energies

MDPI

Brief Report

Low Frequency Injection as a Method of Low-Level DC Microgrid Communication

Matthew Davidson [1] and Andrea Benigni [2,3,*]

[1] Electrical Engineering, College of Engineering and Computing, University of South Carolina, Columbia, SC 29208, USA; davids24@email.sc.edu

[2] Mechanical Engineering, RWTH Aachen University, 52062 Aachen, Germany

[3] Institute for Energy Systems Engineering, Forschungszentrum Juelich GmbH, 52062 Aachen, Germany

* Correspondence: a.benigni@fz-juelich.de; Tel.: +49-2461-61-85523

Received: 27 February 2020; Accepted: 6 May 2020; Published: 13 May 2020

Abstract: This paper provides a simple low-level unidirectional global communication method for DC microgrids, and requires no hardware modifications to the microgrid and interfacing power electronic converters. The underlying premise to this communication method is injecting low-frequency low-voltage sinusoidal components into the DC microgrid power lines. This method deviates from the common bit-level communication scheme by relating parameters and commands with certain frequency components. Communication structures are included as a basis for communication protocols, and a detection method is proposed for detecting the injected frequencies. The injection method, communication structure, and detection method are implemented on a live-scale DC microgrid.

Keywords: communication; dc microgrid; phase locked loop; PLL; power system control; power system optimization; second order generalized integrator; SOGI

1. Introduction

As optimization schemes for DC microgrids improve, the ability to adjust certain parameters of connected devices (CDs) becomes critical. This is especially true if any CDs have storage ability and/or renewables, as each CD would have the ability to consume and/or produce energy. Therefore, in order to change these parameters, a simple unidirectional communication method was created: Low Frequency Components Communication (LFCC).

The novelty of this communication style allows for a simple implementation, in the sense that it requires no physical changes to the CDs or the microgrid structure. Following very few constraints and assumptions, the only change needed is to add lightweight injection (encoding) and detection (decoding) software, which consists of the appropriate method for decoding.

LFCC involves injecting one or several small amplitude sinusoidal tones on the DC power lines. The superimposed sine waves encode information in their spectra, while the amplitude is chosen to ensure that all CDs can properly detect the injected tone. The frequency range is chosen such that any connected converter can easily reject the components as disturbances on the power lines; this communication style is expected to work in the extremely and super low frequency bands and below. The advantage of using low-frequency sinusoidal injection is simplicity in transmitting and receiving, the fact that it can work alongside other communication techniques, that it has virtually no electromagnetic compatibility issues, an extremely low harmonic injection related to the superimposed signal, the ability to work at long distances, and the support of application specific protocols. LFCC is envisioned mainly as unidirectional and to provide global commands across the entire microgrid, however, bidirectional communication and addressability are evaluated in this paper.

This method is certainly not the first to utilize the power lines as a form of infrastructure for communication, as commonly implemented technologies currently include: power line communication (PLC), the utilization of droop style control as a form of communication such as with DC Bus Signaling (DBS), and Power Line Signaling (PLS) used as a way of triggering an operating mode [1]. PLC is a verified technique amongst AC grids; however, this requires additional hardware to achieve, due to its higher data rates (300 Hz to 250 MHz) [2]. The utilization of this frequency band also introduces challenges related to electromagnetic compliance and high probability of signal corruption, due to noisy environments [2]. Droop is typically used as a control by varying the output voltage in response to the load current [3] but this method of altering the DC voltage can certainly be used as a communication method [4] and extended to another parameter. Typically, though, droop is considered only with respect to load conditions, and is not extended to multiple various user-chosen low-level parameters.

Within the concept of utilizing existing hardware to perform communication, there has been some work towards utilizing the switching frequency, since it already exists from the power conversion process. In [1], the switching frequency is varied in response to load or power conditions, which is not too different from droop, and therefore will have the same limitation, as mentioned in the previous paragraph. In order to allow a bit-based communication, the switching frequency can utilize frequency-shift keying (FSK) and phase-shift keying (PSK), as seen in [5,6]. These methods require taking a carrier, in this case the switching frequency, and discretely switching between two different frequencies or phases (in the simplest case), which will represent the data '0' and '1'. This method is dependent on converter topology, conduction mode (continuous or discontinuous), and whether the topology is responsible for power delivery or consumption [5]. The detection methods may not work if a load disconnects from the grid, which could be the case for self-sustaining loads (i.e., with storage elements). Furthermore, altering the switching frequency may not be a suitable method for all converters, due to potential operational issues and any parametric algorithms dependent on the control frequency. Both [1] and [5] utilize Fourier transform based methods, in order to detect the appropriate frequencies, which can be both computationally and memory intensive. However, it must be said that specialized Fourier techniques (such as zoom or sliding), paired with an appropriately coded digital signal processor can be quite efficient. Lastly, in [7], frequencies from 16–26 Hz are injected onto the grid from a voltage source converter in the form of "energy packets" (in regard to energy being a major control parameter), and each frequency represents a 0 bit or 1 bit for a specific converter. This is probably the most similar to LFCC, however, the utilization of the injected frequencies is quite limited, and the use of binary representation could mean a considerable amount of time would have to pass before the communication is decoded.

LFCC aims to allows an alternative communication method on DC power lines without the complications of more advanced methods such as PLC. A succinct advantage of this method is that it can also be implemented alongside the previously mentioned methods for communication, since the method in which communication occurs is decoupled or independent from the methods the others use. The detection of the injected frequencies is performed by a second order generalized integrator (SOGI) phase locked loop (PLL) (SOGI-PLL), in conjunction with infinite impulse response (IIR) bandpass filters. A multiplexed detection scheme is also discussed to ensure a low computational burden on the interfacing converters' controllers. The following sections will provide a common and basic structure for designing the application specific protocol for LFCC.

2. Communication Approach

This section describes the general specifications and metrics for LFCC, how a command will be represented by its parameters, and how to map the frequency space to the command(s). Then, the communication method for sending commands, entitled Layered-Parameter Command (LPC), is presented as a basic structure which can be modified or utilized to create a new communication method. This method, along with all terminology and concepts, was created specifically for the type of communication that is LFCC, and as stated before, this allows LFCC to have an application specific protocol, which is necessary due the high dependence of system variables. Lastly, some more advanced features of LFCC will be discussed relating to channel separation and feedback.

Since the commands are sent via frequency injection on the DC power lines, the DC microgrid voltage, at all times, can be represented by:

$$V_{GRID}(t) = V_{DC} + V_{CMD}(t) \tag{1}$$

where V_{DC} is the nominal DC voltage of the grid and $V_{CMD}(t)$ represents all the injected frequencies that could exist for a given protocol. If the number of those injected frequencies are given by n, then $V_{CMD}(t)$ expands to:

$$V_{CMD}(t) = \sum_{i=1}^{n} V_{p_i}(t) = \sum_{i=1}^{n} A_i \sin(2\pi f_i t) \tag{2}$$

The frequency values f_i and corresponding amplitudes A_i are described in the following sections and the voltage V_{p_i} refers to the parameter that makes up the command.

Before detailing the generalized specifications, a few assumptions are assumed about the DC microgrid system. These assumptions mainly depend on the controller for the interfacing converters (the converter connected to the DC power lines):

1. The control is performed digitally with a microcontroller or digital signal processor (DSP);
2. The control algorithms can be updated in deployment, and there exists read-only memory (ROM) space and computational and memory overhead;
3. There exists a voltage transducer/sensor at the DC grid power lines.

