Software documentation for current-ripple equipment

Alexander Bessman, Rudi Soares

March 21, 2018

Contents

This document describes the software of the current-ripple set-up that was built and used by Alexander Bessman and Rudi Soares, in their respective PhD projects, as well as several related Master degree projects. This document is meant to be used as a complement to the hardware description of the current-ripple set-up, *An Experimental Setup with Alternating Current Capability for Evaluating Large Lithium-ion Battery Cells.*

1 LabVIEW interface

Having multiple indepedent channels was a core design specification, and the software controlling the set-up reflects this. The software was written in LabVIEW. Figure 1 shows the graphical user interface (GUI), also known as the Front Panel when using LabVIEW terminology. On the upper lefthand side of the GUI, settings for each of the sixteen channels can be individually configured, while on the righthand side a number of readouts from the tested cells are shown. See sections 1.1 and 1.2 for thorough descriptions of the available settings and the readouts, respectivelly.

Above the readouts are two buttons: “RUN” and “More info”. The “RUN” button predictably starts the set-up with the selected settings. The “More info” button opens another window with additional readouts, such as accumulator voltage, voltage and current readings from the DCGs and CRGs, timing values for operating modes that depend on them, and more.

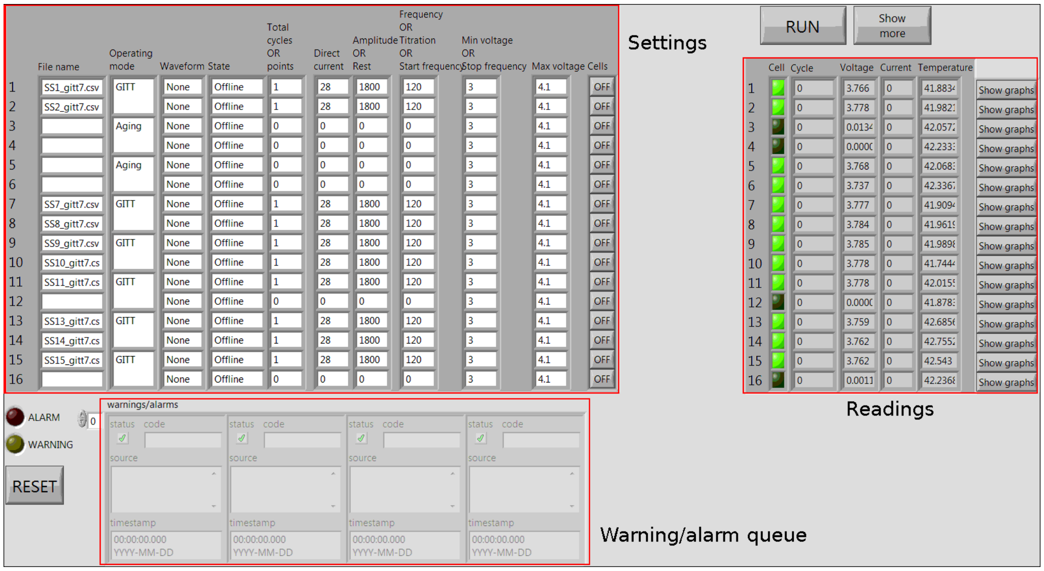
At the bottom is the warning queue, which shows incident reports from the safety subsystem. To the left of the warning queue are two indicator LEDs and a “Reset” button. The indicator LEDs will flash if there is a warning (yellow) or alarm (red) in the warning queue. The reset button clears the warning queue, and also triggers a current offset calibration. See the section 1.3 for a thorough description of the safety system.

1.1 Settings

The configurable settings are as follows:

Filename The name of the file to which data should be logged.

Operating mode The operating mode, or type of test to be performed. See below for a detailed description of the available operating modes.



