# The Medspencer

Matthew Hoover, Sakeenah Khan, Gustavo Morales Burbano, and Ivan Alvarez

Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — This paper presents the electronic hardware and software design for the Medspencer project. The Medspencer is an automatic medication dispenser, and is implemented as a home appliance for multiple users. The main goal of this project is to accurately schedule and track each individual user's medications and dispense a single dose to the correct user at the appropriate times. The Medspencer reminds the user to take their medication appropriately.**

***Index terms* — Power regulation, filtering, processor, microcontroller, resistive touch panel, LCD display, fingerprint recognition, demultiplexer, audio amplifier, PCB**

## I. INTRODUCTION

The motivation for the Medspencer project is to increase medication compliance in patients with multiple prescription medications and schedules. Medication noncompliance is a serious issue that limits the effectiveness of health care services and prescribed medications. About 50% of patients with chronic illnesses do not take medications as prescribed [6]. Medication noncompliance has serious health consequences, which include decreased quality of life, poorly managed symptoms, and even death, as well as extra costs to the health care system.

Our team's main goal for this project is to provide the support to help patients keep to their strict medication regimens and exhibit improved medication compliance. Our main goal is broken down into several specific objectives: 1) to prevent patients from missing a dose or taking a dose at the wrong time; 2) to prevent overdosing; 3) to prevent people from consuming other patients' prescriptions; and 4) to strengthen the patient-provider relationship.

To achieve objective 1), the Medspencer will organize and schedule doses and notify the patient when it is time to take the dose. A speaker is used to sound and notify the patient. In the case of a missed dose, the Medspencer will reschedule the dose. The Medspencer eliminates any human error from manual sorting of medications, which prevents mixing up medications or taking them at the wrong time. To achieve objectives 2) and 3), the Medspencer shall be a secure device that can only be opened or modified by an administrator/caretaker. To access the device, a patient's fingerprint scan is required. This ensures that the correct patient is getting the prescription. Medications will only be dispensed to a patient if it is time for the next dose (for 'fixed schedule' prescriptions), or if enough time has elapsed since the last dose (for 'as needed' prescriptions). To achieve objective 4), the patients' compliance will be recorded in a schedule. This can be compiled and sent to a health care provider.

The Medspencer's hardware and software designs were managed by the team's two electrical and two computer engineers, respectively. The team received advice and funding by Dr. Fredesvinda Jacobs-Alvarez, MD, head physician of Esperanza Behavioral Health and Services.

## II. HARDWARE

### *A. Overview*

For this project, the main components required include the microcontrollers, LCD display, resistive touch panel, fingerprint scanner, speaker, and dispensing mechanism. Additional components include an SD card reader, a touch panel controller IC, an audio amplifier for the speaker, a demultiplexer to control the motors that make up the dispensing mechanism. The basic component set-up is displayed in the hardware block diagram shown in Figure 1.

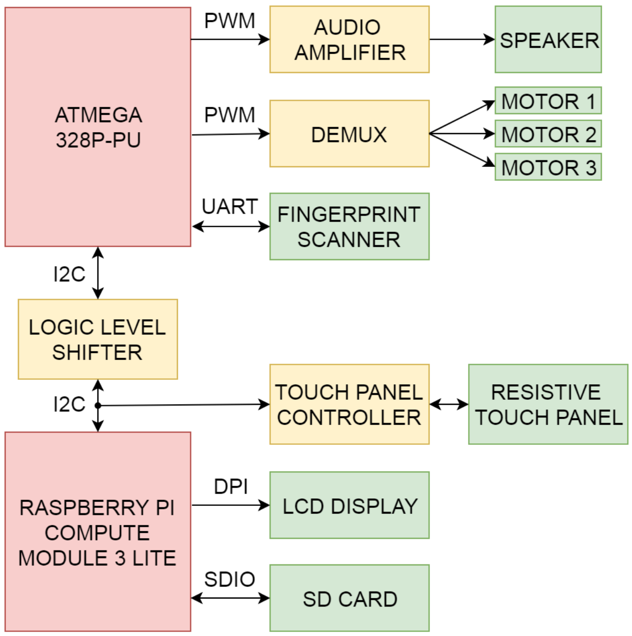



FIGURE 1. Hardware block diagram

To communicate between the microcontrollers and our various peripherals, the interfaces we utilized include Inter-Integrated Circuit (I2C), Universal Asynchronous Receiver/Transmitter (UART), Display Parallel Interface (DPI), Secure Digital Input Output (SDIO), and Pulse Width Modulation (PWM). The communication protocols that each component uses are detailed in Figure 1.

