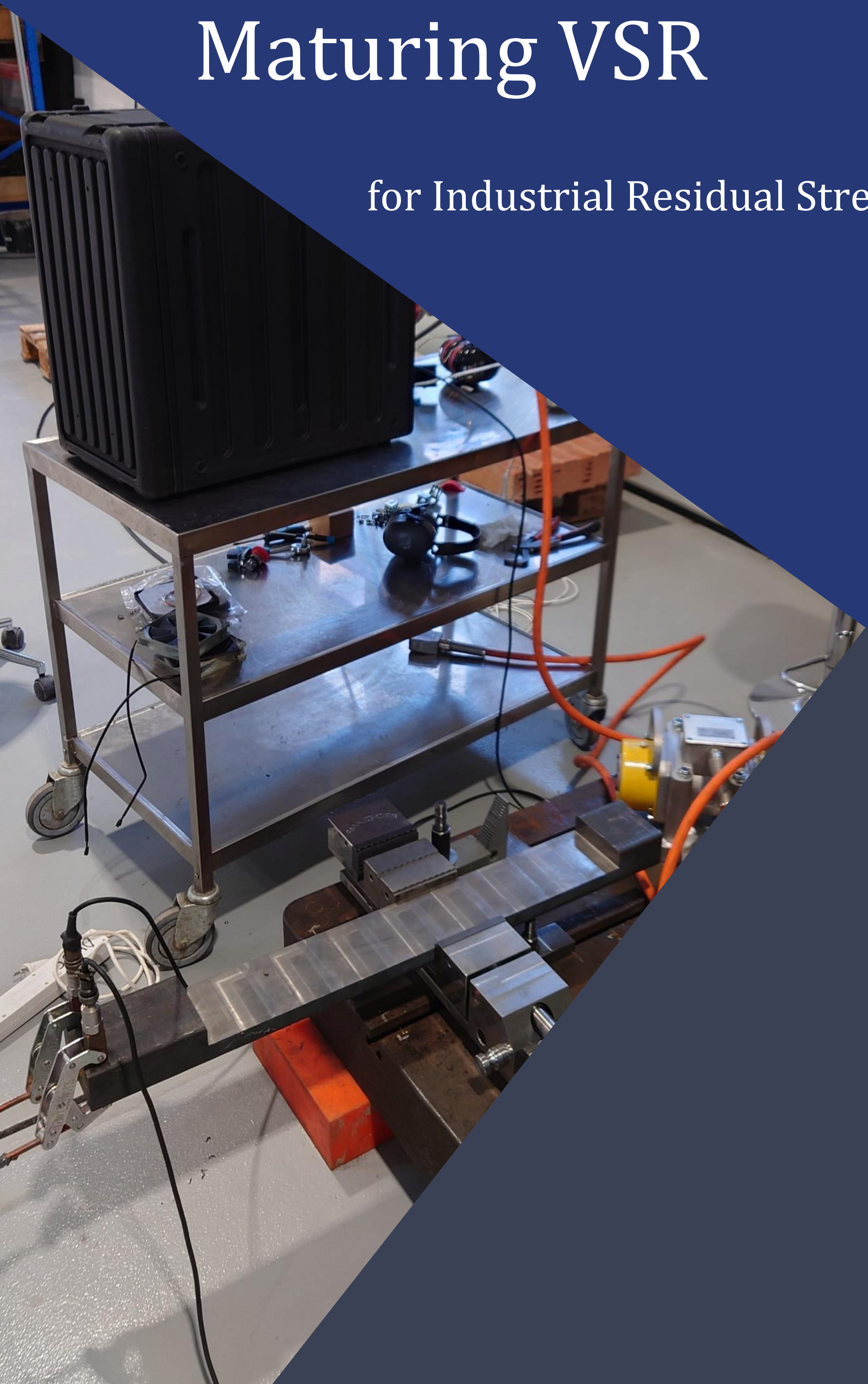
Maturing VSR

for Industrial Residual Stress Relief



Contents

Funding:

INDUSTRIENS FOND

1 Executive Summary

Vibratory Stress Relief (VSR) is an environmentally friendly alternative to Thermal Stress Relief (TSR), offering significant energy savings and cost reductions for stress relief in materials. Despite its proven efficacy, VSR has faced limited adoption in industry due to the technical expertise required for its operation. Recognizing this challenge, DAMRC set out to develop low-cost, user-friendly VSR equipment that would enable broader implementation in industrial environments, in the maturing VSR project.

Building on its extensive experience with VSR since 2014, including insights from the Elforsk VSR project, DAMRC designed and assembled a new system named iVSR. This system incorporates reused components from previous projects, a newly developed electrical assembly, and a user-friendly interface. Leveraging C# programming and a WPF-based user interface, the iVSR system can operate via a commercial computer over Ethernet, streamlining VSR treatments and enhancing standardization and documentation.

The iVSR system was tested with both skilled and unskilled operators, demonstrating its intuitive design and ease of use. In just 30 minutes, unskilled users gained the ability to operate the system effectively. Skilled operators at DAMRC have also validated the system's advanced functionalities, appreciating its improved data collection and usability.

This project highlights the potential for VSR to reduce the carbon footprint of stress relief processes while offering Danish industries an innovative, cost-effective solution. By simplifying operation and improving data capabilities, the iVSR system makes VSR more accessible, paving the way for broader industrial adoption and future research advancements.

2 Introduction

Vibratory Stress Relief (VSR) is a method for relieving residual stresses in materials. It is a greener alternative to Thermal Stress Relief (TSR), as it requires significantly less power to treat an item. To further the green transition while providing cost reduction, implementing VSR in industry would be a significant benefit.

DAMRC has been working with VSR since 2014, where through multiple projects it has been showed that the equipment is more appropriate in a laboratory setting than for industrial use. It requires technical expertise from the user, wherein few companies have the capacity to implement it because of this requirement.

To alleviate this problem DAMRC is set to develop low cost VSR equipment which would be significantly easier to implement in an industrial environment. Herein the previous VSR projects (primarily Elforsk VSR) will be investigated, and some components reused from it. The hardware is then designed and assembled for safe use. This also requires a User Interface (UI) which is user friendly, i.e. intuitive to use for non-skilled operator and safety measures when in use.

3 Pre-analysis

3.1 Investigation of DAMRC's Previous VSR Projects

From the investigation cases for 16+ different companies since 2016 were found. In total 144 treatments were run (one part may have multiple treatments). Herein the documentation varied, but some key parameters were identified as used in the reports: material, treatment RPM, treatment g's, motor imbalance, treatment time, pre-scan peak g's and corresponding RPM, post-scan g's and corresponding RPM, graphs and pictures. From this data the treatment RPM's were between 3400 and 7800 RPM, with amplitudes ranging from 0.16 to 33 g's, with a treatment time between 20-105 minutes. Note that the standard DAMRC VSR equipment is limited to 3000-8000 RPM and 0-50 g's.

Generally, a VSR treatment is set to the natural frequency of the part, however some were not (thereby subharmonic), wherein it was not noted or accounted for in the treatment. VSR treatment changes the natural frequency, which is how the treatment is documented. Most commonly it lowers, however sometimes it was higher, and some cases saw no changes. The range of change from the data was from -350 to 1701 with an RMS change of 268 RPM with a standard deviation of 280 RPM. This shows the complexity of tuning the controller when doing treatments, as it requires incremental positive or negative adjustments of the RPM during treatments. Another observation in the investigation showed that the vibratory Frequency Response Function (FRF) varies widely, and sometime no peak is found within the VSR scanning limits, and often multiple peaks are found, which may confuse as to which frequency should be used.

The size of the parts being treated also range widely, measurements were rarely taken but determining from the pictures the size orders are usually: small 20-50 cm, medium 50-200 cm (pallet size), large 200-1000 cm. Small parts are usually fixtured in vice on a beam to amplify vibrations. Medium parts are either standing on PUR blocks or hanged freely in a harness with the motor mounted on the part. Large parts are considered immoveable, and have the motor mounted on them. The complexity when fixturing occurs due to the vibration mode of the object being highly dependent on the geometry of the part and placement of the motor.

To increase the chances of a successful treatment, multiple treatments (usually 2) are performed. The motor is places in 2 different positions and or orientations. Some parts are also treated in the same operation, to increase the efficiency of the treatment for small parts. However, all of DAMRC's VSR systems only have one sensor, meaning only one part can be analysed, and the others left unknown as to the effect of the treatment. In 2024 DAMRC has implemented ways to simulate VSR treatment to more consistently fixture items and guarantee significant vibrations in the whole part. However, for an un-skilled operator some general guidelines for fixturing may be used.

In summery the following challenges have been found, for an un-skilled operator is to do VSR treatments:

1. The treatment RPM / natural frequency varies widely for different objects
2. The natural frequency changes during treatments
3. No or multiple frequencies may appropriate for treatment
4. The vibration mode of the part changes depending on the object geometry and motor position

3.2 Investigation of safety requirements

VSR systems are electromechanical systems that require knowledge of mechanical construction, electrical components and installation, networking devices, and control programming. The motors used are already CE marked, along with all the electrical devices used in the assembly. Therefore, the most relevant safety requirements are the installation of these components which are given in the device documentation. The relevant Danish laws on electrical are found under "Bekendtgørelse af lov om sikkerhed ved elektriske anlæg, elektriske inssallationer og elektrisk materiel (elsikkerhedsloven)". Here are some relevant paragraphs summarized [1]:

> §33 There must be installed a break from the electrical installation, which makes it possible to break and separate the electrical installation.
>
> §34 When there is a need to immediately break the power to avoid danger, there must be an emergency stop which is easily recognizable, effective and quick to use.

§35 For installation up to and including 20 A, there must be an automatic breaker from the power supply.

§36 There must be appropriate signing for safety concerns.

stk. 3 Wires must be placed or marked so they can be identified.

§39 A protective grounding wire must be color marked with green and yellow.

§75 After inspection of an electrical installation it must be verified that it meets the requirements of the law. Verification takes place before being put into operation.

stk. 2 Verification includes inspection, testing and reporting.

stk. 3 Inspection and testing are done according to HD 60364-6.

§77 Persons who work with or around the installation while powered must be sufficiently qualified to do work, to avoid danger.

4 Hypothesis

DAMRC can develop a user friendly industrially viable VSR system, which have the same core functionalities as other VSR systems: VSR scanning for analysis and quality control, VSR treatment for stress relief obtained at a specific RPM for a given duration, while also allowing documentation of work reports for VSR.

5 Success Criteria

The success criteria are to ind the requirements for maturing VSR for use in industrial environments, for an un-skilled user to be able to do VSR treatments, and etting data and facilitate improved data gathering from VSR treatments for improving and documenting the technology and its viability.

6 Project Scope/ Description

To mature the VSR technology DAMRC knowledge is to be merged and analysed in this report and used to specify the requirements to make the technology more appropriate in industrial applications. Herein the existing components from previous projects are to be assess for their use in a new VSR system.