	File name	Operating mode	Waveform	State	Total cycles OR points	Direct current	Amplitude OR Rest	Frequency OR Titration OR Start frequency	Min voltage OR Stop frequency	Max voltage	Cells
1	SS1_gitt7.csv	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
2	SS2_gitt7.csv		None	Offline	1	28	1800	120	3	4.1	OFF
3		Aging	None	Offline	0	0	0	0	3	4.1	OFF
4			None	Offline	0	0	0	0	3	4.1	OFF
5		Aging	None	Offline	0	0	0	0	3	4.1	OFF
6			None	Offline	0	0	0	0	3	4.1	OFF
7	SS7_gitt7.csv	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
8	SS8_gitt7.csv		None	Offline	1	28	1800	120	3	4.1	OFF
9	SS9_gitt7.csv	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
10	SS10_gitt7.cs		None	Offline	1	28	1800	120	3	4.1	OFF
11	SS11_gitt7.cs	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
12			None	Offline	0	0	0	0	3	4.1	OFF
13	SS13_gitt7.cs	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
14	SS14_gitt7.cs		None	Offline	1	28	1800	120	3	4.1	OFF
15	SS15_gitt7.cs	GITT	None	Offline	1	28	1800	120	3	4.1	OFF
16			None	Offline	0	0	0	0	3	4.1	OFF

	Cell	Cycle	Voltage	Current	Temperature	
1	●	0	3.766	0	41.8834	Show graphs
2	●	0	3.778	0	41.9821	Show graphs
3	○	0	0.0134	0	42.0572	Show graphs
4	○	0	0.0000	0	42.2333	Show graphs
5	●	0	3.768	0	42.0683	Show graphs
6	●	0	3.737	0	42.3367	Show graphs
7	●	0	3.777	0	41.9094	Show graphs
8	●	0	3.784	0	41.9619	Show graphs
9	●	0	3.785	0	41.9898	Show graphs
10	●	0	3.778	0	41.7444	Show graphs
11	●	0	3.778	0	42.0155	Show graphs
12	○	0	0.0000	0	41.8783	Show graphs
13	●	0	3.759	0	42.6856	Show graphs
14	●	0	3.762	0	42.7552	Show graphs
15	●	0	3.762	0	42.543	Show graphs
16	○	0	0.0011	0	42.2368	Show graphs

Figure 1: LabVIEW front panel from which the set-up is controlled.

Unlike the other settings, operating mode applies to two adjacent channels instead of just a single channel. This is because each channel is associated with two measurements (voltage and current), while the PXIe-6124 data aquisition card (DAQ) has four channels. Therefore, the multiplexer selects two channels simultaneously to increase efficiency. However, the sampling rate of the PXIe-6124 configured as a function of waveform frequency, and cannot be set on a per-channel basis. Therefore, a design choice was made to decrease the granularity of the operating mode setting in order to increase measurement efficiency.

Waveform Selects the AC waveform. The available waveforms are Sinus, Triangle, and Square, plus None which results in an AC amplitude of zero regardless of other settings.

State The current state of the cell being tested. The software is designed around the idea of state machines, where each cell/channel can exist in one of a finite number of states. The channel switches from one state to another when certain conditions are fulfilled. For example, if a channel's state machine is currently in **charge** state and the measured cell voltage exceeds a threshold, the state machine could switch to **discharge** state.

Total cycles OR Points In every operating mode except FRA, this setting determines how many times the cell should be cycled before being switched off. In FRA mode, it selects the number of data points to be collected.

Direct current Selects the DC level in Amperes.

Amplitude OR Rest In every operating mode except GITT, this setting determines the AC amplitude in Amperes. In GITT, it sets the rest period in seconds between titration periods.

Frequency OR Titration OR Start frequency In every operating mode except GITT and FRA, this setting determines the frequency in Hz. In GITT it sets the titration period in seconds. In FRA mode, it sets the frequency of the first data point in Hz.

Min voltage OR Stop frequency In every operating mode except FRA, this setting determines the minimum allowed voltage. If the measured voltage is at or below this value, the state is changed to **charge**. If the measured voltage is more than 0.4 V below this value for any reason, the state is changed to **offline** and a warning is sent to the operator(s) via email and text message. In FRA mode, it sets the frequency of the final data point in Hz.

Max voltage This setting determines the maximum allowed voltage. If the measured voltage is at of above this value, the state is changed to **discharge**. If the measured voltage is more than 0.2 V above this value for any reason, the state is changed to **offline** and a warning is sent to the operator(s) via email and text message.

