Presented at the 5th DLSU Innovation and Technology Fair 2017
De La Salle University, Manila, Philippines
November 28 & 29, 2017

Motor Controller for a Pedelec Using Infineon MOSFETs and ARM Cortex-M0 Microcontroller with Android Integration

Kristofferson Reyes[1], Jomel Lorenzo Jr. [1], Jean Clifford Espiritu[1], Pamela Denisse Sacdalan[1], Noriel Mallari[1], and Isidro Marfori[1]
[1]De La Salle University - Gokongwei College of Engineering - REST
**Kristofferson Reyes: kristofferson_reyes@dlsu.edu.ph[1]*
Jomel Lorenzo Jr.: jomel_lorenzo@dlsu.edu.ph[1]
Jean Clifford Espiritu: jean_espiritu@dlsu.edu.ph[1]
Pamela Denisse Sacdalan: pamela_sacdalan@dlsu.edu.ph[1]
Noriel Mallari: noriel.mallari@dlsu.edu.ph[1]
Isidro Marfori: isidro.marfori@dlsu.edu.ph[1]

Abstract: The paper presents a study which aimed to design a motor controller of an electric bicycle with the use of Infineon MOSFET and an XMC microcontroller with smartphone integration. The paper focused on improving existing motor controllers in the market by adding the capability to interpret data obtained from the battery, the smartphone, and the motor. It was possible to obtain data obtained from the e-bike with by including Bluetooth in the design of the motor controller. The design of the smart motor controller was then implemented by integrating a Bluetooth chipset, a MOSFET driver block, the ARM Cortex-M0 microcontroller, and various sensors in the circuit. The microcontroller was also programmed to develop an adaptive motor controller capable of achieving a target speed based on sensor readings while being able to convert the sensor values using an ADC and timely transmitting the sensor readings to the Android application as telemetry using the Bluetooth chipset. The Android application was developed to display the real-time status of the e-bike while it receives and stores the telemetry for future data analysis. The research developed a motor controller, and an Android application for a Pedelec which opens the possibilities of implementing new complex control algorithms in future researches.

Key Words: Electric bicycle; BLDC motor control; ARM-Cortex-M0; Android; Telemetry

1. INTRODUCTION

1.1 Background of the Study

Long distance travels are tiresome for average bikers who are not used to biking such distances. Motorcycles on the other hand, are not eco-friendly and healthy alternatives as compared to using bicycles. The solution is an electric bicycle – a hybrid with the comfort of the motorcycle with the eco-friendliness of the bicycle. Electric bicycles would normally have a hub motor. It may be brushed or ("Sine-wave controllers, making hub-motors super quiet", 2017). A brushless DC motor (BLDC) was used in the study due to its higher efficiency, higher speed ranges, higher torque to size ratio and higher durability. Additionally, the study utilized a microcontroller with the capability to control MOSFETs (Metal Oxide Semiconductor Field Effect Transistor) for the development of an adaptive motor controller with Android integration to the electric bicycle.

To summarize, Electric bicycles are lighter and smaller compared to motorcycles yet it gives the comfort of riding one while being more energy efficient. It also allows its users to regularly exercise while biking. However, to reap these benefits, an advanced brushless DC (BLDC) motor controller for the electric bicycle is needed.

Currently, e-bikes available in the market assist riders when pedaling, however they do not have the capability to process complex data from the e-bike (Faraday Bicycles, 2017). For example, without a complex software algorithm, it would not be possible to compute for the distance remaining in the current state of charge of the e-bike because there are multiple factors that determine the range of the e-bike such as the user-specific biking pattern, and even the varying weight of its users. Furthermore, the e-bike would not be able to perform advanced control methods such as targeting a specific speed or a specific power output without obtaining the present status of the e-bike such as time, speed, throttle input, voltage, current, power, cadence and GPS location. Therefore, to improve the capability of existing e-bikes in the market, the study focused on developing a motor controller with the capability to capture and process data which enables the e-bike to perform more complex control algorithms in the future.