A new VSR system is to be developed, including hardware assessment, assembly and software for intuitive control of the equipment. This is to demonstrate the viability and challenges of implementing VSR in Danish industry and give a process template for companies to use.

Ultimately this is to further the green transition by reducing energy cost for residual stress relief, while also offering an economical alternative to thermal stress relief.

7 Risk Analysis

Although the mechanism for doing VSR are rather simple, the automation and implementation has shown to be challenging. Within the scope of this project the following challenges have been assessed:

1. Cost for electrical components
    a. Reuse of components from previous projects (primarily the frequency converter and PLC)
2. Flawed electrical assembly
    a. Use of appropriate wiring and safety components
    b. Monitoring of operation using software
    c. Test running assembly iteratively
3. Software bugs leading to errors or damages
    a. Safety measures in software
    b. Hardwired safety stops
4. Operating cost due to licensing
    a. Use of opensource free programming tools and packages

8 VSR Equipment Development

8.1 Introduction

This section detail how VSR works on a theoretical level and provides some practical examples from studies. And details the workings of electromechanical systems used in other VSR systems, to then describe how the industrially matured iVSR system is developed.

8.2 VSR Treatment

Vibratory Stress Relief is a method of stress relief by using harmonic vibrations to induce displacements in the atomic crystal lattice.

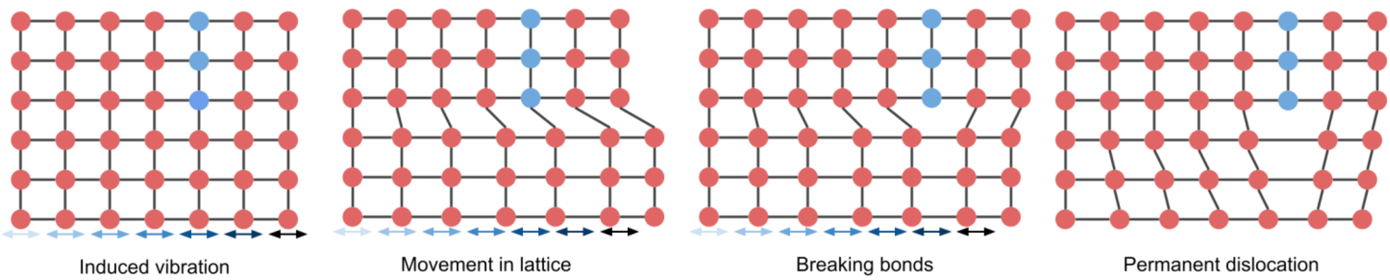


Figure 8-1: Vibrations in atomic lattice.

The induced dislocations are more ductile than a rigid lattice, allowing for internal stresses to be relieved [2]. This has the advantages of reduced power consumption compared to thermal stress relief, being more mobile, and not changing the grain structure of the material (leaving finer grains compared to thermal stress relief) [3].

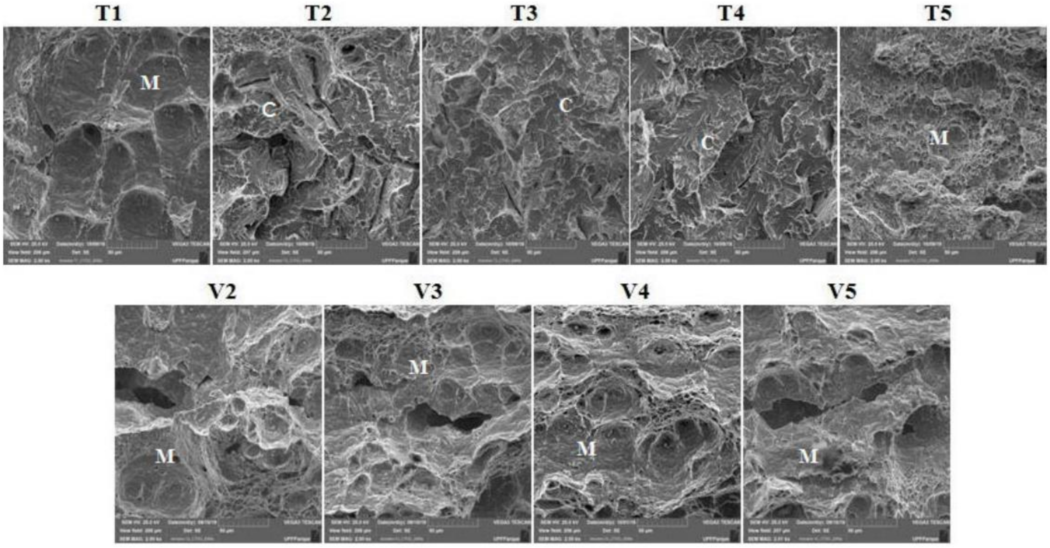


Figure 8-2: Microstructure of thermal (T) stress relief vs vibratory (V) stress relief [4].

DAMRC’s current VSR system includes imbalanced AC motors for generating vibrations. Accelerometers for measuring the amplitude of the vibration. A workstation with a computer for controlling the system (including frequency converter for motor control). It requires many manual actions to run a treatment, which has the following process:

1. Setup the part to freely vibrate.
2. Mount motor and accelerometer to the part.
3. Quick scan for eigenfrequencies in the part (5-30 min).
   a. If none are found, reconfigure the accelerometer and or motor.
4. Pre-scan to find peak eigenfrequencies.
5. Choose appropriate frequency.
6. Manually ramp up the motor frequency until the eigenfrequency is excited.
7. Manually turn off the motor after 30 minutes.
8. Run post-scan to determine changes in the frequency response.
9. Export data by screen shotting the frequency response graphs before and after the treatment as a JPG.

The manual processes result in the need for an operator during the entire treatment. And often lead to user errors for inexperienced operators. The limitations of the JPG format for the work report also means data analysis and process optimisation is challenging. The system is also limited to one sensor input, wherein depending on geometry, some regions of the treated part might not experience proper treatment and making it impossible to document treatment of multiple parts simultaneously. These factors limit the useability in industrial applications.

In a previous project at DAMRC Elforsk VSR (P901), a new VSR system was made to resolve some of these issues. However, the system has problems which will be explored in the following sections, as it will be the basis for these projects new VSR system.

8.3 Elforsk VSR Equipment

8.3.1 Hardware

To determine the viability and need of the Elforsk VSR system, the following hardware components were identified:

Component	Name	Article number	Current function
Programmable logic controller (PLC)	S7-1200C	6ES7 212-1BE40-0XB0	Executing program Communication with PC Input acceleration data
Frequency converter / inverter	G120C	6SL3210-1KE15-8UF2	Driving motor
Signal conditioner	2x SC200	SC207-100A-050GR-020-200-10	Conditioning acceleration analogue inputs

Table 8-1: Elforsk VSR components.

The connectivity is summarized in the following figure:

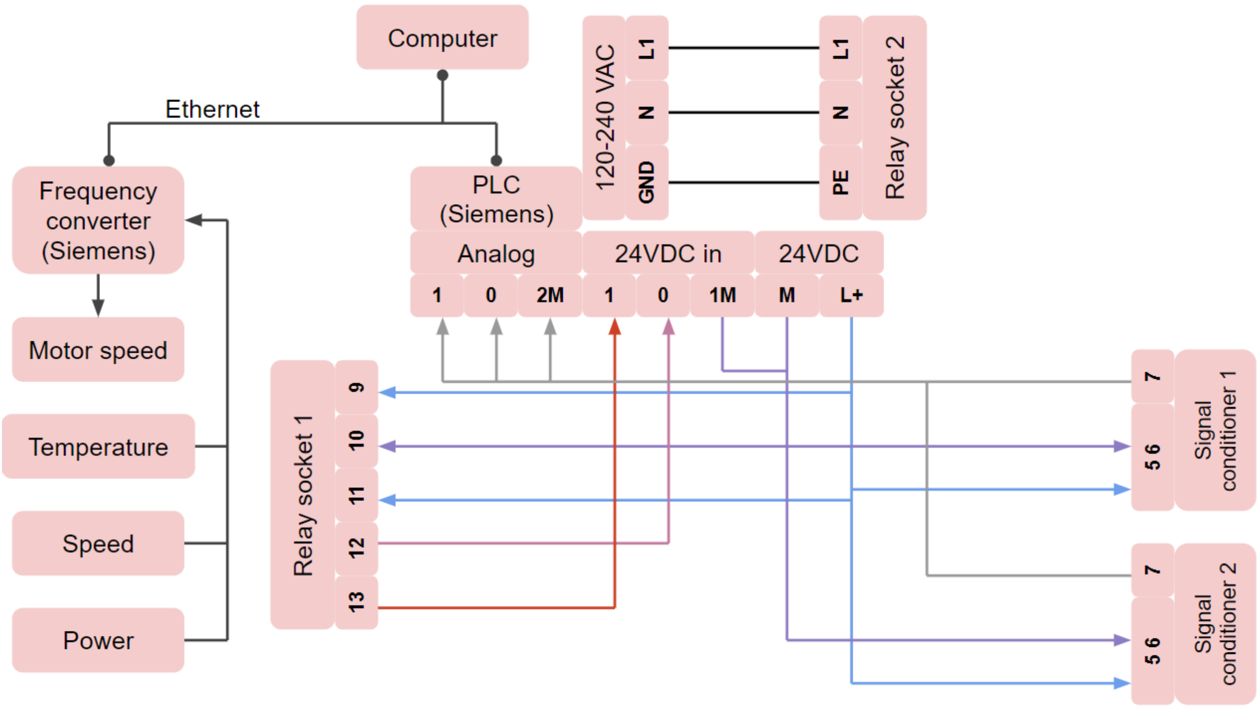


Figure 8-3: Elforsk VSR wiring.

These components do cover the functionalities of a VSR systems hardware. However, there were problems with turning on the frequency converter, which may be due to current surging on startup. As seen in the G120C manual [5] the frequency converter ought to have more electrical components for protection, and supply current conditioning:

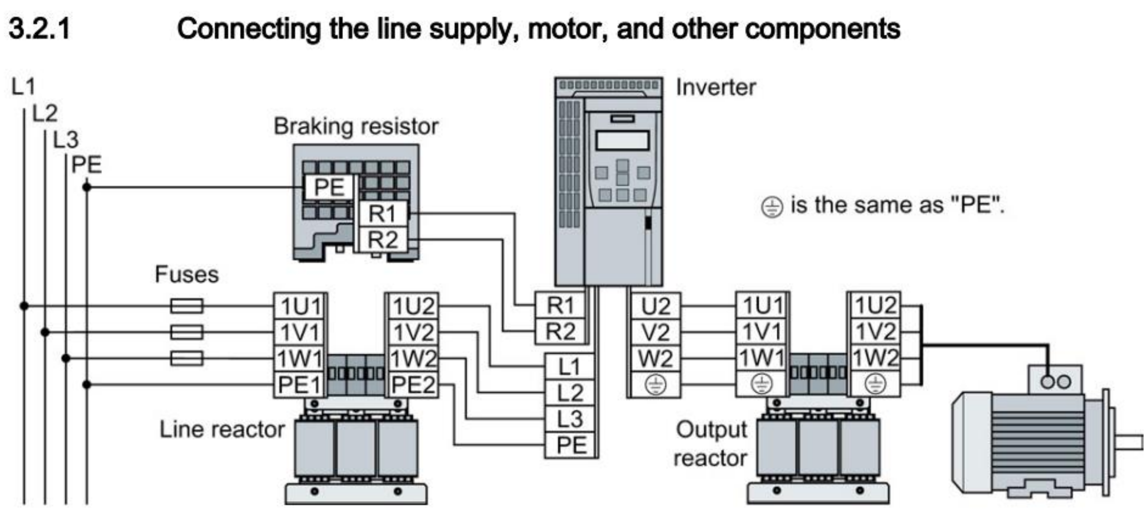


Figure 8-4: Recommended connections and components for G120C [5].