Cells Finally, these buttons function like a hard off-switch for each channel. Regardless of any other setting, unless this button is set to ON, the channel remains offline and no current is drawn from or put into the cell. These buttons also affect the voltage safety check; if a certain channel's ON/OFF button is set to OFF the measured voltage can be above or below the safety limits without a warning being raised. This is because some channels may not have cells connected to them, resulting in a measured voltage close to zero or, in the case of stray voltages, much above or below the normally permitted level. However, the temperature safety check is NOT affected by the ON/OFF button;

a measured temperature above the safety limit will always result in total shut-down of the system (and possibly sounding of the evacuation alarm), regardless of whether the channel in question is active or not.

The different operating modes require further explanation. Each operating mode corresponds to a state machine. A state machine can be thought of as a machine which can exist in a finite number of well-defined states, and in fact always exists in exactly one of the states. The state machine changes states depending on any number of configurable criteria. Below is a list of the available operating modes, or state machines:

Aging The default operating mode, in which a cell is cycled between two voltage limits at a constant current. When the upper voltage limit is reached, the charging changes from galvanostatic (constant current) to potentiostatic (constant voltage). The constant voltage charging is achieved somewhat crudely by decreasing the current by 20% every time the upper voltage limit is exceeded. This continues until the current falls below 10% of its initial value, at which point the charging is considered complete and the state is changed to **discharge**. When the lower voltage limit is reached, the state changes back to **charge**. Unlike **charge**, there is no constant voltage period in the **discharge** state. The cycle count is incremented at the end of each charging period, and when the final cycle is complete the state machine enters **finalize** state, where the cell is discharged to its nominal voltage of 3.67 V and then turned **offline**. Figure 2 shows a state diagram of the aging state machine.

GITT GITT is an abbreviation of Galvanostatic Intermittent Titration Technique. This operating mode does not support any AC. Instead, the cell cycled using long pulses of DC, followed by even longer periods of zero current. In all other ways this operating mode functions in the same way as Aging.

Calendar In Calendar mode, a cell is held at a specific voltage indefinitely. If the measured voltage strays from the set voltage by more than 50 mV, the current set in the Direct current setting is applied to bring the cell back to the set voltage. AC can be applied in this mode. The total cycles setting is ignored, and the average of the Min and Max voltage settings is taken as the set voltage.

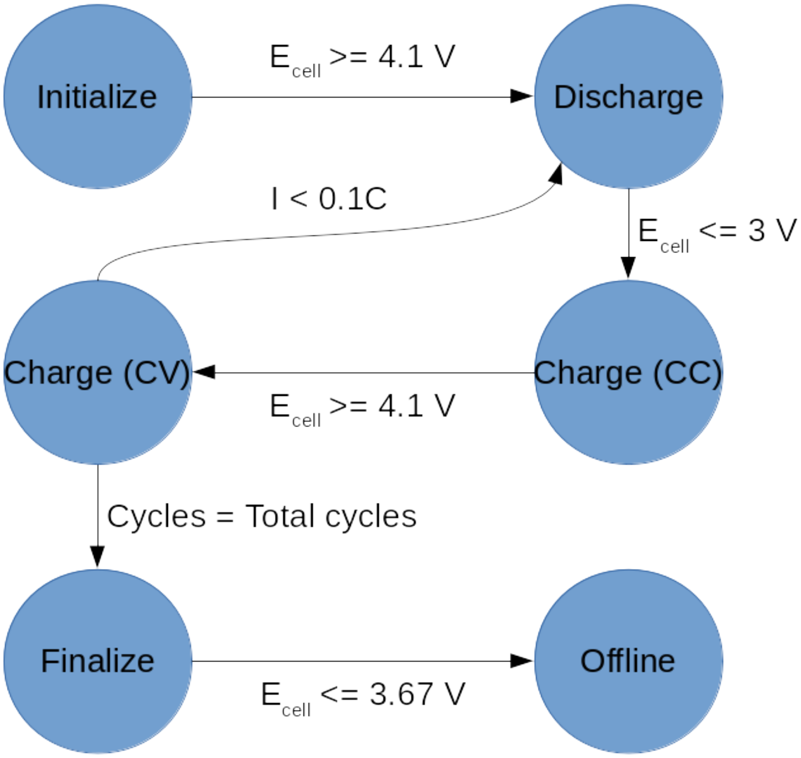


Figure 2: State diagram of the aging state machine.