1.2 Review of Related Literature

A study about a Smart Monitoring system for e-bikes focused on the idea of having sensor data that the e-bike may collect (Kiefer & Behrendt, 2016). Collecting data from the e-bike can potentially increase the efficiency of the system since algorithms may be tuned to realize an optimal performance curve. This research paper also compared recent features of e-bike from different manufacturers and how it is different from their proposed design. The paper concluded that all eight e-bike systems and project utilized the global positioning services (GPS) while not all systems gave real-time information to the user. Furthermore, only three of the eight systems used an open-sourced software while only their proposed Smart E-bike Monitoring System (SEMS) would track the use of the e-bike to provide future assistance. The proponents' study stands out from the crowd since a purpose of the research is to enable the e-bike to monitor the status of the e-bike at multiple instances in time through the Android application.

Moreover, the next study focused on the implementation of a smart motor controller that uses an alternative model-based algorithm for controlling a pedal-assisted bicycle (Mallari, Macaraig, Navarrete & Marfori, 2016). The algorithm determines the intended speed of the rider purely through a cadence sensor. The paper's algorithm computes the intended power output that the motor controller should supply. This study incorporated a proportional control in controlling the speed of the electric bicycle. However, the research paper used a motor controller bought from the market which restricted the response of the electric bicycle. Their system behaved in a way that speed and power control cannot be modeled accurately. The paper suggested that the development of a customized motor controller is crucial to solve the problem in the restricted responses of the electric bicycle.

1.3 Objectives

The study aimed to develop a motor controller for a smart e-bike using Infineon MOSFETs and the XMC1302 microcontroller. Additionally, the study aimed to add features to the developed motor controller. The motor controller would have the ability to capture data obtained from the battery, the smartphone, and the motor. These acquired data would be understood as speed, voltage, current, power and GPS location. The study also aimed to develop an Android application which would present data to the user while at the same time storing it in a comma-separated values (.csv) file for data analysis. The transmission of the data to the Android application will be done using a Bluetooth connection.

1.4 Scopes and Delimitations

The study only aimed to develop the circuitry that can control the BLDC motor while implementing a control algorithm for achieving a target speed. The study did not cover the algorithms that must be considered to fully implement pedal-assist. The microcontroller also transmits excess data to Android application rather than just time, speed, throttle input, voltage, current, power and cadence as discussed earlier, but the excess data will not be analyzed. Additionally, the research was confined to the limits to the motor of the electric bicycle. The motor used in the research is a 350W rated BLDC motor operating at 36V with a top-speed of 35kph during freewheel.

2. METHODOLOGY

The methodology is composed of three different sections, namely the theoretical considerations, the hardware design and considerations, as well as the software design and considerations. The theoretical considerations discussed high side trapezoidal commutation patterns for BLDC motor control, Infineon's 32-bit ARM microcontroller, MOSFETs and gate drivers, Bluetooth connectivity, Android application development, as well as the overall system block diagram. On the other hand, the hardware design focused more on the specifications of the MOSFETs, the sensors used, and the schematic diagram of the circuit. Finally, the software design focused on the development of the code for the 32-bit ARM Microcontroller from Infineon, the telemetry protocol, and the development of the code for the Android smartphone.

2.1 Theoretical Considerations

2.1.1 BLDC Motor Control using MOSFETS

Although BLDC motors are advantageous compared to brushed DC motors, BLDC motors requires complex control algorithms to operate ("BLDC Motor Control Algorithms", 2017). A three-phase inverter designed by using MOSFETs enable the control of a BLDC motor via PWM signals to commutate the motor to its desired position. Hall sensors determine the current position of the motor at any state in time. Table 1 shows the required PWM signals applied to the gate of the MOSFETs that performs a high-side commutation on the BLDC motor.

Table 1. High-Side Commutation with Hall Sensors

#	Hall Sensors			Input MOSFET Gate Signals					
	U	V	W	U_H	U_L	V_H	V_L	W_H	W_L
1	L	L	H	L	L	PWM	L	L	H
2	L	H	H	PWM	L	L	L	L	H
3	L	H	L	PWM	L	L	H	L	L
4	H	H	L	L	L	L	H	PWM	L
5	H	L	L	L	H	L	L	PWM	L
6	H	L	H	L	H	PWM	L	L	L

2.1.2 Block Commutation using High-Side Trapezoidal Control Signals

There are three BLDC control schemes in trapezoidal control: High side, Low side, and Synchronous control. An advantage of High side control is that there is no inversion of signals in the low-side MOSFETs, thus no dead time is required. Figure 1 shows the High side commutation pattern for the BLDC motor.

