

MATLAB-Arduino as a Low Cost Microcontroller for 3 phase inverter

Shamsul Aizam Zulkifli
Department of Electrical Power Engineering
FKKEE, Universiti Tun Hussein Onn Malaysia
Batu Pahat, Johor, Malaysia

Mohd Najib Hussin, Abdul Salam Saad
Department of Electrical and Electronics Engineering
Politeknik Kuala Terengganu,
Kuala Terengganu, Terengganu, Malaysia

Abstract— This paper presents the performance of the Arduino microcontroller board in response to an advanced control strategy for a motor control application. Two set of control theories have been used and combined as known as the Proportional Integration (PI) - Repetitive controller. This controller has been designed in MATLAB-Simulink environment and then downloaded to the Arduino for testing. The test set consists of 3 phase inverter, filter, induction machine (IM) and a set of current controller. The feedback current from the IM will be the control parameter for the controller and the Arduino in order to generate the pulse width modulation (PWM) pattern. Different PWM pattern is generated when the feedback current is changing due to the changing of the load.

Keywords—repetitive control, arduino, proportional integral control, Matlab

I. INTRODUCTION

Before the development of power electronics devices, induction motors are mainly used in fixed speed applications. It is because of its robustness, reliable, low cost, high efficiency and durable [1, 2] system. The IM can be controlled in two modes, speed control and load control. For speed control there are several techniques such as voltage control, frequency control [3], changing the number of stator poles, controlling the supply voltage and others. As for the current control, the Field Oriented Control (FOC) and Direct Torque Control (DTC) with the conventional control strategy can be applied. More advanced controller also can be developed but it needs more input parameters such as the motor flux, motor speed, motor position and load current to be considered [4].

The voltage and current applied to the motor can be changed by employing the power electronic converters [6] such as a inverter. It can change the inverter output voltage depends to the different pattern of PWM that been generated by the controller. In order to have better PWM signal, a combination of two controllers can be applied such as the PI-Fuzzy controller, the PI-Time Delay controller, the PI-Neural Network or others.

The PWM pattern can be generated by using the Digital Signal Processing or microcontroller board. As stated

in [7], Arduino is one of the low cost microcontrollers. It has 6 inputs outputs connectivity where it used ATmega328 processor with the clock speed of 16 Mhz. The limitation of the Arduino is due to the switching frequency is less then 1Khz.

Arduino needs a C language to operate. As known the MATLAB is able to generate this language and then downloaded to the microcontroller board. Another advantages of MATLAB is, the library for the Arduino is available in MATLAB. This library will help the beginner to understand the MATLAB and Arduino more quickly. Due to this advantage, the control theory which is based to mathematical formula can be developed more easily in MATLAB. This has helped the author to apply this technique to this project.

In this paper, a 3-phase inverter which is connected to a 3 phase 0.75kW induction motor has been developed as shown in Fig.1. The PI+Repetitive controller is developed to control the PWM pattern which is depend to the current feedback. The concept of repetitive control was first introduced in [5] for motor control technique. This controller has been proposed due to high performances in different areas such as, the PI is used in the steady state response due to ability to gives a zero error while the closed-loop hybridization of repetitive is able to track or reject unknown periodic signals of a known period [8]. For experimental analysis, Matlab-Simulink Ver.2013a has been used as a medium of controller designing between MATLAB, Aduino and the inverter.

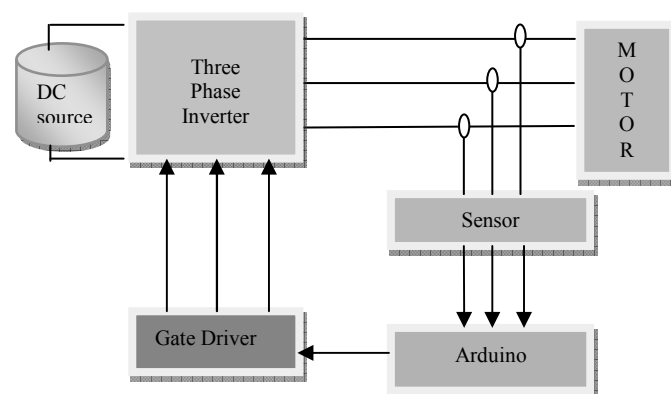


Figure1. The proposed block diagram

II. THE PROPOSED CONTROLLER

Proportional term of output (P_{out}) and Integral term of outputs (I_{out}) for a PI controller are the contributor at the output. The proportional term (sometimes called gain) makes a change to the output which is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant K_p . The proportional term is given in (1):

$$P_{out} = K_p e(t) \quad (1)$$

In the disturbance condition, pure proportional control is not able to follow the target value, but it will retain a steady state error. This error target can be reduced by implementation of Integral (I) function by the integral gain K_i . Equation (2) shows the PI control strategy when the P and I have been combined.

