CHAPTER III

METHODS AND CONFIGURATIONS

2.3 METHODS AND CONFIGURATIONS

The electronic circuits to ensure autonomous acquisition, the hardwares and softwares that involves ADAQ have been configured on a very specific way to ensure the correct recording, processing and analysis from data.

2.3.1 THE ELECTRONICS IMPLEMENTED ON THE VEHICLE

In order to make possible data acquisition and interaction of acquisition phases to the pilot an electronic circuit has been implemented on the vehicle. The central pillar from the implemented electronics in the vehicle is the CDB. CDB at the same time supplies the current to DCB in order to make possible interaction with the pilot.

The CDB is compound by a timer, five relays, a fuse and two integrated boards two allow intermittent functions from the 2 LEDs on the DCB.

- The timer allows the power supply to QuantumX computers once the vehicle is turned off.
- The fuse makes secure the electronics.
- The two integrated boards use the 555 timer integrated circuit as the main component, it is also compound by two fix electric resistance, a variable electric resistance (to control the intermittence frequency) and a capacitor. [10][11]
- The five relays have the function to supply power to one circuit or another circuit according to the CX22-W digital outputs.

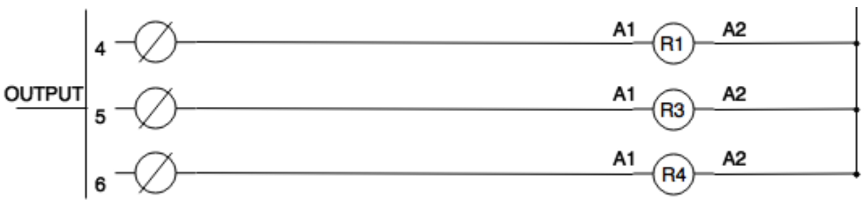


Fig. 26 CX22-W digital outputs 4,5 and 6 pointing to relays 1, 3 and 4 respectively.

Because of confidentiality reasons only a section from the general ADAQ circuit on the vehicle can be shown. Different components previously cited are identified.

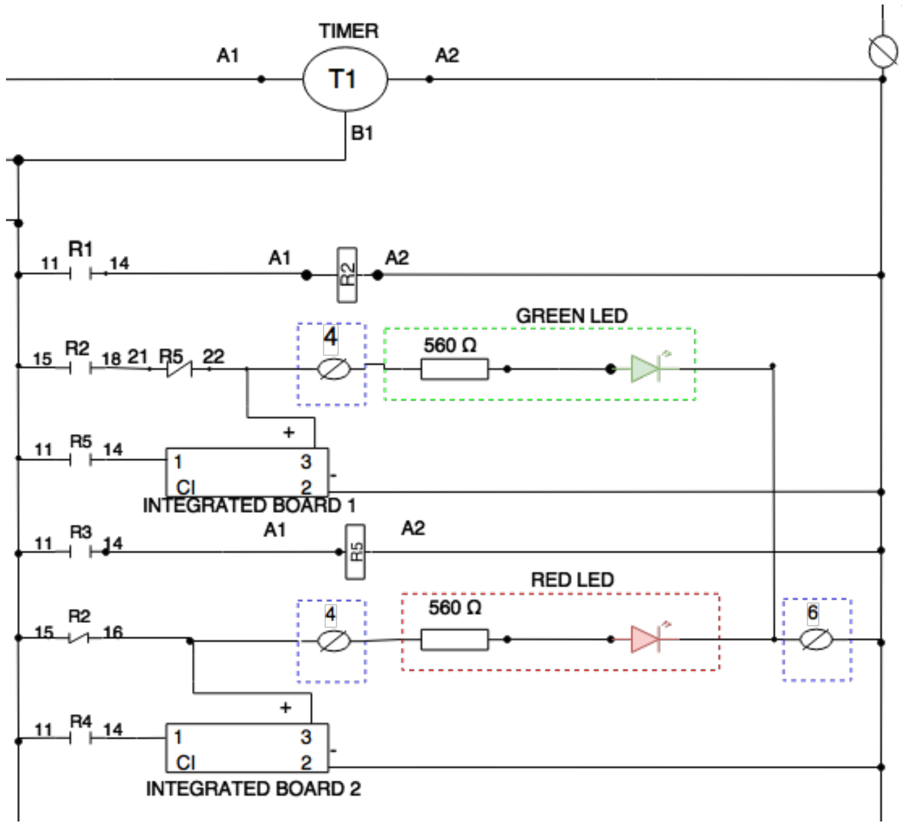


Fig. 27 Circuits implemented on the DCB and DCB.

2.3.2 ACQUISITION PROJECT

Catman SW offers multiple configuration possibilities. On this chapter it is stated a way how can acquisition be configured in order to fulfill most of the targets an ADAQ project might have. No explication about how instrumented or CAN bus channels are configured is stated, only relevant information and particular parameters for every ADAQ project.

PROJECT AUTO-START

Before getting into Catman it is necessary to order Windows XP on the CX22-W to, first, start other hardwares recognition, for this particular case the two MX840 and one MX1615, and then, to auto-boot the desired acquisition project file.

Embedded Win XP offers an easy way to configure these actions through the CX22-W Shell. The only important thing to set is the software break time before starting HWs recognition and the break time between the end of HWs recognition and the start from the acquisition project booting. These two values depend on the amount of HWs plugged into the CX22-W, so the more HWs they are higher is the time value it is need to be introduced. For these particular case, it is determined the project can be correctly booted in no less than 210 seconds (3min 30sec). This is the total time from the moment the pilot tuns on the vehicle through the turn of the key until the vehicle HWs are ready to acquire.

Fig. 28 Time values being set on the CX22-W Shell.

CHANNELS CONFIGURATION

Once instrumented physical channels and CAN bus channels are configured is time to set the computation channels.

Computation channels are created from instrumented and CAN bus channels or from other computation channels. Theses channels can be created with different objectives such as correcting physical channels, initialize to zero CAN bus channels, grouping channels or setting logic conditions in order to use them later to configure DAQ Start/Stop triggers or output signals from the CX22-W.

In order to define ADAQ acquisition parameters different computation channels have been configured:

· Channel_Sum: This computation channel sums the minimum number of channels in order to know if any from the channels has electric continuity or it is unplugged/broken. Obviously if the whole acquisition HW is not working this computation channel does also contemplate it as a group of physical/CAN bus channels would be off. When it is said 'the minimum number of channels' is because it is only necessary to choose one from the CAN bus channels, as all of them are connected to the same slot. The same happens with the three GPS channels (Altitude, Longitude and Latitude), only one of them has to be chosen; if one from this is off the rest will be off as well. One of the reasons of this configuration, apart from saving time, is because Catman allows a maximum number of characters on its computation channels. The logic function implemented on 'Channel_Sum' is the following:

$$Channel\_Sum = if\left(\left(\sum_{i=0}^{n} Minimum\_Channels \leq x\right), 0, 1\right)$$

So if the sum from the minimum number of channels is equal or below x the computation channel indicates value 0 while if this is not true value 1 is sent. X is a deduced value from the set value at channel overflow. When a channel is off Catman sends a very high negative value previously defined. For this project setting x as 10 times bigger than the previous value is enough to detect any error on any of the channels. This channel is used by CX22-W signal outputs configuration.

· Channel_start_adq: This computation channel defines if the vehicle is under concrete parameters or not as long as every channel is active. The channels it point to are GPS longitude and latitude, CAN bus vehicle speed channel and the previously defined Channel_Sum. This channel is used to set DAQ start/stop conditions and to set CX22-W digital outputs.

$$Channel\_start\_ad = if((Channel\_Sum=1) AND (Speed \geq 5) AND (Adq\_area=1),1,0)$$

Where Adq_area refers to an other computation channel:

$$Adq\_area = if((longitude>x) AND (latitude<y) AND (longitude<x') AND (latitude<y'),1,0)$$

This is the zone where the acquisition is going to take place, for example, a specific testing track. Points (x,y) and (x',y') define the two vertex of the diagonal from a rectangle. From the intersection of two plans it is defined a specific geographic area.

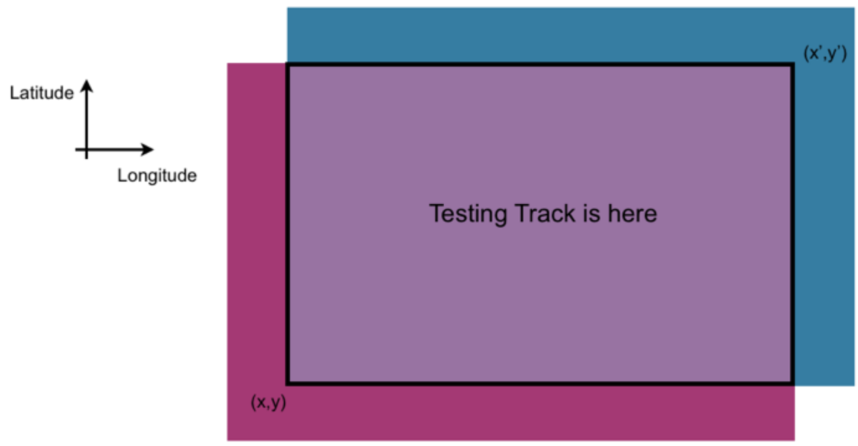


Fig. 29 Intersection from two plans creates a geographic rectangle.

Speed vehicle equal or higher than 5 Km/h is set to make sure the vehicle is on movement.

So as long as no channel error is found, speed is equal or higher than 5 Km/h and the vehicle is inside the testing track area the computation channel will send 1 and if any from the parameters is not true Channel_start_adq indicates value 0.

· Channel_event: This computation channel defines if during Channel_start_adq conditions any event is taking place. The event depends on the target of the project. For instance, it could control the temperature from the 4 suspensions, thought the 4 instrumented thermocouples on each one. Defining a warning value of temperature this channel would have the following expression:

$$Channel\_event = if((Channel\_start\_ad = 1) AND (Temperature = 1), 1, 0)$$

Where Temperature is another computation channel:

$$Temperature = if\left(\left(Rear\_right_s \geq T\right) OR \left(Rear\_left_s \geq T\right) OR \left(Front\_right_s \geq T\right) OR \left(Front\_left_s \geq T\right), 1, 0\right)$$

The four studied points are defined: rear right and left suspensions and front right and left suspensions. T is the warning temperature value it is set for this channel according to the project specifications.

If during Channel_start_adq=1 temperature in the suspensions is higher than T, Channel_event sends 1 while if any from the two conditions is not true Channel_event sends a 0.

Ready to GO!	C	if((Channel_error_1_2=1) AND (GPS1=1),1,0)
Channel_event_sig	C	if((Ready to GO!=1) AND (Channel_event=1) AND (Channel_error_1_2=1),1,0)
Speed more then 5	C	if((Velocidad Vehiculo>=5),1,0)
Condition for DAQ !	C	if((Speed more then 5=1) AND (GPS1=1),1,0)

Fig. 30 Example of other computation channels being configured with Catman.

SET UP CX22-W OUTPUT SIGNALS

Once computation channels are configured is time to configure the CX22-W digital output signals. CX22-W has 3 different outputs. According to its configuration SW Catman can order the HW to output a certain electric current through one of the 3 signals. They are switched off unless the configured condition is true, then one or more from the Digitals I/Os are enabled. Digitals I/O are the 4th, 5th and 6th.

- Condition to enable the 4th digitals I/O

If 'Channel_Sum' value is higher that 0,5 (so when 'Channel_Sum'=1) the 4th digital output will be switched on. Thanks to the implemented electronics the red LED of the DCB will be switched off and the green LED will be on.

-

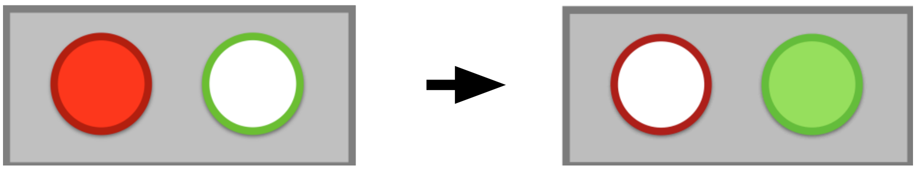


Fig. 31 CX22-W Digital output 4 is switched on now.

- Condition to enable the 5th digitals I/O

If 'Channel_start_adq' value is higher that 0,5 (so when 'Channel_start_adq'=1) the 5th digital output will be switched on. Thanks to the implemented electronics the green LED of the DCB will turn into an intermittent green light.

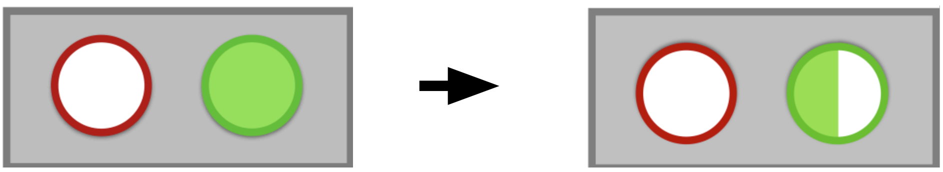


Fig. 32 CX22-W Digital outputs 4 and 5 are switched on now.

- Condition to enable the 6th digitals I/O

If 'Channel_event' value is higher that 0,5 (so when 'Channel_event'=1) the 6th digital output will be switched on. Thanks to the implemented electronics the red intermittent LED on the DCB will switch on.

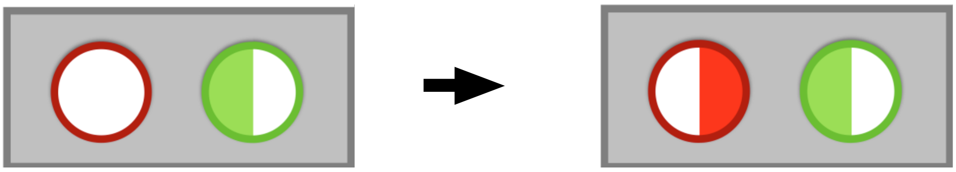


Fig. 33 CX22-W Digital outputs 4,5 and 6 are switched on now.

SET UP DAQ START/STOP TRIGGERS

After setting the digital output signals from the CX22-W is time to set the start and stop from the acquisition. Triggers are not like conditions; to trigger any action only is required the action to take place once at anytime and it will be triggered.

- DAQ START trigger

According to the configured computation channels the trigger allowing acquisition to start recording will be when 'Channel_start_adq' value is higher that 0,5 (so when 'Channel_start_adq'=1).

- DAQ STOP trigger

In order to stop data recording it is set that when 'Adq_area' channel is below 0,5 (so when 'Adq_area'=0). DAQ Stop will be triggered.

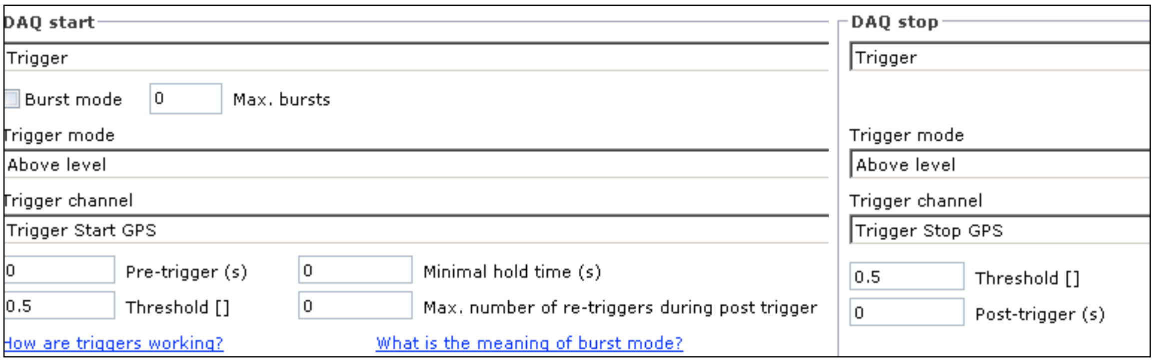


Fig. 34 Trigger DAQ Start/Stop conditions being configured on Catman.

This is a particular configuration just in case the testing track is not near from the circuit Box. In case it is desired to start acquiring once the vehicle has just leave the Box, another configuration would be referring the 'Adq_area' channel to the geographical coordinates from the Box. Then activating DAQ start/stop triggers according to this computation channel.