## *B. Power Requirements*

To power the microcontrollers and all of the peripherals, we use a 20V DC power supply. In total, our project requires 9 different voltage levels. All of the power requirements for the project are summarized in Table 1. The Webench Power Designer by Texas Instruments was utilized as an aid for the power regulation design process [8]. While designing the power regulation system, we carefully considered the required input and output power for each hardware component and the component's datasheet specifications. We also considered the efficiency, size, and availability of the voltage regulators, the required external circuitry, and whether the regulator was adjustable or fixed.

TABLE 1. Power Requirements

| COMPONENT | $V_{SUPP}$ (V) | $I_{SUPP}$ (mA) |
|---|---|---|
| ATMEGA328P-PU | 5 | 4 |
| Raspberry Pi CM3L | 3.3<br>1.8 | 250<br>250 |
| BOB-12009 Bidirectional Logic Level Converter | 5<br>3.3 | 22 |
| 104031-0811 SD Card Reader | 3.3 | 500 |
| AT070TN90 LCD Display | 3.3<br>10.4<br>10.0<br>16.0<br>-7.0<br>3.8 | 10<br>50<br>135<br>1<br>1<br>10 |
| AR1021 Resistive Touch Screen Controller | 3.3 | 0.125 |
| CD74HC238M Demultiplexer | 3.3 | 50 |
| SG90 Servo motors | 5 | 220 |
| LM386 Audio Amplifier | 12 | 4 |
| R307 Fingerprint Scanner | 5 | 50 |

The power input to each component should be filtered of noise. To filter out noise, capacitors can be placed in parallel to a load; these are called decoupling capacitors. Decoupling capacitors suppress high-frequency noise in the power supply, which may cause damage to sensitive hardware components.

## *C. Processors*

For the Medspencer, the processors oversee recording patient and medication data, displaying information on the LCD display, accepting user input from the touchscreen and fingerprint scanner, making alerts via the speaker, and sending commands to the servo motor controller. From a data analysis standpoint, the computing requirements are not heavy. The amount of data that we must record, process, and transmit for our basic functionality is small: even a budget microcontroller could meet those memory requirements. What really dictates the requirements of the Medspencer's central processor is the display. The display we have chosen is a 7" 800 x 480 touchscreen, which presents a processing and memory threshold that cannot be addressed by an MSP430.

For the purposes of our project, we have chosen to utilize the ATMEGA328P microcontroller in conjunction with the Raspberry Pi Compute Module 3 Lite (CM3L). The CM3L was chosen to interface with the touchscreen display, while the ATMega328P controls the fingerprint scanner, speaker, and dispensing mechanism.

The ATMEGA328P, shown in Figure 2, is a low-power 8-bit microcontroller based on the AVR enhanced RISC architecture. It can execute most instructions in one clock cycle, and has 32kB flash memory, 1kB EEPROM, and 2kB internal SRAM. There are 23 programmable I/O lines; available interfaces include USART, PWM, SPI, I2C, and ADC [1].

FIGURE 2. ATMEGA328P-PU microcontroller

The Raspberry Pi CM3L, displayed in Figure 3, is a System on Module (SoM) containing a processor, memory, and supporting power circuitry. The CM3L contains a BCM2837 processor, and the software includes the ARMv8 instruction set and Linux software stack. 1GB RAM is available, and an SD card up to 64 GB can be fitted using the SD/eMMC interface. The CM3L has 48 GPIO pins and supports I2C, SPI, UART, SD/SDIO, HDMI, USB, and DPI communication interfaces [7].

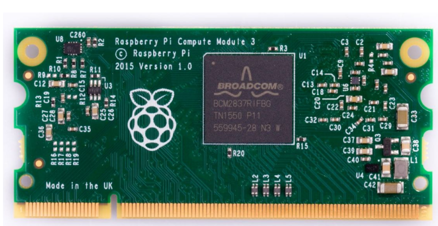



FIGURE 3. Raspberry Pi Compute Module 3 Lite

The CM3L and ATMEGA328P communicate with each other using I2C protocol. Since the two modules use different voltage levels, a logic level shifter is required. Options for the logic level shifter include using a surface mount IC, or implementing a simple level shifting circuit using MOSFETs. The C3ML uses input voltages of 1.8V and 3.3V, while the ATMega328P operates at 5V. These input voltages are supplied by the TPS62745, LP5912-3.3, and LMR23615 voltage regulators, respectively.