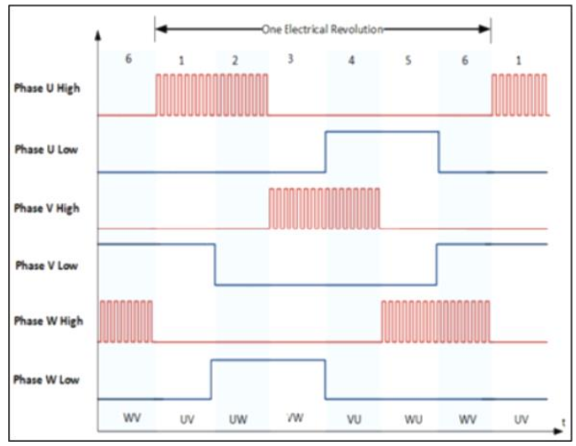

Fig. 1. Trapezoidal Commutation - High Side Modulation

2.1.3 Infineon MOSFETs and Gate Drivers

Infineon MOSFET and drivers are the devices used in the research to implement BLDC motor control. IPP041N12N3 G (Infineon, 2014) is the MOSFET used for the 3-phase inverter while 2EDL05N06PF (Infineon, 2016) is the gate driver that drove the inverter to operate the motor. These devices have low power consumption and are specifically designed for motor control. The PWM signals applied to the gate driver will come from the ARM microcontroller, discussed in the next section. The package and the pin configuration of each devices are shown in Figure 2.

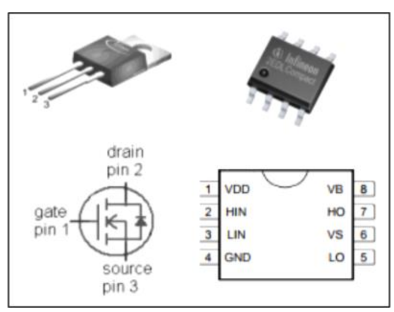

Fig. 2. Package and Pin Configuration of IPP041N12N3 G (left) and 2EDL05N06PF (right)

2.1.4 Infineon XMC 32-bit ARM Microcontroller

The microcontroller used in the research is the XMC 1300 Boot Kit, found in Figure 3, which is reprogrammed using the DAVE development platform. It utilizes an ARM Cortex-M-0 based XMC1302 microcontroller in TSSOP-38 with 200KB Flash and a microprocessor that runs at 32MHz.

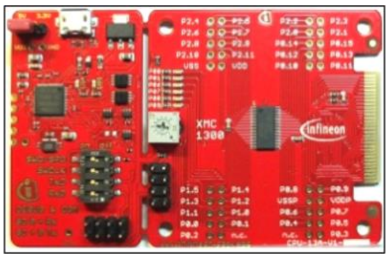

Fig. 3. *Infineon XMC 32-bit ARM Microcontroller Boot Kit*

2.1.5 Relevant XMC Peripherals

1. *ADC_MEASUREMENT_ADV*

The ADC_MEASUREMENT_ADV APP measures multiple analog signals using the Versatile Analog to Digital Converter (VADC) peripheral. It provides advanced features such as sequencing, post processing, synchronized measurements and hardware boundary check.

2. *PWM_BC*

The PWM_BC APP generates a pulse width modulated output for block commutation (trapezoidal) to drive three-phase two level voltage inverter. It uses CCU8 peripheral for generating PWM and it also configures multichannel configurations of POSIF peripheral. The PWM_BC APP requires multichannel pattern (commutation pattern) input and desired amplitude (duty cycle) from user periodically to generate PWM for block commutation.

3. *SYSTIMER*

The SYSTIMER APP uses the SysTick interrupt to call user functions periodically at a specified rate or after a given time period expires.

4. *UART*

UART APP configures the USIC peripheral of the XMC microcontroller, to work in asynchronous serial communication (ASC) mode. The APP provides APIs to transmit data, receive data, get the status flags and clear the status flags. APIs to transmit and receive data are implemented using interrupts to optimize the CPU utilization.

5. *PIN_INTERRUPT*

The PIN_INTERRUPT APP invokes user interrupt handler in a response to rising and/or falling edge event signal on a pin. The APP configures a port pin as general-purpose input pin and uses its transitions to generate an event.