Wherein the line reactor would alleviate the start-up problem. A braking resistor would protect the inverter from feedback current when stopping the motor (a contactor can also be used as seen on some of Advanced VSR's system). The output reactor makes the current voltage more sinusoidal, meaning the motor runs smoother. The general wiring in the system is excessive, as many wires are unused or needlessly long.

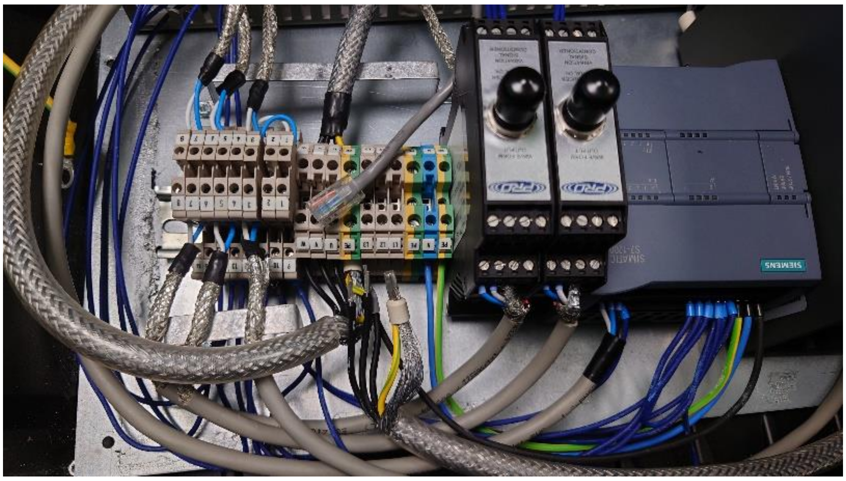


Figure 8-5: Elforsk VSR physical wiring.

The Elforsk VSR system is also designed to run a 3-phase 400V rated for 10A, therefore the inverter uses 3-phase 400V. DAMRC's current VSR motors run single or 3-phase phase 230V.

8.3.2 Software

The Elforsk VSR had software developed externally, wherein only the final product DAMRCVSR.exe was given (no documentation or source code). It uses a STEP7 program downloaded on the PLC, where the user interface (UI) and controls are assumed to have been made in JavaScript with a STEP7 communication library.

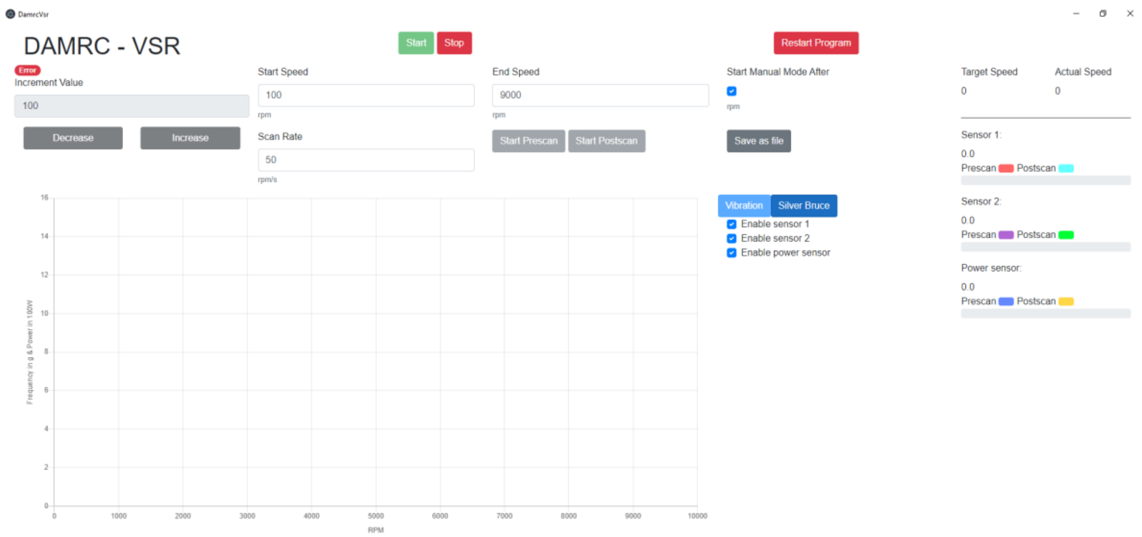


Figure 8-6: DAMRCVSR.exe user interface.

It seems to have the same functionality as other VSR systems, with an additional accelerometer, and the ability to export treatment data. However, it tends to crash, both at startup and while using it. The UI design is also not intuitive for a new user, as there is no guide, buttons are placed non-intuitively, are poorly described and or irresponsive when pressed.

8.4 Remodelled VSR System

8.4.1 Hardware

To improve safety and performance the following components were installed:

Component	Function
400V 10A Line reactor	Reducing surge voltage on startup
140Ω Braking resistor	Protecting driver when braking motor
3x Fuses	Protect components from overcurrent
Load switch	On/off switch
2x fans	Cooling electric components while running

Table 8-2: New components for remodelled VSR.

These are to be assembled according to the following electrical diagram:

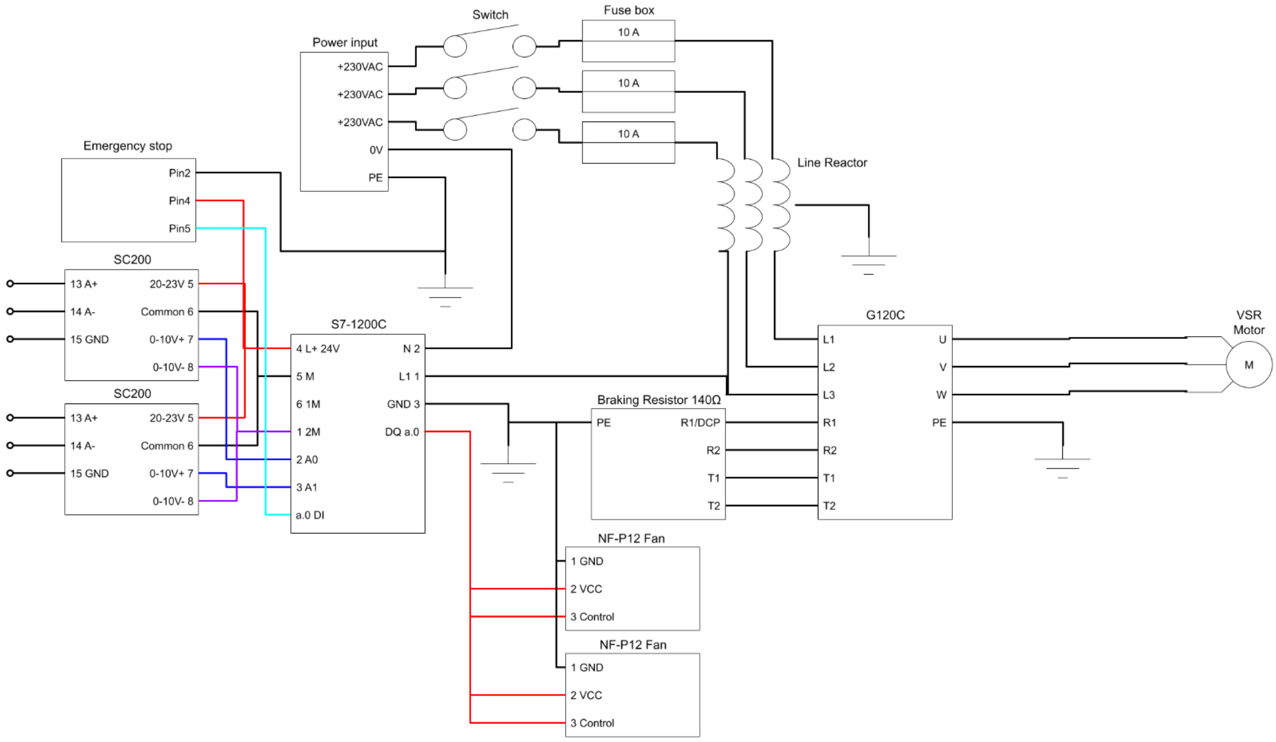


Table 8-3: Electrical diagram for iVSR.

Installing the line reactor allowed the system to start up more consistently.

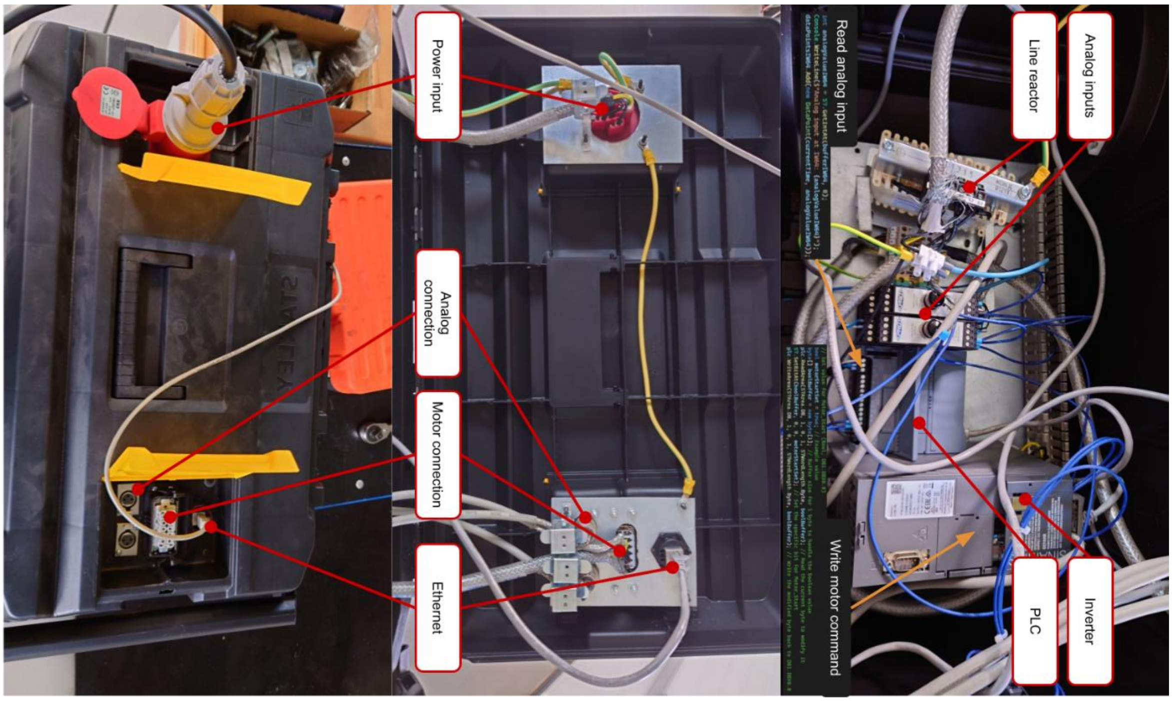


Figure 8-7: Modified Elforsk VSR case.