2.3.3 DATA COMMUNICATION

In order to ensure the right TS communication several commands have been implemented. It is important to remark data is being moved along the three servers, from CX22-W to PC-Box to Shared Server, so it is not used the copy and paste function. That means data is only stored on the Shared Server, so CX22-W and PC-Box are released and ready for the next data transmission routine. Data from the CX22-W to the PC-Box is transferred during the acquisition stops in the Box, every 2 or 3 hours, while data dump to the Shared Server starts at a specific time once the acquisition day has finished, mainly at night, when company's data flow is reduced. On the following table it is shown the symbology from the three commands which start the transmission process on each computer and how are they activated.

Table 2 The three data transmission starting commands on every computer.

Command symbology	Activation method	Activation frequency
Start ADAQ	Double click by the Pilote on the PC-Box.	Only once at the very beginning of the ADAQ Project
CX22-W to PC-Box data transference	CX22-W Windows task Scheduler.	At logging into the system. So every time the Vehicle turns on.
Pre-analysis	PC-Technician Windows task Scheduler.	Every day at the specific configured time.

On the following flowchart 1 it is represented the whole communication process by an ADAQ Project. V2B is an event type interaction, while B2SS is an interaction by programmed time sync between PC-Box and PC-Technician computers. All commands shown are described in detail on this chapter. On the next table is stated information to sum up every ADAQ command specifications and the code where is available from the Annex.

Flowchart 1 Commands governing ADAQ data communications on every computer.

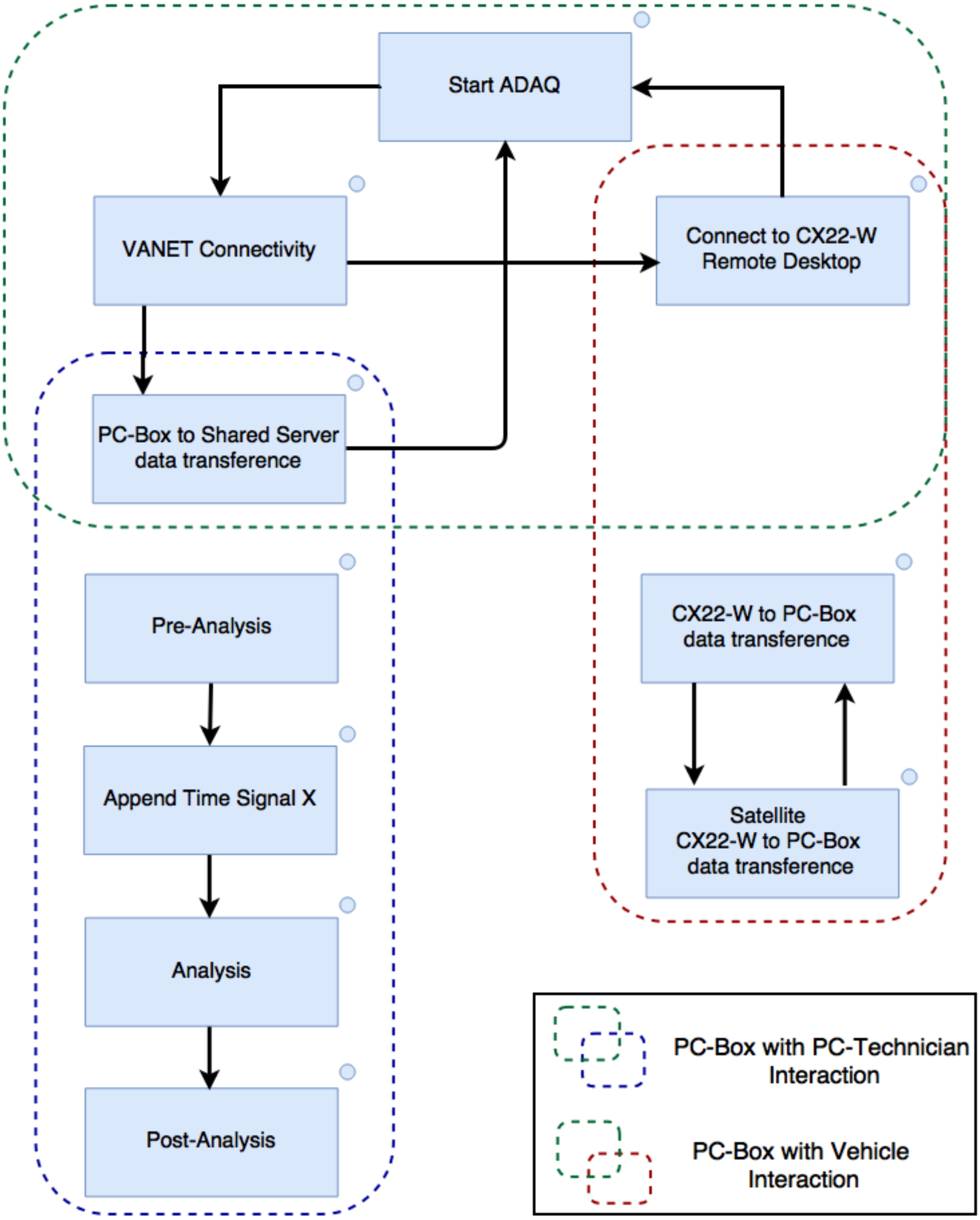


Table 3 Every ADAQ command in charge of data communication/analysis.

File name and extension	Location	Code	Language	Command
CX22toPCBOX.bat	CX22-W (copy on the SS)	Annex Codes C5	Visual Basic	CX22-W to PC-Box Connection
Satellite_CX22toPCBOX.bat	CX22-W (copy on the SS)	Annex Codes C6	Visual Basic	Satellite CX22-W to PC-Box Connection
Star_ADAQ.bat	PC-BOX (copy on the SS)	Annex Codes C1	Visual Basic	Start ADAQ
Vanet.bat	PC-BOX (copy on the SS)	Annex Codes C2	Visual Basic	VANET connectivity
Change_wlan.bat	PC-BOX (copy on the SS)	Annex Codes C3	Visual Basic	Change WLAN
Remote_Desktop.au3	PC-BOX (copy on the SS)	Annex Codes C4	Autoit	Connect to CX22-W through Remote Desktop
B2SS.bat	PC-BOX (copy on the SS)	Annex Codes C7	Visual Basic	PC-Box to Shared Server data transference
Separator.bat	SS (PC-Technician)	Annex Codes C8	Visual Basic	Separator
Counter.bat	SS (PC-Technician)	Annex Codes C9	Visual Basic	Counter
Append_TS_X.bat	SS (PC-Technician)	Annex Codes C10	Tool Command Language	Append Time Signal X
Analysis.bat	SS (PC-Technician)	Annex Codes C11	Tool Command Language	Analysis
Post_analysis.bat	SS (PC-Technician)	Annex Codes C12	Visual Basic	Post-Analysis

VEHICLE TO BOX, V2B: VANET CONFIGURATION

Once the vehicle arrives into the Box area the DAQ Stop trigger is executed. At this point every TS recorded is stored at the CX22-W Memory Card. The vehicle is ready for a new acquisition but first every data from the CX22-W is wirelessly transmitted into the PC-Box.

In parallel, on the PC-Box there is a command running called VANET connectivity. This command is in charge of looking for the right network condition in order to start the activation of the VANET, Vehicular Ad hoc[2] Network, which is already configured for both CX22-W and PC-Box computers.

Before getting deep into how the CX22-W node and its WLAN has been managed through an algorithm, a brief theory of MANETs and general information about WANETs is presented.

Brief Theory of WANETs and MANETs

A wireless Ad hoc network (WANET), is a computer network in which the communication links are wireless. The network is Ad hoc because each node is willing to forward data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity. [12]

A particularly WANET case is a MANET, Mobile Ad hoc Network, where nodes are in movement, and if the node is set in a vehicle, it is called VANET, Vehicular Ad hoc Network.

Ad hoc nets are created for a specific function, they do not need any expensive infrastructure to be installed and allow a quick distribution of information. Nevertheless they do not implement any network access control so this makes a quite vulnerable network by a malicious node.

Several applications using Ad hoc networks are being developed by the main automobile companies, Audi for instance, but also by the European Union, such as the Car-2-Car project. In the near future vehicles will “talk” to each other sending and receiving information in order to make the conductivity more secure and avoid accidents or traffic jams. [13][14]

2 Ad hoc is a Latin phrase meaning "for this".

This purposes are identified under the nomenclature V2V, Vehicle to Vehicle, or C2C, Car to Car, but in order to control more variables vehicles will also communicate to static nodes on the road, so V2R, Vehicle to Roadside or V2X for a general purpose case. Eventually, roadsides will be connected to the internet to send information and, for example, get a feedback from the traffic controlling service to communicate vehicles an alternative route due to an accident or a traffic jam.

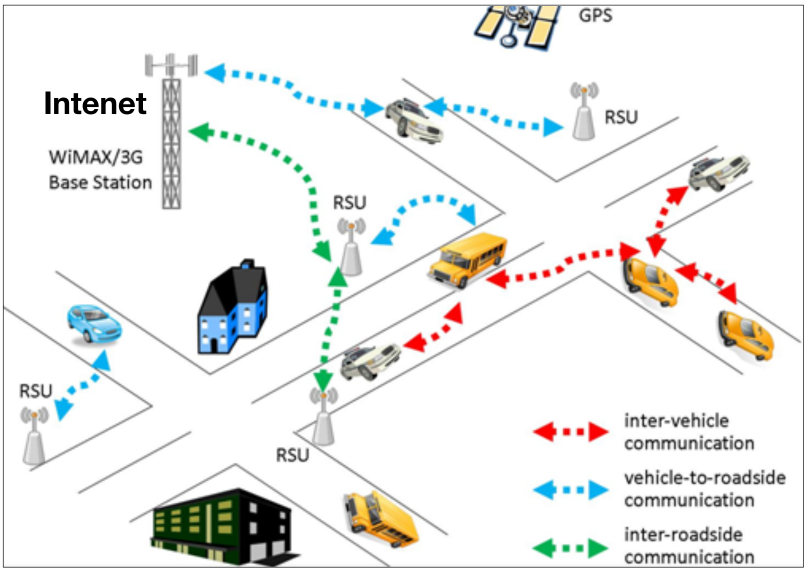


Fig. 35 V2V, V2R and I2R Ad hoc networks .

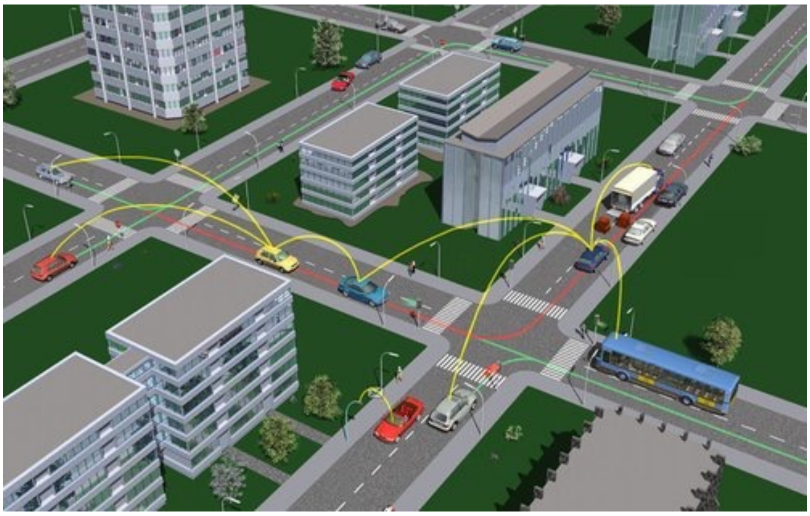


Fig. 36 Car-2-car project by the European Union.

Study of the Network Environment

On the ADAQ project one of the two nodes is the CX22-W which is dynamically ported by the testing vehicle. The characteristics of its WLAN are defined by the standard IEEE 802.11n, which operates on 2.4GHz and 5GHz bands, allowing a maximum net data rate from 54 Mbit/s to 600 Mbit/s. [15]

The CX22-W network is not the only useful WLAN as the PC-Box also connects to the Company Network at a specific time in order to transfer data to the Shared Server, this is the second node, which is static.

The connectivity algorithm has to deal with these two wireless nets. These two WLANs and the other "unpleased" networks from the testing area, if their existence is detected, can be managed in the PC-Box through the Wireless Nets Administrator that Microsoft offers. The order stablished is the following: [16]

1. Company Network (automatic connection if the network is visible)
2. CX22-W Ad hoc WLAN
3. Other "unpleased" WLANs

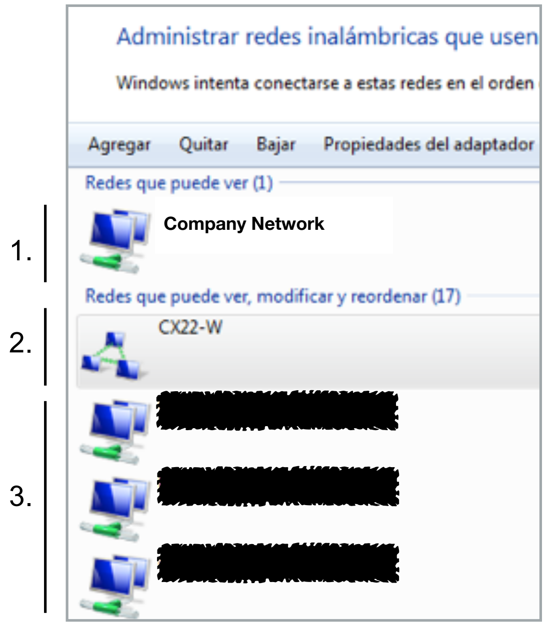


Fig. 37 Wireless Nets Administrator on the PC-Box.

Every wireless network that has access to the Internet follows a standard interconnection method called the Internet Protocol, version 4 and lately version 6, IPv4/IPv6, which was first implemented for the ARPANET project by the American Defense Department in 1983. The creation of the IPv6 is due to IPv4 number of addresses is extinguished since 2011. One of the reasons IPv4 have run out is because it was not given a single IPv4 to big companies, but a range of addresses. This is exactly what it is been noted on the SEAT Company Network. That is why a previous study from the IPv4 Company Network range has been done in order to implement the final connectivity Algorithm. [17]

The IPv4 has the following range: from 0.0.0.0 to 255.255.255.255. Depending on the purpose shorter ranges are defined. The Company Network is included on the Private Networks range which comprehends the 10.0.0.0/8 addresses, so from 10.0.0.0 until 10.255.255.255. The IPv4 address observed is variable with the time so any address from the subrange given to the company can be operating at a specific moment.

The identified Company Network IPv4 range is 10.sn.x.x where sn is a specific number between 0 and 255 (it is not defined following the confidentiality remark), and x are randomly assigned values on time. This knowledge is useful in order to implement the connectivity algorithm as the Company WLAN is identified by the following fraction of IPv4: 10.sn.

The CX22-W also works with a IPv4, not IPv6. It is a Link-local address WLAN, it has reserved the address block 169.254.0.0/16, so from 169.254.1.0 to 169.254.254.255. A specific and single address from the range is operating the CX22-W interconnections.

These addresses change on the PC's IPs configuration report depending on which WLAN it is linked to. This IPs report is where the algorithm looks to in order to execute the right action depending on the networks status in an instant of time.

The following table sums up the two WLANs operating the ADAQ communications.

Table 4 The two nodes ADAQ communications has to deal with.

SSDI	TYPE	STANDARD	IPv4
Company Network	Private Network	802.11g	10.sn
CX22-W	Link-local address	802.11n	169.254.sn.sn

The possible network status in an instant of time or network casuistry is the following.

1. PC-Box not connected to any of the two WLANs. IPs report sends a void IPv4.
2. PC-Box connected to the Company Network. IPs report sends IPv4= 10.sn.
3. PC-Box has the VANET active waiting for the CX22-W node. IPs report sends a void IPv4.
4. PC-Box linked to the Link-local CX22-W WLAN. IPs report sends IPv4= 169.254.sn.sn.

The following screenshots of the available networks panel on the PC-Box shows the different status.