HPPC HPPC is an abbreviation of Hybrid Pulse Power Characterization. In this operating mode, the current set in the Direct current setting is applied in two ten-second pulses, with a one minute rest before, between, and after the pulses. The first pulse uses positive current, while the second pulse uses negative current. Most settings are ignored; only Filename, State, Direct current, and Cells are used directly. Min and Max voltage are used for their usual safety purpose, but are otherwise ignored.

FRA FRA stands for Frequency Response Analyzer. It can be used to perform impedance spectroscopy measurements, in which a AC current with a certain amplitude and frequency is applied and the AC voltage response is measured. This is possible thanks to the PXIe-6124 card's high sampling rate.

In this mode, many settings have a different meaning than in the other modes. Total points sets the number of AC frequencies between Start frequency and Stop frequency to be measured at. Amplitude means just that. Start frequency and Stop frequency determine the first and last frequency at which to measure. It is possible to go from low to high or high to low. Max and Min voltages are ignored except for safety purposes.

Impedances are not calculated on-the-fly, but must instead the calculated after the fact from the current and voltage measurents. This mode has not been tested extensivelly and results should be verified using dedicated FRA equipment.

1.2 Readouts

In the main view, only currents, voltages, and temperatures are shown. The current and voltage readouts are averages of 400,000 samples from the PXIe-6124 measured over the course of one second. The temperature readouts are single-point measurements from the PXIe-4357 measurement card. However, the current voltage readings are only updated when the multiplexer is on those channels, which can happen as seldom as every eight seconds depending on how many cells are active. The temperature readings, on the other hand, are updated simultaneously for all cells every second.

The current readouts shown include offset compensation. Even when there is current applied to a certain channel, the PXIe-6124 will read a

Table 1: Automatic safety actions.

	Cell shutdown	Total shutdown	Operators alerted	Power cut to CRGs	Evacuation alarm
Voltage above 4.2 V or below 2.6 V	Yes	No	Yes	No	No
Temperature above 50 °C	Yes	Yes	Yes	No	No
Temperature above 55 °C	Yes	Yes	Yes	Yes	Yes

nonzero current. This offset current is sometimes quite large and varies slowly over the course of the day. Therefore, every hour the measurements are stopped and no current applied to any of the channels. During this time, the current offsets are updated by measuring the apparent current on each channel and thereafter automatically applied to the measured currents when the experiment restarts.

The current readout for inactive cells is shown as 0 in order to reduce clutter, even though in actuality there is always some nonzero current measured even when a channel is offline.

In additions to the three default readouts, more information can be accessed by clicking the "More info" button and the individual graph buttons. The latter will each open an additional window showing a waveform graph for the cell in question.

1.3 Safety

Since the tested cells could potentially create quite a hazard should they fail in an unsafe manner, much care has been taken when designing the automatic safety protocols. The primary safety metric is the cell temperature, since catastrophic cell failure is preceded by thermal runaway. Cell voltage is a secondary safety metric, since catastrophic failure is unlikely as long as the cell voltage is within safe (manufacturer specified) limits.

Table 1.3 outlines what automatic actions are performed if unsafe conditions arise.

Each incident is reported in the Warning queue and raise a warning. The temperature exceeding 55 °C additionally raises an alarm. An alarm being raised prevents additional warnings from being sent to operators via text message, but all warnings are still added to the Warning queue and sent via email.

2 CAN

Both the DCGs and CRGs are controlled via CAN, Controller Area Network.

2.1 Protocol overview

CAN is a message-based protocol, where each device or *node* on the network is continuously transmitting and receiving messages or *frames* to and from all other nodes. During a communication cycle, all nodes send their frames at the same time and simultaneously read all other frames during an arbitration period. The arbitration determines which frame has the highest priority, based in the IDs of the transmitted frames. For this reason all frames being transmitted on the network must have unique IDs. Once priority has been established all nodes except the one with priority stop transmitting. After the prioritized node has finished transmitting its frame the cycle repeats, and the node that won arbitration last cycle does not transmit again until a timeout has been reached. This allows the frames with lower priority to be transmitted eventually.

Table 2.1 and figure 3 show the different parts that make up a CAN frame. In this case the frame is of *extended* type, meaning that it has a 29 bit identifier. Non-extended frames have only 11 identification bits, but we are only concerned with extended frames since all nodes in this project use them.