6. *INTERRUPT*

The INTERRUPT APP allows the user to overwrite the provided default implementation of the interrupt service routine and to set the interrupt priority. The user needs to specify the implementation of the ISR.

2.1.6 Bluetooth Connectivity

The Bluetooth module HC-05 was used for sending data to the smartphone application. The module was set to operate at a baud rate of 57,600 bits/sec. Figure 4 displays the Bluetooth Module used in the circuit.

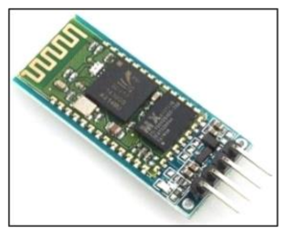

Fig. 4. HC-05 Bluetooth Module

2.1.7 Android Application Development

For the development of the smartphone application, Android Studio (Figure 5), the official software for developing Android applications was used.

Fig. 5. Android Studio IDE Logo

2.1.8 System Block Diagram

The overall block diagram of the system is composed of three blocks found in Figure 6: the smart motor controller, the smartphone, and the sensor block. The smart motor controller block is responsible for delivering the power necessary to run the BLDC motor. The sensor block creates feedback to the microcontroller to intelligently control the motor. On the other hand, the smartphone block computes, stores, and displays data such as speed, power, battery life, distance traveled, and total energy used.

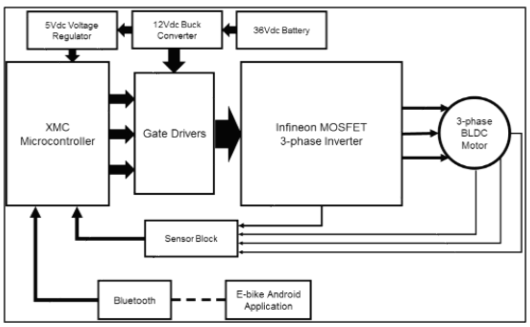

Fig. 6. Overall System Block Diagram

2.2 Hardware Design and Considerations

2.2.1 MOSFET Breakdown Voltage

The 36-Volt battery used in the motor controller, was considered before selecting the right MOSFETs to use. The breakdown voltage of the MOSFET is one of the most important characteristics to be considered in choosing the right MOSFET for the inverter (Davis, 2017). To ensure the reliability of the system, the breakdown voltage of the MOSFET should be greater than twice of the battery voltage. The IPP041N12N3 G, which has a breakdown voltage of 120 Volts, was selected for it still allots sufficient headroom in the circuit if a larger motor were to be used in the future.

2.2.2 Gate Driver Response Time and Determination of the Switching Frequency

The response time of the gate driver is important in selecting the right switching frequency to use for the six PWM signals. The Infineon MOSFET driver specified in section 2.1.3 has a response time of 0.3µs for the MOSFET and 0.8µs for the IGBT. The derived response times support the 20kHz switching frequency used in PWM generation. Furthermore, determining the proper switching frequency varies on the application. The 20kHz frequency was used since 20kHz is the frequency beyond the human audible sound band. 20kHz is also perfect for the MOSFETs due to its low switching losses compared to higher frequency values.

2.2.3 Sensors

For the motor controller to intelligently function, a closed-loop control is required which is possible by using sensors that assist the microcontroller in controlling the BLDC motor. Time, speed, throttle input, voltage, current, power and cadence are some of the important sensors to run the motor controller on a closed loop application. The values of the sensors are processed by the microcontroller to run the BLDC motor.

2.2.4 Schematic Diagram

The smart motor controller consists of the three-phase inverter, gate drivers, buck converter, voltage regulator, microcontroller, and other measurement circuits. Figures 7, 8, 9, and 10 shows the schematic diagram used in the motor controller circuit.