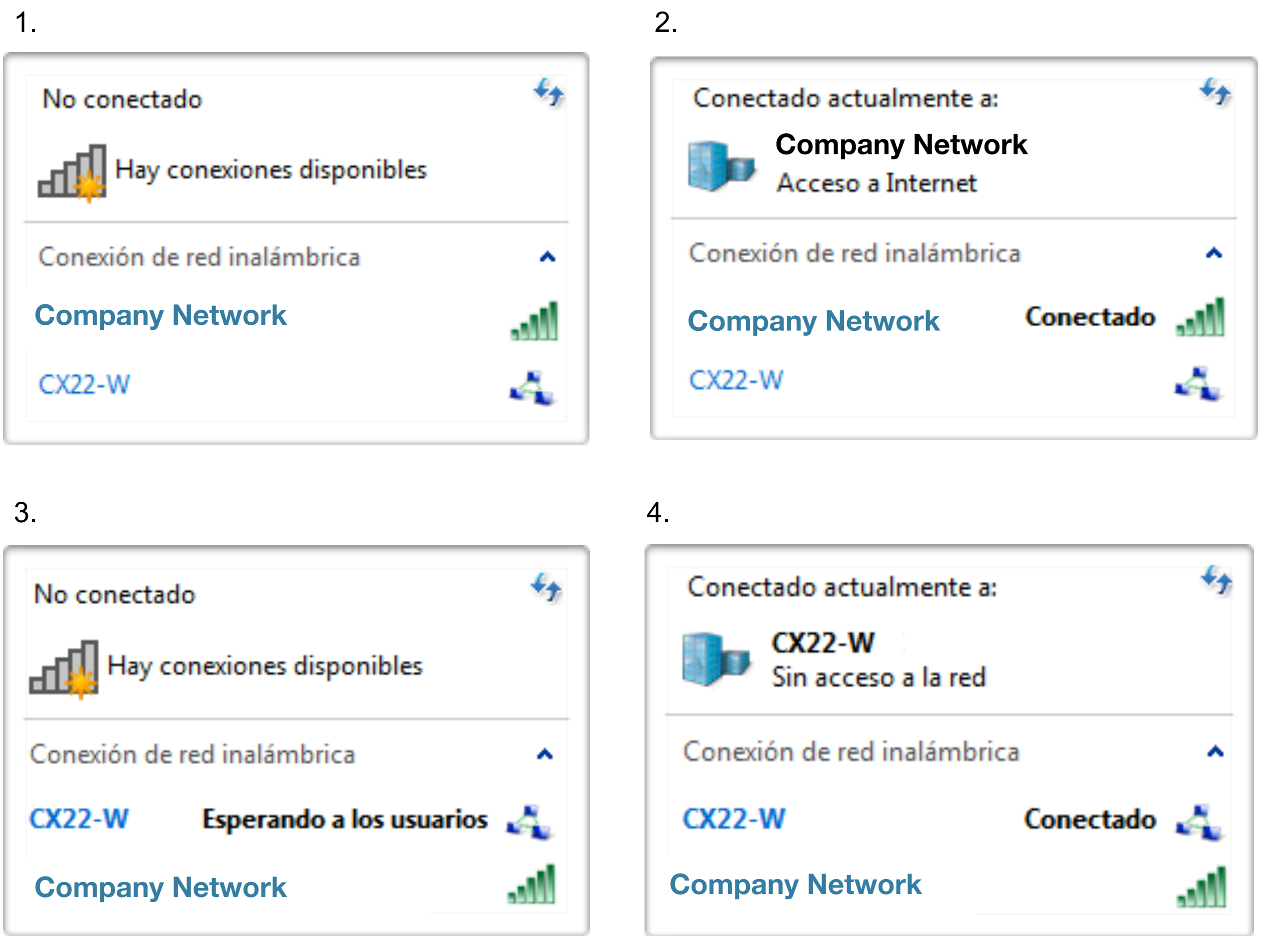


Fig. 38 Different available networks panel on the PC-Box.

During data acquisition process it is important to control network is at status 3, then it will change to status 4 as soon as the CX22-W node is detected, so every time the vehicle arrives to the Box.

At a specific time, once the daily acquisition is completed, the algorithm will check and change, if it is necessary, into state 2 in order to transfer data from the PC-Box to the Shared Server.

Connectivity Algorithm

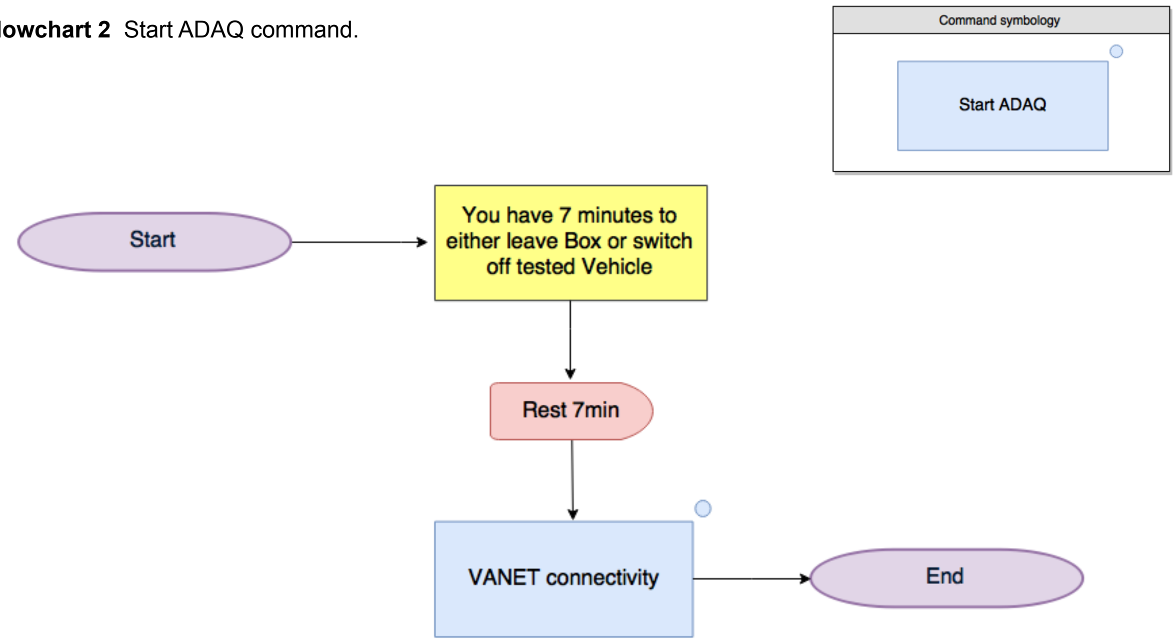
The first step before coding the algorithm is configuring the Ad hoc network on both PC-Box and CX22-W Windows 7 and Windows XP systems respectively; the algorithm is run on both systems and combines different commands. [18]

PC-BOX

On the PC-Box a total of 4 files or commands compound the algorithm. They can be divided in one initiator command, one identifier command and two execution commands.

- The initiator command shows a message warning that ADAQ communication process will start in 7 minutes, this is the time the pilot has to either turn off the vehicle or leave the box.

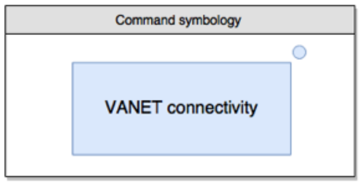
Flowchart 2 Start ADAQ command.

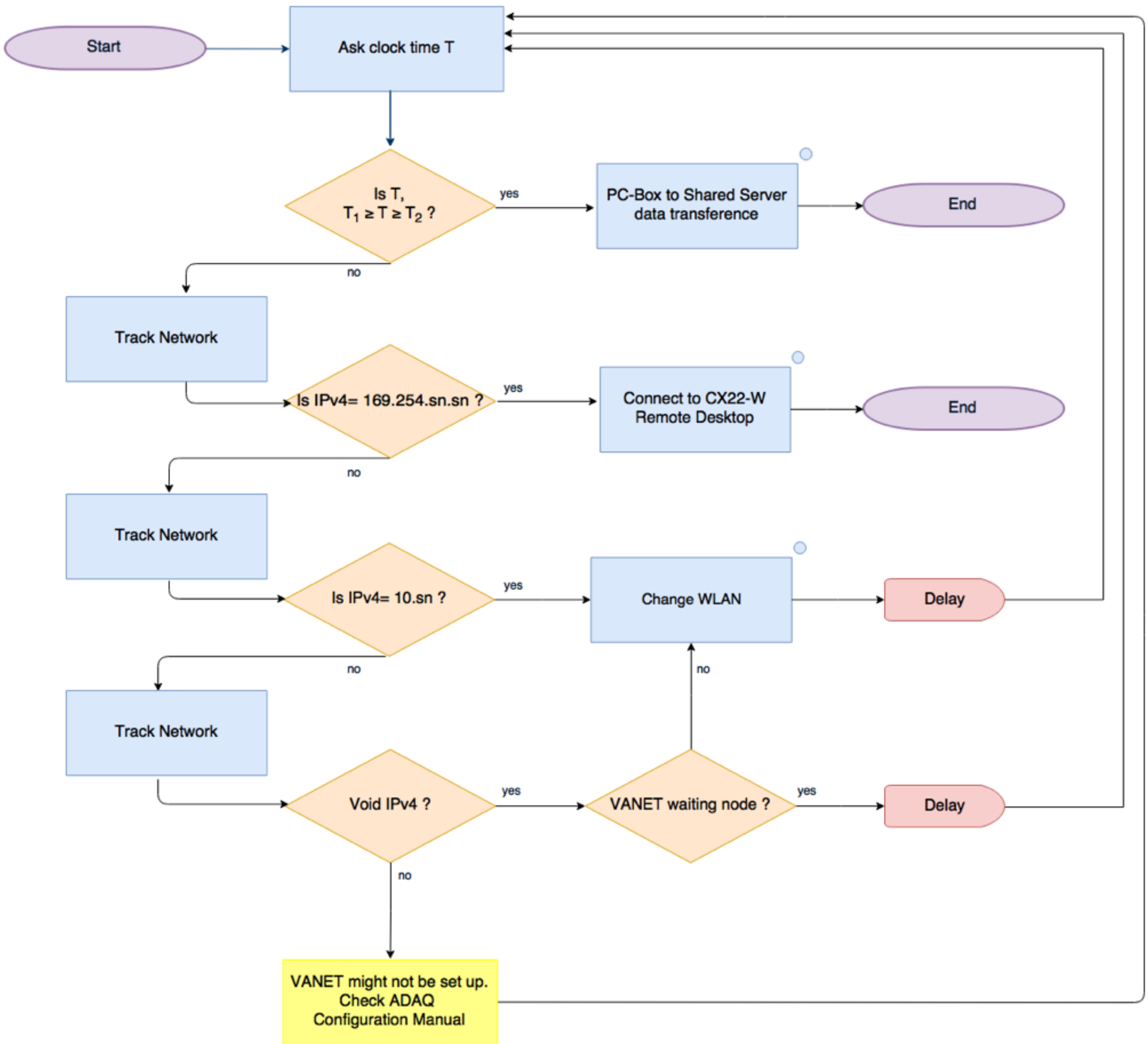


After 7 minutes, before closing, it activates VANET connectivity identifier command.

- The VANET connectivity identifier command rules the whole ADAQ communications. It works in a loop and its inputs are the time, the IPv4 address and the SSID the system is connected to. VANET connectivity command checks the current IPv4 address, through the IPs report on the CMD, and also checks if the SSID assigned to the CX22-W is connected or not, through the WLANs display.

Flowchart 3 VANET connectivity command.





On the sequence they are identified and controlled the 4 possible status. As long as it is not time to transfer data into the Shared Server, when time T is between T_1 and T_2 , the algorithm will change into status 3 and stay on this loop, waiting for the node, in case no WLAN is connected (status 1) or in case Company Network is connected (status 2).

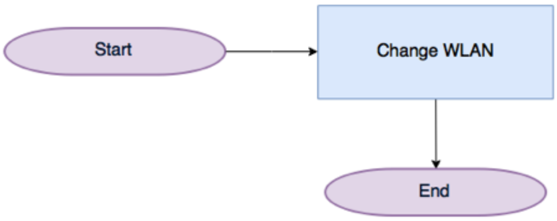
Logically the algorithm might not have a complete sense, cause after the second network tracking, if the answer is negative, so when none of the two IPv4 is connected, it could be automatically assumed the IPv4 is void, and directly ask whether the CX22-W SSDI is waiting for the node or not.

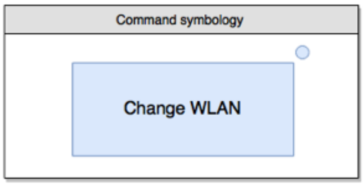
Despite between every track of the network few seconds pass, the network status could have change so it is recommended to do not assume any state and ask specific questions, due to the volatility of the network.

As it is shown on Flowchart 2 two more commands are involved on the V2B communication. They are the executor commands which are activated by VANET connectivity command. 'Change WLAN' command executes the action of connecting to the CX22-W network. 'Connect to CX22-W Remote Desktop' command is in charge of accessing to the CX22-W from the PC-Box in order to start the data transmission, meanwhile the vehicle remains in the Box. A relevant difference between the two commands is 'Change WLAN' is executed while VANET connectivity command is running, while 'Connect to CX22-W Remote Desktop' is started by VANET connectivity just before this command finishes.

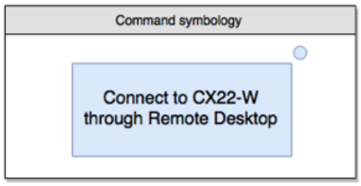
• 'Change WLAN' executes the change on the network status. Basically changes state 1 into 3, or state 2 into 3.

Flowchart 4 Change WLAN command.

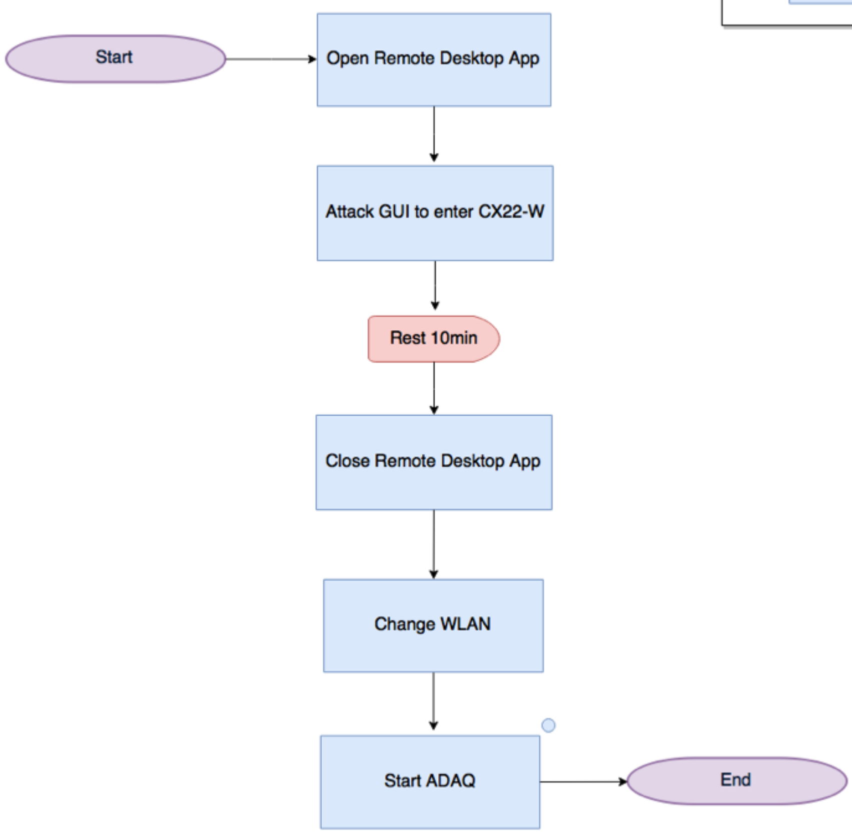




• 'Connect to CX22-W Remote Desktop' is executed by 'VANET connectivity' command as soon as the VANET is active, so when IPv4 is detected to be the one of the CX22-W. This command is build up in Autoit language as some of its coding lines require actions impossible to do programmatically as credentials and permissions are not owned. So it operates directly to the GUI. In order to avoid errors while executing the actions due to external inputs this command disables the keyboard and mouse functions before starting the action.



Flowchart 5 Connect to CX22-W through Remote Desktop command.



Step by step the actions implemented by the command are:

· Open Windows Remote Desktop Application.

· Start running the App under the CX22-W IP Address.

· Introduce CX22-W log in credentials; user name and password.

· Wait 10 minutes for all TS to be transferred into PC-Box.

· Close Remote Desktop App, connect to Company Network, start 'Start ADAQ' command.

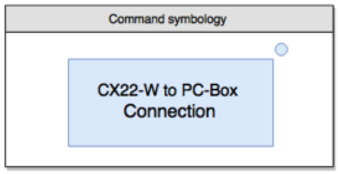
· End command.