We are primarily concerned with the colored parts of table 2.1 and figure 3. The green part is the identifier, which also determines priority. Each node, i.e. every DCG and every CRG, must have its own unique identifier to avoid collisions. A smaller ID gives a higher priority. The yellow part is the data length code, which is necessary because the size of the data field (red) is variable.

Table 2: Components of an extended CAN frame. Colors correspond to figure 3. From Wikipedia.

Field name	**Length (bits)**	**Purpose**
Start-of-frame	1	Denotes the start of frame transmission
Identifier A (green)	11	First part of the (unique) identifier which also represents the message priority
Substitute remote request (SRR)	1	Must be recessive (1)
Identifier extension bit (IDE)	1	Must be recessive (1) for extended frame format with 29-bit identifiers
Identifier B (green)	18	Second part of the (unique) identifier which also represents the message priority
Remote transmission request (RTR)	1	Must be dominant (0) for data frames and recessive (1) for remote request frames
Reserved bits (r1, r0)	2	Reserved bits which must be set dominant (0), but accepted as either dominant or recessive
Data length code (DLC) (yellow)	4	Number of bytes of data (0 - 8 bytes)
Data field (red)	0 - 64	Data to be transmitted (length dictated by DLC field)
CRC	15	Cyclic redundancy check
CRC delimiter	1	Must be recessive (1)
ACK slot	1	Transmitter sends recessive (1) and any receiver can assert a dominant (0)
ACK delimiter	1	Must be recessive (1)
End-of-frame (EOF)	7	Must be recessive (1)

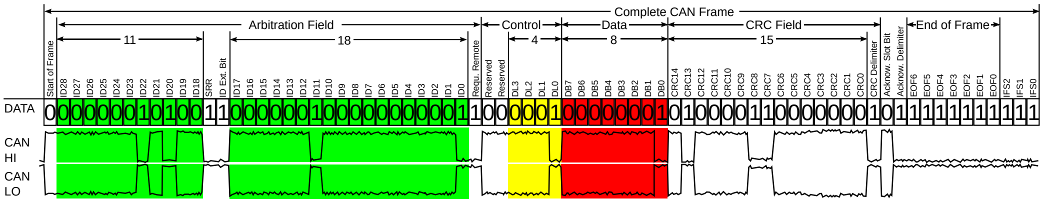
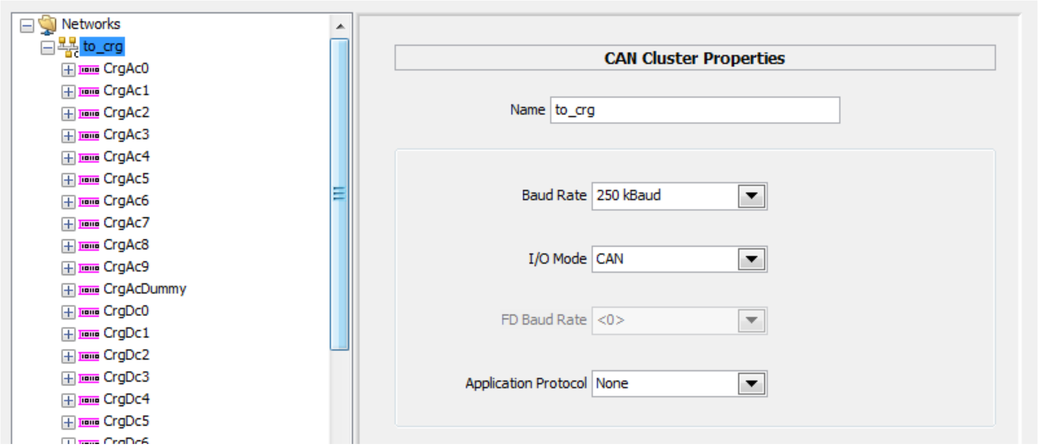


Figure 3: Components of an extended CAN frame. Modified from Wikimedia Commons.