$$PI = K_p e(\tau) * K_i \int_0^t e(\tau) d\tau \quad (2)$$

PI controller is good for tracking or rejecting step signals but for inverter input control, the error signal is in sinusoidal form. In order to have good tracking performance, a repetitive control system should be added where it able to eliminate the harmonic signal [11] from the current feedback. The basic Repetitive controller transfer function is shown in (3).

$$\frac{V}{v} = \frac{\omega}{s^2 + \omega^2} \quad (3)$$

where ω is the resonant frequency which is equal to the synchronous frequency of the stator voltage outputs [9, 10]. The general Repetitive control block diagram is shown in Fig.2.

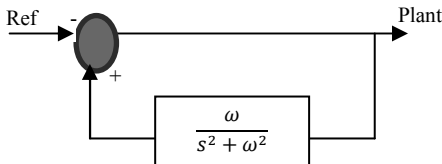


Fig.2. Feedback loop for repetitive controller

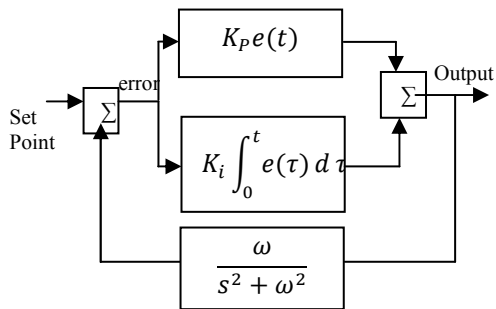


Fig.3. PI-Repetitive controller proposed arrangement

Repetitive control is said to be among the best way in order to make the system stable because of the application of 'close -loop' [12] task. The use of memory-loop is beneficial in systems with repetitive disturbances or tasks.

III. CONTROLLER STABILITY

For the stability test of the PI-resonant controller, the step response test of the plant has been used with the suggested controller.

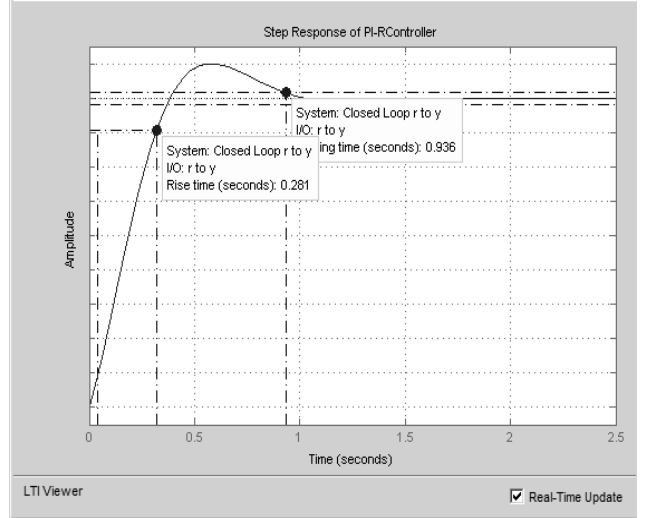


Fig.4. Step response of PI-Repetitive controller

Figs. 4 and 5 show the step response of PI-Repetitive and the root locus of the controller. These results can be collected by using the 'sisotool' command that is available in Matlab. For the step response analysis it shows that, the system is able to track the step signal given to the plant the rise time is about 0.2 second for step input. For the root locus diagram shows, all the poles are located on the zero axis which is the system is marginally stable.