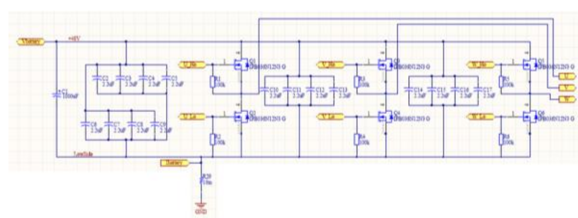

Fig. 7. Schematic Diagram of the Three-phase Inverter

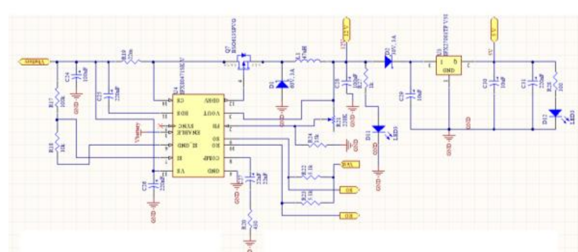

Fig. 8. Schematic Diagram of the Buck Converter and Voltage Regulator

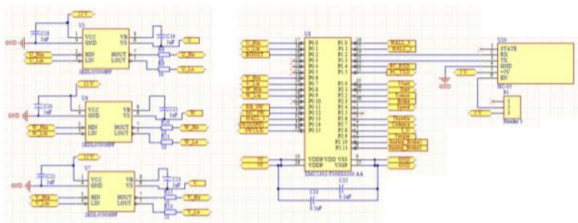

Fig. 9. Schematic Diagram of the MOSFET Gate Drivers and Microcontroller

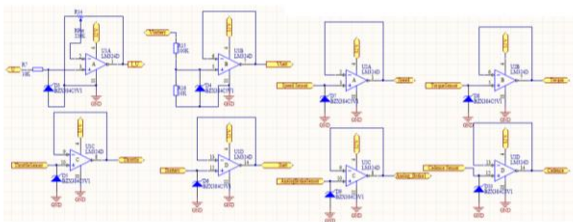

Fig. 10. Schematic Diagram of the Measurement Circuits

2.3 Software Design and Considerations

2.3.1 DAVE APPs Setup

The microcontroller's main DAVE APPs were the ADV_MEASUREMENT, PWM_BC, SYSTIMER, and UART. A total of 3 INTERRUPTs were configured as well as 3 PIN_INTERRUPTS. The APP Dependency is found in Figure 11.

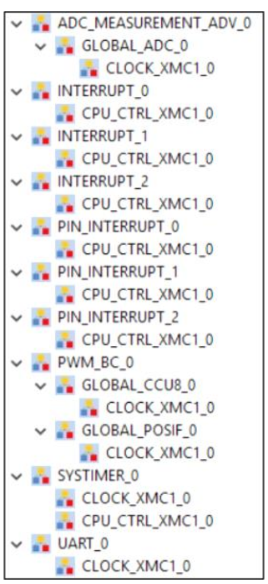

Fig. 11 APP Dependency

2.3.2 Interrupts Configured

The XMC1302 32-bit microcontroller uses a PeriodMatchInterrupt that triggers at the end of every PWM period. It also uses an ADCHandler interrupt that allows sampling and reading of the measured analog values. A TrapInterrupt was also included as it was an interrupt required by the microcontroller.

Three more interrupts that measure values such as speed, cadence, and the brake signal were also included in the system. The brakeHandler interrupt would turn off the motor when the brake is triggered. The brakeHandler is an important interrupt because it is the only interrupt that the user can manually trigger in the system to command the microcontroller to instantaneously switch the motor controller off by setting the PWM duty cycle to 0%. The speedTicksHandler and cadenceTicksHandler interrupt oversee the tracking and processing of how many hall sensors have passed the wheel and the crank respectively to determine the speed and cadence of the system.

2.3.3 Proportional-Integral Control of Speed

Since the motor controller obtains real-time information of the electric bicycle such as speed, throttle input, voltage, current, and power, a closed-loop proportional-integral control of the speed of the system is possible. The actual parameter modified by the PI control is the PWM duty cycle applied to the motor controller since a higher PWM value causes the motor to rotate quicker. The maximum PWM duty cycle configured for the algorithm is 92%. A 100% duty cycle or a DC signal has the capability of destroying the circuit as the high side and low side MOSFETs would have a time interval when they are shorted, thus it is general practice to avoid a 100% duty cycle. Likewise, a value near 100% duty cycle is not recommended due to the capacitor's discharge time in the circuit. Thus, it is expected for the motor to be supplied less than 350W of power.

The desired speed to run the system is determined by the user via the throttle input. The maximum speed that the user can target is 35kph and the PI control algorithm would try to reach the desired speed. In the actual scenario, 35kph would not be achieved since it is the maximum speed the electric bicycle can run at with no load. If the user sets a target speed below the actual speed, the PWM duty cycle would decrease. If the target speed is 0kph, the BLDC motor would automatically turn off due to the nature of the PI control.