The system was tested outside of a case to check the running functionality, as could not fit in the old case:

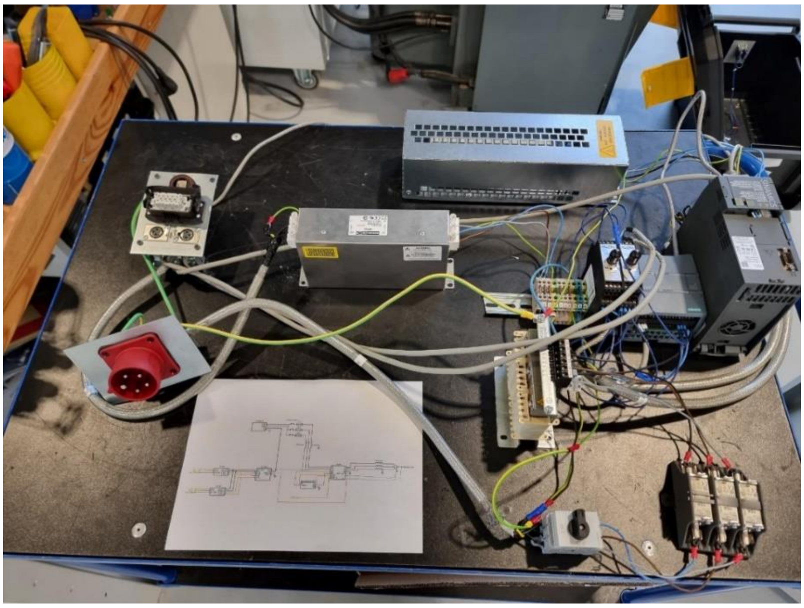


Figure 8-8: Reassembled VSR setup on table.

The braking resistor performed well, as the motor would stop in 10s or less. The components were installed in a bigger case, mounted on a steel sheet, that can slide into the case. The plugs, button and switch are mounted on the top panel made of 3D printed ESD protective PC and PETG. The fans are mounted on the second panel which is made of the ESD protective nylon, thereby protecting the common user from touching electrically hazardous surfaces, while allowing for maintenance and repair by technical personnel.

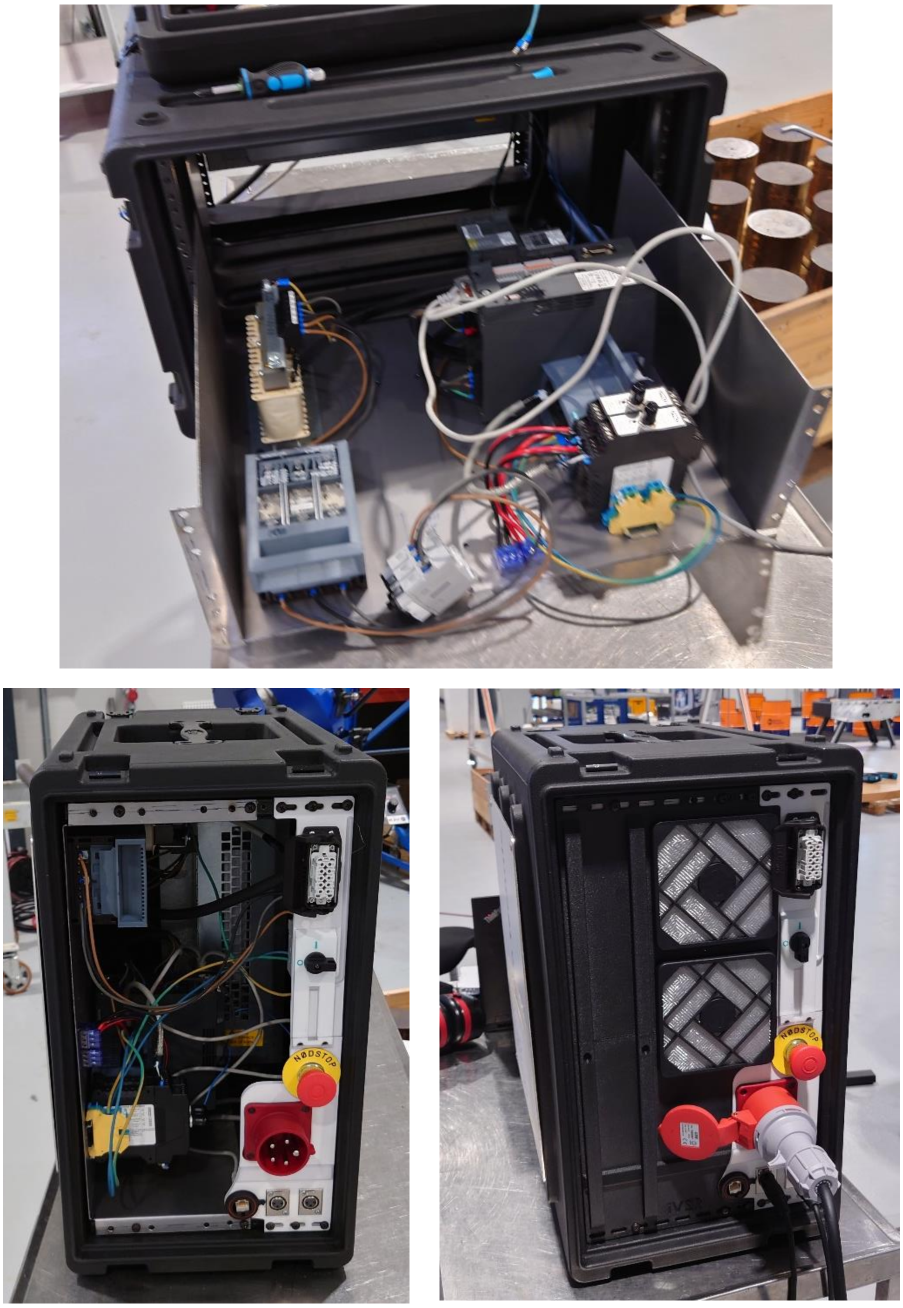


Figure 8-9: VSR electrical installation.

For electrical assembly the following diagram is used:

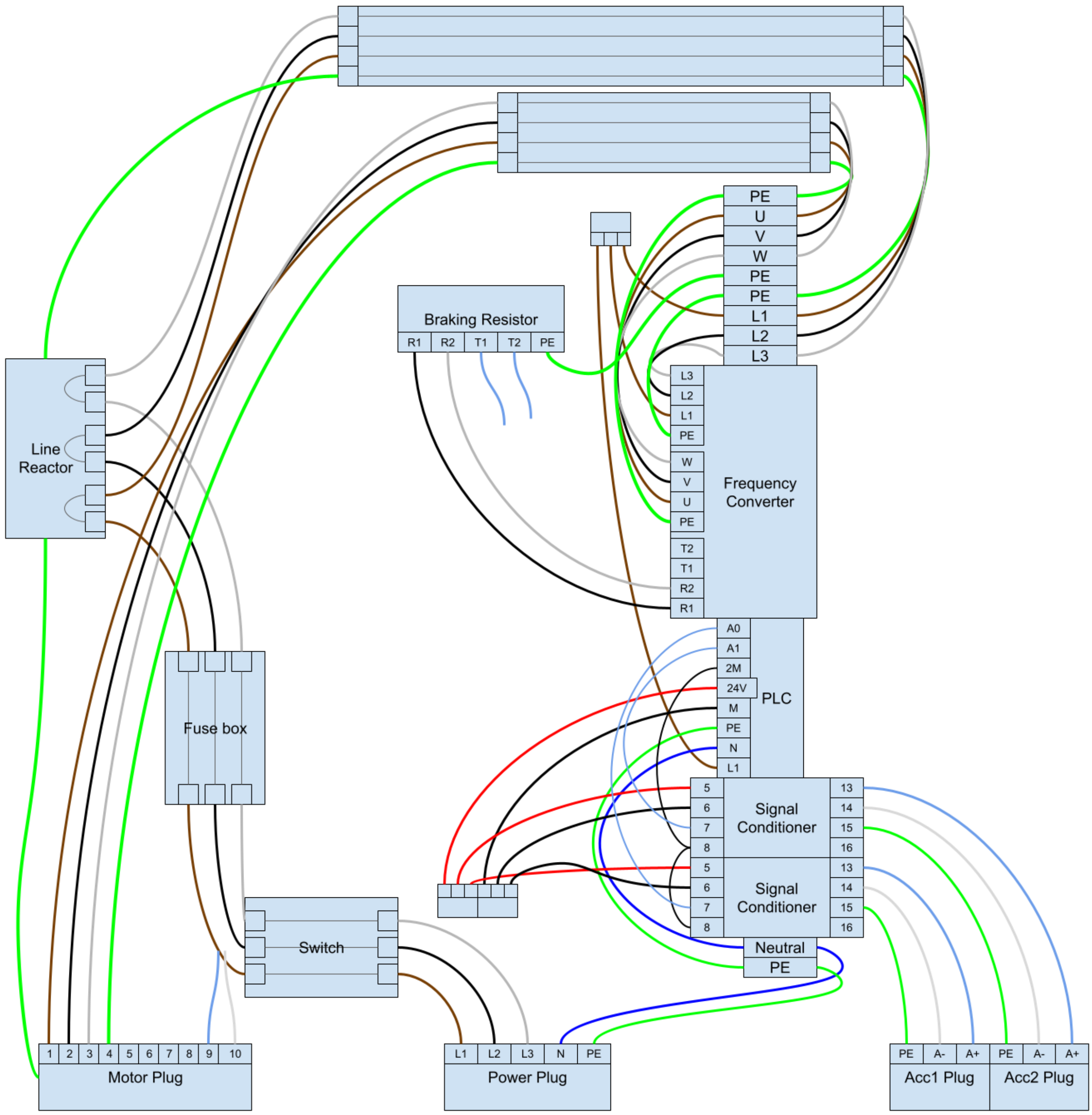


Figure 8-10: Electrical assembly with colour coordinated wiring.