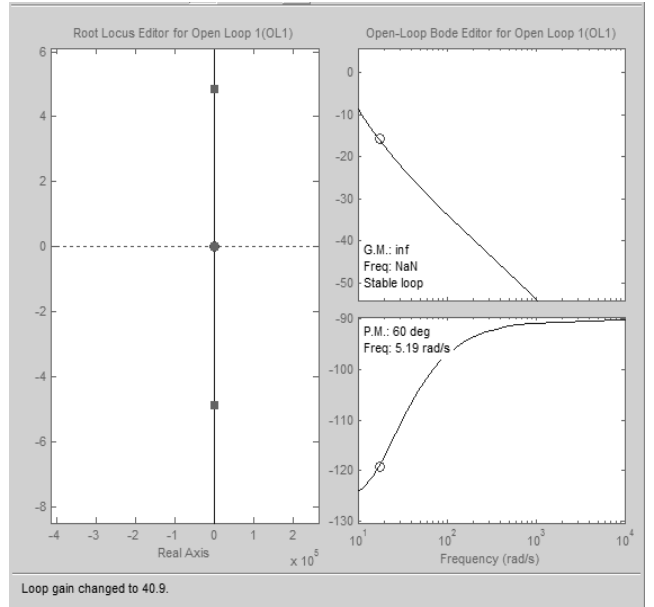


Fig.5. Root Locus and Bode diagram of PI-Repetitive Control

After the stability of the controller has been determined, the next step to model the control block in

MATLAB software. Fig.6 shows the controller model which has been designed in Matlab-Simulink software where is based from Fig. 3. The inputs to the controller are coming from the target references which are the current set point and the current feedback from the motor. In hardware implementation the current sensor will be used with the Analog to Digital (ADC) converter or vice versa that have been developed in the MATLAB. This is the advantages of this project where the ADC and the controller diagram can be developed based on the mathematical equation compared by writing the program in C code.

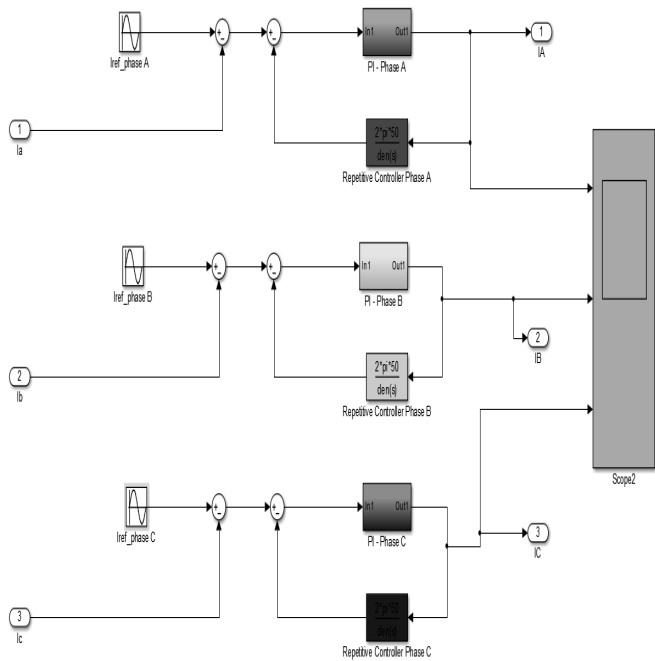


Fig.6. Proportional Integral Repetitive controller In Matlab-Simulink environment

IV. SIMULATION AND EXPERIMENTAL TEST FOR THE PROPOSED CONTROLLER

Firstly, the proposed controlled has been tested in simulation environment in order to see the output voltage from the inverter and also current output. It also a better technique for determine the best controller gain for the K_p and K_i . Fig.8 shows the whole simulation system of the project which consists of four main blocks which are the controller block, the PWM block, the 3-phase inverter and the filter. It can be seen that, the induction motor has been changed to the resistive load for easy understanding.

Fig. 9 and 10 show the output from the inverter. Fig. 9 shows the output voltage which is about 20Vpp when the DC voltage is set to 10V. By changing the DC supply the output o the inverter also will be changed. Fig.10 shows the current at the load which is about 0.2A which is depend to the load size.

