Abstract—This paper explains, the implementation of MATLAB-simulink block diagram uses in Arduino in order to control the output current of the 3-phase inverter for ac motor drive. There is no programming code has been involved but only uses the target preference blocks that are available in the MATLAB-Arduino library. The current controller has been developed by implementing the PID control in order to determine the efficiency of the controller to control the motor. The system has been tested on the 1kW inverter output and connected to the 3 phase induction motor with rating of 375W.

Index Terms—MATLAB, Simulink, Arduino, PID

I. INTRODUCTION

As known, ac motor is widely used to educate non electrical engineering students to understand the concepts of energy theory such as, by changing the electrical energy to the mechanical energy which applies to the motor. This energy output needs to be controller in order to protect the motor. In doing so, the controller needs to be developed which response to the motor load. The common strategy is by combining the microcontroller or Digital Signal Processing (DSP) device to an ac motor. As stated, microcontroller is a device that able to generate mili-seconds response that needs for the motor control. The Arduino, Rassbery PI, ezdsp-TI board or D-space are the examples of the DSP platform. Generally, the TI ezdsp and D-space are applied for high level application such as in power system application or in high precision application, meanwhile the Arduino or Raspberry PI is for low level application. At the meantime, the advancement and a new simple structure of current microcontroller has attracted more and more non technical expert to use and applied especially for Arduino or Rassbery PI.

Today, Arduino capable to communicate with the MATLAB [1][2][3] by using the C programming that is the platform of the MATLAB software. Until this paper has been written, the MATLAB-Simulink for Arduino target has never been tested with the power converter devices such as the inverter but in [4] mentioned Arduino is capable to control the induction motor or in dc motor [2]. Due to this finding, this paper is focusing to test a simple controller strategy that has been developed using the MATLAB-Simulink library and been downloaded to the Arduino microcontroller in order to see the performance of the Arduino with the inverter using MATLAB-Simulink based library. Due to this reason, a simple control method for an ac motor controller is applied to the converter.

As known, many researchers have come out with various controllers in order to control the ac motor, such as a speed, a voltage and a current controller. From the control parameters of the ac motor, the current input is important to determine the smoothness rotation of the ac motor. This current regulation is controlled by controlling the pulse-width-modulated (PWM) that will be developed in the microcontroller device [2][3][5][6].

The type of controllers that can be applied to a ac motor can be categorized into two, which are the active and passive controller [7][8][9]. The active controller is a controller that responds to the reference target such as Field Oriented Control (FOC), Direct Torque Control (DTC), Proportional Integral Derivatives (PID), Fuzzy control, neural network, Sliding Mode Control (SMC), and Hysteresis control. For the passive control, it is based on the time control system response.

These controllers can be designed using programming code or by modeling the controller based on the mathematical equation of the controller. The comparison between both techniques are where the first technique requires an expert to understand the program flow, meanwhile the mathematical based is the easiest technique, where the controller is developed based on the theory. Due to this advantage, the MATLAB which is the mathematical language can be applied [1] for non technical programmer expert and it will be applied in this paper.

In this paper, the controller which is the combination of the current control with the PID controller model has been developed in the MATLAB-Simulink and will be tested in the experimental setup by using the Arduino, the 3-phase inverter and the ac motor in order to see the current output response. As a result, the performance characteristics will be observed in terms of output motor line current, the switching algorithm from the Arduino and the inverter output voltage.

II. THE PROPOSED TOPOLOGY

Figure 1 shows the block diagram of this project. It consists of 6 main parts which are the DC source input, the gate driver circuit, the three phase inverter, the 375W three phase ac motor
as a load, the current sensor and the Arduino target board. The Arduino is used to test the controller that has been developed in the MATLAB-Simulink software.

The 3-phase inverter was designed to produce 1kW output that can support 450Vdc input as the source. Here, the inverter is known as voltage source inverter where it is the standard used inverter nowadays and it is easy to control [10]. The gate driver is needed in order to supply a suitable voltage after the PWM operation for triggering the MOSFET.

![Diagram of the proposed topology](image)

Figure 1: The proposed topology

**A. Development of the controller structure**

The PID controller is the most popular and the most commonly used in industrial recently. The popularity and widespread uses of PID controller contribute to it simplicity and performance characteristics. Although linear fixed-gain PID controller is often adequate for controlling a nominal physical process, the requirements for high-performance control with changes in operating conditions or environmental parameters are often beyond the capabilities of simple PID controller [11]. Figure 2 shows the PID controller block diagram. In this project the PID with current control is used in order to generate the PWM signal. The general equation of PID control is given in Eq. 1

\[ u(t) = PID(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{d}{dt} e(t) \]  

(Eq.1)

Where,

- \( K_p \) = proportional gain,
- \( K_i \) = integral gain and
- \( K_d \) = derivative gain

For this project, the value of \( K_p \), \( K_i \) and \( K_d \) have been selected base on the try and error method. It is when several values have been tested in the simulation environment before been applied to the hardware. In more precise value, there are several techniques that can be used to find the optimal value for the gains such as Ziegler-Nickolas, bode diagram method or pole-placement method.