CAN Cluster Properties

Name to_crg

Baud Rate 250 kBaud

I/O Mode CAN

FD Baud Rate <0>

Application Protocol None

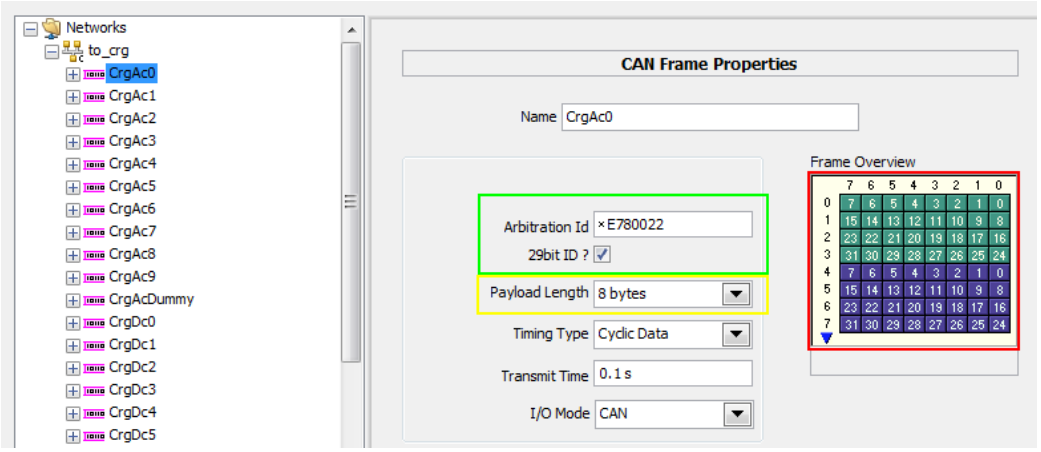
Figure 4: View of a cluster in NI-XNET.

2.2 XNET

XNET is a National Instruments middleware for handling CAN data. It divides CAN data into three layers. The top layer is called a *cluster*, and is where the baud rate of the network is configured. An XNET database can contain multiple clusters. In our case, clusters are separated by the purpose of the frames within; the CRGs have two clusters, one of which contains frames that the CRG will be transmitting, and another which contains frames the CRG will be listening for. Each cluster has a name, which is used to identify it inside LabVIEW. Figure 4 shows a cluster in NI-XNET.

A cluster contains *frames*, the anatomy of which is explained in the previous section. Figure 5 shows the layout of a frame in NI-XNET. Like clusters, each frame has a name which is used to identify it within LabVIEW. The frame's name is different from its *arbitration ID*, shown in green outline in figure 5; as mentioned above, the arbitration ID identifies the frame on the network, while the name identifies it within LabVIEW. The *payload length* highlighted in yellow is XNET's name for the data length code mentioned in the previous section. It determines the size of the data field or *frame overview*, highlighted in red. The frame in figure 5 uses the maximum allowed data field size of 64 bits.

The frames in turn contain *signals*, which specify the frames' data (the red section in table 2.1 and figure 3). The colored fields in the frame overview correspond to different signals, one of which is shown in figure 6. Like clusters



CAN Frame Properties

Name: CrgAc0

Arbitration Id: xE780022

29bit ID ? ☑

Payload Length: 8 bytes

Timing Type: Cyclic Data

Transmit Time: 0.1 s

I/O Mode: CAN

Frame Overview

	7	6	5	4	3	2	1	0
0	7	6	5	4	3	2	1	0
1	15	14	13	12	11	10	9	8
2	23	22	21	20	19	18	17	16
3	31	30	29	28	27	26	25	24
4	7	6	5	4	3	2	1	0
5	15	14	13	12	11	10	9	8
6	23	22	21	20	19	18	17	16
7	31	30	29	28	27	26	25	24

Figure 5: View of a frame in NI-XNET.