Note that there are three mounted components: front plate, steel sheet and backplate. To remove front plate, disconnect: Motor Plug wire 1-4 on the Output Filter side, L1-L3 on the Switch side of the Fuse Box, PE and Neutral at the Power Plug side of the terminals, terminal

13-15 on the Signal Conditioners. To remove the backplate disconnect Braking Resistor PE on PE terminal side, T1, T2, R1, R2 on the Frequency Converter.

The signal conditioners also need to be configured to be compatible with the CTC AC102-1A accelerometers. On the Pro SC200 Series the settings are configured using 3x8 dipswitch arrays to the following setting:

- 100mV/g acceleration input, RMS, metric
- 50g full scale, acceleration output, 0-10V
- High pass filter: 5Hz, Low pass filter: 500Hz, power ON

	1	2	3	4	5	6	7	8
SW1	1	0	1	1	1	1	0	1
SW2	0	0	1	1	1	0	0	1
SW3	1	1	1	1	0	1	1	1

Table 8-4: Signal conditioner switch configuration logic.

Figure 8-11: Dipswitch array on SC200.

8.4.2 Software

A new more reliable and user-friendly software was needed for running the VSR. The S7-1200C PLC is usually programmed using SIEMENS' own PLC programming software called STEP7, which is part of the TIA-portal software package. However, plotting acceleration-frequency as an XY graph would be difficult, and the UI would require a SIEMENS HMI screen or a local runtime license for the user. The PLC and inverter use PROFINET protocol, which has limited support libraries compared to MODBUS or USS. But for the S7-1200C it is possible to use the Snap7 library for communication. It is available for many programming languages, where C# (Sharp7) was chosen as it is the only driver (as of 2024) that can communicate with Universal Windows Platform (UWP) (meaning it can run on most Windows devices), and C# is a relatively easy programming language to write and learn.

Using WPF a UI was made, wherein OxyPlot is used for plotting data. Initial tests were done by reading the raw analog data from the PLC's two analog pins and manually stimulating the accelerometers. This showed the connection and communication with the PLC was successful.

Figure 8-12: Initial test of Sharp7.

In Step7 a database (DB2) with the following values in the PLC:

Description	**Address (offset)**	**Type**	**Range**
Set motor speed	0	Short integer	0 - 32767
Enable motor	2	Bool	0 or 1
Accelerometer 1	64	Short integer	0 - 32767
Accelerometer 2	66	Short integer	0 - 32767
Read motor speed	68	Short integer	0 - 32767

Table 8-5: PLC database.

With the use of Sharp7 the PLC values can be written and read for the computer application over the network using the ReadArea, SetBitAt and SetIntAt functions. Note, that the acceleration is given as an integer that needs to be converted to the correct real number. For converting to g's:

$$a_g = \frac{a_{int} - n_{min}}{n_{max} - n_{min}} \times (g_{max} - g_{min})$$

Using an asynchronous method (function) in C# the values can be read and written continuously without interrupting other processes.

In the software named iVSR, the following methods are used in the main backend functionality to interact with the PLC:

Name	**Description**
ConnectPLC_ClickAsync	Connects to the PLC and FC IP addresses and starts PLC reading
InitializePlot	Creates plot model for displaying data
StartScan_ClickAsync	Asynchronously runs VSR scanning to determine FRF
StartTreatment_Click	Runs VSR treatment to a given RPM for a given time
ToggleMotor_Click	Toggles motor on and off
StopMotor_Click	Stops motor and disconnects from devices

Table 8-6: main iVSR methods.

iVSR contains multiple scripts, where on startup MainWindow.xaml.cs is initialized, and uses other scripts that are used to add windows, data logging and methods:

Name	**Description**
FileExport.cs	Used to format data and export as CSV file (after scanning and treatment)
GetData.cs	Live data read from PLC
HelpWindow.xaml.cs	Window to help user using the software
MainWindow.xaml.cs	Window for user to connect devices, do VSR scanning and

	treatment, and open other windows
MotorFunctions.cs	Methods for specialized setting of motor speed (ramping, scanning)
ParametersWindow.xaml.cs	Window for setting motor limits, scanning range, treatment RPM and time
PlcCom.cs	Communication between application and PLC
ReportWindow.xaml.cs	Window for making VSR work reports
SetData.cs	Database of set parameters

Table 8-7: iVSR scripts.

For secure operation, whenever the application tries to send data to the PLC the connection is checked, and a warning is provided to the user. Along with checks for data formatting, to ensure only valid data formats are sent. MainWindow and ParametersWindow are the most crucial for operating the VSR system. They are also the most complex in the ways they can interact with other files and methods, the following diagram documents how the methods (yellow) interact roughly with other methods and classes (blue):

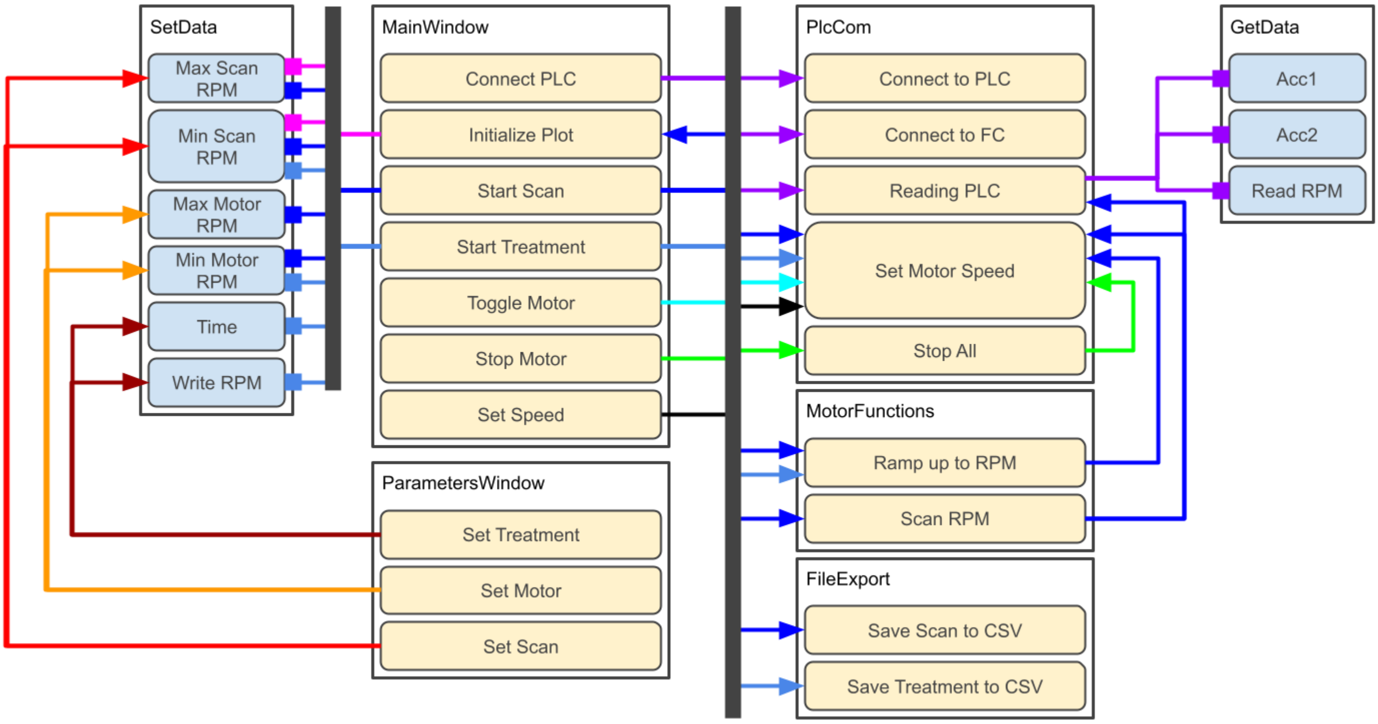


Figure 8-13: iVSR program interactions (arrow sets or runs another element, plug pulls values).

When starting the application, the MainWindow will open and initialize:

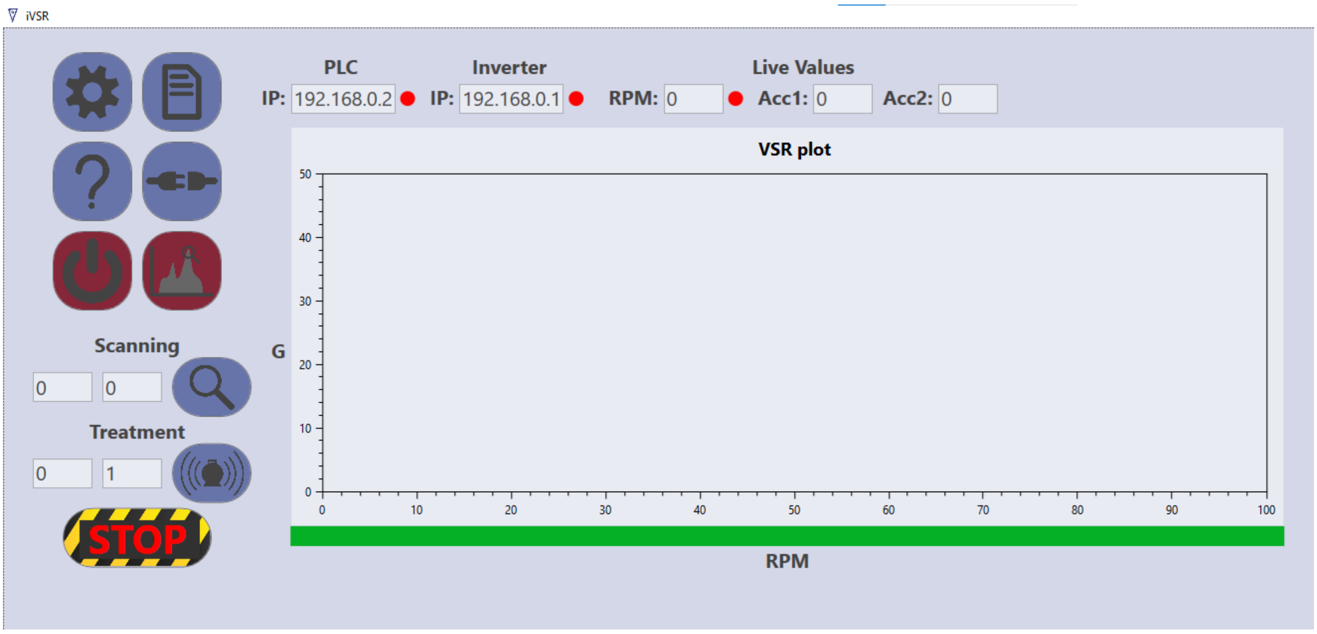
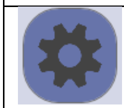
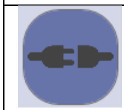


Figure 8-14: iVSR MainWindow.