![PID controller diagram](image)

Figure 2: PID controller

**B. Analog to Digital converter in Arduino**

The function of Analog to Digital (ADC) or Digital to Analog (DAC) is used to change the input signal from the analog signal to the digital signal or vice versa. These blocks are available in the Matlab-Simulink and these models can be embedded in the Arduino target installer. The Arduino will automatically convert the input signal to the 10 bit digital format as the digital input. In MATLAB environment, it is necessary to use the DAC equation for converting the digital signal to the analog display that is required for the control process of the current controller. The DAC equation that will be applied to the Simulink block can be determined based on the calculation given in Figure 3.

![DAC equation graph](image)

Figure 3: The graph to create the DAC function

The DAC block function equation is as stated below: From the slope’s, the formula is

\[ y = mx + c \]  

(Eq.2)

Where, \( m \) = slope, \( c \) = y-intersect
\[ m = \left( \frac{y_2 - y_1}{x_2 - x_1} \right) \quad \text{(Eq.3)} \]

\[ m = \left( \frac{50 + 50}{1023} \right) = 0.09775171 \]

Finally, the function for DAC block is:

\[ y = 0.09775171(x) - 50 \quad \text{(Eq.4)} \]

Figure 4 shows the ADC, DAC and proposed controller block that have been built in MATLAB-Simulink.

C. The experimental setup

The hardware or experimental setup was developed as shown in Figure 6 at Universiti Teknologi Mara, Dungun, Terengganu. Figure 6 shows the three phase induction motor has been selected as a test load. The main parts of this experimental setup are the Arduino, the gate driver and the 3 phase inverter that have been modeled using Proteus and changed to 2 layer printed circuit board. Since the PWM output from the Arduino is 5V, the gate driver is needed to increase this voltage before being connected to the 3 phase inverter.

The power transistor (MOSFET SPP11N60C3) and capacitor are act as voltage source inverter or full bridge inverter. It is where the power transistor is capable to handle the maximum voltage up to 600V.

III. Simulation verification

For the simulation purpose, the three phase ac motor is replaced with the 1Ω resistor and 5mH inductor. The simulation has been conducted in MATLAB 2013 V(b) with 1e-6 discrete time sampling.
The output voltage can be seen from Figure 7 after the inverter receives the switching pattern. The amplitude of the output voltage is about 20 Vp. The three signals represent the line to line voltage at \( V_{AB} \), \( V_{BC} \), and \( V_{CA} \) but in the square waveform. This output generates when 20Vdc has been supplied to the source. To have smooth sine waveform, the filter is added on every phase of the inverter where it be not discussed in this paper. The improve three phase AC sine waveform after the filter is shown in Figure 8. For these signals, the voltage magnitude is dropped rapidly due the not properly designed of the filter.

Figure 8: Inverter output at inverter after filter

Figure 9 shows the response of the feedback current from the load with the target reference current in simulation environment. It shows that, the feedback current (green) follows the reference current (red). It indicates that the controller that has been designed is capable to track the reference current by using the PID controller.

Figure 9: Reference and feedback current

IV. HARDWARE VERIFICATION

For the hardware testing, the output that needs to be generated from the Arduino is in Sinusoidal Pulse Wave Modulation (SPWM) signal. Figure 11 shows the SPWM output signal after the gate driver circuit. These signals will be used to switch on and off the power MOSFET inside the inverter circuit.

Figure 11: SPWM switching using Arduino

Figure 10 shows the three phase output line currents at the load. It shows that the line currents are balanced and the three phases are shifted equally in phase. These line currents are in sinusoidal waveform due to the low pass filter that has been used at the inverter output.

Figure 10: Output current at the inverter

Figure 12 shows the inverter output phase voltage when 20Vdc input source is applied. This output is measured at the inverter circuit before the load and filter. It shows, the
phase voltage is in the square-wave signal mode. To get a smooth sine waveform, a filtering process must be done to the output signal. It can been seen that, the peak to peak voltage is at 40Vpp and the phase shift is happen to all phases.

Figure 13 shows the inverter current that has been measured before the filter. It shows that current contains higher harmonics that caused the current profile to have weak signal.

Figure 12 : Inverter output voltage

Figure 13: Inverter current before filter

Figure 14: Inverter current output after filter

As a conclusion, the closed loop system from simulation and hardware results have shown the functionality of the Arduino as a low cost microcontroller that able to read the controller design block that has been developed in the MATLAB-Simulink environment. It indicates that, with the mathematical equations of any controller and without any programming code, the Arduino can still work as a microcontroller platform. Due to this advantage, it will help the new players or educators with less experience in microcontroller programming, to use the Arduino and the power converter circuit such as inverter to control the ac motor as a teaching material to teach non electrical engineering student or undergraduate electrical engineering student to understand the concept of ac motor control using a low cost microcontroller.

IV. REFERENCES