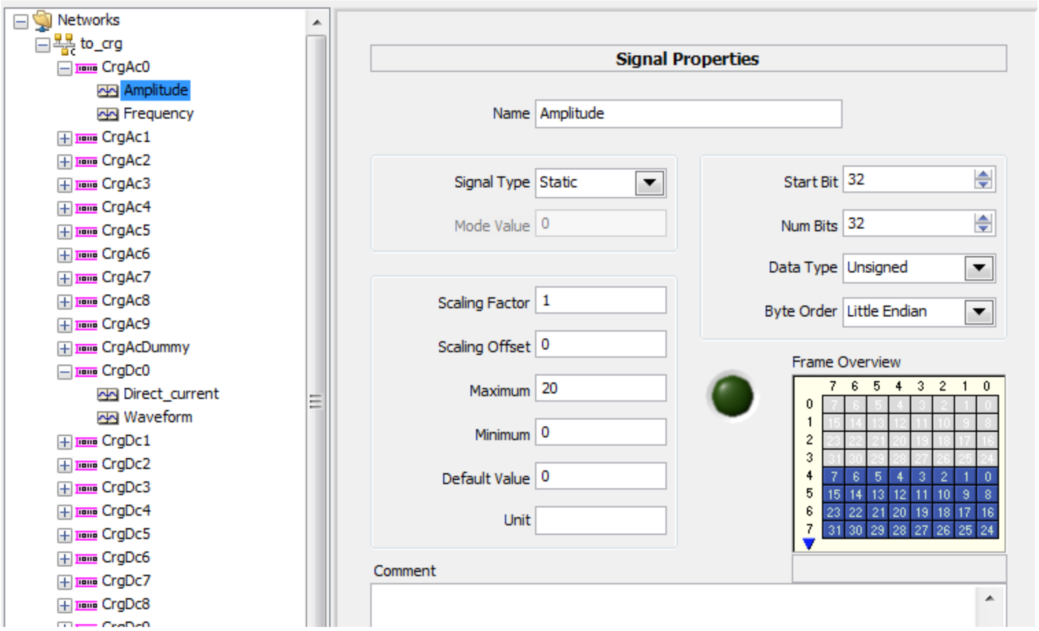
and frames, signals have names to identify them in LabVIEW. In addition, they have a number of settings which determine how data passed to the network via the signal is handled. In this project, all signals are of *static* type, meaning that every signal is transmitted with each instance of a certain frame. NI-XNET also supports *dynamic* signals, which may or may not be transmitted at the discretion of the developer, but these are not used in this project.

The purpose of *start bit* and *num bits* are obvious enough; they specify where in the data field the signal starts and how much of the data field it takes up. *Data type* can be either signed, unsigned, or IEEE float. Signed and unsigned signals can only pass integer values, with unsigned only able to pass positive integers. Float can pass non-integer values with up to 15 significant digits. *Byte order* sets the endianness of the the signal, i.e. whether the most significant byte comes at the start (big-endian) or end (little-endian). The signal in figure 6 is little-endian (as are all signals to and from the CRGs), meaning that the most significant byte is transmitted last.

The remaining settings, from *scaling factor* to *units* are ignored and have no effect.

2.3 Communicating with the CRGs

Table 2.3 shows all clusters, frames, and signals recognized and transmitted by the CRGs. The CRGs do not understand nor send true values for voltage,



- Networks
  - to_crg
    - CrgAc0
      - Amplitude
      - Frequency
    - CrgAc1
    - CrgAc2
    - CrgAc3
    - CrgAc4
    - CrgAc5
    - CrgAc6
    - CrgAc7
    - CrgAc8
    - CrgAc9
    - CrgAcDummy
    - CrgDc0
      - Direct_current
      - Waveform
    - CrgDc1
    - CrgDc2
    - CrgDc3
    - CrgDc4
    - CrgDc5
    - CrgDc6
    - CrgDc7
    - CrgDc8
    - CrgDc9

Signal Properties

Name: Amplitude

Signal Type: Static

Mode Value: 0

Scaling Factor: 1

Scaling Offset: 0

Maximum: 20

Minimum: 0

Default Value: 0

Unit:

Start Bit: 32

Num Bits: 32

Data Type: Unsigned

Byte Order: Little Endian

Frame Overview

	7	6	5	4	3	2	1	0
0	7	6	5	4	3	2	1	0
1	15	14	13	12	11	10	9	8
2	23	22	21	20	19	18	17	16
3	31	30	29	28	27	26	25	24
4	7	6	5	4	3	2	1	0
5	15	14	13	12	11	10	9	8
6	23	22	21	20	19	18	17	16
7	31	30	29	28	27	26	25	24

Comment

Figure 6: View of a signal in NI-XNET.

current, or frequency. Rather, all data transmitted to or received from the CRGs must be translated using the formula in the *Translation* column of table 2.3. For example, to tell a CRG to produce DC with a magnitude of 20 A, one must send a data package with the number $\frac{4294967295(20+60)}{120} = 2863311530$ to the CRG. If the CRG is measuring an input voltage of 24 V it will send 7864.2, since $\frac{200 \cdot 7864.2}{65535} = 24$.