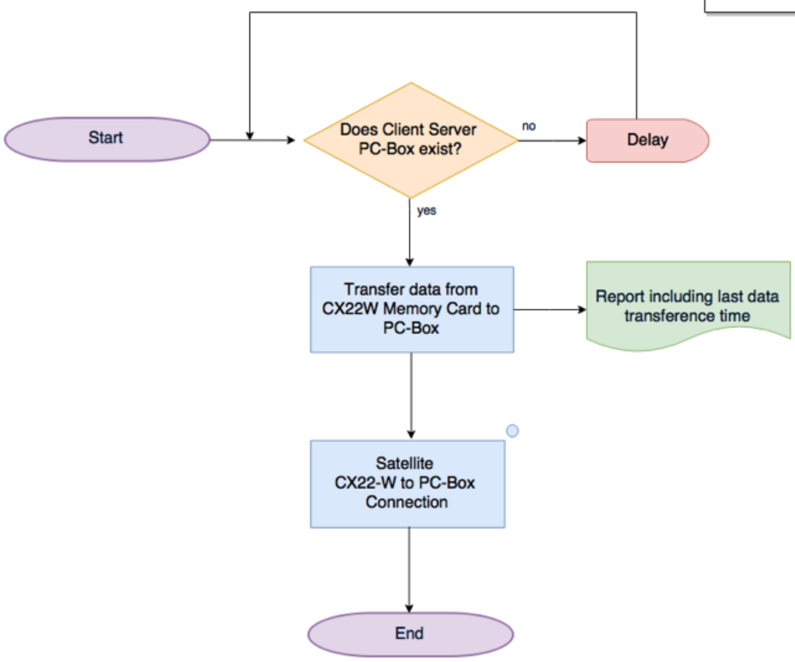
CX22-W

In parallel on the CX22-W two commands are continuously running. The first one is an identifier and executor command, while the second command is only an identifier command. They are calling each other on a continuous loop since the very first moment WinXP on the CX22-W is loaded till the system is switched off, so since turning on the vehicle until turning it off.

• 'CX22-W to PC-Box data transference' command is looking for the right condition in order to transfer every data contained on the CX22-W Memory Card to the PC-Box. It is in a continuous loop searching if PC-Box is visible. At the point 'Connect to CX22-W Remote Desktop' PC-Box command enters the CX22-W Remote Desktop, 'CX22-W to PC-Box data transference' gets out of the searching loop and starts transferring every TS into the folder 'Canal_YYYYY', on the PC-Box.

Flowchart 6 CX22-W to PC-Box Connection command.

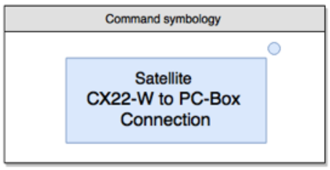


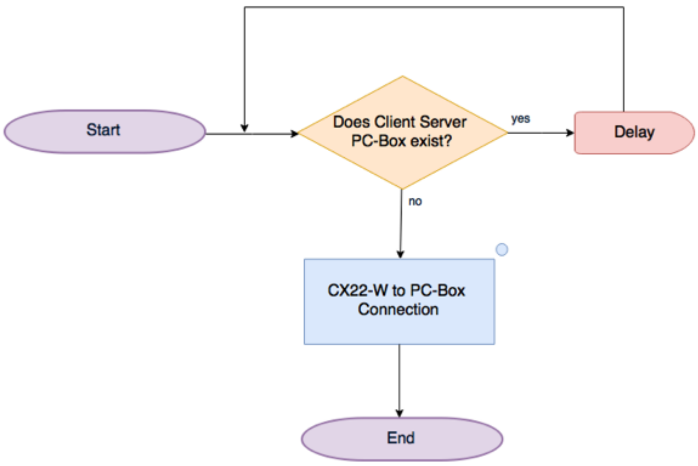


This command outputs a simple report called “info_transference.txt” including the date and time when last data transference was done. Before closing the command activates ‘Satellite CX22-W to PC-Box data transference’.

• ‘Satellite CX22-W to PC-Box data transference’ identifier is the complementary command from the previous one. It also recognizes the existence from the PC-Box and does not get out of the loop until the PC-Box is not visible, so when the Remote Desktop application is finished. At this point the command will activate the previous command.

Flowchart 7 Satellite CX22-W to PC-Box Connection command.





The vehicle is ready to leave the Box and both CX22-W and PC-Box commands are waiting for the next vehicle arrival to the Box in order to start its interaction and make possible wireless data transference one more time.

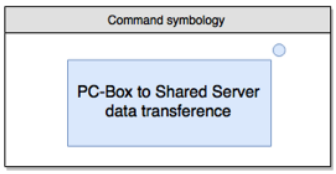
B2SS: PC-BOX TO SHARED SERVER

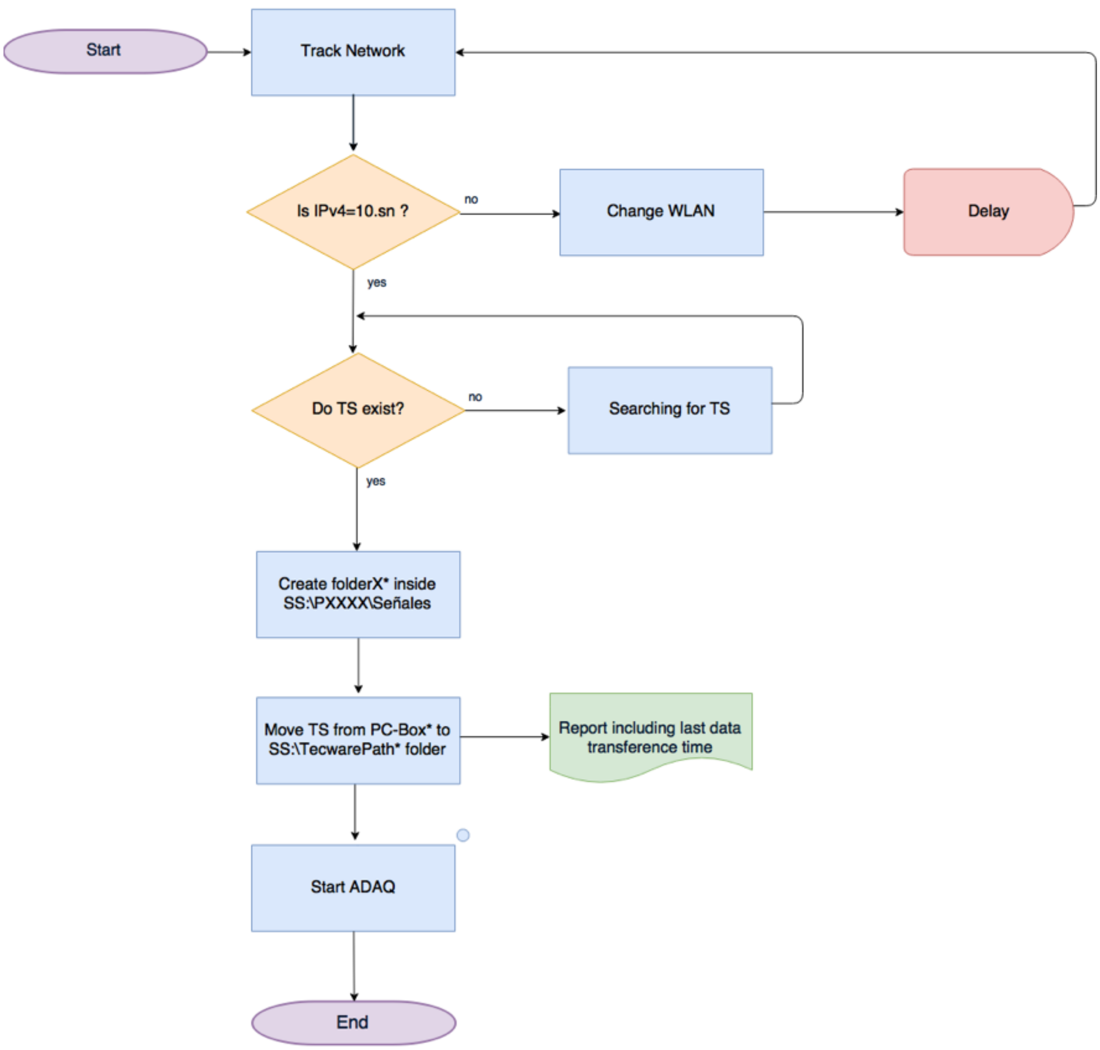
PC-BOX

At specific time, once data acquisition process is completed, every data will be transferred into the Shared Server in order PC-Technician can analyze the TS recorded during the day. The following commands do not only transfer TS into the SS but also prepare those TS on the right way for the analysis Software LMS Tecware can start its analysis.

At the configured time, 'VANET connectivity' command will activate the command 'PC-Box to Shared Server data transference'.

Flowchart 8 PC-Box to Shared Server data transference command.





The first action from the command is to connect into the Company Network in case it is not connected. Once in the right network the Shared Server is visible and available. Next action from the command is checking if the acquisition and data transference was correctly done. If no errors where found TS will be moved into 'SS:\TecwarePath' folder on the Shared Server. At the same time the folder including project ID and data acquisition day will be created inside 'SS:\PXXXX \Señales'. A simple report called "info_transference.txt" including the date and time when last data transference was done will be saved on PC-Box. After data transference the command 'Star ADAQ' will be executed; in an ordinary acquisition project the vehicle will be already turned off.

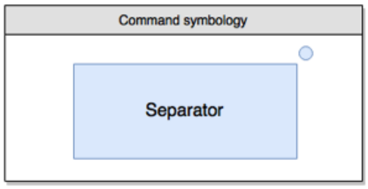
PC-Technician

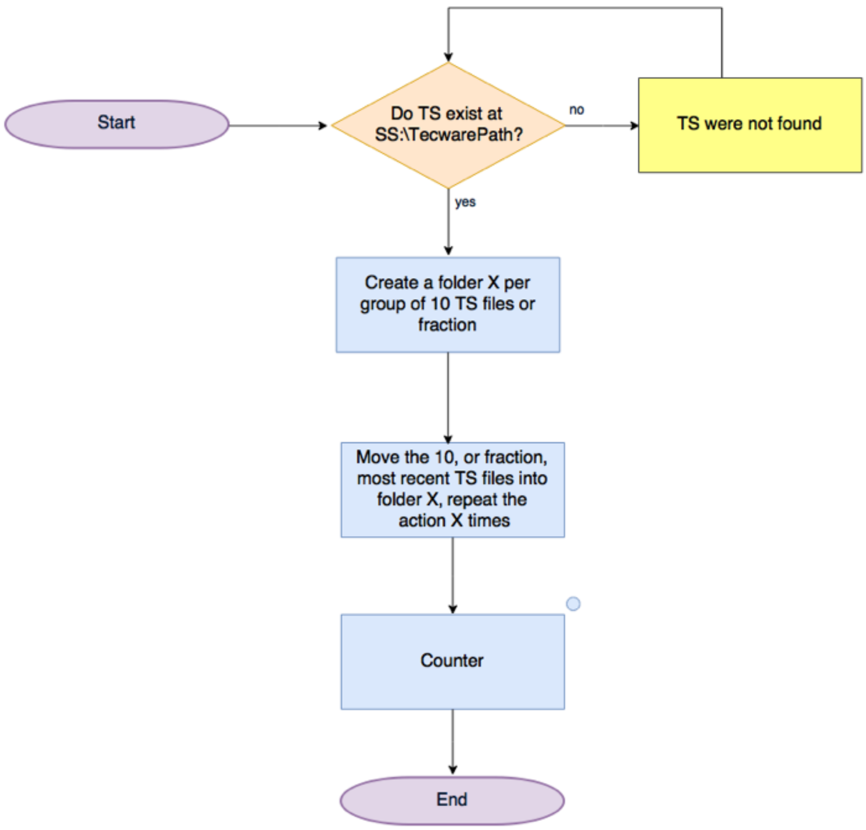
Once data is located on the Shared Server the PC-Technician is ready to start TS pre-analysis. This process will be started at a programmed time. Windows task scheduler will be set to execute the initiator command at a specific time.

• Pre-analysis commands are in charge of grouping every data in order TS are ready to be appended. It encompasses the commands 'Separator' and 'Counter'.

- 'Separator' command is in charge of classifying TS files contained in TecwarePath in different folders. The command will create as many folders as groups of 10 TS files or fraction exist. Then the 10, or fraction, most recent TS files, the ones last recorded, will be moved into the first folder created. This action will be repeated till every TS is classified on its sub-folder. 'Separator' command will work correctly as long as no more that 1500min, 25 hours, of TS are recorded (so 150 TS files), in this extreme case 15 folders will be created, each containing 10 TS files. Last action from the command is executing 'Counter' command.

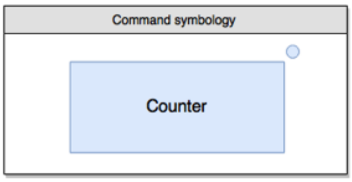
Flowchart 9 Separator command.

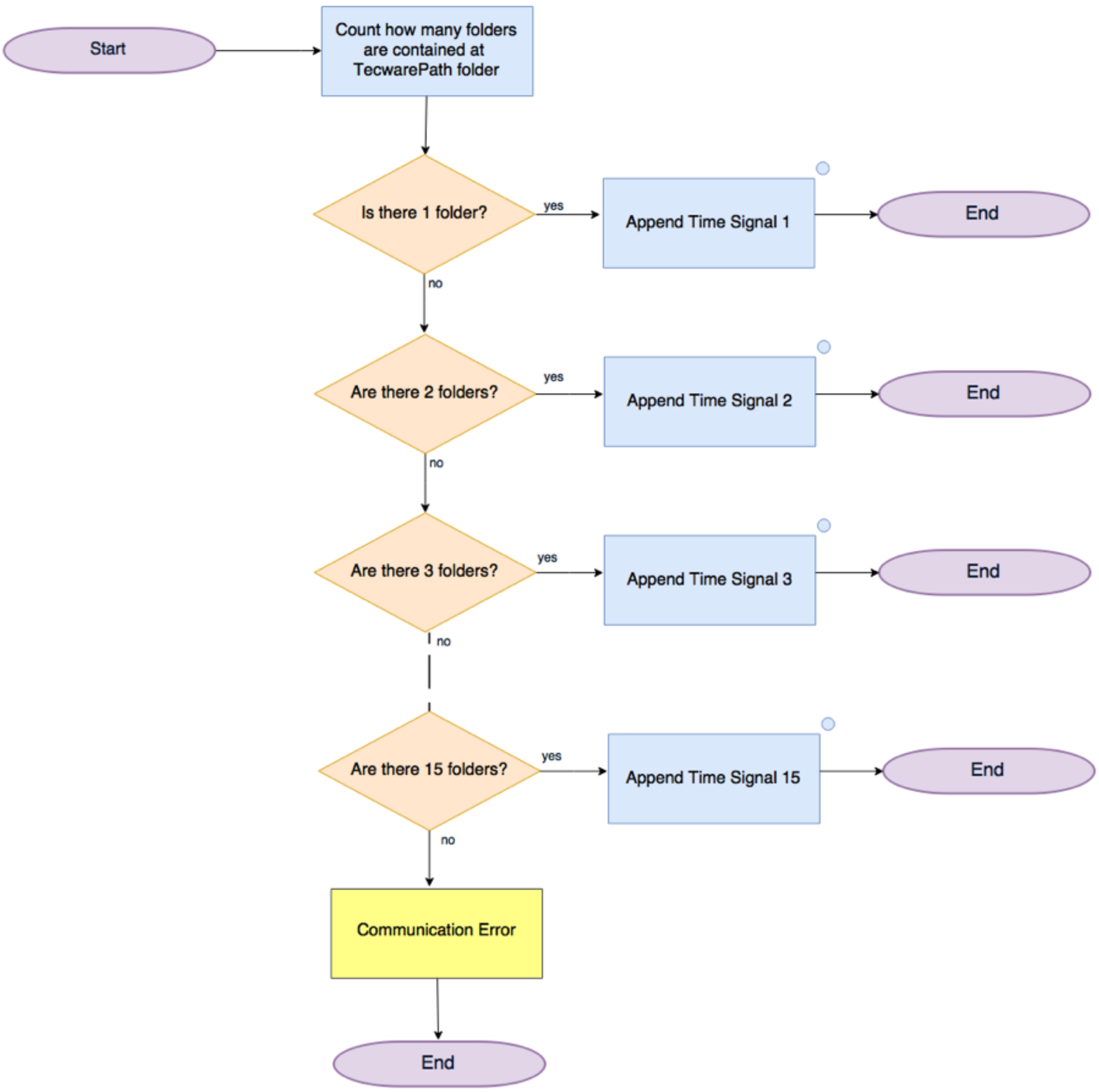




- 'Counter' command will count how many sub-folders are contained inside TecwarePath folder. Depending on the amount of data this command will decide which append TS routine fits better. This command will execute the action that directly orders the software analysis: LMS Tecware. There are 15 possible append routines, so the command will match the number of folders inside 'SS:\TecwarePath' with its corresponding routine.