Assumption 2 is only intended for already deployed applications, and Assumption 3 is reasonable, due to typical signals needed for either control feedback or fault purposes. These assumptions are made, so that this communication method requires no additional hardware.

2.1. Specifications and Metrics

The specifications of LFCC are explained here, in order for the user to properly design LFCC for their system. Each system will be different, and therefore the capabilities of LFCC will depend on the DC microgrid configuration, as well as the interfacing converters. Therefore, the specifications and metrics are usually defined in the fuzzy sense (e.g., small, large, low, high, etc.), as they are highly system dependent.

The first set of specifications deal with the signal attributes of the injected frequency: the frequency range or bandwidth of LFCC and the injection amplitude. The smallest frequency will mainly depend on the detection algorithm, but it is reasonable to assume to be within the range of 0.5–10 Hz. The highest frequency will largely be determined by the input bandwidth of the converter. The interfacing converters need to reject these sinusoidal disturbances, in order to not compromise converter regulation. It is also assumed the sampling rate of the control algorithm is significantly faster than the largest frequency of LFCC, therefore the detection algorithm offers no substantial constraint. With respect to amplitude, information is only encoded in the spectra, however, the amplitude must be chosen in order satisfy the signal-to-noise (SNR) requirements in terms of detectability for all CDs. This should be in consideration to possible distortion from noise (coupled or conducted from other devices), sensor/transducer non-linearities and sensitivity, and digital specifications, mainly being the analog to digital converter (ADC) effective number of bits (ENOB).

The second set of specifications deal with how LFCC is used: the criticality of the parameters to be changed and latency. The criticality is defined by the largest consequence from either loss of communication or erroneous detection. This means that should one of those two events occur, it is desirable for the quality, stability, and safety of the grid to not be compromised. It is the authors' opinion that, because this communication scheme is unencrypted and doesn't typically have feedback (or message verification), although this is discussed in Section 2.4, the commands used in LFCC should be of low criticality, and thus better suited for optimization purposes. Since the criticality is made low, this means the requirements for latency are also made low, as the dynamics of low criticality optimization parameters are most likely expected to occur over minutes to hours. Latency is defined by the maximum length of time required for a CD to decode a command sent via LFCC, which is used to calculate throughput (as commands per second (cps)). Because this is not a bit-level scheme, and more an instruction or command scheme, latency is mainly determined by how quickly the detection method detects all injected frequencies, plus an additional buffer time or headroom. Latency can typically be expected to exist between 50–500 ms, or 2–20 cps, which would be the time interval the next command can be sent. This number is derived from the time it takes for the chosen decode method. Since this is dependent on processor speed and algorithm choice, an estimation is derived from experimental results and from [5–7]. Slower latencies may also aid in reducing computational burden, by allowing the CDs to periodically sample the grid instead of a constant monitoring scheme.

In order to evaluate the communication method, two criteria have been developed to outline the method's performance in terms of: parameterization depth and frequency space utilization. Parameterization depth provides an insight to the quantity of parameters a single command can reasonably take, as well as the number of sub-parameters a parameter can contain, and frequency space utilization provides an insight to how much of the frequency space a method will consume. Additionally, latency is also included as a metric, which has already been defined.

2.2. Command Mapping and Definitions

Here, several terms common to LFCC are further explained for better understanding. Also described is the parameterization of commands. The secondary purpose of this section is to assist in the decoding of injected frequencies.

The frequency or spectrum allocation towards LFCC will be made up of 'spaces' and 'bands'. The entire spectrum is referred to as the frequency space: the frequencies from the smallest to the largest frequency that LFCC uses. If referencing a particular portion of the frequency space, that is referred to as a frequency band. There will also be frequency bands in which it is desirable to have no injection or detection take place, aptly named the null-frequency band. This exists as a construct to remove frequency bands from the frequency space that would cause issues, such as known system frequencies, which could cause false detection, or as a way of providing a buffer or isolation between two frequency bands.

A command is comprised of one or several parameters and only serves as a structural "parent" to organize the associated parameters. This helps to partition the available spectrum in LFCC. As seen in Figure 1, the command/parameter hierarchy has a type of tree structure in the sense that there is a downward traversal, where each parameter belongs to its own frequency band defined by the min and max frequency of that band. It may also be noticed that the parameters in each branch can be of two types: a selection parameter or a value parameter. The selection parameter presents itself as a way of selecting options or sub-parameters, which may also allow a further traversal to another parameter. In the case where an option has an associated value that needs to be chosen, the value parameter is used. The value need not be numeric, but is rather based on what the option of the selection parameter requires for its specific application.

For the selection parameter, the frequency band is evenly partitioned, based on the possible number of options. These partitions collectively make the parameter space, and may be referred to as sub-parameters. All parameter spaces that make up a command will be collectively known as the command space. A single parameter command's command space will be the same as the parameter space. The value parameter may also be partitioned, or numeric values may be linearly mapped within the frequency band. The value parameter is typically the final parameter of a branch. The purpose of these two parameters is to discern the decoding process, and reveal the intention of the command more clearly.

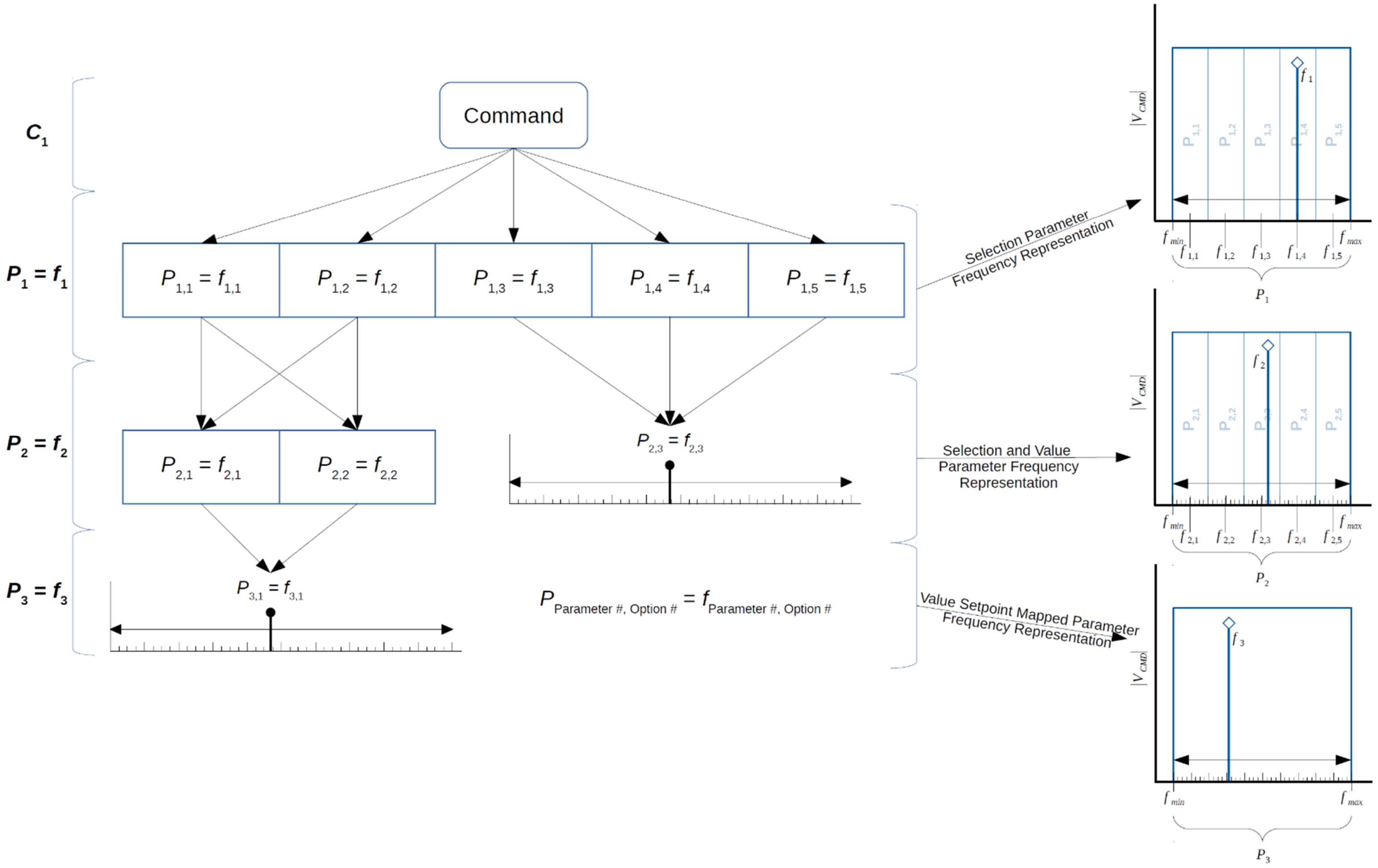


Figure 1. Command and parameter structure (command tree) (**left**), demonstrating selection and value parameters with their associated frequency representation (**right**).

For frequency bands that are evenly partitioned, each partition is described by its center frequency, whose width is dependent on the parameter space width and number of partitions, which is described by Equation (3) and seen in Figure 2.

$$\begin{cases} C_{sp1}, P_{sp1}, f_1 \in \ [f_{min}, f_{max}], \ \ P_1 \in \left[1, n_{p_1}\right] \\ P_{1,n} \in [f_{min} + \rho(n-1), \ f_{min} + \rho n] \\ P_{1,n} := \ f_{1,n} = f_{min} + \frac{\rho}{2}(2n-1) \\ s.t. \ n = 1, 2, \ldots n_{p_1}, \quad \rho = \frac{f_{max} - f_{min}}{n_{p_1}} \end{cases} \tag{3}$$

where $P_{1,n}$ indicates each partition and thus is P_1's sub-parameters contained in the parameter space P_{sp1}, which, in this case, is also the command space C_{sp1}, since there exists only one parameter for this command. Variable f_1 is the injected frequency, $f_{1,n}$ is each partition's center frequency (which defines $P_{1,n}$), f_{min} and f_{max} are the minimum and maximum frequencies parameter P_1 exists in, and n_{p_1} is the number of partitions within P_1. The width of each partition should be determined with consideration to the accuracy of the detection algorithm, sensor ability, and max deviation of signal (i.e., signal with noise). Therefore, the width of each partition will be dependent on the frequency range of P_{sp1} and the amount of required partitions. The value of P_1 is chosen by examining the partition that frequency f_1 falls in. This partition can be represented by its corresponding center frequency $f_{1,n}$, where the second line in Equation (3) defines that frequency's allowed deviation, while still being able to be correctly detected. Therefore, the partition can be found by comparing the nearness of f_1 to each $f_{1,n}$, so whichever element satisfies $\min\left\{\left|f_1 - f_{1,n}\right|\right\}$, is the appropriate partition and thus parameter for P_1. Conversely, simply utilizing if-else statements to check which frequency partition the detected frequency falls within, utilizing the second line of Equation (3), is acceptable as well.

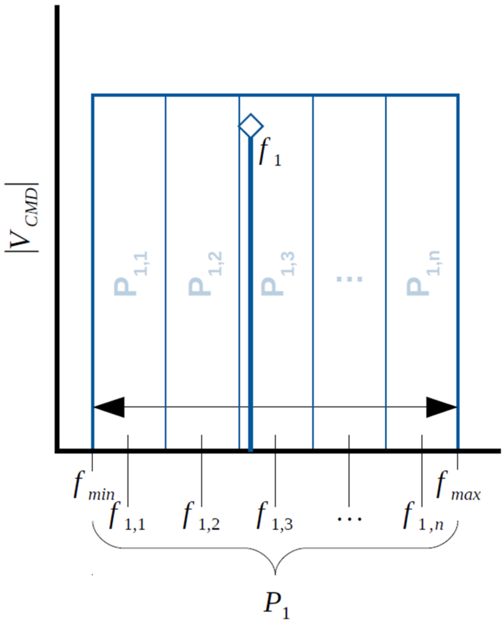


Figure 2. Exemplifying how a frequency band is evenly partitioned and defined by each sub-parameter's center frequency.

When the frequency value is linearly mapped to a frequency band, then it could prove useful to have that represent a certain setpoint or rate, the two variations typical of a value parameter. The setpoint mapping compares the injected frequency with the frequency width of the frequency band, and computes a percentage that can be mapped to a real system variable. The rate mapping compares the injected frequency, not only in terms of the frequency band, but also with respect to the band's center frequency. Instead of producing a reference or setpoint, this method would set a positive or negative rate of change for the respective variable. For the rate case, the maximum frequency of that band would be defined as the maximum rate of change for the assigned variable.

For the setpoint mapped parameter, consider P_1 from Equation (3). To produce a value from 0% to 100%, the encoding (mapping a parameter to a frequency) and decoding (mapping a frequency back to a parameter) are defined by Equation (4) and Equation (5) respectively, where P_1 is a unit-less value.

$$f_1 := \left[\left(\frac{P_1}{100\%}\right)(f_{max} - f_{min})\right] + f_{min}, \;\; s.t.\; P_1 \in [0, 100] \tag{4}$$

$$P_{setpoint} = P_1 := \tfrac{f_1 - f_{min}}{f_{max} - f_{min}} \times 100\% \tag{5}$$

In the case of the rate mapped parameter, the intention is to produce a value from −100% to 100%; the encoding Equation (6) and decoding Equation (7) are a modification of Equation (4) and Equation (5):

$$f_1 := \left[\left(\tfrac{(P_1+100)}{2\times 100\%}\right)(f_{max} - f_{min})\right] + f_{min}, \;\; s.t.\; P_1 \in [-100, 100] \tag{6}$$

$$P_{rate} = P_1 := 2P_{setpoint} - 1 \tag{7}$$

It is not uncommon that the value parameter is itself the selection parameter. For example, if one parameter of the system needs constant adjustment, it makes sense to dedicate a frequency band to that parameter. The rate and setpoint parameters would be commonly used, although not exclusively, in the constant adjustment case.

2.3. Communication Method

A detailed evaluation of the communication method is presented here. In most cases, the frequency space will consist of multiple commands, but the method is presented here consisting of a single command with multiple parameters. Therefore, the command space will be the frequency space.

Consider for a two-parameter command, each parameter has its own frequency band, thus the command space is comprised of two frequency bands with a null-frequency band in between described by Equation (8), and shown in Figure 3.

$$\begin{cases} C_{sp1} \in [f_{min}, f_{max}], \;\; f_1 \in [f_{min}, f_{null-}], \;\; P_1 \in [1, 3], \\ f_2 \in [f_{null+}, \; f_{max}], \;\; P_2 \in [0, 100], f_{null} \in [f_{null-}, f_{null+}], \\ f_{null-} = f_{null} - \epsilon, \;\; f_{null+} = f_{null} + \epsilon, \\ \epsilon = \frac{f_{null+} - f_{null-}}{2} \end{cases} \tag{8}$$

where, with respect to the null-frequency band, f_{null-} and f_{null+} describe the upper and lower frequency limits, f_{null} is the center frequency, and ϵ is the half interval width. Consider the first parameter P_1 to be a selection parameter comprised of three sub-parameters ($n_{p_1} = 3$) as defined in Equation (3), but with the intervals defined in Equation (8), and the second parameter, P_2, to be a setpoint value parameter.

Considering the two-parameter case defined above, LPC will describe the method of communicating the parameters of the command over the DC microgrid. The injected frequency for each parameter of the command space are produced at the same time for some time equal to the latency, where each parameter belongs to its own frequency band (Figure 3b). This can mathematically be seen in Equation (9).

$$V_{CMD}(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t) \tag{9}$$

where $V_{CMD}(t)$ is the command voltage superimposed over the DC bus, as was seen in Equation (1). In this format, commands are difficult to misinterpret, and since all parameters are available for decoding, the main contributor of latency for this method is the decoding process. These advantages come at the expense of a non-optimized utilization of the frequency space, since each command's parameters have their own respective frequency band. The parameterization depth can be reasonably high along with the quantity of sub-parameters a selection parameter can contain, however this is somewhat dependent on the frequency space consumed, which can be the bottleneck.

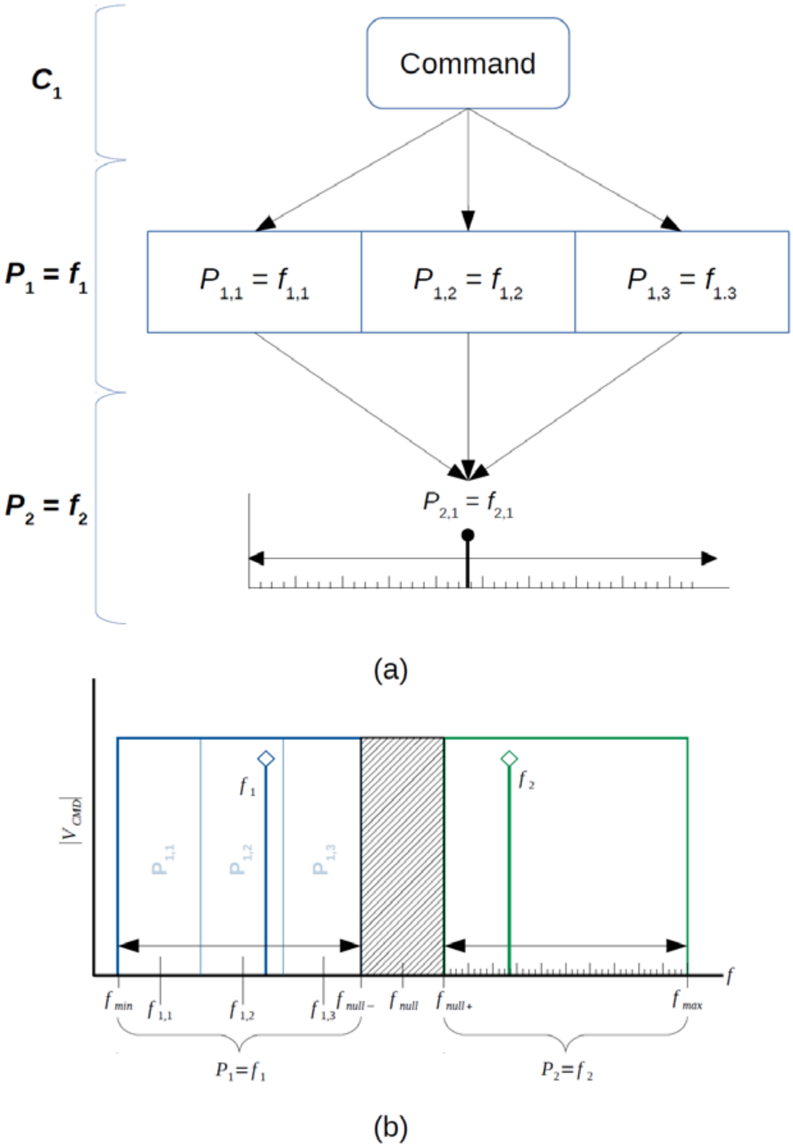


Figure 3. Example two-parameter command. Command tree (**a**) and frequency allocation graph (**b**) for each parameter.

2.4. Channel Separation, Addressability, Feedback, and Feasibility

This section examines additional properties of LFCC that could be implemented: channel separation, addressability, feedback, and feasibility. Channel separation takes advantage of system variables to expand the number of commands, and possibly differentiate between them, addressability assesses how and if each CD can be individually controlled with LFCC, feedback allows a form of response between all CDs, and feasibility examines signal to noise ratio (SNR) and channel attenuation. Each property may be utilized independently or together.

Channel separation will utilize different parts of the DC microgrid system, in order to expand the frequency space. The maximum frequency of the frequency space will typically stay the same, however, the same frequency space can be utilized elsewhere allowing a possible doubling or tripling of commands and or parameters. This may allow certain devices to communicate exclusively or enable a bidirectional communication structure thus better enabling feedback. This can be done in at least two ways: current injection and utilization of multi-level voltage structures. The current injection would prove to be more difficult than the voltage injection and thus it will not be discussed, however it is mentioned here as a possible method. The more likely form of channel separation will appear in the multi-level voltage structure commonly implemented in DC microgrids as a bipolar voltage structure. Therefore, as long as each voltage rail can be independently controlled, one frequency space can be implemented on the positive voltage rail, while the other can be performed on the negative voltage rail.

Addressing may be desirable if more customized commands are needed to be sent. Additionally, it may allow CDs to send back state and fault information. If the quantity of CDs can be reasonably made as the sub-parameters of a parameter, then individual addressing of each CD can be performed, by assigning that as the primary parameter of a command. Likewise, if bidirectional communication occurs, this method also allows the CD to identify itself. However, for larger scale microgrids, this is not feasible. But even if individual addressing cannot be performed, regional addressing can, being performed the same way as individual addressing. Typically, the microgrid will serve a community, which can be organized via streets, cul-de-sacs, or quadrants, and addressing it this way can still allow a more customized control, and can take advantage of characteristics of that grouping.

Feedback is explained third, due to its usual need for individual addressing, and therefore it will have to utilize the addressability and channel separation paragraphs. Feedback, in this sense, aims at certifying if the message from the transmitter has been received and, more so, correctly. While this can be implemented, especially taking advantage of channel separation, it is not feasible on a large scale (>10–20 CDs), therefore, feedback should be performed implicitly. The parameters LFCC adjusts will ultimately affect the operating state of the microgrid, and, assuming the number and type of CDs are known, a predicted response can be estimated with respect to the current command. Therefore, if the microgrid behaves abnormally to the command, then either there may exist erroneous or missed communication, incorrect encoding or decoding settings, or a possible nefarious actor.

The assessment of SNR and channel attenuation are quantitatively outside the scope of this paper; however, they will be addressed in order to understand the general implications towards this communication method. SNR is mainly determined from the transmitter, receiver, and transmission medium. For this method, the transmitter and receiver present the most significant effect towards SNR, whereas the DC microgrid transmissions lines can be neglected for this analysis. Furthermore, the frequencies used with LFCC are so far removed from typical conduction sources that interference is also highly unlikely (but should it be, the null-frequency band exists for this reason). Transmitting will mainly depend on the sensitivity of voltage feedback transducers, ADC resolution and quantization, and power electronic switch drive resolution. Similarly, for the receiver, it will also depend on voltage transducers and their corresponding ADC performance, alongside the discrimination ability of the chosen decoding algorithm. Another strength of injecting at such low frequencies is the minimal channel attenuation, or attenuation through the DC microgrid power lines, that occurs. At frequencies utilized by LFCC, the resistive losses in the cable will be the dominant impedance, as inductive impedance is several orders of magnitude smaller, and capacitive impedance remains significantly large. Furthermore, skin effect has a negligible effect (on the order of hundreds of microvolts) towards the resistance of the power lines. Due to this, it is not unexpected to only see 0.05 dB of loss per every kilometer, which accounts for less than a 0.5% signal loss per kilometer.

3. Detection Method

This section outlines how to detect the injected frequencies based on the methods presented in the previous section. It must be noted that the injection method is not covered in great detail due to its simplicity: once the parameter to frequency encoding process is complete (either from Equation (3), Equation (4), or Equation (6)), the DC microgrid regulating converter's controller reference will include a sine wave generator to be added to the DC reference. And since the commands are comprised of sine waves, the detection can be limited to single-tone detection methods, as only one tone will be present per frequency band. Of course, there will be multiple tones present for LPC or just multiple commands in general. Regardless, the method for detection can still be of the single-tone method.

This section is not a survey on all possible methods, but rather exemplifies one possible technique, which can be replaced by another if desired. Possible alternative detections methods may include zero-crossing detection, band-pass filtering with envelope detection, or spectral techniques such as the zoom or sliding DFT. This paper utilizes an infinite impulse response (IIR) bandpass filter [8], followed by a phase-locked loop (PLL) that uses a second order generalized integrator (SOGI) as the phase detector [9–13]. This is chosen due to its robustness and low computational burden as an additional control loop, where the SOGI phase detection helps alleviate the digital input filter burden due to its natural bandpass filtering around its resonant frequency. PLLs are also used, due to the frequency-tracking requirement from the value parameters. Additionally, since most power system engineers that work with power electronics have worked with AC grids, PLLs would be a method the engineer is most likely familiar with. For multiple frequencies, a multiplexed detection method is discussed to allow evaluation of several frequency bands over time, while not increasing computational load at the expense of additional ROM space. This detection method works for both the layered and packet-based commands.

3.1. Filter and PLL Structure

The algorithm consists of two main parts for frequency detection: the filtering/signal conditioning stage and the actual PLL stage. The filter/signal condition stage removes the DC component, bandpass filters the signal within the frequency space, in order to isolate the signal from other injected frequencies and noise, and finally normalizes the amplitude to $1V_{peak}$ with an automatic gain control (AGC) [14], in order to avoid issues with the PLL. The SOGI-PLL block diagram is depicted in Figure 4. The PLL is constructed from a phase detector, or orthogonal signal generator (OSG), park transformation, loop filter (PI filter), and VCO. The SOGI algorithm is what makes up the phase detector. The output of the VCO provides phase information to the park transformation and the output of the PI filter provides the detected frequency. The frequency measurement block filters the frequency signal in order to provide a more stable measurement and also determines if the signal is stable enough to consider the PLL locked. The lock state is determined by measuring the rate of change of the frequency signal, and if there is a frequency component of significant amplitude by measuring v_d from the park transform block. This condition frequency (f_{cond}) is also fed back to the SOGI block, making it frequency adaptive, in order to tune the SOGI resonant frequency to increase performance.

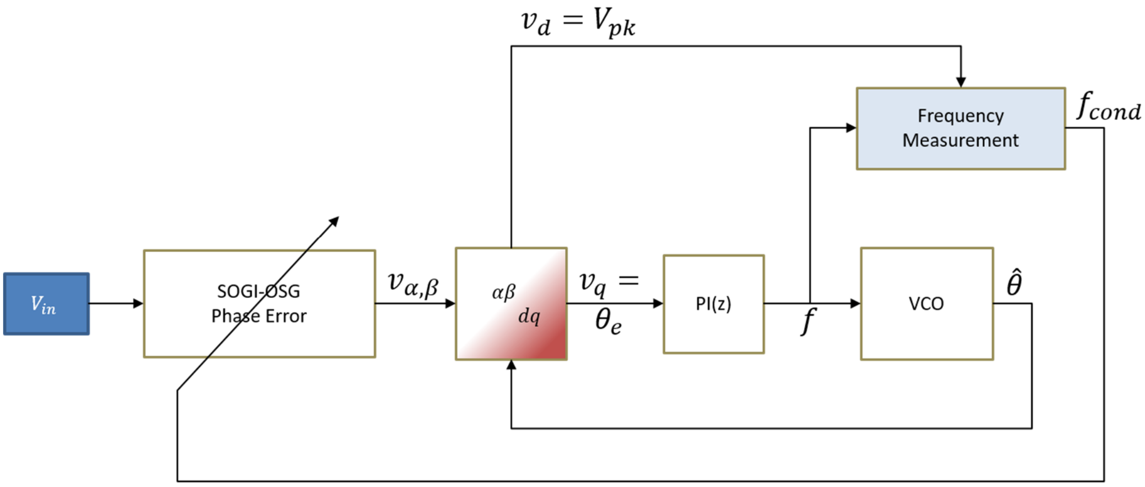


Figure 4. Block diagram of the second order generalized integrator (SOGI) phase locked loop (PLL) (SOGI-PLL).

3.2. Multiplexed Detection Scheme

The multiplexed detection scheme allows the SOGI-PLLs to be scheduled to run in a staggered manner. An example of this is demonstrated in Figure 5 for a three PLL case. This essentially down-samples the SOGI-PLL by the total number of PLLs, and allows only one SOGI-PLL to run per control cycle. Considering the control loop period is significantly shorter than the detected frequencies, it is possible to allow for 10–15 SOGI-PLLs in a single program to run properly, before sampling issues arise. Furthermore, it is also possible to reduce the order of the digital filters, since the order typically falls with decreasing sampling frequency. Since the signal conditioning and SOGI-PLL are parametric, this approach can be thought of as gain scheduling, where each frequency band has its own corresponding gains and coefficients.

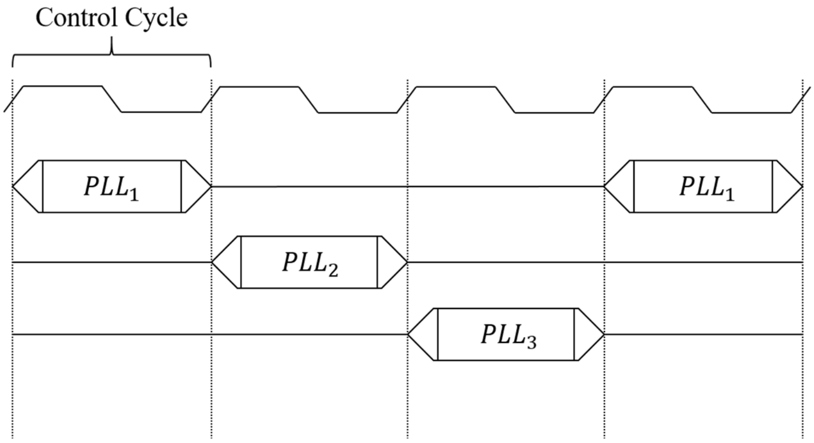


Figure 5. PLLs operated under the multiplexed detection scheme.

4. Experimental Results

This section describes the results found from the experimental implementation. The detection method was first tested and evaluated in a Matlab/Simulink environment, and the experiments were carried out on an Imperix rapid prototyping power electronics platform. The Imperix system consists of control hardware (BoomBox), which utilizes a TI DSP (TMS320C28346) that can be used with Simulink's code generation feature. A scale microgrid is designed using Imperix half-bridge modules (PEB8032) and LEM voltage and current transducers to design appropriate DC-DC converters for DC transmission line interfacing and battery charging/discharging. For the experimental setup, a CES (community energy source) and three houses are designed with the half-bridge modules and transducers. These half-bridge modules comprise the corresponding buck and boost converters for each CD for the positive and negative rail. Each half-bridge is rated for 800 V, 32 A, up to 10 kVA, and is operating at 20 kHz. The CES can be powered from lab power supplies or a Power Hardware in the Loop (PHIL) interface, and the house loads are also connected to a PHIL interface, which mimic DC loads. The remaining houses of the microgrid can be simulated and represented cumulatively as a single PHIL interface. To emulate distance from the CES to the house, an equivalent impedance PHIL interface. To emulate distance from the CES to the house, an equivalent impedance circuit that represents 400 m is placed between each connection. This circuit is designed to have inductance, capacitances, and resistance values of 66 uH, 4.7 nF line-to-line, 22 nF line-to-neutral, and 22 mΩ per 100 m. Since this experiment is being conducted on House 2, there will be a total of 800 m equivalent impedance from the CES. This aspect and all salient details of the experiment can be seen in Figure 6. The physical setup can be seen in Figure 7.

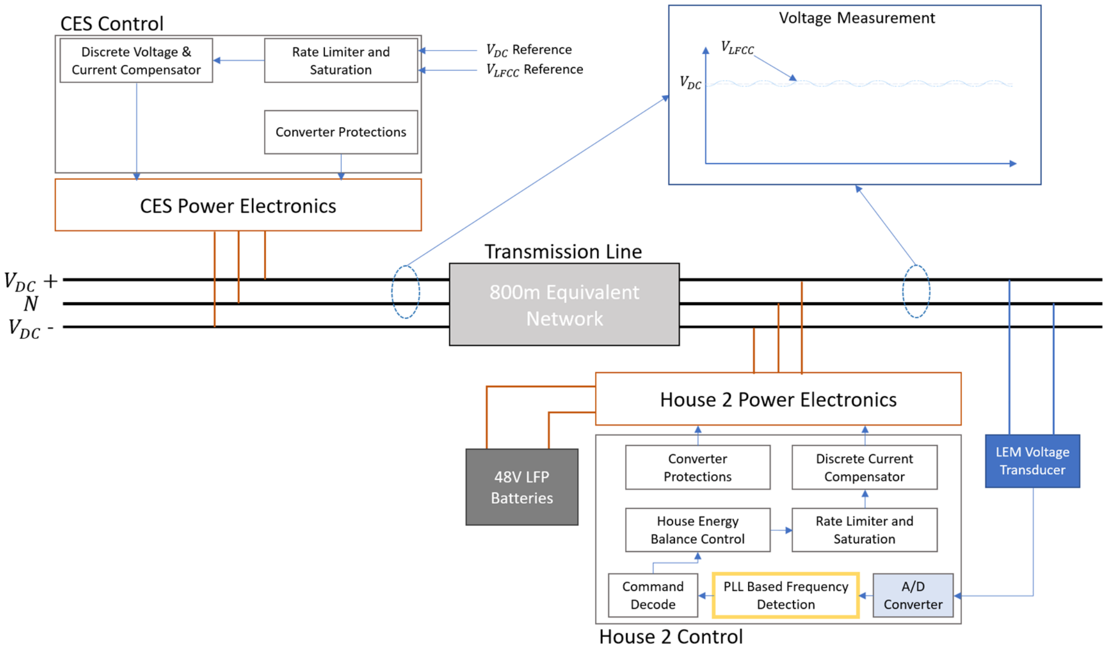


Figure 6. Live-scale DC microgrid experimental setup diagram.

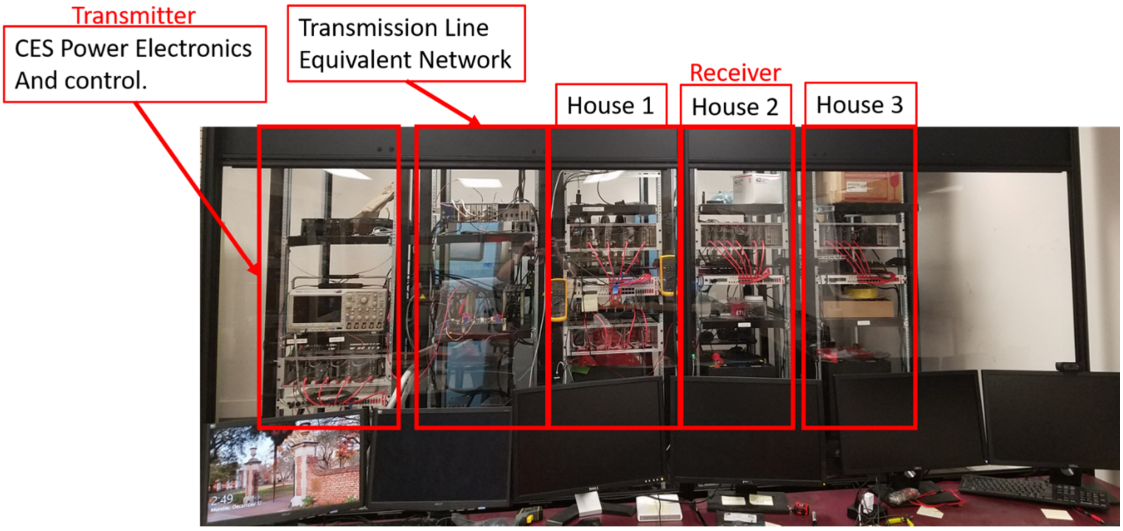
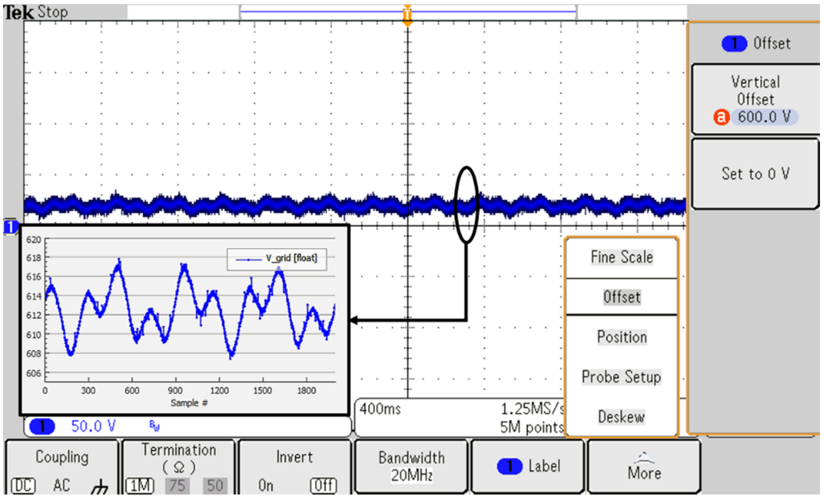

Transmitter
CES Power Electronics And control.
Transmission Line Equivalent Network
Receiver
House 1
House 2
House 3

Figure 7. Live-scale DC microgrid physical setup.