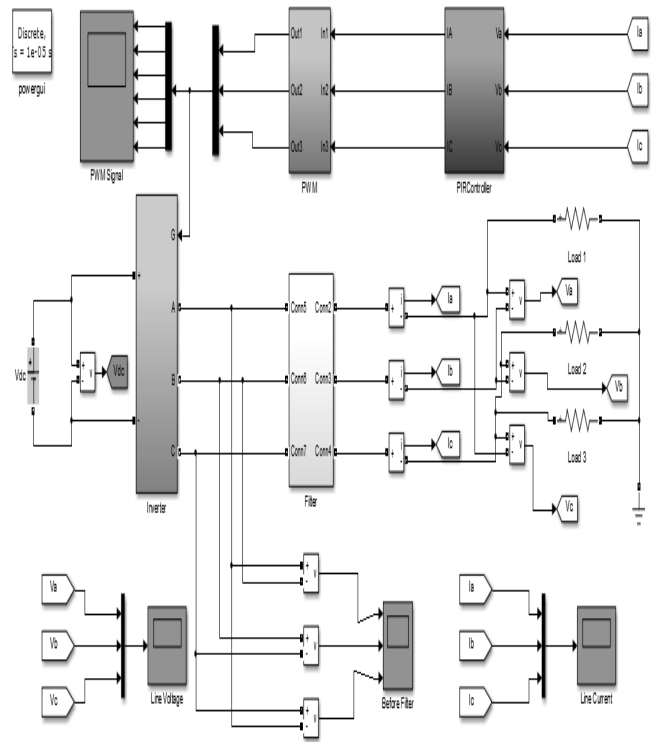


Fig.8. Overall simulation diagram develop in the Matlab

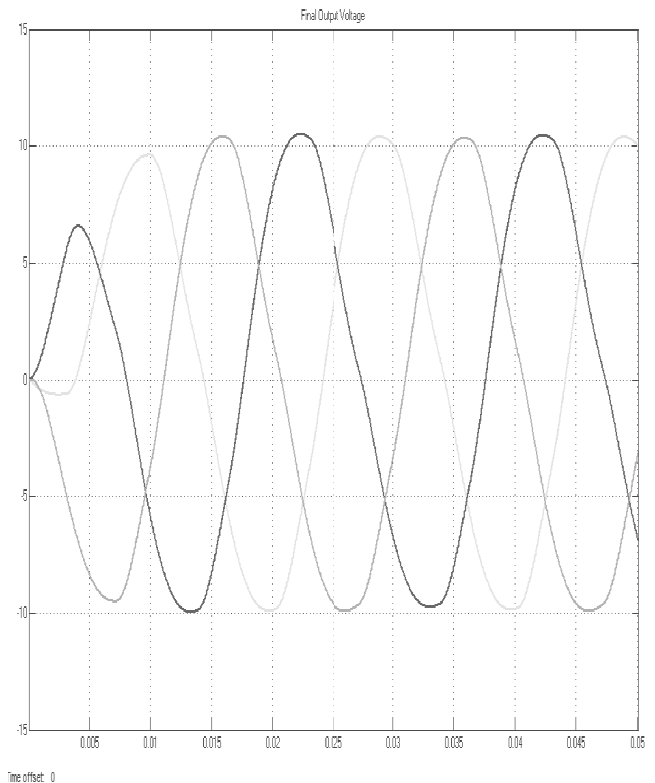


Fig.9. The output voltage of the PI-Repetitive-controller

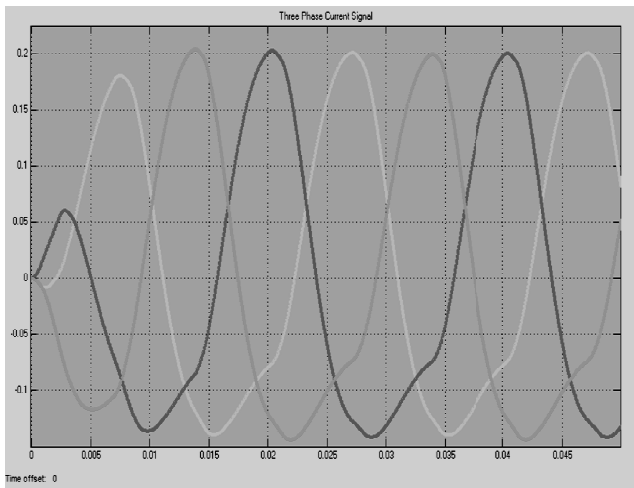


Fig.10. The output current of the PI-Repetitive- controller

The hardware testing has been conducted after the suitable K_p and K_i have been determined. Fig.11 shows the experiment set-up that has been developed in the lab. There is a small modification in experimental set-up, where the sensor current is different from the simulation. This is because, the current sensor that has been used here has a bigger sensitivity that will affect the range of output voltage from the sensor. The parameters for the experiment are given in Table 1.