Flowchart 10 Counter command.





2.3.4 ANALYSIS BY LMS TECWARE PROCESSBUILDER

INTRODUCTION

Once data is classified in sub-folders it is ready to be analyzed. First action before starting its analysis is to append TS. Every TS file contained on the sub-folders will be appended into a one single file with length from 10min, a single TS, until 100min, 10 appended TS files.

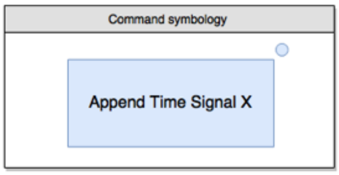
After Append TS routine is completed, a command will order the first analysis to begin, and so on after the whole analysis process is completed. The type and amount of analysis, for the selected channels, is chosen by the technician at the configuration of the project. Some of the analysis routines are compulsory, they are the anomalies detection and the driven kilometers report. On this chapter they are explained all of the analysis routines, the 2 compulsory ones and up to 6 possible analysis routines to be done.

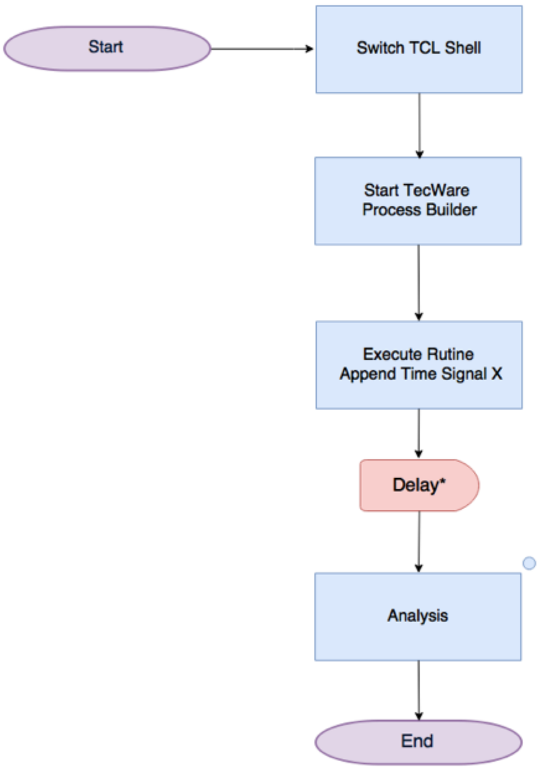
Once the last analysis is completed a command will be in charge of reorganize the whole data, classify reports to the right folder, delate original TS files, store appended files and delate extra files the analysis software outputs during its routines. To sum up what the post-analysis actions do is to reset TecwarePath folder to the initial conditions and ready to receive the next set of recorded TS.

APPEND TS

'Append Time Signal X' routine comprehends 15 different routines; according to the dimension of data from the acquisition day one routine or the other will be activated. The command that activates this routine has the following structure.

Flowchart 11 Append Time Signal X command.





First thing the command does is changing the language into TCL, then starting the software analysis and activating the correspondent routine 'Append Time Signal X'. The Delay* is not a configured time; it depends on the same process of appending TS. After this routine is done, it activates the next command: Analysis.

Every 'Append Time Signal X' routine uses the same process blocks. Append Time Signal 1 will use only one of this process block, while Append Time Signal 15 uses 15 process blocks working in parallel. Depending on the 'Counter' command one routine or the other will be activated.

All of these routines are already configured to make the import from data from the right folder. Append Time Signal 1, for instance, will point to 'TecwarePath' subfolder named '1', while Append Time Signal 15 will point to 'TecwarePath' subfolders: '1', '2', '3'… '15'. Definitely, the right routine will be executed for every case in order that it uses the data from all of the previously created subfolders in 'TecwarePath' folder.

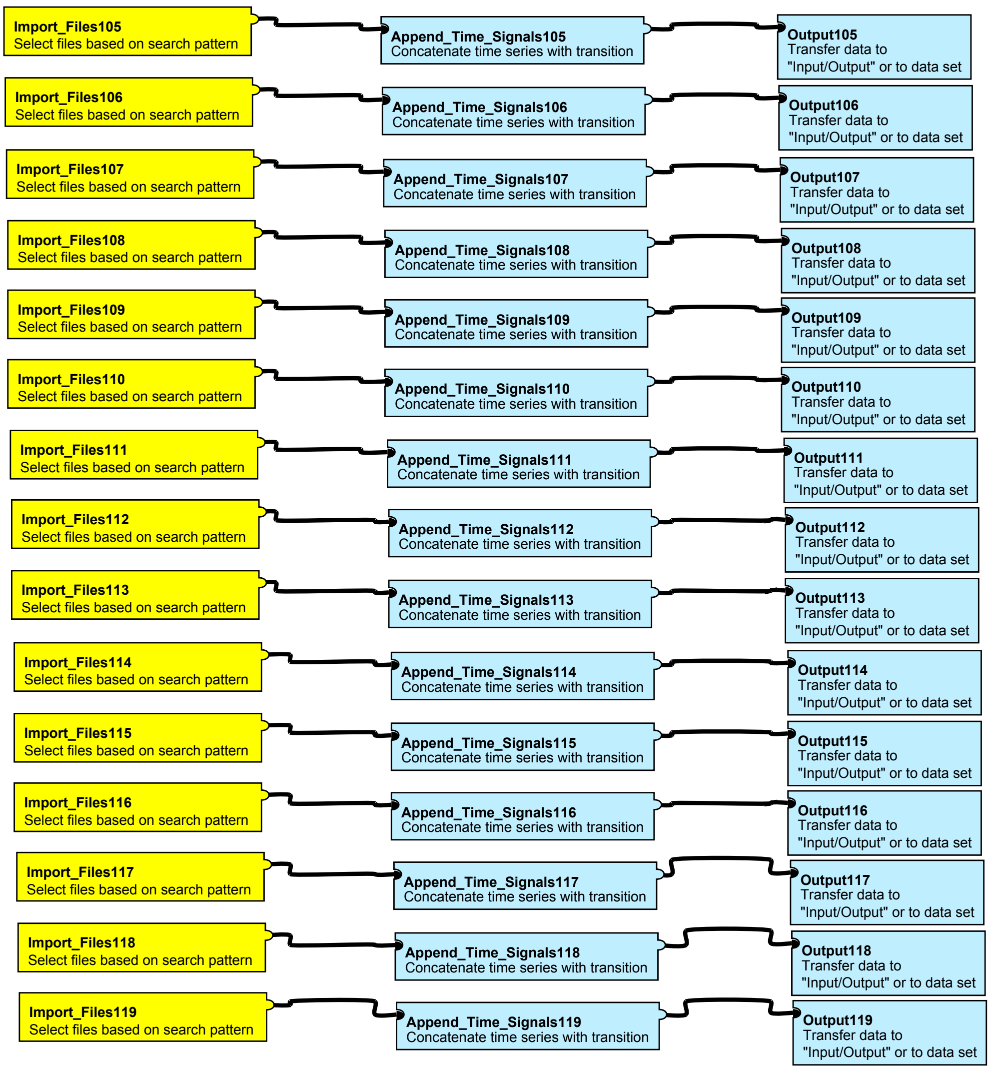


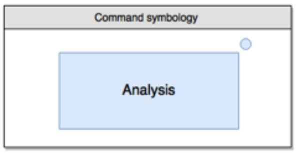
Fig. 39 'Append Time Signal 15' routine to append 1400 to 1500min of acquisition.

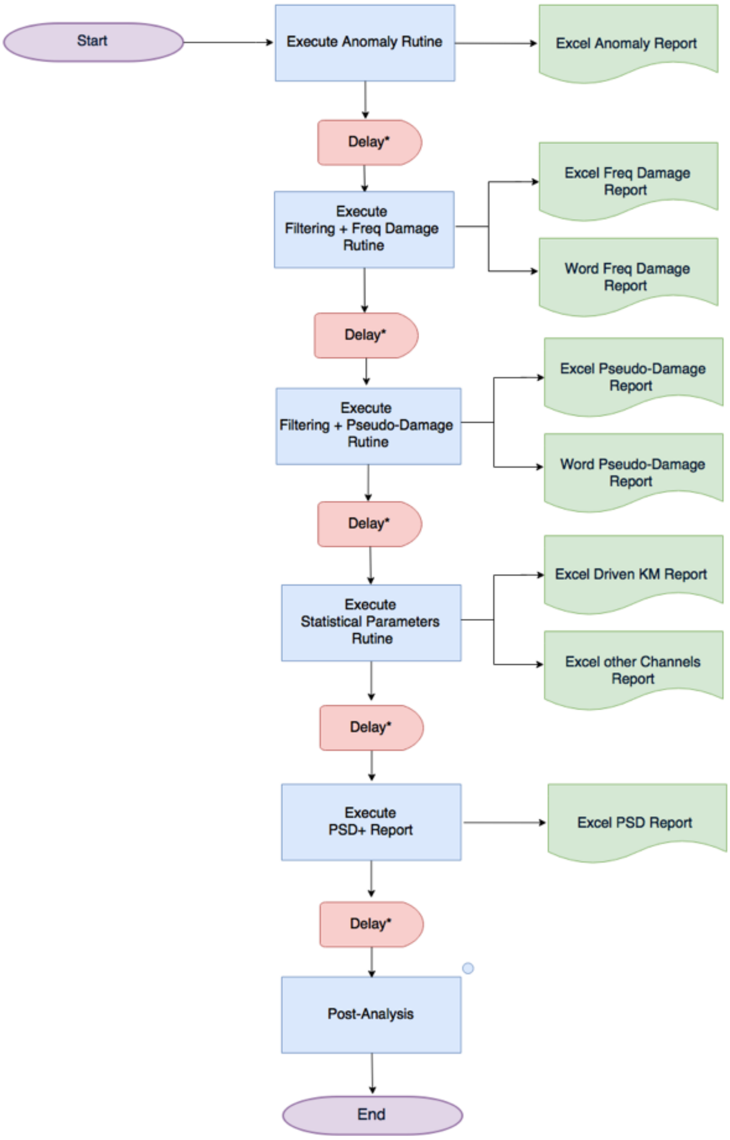
ANALYS ROUTINES

Once the Append TS routine has finished it activates 'Analysis' command. This command is created while the technician is configuring the ADAQ project and will contain every analysis the technician selected. Apart, it will contain the two mandatories analysis routines, the anomaly detection routine and the driven kilometers statistics.

The following command states the selection from every available analysis by the technician.

Flowchart 12 Analysis command.





Note Delay* time is again determined by the duration of the analysis. A few explanation from every generate report will be stated.

Anomalies

Is executed for every channel from every appended file. This it how it looks.

Table 5 Anomalies report with an example of every problem detected.

Time History Name	Ch#	Channel Description	Unit	Problem
TS 1	1	Channel 1	g	ImproperHeader
TS 2	2	Channel 2	m/m	Unipolar
TS 3	3	Channel 3	m/m	Spikes
TS 4	4	Channel 4	m/m	Saturation
TS 5	5	Channel 5	C	Flat Line
TS 6	6	Channel 6	C	Unipolar
TS 7	7	Channel 7	C	Overrange

Every anomaly or problem can be configured as the technician decides for every specific ADAQ project.

Statistics (Driven KM report)

Table 6 Statistics Driven KM report example.

			Chan1	
			Kilometers	time
Channel	Channel Name		Km	s
1	Kilometers	SV Min	3	0
		SV Max	63	3600

It states the initial KM from the TS file, the final KM for the same TS and the elapsed time between this two values. Following the table, the total completed KM is 60Km (63-3).

Pseudo-Damage

The pseudo-damage report shows, from the top to the bottom and from left to right, the appended TS for the specific channel, some statistic values for the channel (in red the relevant one, the pseudo-damage value of the original TS). Also offers a PSD graphic comparing the original and the filtered TS (to see whats the predominant frequency from the instrumented point of study of the vehicle). Then rainflow matrix is exposed, this matrix divides in classes the values from the TS obtained by specific method for fatigue data analysis. [19][20]

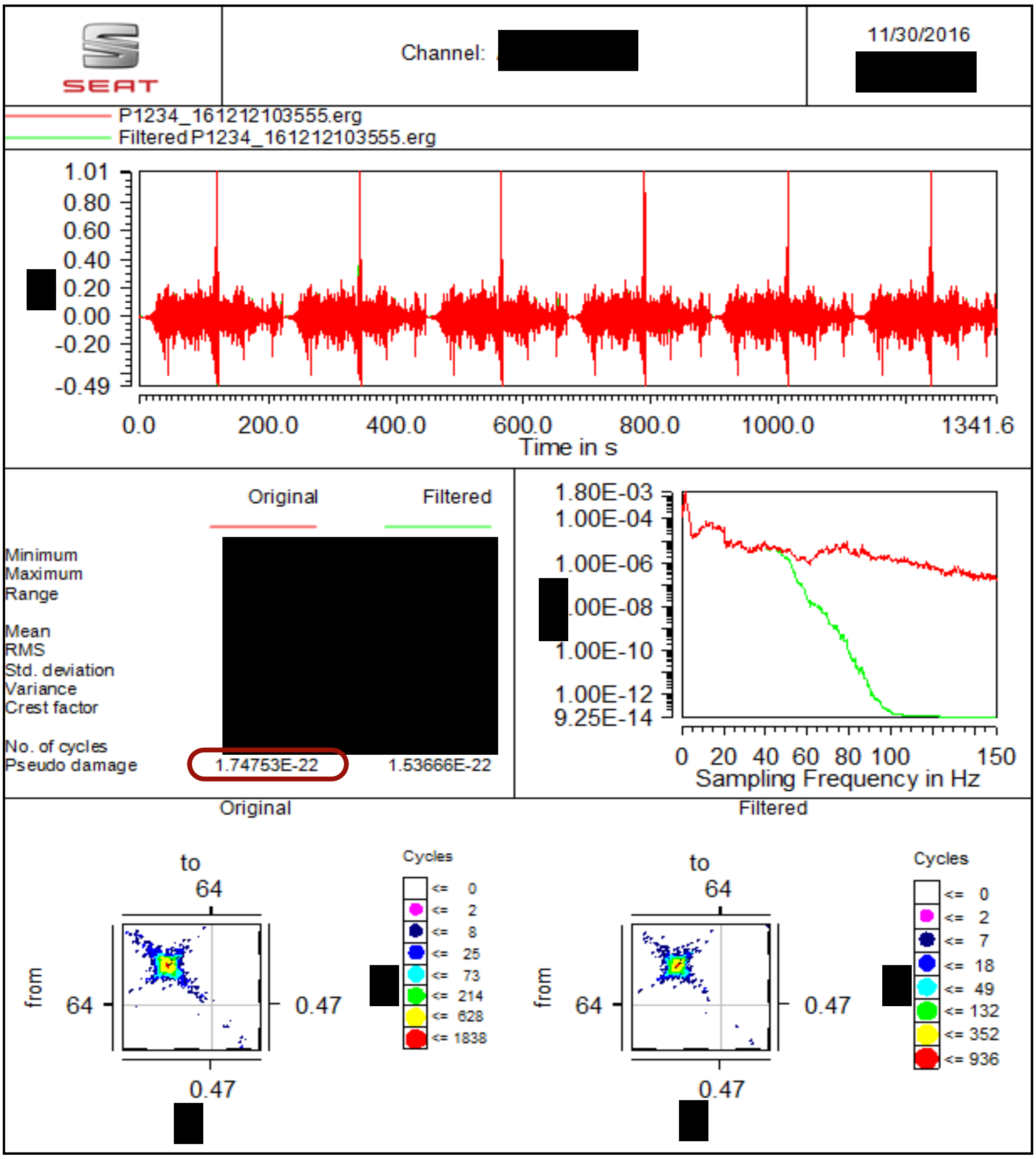


Fig. 40 Pseudo-damage report by LMS Tecware.

Frequency Damage

Frequency damage is a similar concept to pseudo-damage. Frequency-damage divides the total damage from the TS in band frequencies in order to see on what own frequency from the instrumented point of study is produced the highest pseudo-damage. The report shows a PSD per band of frequency and a bar chart of the damage caused per band of frequency.