The main window has the following interactive elements:

Icon	**Type**	**Function**
	Button	Opens parameters window
	Button	Opens report window
	Button	Opens help window
	Button	Tries to connect to the given IP addresses
	Button	Toggles motor on and off (red = off, green = on)
	Button	Toggles adaptive scanning method (red = off, green = on)
	Button	Starts/ends VSR scanning

	Button	Starts/ends VSR treatment
	Button	Stops motor and all connections
IP: 192.168.0.2 ●	Text box	Text input for PLC IP address (red = not connected, green = connected)
IP: 192.168.0.1 ●	Text box	Text input for frequency converter IP address (red = not connected, green = connected)

Table 8-8: iVSR MainWindow interactive elements.

The live values are displayed in the top bar. During scanning the plot is updated with the measured motor speed and acceleration values, and the progress bar is tied to the current motor speed over the scanning range. To the left of the scanning button the minimum and maximum scanning RPM. Likewise, to the left of the treatment button the current treatment RPM and remaining treatment time is displayed. An extra button was added that puts the software into safe mode, where the buttons only display their functionality but don’t run any functions.

Clicking the parameters button shows the following window:

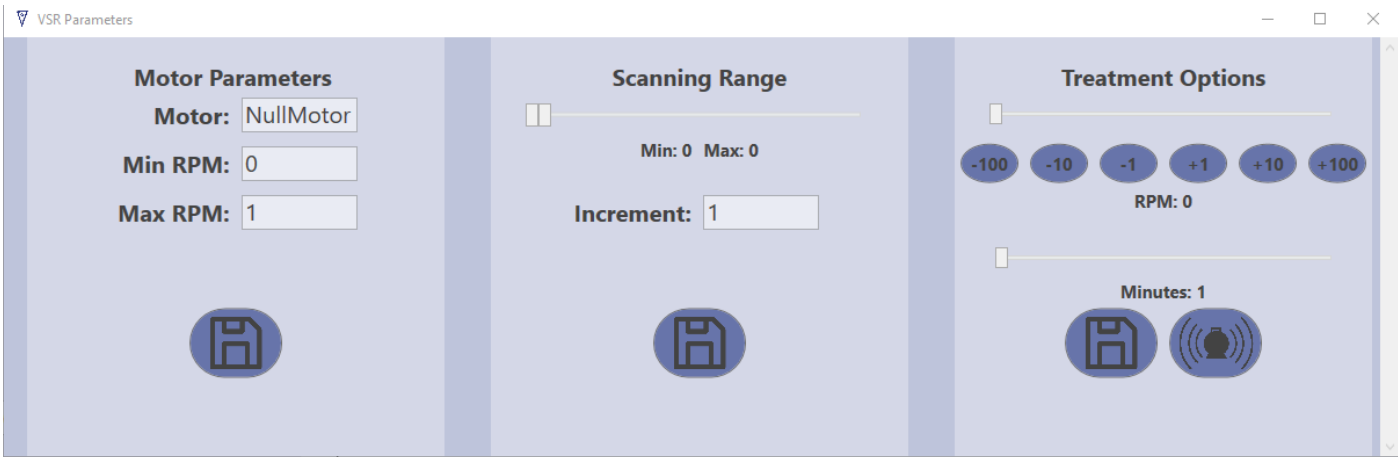


Figure 8-15: iVSR ParametersWindow.

The motor’s name, minimum and maximum RPM are inserted in the left column, and to set them the save icon is clicked. The scanning range is set by a double range slider, and an RPM increment is set in the textbox, clicking the save icon sets these values. The slider range limits are set by the motor minimum and maximum RPM. In the right column the treatment

RPM is set (with limits given by the scanning range) and can be incrementally adjusted using the buttons below. The time is set by a slider between 1-60 minutes. The save button saves these values, where the motor button sets the RPM immediately.

Clicking the report button shows the following window:

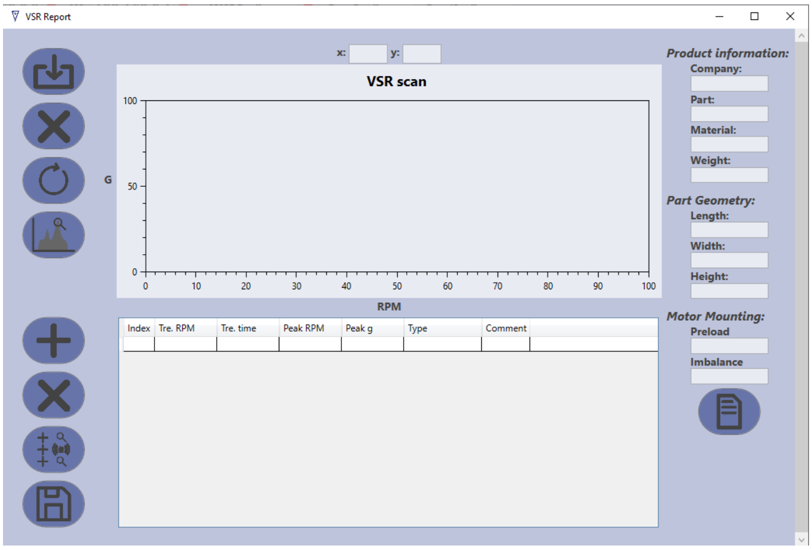


Figure 8-16: iVSR ReportWindow.

This window allows the user to make quick standardized work reports for VSR treatments. For this the graph has the following functionalities:

1. Import CSV button, to import the previous VSR scans.
2. Delete data series button, to remove specific graphs.
3. Reset button, to reset the plot view.
4. Peak finding button, finds the maximum acceleration of all the plots.
5. X and Y coordinate indicators, by right clicking on a graph the coordinate of the point is displayed.

To document the process a table used for writing and connecting data entries, with the following functionalities:

1. Add row button, to add an entry which can be: prescan, treatment, postscan, setspeed (when having changed the speed during treatment. The column at set with relevant data (example: Type = treatment, Tre. RPM = 5000, Tre. Time = 30).
2. Remove selected row button, for removing rows.

3. Add template button, which adds 3 rows with the standard prescan, treatment, postscan.
4. Save button, to save the table as a CSV file.

After having filled out the graph and table, other relevant information can be filled out in the right column, and clicking the report button will then create a PDF file with the formatted information (it will also prompt the user if fields are not filled out), and pictures may be added.

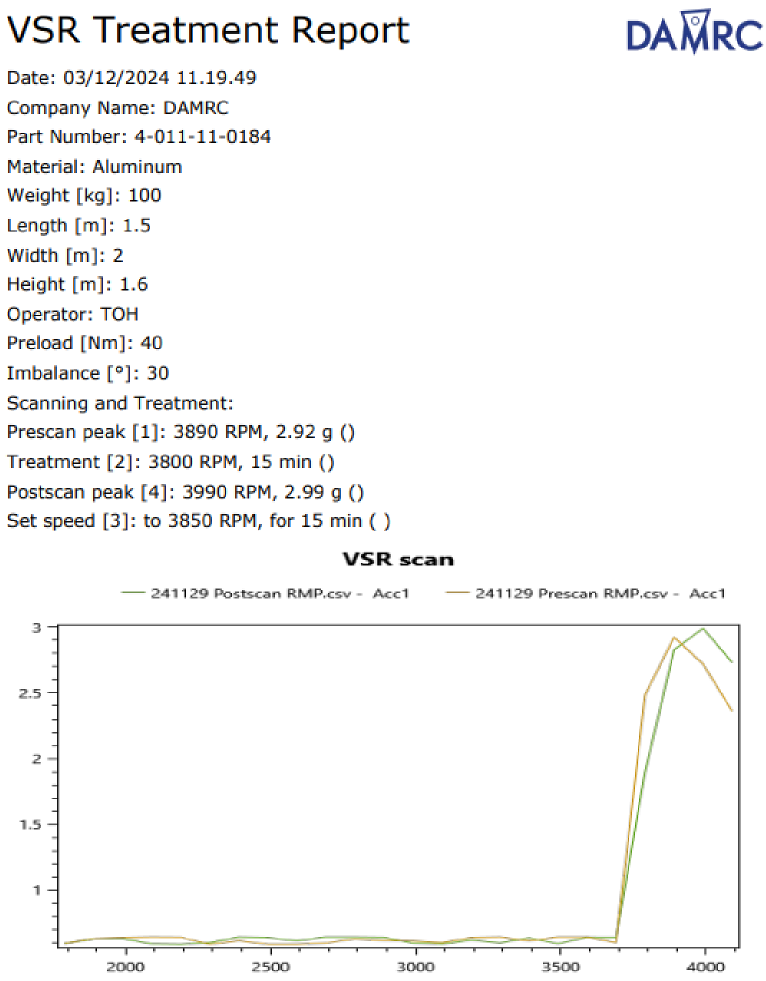

VSR Treatment Report
DAMRC
Date: 03/12/2024 11.19.49
Company Name: DAMRC
Part Number: 4-011-11-0184
Material: Aluminum
Weight [kg]: 100
Length [m]: 1.5
Width [m]: 2
Height [m]: 1.6
Operator: TOH
Preload [Nm]: 40
Imbalance [°]: 30
Scanning and Treatment:
Prescan peak [1]: 3890 RPM, 2.92 g ()
Treatment [2]: 3800 RPM, 15 min ()
Postscan peak [4]: 3990 RPM, 2.99 g ()
Set speed [3]: to 3850 RPM, for 15 min ( )

Figure 8-17: Example of VSR work report.

Clicking the help buttons opens a window with relevant links for using the beforementioned functionalities, and other information related to VSR.

9 Experiment Design

9.1 Introduction

The equipment's use and stability are to be tested by an un-skilled operator. As described in the previous section the connections and assembly has been tested, along with initial test of the software communication with the PLC. Before putting iVSR to use some key aspects are to be tested:

1. **Accelerometer calibration**
   The accelerometers ought to have the same accuracy as other VSR equipment, wherein the integer value read by the PLC needs to be scaled appropriately.
2. **Monkey test**
   The software needs to be robust to accidental or malicious control inputs, meaning that when connected it must be observed what flaws may occur and why.
3. **Un-skilled operation**
   The goal of the project is having an un-skilled operator use it to make a VSR treatments. Observations and interview are to show how well the system is designed.

9.2 Test Design/Process

9.2.1 Accelerometer Calibration

To test if the accelerometers, correspond with other VSR systems, the accelerometers will be run first using the Advanced VSR system at DAMRC and note the values, by exciting an object at a 2 different motor speeds, then switch the accelerometer to the iVSR system and run the same 2 motor speeds and note the integer value. Then using the formula:

$$a_g = \frac{a_{int} - n_{min}}{n_{max} - n_{min}} \times (g_{max} - g_{min})$$

The acceleration can now be calibrated.

