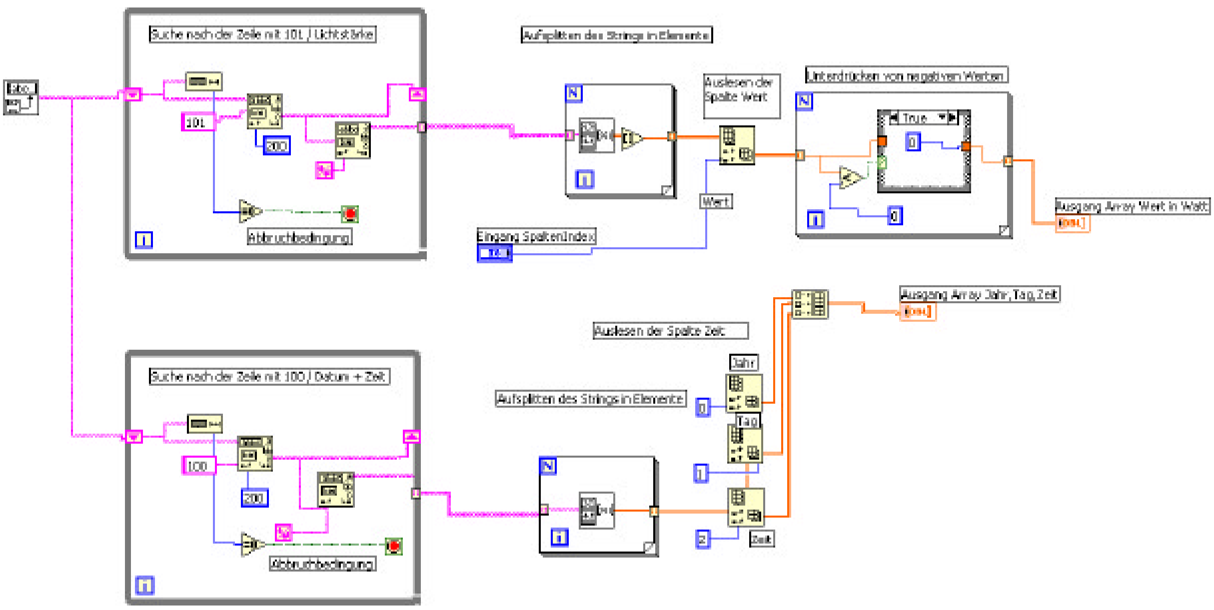
Error Message:



Reference Table:



Search Algortihm:



OK Butons: