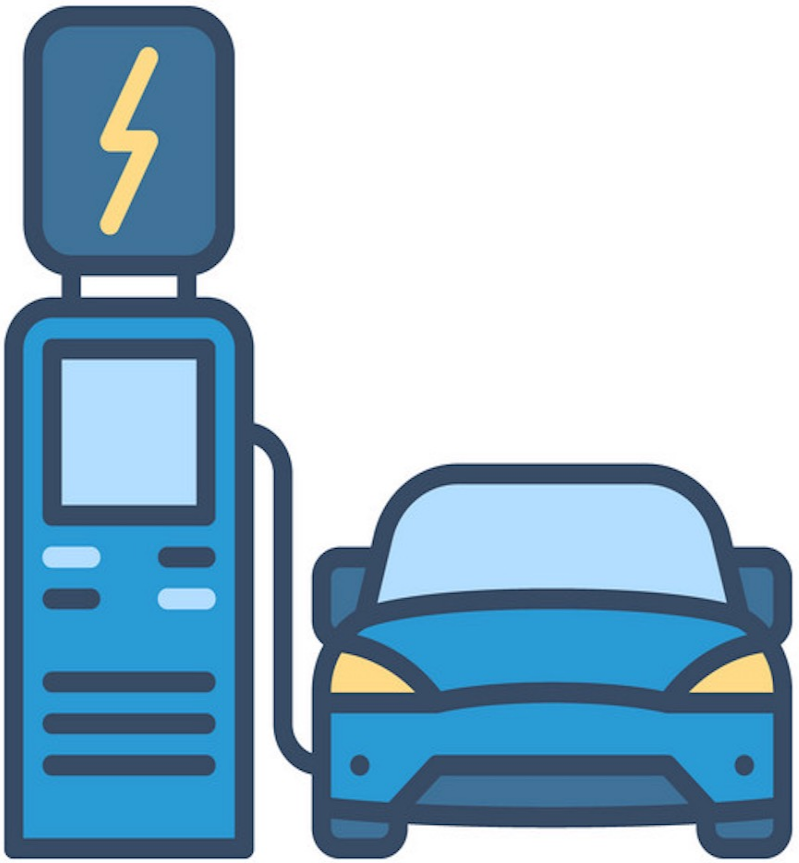
MSc Thesis in Electrical Power Engineering

IoT Based Online Harmonic Emission Estimation of DC Fast Chargers

Yawen Liang
2022



TU Delft

IoT Based Online Harmonic Emission Estimation of DC Fast Chargers

by

Yawen Liang
Student number: 5281997

in partial fulfillment of the requirements for the degree of

Master of Science
in Electrical Engineering, specialized in Electrical Power Engineering

at Faculty of Electrical Engineering, Mathematics & Computer Science,
Delft University of Technology

to be defended publicly on Monday July 11th, 2022 at 2:00 PM

Supervisor:	Prof. dr. ir. Pavol Bauer,	TU Delft
Daily supervisor:	Dr. ir. Zian Qin,	TU Delft
Thesis committee:	Prof. dr. ir. Pavol Bauer,	TU Delft
	Dr. ir. Zian Qin,	TU Delft
	Dr. Aleksandra Lekic,	TU Delft

ABSTRACT

Nowadays, electric vehicles (EVs) in the market only travel a short distance and are usually charged at home which can be the obstacle in the way of EVs development. As a result, fast charging that can ensure long-distance drive with shorter charging time draws more and more attention.

In emerging fast-charging stations, DC fast chargers (DCFCs) are employed which rely on power electronics and control to achieve the required performance. Harmonic current emission induced by the complex system behavior is of great concern in the DC fast charger (DCFC) system. This thesis proposes a harmonic emission model for the typical DCFC design, i.e., the two-level active front end. The technique is based on the Fourier series method and the impedance model which is able to reveal the harmonic current emission of DCFCs under different grid conditions. Time-domain simulations and experiments are presented subsequently to validate the proposed model.

The analytical model can be implemented in cloud-based charging, which is a popular topic right now because of the lower overall cost. Besides, cloud-based systems have a larger storage capacity and are easier to maintain. Another reason for the appealing cloud-based charging is that all of these Internet of things (IoT) devices are interconnected, and vehicle data are shared. Therefore, an IoT based online harmonic emission estimation tool for DCFC is built based on the aforementioned proposed model in this thesis.

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor Prof. dr. ir. Pavol Bauer and Dr. ir. Zian Qin for their continuous help on the research direction of my thesis. At the same time, I would like to thank Ph.D. Candidate Lu Wang, who gave me timely guidance on the difficulties and doubts I encountered during the thesis writing process, put forward many useful suggestions for improvement and invested a lot of effort and energy. I would like to express my sincere gratitude for their help and care.

In addition, I would like to thank my friends and parents for their great support and help in the preparation of my thesis, which has brought me great inspiration. I would also like to thank the authors in the references for giving me a great favor for my research topic through their research articles.

Finally, I would like to thank my thesis committee for their hard work, and the guiding comments and suggestions.

Yawen Liang
Delft, July 2022

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

1

INTRODUCTION

1.1 BACKGROUND

With the roll-out of EVs, a massive installation of FCSs is expected [1]. However, same as wind turbine inverters and photovoltaics (PV) inverters [2, 3], DCFCs installed in FCSs are essentially power electronic based devices leading to power quality degradation because of their harmonic current emission [4]. Thus, harmonic current emission modeling is essential which avails the analysis of the root cause of DCFCs' harmonic emission and the design of DCFCs to prevent severe harmonic issues.

Harmonic spectrum for the two-level converter has been discussed based on the double Fourier series method in literature [5, 6, 7]. However, most of them focus only on high frequencies. The low order harmonics are thought to be easily eliminated by current control [8]. Nevertheless, such neglect of low order harmonics may be incorrect when the low order harmonics are amplified because of the resonance in a charger-grid system. Hence, low-frequency harmonics modeling is studied in this thesis.

Besides the harmonics calculation, cloud-based charging is a popular topic for now because of the reduced overall cost. Moreover, cloud-based solutions own a bigger storage capacity and are easier to maintain, i.e., it is simple to update the cloud system. Another reason that accounts for the popularity of cloud-based charging is that all these IoT devices are interconnected, and vehicle data are shared which can be used for optimization. Therefore, IoT based online harmonic emission tool for DC fast chargers is another topic of this thesis.

1.2 RESEARCH OBJECTIVE

This thesis aims at building an IoT platform to achieve the harmonic emission estimation which can facilitate the operation of DCFCs, as shown in Fig. 1.1. To be more specific, when the charge point operators input the desired charging power on the IoT platform, the IoT platform will estimate the harmonics based on the parameters of the EV charger and the charging power. If the calculated

harmonics exceed the harmonics limit, the EV charger will not start to charge the EV. If the calculation result satisfies the harmonic requirement, the charger will be allowed to charge the EV at that power level and the IoT platform will display the harmonic spectrums on the webpage which is visible to the user.

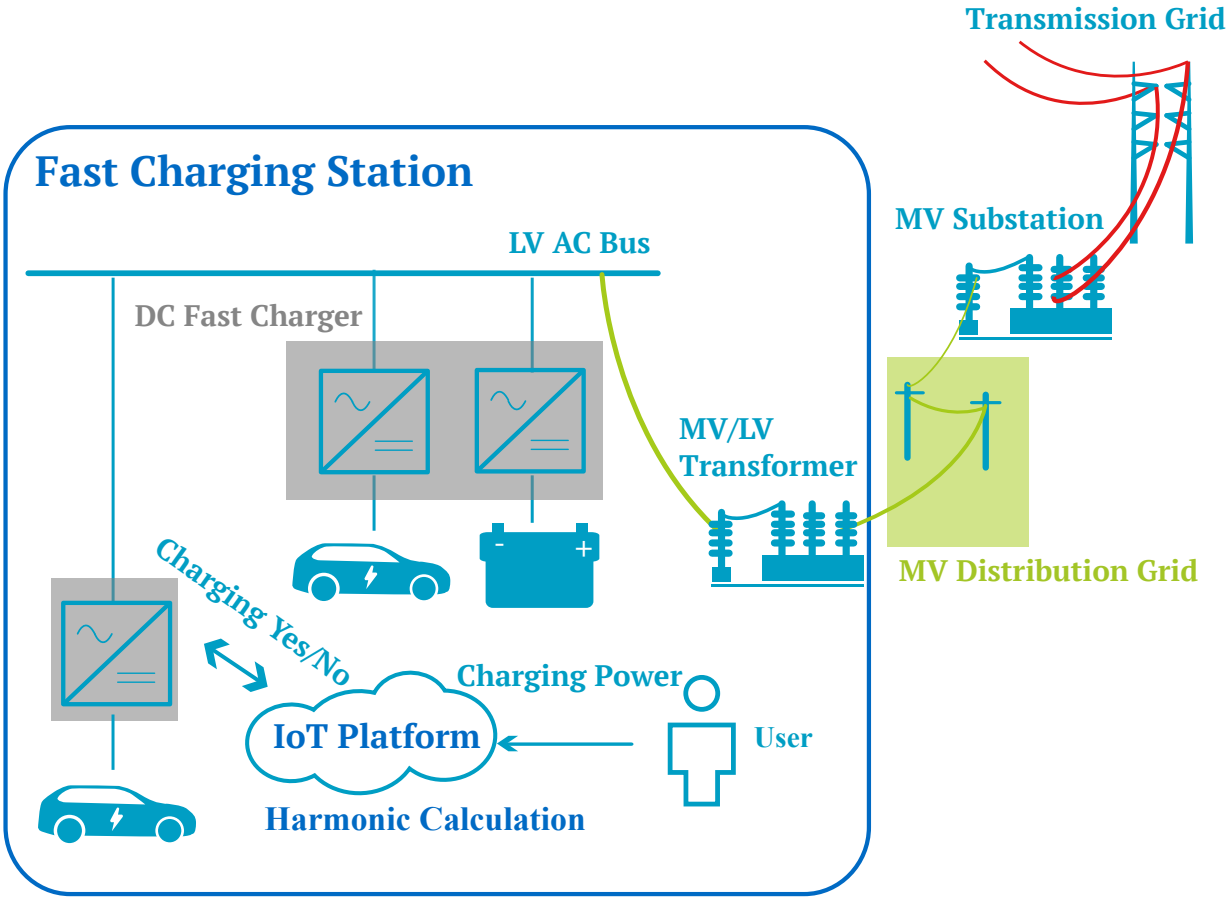


Figure 1.1: The objective of this master thesis.

1.3 OVERVIEW OF THE INVESTIGATION APPROACH

Generally speaking, the main purpose of this thesis is to propose an online harmonic emission estimation tool for DCFCs. To build the online IoT platform with the function of harmonic emission estimation for DCFCs, the first step is to study the grid impact of EV fast charging stations. Then previous work on input impedance modeling for the typical AC/DC converter of DCFCs is understood. Thirdly, the current harmonic emission of the AC/DC converter of DCFCs needs to be

modeled. Simulations and experiments will be conducted subsequently to verify the validity of the proposed model. Fifthly, a backend tool in python for harmonic calculation is created based on the previous analytical modeling. Finally, build an IoT platform for device-to-cloud messaging based on the Microsoft Azure service and merge the backend tool into the IoT platform.

1.4 STRUCTURE OF THE WORK

The thesis consists of the following six parts:

1. Chapter 1 introduces the background, the research objective and the overview of the investigation approach.
2. Chapter 2 elaborates on the impedance modeling of the DC fast charger system.
3. Chapter 3 presents the harmonic emission modeling of the DC fast chargers.
4. Chapter 4 conducts the time-domain simulations and experiments to validate the proposed model.
5. Chapter 5 presents the IoT platform design.
6. Chapter 6 summarizes the work and draws conclusions based on the output results of every stage. Besides, future work is given.

2

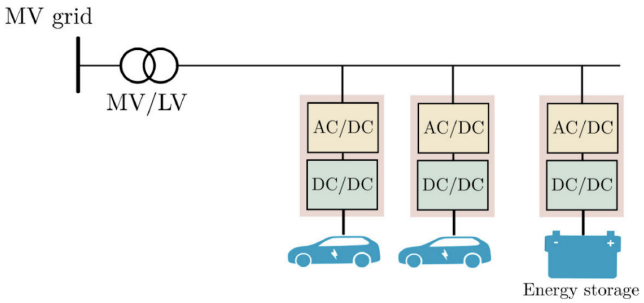
INPUT IMPEDANCE MODELING OF DC FAST CHARGERS

To estimate the harmonic emission induced by DCFCs, the impedance method is applied. In this chapter, the detailed input impedance modeling of the AFE is given where the influences of the SRF-PLL, the inner current control loop, the outer voltage control loop, and the DC-link voltage fluctuation are thoroughly discussed.

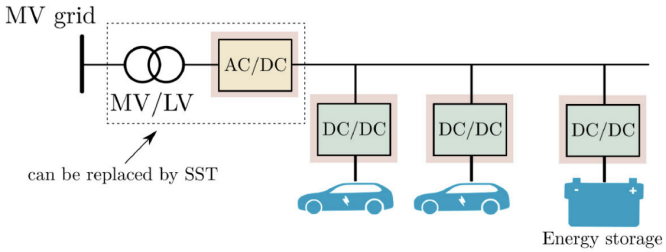
2.1 ARCHITECTURE OF THE DC FAST CHARGERS SYSTEM

In general, the distribution network supplying the FCSs is medium voltage (MV). This voltage is converted to low voltage (LV) by a MV/LV service transformer or a power electronics-based solid-state transformer (SST). As shown in Fig. 2.1, the LV distribution network can be either AC or DC depending on the structure [9]. FCSs with AC power distribution network is investigated in this thesis because this structure is mature and also the mainstream solution for now.

In the FCSs with AC distribution network, there are typically two power conversion stages in the DCFC, one front-end AC/DC stage and one DC/DC stage. Between these two stages, large DC-link capacitors are applied to stabilize the DC voltage. Therefore, the power quality impact of DCFC is mainly determined by the AC/DC stage since the DC/DC stage is decoupled by the DC-link capacitor. In one stage, there are several converters encompassed, thus all the following analysis focuses on the AC/DC converter. For the AC/DC converter, the popular implementation includes Vienna rectifier, two-level AFE, and multi-pulse rectifier [10]. One of the typical designs, i.e., two-level AFE, which is considered in this thesis, is illustrated in Fig. 2.2.



(a) The structure of typical FCSs with AC distribution network.



(b) The structure of typical FCSs with DC distribution network.

Figure 2.1: The structure of typical FCSs.

2.2 CONTROL OF ACTIVE FRONT-END INCLUDING PHASE-LOCKED LOOP

Of interest is the control system of the AFE shown in Fig. 2.2 because the input impedance of the AFE needs to be modeled. The typical cascaded control scheme is applied for the AFE, in which both the outer voltage control loop and the inner current control loop utilize the Proportional Integral (PI) controller while the outer control loop is responsible for regulating the DC-link voltage u_{dc} to be a constant, and the inner control loop controls the AFE side current dynamics. For grid synchronization, SRF-PLL is used.

2.2.1 *Grid Synchronization*

As the phase angle of the grid voltage is necessary when adopting the dq-frame transformation, the grid synchronization is one of the critical parts of the grid-tied AFE control strategy. In this thesis, SRF-PLL serves as the grid synchronization part which is implemented in the dq-frame.

There are three parts consisting of the SRF-PLL, as shown in Fig. 2.3, which includes phase detector, low-pass filter, and voltage-controlled oscillator [11]. The phase detector intends to

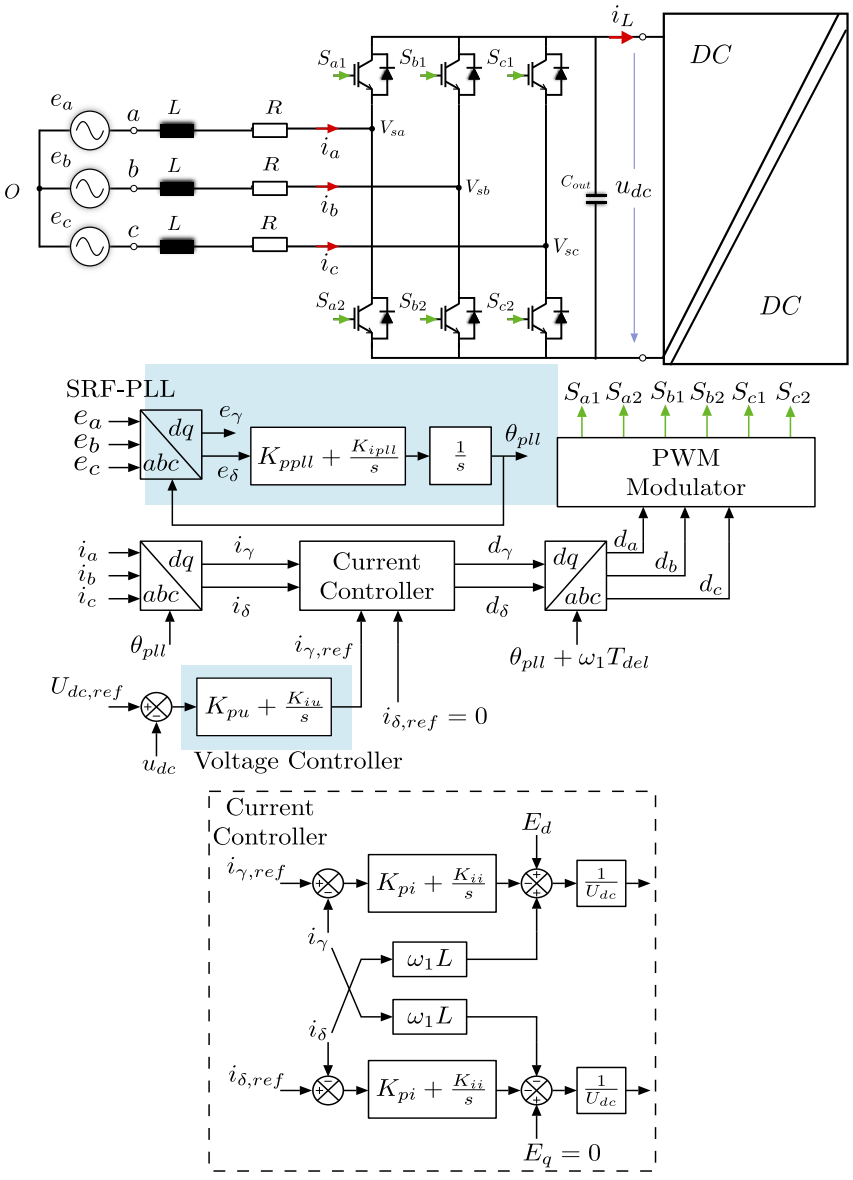


Figure 2.2: AFE of EV charger with the control block diagram.

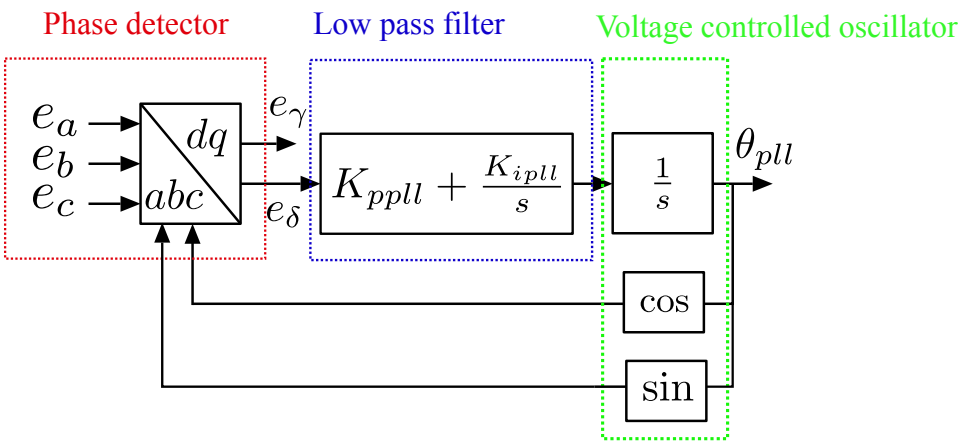


Figure 2.3: Schematic of the SRF-PLL.

transform the three-phase grid voltage e_a, e_b and e_c into the two-phase voltage e_γ and e_δ in dq-frame based on θ_{PLL}. The PI controller works as the low-pass filter in SRF-PLL. The input of the low-pass filter is the q-axis grid voltage e_δ which is suppressed to zero, so that the grid voltage can be aligned with the d-axis, in other words, the power factor is 1. The output of the low-pass filter is the angular frequency ω_1 of the grid voltage. With the integrator in the voltage-controlled oscillator, the angular frequency ω_1 is converted to the phase angle θ_{PLL}. $\cos\theta_{PLL}$ and $\sin\theta_{PLL}$ is delivered to the phase detector for Park transformation.

2.2.2 *Inner Current Control Loop*

As mentioned before, the inner current control loop is employed in the grid-tied converter system for AFE side current harmonics control. First, to understand the inner current control loop better, the average model for the AFE is used.

$$\begin{cases} e_a = L\frac{di_a}{dt} + Ri_a + U_{dc}(d_a - \frac{d_a+d_b+d_c}{3}) \\ e_b = L\frac{di_b}{dt} + Ri_b + U_{dc}(d_b - \frac{d_a+d_b+d_c}{3}) \\ e_c = L\frac{di_c}{dt} + Ri_c + U_{dc}(d_c - \frac{d_a+d_b+d_c}{3}) \\ C_{out}\frac{d}{dt}U_{dc} = d_a i_a + d_b i_b + d_c i_c - i_L \end{cases} \tag{2.1}$$

where e_a, e_b, and e_c are the input grid voltages, L is the input filter inductance, R is the input filter resistance which is induced by the cable and connection, i_a, i_b, and i_c are the AC phase currents,

U_{dc} is the DC-link voltage, d_{a}, d_{b}, and d_{c} are the phase leg average duty cycle [12], C_{out} is the output capacitance, and i_{L} is the load current.

By adding perturbation to the steady state of the AFE, the small-signal model of the AFE in dq-frame can be achieved,

$$\begin{cases} \tilde{e}_\gamma = L\frac{d\tilde{i}_\gamma}{dt} + R\tilde{i}_\gamma - \omega_1 L\tilde{i}_\delta + D_\gamma\tilde{u}_{dc} + U_{dc}\tilde{d}_\gamma \\ \tilde{e}_\delta = L\frac{d\tilde{i}_\delta}{dt} + R\tilde{i}_\delta + \omega_1 L\tilde{i}_\gamma + D_\delta\tilde{u}_{dc} + U_{dc}\tilde{d}_\delta \\ C_{out}\frac{d}{dt}\tilde{u}_{dc} = \frac{3}{2}(D_\gamma\tilde{i}_\gamma + I_\gamma\tilde{d}_\gamma + D_\delta\tilde{i}_\delta + I_\delta\tilde{d}_\delta) - \tilde{i}_L \end{cases} \tag{2.2}$$

where X represents the DC component, and $\tilde{x}$ denotes the perturbation. Assuming the AFE works under stable power grid condition, disturbance from the grid voltage $\tilde{e}_{\gamma,\delta}$ can be ignored. Besides, the DC-link voltage U_{dc} can be considered as a constant without perturbation because the response speed of the inner current control loop is much higher than the outer control loop for DC-link voltage. Based on the aforementioned assumptions, (2.2) can be simplified as,

$$\frac{d}{dt}\begin{bmatrix}\tilde{i}_\gamma \\ \tilde{i}_\delta\end{bmatrix} = -\begin{bmatrix}0 & -\omega \\ \omega & 0\end{bmatrix}\begin{bmatrix}\tilde{i}_\gamma \\ \tilde{i}_\delta\end{bmatrix} - \frac{1}{L}\begin{bmatrix}\tilde{d}_\gamma \\ \tilde{d}_\delta\end{bmatrix}U_{dc} - \frac{R}{L}\begin{bmatrix}\tilde{i}_\gamma \\ \tilde{i}_\delta\end{bmatrix} \tag{2.3}$$

Coupling components can be observed in (2.3), e.g., the control of $\tilde{d}_d$ affects both $\tilde{i}_\gamma$ and $\tilde{i}_\beta$. To decouple the current control loop, modification to the output of the current controller is made,

$$\begin{cases} \tilde{d}_\gamma &= G_{PI}(\tilde{i}_{\gamma,ref} - \tilde{i}_\gamma) - \omega_1 L\tilde{i}_\delta/U_{dc} \\ \tilde{d}_\delta &= G_{PI}(\tilde{i}_{\delta,ref} - \tilde{i}_\delta) + \omega_1 L\tilde{i}_\gamma/U_{dc} \end{cases} \tag{2.4}$$

where the first term of (2.4) is the original output of the PI current controller, and the second term is the manually added compensation term. The modified output yields the simplified current controller system as follows,

$$\frac{d}{dt}\begin{bmatrix}\tilde{i}_\gamma \\ \tilde{i}_\delta\end{bmatrix} = -\frac{1}{L}\begin{bmatrix}\tilde{d}_\gamma \\ \tilde{d}_\delta\end{bmatrix}U_{dc} \tag{2.5}$$

The decoupling control block diagram in dq-frame is shown in Fig. 2.4. It can be observed that feed-forward decoupling loops are added to cancel out the coupling components in the current controller.

By using the grid voltage orientation control, the feed-forward DC bias E_{d} equals the reference peak value of the power grid phase voltage, E_{q} is zero because the reference reactive power determined by the q-axis current is zero.

The pulse-width modulation (PWM) delay is also one crucial part of the overall control block diagram of the inner-loop current control. Since the delay caused by PWM converter is in the order of hundreds of microseconds, the PWM delay can be approximated as a first-order inertia unit. For digital control with symmetric sampling, T_{delay} is $1.5T_{\text{SW}}$ considering the worst case, which is comprised of the computational delay (one sampling period T_{SW}) and the modulation delay (half sampling period $0.5T_{\text{sw}}$). For asymmetric sampling, T_{delay} equals $0.75T_{\text{SW}}$. With the PWM transport delay, the overall block diagram of the inner current control loop is shown in Fig. 2.4.

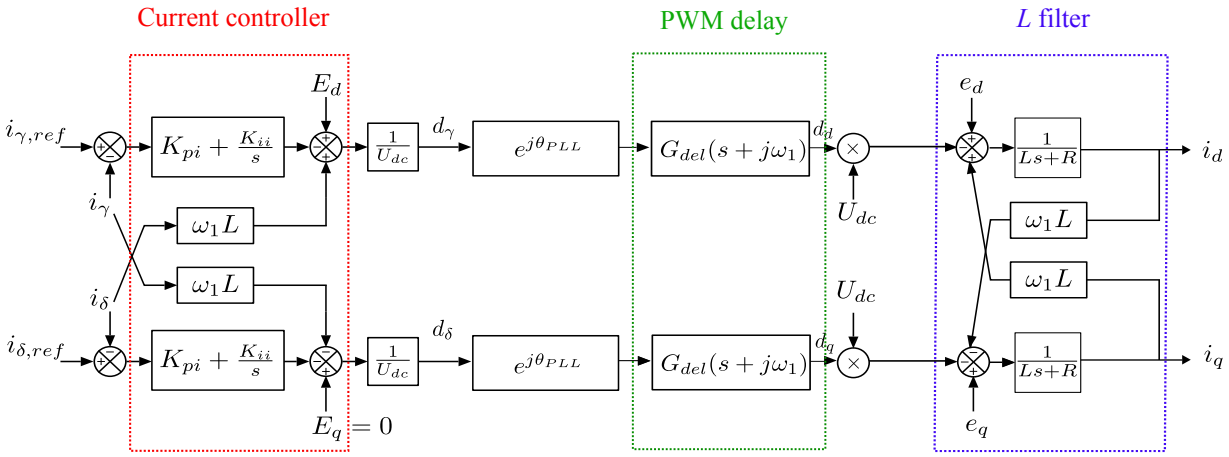


Figure 2.4: Overall control block diagram of the inner-loop current control.

where $d_{\gamma,\delta}$ are the duty cycles in the converter frame, $d_{d,q}$ are the duty cycles in the grid frame. $e_{d,q}$ are the actual grid voltage in dq-frame.

2.2.3 *Outer Voltage Control Loop*

As previous stated, the outer voltage control loop aims at stabilizing the DC-link voltage U_{dc}. Additionally, the output of the outer voltage control loop is the reference current in d-axis for the inner current control loop $i_{\gamma,ref}$. The same as the inner current control loop, PI controller is applied in the outer voltage control loop, as shown in Fig.2.5.

It is worth noting that the bandwidth of the outer voltage control loop is much lower than that of the inner voltage control loop (around 1/50 and 1/10) [11]. It is because the inner loop and the outer loop should be separated so that their controllers can be designed independently. With the high response speed of the inner current control loop, the inner loop can be considered as a constant for the outer loop.

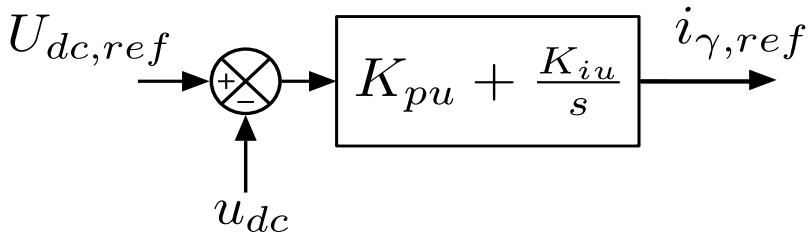


Figure 2.5: Block diagram of the outer voltage control loop.

2.3 MODELING OF THE INPUT IMPEDANCE OF THE AC/DC CONVERTER

One effective technique for the harmonic current emission analysis is the impedance based method [13, 14, 15, 16]. The impedance model of the charger-grid system is depicted in Fig. 2.6. The DCFC is modelled as a harmonic current source I_c in parallel with the converter input impedance Z_c. The grid is simplified as a background voltage source V_g in series with the grid impedance Z_g.

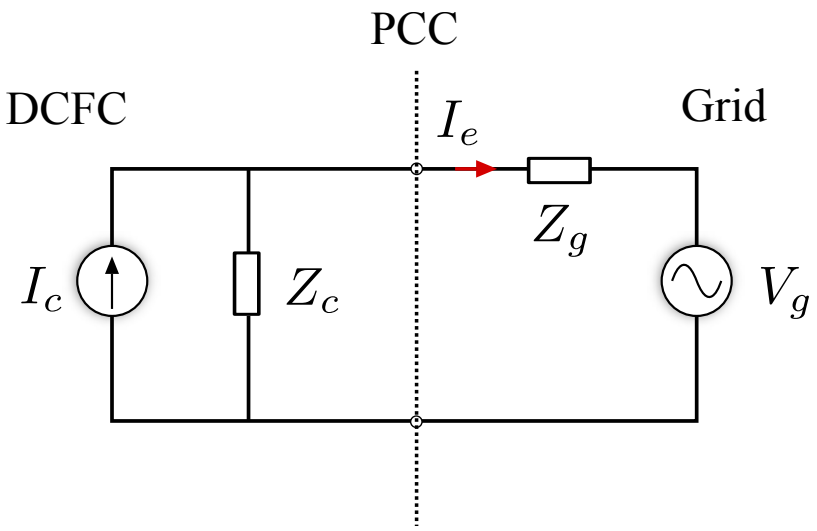


Figure 2.6: Impedance model of a charger-grid system.

Based on the the model shown in Fig. 2.6, the harmonic current emission I_e in different grid conditions can be expressed as (2.6). To obtain I_e, the input impedance of converter Z_c need to be modeled first. In this thesis, the analytical model of the input impedance Z_c for the AFE shown in Fig. 2.2 is studied. The influences of the phase-locked loop (PLL) dynamics, the inner current

control loop, the outer voltage control loop, and the fluctuation of DC-link voltage on the input impedance modeling are taken into account.

$$I_e(s) = \frac{Z_c(s)I_c(s)}{Z_c(s)+Z_g(s)} - \frac{V_g(s)}{Z_c(s)+Z_g(s)} \tag{2.6}$$

Since the SRF-PLL, the current controller and the voltage controller are realized with the PI controller in dq-frame, Clarke (abc to $\alpha\beta$) and Park transformation ($\alpha\beta$ to dq) are applied, so that (2.1) can be transformed to the model under dq-frame,

$$\begin{cases} e_\alpha = Lsi_\alpha + Ri_\alpha + U_{dc}d_\alpha \\ e_\beta = Lsi_\beta + Ri_\beta + U_{dc}d_\beta \\ C_{out}sU_{dc} = \frac{3}{2}(d_\alpha i_\alpha + d_\beta i_\beta) - i_L \end{cases} \tag{2.7}$$

$$\begin{cases} e_d = Lsi_d + Ri_d - L\omega_1 i_q + U_{dc}d_d \\ e_q = Lsi_q + Ri_q + L\omega_1 i_d + U_{dc}d_q \\ C_{out}sU_{dc} = \frac{3}{2}(d_d i_d + d_q i_q) - i_L \end{cases} \tag{2.8}$$

Based on (2.8) and the aforementioned analysis in section 2.2, the overall block diagram of the AFE control system is illustrated in Fig. 2.7. The detailed description and the mathematical expression of the symbols are listed in Table 2.1 where bold letters denotes matrix or vector.

$\boldsymbol{G_{del}}(s+j\omega_1)$ and $\boldsymbol{Y}(s+j\omega_1)$ are the transformation of $\boldsymbol{G_{del}}(s)$ and $\boldsymbol{Y}(s)$ respectively, which are the transfer functions in $\alpha\beta$-frame. The detailed deviations are shown in (2.9) and (2.10),

$$\begin{aligned} \boldsymbol{G_{del}}(s) &= \begin{bmatrix} e^{-T_{del}s} & 0 \\ 0 & e^{-T_{del}s} \end{bmatrix} \\ \boldsymbol{G_{del}}(s+j\omega_1) &= \boldsymbol{e}^{-j\omega_1 t}\boldsymbol{G_{del}}(s)\boldsymbol{e}^{j\omega_1 t} \end{aligned} \tag{2.9}$$

$$\begin{aligned} \boldsymbol{Y}(s) &= \begin{bmatrix} Ls+R & 0 \\ 0 & Ls+R \end{bmatrix}^{-1} \\ \boldsymbol{Y}(s+j\omega_1) &= \boldsymbol{e}^{-j\omega_1 t}\boldsymbol{Y}(s)\boldsymbol{e}^{j\omega_1 t} \end{aligned} \tag{2.10}$$

where ω_1 is the grid fundamental angular frequency, $\boldsymbol{e}^{-j\omega_1 t}$ is the matrix equivalent form of the Park transformation, $\boldsymbol{e}^{j\omega_1 t}$ is the matrix equivalent form of the inverse Park transformation[17],

$$\boldsymbol{e}^{-j\omega_1 t} = \begin{bmatrix} \cos\omega_1 t & \sin\omega_1 t \\ -\sin\omega_1 t & \cos\omega_1 t \end{bmatrix} \tag{2.11}$$

$$\boldsymbol{e}^{j\omega_1 t} = \begin{bmatrix} \cos\omega_1 t & -\sin\omega_1 t \\ \sin\omega_1 t & \cos\omega_1 t \end{bmatrix} \tag{2.12}$$

Table 2.1: Description and math expression of the symbols in Fig. 2.7.

Symbol	Description	Math expression
$V_{\text{dc,ref}}$	DC-link voltage reference	$V_{\text{dc,ref}}$
V_{dc}	DC-link voltage	V_{dc}
C_{out}	output capacitance of the AC/DC capacitance	C_{out}
$\boldsymbol{i_{\gamma\delta,ref}}$	input current $i_{\text{a,b,c}}$ reference in dq-frame in the converter frame	$\begin{bmatrix} i_{\gamma ref} & i_{\delta ref} \end{bmatrix}^T$
$\boldsymbol{i_{\gamma\delta}}$	input current $i_{\text{a,b,c}}$ in dq-frame in the converter frame	$\begin{bmatrix} i_\gamma & i_\delta \end{bmatrix}^T$
$\boldsymbol{i}_{\text{dq}}$	input current in dq-frame in the grid frame	$\begin{bmatrix} i_{\text{d}} & i_{\text{q}} \end{bmatrix}^T$
$\boldsymbol{d_{\gamma\delta}}$	duty cycle in dq-frame in the converter frame	$\begin{bmatrix} d_\gamma & d_\delta \end{bmatrix}^T$
$\boldsymbol{d}_{\text{dq}}$	duty cycle in dq-frame in the grid frame	$\begin{bmatrix} d_{\text{d}} & d_{\text{q}} \end{bmatrix}^T$
$\boldsymbol{e}_{\text{dq}}$	PCC voltage in dq-frame in the grid frame	$\begin{bmatrix} e_{\text{d}} & e_{\text{q}} \end{bmatrix}^T$
$\boldsymbol{J\omega_1 L}$	current decoupling component of the current controller	$\begin{bmatrix} 0 & -\omega_1 L \\ \omega_1 L & 0 \end{bmatrix}^{-1}$
$\boldsymbol{G_v(s)}$	transfer function of the voltage controller in dq-frame	$\begin{bmatrix} k_{\text{pu}} + \frac{k_{\text{iu}}}{s} & 0 \end{bmatrix}^T$
$\boldsymbol{G_i(s)}$	transfer function of the current controller in dq-frame	$\begin{bmatrix} k_{\text{pi}} + \frac{k_{\text{ii}}}{s} & 0 \\ 0 & k_{\text{pi}} + \frac{k_{\text{ii}}}{s} \end{bmatrix}$
$\boldsymbol{G_{del}}(s + j\omega_1)$	transfer function of the control delay in dq-frame	$\begin{bmatrix} \cos(\omega_1 T_{del}) e^{-T_{del}s} & \sin(\omega_1 T_{del}) e^{-T_{del}s} \\ -\sin(\omega_1 T_{del}) e^{-T_{del}s} & \cos(\omega_1 T_{del}) e^{-T_{del}s} \end{bmatrix}$
$\boldsymbol{Y}(s + j\omega_1)$	transfer function of power filter in dq-frame	$\begin{bmatrix} Ls + R & -\omega_1 L \\ \omega_1 L & Ls + R \end{bmatrix}^{-1}$

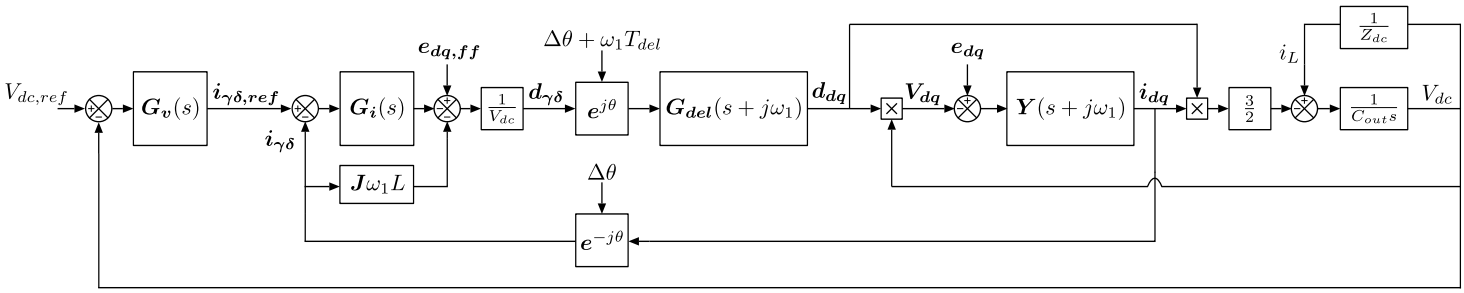


Figure 2.7: Block diagram of the control system shown in Fig. 2.2.

As previously stated, the impedance model is established as a small signal model which means that the DCFC works at the steady state with small perturbation. It is necessary to convert the control diagram in Fig. 2.7 into the linearized small signal model.

First, the expression of the detected phase variation $\Delta\theta$ provided by the SRF-PLL is derived [16]. Fig. 2.3 depicts the SRF-PLL block of the AFE control system. This SRF-PLL block diagram is in dq-frame where the q-axis voltage e_δ is used for phase tracking [18].

To establish the small-signal model of SRF-PLL, a small perturbation $\tilde{e}_{dq}$ is applied on the point of common coupling (PCC) voltage. The corresponding PCC voltage in the $\alpha\beta$-frame can be obtained using (2.12),

$$e_{\alpha\beta} = (e_{dq} + \tilde{e}_{dq})e^{j\omega_1 t} \tag{2.13}$$

where $e_{dq} = e_{1d} + j0$ is the corresponding complex space vector of the PCC voltage in steady state. The output of the SRF-PLL that corresponds to the aforementioned voltage perturbation is θ_{PLL}, which can be written as,

$$\theta_{PLL} = \omega_1 t + \Delta\theta \tag{2.14}$$

According to (2.13) and (2.14), the PCC voltage in the dq-frame e_{dq} can be given as,

$$\begin{aligned}
e_{dq} &= e_{\alpha\beta}e^{-j\theta_{PLL}} \\
&= e_{\alpha\beta}e^{-j\omega_1 t}e^{-j\Delta\theta} \\
&= (e_{dq} + \tilde{e}_{dq})e^{-j\Delta\theta} \\
&\approx (e_{dq} + \tilde{e}_{dq})(1 - j\Delta\theta) \\
&= (e_{1d} + \tilde{e}_d + j\tilde{e}_q)((1 - j\Delta\theta)) \\
&= (e_{1d} + \tilde{e}_d + \tilde{e}_q\Delta\theta) + j(\tilde{e}_q - \Delta\theta e_{1d} + \Delta\theta\tilde{e}_d) \\
&\approx (e_{1d} + \tilde{e}_d) + j(\tilde{e}_q - \Delta\theta e_{1d}) && (2.15) \\
&= e_d + je_q && (2.16)
\end{aligned}$$

The simplification in (2.21) neglects the product of two small signal. Therefore, the q-axis voltage is,

$$e_q = \tilde{e}_q - \Delta\theta e_{1d} \tag{2.17}$$

Since $\Delta\theta$ is regulated by the PI controller of the SRF-PLL, it can be expressed as,

$$\Delta\theta = \frac{K_{pPLL} + \frac{K_{iPLL}}{s}}{s} e_q \tag{2.18}$$

Replacing e_q with (2.17) in (2.18), the small-signal transfer function of SRF-PLL can be expressed as,

$$T_{PLL}(s) = \frac{K_{pPLL} + \frac{K_{iPLL}}{s}}{s + (K_{pPLL} + \frac{K_{iPLL}}{s}) e_{1d}} \tag{2.19}$$

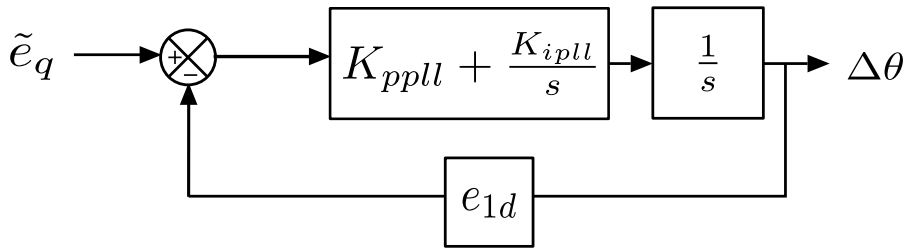


Figure 2.8: Block diagram of the SRF-PLL small-signal model.

Besides the SRF-PLL, the transform matrix $\boldsymbol{e}^{-j\Delta\theta}$ of the small-signal model can also be linearized as (2.20),

$$\begin{aligned} \boldsymbol{e}^{-j\Delta\theta} &= \begin{bmatrix} \cos(\Delta\theta) & \sin(\Delta\theta) \\ \sin(-\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \\ &\approx \begin{bmatrix} 1 & \Delta\theta \\ -\Delta\theta & 1 \end{bmatrix} = \boldsymbol{I} - \boldsymbol{J}\Delta\theta \end{aligned} \tag{2.20}$$

where $\boldsymbol{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\boldsymbol{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$.

Next is to linearize the product of $\boldsymbol{e}^{j\Delta\theta}$ and $\boldsymbol{d}_{\gamma\delta}$,

$$\begin{aligned} \boldsymbol{d}_{\gamma\delta}\boldsymbol{e}^{j\Delta\theta} &= (\boldsymbol{I} + \boldsymbol{J}\Delta\theta)\left(\begin{bmatrix} D_{cd} \\ D_{cq} \end{bmatrix} + \begin{bmatrix} \tilde{d_{cd}} \\ \tilde{d_{cq}} \end{bmatrix} \right) \\ &\approx \begin{bmatrix} D_{cd} \\ D_{cq} \end{bmatrix} + \boldsymbol{J} \begin{bmatrix} D_{cd} \\ D_{cq} \end{bmatrix} \Delta\theta + \begin{bmatrix} \tilde{d_{cd}} \\ \tilde{d_{cq}} \end{bmatrix} \end{aligned} \tag{2.21}$$

where $\begin{bmatrix} D_{cd} \\ D_{cq} \end{bmatrix}$ denotes the duty cycle in steady state, $\begin{bmatrix} \tilde{d_{cd}} \\ \tilde{d_{cq}} \end{bmatrix}$ represents the small signal duty cycle. (2.21) neglects the product of two small signal, i.e., $\boldsymbol{J} \begin{bmatrix} \tilde{d_{cd}} \\ \tilde{d_{cq}} \end{bmatrix} \Delta\theta$. Therefore, the small signal component of $\boldsymbol{d}_{\gamma\delta} \boldsymbol{e}^{j\Delta\theta}$ is given as (2.21) minus the duty cycle of steady state,

$$\tilde{\boldsymbol{d}_{\gamma\delta}} \boldsymbol{e}^{j\Delta\theta} = \boldsymbol{J} \begin{bmatrix} D_{cd} \\ D_{cq} \end{bmatrix} \Delta\theta + \begin{bmatrix} \tilde{d_{cd}} \\ \tilde{d_{cq}} \end{bmatrix} \tag{2.22}$$

Based on the previous analysis, the block diagram of the linearized small signal model can be obtained. as shown in Fig. 2.9. In this linearized small signal control diagram, symbols using capital represents the steady state value, whereas the symbols with tilde denote the small signal value.

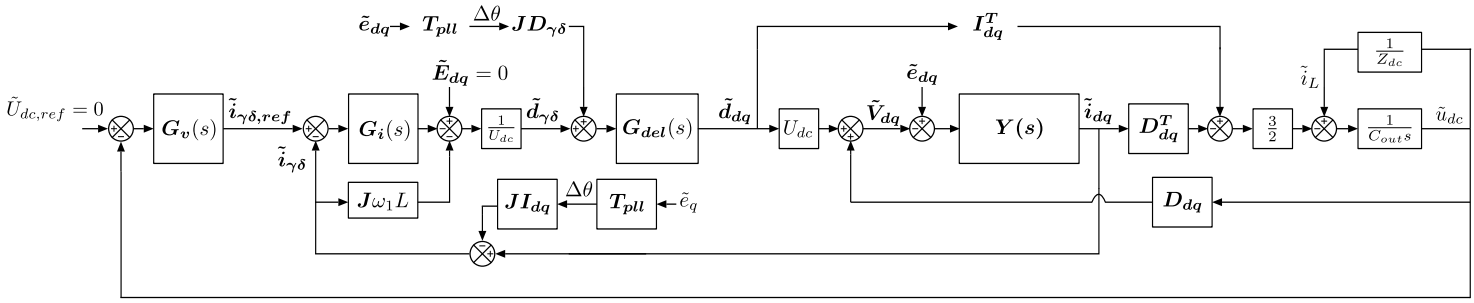


Figure 2.9: Block diagram of the linearized small signal model of the system shown in Fig. 2.2.

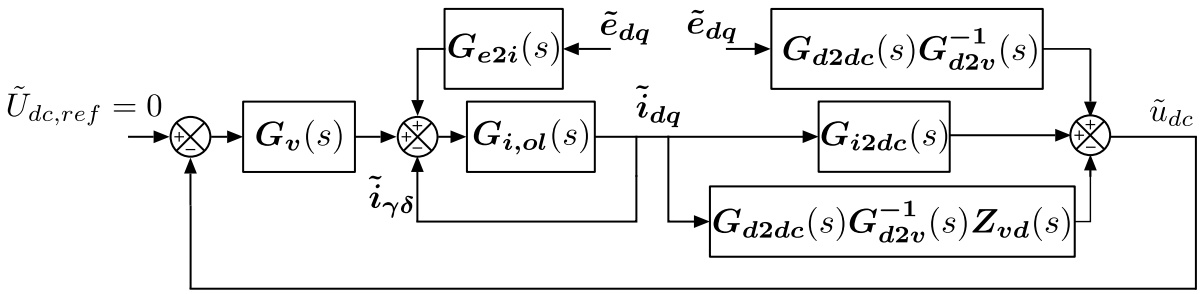


Figure 2.10: Simplified block diagram of the linearized small signal model of AFE.

To get the input impedance of AFE, the linearized small signal model in Fig. 2.9 is simplified, as shown in Fig. 2.10. Detailed expressions of the simplified transfer functions are expressed as follows,

$$
\begin{aligned}
\boldsymbol{G_{d2dc}}(s) &= \frac{3}{2}\frac{1}{C_{out}s + \frac{1}{Z_{dc}}}\boldsymbol{I_{dq}^T} \\
\boldsymbol{G_{i2dc}}(s) &= \frac{3}{2}\frac{1}{C_{out}s + \frac{1}{Z_{dc}}}\boldsymbol{D_{dq}^T} \\
\boldsymbol{G_{d2v}}(s) &= U_{dc} + \frac{3}{2}\frac{1}{C_{out}s + \frac{1}{Z_{dc}}}\boldsymbol{D_{dq}I_{dq}^T} \\
\boldsymbol{Z_{vd}}(s) &= \boldsymbol{Y^{-1}} + \frac{3}{2}\frac{1}{C_{out}s + \frac{1}{Z_{dc}}}\boldsymbol{D_{dq}D_{dq}^T} \\
\boldsymbol{H(s)} &= \frac{1}{U_{dc}}\boldsymbol{Z_{vd}^{-1}G_{d2v}G_{del}} \\
\boldsymbol{H_{de}}(s) &= (\boldsymbol{I} - \boldsymbol{HJ}\omega_1 L)^{-1}\boldsymbol{H} \\
\boldsymbol{G_{i,ol}}(s) &= \boldsymbol{H_{de}G_i} \\
\boldsymbol{T_{PLL}^M} &= [0 \quad T_{PLL}] \\
\boldsymbol{Y_{PLL}} &= \boldsymbol{JI_{dq}T_{PLL}^M} \\
\boldsymbol{H_{PLL}} &= \boldsymbol{JD_{\gamma\delta}T_{PLL}^M} \\
\boldsymbol{G_{e2i}}(s) &= (U_{dc}\boldsymbol{G_{del}^{-1}G_{d2v}^{-1}} + (\boldsymbol{G_i} - \boldsymbol{J}\omega_1 L)\boldsymbol{Y_{PLL}} - U_{dc}\boldsymbol{H_{PLL}})\boldsymbol{G_i^{-1}}
\end{aligned}
\tag{2.23}
$$

Since the desired result is the input impedance of AFE, the input of the control diagram needs to be shifted to the small signal grid voltage $\tilde{\boldsymbol{e_{dq}}}$, and the output is the small signal AC phase current $\tilde{\boldsymbol{i_{dq}}}$. The final transformed simplified block diagram of the linearized small signal model of AFE is illustrated in Fig. 2.11. The input impedance of the AFE $\boldsymbol{Z_{AFE}}(s)$ in DCFC system can be derived as,

$$
\begin{aligned}
\boldsymbol{Y_{AFE}}(s) &= \frac{(\boldsymbol{G_{e2i}} - \boldsymbol{G_v G_{d2dc} G_{d2v}^{-1}})(\boldsymbol{I} + \boldsymbol{G_{i,ol}})^{-1}\boldsymbol{G_{i,ol}}}{\boldsymbol{I} - \boldsymbol{G_v G_{i2dc,tot}}(\boldsymbol{I} + \boldsymbol{G_{i,ol}})^{-1}\boldsymbol{G_{i,ol}}} \\
\boldsymbol{Z_{AFE}}(s) &= \boldsymbol{Y_{AFE}^{-1}}(s)
\end{aligned}
\tag{2.24}
$$

where $\boldsymbol{G_{i2dctot}}(s) = \boldsymbol{G_{i2dc}} - \boldsymbol{G_{d2dc}G_{d2v}^{-1}Z_{vd}}$.

Yet, aforementioned derivation of $\boldsymbol{Z_{AFE}}(s)$ is established in the rotating dq-frame without physical meaning. It fails to estimate the actual harmonic current emission of DCFC, which is

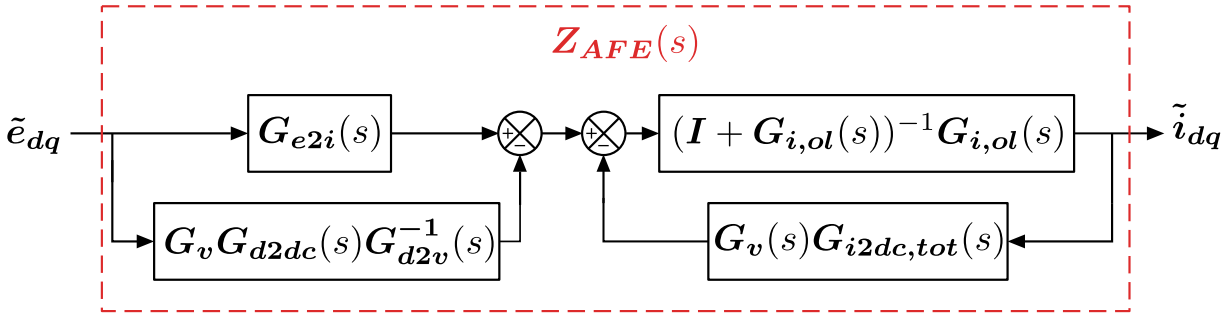


Figure 2.11: Transformed simplified block diagram of the linearized small signal model of AFE.

induced by the grid harmonic voltage. Hence, $\boldsymbol{Z_{AFE}}(s)$ should be transformed to $\alpha\beta$-frame. First, $\boldsymbol{Z_{AFE}}(s)$ in the dq-frame is expressed as,

$$\boldsymbol{e_{dq}}(s) = \boldsymbol{Z_{AFE}}(s)\boldsymbol{i_{dq}}(s) \leftrightarrow \begin{bmatrix} e_d \\ e_q \end{bmatrix} = \begin{bmatrix} Z_{dd}(s) & Z_{dq}(s) \\ Z_{qd}(s) & Z_{qq}(s) \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} \tag{2.25}$$

The complex equivalent of (2.25) involving two complex transfer functions can be derived as [19],

$$e_{dq} = Z_{+,dq}(s) i_{dq} + Z_{-,dq}(s) i_{dq}^* \tag{2.26}$$

where $Z_{+,dq}(s)$ and $Z_{-,dq}(s)$ are the complex equivalent of $\boldsymbol{Z_{AFE}}(s)$, i_{dq}^* is the complex conjugate of i_{dq}.

The expressions of $Z_{+,dq}(s)$ and $Z_{-,dq}(s)$ are expressed as (2.27).

$$\begin{aligned} Z_{+,dq}(s) &= \frac{Z_{dd}(s) + Z_{qq}(s)}{2} + j\frac{Z_{qd}(s) - Z_{dq}(s)}{2} \\ Z_{-,dq}(s) &= \frac{Z_{dd}(s) - Z_{qq}(s)}{2} + j\frac{Z_{qd}(s) + Z_{dq}(s)}{2} \end{aligned} \tag{2.27}$$

It should be noted that (2.26) only models the system in the positive sequence. To obtain the impedance model in the stationary $\alpha\beta$-frame, DCFC system in the negative domain cannot be neglected, i.e., sequence-domain impedance model [20]. The double frequency model is given by,

$$\begin{bmatrix} e_{dq} \\ e_{dq}^* \end{bmatrix} = \begin{bmatrix} Z_{+,dq}(s) & Z_{-,dq}(s) \\ Z_{-,dq}^*(s) & Z_{+,dq}^*(s) \end{bmatrix} \begin{bmatrix} i_{dq} \\ i_{dq}^* \end{bmatrix} \tag{2.28}$$

where e^*_{dq} is the PCC voltage in the negative sequence domain, $Z^*_{+,dq}(s)$ and $Z^*_{-,dq}(s)$ are the complex conjugates of $Z_{+,dq}(s)$ and $Z_{-,dq}(s)$ respectively.

Using (2.11) and (2.28), the sequence-domain impedance model can be obtained as follows.

$$\begin{aligned} e^{-j\omega_1 t}e_{\alpha\beta}(s) &= Z_{+,dq}(s)e^{-j\omega_1 t}i_{\alpha\beta} + Z_{-,dq}(s)e^{j\omega_1 t}i^*_{\alpha\beta} \\ \rightarrow e_{\alpha\beta}(s) &= e^{j\omega_1 t}Z_{+,dq}(s)e^{-j\omega_1 t}i_{\alpha\beta} + e^{j\omega_1 t}Z_{-,dq}(s)e^{j\omega_1 t}i^*_{\alpha\beta} \\ &= Z_{+,dq}(s-j\omega_1)i_{\alpha\beta} + Z_{-,dq}(s-j\omega_1)e^{2j\omega_1 t}i^*_{\alpha\beta} \end{aligned} \tag{2.29}$$

$$\begin{aligned} e^{j\omega_1 t}e^*_{\alpha\beta}(s) &= Z^*_{-,dq}(s)e^{-j\omega_1 t}i_{\alpha\beta} + Z^*_{+,dq}(s)e^{j\omega_1 t}i^*_{\alpha\beta} \\ \rightarrow e^{2j\omega_1 t}e^*_{\alpha\beta}(s) &= e^{j\omega_1 t}Z^*_{-,dq}(s)e^{-j\omega_1 t}i_{\alpha\beta} + e^{j\omega_1 t}Z^*_{+,dq}(s)e^{j\omega_1 t}i^*_{\alpha\beta} \\ &= Z^*_{-,dq}(s-j\omega_1)i_{\alpha\beta} + Z^*_{+,dq}(s-j\omega_1)e^{2j\omega_1 t}i^*_{\alpha\beta} \end{aligned} \tag{2.30}$$

Combining (2.29) and (2.30), the sequence-domain impedance is expressed as,

$$\begin{bmatrix} e_{\alpha\beta} \\ e^{2j\omega_1 t}e^*_{\alpha\beta} \end{bmatrix} = \underbrace{\begin{bmatrix} Z_{+,dq}(s-j\omega_1) & Z_{-,dq}(s-j\omega_1) \\ Z^*_{-,dq}(s-j\omega_1) & Z^*_{+,dq}(s-j\omega_1) \end{bmatrix}}_{\boldsymbol{Z_{AFE,\alpha\beta}}(s)} \begin{bmatrix} i_{\alpha\beta} \\ e^{2j\omega_1 t}i^*_{\alpha\beta} \end{bmatrix} \tag{2.31}$$

where $\boldsymbol{Z_{AFE,\alpha\beta}}(s)$ is the input impedance of the DCFC system in the sequence domain, $i^*_{\alpha\beta}$ is the complex conjugate of $i_{\alpha\beta}$ in $\alpha\beta$-frame. Observing from (2.31), it can be discovered that a given PCC voltage harmonic with frequency f_h can yield current harmonics with two frequencies, one is f_h, the other is the frequency coupling component $2f_1 - f_h$.

2.4 SUMMARY

In chapter 2, the architecture of the DCFC system is introduced first. The DCFC with AC distribution network is chosen for investigation in this thesis. Between the AC/DC stage and DC/DC stage of the DCFC, larger DC-link capacitors are employed which serves as a buffer. Hence, only harmonics induced by the AC/DC stage are considered. After that, the modeling of the input impedance of the AC/DC converter is elaborated in detail: first, the design of the AC/DC converter is decided, which is the two-level AFE with PI controller and L filter. Then the control diagram of AFE is transformed to the linearized small signal model since the impedance model is established on DCFC's steady state with small perturbation. Finally, the linearized small signal model is simplified to get the input impedance. The input of the final control diagram is the PCC voltage perturbation and the output is the corresponding AC phase current.

3

HARMONIC EMISSION MODELING OF DC FAST CHARGERS

Efforts have been devoted to the calculation of harmonic components around the carrier and carrier-sideband frequency based on the double Fourier series method. However, low frequencies below 5 kHz are the most damaging frequencies to power devices and grid [21]. Till now, there is no in-depth study focusing on these low-frequency harmonics. In this chapter, harmonic current source modeling within the low-frequency range for the DCFC is introduced.

3.1 BRIEF INTRODUCTION OF THE DOUBLE FOURIER SERIES METHOD FOR HARMONIC ANALYSIS

To understand the general idea behind the harmonic analysis, previous work on DC-link current harmonics of the AFE is studied first [5]. In this analysis, the double Fourier series method is applied. First, AC phase currents $i_{a,b,c}$ in Fig. 2.2 are assumed to be ideal sinusoidal waveforms, which are expressed as follows,

$$\begin{cases} i_a = I_m \cos[\omega_1(t-t_0)+\phi] \\ i_b = I_m \cos[\omega_1(t-t_0)-\frac{2}{3\pi}+\phi] \\ i_c = I_m \cos[\omega_1(t-t_0)+\frac{2}{3\pi}+\phi] \end{cases} \tag{3.1}$$

where I_m is the amplitude of the AC phase currents, ω_1 is fundamental angular frequency, t_0 is the initial time, and ϕ is the power factor angle of the grid side. References of the modulation (i.e. Ref_a, Ref_b, and Ref_c) are given as,

$$\begin{cases} Ref_a = M\cos[\omega_1(t-t_0)]+M_3 \\ Ref_b = M\cos[\omega_1(t-t_0)-\frac{2}{3\pi}]+M_3 \\ Ref_c = M\cos[\omega_1(t-t_0)+\frac{2}{3\pi}]+M_3 \end{cases} \tag{3.2}$$

where M is the modulation index, M_3 is the third-harmonic injection. Based on the cross-points of the modulation waveforms and the triangular carrier waveforms, the load current i_{L} can be divided into six segments within one fundamental period. In each segment, i_{L} comprises two AC phase currents, one is the corresponding phase current of the top modulation waveform, the other is the negative corresponding phase current of the bottom modulation waveform. For illustration, i_{L} equals i_{a} plus $-i_{\mathrm{b}}$ when Ref_{a} is the top reference and Ref_{b} is the bottom reference.

To obtain current harmonics of the DC load current i_{L}, double Fourier series method is applied [22],

$$\begin{aligned} i_L =& I_0 + \sum_{n=1}^{\infty} I_n \cos\left[n\omega_1(t-t_0)+\theta_n\right] \\ &+ \sum_{m=1}^{\infty}\sum_{n=-\infty}^{\infty} I_{mn}\cos\left(\left[m\omega_{\mathrm{s}}+n\omega_1\right)(t-t_0)+\theta_{mn}\right] \end{aligned} \tag{3.3}$$

where I_0 is the DC component, I_{n} is the amplitude of the fundamental and baseband harmonics (i.e. n_{th} harmonics), θ_{n} is the initial phase of the n_{th} harmonics, I_{mn} and θ_{mn} are the amplitude and the initial phase of the carrier and carrier-sideband harmonics respectively, i.e., $(m+n\cdot\frac{\omega_c}{\omega_1})_{\mathrm{th}}$ harmonics, ω_c is the angular frequency of the triangular carrier.

Based on (3.3), the carrier and carrier-sideband frequency harmonics can be calculated as,

$$\begin{aligned} \frac{1}{2}I_{mn}^2 &= \frac{3\omega_1}{2\pi}\int_{t_0}^{t_0+\frac{2\pi}{3\omega_1}} i_L i_{mn}dt \\ &= \frac{3\omega_1}{2\pi}\sum_{j=0}^{1}\sum_{k=0}^{N-1}[(\int_{t_{01}+kT_s+jNT_s+t_0}^{t_{02}+kT_s+jNT_s+t_0} + \int_{t_{05}+kT_s+jNT_s+t_0}^{t_{06}+kT_s+jNT_s+t_0})i_1\times i_{mn}dt \\ &+ (\int_{t_{02}+kT_s+jNT_s+t_0}^{t_{03}+kT_s+jNT_s+t_0} + \int_{t_{04}+kT_s+jNT_s+t_0}^{t_{05}+kT_s+jNT_s+t_0})i_2\times i_{mn}dt] \end{aligned} \tag{3.4}$$

where the fundamental period of the DC load current i_{L} is $\frac{3\omega_1}{2\pi}$, so that there are two segments within one fundamental period of i_{L}. $i_{mn}=I_{mn}\cos[(m\omega_c+n\omega_1)(t-t_0)+\theta_{mn}]$, k means that (3.4) is in $(k+1)_{\mathrm{th}}$ switching period, $N=\frac{\omega_c}{6\omega_1}$, i_1 and i_2 are the amplitudes of the DC load current during different time periods, $t_{o1}-t_{o6}$ are the switching time instants, j represents two segments respectively.

As illustrated in Fig. 3.1,

$$j=0,\quad \begin{cases} i_1 &= i_a \\ i_2 &= i_a+i_b=-i_c \end{cases} \tag{3.5}$$

$$j=1,\quad \begin{cases} i_1 &= i_b \\ i_2 &= i_a + i_b = -i_c \end{cases} \tag{3.6}$$

Since the switching frequency is much higher than the fundamental frequency, i_1 and i_2 are assumed to be constant during one switching period. Therefore, (3.7) and (3.8) can be obtained by substituting (3.1) to (3.5) and (3.6).

$$j=0,\quad \begin{cases} i_1 &= I_m \cos(\omega_1 k T_{SW} + \phi) \\ i_2 &= -I_m \cos(\omega_1 k T_{SW} + \frac{2\pi}{3} + \phi) \end{cases} \tag{3.7}$$

$$j=1,\quad \begin{cases} i_1 &= I_m \cos(\omega_1 k T_{SW} - \frac{2\pi}{3} + \frac{\pi}{3} + \phi) \\ i_2 &= -I_m \cos(\omega_1 k T_{SW} + \frac{2\pi}{3} + \frac{\pi}{3} + \phi) \end{cases} \tag{3.8}$$

To get $t_{o1} - t_{o6}$, pulse width T_{P1} and T_{P2} need to be solved first, which can be derived as,

$$j=0,\quad \begin{cases} T_{P1} &= \frac{Ref_a - Ref_b}{2}\frac{T_s}{2} \\ T_{P2} &= \frac{Ref_b - Ref_c}{2}\frac{T_s}{2} \end{cases} \tag{3.9}$$

$$j=1,\quad \begin{cases} T_{P1} &= \frac{Ref_b - Ref_a}{2}\frac{T_s}{2} \\ T_{P2} &= \frac{Ref_a - Ref_c}{2}\frac{T_s}{2} \end{cases} \tag{3.10}$$

where the first 2 represents the peak-to-peak value of the carrier. It should be noted that there is a $\frac{\pi}{3}$ phase shift of the three-phase modulation waveforms between $j=0$ and $j=1$, e.g., $Ref_a = M\cos[\omega_1(kT_{SW} - t_0)] + M_3$ when $j=0$, $Ref_a = M\cos[\omega_1(kT_{SW} - t_0) + \frac{\pi}{3}] + M_3$ when $j=1$. As illustrated in Fig. 3.1, the relationship between the pulse width and switching instants is expressed as,

$$\begin{cases} T_{00} &= T_{SW} - 2(T_{P1} + T_{P2}) \\ t_{01} &= \frac{T_{00}}{4} \\ t_{02} &= t_{01} + T_{P1} \\ t_{03} &= t_{02} + T_{P2} \\ t_{04} &= t_{03} + \frac{T_{00}}{2} \\ t_{05} &= t_{04} + T_{P2} \\ t_{06} &= t_{05} + T_{P1} \end{cases} \tag{3.11}$$

Given (3.9), (3.10) and (3.11), detailed expressions of t_{01} and t_{02} can be derived as,

$$j=0,\begin{cases} t_{01} &= \frac{1}{4}T_{SW}-\frac{\sqrt{3}M}{8}\cos(k\omega_1 T_{SW}-\frac{\pi}{6})T_{SW} \\ t_{02} &= \frac{1}{4}T_{SW}+\frac{3M}{8}\cos(k\omega_1 T_{SW}+\frac{\pi}{3})T_{SW} \end{cases} \tag{3.12}$$

$$j=1,\begin{cases} t'_{01} &= \frac{1}{4}T_{SW}-\frac{\sqrt{3}M}{8}\cos(k\omega_1 T_{SW}-\frac{\pi}{6})T_{SW} \\ t'_{02} &= \frac{1}{4}T_{SW}+\frac{3M}{8}\cos(k\omega_1 T_{SW}+\frac{\pi}{3})T_{SW} \end{cases} \tag{3.13}$$

Substituting (3.12), (3.13), (3.7), and (3.8) into (3.4), I_{mn} can be derived as,

$$I_{mn}=\frac{\omega_1}{\pi}\frac{3}{m\omega_c+n\omega_1}\sum_{k=0}^{N-1} \tag{3.14}$$

$$[I_m\cos(k\omega_1 T_{SW}+\phi)\times A(t)(\,|_{t_{01}}^{t_{02}}+|_{t_{05}}^{t_{06}}) \tag{3.15}$$

$$-I_m\cos(k\omega_1 T_{SW}+\frac{2\pi}{3}+\phi)\times A(t)(\,|_{t_{02}}^{t_{03}}+|_{t_{04}}^{t_{05}}) \tag{3.16}$$

$$+I_m\cos(k\omega_1 T_{SW}-\frac{\pi}{3}+\phi)\times A(t)(\,|_{t'_{01}}^{t'_{02}}+|_{t'_{05}}^{t'_{06}}) \tag{3.17}$$

$$-I_m\cos(k\omega_1 T_{SW}+\pi+\phi)\times A(t)(\,|_{t'_{02}}^{t'_{03}}+|_{t'_{04}}^{t'_{05}})] \tag{3.18}$$

where $A(t)=\sin[(m\omega_c+n\omega_1)t+n\omega_1(k+jN)T_{SW}+\theta_{mn}]$. After the approximation and simplification [5], I_{mn} can be gained as,

if m is odd, n=3z, and z is odd,

$$I_{mn}=\frac{\omega_0}{\pi}\frac{12}{m\omega_c+n\omega_0}\sum_{k=0}^{N-1}[I_m\cos(k\omega_1 T_s+\frac{\pi}{3}+\phi)\times$$
$$(\sin(m\omega_c t_{02})-\sin(m\omega_c t_{01}))\cos(n\omega_1 kT_{SW}+\theta_{mn})] \tag{3.19}$$

if m is even, n=3z, and z is even

$$I_{mn}=\frac{\omega_0}{\pi}\frac{12}{m\omega_c+n\omega_0}\sum_{k=0}^{N-1}[I_m(\cos(k\omega_1 T_s+\frac{\pi}{3}+\phi)\times\sin(m\omega_c t_{02})$$
$$-\sqrt{3}I_m\cos(k\omega_1 T_s-\frac{\pi}{6}+\phi)\sin(m\omega_c t_{01}))\cos(n\omega_1 kT_{SW}+\theta_{mn})] \tag{3.20}$$

In other cases, $I_{\mathrm{mn}} = 0$. Then, the magnitude of the DC-link current harmonics of carrier and carrier-sideband can be expressed as,

$$|I_{mn}| = \sqrt{I_{mn}^2(\theta_{mn} = 0) + I_{mn}^2(\theta_{mn} = \frac{\pi}{2})} \tag{3.21}$$

The aforementioned analysis yields detailed calculations of harmonics in DC load current i_{L} with the frequency of carrier and carrier-sideband based on the assumption that AC phase currents are ideal sinusoidal waveforms without harmonics. However, this assumption does not hold because harmonics can be observed under ideal grid voltage conditions. Additionally, harmonics of the fundamental and baseband cannot be neglected while analyzing the grid impact of DCFC. Besides, influence of the control system on harmonics needs to be taken into account. Consequently, modeling of the AC phase current harmonics mainly focusing on the low-frequency components is elaborated in section 3.2.

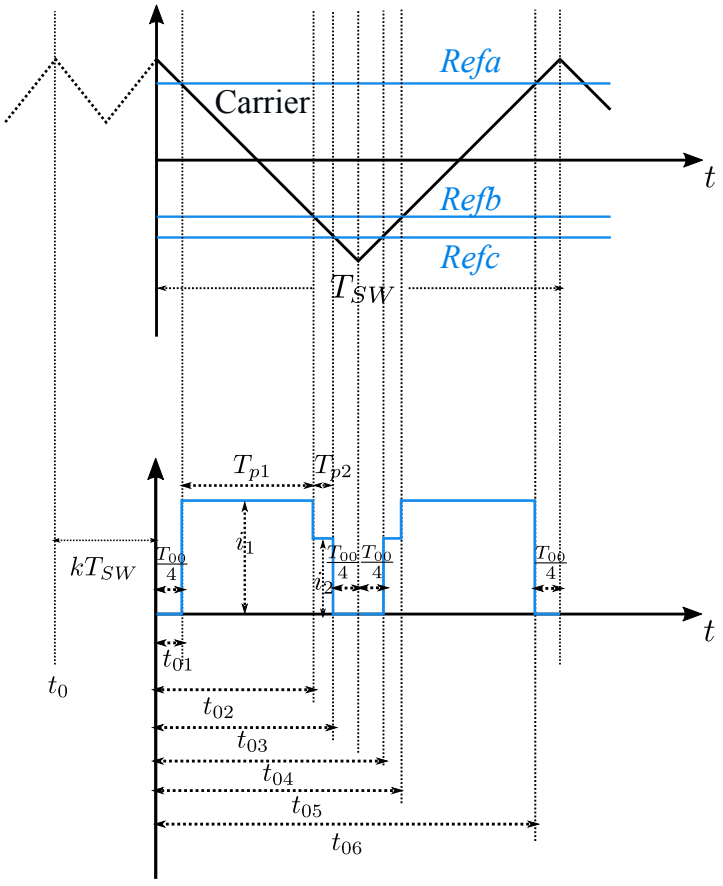


Figure 3.1: DC-link current i_L waveform during one switching period.

3.2 MODELING OF THE HARMONIC CURRENT SOURCE

With the modeling of the AFE input impedance Z_c, the current harmonic emission induced by the grid harmonic voltage can be estimated. However, when the grid voltages are ideal sinusoidal waveforms without harmonics, there are still current harmonics that exist in the input AC phase current of AFE due to the harmonic current source I_c. This harmonic current source is determined by the topology, the modulation method, the power filter, the switching frequency, the dead time, and also the DC-link voltage ripple.

As illustrated in Fig. 2.2, i_a, i_b, and i_c are the AC phase currents, V_{sa}, V_{sb}, and V_{sc} are the switch node voltages which are determined by the state of switches. To analyze the low frequency current harmonics drawn by the AFE, it is essential to first acquire the expression of the switch node voltage which is used to obtain the Fourier series of the switch node voltage. Then the h_{th} current harmonics can be derived from the h_{th} components of the switch node voltage.

3.2.1 *Modeling of the Switch Node Voltage*

As shown in Fig. 3.2, the switch node voltage of phase A can be expressed as (3.22),

$$V_{sa} = \begin{cases} U_{dc} & t \in (0, t_1), (t_2, t_3), \cdots, (t_N, T) \\ 0 & t \in (t_1, t_2), (t_3, t_4), \cdots, (t_{N-1}, t_N) \end{cases} \tag{3.22}$$

Grounded on the principle of asymmetrical sampled modulation technique (shown in Fig. 3.2), the switching time instant can be established as follows [23]. First, the wave angle y_k of the modulation waveform sampling time instant can be represented as,

$$y_k = \frac{\omega_1}{\omega_c}\pi(k-1) \quad k = 1, 2, 3, \cdots, N \tag{3.23}$$

where ω_1 is fundamental angular frequency, ω_c is angular frequency of triangular carrier, and $N = \frac{2\omega_c}{\omega_1}$. It should be noted that y_k is in reference to angle of the modulation waveform. Based on (3.23), the switching angle x_k can be derived as,

$$\begin{aligned} x_k &= x_T + x_s + \Delta x \\ &= \frac{\pi}{2} + (k-1)\pi + (-1)^{(k+1)} \cdot \frac{\pi}{2} \cdot M \sin y_k \end{aligned} \tag{3.24}$$

where M is the modulation index. The first part x_T of (3.24) represents the correction of x_k start point. The second part x_s of (3.24) is the wave angle of the modulation waveform sampling time which is in reference to the angle of the carrier. The third part Δx of (3.24) means the angle difference due to the intersection-point of the carrier and modulation waveforms.

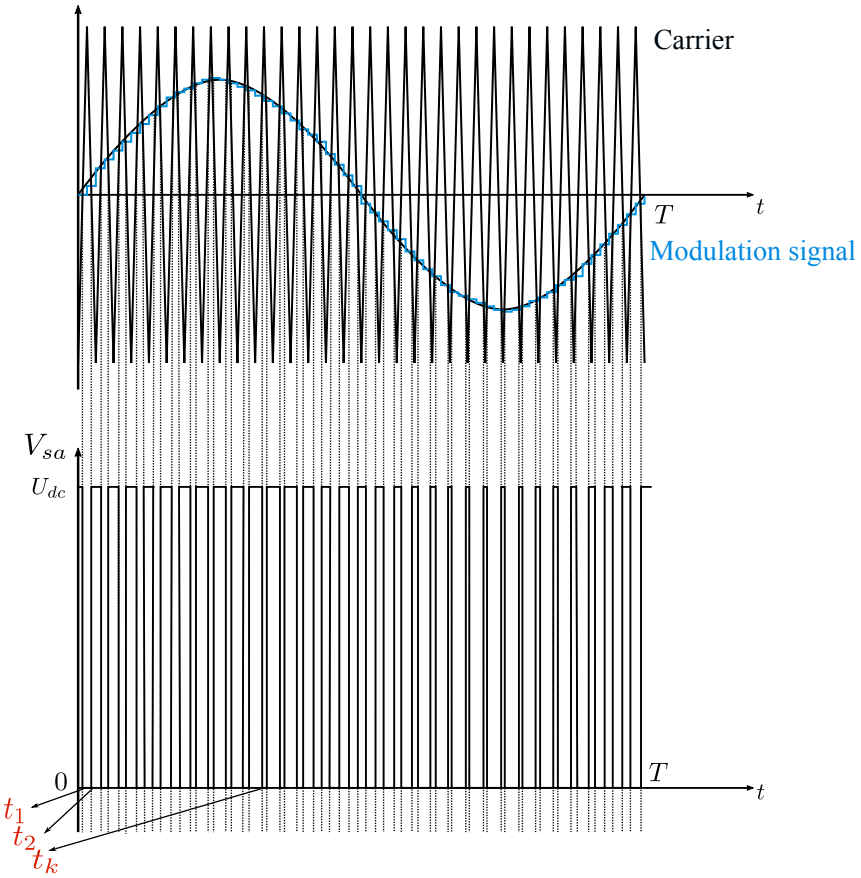


Figure 3.2: The Principle of regular-sampled PWM technique (asymmetrically sampling).

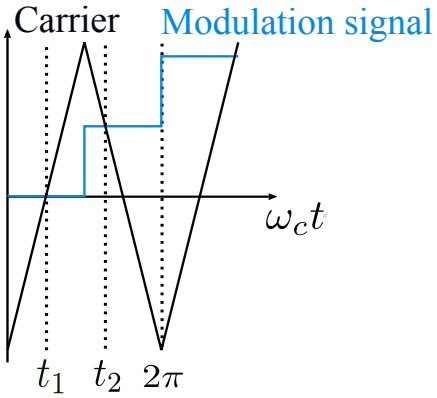


Figure 3.3: Part of the modulation and carrier waveforms.

According to (3.24), when $k = 1$, $x_1 = \frac{\pi}{2}$ which agrees with Fig. 3.3. When $k = 2$, $x_2 = \frac{\pi}{2} + \pi + (-1)^{(2+1)} \cdot \frac{\pi}{2} \cdot M \sin y_k$ which is also in accordance with Fig. 3.3. This confirms the correctness of the proposed equation.

In the next step, dead time effect on x_k is taken into consideration. The effect of the dead time t_d on the switch node voltages is demonstrated in Fig. 3.4, where the switching delay of the power switching devices is ignored [24].

It should be noted that the dead time effect on the switch node voltage (e.g. V_{sa}) depends on the polarity of the phase current (e.g., i_a). This yields the modified switching angle x_k^* as,

$$x_k^* = x_k + \frac{1 - sign(i)(-1)^k}{2} \cdot \omega_c T_d \tag{3.25}$$

where $sign(i) = 1$ if phase current $i_{a,b,c} \geq 0$, $sign(i) = -1$ if phase current $i_{a,b,c} < 0$, and T_d is the dead time period.

Apart from the previous analysis, the impact of actual waveforms of phase current (e.g. i_a) needs to be taken into account. As shown in Fig. 3.5, the phase current i_a jumps between positive and negative in the initial phase of the modulation signal. On the one hand, $i_a < 0$ during the first dead time period, which results in the switch node voltage V_{sa} being negative. On the other hand, $i_a > 0$ during the second dead time period, which leads to a positive switch node voltage. Therefore, the actual switch node voltage around the starting point of the modulation signal is not affected by the dead time. This phenomenon also applies to switch node voltages in the final phase of the modulation signal.

To obtain the time period without dead time effects, the average model for the two-level AFE is used (expressed in 2.1). The accurate phase current ripple shown in Fig. 3.5 can be obtained using (3.25). Therefore, the basic idea of time period T_{nd} without dead time effects calculation can be expressed as,

$$I_{average(a,b,c)}(T_{nd}) - \Delta I_{ripple(a,b,c)}(T_{nd}) = 0 \tag{3.26}$$

where $I_{average(a,b,c)}$ is the phase current calculated by the average model in (2.1), and $\Delta I_{ripple(a,b,c)}$ denotes the current ripple derived from the expression of $V_{sa,b,c}$ and $e_{a,b,c}$. Consequently, the switching time instant t_k within one fundamental period T can be finally obtained by the (3.27) and (3.28),

$$x_k^{**} = \begin{cases} x_k & t \in (t_0, t_0 + T_{nd1}) \\ x_k^* & t \in (t_0 + T_{nd1}, T - T_{nd2}) \\ x_k & t \in (T - T_{nd2}, T) \end{cases} \tag{3.27}$$

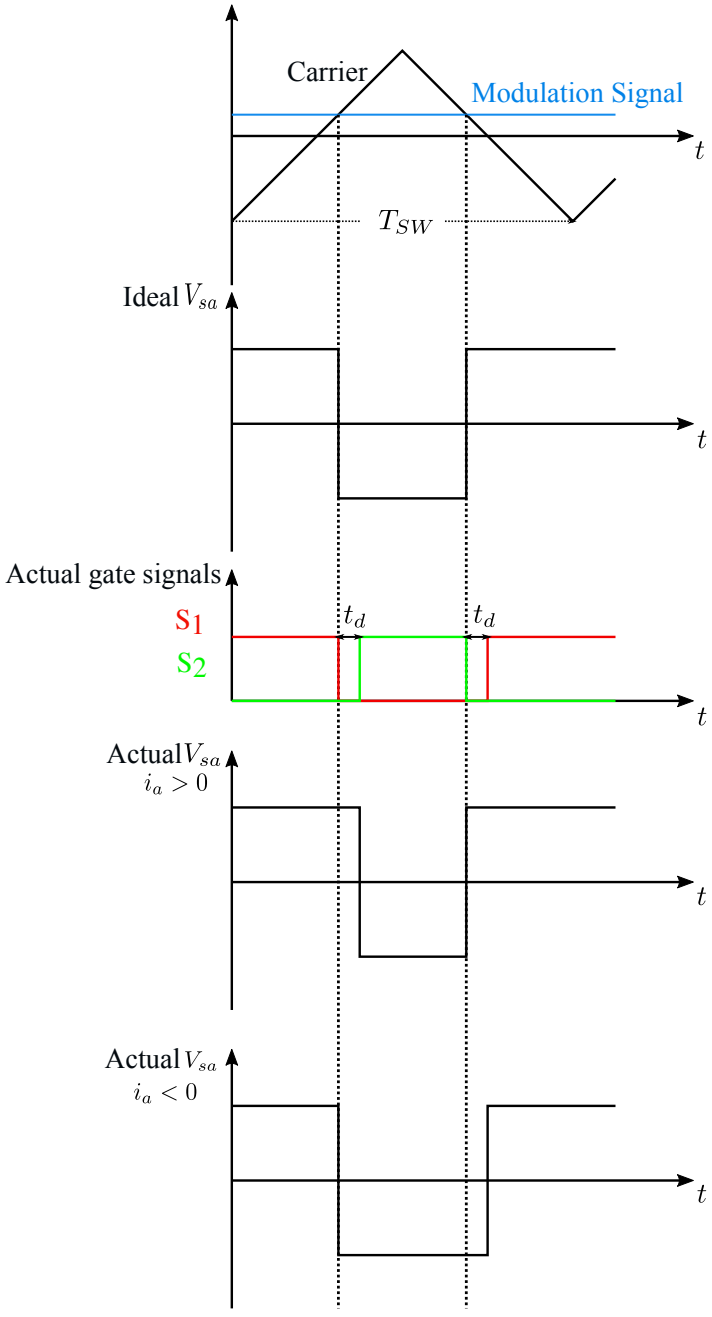


Figure 3.4: Switch node voltage of Leg-A considering the dead time effect.

where T_{nd1} is the first time period without dead time effects, and T_{nd2} is the second time period without dead time effects.

$$t_k = \frac{x_k^{**}}{N\pi} \times T_1 \tag{3.28}$$

where T_1 is the time period of the fundamental waveform.

According to (3.22) and (3.28), coefficient a_h of Fourier series can be gained,

$$\begin{aligned} a_h &= \frac{2}{T}\int_{t_0}^{t_0+T} V_{sa}(t)\cos(\frac{2\pi}{T}\cdot ht)dt \\ &= \frac{U_{dc}}{h\pi}\sum_{k=1}^{N}\sin(\frac{2\pi}{T}\cdot ht_k)\cdot(-1)^{(k+1)} \end{aligned} \tag{3.29}$$

Coefficient b_h of Fourier series can be calculated in the same way,

$$\begin{aligned} b_h &= \frac{2}{T}\int_{t_0}^{t_0+T} V_{sa}(t)\sin(\frac{2\pi}{T}\cdot ht)dt \\ &= \frac{U_{dc}}{h\pi}\sum_{k=1}^{N}\cos(\frac{2\pi}{T}\cdot ht_k)\cdot(-1)^{k} \end{aligned} \tag{3.30}$$

Based on (3.29) and (3.30), the h_{th} component of the switch node voltage (i.e., $V_{sa(h)}$, $V_{sb(h)}$, and $V_{sc(h)}$) can be obtained.

3.2.2 *Modeling of the Current Harmonics*

To calculate current harmonics, the per-phase voltage needs to be determined first. The potentials of switching nodes $V_{sa(h)}$, $V_{sb(h)}$, and $V_{sc(h)}$ derived in Section 3.2.1 are in reference the same node as node O in Fig. 2.2. The phase currents in the equivalent circuit (shown in Fig. 3.6) can be written as, using Kirchhoffs Voltage Law.

$$\begin{cases} V_o - V_{sa(h)} &= L\frac{di_{a(h)}}{dt} + Ri_{a(h)} \\ V_o - V_{sb(h)} &= L\frac{di_{b(h)}}{dt} + Ri_{b(h)} \\ V_o - V_{sc(h)} &= L\frac{di_{c(h)}}{dt} + Ri_{c(h)} \end{cases} \tag{3.31}$$

Similarly, it can be discovered using Kirchhoff's Current Law that,

$$i_{a(h)} + i_{b(h)} + i_{c(h)} = 0 \tag{3.32}$$

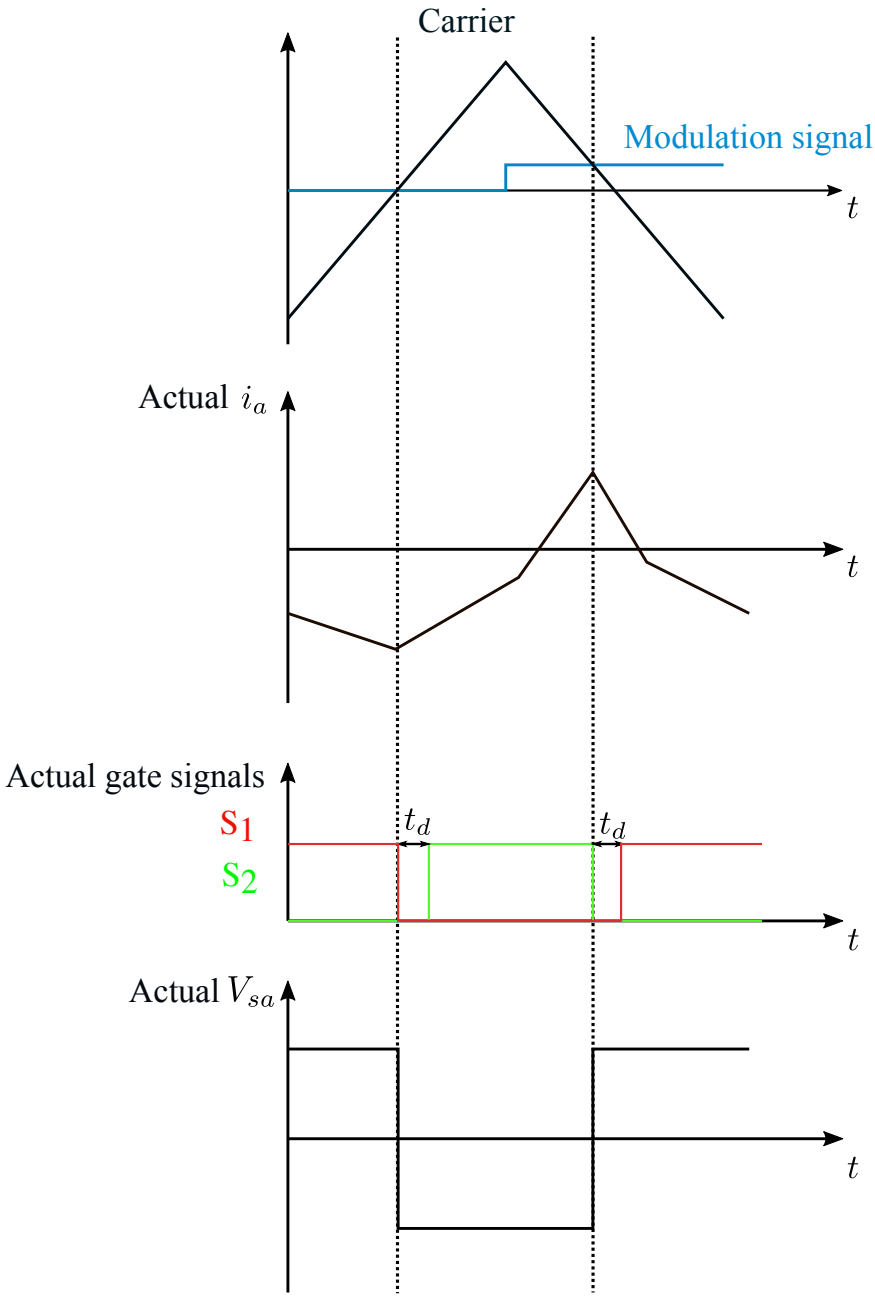


Figure 3.5: Actual waveforms of phase current i_a, switching node voltage V_{sa} at the start point of one switching period.

The following expression for V_o can be obtained by substituting (3.31) to (3.32),

$$V_o = \frac{V_{sa(h)} + V_{sb(h)} + V_{sc(h)}}{3} \tag{3.33}$$

Substituting (3.33) to (3.31),

$$\begin{aligned} L\frac{di_{a(h)}}{dt} + Ri_{a(h)} &= V_o - V_{sa(h)} \\ &= \frac{V_{sa(h)} + V_{sb(h)} + V_{sc(h)}}{3} - V_{sa(h)} \\ &= \frac{V_{sb(h)} + V_{sc(h)} - 2V_{sa(h)}}{3} \end{aligned} \tag{3.34}$$

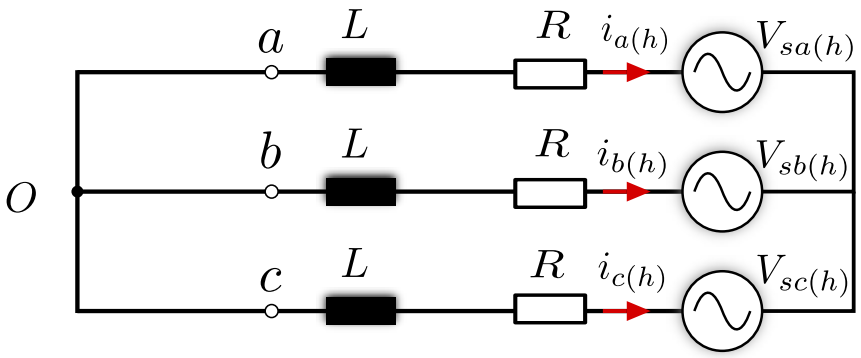


Figure 3.6: Equivalent circuit of the AFE.

Hence, the current harmonic of order h in frequency domain can be derived as,

$$i_{a(h)} = \frac{(V_{sb(h)} + V_{sc(h)} - 2V_{sa(h)})}{3(R + jh\omega L)} \tag{3.35}$$

where R is the resistance of the input filter, L is the inductance of the input filter, and h is the harmonic order. It is important to note that the grid voltages (i.e., e_a, e_b, and e_c) are assumed to be perfect sinusoidal waveforms with no bearing on the harmonics calculation.

In the previous calculation, only the modulation method and the dead time effects are taken into account. Regarding the high-frequency current harmonics exceeding the controller bandwidth, control system effects can be ignored. However, this doesn't apply to the low-frequency current harmonics. To determine the low-frequency current harmonics influenced by the control system and the DC-link voltage fluctuations, the open-loop current harmonics need to be integrated into

the control block diagram to obtain the accurate values of the low-frequency harmonics in the AC phase currents $i_{\mathrm{a,b,c}}$.

As shown in Fig. 3.7, a new branch for the open-loop current harmonics ($\tilde{\boldsymbol{i}}_{\boldsymbol{dqDT}}$) is added to linearized small signal model in Fig. 2.9. To obtain the closed-loop current harmonics, the grid voltage perturbation $\tilde{\boldsymbol{e}}_{\boldsymbol{dq}}$ is ignored. As illustrated in Fig. 3.8, the transfer blocks in green dashed boxes can be omitted.

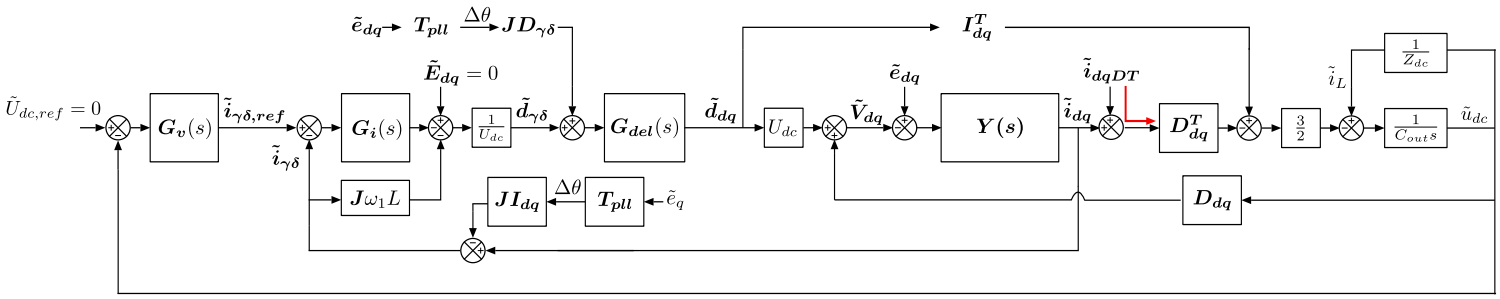


Figure 3.7: Block diagram of the linearized small signal model of AFE control system considering the open-loop current harmonics.

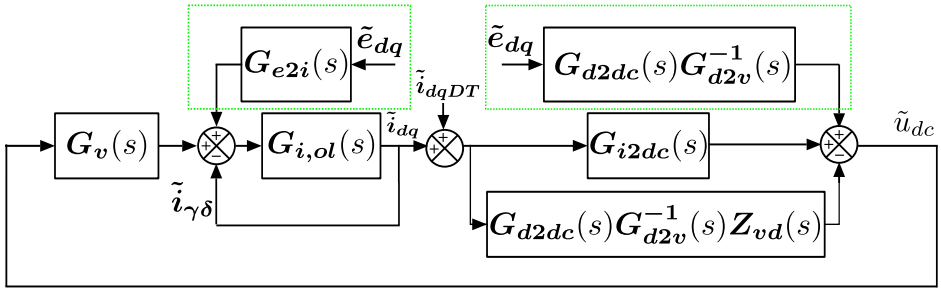


Figure 3.8: Simplified block diagram of the linearized small signal model of AFE control system considering the open-loop current harmonics.

The final simplified block diagram of the control system for closed-loop input current harmonics calculation is illustrated in Fig. 3.9, where the input $\tilde{\boldsymbol{i}}_{\mathrm{dqDT}}$ is $i_{\mathrm{a(h)}}$ in (3.35), $\tilde{\boldsymbol{i}}_{\mathrm{dqtot}}$ is the DCFC harmonic current source I_{c}, $\boldsymbol{G}_{\mathrm{v}}(s)$ is the transfer function of the voltage controller, $\boldsymbol{G}_{\mathrm{i2dctot}}$ and $(\boldsymbol{I}+\boldsymbol{G}_{\mathrm{ol}})^{-1}\boldsymbol{G}_{\mathrm{ol}}$ are the simplified blocks of the control diagram in Fig. 3.7 (expressed in (3.36) (3.37)).

$$
\begin{aligned}
\boldsymbol{G_{d2dc}}(s) &= \frac{3}{2}\frac{1}{C_{out}s+\frac{1}{Z_{dc}}}\boldsymbol{I_{dq}^T} \\
\boldsymbol{G_{i2dc}}(s) &= \frac{3}{2}\frac{1}{C_{out}s+\frac{1}{Z_{dc}}}\boldsymbol{D_{dq}^T} \\
\boldsymbol{G_{d2v}}(s) &= U_{dc}+\frac{3}{2}\frac{1}{C_{out}s+\frac{1}{Z_{dc}}}\boldsymbol{D_{dq}I_{dq}^T} \\
\boldsymbol{Z_{vdpasdq}}(s) &= \boldsymbol{Y^{-1}}+\frac{3}{2}\frac{1}{C_{out}s+\frac{1}{Z_{dc}}}\boldsymbol{D_{dq}D_{dq}^T} \\
\boldsymbol{G_{i2dctot}}(s) &= \boldsymbol{G_{i2dc}}-\boldsymbol{G_{d2dc}G_{d2v}^{-1}Z_{vdpasdq}}
\end{aligned}
\tag{3.36}
$$

$$
\begin{aligned}
\boldsymbol{H}(s) &= \frac{1}{U_{dc}}\boldsymbol{Z_{vdpasdq}^{-1}G_{d2v}G_{del}} \\
\boldsymbol{H_{de}}(s) &= (\boldsymbol{I}-\boldsymbol{HJ}\omega_1 L)^{-1}\boldsymbol{H} \\
\boldsymbol{G_{ol}}(s) &= \boldsymbol{H_{de}H_i}
\end{aligned}
\tag{3.37}
$$

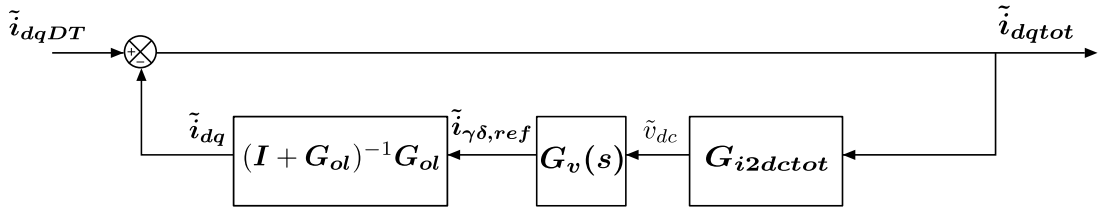


Figure 3.9: Final simplified block diagram for current harmonic source I_c.

Nevertheless, the frequency coupling effect cannot be ignored. As illustrated in (2.24), an open-loop current harmonic with frequency ω_h yields current harmonics with two frequencies, one is ω_h, the other one is $2\omega_1 - \omega_h$ which is resulted from frequency coupling effect. For instance, suppose the open-loop 5_{th} (frequency -$5\omega_1$) current harmonic exists. It should first be translated into dq-frame before being included into the control diagram. The dq transformation implies translating the frequency spectrum from ω to $\omega - \omega_1$ [17]. Hence, the frequency spectrum of the 5_{th} negative sequence harmonic becomes $-5\omega_1 - \omega_1 = -6\omega_1$ in the dq-frame. Via $\boldsymbol{G}_{i2dctot}$, $\tilde{v}_{dc}$ induced by $\tilde{\boldsymbol{i}}_{dqDT}$ can be acquired. This complex transfer matrix results in a double-frequency $\tilde{v}_{dc}$ which is composed of positive ($+6\omega_1$) and negative sequence ($-6\omega_1$) components. These two components yields the closed-loop current harmonics with frequency $+7\omega_1$ and $-5\omega_1$ respectively. For the same reason, an open-loop current harmonic with frequency $+7\omega_1$ will result in two

closed-loop current harmonics, $+7\omega_1$ and $-5\omega_1$. Therefore, both 5_{th} open-loop current harmonic and 7_{th} open-loop current harmonic produce the 5_{th} closed-loop current harmonic.

3.3 SUMMARY

Chapter 3 first introduces the prior work on harmonic emission modeling using the Double Fourier Series Method, which mainly focuses on the harmonic components around the carrier and carrier-sideband frequency. Then, to model the low-frequency harmonic current source of AFE I_c, the switching node voltage is computed first, and open-loop current harmonics can be estimated using the Fourier series of the switching node voltage. Finally, the AFE control diagram is combined with the open-loop current harmonic to obtain the I_c while the frequency coupling effect is taken into account.

4

SIMULATION AND EXPERIMENT

To verify the correctness of the proposed model in Chapter 2 and Chapter 3, case studies in both simulation and experiment are elaborated in this chapter.

4.1 SIMULATION VALIDATION

To start with, time-domain simulations are conducted in PLECS Standalone. The AFE system in PLECS is shown in Fig. 4.1. The control system parameters and the electrical parameters chosen for the PLECS simulation are illustrated in Tables 4.1 and 4.2.

Based on the analytical solution achieved in Chapter 3, the calculation is carried out in Python to obtain the theoretical harmonic spectrums of AC phase current. Low frequency harmonics with order 5_{th}, 7_{th}, 11_{th}, and 13_{th} are selected for comparison between simulation and analytical model since they are the most damaging frequency components to power devices and grid [21].

An AC power system with a low SCR is characterized as a weak grid. Conversely, a high SCR results in a strong grid. In order to compare the simulation with the analytical model under different grid conditions, case studies with two different SCR are conducted in the PLECS simulation: one is SCR=5, the other is SCR=30. The results are shown in Fig. 4.2 and Fig. 4.3 respectively. It was found that the results achieved with the proposed analytical model are matching the results achieved by the PLECS simulation well.

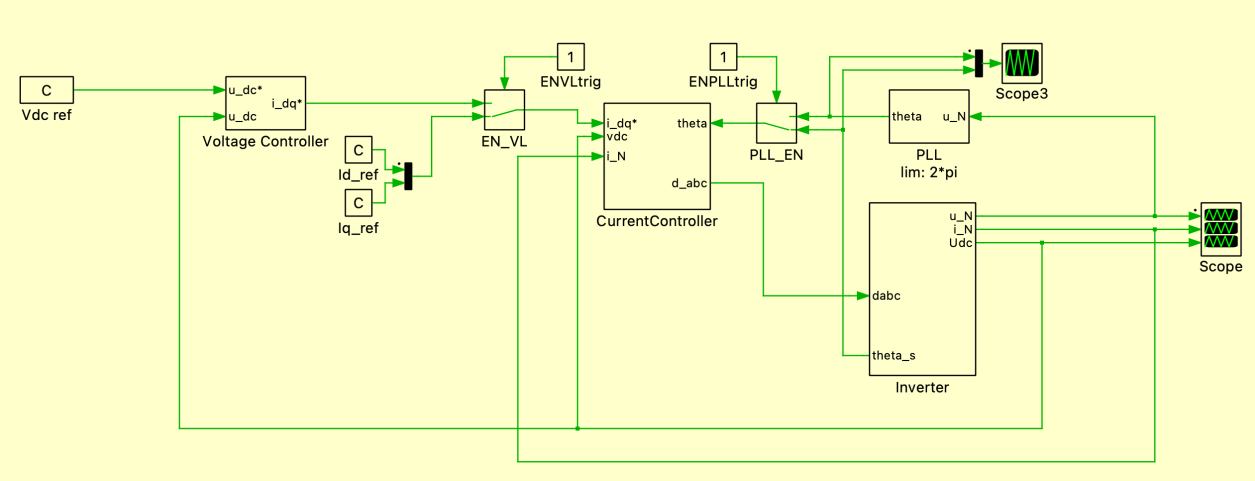


Figure 4.1: AFE with the control system in PLECS standalone.

Table 4.1: Circuit Parameters of the AFE in Fig. 2.2.

Parameters	Value
Grid line-to-neutral peak voltage E_g/V	$230\sqrt{2}$
Grid fundamental frequency f_1/Hz	50
Input filter inductance L/µH	250
Input filter resistance R/mΩ	20
DC-link voltage U_{dc}/V	800
Output capacitor V_{out}/µF	1500
AFE switching frequency f_{sw}/kHz	40
AFE control sampling frequency f_s/kHz	80
AFE dead time T_d/µs	1
Reference modulation index M	0.40577494
Output power P_{out}/kW	30

Table 4.2: Controller Parameters of the AFE in Fig. 2.2.

Parameters	Value
Proportional gain of current controller K_{pi}	1.256637
Integral gain of current controller K_{ii}	100.530965
Proportional gain of voltage controller K_{pu}	6.148755
Integral gain of current controller K_{iu}	254.664516
Proportional gain of SRF-PLL K_{ppll}	0.57950647
Integral gain of SRF-PLL K_{ipll}	109.23439616

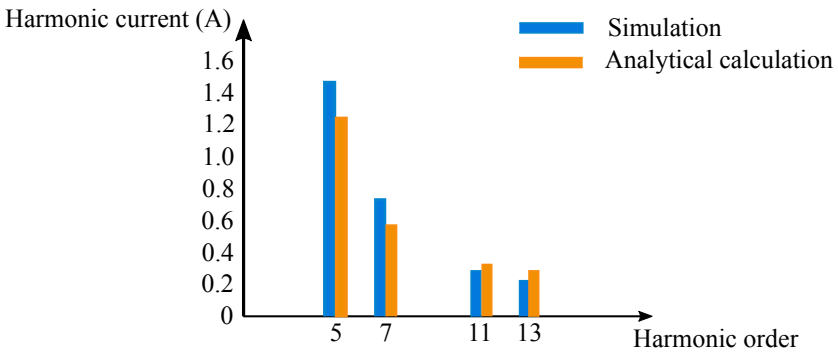


Figure 4.2: The harmonic spectrum of input phase current (SCR=5).

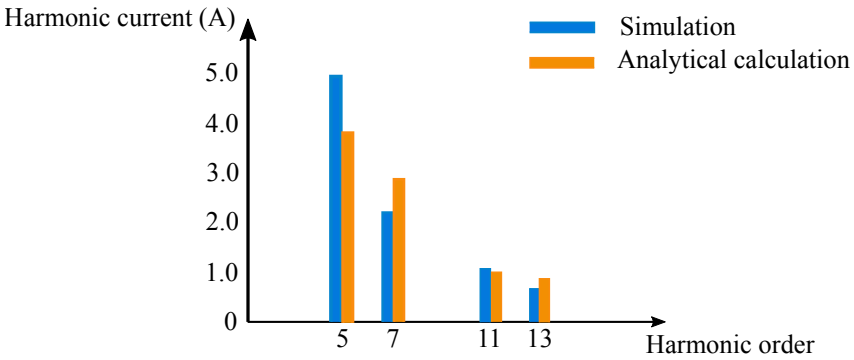


Figure 4.3: The harmonic spectrum of input phase current (SCR=30).

4.2 EXPERIMENT VALIDATION

Following the PLECS simulation, experiments are carried out for further validation. The actual topology for the experiment is shown in Fig. 4.4. Two relays are added after the grid emulator to guarantee safety. The experiment equipment and the measuring instruments are specified in Table 4.3 and 4.4.

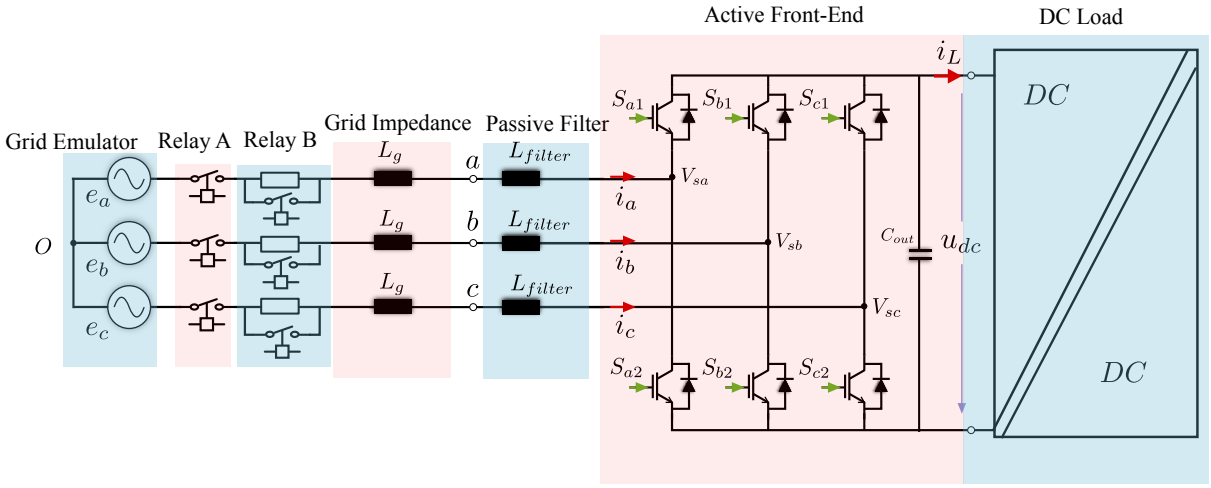


Figure 4.4: Topology of the experiment setup.

Table 4.3: Type of experiment equipment.

Type	Function
IMPERIX Rapid prototyping controller B-Box RCP 3.0	Modular control platform for AFE control
IMPERIX PEB 8032/6-10kW	Implementation of AFE
IMPERIX passive filters box	Grid impedance and L filter
CINERGIA grid emulator (GE-20)	Reproduce grid voltages under various conditions
DELTA ELEKTRONIKA DC power supply SM500-CP-90	DC load of AFE

Table 4.4: Type of measuring instruments.

Type	Function
KETSIGHT N2791A	Differential probe for DC-link voltage U_{dc} measurement
YOKOGAWA PCB050 701929	Current probe for AC phase current $i_{\mathrm{a,b,c}}$ measurement
YOKOGAWA DLM2034	Oscilloscope for signal observation

Parameters provided in table 4.5 and 4.6 are employed in the experiment. Note that the experiment adopts different parameters from the simulation. The reason for this is that personal safety must be ensured during the experiment. As a result, lower DC-link and grid voltages compared with simulation are chosen. Besides, the values of the grid filter inductance and the output capacitor change because of the actual device types. These changes result in different controller parameters in Table. 4.6.

Table 4.5: Circuit parameters of the AFE in the experiment.

Parameters	Value
Grid line-to-neutral peak voltage E_g/V	$110\sqrt{2}$
Grid fundamental frequency f_1/Hz	50
Input filter inductance L/µH	250
DC-link voltage U_{dc}/V	385
Output capacitor C_{out}/mF	1.67
AFE switching frequency f_{sw}/kHz	20
AFE control sampling frequency f_s/kHz	40
AFE dead time T_d/µs	1

Table 4.6: Controller parameters of the AFE in experiment.

Parameters	Value
Proportional gain of current controller K_{pi}	9
Integral gain of current controller K_{ii}	1000
Proportional gain of voltage controller K_{pu}	0.2
Integral gain of current controller K_{iu}	2
Proportional gain of SRF-PLL K_{ppll}	1.21
Integral gain of SRF-PLL K_{ipll}	228.40

4.2.1 *Experiment Procedure*

Firstly, a brief introduction about the devices used is presented. In the software interface of CINERGIA grid emulator (shown in Fig. 4.5), operation status can be controlled directly on the personal computer (PC). Additionally, each phase grid voltage can be independently configured in the AC tab: RMS value, phase delay, harmonics content, etc. For B-Box RCP 3.0 provided by IMPERIX, the user can compile the code in Simulink by hitting Ctrl + B. This action will generate and compile the code which is posted to the B-Box RCP automatically. At the same time, the BB control utility software is launched. Its role is to control the B-BOX RCP and keep track of the measured variables. The initial window (illustrated in Fig. 4.6) allows the user to select the IP address of the B-Box RCP so that the specific device can be connected. Depending on whether the B-Box RCP is connected to the computer directly or through a network, this IP will be dynamic or static [25].

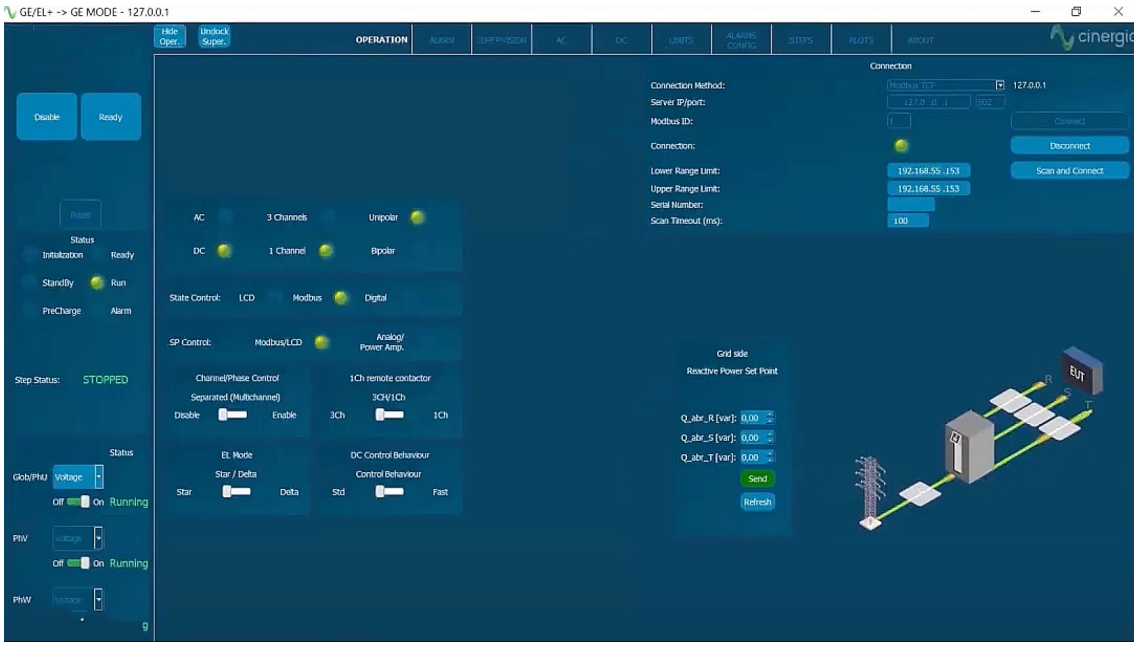


Figure 4.5: Interface of the Cinergia grid emulator.

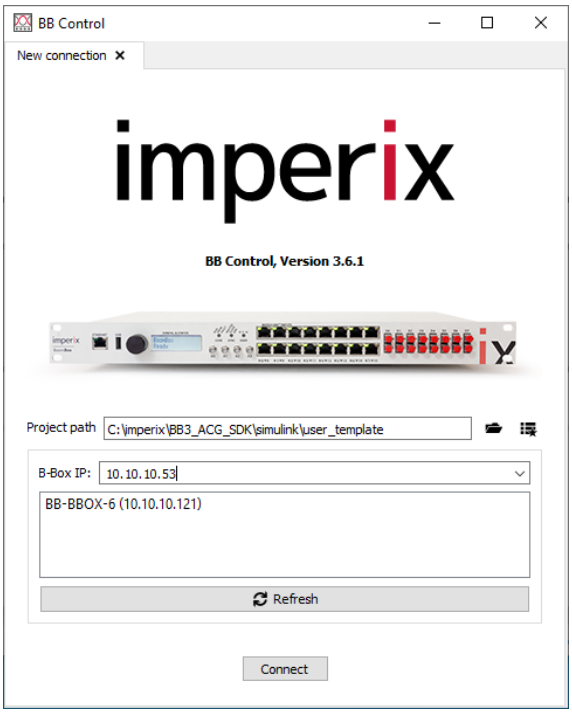


Figure 4.6: IMPERIX BB control window for connection.

After connection, the BB Control utility software (shown in Fig. 4.7) can add the control variables in the Debugging tab which can be adjusted during connection. In the Datalogging tab, variables can be monitored. In subsequent experiments, grid voltage $e_{a,b,c}$, AC phase current $i_{a,b,c}$, DC-link voltage u_{dc} are monitored.

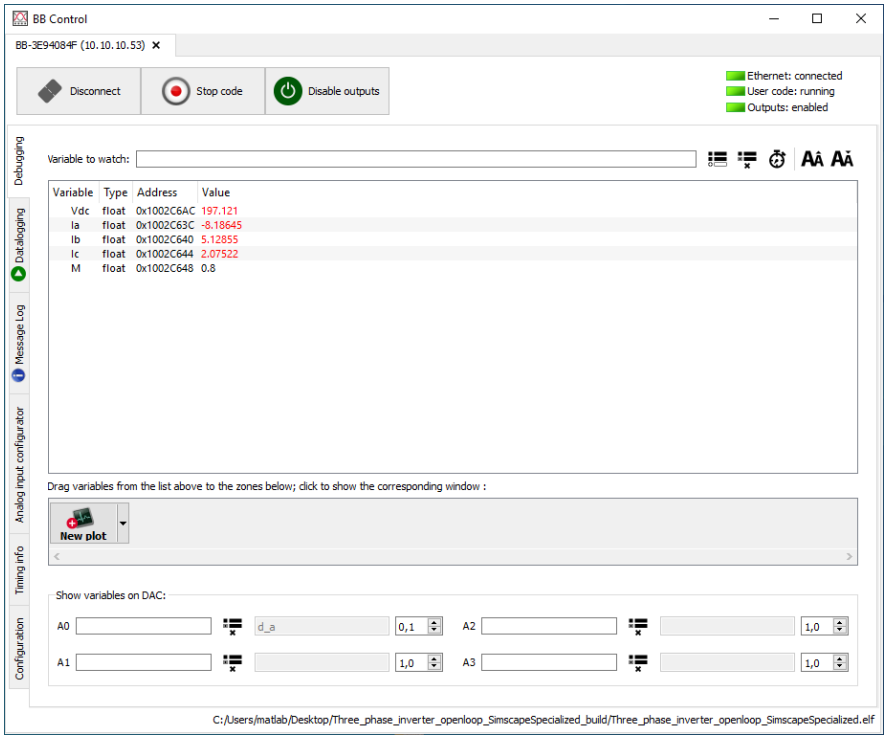


Figure 4.7: IMPERIX BB control window after connection.

Fig. 4.8 indicates the experiment setup. To conduct the experiment, the grid emulator should be enabled first to be ready for serving as grid voltages $e_{a,b,c}$. Then desired values for grid voltages are set. After that, values are sent to the grid emulator and it starts to run. Hereafter, the status of relay A and relay B needs to be changed from 0 to 1 in the Debugging tab of IMPERIX BB control window. Till now, the whole system worked as an uncontrolled grid-tied rectifier. Then outputs of IMPERIX BB control window are enabled and the controller starts to work. At this time, I_{max} (shown in Fig. 4.9) for voltage controller protection in B-Box RCP 3.0 is increased so that the load current can reach the desired level. In addition, the current controller's V_{dcsw} (depicted in Fig. 4.10) is changed from 0 to 1, changing the switch's input to V_{dcref}. Finally, the DC load is adjusted to the required value and the waveforms data can be saved.

After data saving, the DC load current should be set to zero first. Afterwards, the output of the grid emulator is disabled while values of grid voltages are cleared and sent to the grid emulator. Then relay A and relay B are off. V_{dcsw} and I_{max} come back to the initial values before experiment.

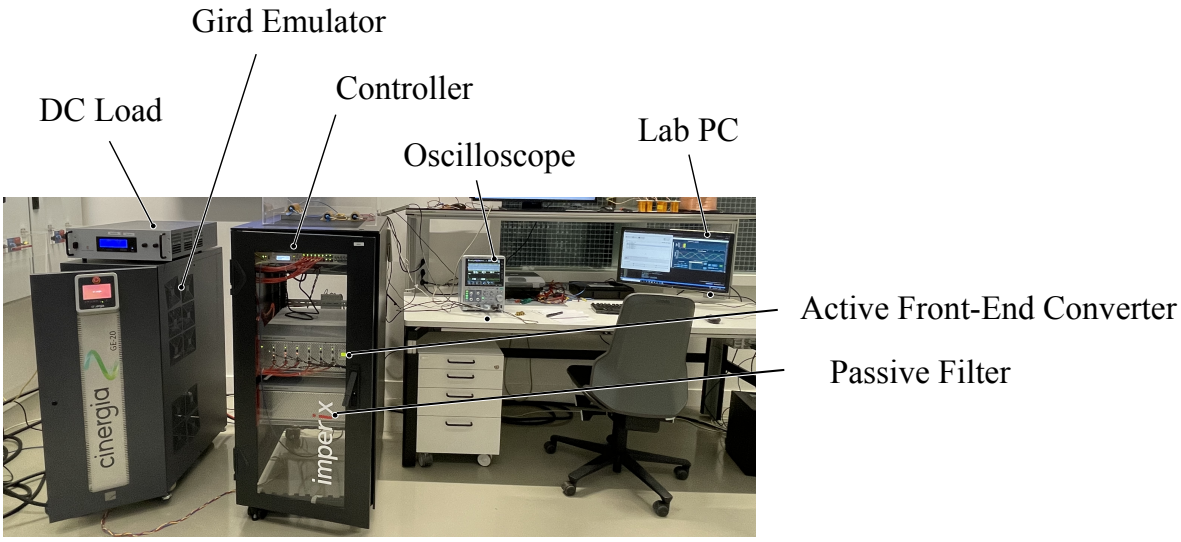


Figure 4.8: Experiment setup including the grid emulator, AFE, controller, etc.

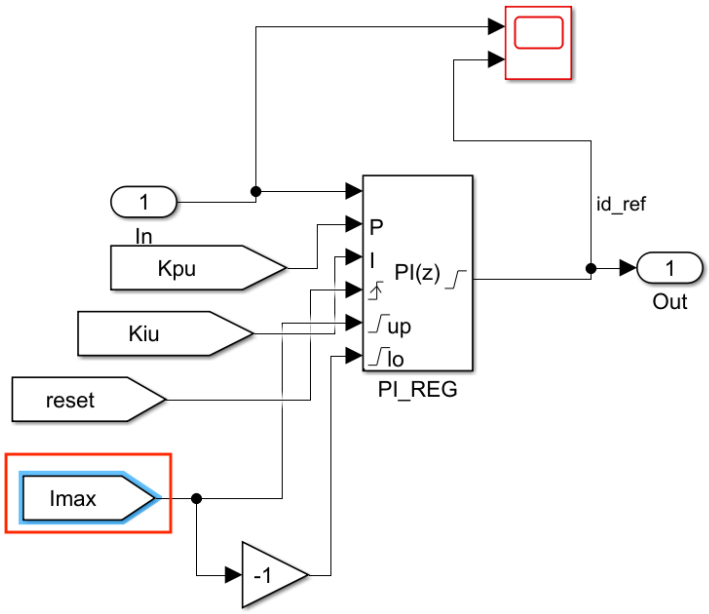


Figure 4.9: Voltage controller in MATLAB/Simulink of the experiment.

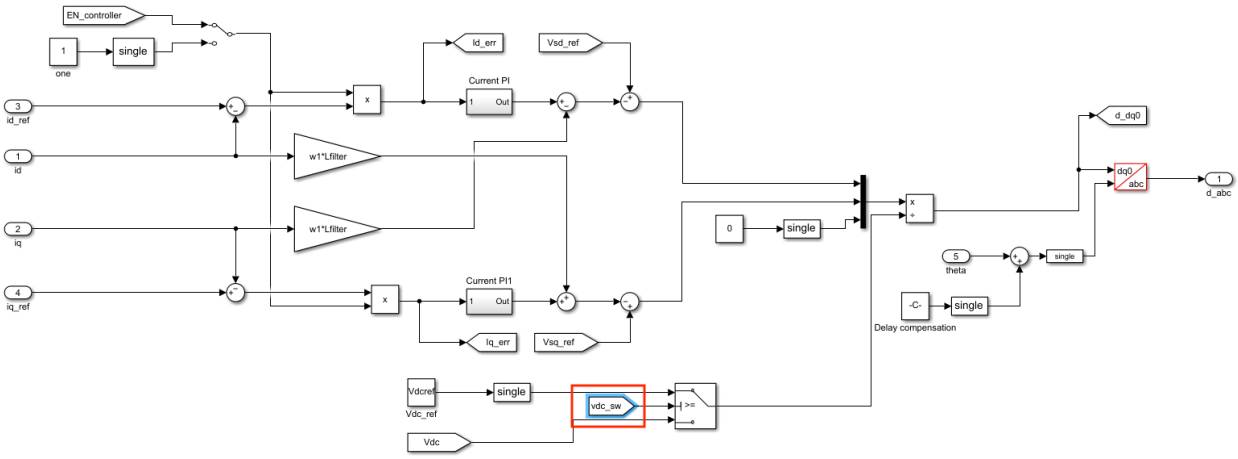


Figure 4.10: Current controller in MATLAB/Simulink of the experiment.

4.2.2 *Experiment Results*

Experiments with four operating points are carried out: the DC load power can be 1 kW or 2 kW, and the grid impedance L_g can be 0 or 2.5 mH which represent different load and grid conditions. Experiment results are shown from Fig. 4.11 to Fig. 4.14. The simulation results are consistent with the expected harmonics as discussed in Chapter 3.

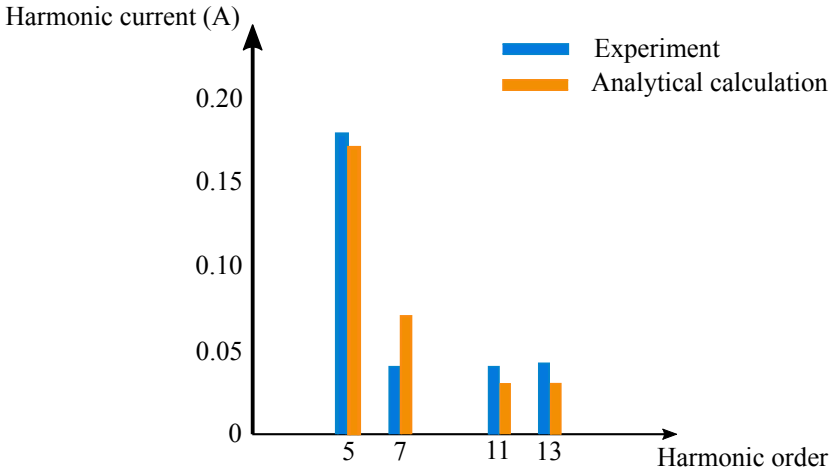


Figure 4.11: The harmonic spectrum of input phase current ($P_L = 1$ kW, $L_g = 2.5$ mH).

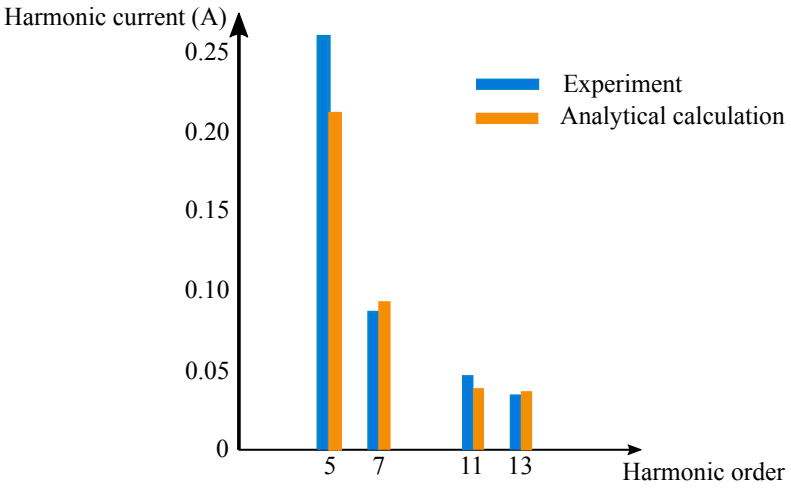


Figure 4.12: The harmonic spectrum of input phase current ($P_L = 2\,\text{kW}$, $L_g = 2.5\,\text{mH}$).

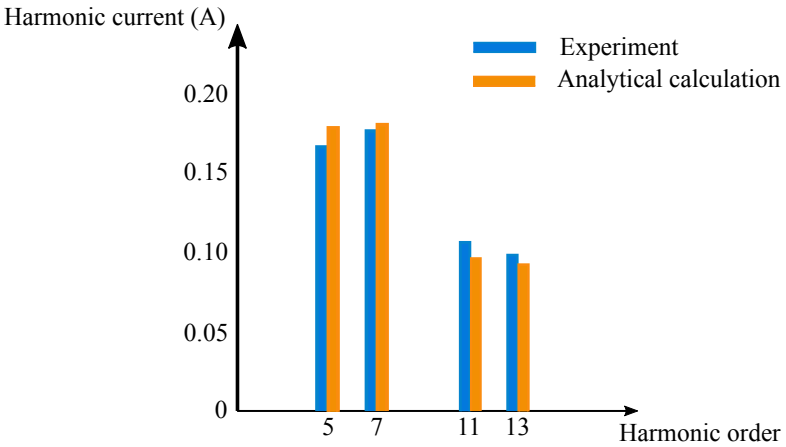


Figure 4.13: The harmonic spectrum of input phase current ($P_L = 1\,\text{kW}$, $L_g = 0\,\text{mH}$).

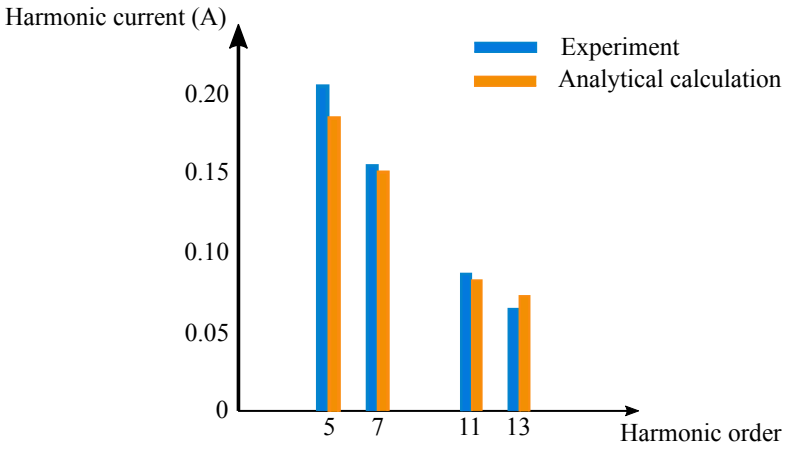


Figure 4.14: The harmonic spectrum of input phase current ($P_{\mathrm{L}} = 2\,\mathrm{kW}$, $L_{\mathrm{g}} = 0\,\mathrm{mH}$).

4.3 SUMMARY

In this chapter, the time-domain simulation in PLECS and the experiment in the lab are carried out to verify the correctness of the proposed analytical model for harmonic current source I_{c}. The simulation and experiment results show that the proposed model works well in estimating the low-frequency current harmonics.

5

DESIGN AND IMPLEMENTATION OF CLOUD PLATFORM

The modeling procedure for the input impedance Z_c and the harmonic current source I_c of DCFCs is explained in detail in the previous chapters. In this chapter, the IoT based harmonic emission estimation tool for DCFC is built based on the proposed analytical model. The IoT service is provided by Microsoft Azure.

5.1 BI-DIRECTIONAL COMMUNICATION BETWEEN DC FAST CHARGER AND BACK-END APP

The Internet of Things, or IoT, is a collection of managed and platform services that connect, monitor, and control billions of IoT assets across edge and cloud. It also comprises security and operating systems for IoT device and equipment, as well as data and analytics that enable users to design, deploy, and operate IoT applications [26].

Among the Microsoft Azure services, IoT Hub is an IoT service that operates as a central message hub for two-way communication between an IoT application and the devices it administers. It is a Platform-as-a-Services (PaaS) managed service housed in the cloud which provides highly secure and dependable communication. Furthermore, the cloud-based application allows remote design adaptation and considerably easy online parameter adjustment. Consequently, in this thesis, the Azure IoT hub is used for bi-directional communication between the DCFC and the cloud-based harmonic emission estimation application.

To achieve the aforementioned bi-directional communication function, two apps are required, one is the back-end app for device management, and the other is the simulated DCFC app. These two apps are both written in python which can work together to remotely reboot the charger, and calculate the actual operating power based on the users' input parameters.

According to the Microsoft tutorial [27], the first step is to create an IoT hub using Azure portal and then create a device identity in this IoT hub. The IoT hub named *IoThubYL* is shown in Fig. 5.1, which belongs to the resource group *RG_YL* and is located in West Europe. Using Azure CLI, the device identity named *myDeviceId* is created, as illustrated in Fig. 5.2. With specific commands, the device connection string is retrieved.

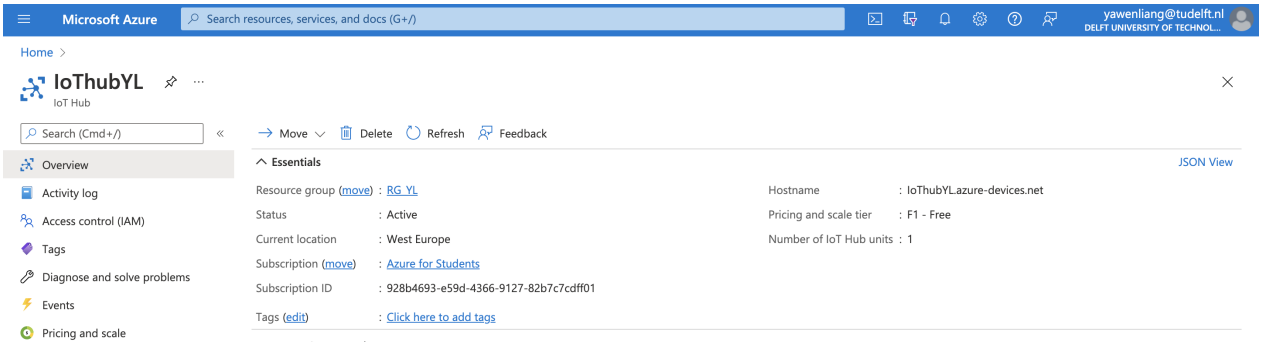


Figure 5.1: The screenshot of the IoT hub overview.

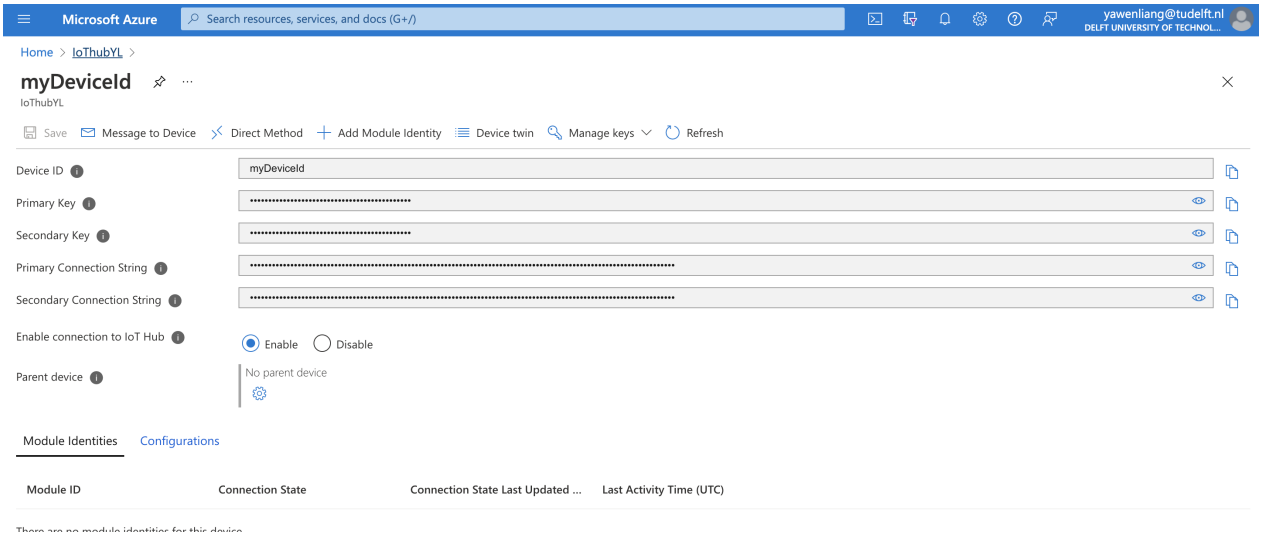


Figure 5.2: The screenshot of the IoT hub device.

The second step is to create a DCFC app that contains direct methods that can reboot the charger and calculate the actual operating power based on the maximum power setting from the user. The direct methods are invoked from the cloud and enable bi-directional communication. To be more specific, the simulated DCFC app is connected to the IoT hub *IoThubYL* with the connection string of the device identity, which can respond to the power calculation direct method and the reboot direct method called by the cloud. After being invoked by the cloud, the DCFC app simulates a physical reboot and reports the actual operating power and the time for the last reboot back to the cloud.

The third step is to create a backend service app that invokes the power calculation direct method and the reboot direct method in the simulated DCFC app through the IoT hub *IoThubYL*, displays the response, and displays the updated reported properties. To invoke a direct method on a device through IoT Hub, the backend service app needs the service connect permission. By default, every

IoT Hub is created with a shared access policy named service that grants this permission. Therefore, the IoT Hub connection string is needed for the service policy. The function *iothub_reboot()* in A.2 is responsible call the direct method to reboot the target device, then query for the device twins and get the last reboot time. The function *iothub_devicemethod_sample_run(maximumpower)* can call the direct method to calculate the actual power and get the calculation result.

An overview of the system structure of the IoT platform is illustrated in Fig. 5.3. The device (simulated DCFC) block and the backend block have already been discussed in detail before. The next step is to create the front-end, which is elaborated on in the next section.

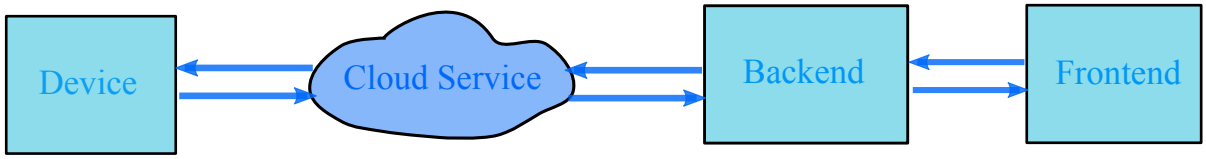


Figure 5.3: An overview of the system structure of the IoT platform.

5.2 WEB FRAMEWORK

Django is chosen in this thesis for the front-end(webpage) design since it is a free and open-source web framework that is based on python. This web framework follows the Model View Template (MVT) architecture which can be developed easily and rapidly. To understand the structure of Django deeper, the role of each part in Fig. 5.4 is elaborate in detail subsequently.

The **Model** is responsible for the data and databases behind the entire web application. In other words, the model works as a logic tool to interact with a model to carry the data.

The **template** layer retrieves the users' input data and determines how a user sees the response so that it is responsible for the interface design.

Universal Resource Locator (URL) is used to provide the addresses of the resources, e.g. input parameters, on the webpage. The **URLs.py** stores all the Universal Resource Locators (URLs) of our web application in the URL patterns

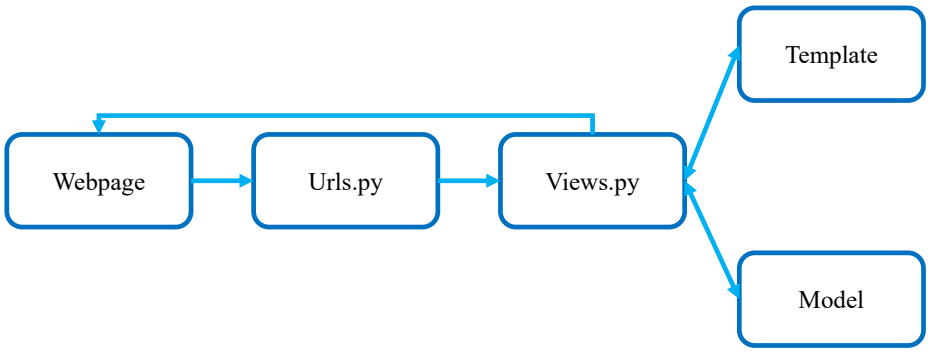


Figure 5.4: Diagram of django's structure.

The **View.py** handles the communication between the **Model** and users. Once there is a web request from the **Webpage**, it will take the URL patterns of the web request and Django will try to search the URL in **Urls.py**. If something is matched, then Django will trigger the associated function in the **Views.py**, which means that **Views.py** will take up the request and look for **models** and **templates** to return a response which gets renders into the HTML **templates**. In this webpage framework, the request can be POST or GET type, and the response can be a 404 error, the reboot time, the actual power, the impedance curve image, and the bar graph indicating current harmonics.

There are three main functions in the **View.py** to achieve the desired performance, i.e, *reboot*, *get_power*, and *get_parameter*. Function *reboot* can reboot the simulated DCFC and return the device response, i.e., reboot successfully with the reboot time or reboot failure. For the function *get_power*, it will sent the power input from the **webpage** to the simulated device app once the user click 'SUBMIT' button and then return the device response back to the webpage. The function *get_parameter* is responsible for sending input parameters from the **webpage**, e.g. filter inductance, to the backend app for harmonics and input impedance calculations. The calculation result will be returned back to the webpage for display.

The screenshot of this Django project in Visual Studio Code is given in Fig. 5.5. It can be observed that there are two terminals in the VS code, one is the backend of webpage, the other one is the simulated DCFC.

Based on the aforementioned discussion, the work flow of Django-based webpage can be concluded as follows: first, a user types in the required information and sends a web request, e.g. click the 'Submit' button. Django considers the request as a URL and matches it with the existing

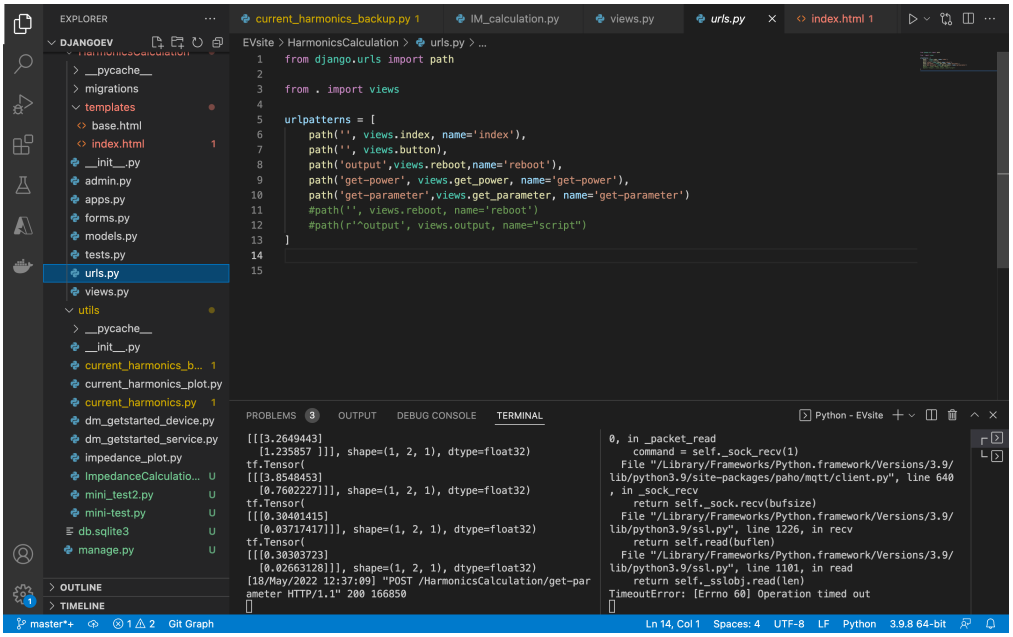


Figure 5.5: Screenshot of the webpage backend project.

URL paths in the **urls.py** file. Once the URL matches, Django jumps to the **views.py** folder and calls the corresponding function. The triggered **views** looks for **models** and **templates** to interact with, and then it returns the response back to the user. The screenshot of the webpage is shown in Fig. 5.6.

To show the webpage function clearly, one example is given. First, users enter the maximum power of the DCFC and click the 'SUBMIT' button. After that, the corresponding URL will be matched with the existing URL path *get-power* in the **urls.py** file. Then the corresponding function *get_power* in **views.py** will be triggered. In particular, this function will send the input maximum power to the simulated DCFC app via cloud communication. Then the DCFC app will calculate the actual operating power based on the received message and send the calculation result back to function *get_power* in **views.py**. Finally, this response will be returned back to the webpage. After getting the actual operating power, parameters shown in Tab. 5.1 are typed in to get the input impedance curves and current harmonics spectrum. Once the 'OK' button is clicked, the same workflow as the previous will be executed. The corresponding function in **views.py** calculates the impedance curves and current harmonics based on these input information. When the calculation is finished, the figures of impedance curves and current harmonics will be returned back to the webpage. The webpage response is illustrated in Fig. 5.8 and Fig. 5.9.

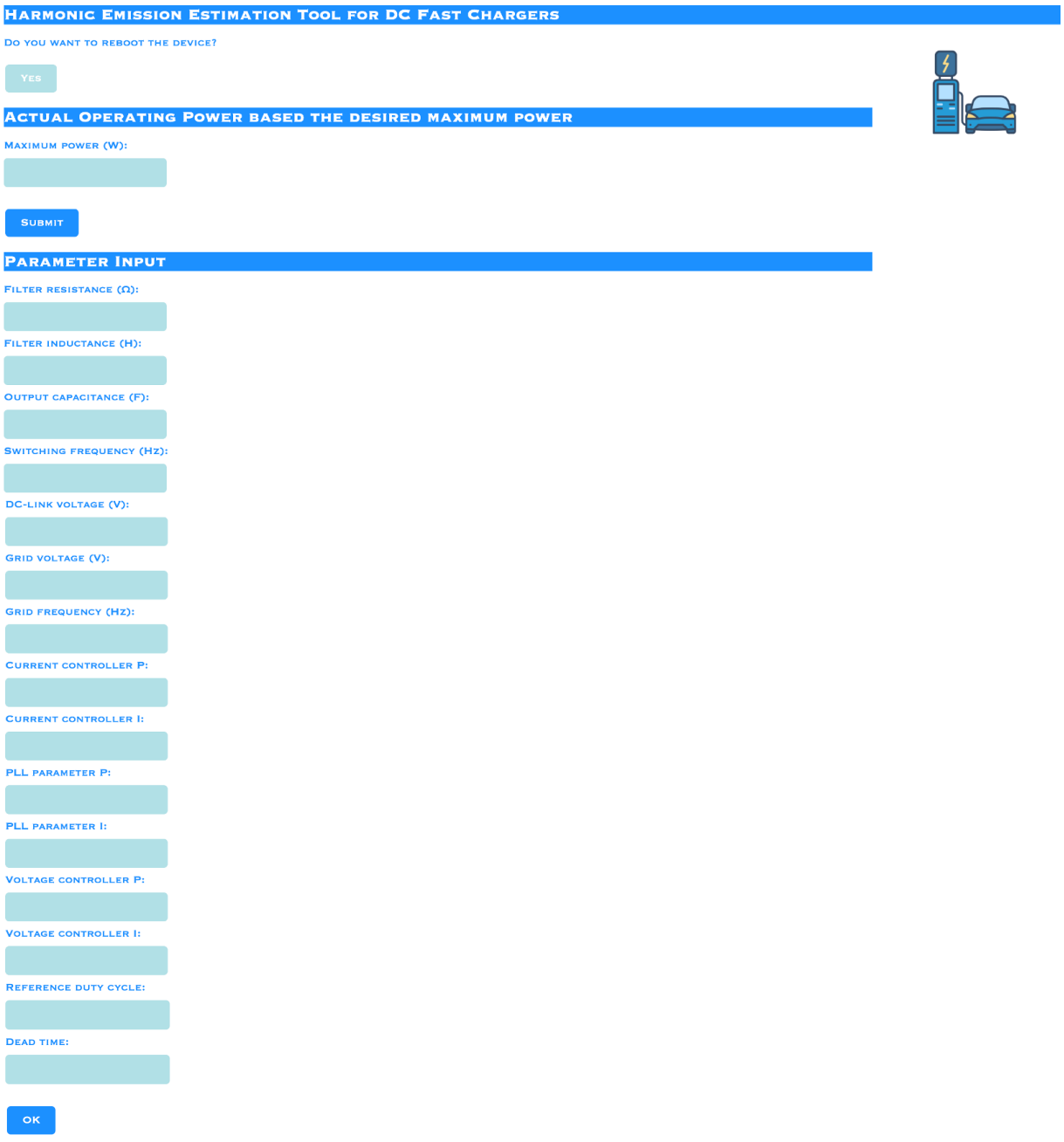


Figure 5.6: The screenshot of the webpage.

Table 5.1: Parameters for the input impedance calculation.

Parameters	value
filter resistance	$2 \times 10^{-2}\,\Omega$
filter inductance	$250 \times 10^{-6}\,\mathrm{H}$
output capacitance	$3000 \times 10^{-6}\,\mathrm{F}$
switching frequency	$20 \times 10^{3}\,\mathrm{Hz}$
DC charger power	$30 \times 10^{3}\,\mathrm{W}$
DC-link voltage	800 V
grid voltage	230 V
grid frequency	50 Hz
current controller P	1.256637
current controller I	100.530965
PLL parameter P	0.57950647
PLL parameter I	109.23439616
voltage controller P	6.148755
voltage controller I	254.664516

Figure 5.7: The screenshot of the web page response: actual power.

5.3 SYSTEM COST

As mentioned in section 1.1, the lower overall system cost is the advantage of the cloud-based solution. Therefore, it is necessary to evaluate the system cost of IoT platform.

The IoT platform in this thesis is a beta version that only uses the IoT hub service provided by Azure. The Azure IoT hub pricing is listed in Table. 5.2 [28]. The free edition of the Azure IoT hub is able to satisfy the requirement in this thesis since it enables to transmit up to 8000 messages per day with 0.5KB message meter size. Moreover, up to 500 device identities can be registered in the free tier IoT hub. Therefore, the free edition is chosen for this project. If higher message traffic flow or more device registration is required in the future, the pricing plan can be swapped to S1 or S2, S3 depending on platform requirements.

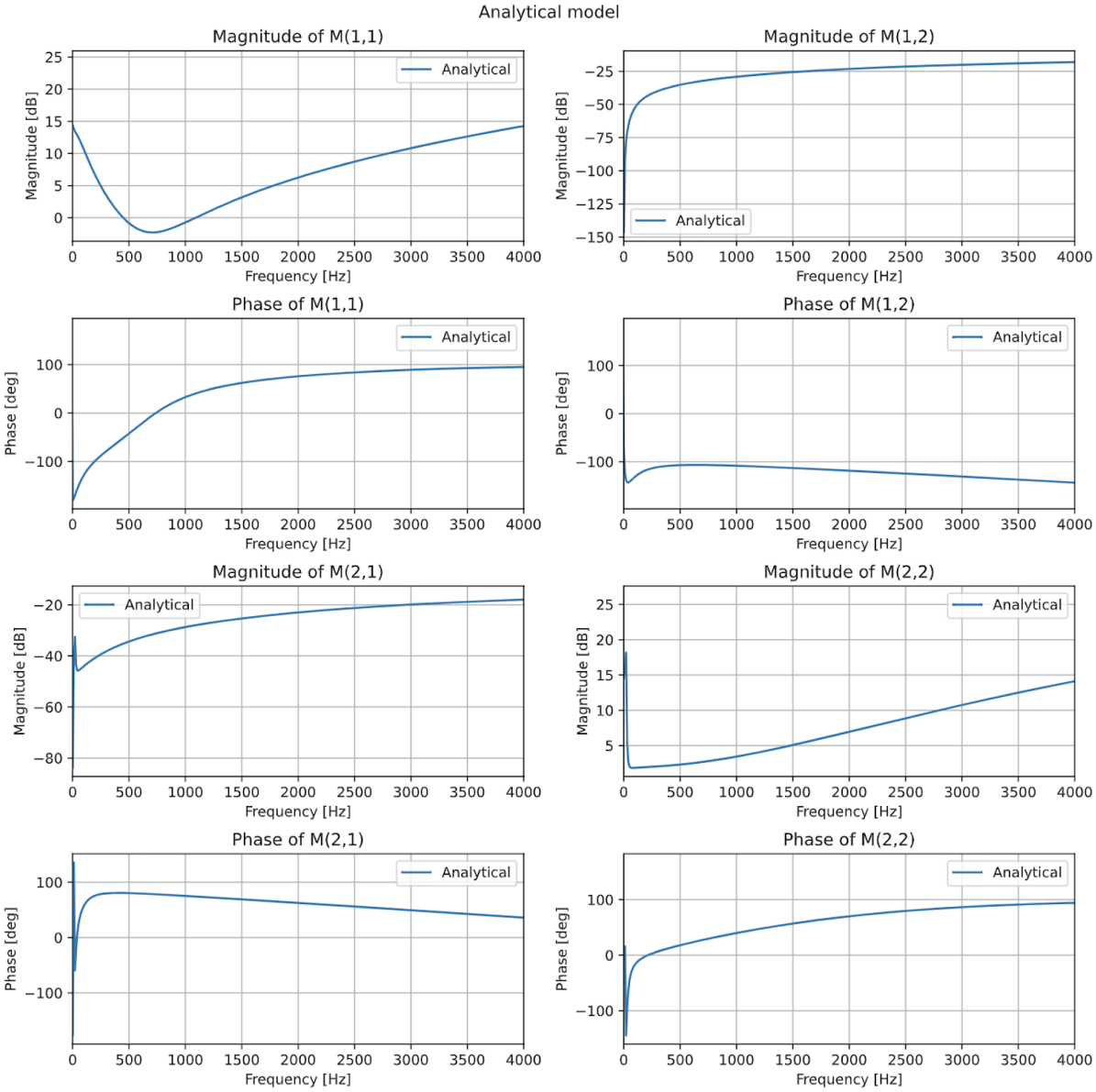


Figure 5.8: The screenshot of the web page response: input impedance curves.

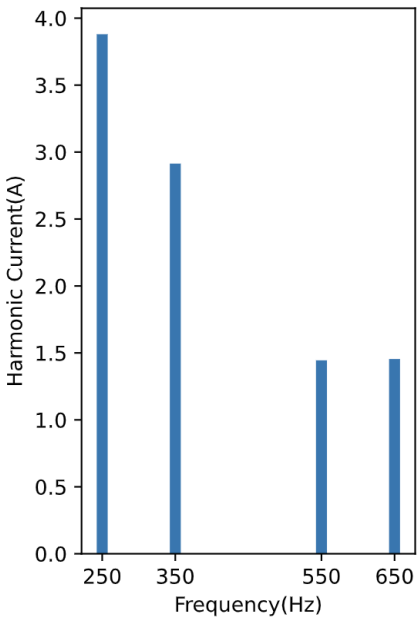


Figure 5.9: The screenshot of the web page response: current harmonics spectrum.

Table 5.2: Azure IoT hub pricing with standard tier.

Edition Type	**Price per IoT Hub unit (per month)**	**Total number of messages/day**	**Message meter size**
Free	Free	8,000	0.5KB
S1	€23.770	400,000	4KB
S2	€237.770	6,000,000	4KB
S3	€2,376.991	300,000,000	4KB

5.4 SUMMARY

The design and implementation of the IoT-based platform are elaborated in this chapter. First, the bi-directional communication between the DCFC and back-end app based on Microsoft Azure service is discussed. The web framework of the IoT platform is then presented. Finally, the system cost regarding the platform is given.

6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

An IoT based online harmonic emission estimation of DCFC is presented. The online platform can estimate the current harmonics of DCFCs under different grid conditions. The development of the IoT platform is divided into 3 parts: input impedance Z_c modeling of the AFE, current harmonic source I_c modeling of the AFE, and the design of the IoT based platform. The conclusions from each part can be drawn as below,

1. For the input impedance modeling of the DCFC elaborated in the Chapter 2, it can be seen that:
   - DCFC with AC distribution network is the mainstream solution for now which consists of two stages: AC/DC stage and DC/DC stage.
   - Considering the large DC-link capacitor between these two stages, the current harmonics emission is mainly determined by the AC/DC stage. In this thesis, the typical design of AC/DC stage, i.e., two-level AFE is chosen for input impedance modeling.
   - The influences of the PLL dynamics, the inner current control loop, the outer voltage control loop, and the fluctuation of DC-link voltage on the input impedance modeling are taken into account.

2. The current harmonic source modeling of the DCFC is discussed in Chapter 3:
   - The harmonic analysis based on the double Fourier series method has been under intensive investigation, whereas most of them focus on carrier and carrier-sideband harmonics.
   - To model the harmonic current source, the switch node voltage should be solved first so that the open-loop current harmonic can be obtained.

- The closed-loop harmonic is acquired based on the open-loop current harmonic and the control diagram of the AFE. Note that the frequency coupling effect should be taken into account.

3. For the design and implementation of the IoT based platform part in Chapter 5:
   - The bi-directional communication between the simulated DCFC app and the back-end app is realized via the service provided by Microsoft Azure.
   - The web framework of the IoT platform is built on the Django structure.
   - Since the platform in this thesis is a beta version, no system cost is required.

To conclude, a harmonic current emission model mainly considering low-frequency harmonics is proposed in this paper. The technique is based on the Fourier series method and the impedance model which is able to reveal the harmonic current emission of DCFCs under different grid conditions. The time-domain simulations and experiments are presented subsequently to validate the analytical model. Based on the proposed model, an IoT based platform for online harmonic emission estimation of DCFC is built.

6.2 FUTURE WORK

It is worth mentioning the future work which concerns a deeper analysis of the proposed model and the refinement of system functions:

- The solution for the time period without dead time effect has inaccuracy that affects the harmonic computation. As a result, it's vital to improve model accuracy.
- Due to the limit of existing experiment equipment, the experiment is conducted with grid impedance $L_g = 2.5\,\text{mH}$. The grid impedance in reality is relatively lower. Thus, experiments with lower grid impedance can be carried out in the future.
- For the IoT platform, the next step can be to combine the platform with the real DCFC. In addition, the function of charging power suggestion based on the DCFC parameters and the grid condition can be added.

BIBLIOGRAPHY

[1] L. Wang, Z. Qin, T. Slangen, P. Bauer, and T. van Wijk, "Grid impact of electric vehicle fast charging stations: Trends, standards, issues and mitigation measures - an overview," *IEEE Open Journal of Power Electronics*, vol. 2, pp. 56–74, 2021.

[2] J. Enslin and P. Heskes, "Harmonic interaction between a large number of distributed power inverters and the distribution network," *IEEE Transactions on Power Electronics*, vol. 19, no. 6, pp. 1586–1593, 2004.

[3] L. Beloqui Larumbe, Z. Qin, and P. Bauer, "Introduction to the analysis of harmonics and resonances in large offshore wind power plants," in *2018 IEEE 18th International Power Electronics and Motion Control Conference (PEMC)*, 2018, pp. 393–400.

[4] T. Thiringer and S. Haghbin, "Power quality issues of a battery fast charging station for a fully-electric public transport system in gothenburg city," *Batteries*, vol. 1, no. 1, pp. 22–33, 2015. [Online]. Available: https://www.mdpi.com/2313-0105/1/1/22

[5] Z. Qin, H. Wang, F. Blaabjerg, and P. C. Loh, "Investigation into the control methods to reduce the dc-link capacitor ripple current in a back-to-back converter," in *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2014, pp. 203–210.

[6] J.Shen, J.A.Taufiq, and A.D.Mansell., "Analytical solution to harmonic characteristics of traction pwm converters," *IEE Proceedings - Electric Power Applications*, vol. 144, no. 2, pp. 158–168, 1997.

[7] H. Deng, L. Helle, Y. Bo, and K. B. Larsen, "A general solution for theoretical harmonic components of carrier based pwm schemes," in *2009 Twenty-Fourth Annual IEEE Applied Power Electronics Conference and Exposition*, 2009, pp. 1698–1703.

[8] T. Xu, F. Gao, X. Wang, T. Hao, D. Yang, and F. Blaabjerg, "Closed-loop elimination of low-order sideband harmonics in parallel-connected low-pulse ratio vsis," in *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2018, pp. 2271–2277.

[9] H. Tu, H. Feng, S. Srdic, and S. Lukic, "Extreme fast charging of electric vehicles: A technology overview," *IEEE Transactions on Transportation Electrification*, vol. 5, no. 4, pp. 861–878, 2019.

[10] P. Chatterjee and M. Hermwille, "Tackling the challenges of electric vehicle fast charging," *ATZelectronics Worldwide*, vol. 15, no. 10, pp. 18–22, 2020.

[11] F. Blaabjerg, *Control of Power Electronic Converters and Systems Vol 1*, 02 2018.

[12] R. Lai, F. Wang, R. Burgos, D. Boroyevich, D. Jiang, and D. Zhang, "Average modeling and control design for vienna-type rectifiers considering the dc-link voltage balance," *IEEE Transactions on Power Electronics*, vol. 24, no. 11, pp. 2509–2522, 2009.

[13] J. Sun, "Impedance-based stability criterion for grid-connected inverters," *IEEE Transactions on Power Electronics*, vol. 26, no. 11, pp. 3075–3078, 2011.

[14] B. Wen, D. Boroyevich, R. Burgos, P. Mattavelli, and Z. Shen, "Analysis of d-q small-signal impedance of grid-tied inverters," *IEEE Transactions on Power Electronics*, vol. 31, no. 1, pp. 675–687, 2016.

[15] J. Sun, "Small-signal methods for ac distributed power systems–a review," *IEEE Transactions on Power Electronics*, vol. 24, no. 11, pp. 2545–2554, 2009.

[16] X. Wang, L. Harnefors, and F. Blaabjerg, "Unified impedance model of grid-connected voltage-source converters," *IEEE Transactions on Power Electronics*, vol. 33, no. 2, pp. 1775–1787, 2018.

[17] L. Harnefors, "Modeling of three-phase dynamic systems using complex transfer functions and transfer matrices," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 2239–2248, 2007.

[18] S.-K. Chung, "A phase tracking system for three phase utility interface inverters," *IEEE Transactions on Power Electronics*, vol. 15, no. 3, pp. 431–438, 2000.

[19] K. Martin, "Complex signal processing is not complex," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 9, pp. 1823–1836, 2004.

[20] A. Rygg, M. Molinas, C. Zhang, and X. Cai, "A modified sequence-domain impedance definition and its equivalence to the dq-domain impedance definition for the stability analysis of ac power electronic systems," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 4, no. 4, pp. 1383–1396, 2016.

[21] P. S. Harmonics, "Power system harmonics: An overview," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-102, no. 8, pp. 2455–2460, 1983.

[22] D. Holmes and T. Lipo, *Pulse Width Modulation for Power Converters: Principles and Practice*, ser. IEEE Press Series on Power and Energy Systems. Wiley, 2003. [Online]. Available: https://books.google.nl/books?id=8LGi1AjSfpcC

[23] N. Jiao, S. Wang, T. Liu, Y. Wang, and Z. Chen, "Harmonic quantitative analysis for dead-time effects in spwm inverters," *IEEE Access*, vol. 7, pp. 43 143–43 152, 2019.

[24] Y. Yang, K. Zhou, H. Wang, and F. Blaabjerg, "Analysis and mitigation of dead-time harmonics in the single-phase full-bridge pwm converter with repetitive controllers," *IEEE Transactions on Industry Applications*, vol. 54, no. 5, pp. 5343–5354, 2018.

[25] J. Ançay. (2021) How to build a 3 phase inverter. [Online]. Available: https://imperix.com/doc/help/how-to-build-a-3-phase-inverter

[26] Microsoft. (2022) Azure IoT Overview. [Online]. Available: https://azure.microsoft.com/en-us/overview/iot/#overview

[27] ——. (2022) Get started with device management (Python). [Online]. Available: https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-python-python-device-management-get-started

[28] ——. (2022) Azure IoT hub pricing. [Online]. Available: https://azure.microsoft.com/en-us/pricing/details/iot-hub/

A

PYTHON CODE FOR CLOUD-BASED COMMUNICATION

A.1 PYTHON CODE FOR THE SIMULATED DC FAST CHARGER APP

```
import time
import datetime
from azure.iot.device import IoTHubDeviceClient, MethodResponse

RECEIVED_MESSAGES = 0

CONNECTION_STRING = "HostName=IoThubYL.azure-devices.net;DeviceId=myDeviceId;
SharedAccessKey=4V5PudifE5BPRlrAoUTQsRTzVVUEuTK1S0qUEDIRinc="
def create_client():
    # Instantiate the client
    client = IoTHubDeviceClient.create_from_connection_string(CONNECTION_STRING)

    # Define the handler for method requests
    def method_request_handler(method_request):
        if method_request.name == "PowerCalculation":
            global RECEIVED_MESSAGES
            RECEIVED_MESSAGES += 1
            print("Message received")
            Receivepower=method_request.payload
            ReceivepowerNumber=int(Receivepower)
            print('Receive power', ReceivepowerNumber)
            ActualPower=ReceivepowerNumber*0.8
            print('Actual power', str(ActualPower))

            # Create a method response indicating the method request was resolved
```

```
        resp˙status = 200
        resp˙payload = –'Actual power': str(ActualPower)″
        #resp˙payload = –"Actual Power": 'Test'″
        method˙response = MethodResponse(method˙request.request˙id, resp˙status, resp˙payload)


    elif method˙request.name == "reboot":
        # Act on the method by rebooting the device
        print("Rebooting device")
        time.sleep(20)
        print("Device rebooted")

        # ...and patching the reported properties
        current˙time = str(datetime.datetime.now())
        reported˙props = –"rebootTime": current˙time″
        client.patch˙twin˙reported˙properties(reported˙props)
        print( "Device twins updated with latest rebootTime")

        # Create a method response indicating the method request was resolved
        resp˙status = 200
        resp˙payload = –"Device Response": "Reboot Successful"″
        method˙response = MethodResponse(method˙request.request˙id, resp˙status, resp˙payload)

    else:
        # Create a method response indicating the method request was for an unknown method
        resp˙status = 404
        resp˙payload = –"Response": "Unknown method222"″
        method˙response = MethodResponse(method˙request.request˙id, resp˙status, resp˙payload)

    # Send the method response
    client.send˙method˙response(method˙response)

try:
    # Attach the handler to the client
    client.on˙method˙request˙received = method˙request˙handler
except:
    # In the event of failure, clean up
    client.shutdown()
```

```
    return client


def main():
    print ("Starting the IoT Hub Python sample...")
    client = create_client()

    print ("Waiting for commands, press Ctrl-C to exit")
    try:
        # Wait for program exit
        while True:
            time.sleep(1000)
    except KeyboardInterrupt:
        print("IoTHubDeviceClient sample stopped")
    finally:
        # Graceful exit
        print("Shutting down IoT Hub Client")
        client.shutdown()

if __name__ == '__main__':
    main()
```

A.2 PYTHON CODE FOR THE BACKEND SERVICE APP

```
import sys,time

from azure.iot.hub import IoTHubRegistryManager
from azure.iot.hub.models import CloudToDeviceMethod, CloudToDeviceMethodResult, Twin
CONNECTION_STRING = "HostName=IoThubYL.azure-devices.net;SharedAccessKeyName=service;SharedA
DEVICE_ID = "myDeviceId"

TIMEOUT = 60
WAIT_COUNT = 5
MESSAGE_COUNT = 1

def iothub_reboot():
```

```
try:
    # Create IoTHubRegistryManager
    registry_manager = IoTHubRegistryManager(CONNECTION_STRING)

    print ( "" )
    print ( "Invoking device to reboot..." )

    METHOD_PAYLOAD = "-""method_number"":""42"" ""
    # Call the direct method.
    deviceMethod = CloudToDeviceMethod(method_name="reboot", payload=METHOD_PAYLOAD)
    response = registry_manager.invoke_device_method(DEVICE_ID, deviceMethod)

    print ( "" )
    print ( "Successfully invoked the device to reboot." )

    print ( "" )
    print ( response.payload)

    #while True:
    print ( "" )
    print ( "IoTHubClient waiting for commands, press Ctrl-C to exit" )

    status_counter = 0
    while status_counter ¡= WAIT_COUNT:
        twin_info = registry_manager.get_twin(DEVICE_ID)

        if twin_info.properties.reported.get("rebootTime") != None :
            print ("Last reboot time: " + twin_info.properties.reported.get("rebootTime"))
        else:
            print ("Waiting for device to report last reboot time...")

        time.sleep(1)
        status_counter += 1
    return (response.payload, "Last reboot time: " + twin_info.properties.reported.get("rebootTime"))

except Exception as ex:
    print ( "" )
    print ( "Unexpected error -0"".format(ex) )
```

```
        response˙error="Unexpected error –0""
        return response˙error


    except KeyboardInterrupt:
        print ( "" )
        print ( "IoTHubDeviceMethod sample stopped" )


def iothub˙devicemethod˙sample˙run(maximumpower):
    try:
        # Create IoTHubRegistryManager
        registry˙manager = IoTHubRegistryManager(CONNECTION˙STRING)

        print ( "" )
        print ( "Sending message to the deivce..." )

        METHOD˙PAYLOAD = maximumpower
        # Call the direct method.
        deviceMethod = CloudToDeviceMethod(method˙name="PowerCalculation", payload=METHOD˙PAYLO
        response = registry˙manager.invoke˙device˙method(DEVICE˙ID, deviceMethod)

        print ( "" )
        print ( "Successfully send message." )
        time.sleep(5)
        print ( "" )
        print ( response.payload)

        #while True:
        print ( "" )
        print ( "IoTHubClient waiting for commands, press Ctrl-C to exit" )
        return response.payload

    except Exception as ex:
        print ( "" )
        print ( "Unexpected error –0"".format(ex) )
        response˙error="Unexpected error –0""
        return response˙error
```

```
    except KeyboardInterrupt:
        print ( "" )
        print ( "IoTHubDeviceMethod sample stopped" )

if __name__ == '__main__':
    print ( "Starting the IoT Hub Service Client DeviceManagement Python sample..." )
    print ( "    Connection string = {0}".format(CONNECTION_STRING) )
    print ( "    Device ID         = {0}".format(DEVICE_ID) )

    iothub_devicemethod_sample_run()
```