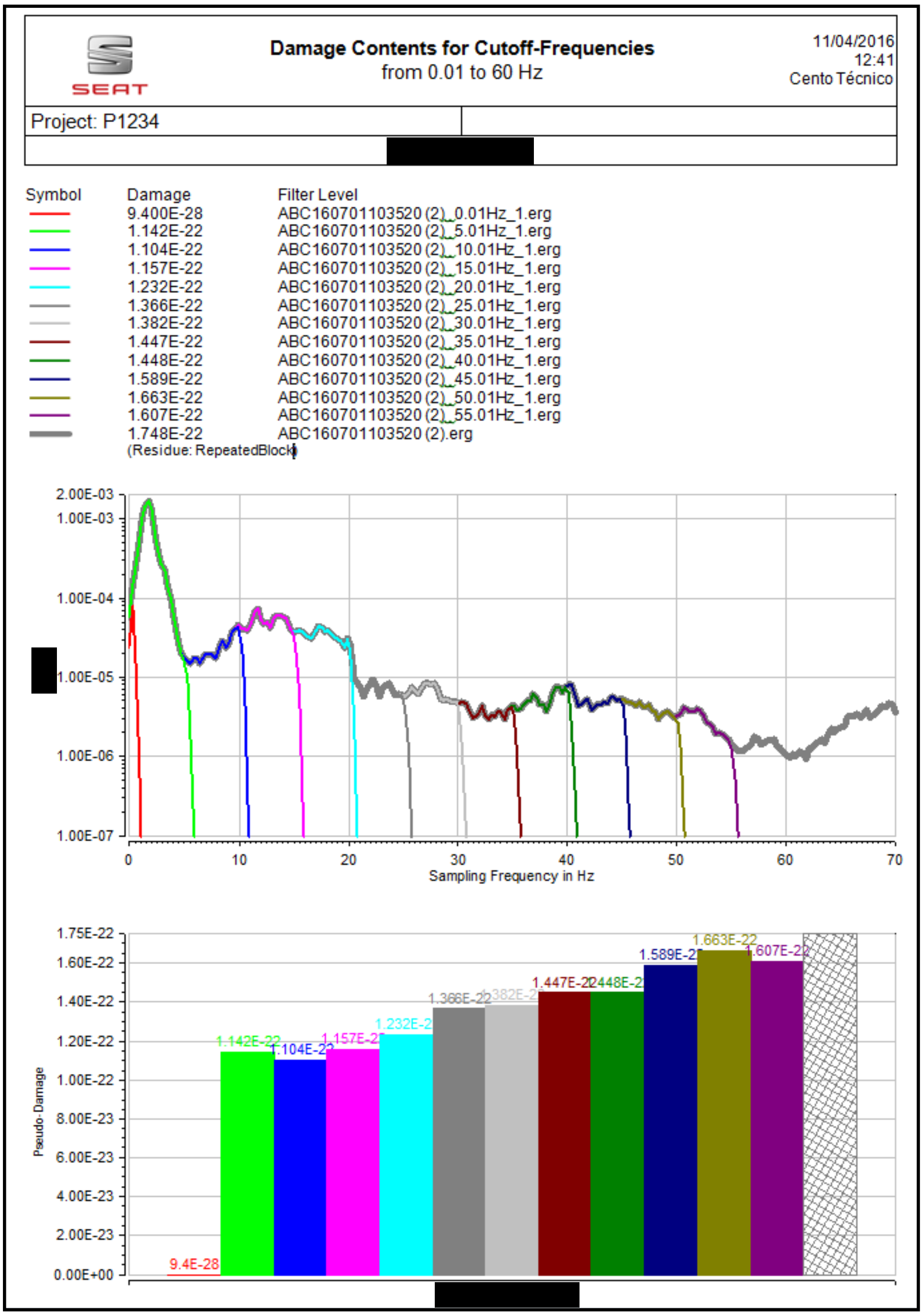
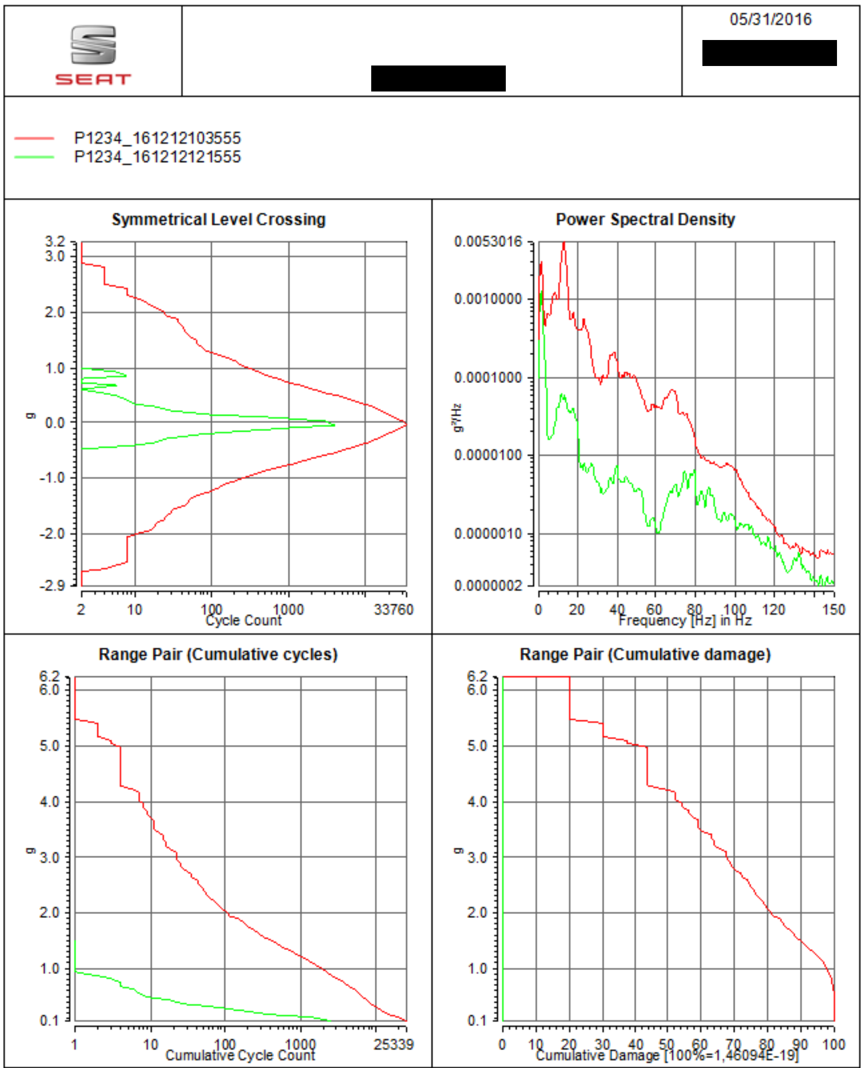


Fig. 41 Frequency-damage report by LMS Tecware.

PSD+

PSD+ report offers the Power Spectral Density and other methods for cycles counting. This methods are the Symmetrical Level Crossing, the Range Pair (Cumulative cycles) and the Range Pair (Cumulative damage). Relevant statistics form the TS are also offered on the bottom of the report. More information from the reports and its theory can be found on LMS Tecware Manual volume 4.

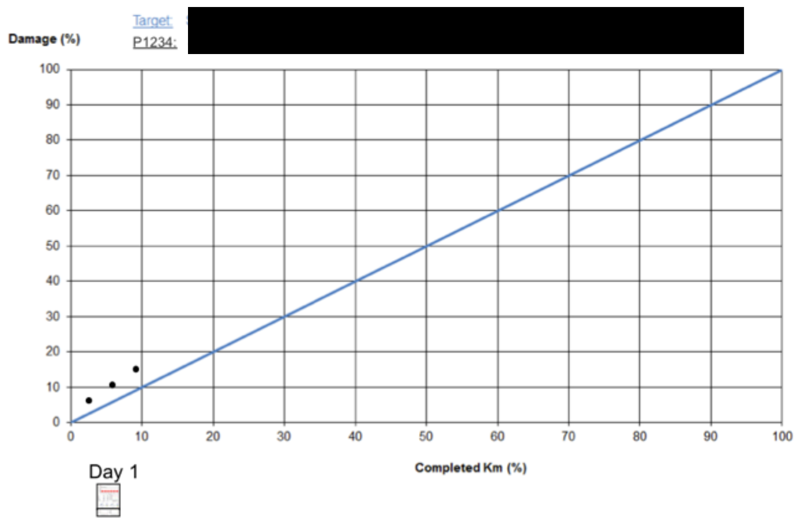


Legend Name	Unit	Minimum	Maximum	Range	Mean	RMS	Damage
P1234_161212103555	g	-3.0	3.3	6.2	-0.0	0.2	1.46E-19
P1234_161212121555	g	-0.5	1.0	1.5	-0.0	0.1	3.07E-23

Fig. 42 PSD report by LMS Tecware.

Cumulative Pseudo-Damage

This report is a XY graphic where X axe are KM and Y axe is the cumulative pseud-damage. This report determines the evolution during the test from the instrumented point of study and is compound by the information provided from the daily reports previously exposed. Basically, the graphic will be updated every day with two values, the completed KM by the daily acquisition and the pseudo-damage caused in this distance. The evolution from the graphic will be compared with the target value selected at the configuration of the project. This graphic can compare the evolution from the pseudo-damage caused to different points of study, to different vehicles with different configurations, and also in different testing tracks of different lengths.



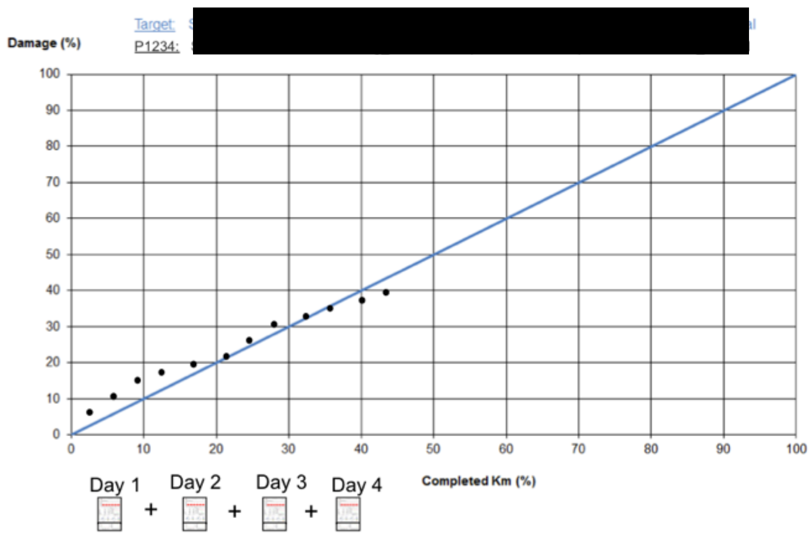
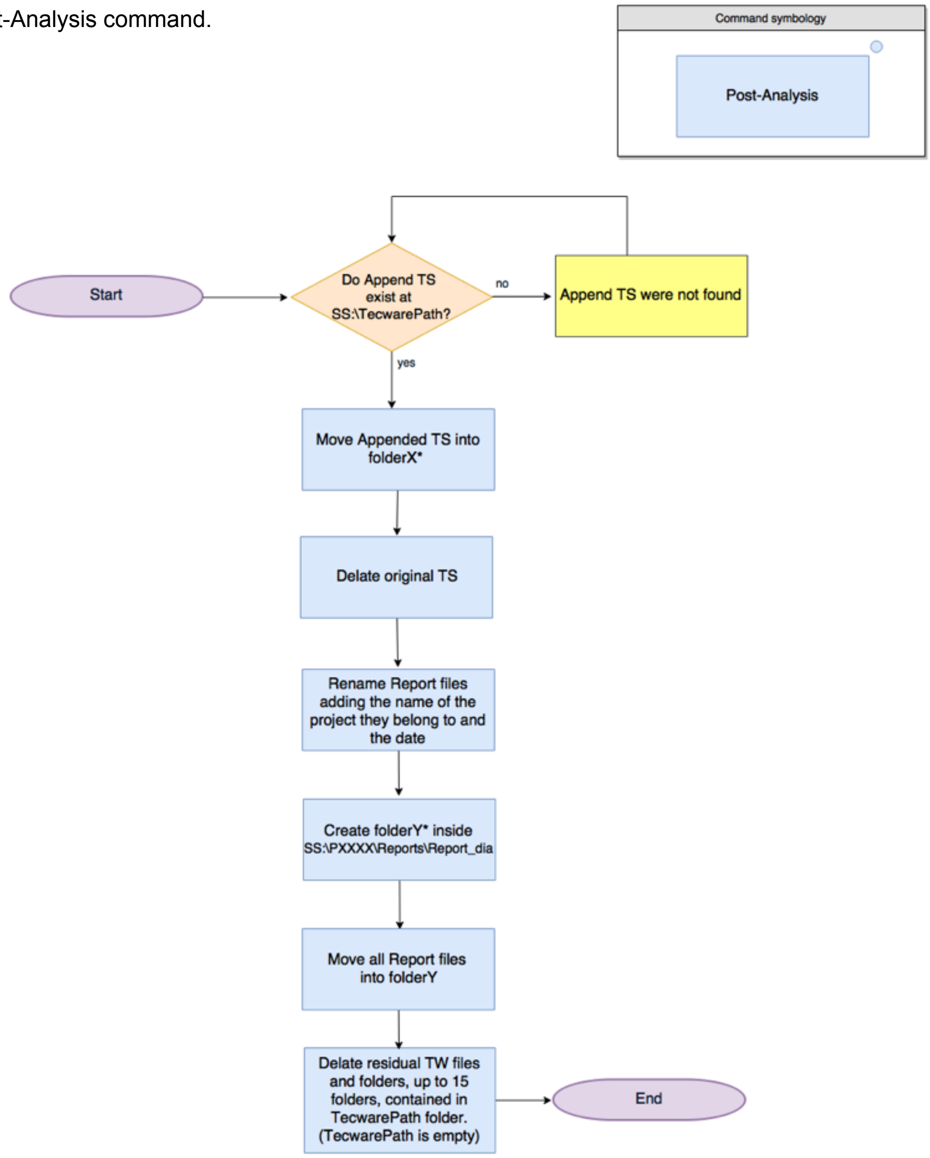


Fig. 43 Evolution from day 1 to day 4 of acquisition of the cumulative damage compared with a target value.

Post-Analysis

After every daily report is generated, the command 'Analysis' activated 'Post-analysis' command. This command is in charge of renaming the Word/Excel reports generated including the date of the acquisition on the file name and sending them form TecwarePath folder to the right subfolder of daily reports inside the project's folder they belong to. The same happens with the TS, orinal ones are delated while the appended TS are transferred to the daily TS subfolder inside the project's folder they belong to. Extra files generated from software analysis LMS Tecware are delated. and subfolder from 1, up to 15 are delated too. To sum up, every information is classified to its right place and 'TecwarePath' folder is released for the next data acquisition.

Flowchart 13 Post-Analysis command.



folderY*: Specific name is YYMM(DD-1); meaning the year, month and day before today, so when the adquisition was done.

CHAPTER IV

PROJECT MANAGEMENT

2.4 PROJECTS MANAGEMENT

Once setting every single phase from ADAQ and making all of them work synchronously, is time to make them easy to configure by the technician. Thanks to the application designed with software LabVIEW, the technician creates every command, data infrastructure and relevant project information by only typing few words. While the technician is filling the gaps or selecting the desired analysis, the app, behind its GUI is executing instructions on real time that create the necessary folders, commands, excels, etc in order the technician has on the right data base the project entered into the app. This app is being run on the PC-Technician and it not only points to Windows in order to create every data infrastructure but also communicates with the software analysis LMS Tecware, Google Maps to get the geographical coordinates, with Windows task Scheduler to set the start from the analysis and with the external libraries to import any information to the project.

LabVIEW software offers all of theses possibilities. The software works by the union of different commands that are called virtual instruments (VI). This VIs are compound by a front panel which is the GUI and by the structure or code that the developer from the VI has designed. For this app, the technician only has access to the executable from this VIs, so the GUI (also designed by the developer). Behind every screen from ADAQ app one or several VIs set the functions to every button, gap to fill in and selectable from the front panel.

On this app are identified 114 VIs and subVIs, each one with a specific function. Due to the complexity from the VIs codes and the global structure from the app only a general idea from the app is presented. In order to make this understandable, only a few VIs have been chosen in representation from the whole VIs.