2.3.4 Telemetry Protocol

The HC-05 Bluetooth module was configured to run at 57,600 baud. It is necessary to design a protocol for transmitting data between the Android application and the smart motor controller so that the data may be understood between the devices. The protocol for sending packets is found in Figure 12.

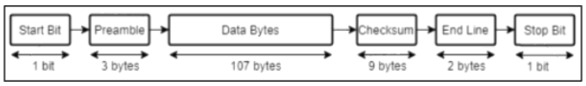

Fig. 12. Telemetry Protocol

2.3.5 Android Application

An Android application was developed for two purposes. One, to display real-time information of the electric bike to the user, and two, to store data for later analysis. It's able to show where the user is in the world with the use of the GPS. The user is not expected to interact with the application while cycling, but instead may use the application to read the sensor data of the e-bike, much like how the dashboard of a vehicle operates. The most useful data for the biker is the battery life since it serves as an indicator of when the e-bike should be recharged. The Android Application GUI is observed in Figure 13.

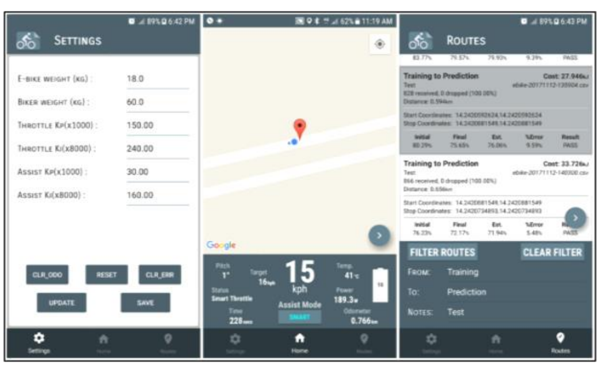

Fig. 13. Screenshots of the Android Application

3. RESULTS AND DISCUSSION

3.1 Speed (kph) vs. Time (s)

The graph in Figure 14 below represents the telemetry data recorded by the smartphone application regarding the electric bike's speed vs. time as it traversed the rotunda outside De La Salle University - Laguna Campus. The microcontroller can successfully compute the velocity of the electric bicycle. At the same time PI control was utilized to meet the desired speed. The significance of obtaining the data opens the possibilities of processing data to implement even more complex control algorithms that can make the e-bike perform in a variety of modes.

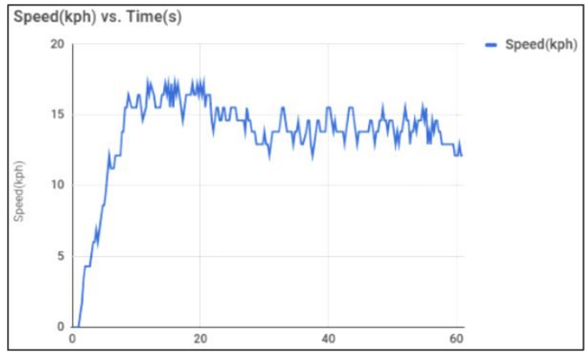

Fig. 14. Speed vs. Time Graph from Telemetry

3.2 Speed (kph) and Power (W) vs. Time (s)

The graph in Figure 15 includes the power consumption as the e-bike traversed the rotunda. When the e-bike was moving downhill between time 20s and time 30s, the motor was not needed, thus the PI algorithm slowly reduced the PWM duty cycle applied to the motor until it eventually turned off the motor. Going uphill required the PI algorithm to increase the duty cycle, thus increasing the system's power to maintain the speed. The algorithm caused the BLDC motor to re-engage as needed to maintain the target speed set by the user.