9.2.2 Monkey Test

The monkey test is testing for software error, which can occur due to improper use. Relevant test involves pushing different buttons and observe how the system responds. iVSR has nine buttons and two text boxes in the main window, 10 buttons, four textboxes and three sliders in the parameters window, nine buttons and nine textboxes in the report window. Some components are simple enough that they would not course failure, these include: four sliders as they only impact other components when set by a button, six buttons for adjusting slider values, all buttons in report window, as they are self-contained in the report window, three buttons in the main window, as they only open other windows. There is also the emergency stop button and on/off switch that are hardwired. It may also occur that cables are not connected, this should also be tested. This leaves the following interactive components:

1. Connect button (and the tied text boxes)
2. Toggle button
3. Adaptive treatment button
4. Start scanning button
5. Start treatment button
6. Stop button
7. Save values buttons in parameters window

8. Set speed button
9. Emergency stop button
10. On/off switch
11. Disconnected cables for: motor, power, ethernet and accelerometers

As this leaves with 16! possible combinations, the test is narrowed to situations where the motor is running, i.e. first start scanning, start treatment or set speed. It can be further narrowed by assuming flaws where the motor is stopped are safe, thereby removing: toggle, stop, emergency stop, on/off switch, power not connected. And the save values do not send values to PLC and are therefore disregarded. This leaves the following table:

1st \ 2nd event	Connect	Adaptive	Scan	Treat	Set speed	Dis. Ethernet	Dis. Acc
Connect							
Adaptive							
Scan							
Treat							
Set speed							
Dis. Ethernet							
Dis. Acc							

Table 9-1: Monkey test table.

9.2.3 Un-skilled operation

Participants without prior experience with VSR treatment tested the system. Here they were given a short introduction of how to use it by an instructor. Then they were tasked with setting up a typical VSR treatment:

1. Set up and perform a VSR scan.
2. Find appropriate treatment values and run a treatment.
3. Perform a post scan.
4. Make a work report using the iVSR.

In the interest of time the scan and treatment times are shortened to a total of 15 minutes. The participant is observed by the instructor, to observe if the participant is having problems or errors occur. The following interview questions were used (higher values are better):

1. From 1 to 10 how easy was it to setup a VSR scan?
2. From 1 to 10 how easy was it to setup a VSR treatment?
3. From 1 to 10 how easy was it to setup a VSR report?
4. What did you find the most confusing when using iVSR?
5. What would you need if you were to do a VSR treatment by yourself in the future?
6. What would you add to the iVSR system?

9.3 Equipment for the Test

For all the tests the VSR is set up fully, the following equipment was used:

1. Metal part
2. Vise for holding part
3. Vibration motor (Brecon 18 213 514)
4. Fastening equipment (bolts, nuts)
5. iVSR hardware box
6. Laptop
7. iVSR software
8. 2x Accelerometer (AC102-1A)
9. Cables (ethernet, 400V power cable, motor cable, accelerometer cable)

Advanced VSR system (only for calibration, used as reference)

9.4 Conduction of the Test

9.4.1 Accelerometer Calibration

1. Mount the motor and vise on the vibration platform.
2. Fixture the beam in the vise with adequate stick out to amplify vibrations.
3. Mount the accelerometers at the end of the beam.

4. Wire up motor, accelerometer and laptop to the iVSR box, and second accelerometer to Advanced VSR system.

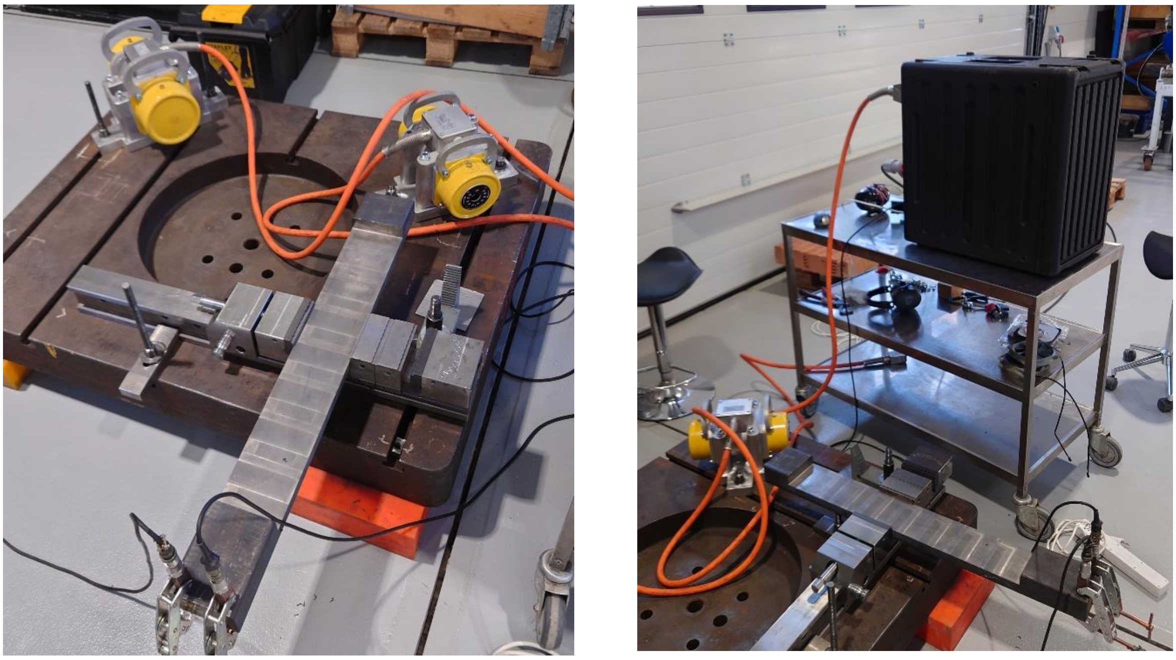


Figure 9-1: VSR test setup.

5. Run VSR scan using the iVSR software.

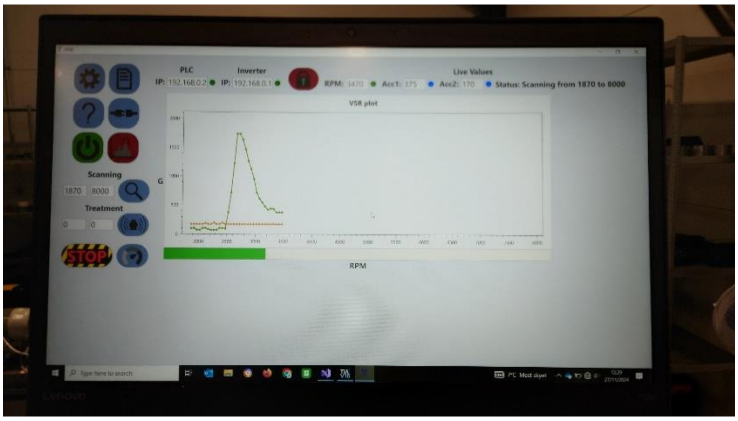


Figure 9-2: VSR scan for test (raw integer reading).

6. Run treatments at various amplitudes.
   a. Note the amplitudes in the iVSR (integer value) and advanced VSR (g's) systems.
7. Find relation between integer value and g's. Assuming linearity, the points were graphed, and a linear regression was used.

9.4.2 Monkey Test

The monkey test used the same setup, but where varies of the previous combinations were used. If any faults or bugs were observed, they are noted.

9.4.3 Un-skilled operation

Three participants were used to evaluate the performance and use of iVSR. They were given a 10-minute introduction for using iVSR, and the task of VSR treating a part (given 20 minutes to setup a short scan, 2 minutes treatment, post-scan and make a report). The hardware and fixturing were setup prior to the test. After the test they were given the interview questions in hand to fill out.

10 Conclusion of the Test Results

For the calibration the following five datapoints were used to make a linear regression:

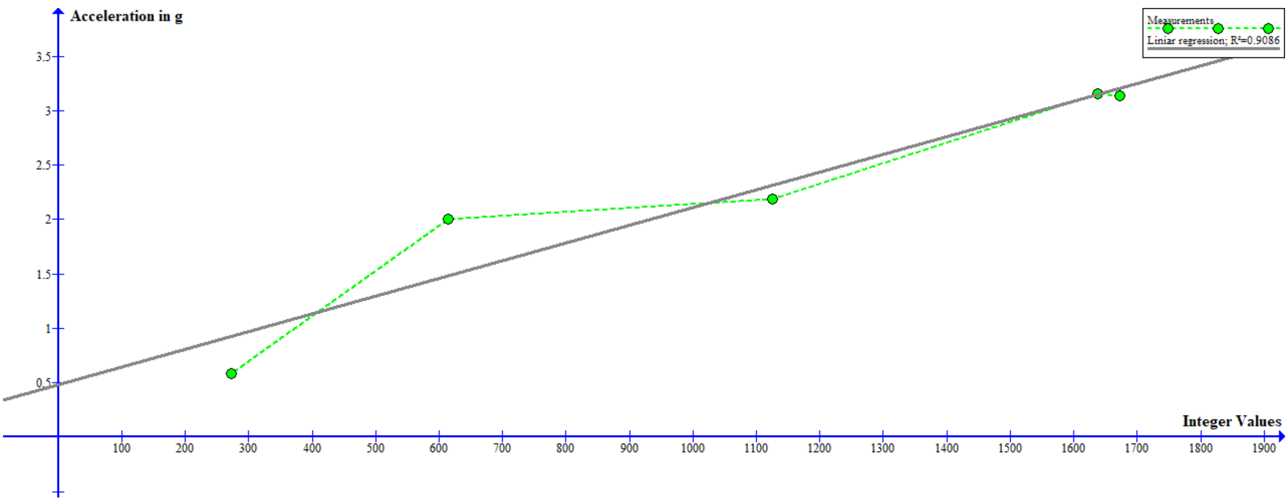


Figure 10-1: VSR calibration points.

This gives the following transform from integer value to g's:

$$a_g = 1.16838\ 10^{-3} a_{int} + 0.48042494$$

This gives a decent scaling of the integer values, however the measurements were not conclusively linear, with a R2 of 0.91. And the calibration of the Advanced VSR is also unknown. But for the purpose of normal treatments, the scaling is a lesser factor, as it is the frequency of the eigenvalues that are more relevant than the amplitude (though the amplitude should be at least 3 g from experience). For common usage it is therefore and adequate approximation. For a more accurate reading, a more thorough test ought to be conducted with a wider range of amplitudes.

The unskilled operation test had the following feedback:

1. From 1 to 10 how easy was it to setup a VSR scan?

   5, 10, 6
2. From 1 to 10 how easy was it to setup a VSR treatment?

   5, 10, 5
3. From 1 to 10 how easy was it to setup a VSR report?

   6, 5, 7
4. What did you find the most confusing when using iVSR?

   Visuals, report generation, symbols
5. What would you need if you were to do a VSR treatment by yourself in the future?

   More intuitive, tooltips, more explanation/manual
6. What would you add to the iVSR system?

   Add tooltips, add help, remove IP address, manual, tooltips

Scanning, treating and making a report were all easy to medium to setup. The qualitive questions show that it was mostly the symbols that were confusing, wherein tooltips ought to be added (new icons and tooltips were added in the final version).