2.4.1 VIs INTRODUCTION

LabVIEW follows a data flow model for running VIs. A block diagram node is ejected when it receives all the required inputs. When the node is run, it produces output data and passes the data to the next node in the data flow tray. The data movement through the nodes determines the execution order of the VIs and the functions in the block diagram.

Every command (batch file) from ADAQ is generated thanks to a block diagram node of an uncompleted code divided per coding lines that is completed and ejected when this block diagram receives the required inputs, so the gaps that the user fills in on the GUI. This explains every VI referred to outputted commands from the application. It can be clearly seen on the Annex VI11 and VI12.

Other VIs need also logic conditions in order make sure every required specification for a new project is being introduced. Every button and gap to fill in from the app front panel generally work on a loop, so the VI data flow is not going on until the button is pulsed or the gap is filled in. When this is done a sequence of actions will be triggered. [21]

2.4.2 VIs PRESENTATION

The representative VIs of the app and the corresponding front panel to each one is presented on this table. The table from top to the bottom also shows how data flow advances through the VIs.

Table 7 Virtual Instruments (VI) presentation.

VI Name	VI icon	VI code (not all of it)	VI GUI Screen
Start_ADAQ.vi		-	Annex VI Screen 1
Project_ok.vi		Annex VI3	Annex VI Screen 2
Vehicle_specifications.vi		Annex VI1 & VI2	Annex VI Screen 3
Track_target.vi		-	Annex VI Screen 4
Control_usb.vi		Annex VI4	Annex VI Screen 5
Modules_selector.vi		Annex VI8 & VI9	Annex VI Screen 6
Channel_selector.vi		Annex VI6 & VI10	Annex VI Screen 7
Analysis.vi		Annex VI7 & VI11	Annex VI Screen 8

2.4.3 VIs INPUTS/OUTPUTS

As it is said the app generates every data infrastructure and relevant information for a new project. ADAQ app also calls external libraries or the internet, it can recognize external USB memories and also has direct access the softwares that need to be configured on every ADAQ project.

Table 8 Virtual Instruments (VI) inputs/outputs.

VI Name	Inputs	Outputs
Start_ADAQ.vi	Selecting New Project/Open project.	None
Project_ok.vi	Introducing Project name.	Creation project folder and every sub-folder.
Vehicle_specifications.vi	Project name from the previous VI. Introducing vehicle specifications. Excel with specifications from another project.	Creation from every command that rules ADAQ. Excel file and screenshot with this information saved on the project's folder. Printed front panel.
Track_target.vi	Selecting testing track and target pseudo-damage value from ADAQ data base.	Excel file and screenshot with this information saved on the project's folder.
Control_usb.vi	Identifying information from previous VI. Insert USB into the PC-Technician.	Copy into the USB of the commands that rule ADAQ on the CX22-W.
Modules_selector.vi	Identifying information from previous VI. Selecting the QuantumX modules for the project and the number of channels of each type.	Excel file with this information saved on the project's folder. Printed front panel.
Channel_selector.vi	Identifying information from previous VI. Filling the gaps with channels specifications. Charging channels specifications from data base. Number and position of channels to be analyzed.	Direct access to Google Maps. Excel file with this information saved on the project's folder. Printed front panel.
Analysis.vi	Identifying information from previous VI. Channels name to be analyzed. Analysis to be done for each channel	Excel file with this information. Direct access to LMS Tecware. Direct access to Win Task Scheduler.

2.4.4 VIs FUNCTIONALITY AND SPECIFICATIONS

On the following table are stated the specifications by logic functions form every VI and a general view of how do they work and under what circumstances do they allow data flow.

Table 9 Virtual Instruments (VI) functionality and specifications.

VI Name	Functionality and Specifications
Start_ADAQ.vi	Data will not flow unless any button is being selected. At selecting 'New Project' the next VI is executed.
Project_ok.vi	When project name is written and OK button is selected the next VI is executed. If project name is not introduced and button OK is selected appears a message notifying is not possible to advance.
Vehicle_specifications.vi	When every green gap is written and 'NEXT' button is selected the next VI is executed. 'OPEN FILE' button opens Windows and waits for an Excel file to be chosen. 'CLEAR' button delates every written information except the project name. 'PRINT' button prints the screen without the buttons box. If any from the green gaps is not introduced and button OK is selected appears a message notifying is not possible to advance.
Track_target.vi	'VIEW' and 'OBJECTIVE DAMAGE' buttons trigger a window from Win7. 'IMPORT' buttons opens Windows and waits for an Excel file to be chosen. When both requests are imported and 'NEXT' button is selected the next VI is executed.
Control_usb.vi	A previously created batch file is executed by Track_target.vi. The VI has no conditioning specifications but it is written on it to select 'NEXT' when the Win console has closed, so few seconds later after introducing an USB memory.
Modules_selector.vi	It is not possible to advance unless the MX840 is selected as the first module and the CX22-W as the last module. Also CAN bus channels must be associated to the first MX840 module while GPS channels are associated to the CX22-W. Only one CX22-W can be selected.
Channel_selector.vi	It is not possible to advance unless every channel specification is completed. It is not possible to advance unless channels to be analyze are not defined. Is not possible to charge a channel from the library unless column of 'TRANSDUCER' is empty.
Analysis.vi	When the selection is done and 'NEXT' button is selected LMS Tecware will be opened with only the selected routines of analysis, after LMS Tecware is closed the Windows Task Scheduler will be opened. After Windows Task Scheduler is closed the VIs are totally stopped.

CHAPTER V

EXPERIMENTATION & RESULTS

2.5 EXPERIMENTATION AND RESULTS

ADAQ system has been tested several times and on three official presentations, to the HBM company, to different departments from the company's Technical Center and to the manager from all these departments. The test done consisted on leaving the Box and start acquiring inside the company's internal roads, at a sample rate of 300Hz for all channels, then getting back into the Box and repeat this action again. After this, at a specific time, sending data into the company's shared server and executing append TS routine, anomaly detection and finally executing kilometers and pseudo-damage analysis. Results have been satisfactory and are the following ones:

Table 10 ADAQ results from the tests done.

TIME OF THE DAY	WHERE	ACTION
08:00:00	**Vehicle**	The pilote turns on the vehicle. Red LED light on the dashboard is on.
08:04:20	**Vehicle**	Acquisition computers and specific project is booted. No channel errors. Red LED switches off and the green LED is on.
08:04:20	**PC-Box**	The Technician starts ADAQ 'VANET connectivity' command on the PC-Box. The whole ADAQ process is initiated. PC-Box screen turns green.
08:04:20	**Vehicle**	The pilote stars moving the vehicle towards the Box exit.
08:04:25	**Vehicle**	The vehicle leaves the Box. DAQ start condition is triggered. Green LED is now intermittent. Computers are acquiring.
08:11:20	**PC-Box**	PC-Box screen turns red. The computer is waiting for the vehicle.
08:50:00	**Vehicle**	During data acquisition an event is happening. The intermittent red LED is on. At least one of the dumpers has reach a temperature of 90ºC or higher. The pilote decides to stop the vehicle.
08:57:00	**Vehicle**	The red intermittent light is gone. The pilote gets back into the testing track.
10:05:00	**Vehicle**	The vehicle gets into the Box. Green intermittent light is now permanent.
10:05:30	**Vehicle**	The pilote turns off the vehicle through the vehicle keys. Green LED is still on thanks to the timer.
10:06:00	**PC-Box/ Vehicle**	PC-Box detects CX22-W WLAN and starts linking to it.
10:07:20	**PC-Box/ Vehicle**	The connection is now done. PC-Box connects to CX22-W remotely.

TIME OF THE DAY	WHERE	ACTION
10:07:40	PC-Box/ Vehicle	CX22-W is being seen on PC-Box screen. Now CX22-W command sends data from CX22-W Memory Card into PC-Box. CX22-W command is ready to send data again on the next acquisition.
10:09:00	PC-Box	PC-Box command exits the remote access to CX22-W. And disconnects from CX22-W WLAN. Green screen on PC-Box; after 7 min the command will starts looking for CX22-W WLAN again.
10:15:30	Vehicle	After 10 minutes the timer stops allowing power supply from the vehicle battery to the acquisition computers. The green LED light is off. Every electronic on the vehicle is now off.
10:16:00	PC-Box	VANET connectivity' command is activated. It is searching CX22-W WLAN but the vehicle is turned off and its node too.
10:30:00	Vehicle	The pilot turns on the vehicle through the key. Red LED on the dashboard is on and PC-Box screen is also red.
10:31:00	PC-Box/ Vehicle	CX22-W WinXP is booted and activates the node. PC-Box starts linking to CX22-W.
10:32:20	PC-Box/ Vehicle	The connection is now done. PC-Box connects to CX22-W remotely.
10:32:40	PC-Box/ Vehicle	CX22-W is being seen on PC-Box screen. CX22-W tries to send data into the PC-Box but no data is found. The process continues normally.
10:34:00	PC-Box/ Vehicle	PC-Box command exits the remote access to CX22-W. And disconnects from CX22-W WLAN. Green screen on PC-Box; after 7 min the command will starts looking for CX22-W WLAN again.
10:34:40	Vehicle	Acquisition computers and specific project is booted. No channel errors. Red LED switches off and the green LED is on.
10:34:40	Vehicle	Both lights green, vehicle LED and PC-Box screen. The pilote stars moving the vehicle towards the Box exit.
10:34:45	Vehicle	The vehicle leaves the Box. DAQ start condition is triggered. Green LED is now intermittent. Computers are acquiring.
11:30:00	Vehicle	Channel X is manually unplugged. Catman detects the disconnection from a channel. Green intermittent LED is off. Red light is on. DAQ stop condition has a post-trigger of one second. So after channel disconnection the acquisition still records for one more second and stops.
11:30:01	Vehicle	DAQ Stop trigger is activated.
11:30:00	Vehicle	The pilot goes towards the Box as no data is being recorded.
11:35:00	Vehicle	The vehicle gets into the Box. The pilote turns off the vehicle through the key. Red permanent light is still on. The pilot's contribution has finished.
11:35:30	PC-Box/ Vehicle	PC-Box detects CX22-W WLAN and starts linking to it.
11:36:50	PC-Box/ Vehicle	The connection is now done. PC-Box connects to CX22-W remotely.

TIME OF THE DAY	WHERE	ACTION
11:37:10	**PC-Box/ Vehicle**	CX22-W is being seen on PC-Box screen. Now CX22-W command sends every recorded data until one second after the channel error. Transference is from CX22-W Memory Card into PC-Box. CX22-W command is ready to send data again on the next acquisition.
11:38:30	**PC-Box**	PC-Box command exits the remote access to CX22-W. And disconnects from CX22-W WLAN. Green screen on PC-Box; after 7 min the command will starts looking for CX22-W WLAN again.
11:45:00	**Vehicle**	After 10 minutes the timer stops allowing power supply from the vehicle battery to the acquisition computers. The red LED light is off. Every electronic on the vehicle is now off.
11:45:30	**PC-Box**	'VANET connectivity' command is activated. It is searching CX22-W WLAN but the vehicle is turned off and its node too.
11:45:30	**PC-Box**	Every data is on the PC-Box right now. A total 175min 18sec have been recorded. This means PC-Box contains 18 TS files of 10min or fraction.
12:30:00	**PC-Box**	PC-Box stops CX22-W WLAN search. Red screen turns off. 'PC-Box to Shared Server data transference' command is activated. It changes into company's WLAN and sends every TS file into the Shared Server at folder TecwarePath.
12:40:00	**PC-Technician**	Windows Task Scheduler on PC-Technician activates 'Separator' command. Two folders are created inside TecwarePath folder. First folder contains the 10 most recent TS files the second folder contains the 8 remaining files.
12:41:01	**PC-Technician**	'Separator' command activates 'Counter' command. It counts how many folders exist at TecwarePath. 'Counter' counts two so it activates 'Append Time Signal 2'
12:42:02	**PC-Technician**	LMS Tecware is activated on a hidden mode and routine 2.pb starts.
13:05:00	**PC-Technician**	2.pb routine outputs two files of TS on TecwarePath folder. First one is 100min of recorded TS and the second one is of 75min 18sec. Append Time Signal 2 activates 'analysis' command as 2.pb routine has finished.
13:05:00	**PC-Technician**	'Analysis' command activates anomaly routine.
13:30:00	**PC-Technician**	Anomaly routine outputs an Excel into TecwarePath folder. The Excel contains an anomalies check to all of the channels from both TS files. 'Analysis' command activates kilometer stadistic routine when it detects anomaly routine has finished.
13:39:00	**PC-Technician**	Kilometer statistic routine outputs an Excel into TecwarePath folder with the total distance completed by the vehicle for both TS files. 'Analysis' command activates pseudo-damage routine when it detects kilometer stadistic routine has finished.
14:30:00	**PC-Technician**	Pseudo-damage routine outputs two Word files into TecwarePath folder with pseudo-damage value among other information. 'Analysis' command activates 'Post-analysis' command when it detects pseudo-damage routine has finished.
14:30:00	**PC-Technician**	Anomaly, Kilometers and Pseudo-damage files are renamed and send it to a folder identified with the name Daily_report and today's date. Append TS files are sent into TS folder for today's date. Every file and folder on TecwarePath folder is delated (original TS and LMS Tecware extra files).
14:32:00	**PC-Technician**	TecwarePath folder is empty and ready for the next data transference.

TIME OF THE DAY	WHERE	ACTION
15:00:00	**PC-Technician**	The technician arrives into the office. At checking the anomaly report Channel X reports a 'Flat line' anomaly. It is conclude Channel X it's been disconnected or broken. ('Flat line' anomaly is reported when 200 measured points in a row are detected).
15:01:00	**PC-Technician**	Technician also checks pseudo-damage and completed kilometers value. He/She can now introduce this values into the XY cumulative graphic of 'kilometers completed vs accumulated damage' and check the evolution from the specific point of study (e.g channel Y is the studied point).

The transference times from V2B and B2SS have been reported from the tests done. The most unfavorable situation is by dimensioning the acquisition to the limit ADAQ data's infrastructure can host. It is defined 1500min of maximum acquisition length, with arrivals from the vehicle to the Box every 3 hours. The number of acquired channels is of 34 and the sample rate is of 300Hz.

Before starting the analysis the 150 TS files of 10min are appended into 15 TS files of 100min length each one. The standard analysis stablished is the anomaly detection to all of the channels, the statistic kilometers report and the pseudo-damage reporting for only one channel.

Table 11 Transference and Analysis times study.

TYPE	ACTION	MAXIMUM DURATION
Data Transference	V2B wireless transmission from 180 minutes of data acquisition	4min 20sec
Data Transference	B2SS wireless transmission from 1500min of data acquisition	13min 40sec
Data Analysis	Append 150 TS files into 15	110min
Data Analysis	Anomaly detection report	150min
Data Analysis	Kilometers reporting	35min
Data Analysis	Pseudo-Damage reporting	90min
ANALYSIS	ALL	385min

Wireless data transference time is variable since it depends from the network environment status, the distance between the node and the client server and more variables. Data transference information is then, a report from the observations on specific instance of time in order to have an approximate data transference time.

Talking about the TS storage, the 15 TS files (1500min) represente 3.42GB of stored information. So there are two main conclusions from the study of the reported results.

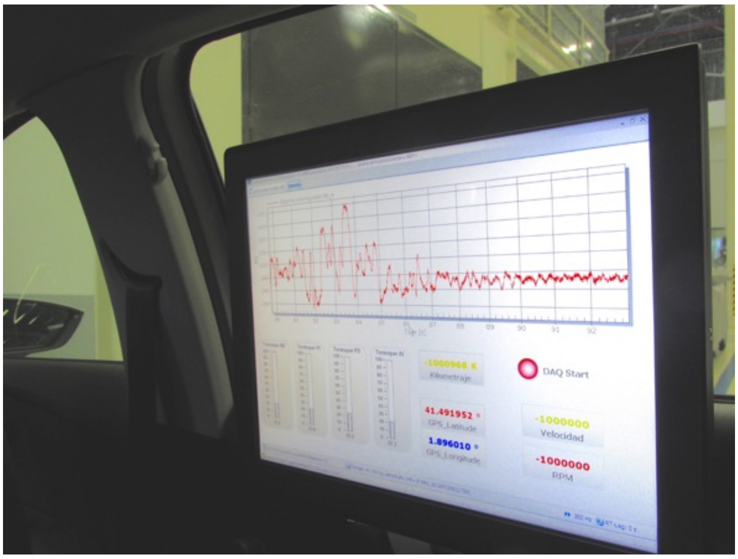
1. Due to the high quantity of resources they represent, TS files will be manually delated by the analyst once it is checked the signal and results are correct. Only reports will be stored.