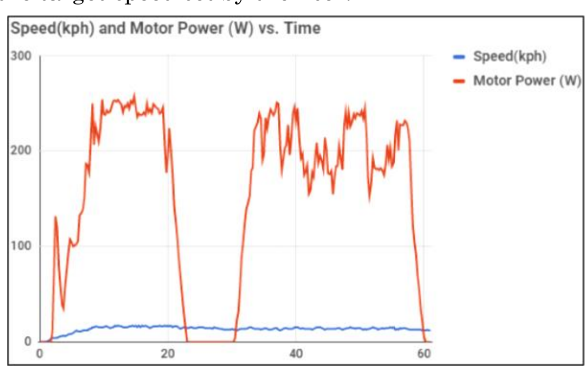

Fig. 15. Speed and Motor Power vs. Time

4. CONCLUSIONS

The research successfully developed a motor controller for a smart e-bike using Infineon MOSFETs and the XMC1302 microcontroller. The XMC microcontroller properly transmitted telemetry data over to the smartphone application by having both devices follow the same communications protocol which enabled the system to perform more complex control algorithms such as PI control. Additionally, the smartphone application displayed the received data in the user-interface and stored the data for further analysis.

The analyzed data represented the actual behavior of the e-bike when being used under normal road conditions. When the user inputs a target speed higher than the current speed, the PI control enabled the motor to accelerate by increasing the PWM duty cycle applied to the MOSFETs. The algorithm also has the capability of turning the BLDC Motor off when power is not needed and turning it back on when it is. The two cases where the motor can turn off is when the user manually enables the brake of the electric bicycle or when the target speed is below the actual speed, which adjusts the PWM duty cycle to 0% based on the PI control algorithm.

As for the future directives of the system, the next phase is implementing a more complex control method, called pedal-assist which would require a cadence and torque sensor to obtain the input human power. Furthermore, since the Android application reads the voltage and current of the battery, obtaining the characteristics of the battery would allow future researches to estimate and predict the state of charge (SoC) of the battery.

5. ACKNOWLEDGMENTS

The proponents would like to thank Infineon Technologies and Mr. Antonio Cordero for providing the 32-bit ARM Cortex-M0 microcontroller as well as the MOSFETs and Gate Drivers used in the research. The researchers would also like to thank Nyfti Inc. for providing the chassis of the bicycle used in the research. Finally, the researchers would like to thank Jonathan Macaraig for providing several sensors used in the research.

6. REFERENCES

Faraday Bicycles. (2017). Faraday Bikes. Retrieved 22 November 2017, from https://www.faradaybikes.com/features/

BLDC Motor Control Algorithms. (2017). Renesas Electronics India. Retrieved 22 November 2017, from https://www.renesas.com/en-in/solutions/key-technology/motor-control/motor-algorithms/bldc.html

Davis, S. (2017). Semiconductor Back-to-Basics: Power MOSFETs. Power Electronics. Retrieved 23 November 2017, from http://www.powerelectronics.com/discrete-power-semis/semiconductor-back-basics-power-mosfets

Hazari, M., Jahan, E., Siraj, M., Khan, M., & Saleque, A. (2014). Design of a Brushless DC (BLDC) motor controller. 2014 International Conference On Electrical Engineering And Information & Communication Technology. http://dx.doi.org/10.1109/iceeict.2014.6919048

Infineon. (2016). EiceDRIVER High voltage gate driver IC. Datasheet.

Infineon. (2014). OptiMOS Power-Transistor. Datasheet.

Kiefer, C., & Behrendt, F. (2016). Smart e-bike monitoring system: real-time open source and open hardware GPS assistance and sensor data for electrically-assisted bicycles. IET Intelligent Transport Systems, 10(2), 79-88. http://dx.doi.org/10.1049/iet-its.2014.0251

Mallari, N., Macaraig, J., Navarrete, D., & Marfori, I. (2016). Smart motor controller using model-based algorithm for control of pedal-assisted electric bicycle.

NOGUCHI, S., SUZUKI, K., & DOHMEKI, H. (2015). Comparison of efficiencies depending on excitation angle of a square-wave brushless DC motor. Journal Of The Japan Society Of Applied Electromagnetics And Mechanics, 23(2), 276-281. http://dx.doi.org/10.14243/jsaem.23.276

Sine-wave controllers, making hub-motors super quiet. (2017). ElectricBike.com. Retrieved 22 November 2017, from https://www.electricbike.com/sine-wave/

Tariq, M., Bhattacharya, T., Varshney, N., & Rajapan, D. (2016). Fast response Antiwindup PI speed controller of Brushless DC motor drive: Modeling, simulation and implementation on DSP. Journal Of Electrical Systems And Information Technology, 3(1), 1-13. http://dx.doi.org/10.1016/j.jesit.2015.11.008