Table 1: Experiment's parameters

No	Parameters	Value
1	Input DC Voltage	30 volt
2	PWM Frequency Switching in Matlab	5kHz
3	Target Voltage	(10 – 30) volt \pm 1

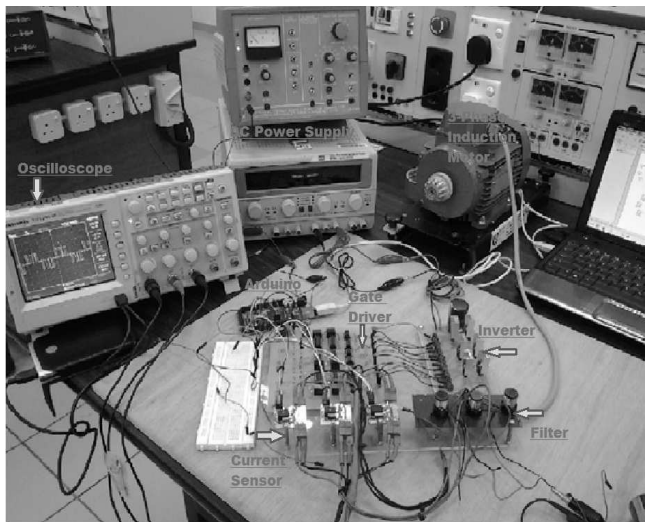


Fig.11. Experimental setup

Fig 12 shows the output voltage after the filter at phase a when 20Vdc is applied to the inverter. It shows, the output after the inverter is non pure sinusoidal waveform but

has frequency of 50Hz. The voltage magnitude generate from the hardware is only 10V that is caused by the wrong design of the filter size. It because the filter was designed using coil arrangement which that effect the cut of frequency of the filter. The filter problem will be solved in the next paper.

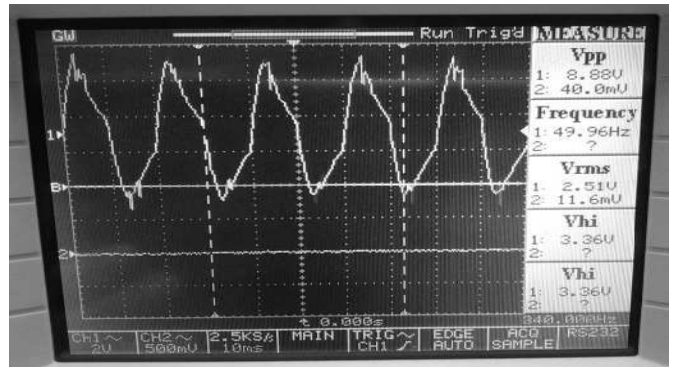


Fig.12. The inverter output voltage in the hardware

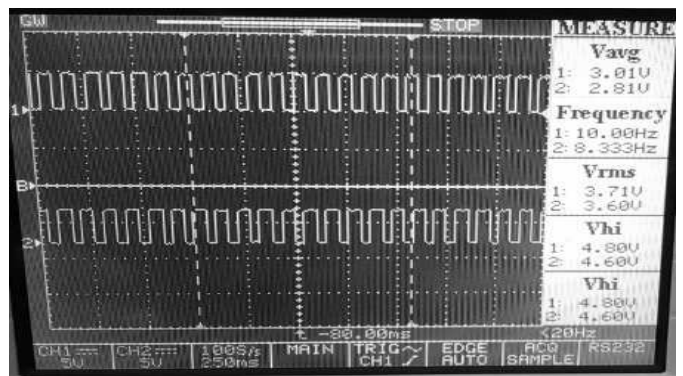


Fig. 13. The PWM signal for input voltage 30V

Fig.13 shows the first pattern of the PWM signal when the input current sensor/feedback current was measured at 0.8A. It then been converted to the suitable dc voltage by the current sensor for Arduino process when the designed controller has been downloaded to the board. At the end the Arduino will generate this PWM pattern that corresponding to the input.