The experiment will demonstrate assigning numeric values to three variables using the layered parameter communication method, while 1 A is being drawn from the DC grid (CES), and all connected converters are actively switching. The CES will inject two simultaneous frequencies at $2V_{peak}$ on the positive ~600 VDC rail (Figure 8), while House 2 will detect the frequencies with two simultaneously running SOGI-PLLs. This experiment will utilize the command tree example shown in Figure 3, where the command and parameter spaces are numerically defined in Equation (10). The command tree with numerical definitions can be seen in Figure 9.

$$C_1 \in [3,\ 15],\ \ P_1 \in [3,\ 7],\ \ P_2 \in [9,\ 15] \tag{10}$$



Vertical Offset 600.0 V

50.0 V

400ms 1.25MS/s 5M points

Bandwidth 20MHz

Figure 8. Oscilloscope screen capture of 600 V positive DC microgrid voltage rail.

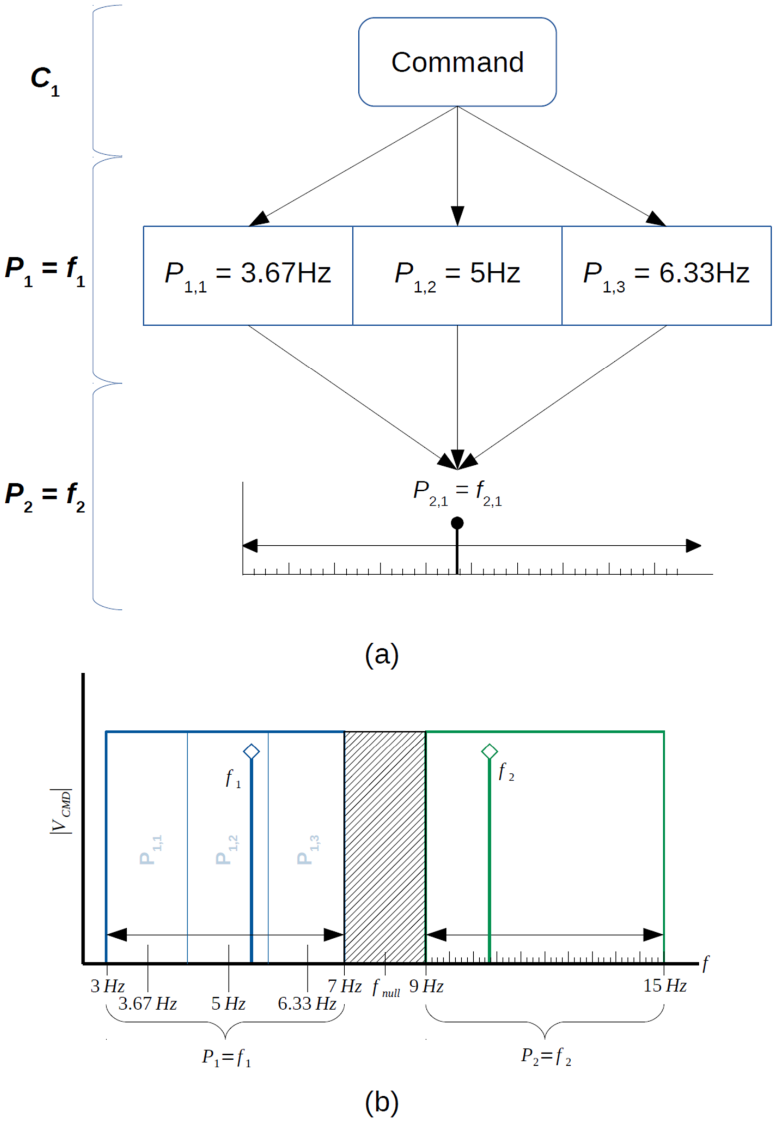


Figure 9. Two-parameter command with numerically defined values defined by the command tree (**a**) and frequency allocation graph (**b**) for each parameter.

The three variables are the three sub-parameters of P_1, where the parameter P_2 is a setpoint mapped parameter, which will be used to define each variable. This demonstration exhibits $P_{1,1}$ being set to 25%, $P_{1,2}$ to 50%, and $P_{1,3}$ to 75% in that order, and then back to 0% in reverse order, which can be seen in Figure 10. Examining the lower graph of Figure 10, PLL_1 is detecting the frequency value that represent the numeric value (P_2) to be written to the sub-parameter (P_1), detected by PLL_2. When the detection of a new parameter occurs, the last state of the previous parameter will be held, since only one PLL is being used to detect the corresponding values for each parameter.

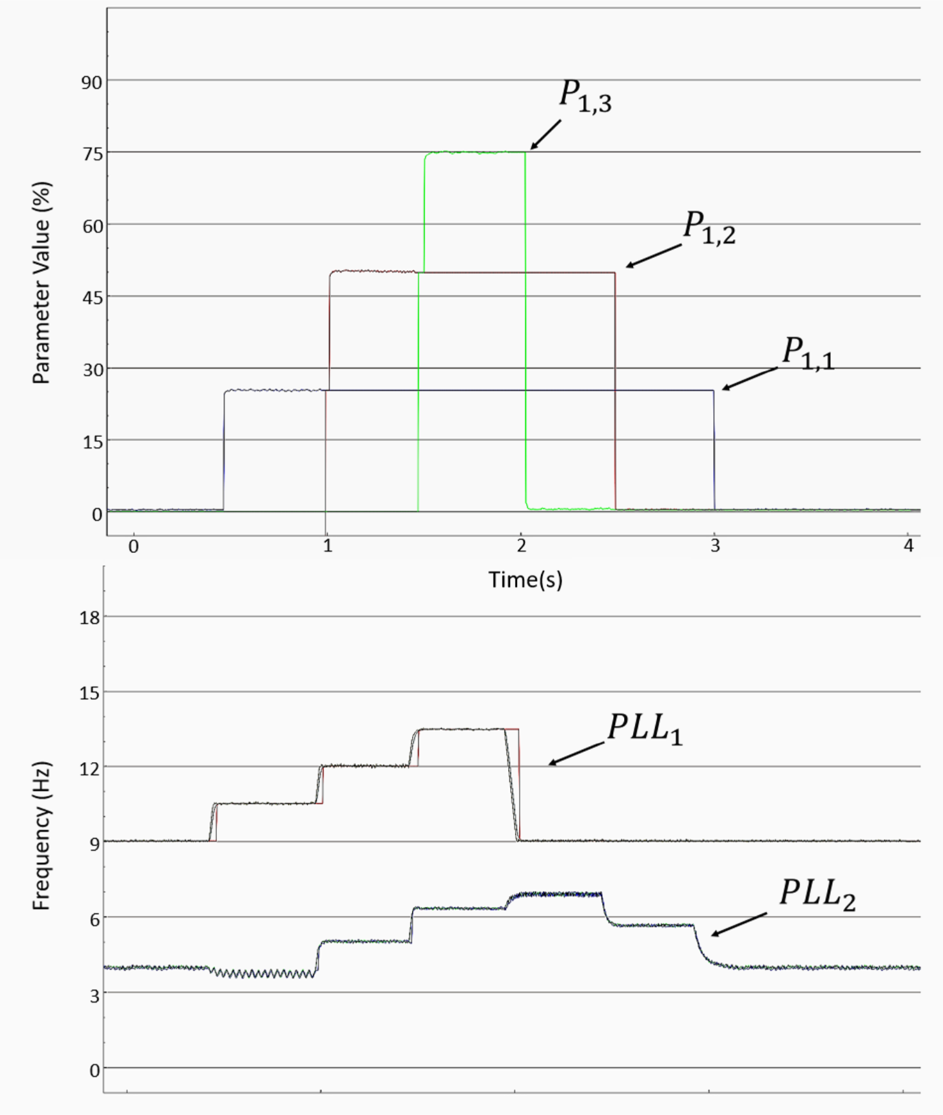


Figure 10. Results of the experiment: Assigning values to three sub-parameters (**upper**) and frequency detection by each PLL (**lower**).