The monkey test showed the following results:

1st \ 2nd Event	Connect	Adaptive	Scan	Treat	Set Speed	Dis. Ethernet	Dis. Acc
Connect			Scans as Normal	Treats as normal	Sets the speed as normal		
Adaptive			Performs the scan than an adaptive scan at the peak value	Treats and does the adaptive scanning as expected	Ramps the rpm to the set value		
Scan	Scans as normal and states that the PLC is successfully connected	Performs the scan than an adaptive scan at the peak value		Stops the Scan and saves the values.	Stops the scan and sets the motor to the set rpm.	States failed to connect to the PLC. Then immediately does the scan.	Graph flatlines
Treat	Treats as normal and states that the PLC is successfully connected	Treats and does the adaptive scanning as expected	Stops the treatment and does a scan.		Does the treatment as expected.		Graph flatlines
Set Speed	Sets as normal and states that the PLC is successfully connected	Sets as normal	Stops the set speed and performs a scan.	Sets speed, then ramps up and does the treatment.			Graph flatlines
Dis. Ethernet			States that the PLC is not connected.	States that the PLC is not connected.	States that the PLC is not connected.		
Dis. Acc.			Graph flatlines	Graph flatlines	Graph flatlines		

Table 10-1: Monkey test result table.

No fatal errors were found in the monkey test. The worst case was found in mashing the Scan button quickly 20+ times, which slowed down the software, but still it did not crash.

11 Discussion

In previous VSR treatment projects the reporting and treatments were found to be inconsistent and or missing documentation and had the need for an experienced operator to use it. The main obstacles were identified for common industrial use: operator to use proper fixturing that is highly geometrically depended, varying natural frequencies during treatment, complex frequency responses. These factors result in questionable results and efficiencies for VSR treatment.

Using components from a previous VSR system it was possible to build a new VSR system to run motors and accelerometers using a laptop. C# was then used to make control functions to run VSR treatments, analyse and document the results. WPF was used as a framework for the user interface. This allowed the use of open-source libraries which sped up software development. The versatility and useability of the equipment is also improved, as it can be compiled into an app for any Windows, OS or Linux based computer.

The testing showed equivalent results with Advanced VSR's systems, but with additional benefits like the report builder, customization of scans, customization of treatments and automation. With only a short introduction it was possible for un-skilled operators to get a decent level of confidence in doing VSR treatments. Note, that the fixturing of parts is still complex but guides and manuals have also been made during the project to alleviate these issues. The system cost around 16 000 DKK for the components. This ease of use and low cost allows for wider adoption of VSR in industry.

The addition of formatted data capture will also improve the research and development or VSR greatly, as this can be used in analysis to improve VSR techniques and knowledge. Wherein the use of open-source framework can allow researchers to change methods indefinitely and without the limitations of common PLC programming software.

The development and research in this project also give possibilities for implementing similar techniques for other closed, hard to use or underdeveloped electro-mechanical systems. Including expanding the iVSR network for technologies such as vibration assisted welding.

Further research projects may look at integrating Barkhausen noise analysis or X-ray imagining of the residual stress, to quantify localized stress relief. More accelerometers could also be used to get a more detailed view of the vibration modes of objects, to optimize vibrational modes. As VSR is still largely unknown in Danish industry, a comparison of VSR and TSR could provide a reference point for integrating the technology, as they both have strengths and weakness as seen in studies comparing them.

Another barrier not covered in this project is industry standards and practice. As TSR is a very well-integrated and standardized technology, with a lot of dedicated infrastructure, standards and education related to it. This gives the costumer great confidence in the process. Where from DAMRC's experience, the products always undergo a trial run, and even if some companies have the authority to switch from TSR to VSR, many cannot get the permission from the end customer/designer to use uncommon technology.

12 Conclusion

Vibratory stress relief is tried and tested at DAMRC, but has had difficulties with wider adoption industries, due the complexity of understanding mechanical vibrations and the lack of knowledge and research on the subject. An easier to use and low cost VSR system was then developed to promote wider adoption in industry. Components from a previous VSR project were used to make a new VSR system dubbed iVSR. Herein, a new electrical assembly was developed, tested and documented. Showing it could run VSR motors and reading accelerometers.

To facilitate low-cost development and access to open-source libraries the controls were programmed using C# with a WPF user interface. The system can then be programmed and run using a commercial computer over ethernet. The also allowed for the main functionalities (VSR treatment and scanning) and new functionaries to be implemented, where a key aspect was standardising treatment procedure and documentation.

The system was tested using three unskilled operators, which in only 30 minutes had decent understanding of how to operate the system. A skilled operator at DAMRC also uses the system regularly now, wherein the new UI and functionalities are appreciated. This shows great promise in making VSR more approachable by industry and allows DAMRC to gather valuable data from VSR treatment, that can be used in future research.

More generally the adoption of VSR would provide Danish industry and engineers with a new method of stress relief, that is low cost and has a significantly lower carbon footprint compared to TSR. The success of this project has shown that an un-skilled operator can do VSR scans and improved the data gathering and documentation capabilities of VSR.

13 Appendix

13.1 DAMRC's VSR Motors Nameplates and Configurations

Nickname: Big Yellow (Induction motor, Delta)

Brecon Germany			
www.brecon.de			
18 213 514			
250V		150Hz	
3 ~	4.7A	8976 1/min	
1800W		76263	

Nickname: Small Yellow (Induction motor, Delta)

Brecon Germany			
www.brecon.de			
18 219 529			
250V		200Hz	
3~	4.1A	11 660 1/min	
1500W		74960	

G120C frequency converter configuration:

DI 0:	p1055[1] BI: Jog bit 0
DI 1:	p1056[1] BI: Jog bit 1
DI 2:	p2103[1] BI: 1st acknowledge faults p2104[0] BI: 2nd acknowledge faults
DI 3:	p810 BI: Command data set selection CDS bit 0
DO 0:	r52.3 CO/BO: Status word 1: Fault present

AO 0: r21 CO: Actual speed smoothed
[i] Speed setpoint 16-bit
Drive unit line supply voltage: 400 V
Maximum braking power: 4.00 kW
Drive filter type on motor side: none
Select motor type: [1] Induction motor
Motor connection type: Delta
P305[0] Rated motor current: small yellow 4.10 Arms / big yellow 4.7 Arms
P307[0] Rated motor power: small yellow 1.5kW / big yellow 1.8kW
P311[0] Rated motor speed: small yellow 12000 RPM / big yellow 9000 RPM
P304[0] Rated motor voltage: small yellow 250 Vrms / big yellow 250Vrms
P310[0] Rated motor frequency: small yellow 200 Hz / big yellow 150 Hz
P335[0] Motor cooling type [0] Natural ventilation
Temperature sensor: none
Motor holding brake configuration: [0] No motor holding break available
Synchronization of the speed of the drive with the speed of the PLC:
Reference speed: 6000*
Maximum speed: 12000
Ramp-up time: 10 s
Ramp-down time: 10 s
(quick stop) ramp-down time: 0 s
Current limit: 4.00 Arms
[0] Constant load (linear characteristic)
[2] Identifying motor data (at standstill)

Step7 Sinaspeed control function used in PLC:

Table 2-1: "SinaSpeed" input parameters

Name	Type	Start value	Function
EnableAxis	BOOL	FALSE	Start/stop the drive (drive control word 1 bit 0 is assigned)
AckError	BOOL	FALSE	Acknowledge faults in the drive (assignment of drive control word 1 bit 7)
SpeedSp	REAL	0.0	Specify setpoint speed [rpm]
RefSpeed	REAL	0.0	Drive reference speed. (Entry must match drive parameter p2000)
ConfigAxis	WORD	16#003F	Assignment of the drive control word (drive parameter r2090). The initial value 16#003F sets bits 1 to 6 to TRUE: Bit 1: OFF2 Bit 2: OFF3 Bit 3: Enable operation Bit 4: Enable ramp-function generator Bit 5: Continue ramp-function generator Bit 6: Enable the speed setpoint
HWIDSTW	HW_IO	0	Hardware ID setpoint (see Parameterization section)
HWIDZSW	HW_IO	0	Hardware ID actual value (see Parameterization section)

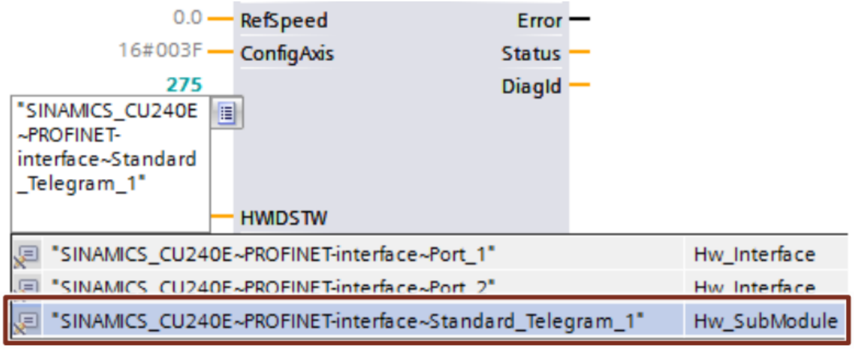
Table 2-2: "SinaSpeed" output parameters

Name	Type	Start value	Function
AxisEnabled	BOOL	FALSE	Drive operation enabled
Lockout	BOOL	FALSE	Drive switch-on lock is active
ActVelocity	REAL	0.0	Actual speed of the drive
Error	BOOL	FALSE	Drive error active

Name	Type	Start value	Function
Status	WORD	0	Display status values: 16#7002: No error present 16#8401: Error active in drive 16#8402: Switch-on lock active 16#8600: DPRD_DAT error 16#8601: DPWR_DAT error
DiagId	WORD	0	Advanced communication fault (error when calling an instruction)

The HWIDSTW and HWIDZSW block inputs must point to the hardware identifier of the standard telegram.

Figure 2-3: Telegram slot assignments



14 References

[1] Retsinformation, “retsinformation.dk,” Retsinformation, 10 January 2019. [Online]. Available: https://www.retsinformation.dk/eli/lta/2019/26. [Accessed 14 November 2024].

[2] C. A. W. A. J. W. D. J. C A Johnston, “Vibratory stress relief – an investigation of the underlying processes,” 1995. [Online]. Available: https://doi.org/10.1243/PIME_PROC_1995_209_228_02.

[3] T. G. R. C. I. L. D. Amauri Gomes de Moraes, “Quantification of CTOD fracture toughness in welded joints to evaluate the efficacy of vibration stress relief compared to thermal stress relief,” Materials Research., 2020. [Online]. Available: https://doi.org/10.1590/1980-5373-MR-2020-0157.

[4] A. G. de Moraes, T. G. R. Clarke and I. L. Diehl, “SciELO Brazil,” Ibero-american Journal of Materials, 2020. [Online]. Available: https://www.scielo.br/j/mr/a/y5DQXNdg6CWBrvW85g3prpS/?lang=en#. [Accessed 6 12 2024].

[5] SIEMENS, “SINAMICS Low-voltage SINAMICS G120C converters,” SIEMENS, 2014.