2. The standard analysis lasts 385min (almost 6.5h). So PC-Technician has capacity for assuming the analysis for only one vehicle per night. That is why an important improvement to ADAQ system would be setting a higher processing speed and RAM memory to PC-Technician, because acquiring an other LMS Tecware software license is more expensive (about 30.000€ per license).

Images from the tests done

```
YOU HAVE 7 MINUTES TO EITHER
1. LEAVE THE BOX
2. TURN OFF THE VEHICLE BATTERY

Esperando 304 segundos, presione una tecla para continuar ...
```




```
VEHICLE IN THE BOX. CX22W CONNECTED CORRECTLY.
COUNTDOWN TO START CX22W REMOTE DESKTOP
.

Esperando  6 segundos, presione una tecla para continuar ...
```

```
TS FILES ON THE SHARED SERVER
ANALYSIS BEING EXECUTED
.
Presione una tecla para continuar . . . _
```

Fig. 44 From top to bottom, ADAQ started on the PC-BOX, vehicle acquiring, vehicle arrived into the Box, PC-Box connecting to the vehicle and no GUI analysis being executed on PC-Technician.

CHAPTER VI

NEXT IMPROVEMENTS

2.6 NEXT IMPROVEMENTS

As it is previously mention, this is the version zero from ADAQ system or ADAQ v.0. In order to go a step further several modifications and improvements have been thought. Some of them are already in progress, other are ideas. Among all improvements a more detailed information will be stated about multi-vehicle ADAQ system.

1. CX22-W integration into the company's network.

- Direct data transmission from the vehicle into the company's shared server. (V2SS)
- Intraday reports few hours later the acquisition is done.
- PC-Box is not necessary. Easier configurations.
- In a multiple vehicle case V2SS transference does not need to be sequential. The different vehicles can transfer its recorded data at the same time.
- Acquisition software Catman can be configured remotely from the headquarters.
- It is not need to use the USB key to configure the ADAQ on the CX22-W. Its configuration would be similar to the PC-Box.

2. Higher processing speed and RAM memory of PC-Technician.

- Faster data analysis and reporting.
- More vehicles can be hosted by PC-Technician's autonomous analysis.

3. Testing track Box infrastructure to host wireless data transmission for multiple vehicles.

- Thanks to the informatics, no more PC-Box needs to be acquired. One is enough.
- Indication of the transmission state to the pilots through a screen divided by stripes.

2.6.1 MULTIPLE VEHICLES ON TRACK, SEQUENTIAL VANETS

The presented solution solves the problem of a fleet of vehicles acquiring on the same testing track area.

As long as two or more vehicles do not coincide on the Box the system would be the same as the already stated for a single vehicle. The system would work as a FIFO.

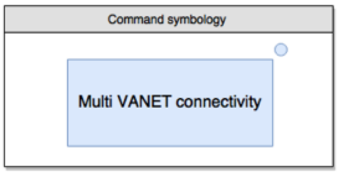
The command implemented on the PC-Box, as most from ADAQ commands, works in a loop. This means there is not a priority connection before another, it just depends on the casuistry of the loop on a specific instant of time. Once a second vehicle has arrived, at the time the first vehicle is transferring its data to the PC-Box, this vehicle will need to wait to firsts data transference is completed.

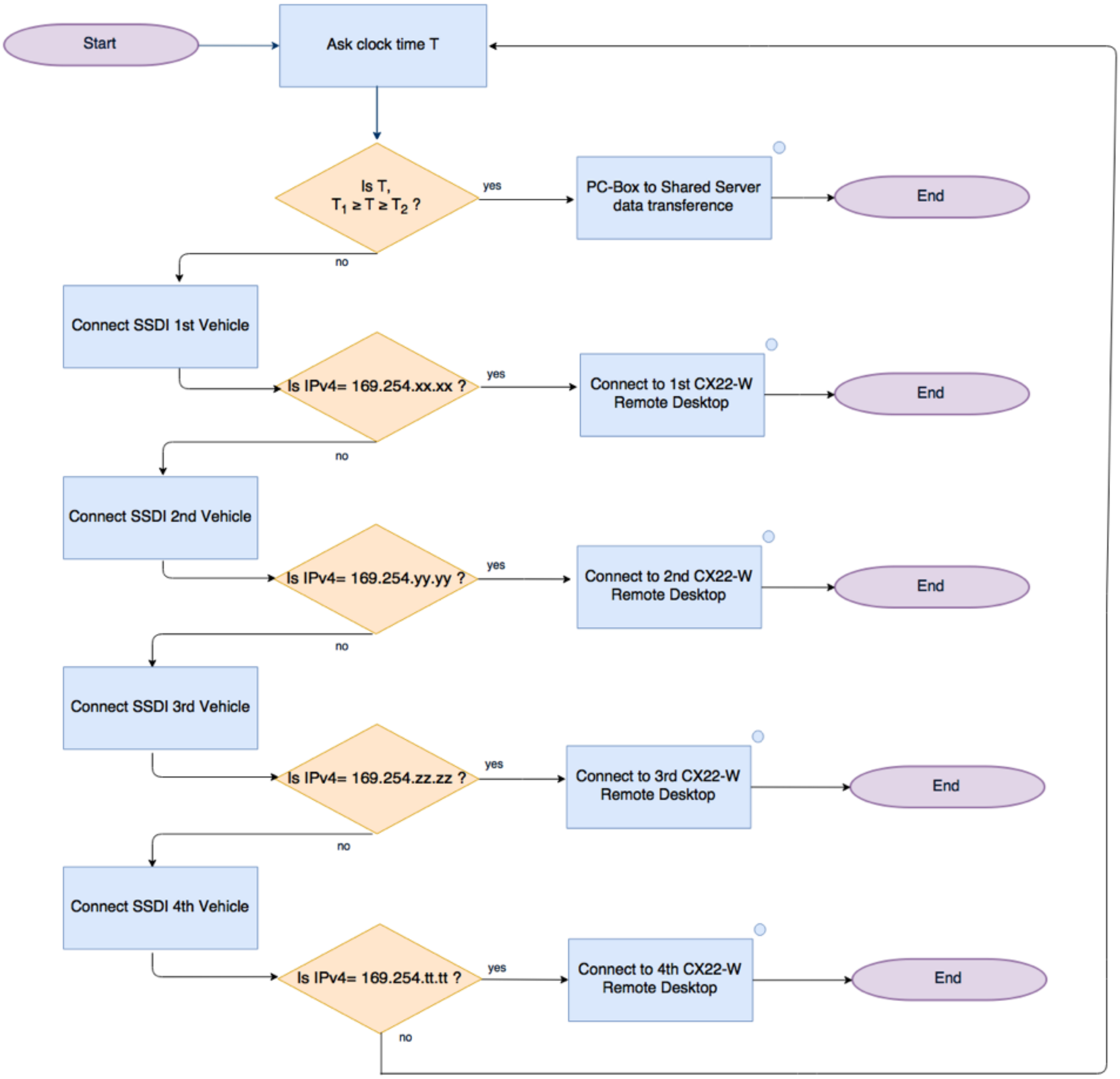
Then, the second vehicle will need to wait again until PC-Box connects to it. This time the second vehicle needs to wait again is the time the loop will be sequentially 'asking' every vehicle if they have arrived or not until is time to ask to this second vehicle. At this point the vehicle will answer 'yes' I am on the Box ready to transfer the data I recorded.

A possible situation could be if this second vehicle is waiting but one ore more vehicles arrive into the Box, so a third vehicle or more. Then, the priority of connection is not determinate by the arrival other but by the sequential order this connections are being checked by PC-Box.

The following flowchart presented and its code (available from Annex C13) rules the connections for four vehicles. The command connects to the different VANETS sequentially and then asks whether the PC-Box is connected to the IPv4 from the corresponding CX22-W or not. If the answer is no it asks the sequentially next vehicle. If the answer is yes it starts data transference as it was stated on the single vehicle case.

Flowchart 14 Multi VANET connectivity command.





On a single vehicle case the PC-Box's screen communicates with the pilot, who knows that if the screen is green it is possible to leave the Box. The same happens now with a multiple vehicles case. In this case the PC-Box will be connected to a panel that informs the pilot if data transference from each vehicle has been done or not yet. This is an example.

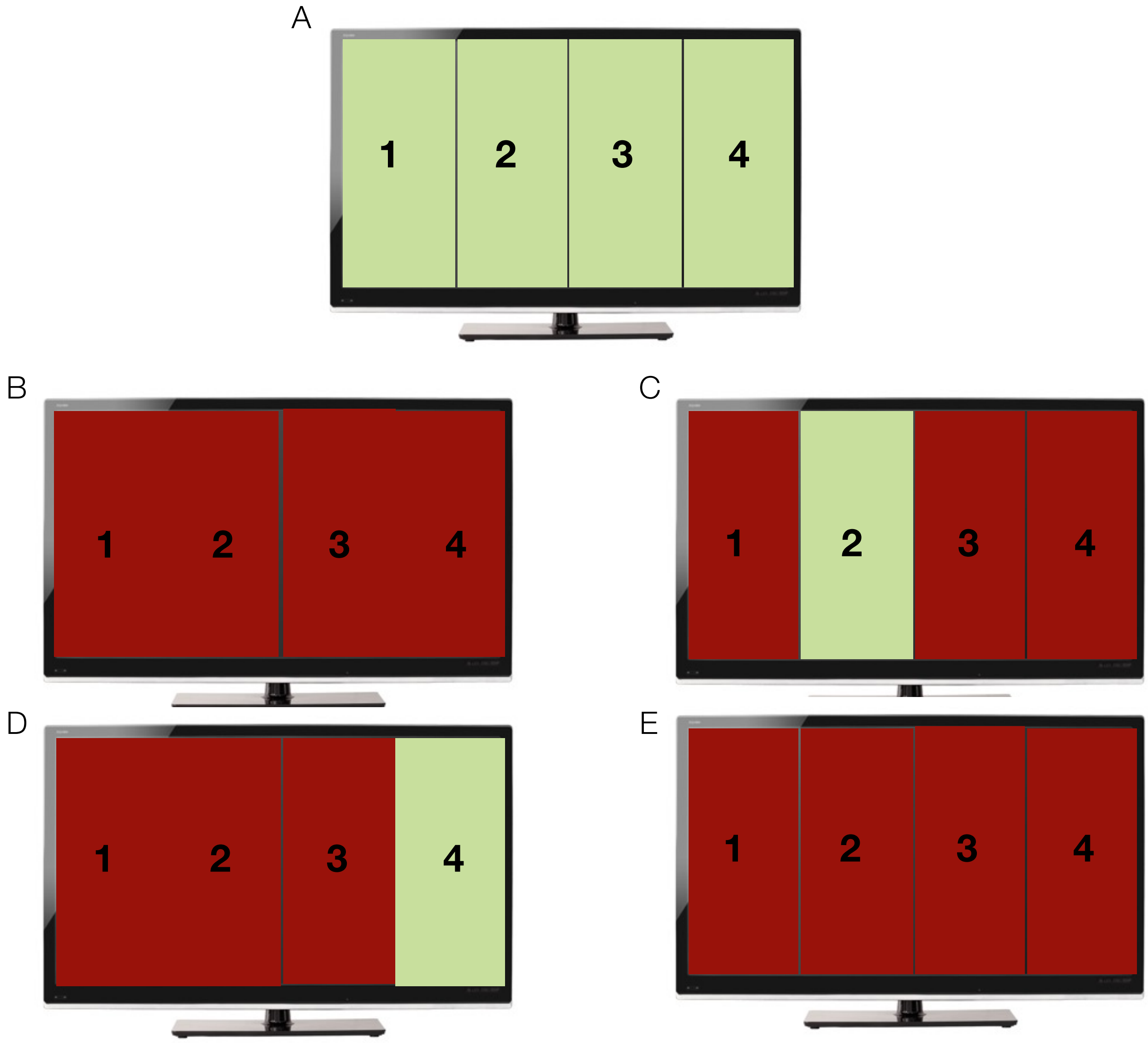


Fig. 45 Possible screens from the panel on the Box of 4 vehicles acquiring.

A. ADAQ system on the PC-Box has been activated, every vehicle can leave the Box.

B. After 7min the 4 stripes for the 4 vehicles become red as PC-Box is searching for their IPv4.

C. Vehicle number 2 arrives first into the Box, few minutes later arrives vehicle number 4. Vehicle 2 is the first on being attended and its data transference is completed so the second screen stripe becomes green.

D. After 7min the second screen stripe becomes red again as the PC-Box is looking for vehicle's number 2 IPv4 once again. As vehicle number 4 was waiting in the Box, PC-Box connects to it and once the data transference is done the fourth screen stripe becomes green so the pilot knows it is possible to leave the Box.

E. After 7min the fourth stripe becomes again red. No more vehicles are in the Box. PC-Box continues working on the loop looking for the next arrival from vehicles.

A and B screens are common for every start of ADAQ. Then, only one green stripe will be shown on the screen at the same time. There will be as many stripes as vehicles acquiring on the same testing track.

Conclusion

After the different tests done, ADAQ has proved to be reliable. Every stage of ADAQ works on the expected way and the company has allowed other departments doing some of their tests using ADAQ system, as an experimental phase. On this experimental phase, ADAQ will be tested on more extreme conditions and longer acquisitions in order to see the limits of the system and how it behaves.

ADAQ has also achieved its proposal of being versatile. Every from its stages, data acquisition, communication and analysis allows specific configurations depending on the goal of the test and can be re-configured while the test is on course. ADAQ has also the possibility to be used on its non-autonomous way, called DAQ system.

This system is ready for next improvements and ready to be tested on a multiple vehicle case acquiring on the same track area. If some of the next improvements are carried out, mainly the incorporation from the CX22-W into the company's network, ADAQ projects would be easier to configure and in a shorter time. This would reduce the possibility of human errors at configuring projects and reduce the possibility of errors while data is transferring, since data would go straight into the company's shared server.

The development from the zero version of ADAQ has conclude with satisfactory results.

Bibliography

1. *ADAQ on windmills.* Available from: http://docplayer.es/9808105-Adquisicion-de-datos-recurso-eolico.html

2. *CX22-W purchase documentation.* Also available from: https://www.hbm.com/en/2486/quantumx-cx22bw-data-recorder/

3. *MX840 purchase documentation.* Also available from: https://www.hbm.com/en/2129/quantumx-mx840b-8-channel-universal-amplifier/

4. *MX1615 purchase documentation.* Also available from: https://www.hbm.com/en/3053/quantumx-mx1615b-bridge-amplifier-for-strain-gauges/

5. *Catman Software.* Available from: https://www.hbm.com/es/2290/software-adquisicion-de-datos-catman/

6. *Navilock products.* Available from: http://www.navilock.de/produkte/F_781_USB_61422/merkmale.html?setLanguage=en

7. *Autoit Software.* Available from: https://www.autoitscript.com/site/autoit/

8. *LMS Tecware. Software.* Available from: https://www.plm.automation.siemens.com/es_es/products/lms/testing/tecware/

9. *LabVIEW Software.* Available from: http://www.ni.com/labview/esa/

10. NE*555 Timer.* Available from: http://www.ti.com/lit/ds/symlink/lm555.pdf

11. *Intermittent LED circuit.* Available from: https://www.youtube.com/watch?v=M_0aAoFdiiI

12. *WANET information.* Available from: https://en.wikipedia.org/wiki/Wireless_ad_hoc_network

13. *Vehicle to vehicle information.* Available from: https://en.wikipedia.org/wiki/Vehicle-to-vehicle

14. *Car to car information.* Available from: https://www.car-2-car.org/index.php?id=5

15. *IEEE 802.11n-2009 information.* Available from: https://en.wikipedia.org/wiki/IEEE_802.11n-2009

16. See 'Pasos para cambiar la prioridad de conexión de las redes inalámbricas en Windows 7'. Available from: https://support.microsoft.com/es-es/kb/2526067

17. *IPv4 address.* Available from: https://en.wikipedia.org/wiki/IPv4

18. *Ad hoc configuration*. Available from: Manual de Configuración, puntos 1.3/2.1

19. *Spectral density*. Available from: https://en.wikipedia.org/wiki/Spectral_density

20. *Rainflow-counting algorithm.* from: https://en.wikipedia.org/wiki/Rainflow-counting_algorithm

21. *VIs from LabVIEW.* Available from: https://en.wikipedia.org/wiki/LabVIEW