The decoding process can be viewed in the upper graph of Figure 10. Notice that PLL_2 seems to detect different values for the same parameter (the second half has higher frequency detection than the first half). This is not because the CES is injecting a different frequency, but because the parameter space that determines P_2 (detected by PLL_1) is so close to the other parameter space that it affects the accuracy of the other PLL. This is left in to demonstrate that due to the partitioning of P_1, the detection for each sub-parameter (or partition) can be robust to some amount of error. This is also left in to advise on the correction of this issue: either the bandpass filter will have to be made more aggressive, either at the stop frequency location or attenuation level, or the null frequency band will need to be made wider. Changing the filters will most likely add some computational cost while increasing the null frequency band width will consume more of the frequency space and may require re-tuning of the PLL for a different frequency band. It may also be noticed that each parameter ($P_{1,1} : P_{1,3}$) sometimes contains some noise, which is due to the direct use of the frequency variable from the PLL control. Therefore, noise and control artifacts will be present if a value parameter is required to have a high amount of resolution; for this experiment, since the frequency variable is being directly used, this represents the worst-case scenario for this setup. Either additional filtering will be required for value parameters of this type or the maximum resolution of the value parameter must be set.

5. Conclusions

This paper demonstrates the use of a simple communication method by injecting low frequency sinusoidal components on a DC microgrid voltage rail. Through only software modifications and the provided method, low-level communication can be effectively achieved by creating an application specific protocol catered to the site of deployment. The ability of channel separation, addressability, and feedback were also assessed for more customized control. Finally, this was demonstrated on a live-scale DC microgrid with actual converters and digital control.

For most systems, commands will be comprised of few parameters and, likewise, parameters will not contain many sub-parameters. This method does not replace dedicated communication lines with either copper or fiber—in other words, communication methods designed for large throughput—but rather allows global or semi-global system parameters to easily be modified. Parameters of interest in a DC microgrid would be setting maximum current consumption or output (for renewables), desired levels of state of charge (SOC) for CDs with grid storage, and better utilization of renewables.

Author Contributions: Conceptualization, M.D. and A.B.; Funding acquisition, A.B.; Investigation, M.D.; Supervision, A.B.; Writing—original draft, M.D.; Writing—review & editing, A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Emera Tecnologies.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Choi, H.J.; Jung, J.H. Enhanced Power Line Communication Strategy for DC Microgrids Using Switching Frequency Modulation of Power Converters. *IEEE Trans. Power Electron.* **2017**, *32*, 4140–4144. [CrossRef]
2. Chao, C.W.; Ho, Q.D.; Le-Ngoc, T. Challenges of Power Line Communications for Advanced Distribution Automation in Smart Grid. In Proceedings of the 2013 IEEE Power & Energy Society General Meeting, Vancouver, BC, Canada, 21–25 July 2013. [CrossRef]
3. Lu, X.; Guerrero, J.M.; Sun, K.; Vasquez, J.C. An Improved Droop Control Method for Dc Microgrids Based on Low Bandwidth Communication with Dc Bus Voltage Restoration and Enhanced Current Sharing Accuracy. *IEEE Trans. Power Electron.* **2014**, *29*, 1800–1812. [CrossRef]
4. Liu, H.; Yang, Y.; Loh, P.C.; Blaabjerg, F.; Angjelichinoski, M.; Stefanovic, C.; Popovski, P. Power Talk: A Novel Power Line Communication in DC MicroGrid. In Proceedings of the 2016 IEEE 8th Int. Power Electron. Motion Control Conf. IPEMC-ECCE Asia 2016, Hefei, China, 22–26 May 2016. [CrossRef]
5. Wu, J.; Du, J.; Lin, Z.; Hu, Y.; Zhao, C.; He, X. Power Conversion and Signal Transmission Integration Method Based on Dual Modulation of DC-DC Converters. *IEEE Trans. Ind. Electron.* **2015**, *62*, 1291–1300. [CrossRef]
6. Kohama, T.; Hasebe, S.; Tsuji, S. Simple Bidirectional Power Line Communication with Switching Converters in DC Power Distribution Network. In Proceedings of the 2019 IEEE International Conference on Industrial Technology (ICIT), Melbourne, Australia, 13–15 February 2019; pp. 539–543. [CrossRef]
7. Monti, A.; De Din, E.; Muller, D.; Ponci, F.; Hagenmeyer, V. Towards a Real Digital Power System: An Energy Packet Approach. In Proceedings of the 2017 IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, China, 26–28 November 2017; pp. 1–6. [CrossRef]

8. Aboagye, A.K. *Overflow Avoidance Techniques in Cascaded IIR Filter Implementations on the TMS320 DSP's*; Technical Report; Texas Instruments: Dallas, TX, USA, May 1999; pp. 1–17. Available online: http://www.ti.com/lit/an/spra509/spra509.pdf?ts=1588655003153 (accessed on 12 January 2019).
9. Możdżyński, K.; Rafał, K.; Bobrowska-Rafał, M. Application of the Second Order Generalized Integrator in Digital Control Systems. *Arch. Electr. Eng.* **2014**, *63*, 423–437. [CrossRef]
10. Ciobotaru, M.; Teodorescu, R.; Blaabjerg, F. A New Single-Phase PLL Structure Based on Second Order Generalized Integrator. In Proceedings of the 2006 37th IEEE Power Electronics Specialists Conference, Jeju, Korea, 18–22 June 2006. [CrossRef]
11. Xiao, F.; Dong, L.; Li, L.; Liao, X. A Frequency-Fixed SOGI-Based PLL for Single-Phase Grid-Connected Converters. *IEEE Trans. Power Electron.* **2017**, *32*, 1713–1719. [CrossRef]
12. Kulkarni, A.; John, V. A Novel Design Method for SOGI-PLL for Minimum Settling Time and Low Unit Vector Distortion. In Proceedings of the IECON 2013—39th Annual Conference of the IEEE Industrial Electronics Society, Vienna, Austria, 10–13 November 2013; pp. 274–279. [CrossRef]
13. Renesas. Loop Filter Component Selection for VCXO Based PLLs. In *Application Note Renesas*; Renesas: Tokyo, Japan, 2020; pp. 1–9. Available online: https://www.idt.com/us/en/document/apn/849-loop-filter-component-selection-vcxo-based-plls (accessed on 13 February 2019).
14. Lyons, R. *Understanding Digital Signal Processing*, 3rd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2010.