In these calculations, 4294967295 is the largest number than can be represented by a 32-bit unsigned integer. The translation function normalizes the requested current against a 32-bit unsigned integer, between the limits -60 and 60. Similarly, 65535 is the largest number that can be held in a 16-bit unsigned integer, and the value sent by the CRG is a normalization of the input voltage against a 16-bit unsigned integer between the limits 0 and 200. Other “magic numbers” in the translations come from similar hardware-related causes.

In each case the data is identified with the arbitration ID from the *Frame ID* field, where a is the zero-indexed CRG number counting from the upper left of the mounting rack. For example, the third CRG from the left on the upper row, i.e. CRG no. 2 will recognize a CrgAc frame with the hexadecimal arbitration ID xE78004 as being intended for itself. Similarly, it will transmit its CrgCurrVolt frames with the identifier x15AC0004.

One caveat of the above system is that CRG no. 7 is currently (as of March 2018) broken, and CRG no. 9 is in its place, i.e. a CrgAc frame with ID xE7800B must be used to address the third CRG from the right on the lower row, rather than xE78009 as might be expected. Other frames follow the same pattern, of course. One of the phases on CRG no. 7 is broken, but could probably be repaired without too much effort.

Arbitration IDs ending in 1 was used to address a “dummy CRG” which was used in testing but is currently not present in the system. The dummy CRG has two phases instead of three, and could possibly be rebuilt into a “full CRG” if necessary.

The signal *pwm_frequency* requires a little bit of explanation. Currently, it should always be set to 70 kHz, which is the highest frequency the CRG can handle. Since this value should not be changed, it is curious to have a CAN signal for it rather than just hardcoding it in firmware. However, it could also be used to drive a square waveform directly, instead of being used to modulate a waveform of arbitrary shape as is currently the case. This would allow the system to reach frequencies up to 70 kHz, instead of being limited to one or two kHz as is the case today. This functionality is not implemented, but explains why a CAN signal exists for this setting.

The CRG-side code for handling the CAN communication can be found in Appendix C of the related Master’s thesis [1].

2.4 Communicating with the DCGs

See documentation from Elektronikkonsult.

References

[1] S. Vadivelu, “Investigation of sinusoidal ripple current charging techniques for Li-ion cells,” KTH, Tech. Rep., 2016. [Online]. Available: http://kth.diva-portal.org/smash/get/diva2:1044206/FULLTEXT01.pdf

Table 3: CRG communication.

Cluster Frame *Frame ID*	Signal	Meaning	Valid values	Translation	Position in data field
to_crg					
CrgAc *xE780022 + a*	Frequency	Frequency of the AC	0 - 2000	$\frac{4294967295y}{1000}$	0 - 31
	Amplitude	Peak amplitude of the AC	0 - 60	$\frac{4294967295y}{60}$	32 - 63
CrgDc *xE780002 + a*	Direct_current	Magnitude of the DC	-60 - 60	$\frac{4294967295(y+60)}{120}$	0 - 31
	Waveform	Shape of the AC waveform	0 - 2	y	32-47
CrgKpKi *xF780002 + a*	k_p	Proportional gain factor for PID controller	0+	$\frac{4294967295y}{10000}$	0 - 31
	k_i	Integral gain factor for PID controller	0+	$\frac{4294967295y}{10000}$	32 - 63
CrgSettings *xE780012 + a*	pwm_frequency	Switching frequency of the PWM controller	70000	$\frac{4294967295}{39}(\frac{80000}{y} - 1)$	0 - 31
	sys_on	Turns PWM controller on and off	0 - 1	y	32 - 47
to_pc					
CrgCurrVolt *x15AC0002 + a*	Current	Total magnitude of the current being delivered by the CRG	N/A	$\frac{200y}{65535} - 100$	0 - 31
	Voltage	Input voltage to the CRG	N/A	$\frac{200y}{65535}$	32 - 63
CrgError *x15AC0012 + a*	error_flag	Error status of CRG	0 (no error) 1 (current error) 2 (temp. error) 4 (CAN error)	y	0 - 7