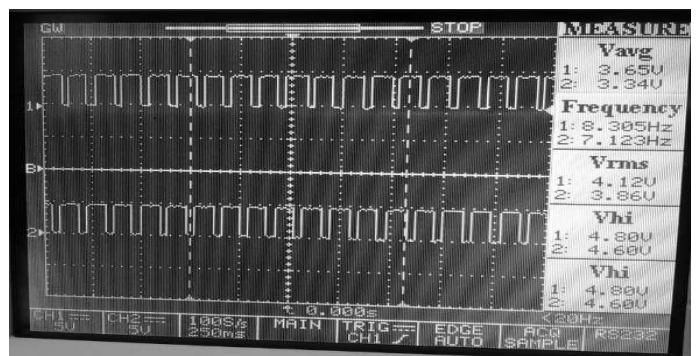


Fig.14. The PWM signal for input voltage 20V

Fig.14 shows the different pattern of PWM signals when the current sensor was measured at 1.0A and used the same procedure has been described before. Figs. 13 and 14

show the Arduino is able to change the PWM pattern based on the current sensor input and also with the proposed controller

V. CONCLUSION

As a conclusion, by using the low cost microcontroller board such as Arduino, the output from the inverter can be controlled. It also shows that the Arduino is able to use more advanced controller which is the PI-Repetitive controller that been developed to generate different pattern of PWM signals. It shows that, the MATLAB is a best candidate to be applied with the Arduino, when the designer is only need to determine the mathematical formulation compared to writing the C programming. This technique will help the beginner or student to use a low cost microcontroller for electrical drive application for motor control.

REREFENCES

- [1] P. Farmanzadeh, M. R. Azizian, and A. Ebrahimi, "A robust controller design to stabilize induction motor drive operation using (IFOC) method," in *Power Electronics and Drive Systems Technology (PEDSTC)*, 2012 3rd, 2012, pp. 218-222.
- [2] B. A. a. N. Garg, "Scalar (V/f) Control of 3-Phase Induction Motors," Available in: <http://www.ti.com/lit/an/sprabq8/sprabq8.pdf>
- [3] R. Echavarria, S. Horta, and M. Oliver, "A three phase motor drive using IGBTs and constant V/F speed control with slip regulation," in *Technical Proceedings Power Electronics Congress, 1995.. CIEP 95., IV IEEE International*, 1995, pp. 87-91.
- [4] P. Yedamale, "Speed Control of 3-Phase Induction Motor Using PIC18 Microcontrollers," *Application Note (AN843)*, Microchip Technology Inc, 2002.
- [5] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems," in *The 23rd IEEE Conference on Decision and Control*, 1984, pp. 1064-1069.
- [6] M. Electric, "Introduction To Electrical Drive." [Online]. Available:[ftp://ftp.dei.polimi.it/outgoing/Massimo.Ghioni/Power Electronics /Motor control/motor control overview/INTRODUCTION TO ELECTRICAL DRIVES.pdf](ftp://ftp.dei.polimi.it/outgoing/Massimo.Ghioni/Power%20Electronics/Motor%20control/motor%20control%20overview/INTRODUCTION%20TO%20ELECTRICAL%20DRIVES.pdf).
- [7] Z. Othman, "High-Efficiency Dual-Axis Solar Tracking Development using Arduino," pp. 43-47, 2013.
- [8] T.Hornik and Q.-C. Zhong, " H_∞ repetitive current controller for grid-connected inverters," in *35th Annual Conference of IEEE Industrial Electronics, 2009. IECON'09*, pp. 554-559.
- [9] J.-b. HU, Y.-k. HE, H.-s. WANG, and X. LIE, "Proportional-resonant Current Control Scheme for Rotor-side Converter of Doubly-fed Induction Generators Under Unbalanced Network Voltage Conditions [J]," *Proceedings of the CSEE*, vol. 6, p. 009, 2010.
- [10] A. Junyent-Ferre, O. Gomis-Bellmunt, T. C. Green, and D. E. Soto-Sanchez, "Current control reference calculation issues for the operation of renewable source grid interface VSCs under unbalanced voltage sags," *IEEE Transactions on Power Electronics*, , vol. 26, pp. 3744-3753, 2011.
- [11] T. Hornik and Q.-C. Zhong, " H_∞ repetitive voltage control of grid-connected inverters with a frequency adaptive mechanism," *IET power electronics*, vol. 3, pp. 925-935, 2010.
- [12] S. Hara, T. Omata, and M. Nakano, "Synthesis of repetitive control systems and its application," in *24th IEEE Conference on Decision and Control*, 1985 pp. 1387-1392
- [13] R. Teodorescu, F. Blaabjerg, M. Liserre, and P. C. Loh, "Proportional-resonant controllers and filters for grid-connected voltage-source converters," *IEE Proceedings-Electric Power Applications*, vol. 153, pp. 750-762, 2006.