*D. Touchscreen Display*

The screen chosen for this project is the Innolux AT070TN90 shown in Figure 4, a Chinese display commonly found in consumer electronics. This display has a WVGA resolution (800 x 480), supports 24-bit color, and comes with a resistive touch panel. The datasheet contains a pinout for the 50-channel flexible flat cable (FFC), which contains all of the color channels, power connections, and timing signals for the display; it also contains specifications for the exact timing to send the screen color data. In addition, the working principle and usage for the touch panel can be found in the same datasheet [3].

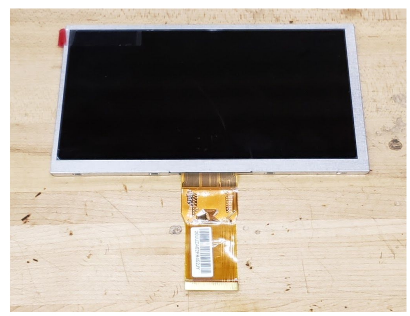


FIGURE 4. Innolux AT070TN94 LCD display

The LCD display and touch panel will be controlled by the Raspberry Pi CM3L. The display interfaces with the CM3L via DPI. The touch panel utilizes the AR1021 resistive touch screen controller, which communicates to the CM3L using I2C. The display requires several different voltage levels to operate, as shown in Table 1.

*E. Fingerprint Recognition*

The fingerprint scanner that we utilized for this project is the R307 Fingerprint Identification Module by Hangzhou GROW, as shown in Figure 5. This fingerprint scanner utilizes an optical sensor to scan the fingerprint and create a digital image. The module supports 1:1 or 1:N fingerprint matching. The fingerprint library can hold up to 1000 fingerprint templates.

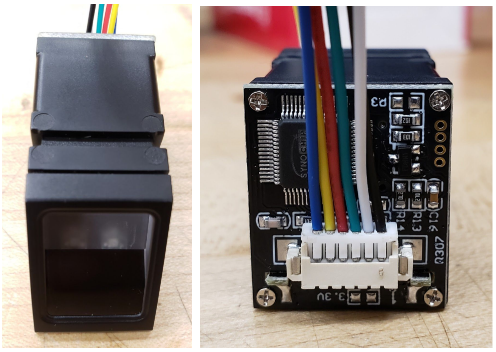


FIGURE 5. R307 fingerprint module

The fingerprint module contains both a scanner and processing microcontroller. It is controlled by commands sent over its UART serial connection. Fingerprint processing involves two parts, fingerprint enrollment and fingerprint matching. When enrolling a fingerprint, the user must enter the fingerprint two times. The system will process the two finger images, then generate and store a template of the finger. For 1:N matching, the user enters the fingerprint and the system will generate a template of the finger and compare it with the templates in the fingerprint library. The system will return either a success or failure, depending on whether a match is found.

The fingerprint module is interfaced to the ATMEGA328P microcontroller using UART communication. To power the R307 fingerprint module, 5V will be supplied by the LMR23615 step-down regulator.

### *F. Dispensing mechanism*

To facilitate the dispensing mechanism of the Medspencer, individual servo motors are used to dispense medicine from each medicine vial. For our prototype, we utilize 5 servo motors, which are accessed by the ATMEGA328P microcontroller via the CD74HC238M demultiplexer. Using a demultiplexer, the amount of servo motors and medicine vials could be expanded in the future.

We chose to use SG90 servo motors shown in Figure 6. These little devices are small enough to fit within the MedSpencer enclosure. For the purpose of this project, we created a filter that attaches to a motor and allows a pill to be dispensed when the motor is rotated. By default this servo motor can rotate 180 degrees maximum [9]. The ATMEGA328P microcontroller will control the servo motors using pulse width modulation (PWM).


FIGURE 6. SG90 servo motors

The operating voltage of the SG90 servo motor is between 4.8V to 6V. The LMR23615 step-down regulator was used to supply a regulated 5V to the servo motors.

### *G. Speaker*

In order to notify the patient when it is time to take their dose of medicine, we utilized a small speaker. When it is time for a scheduled dose, the speaker plays a simple sound to alert the patient. The speaker we chose is a 2W, 4Ω speaker by CUI, Inc.

The ATMEGA328P microcontroller will generate waveform signals for the speaker using PWM. The LM386 audio amplifier was utilized the amplify the PWM signal. The audio amplifier requires 12V, supplied by the LM5009 switching regulator.

### *H. PCB Design*

The PCB design and schematic software that we utilized for this project is EAGLE [10]. We designed two 80 x 100 mm PCBs. The first PCB (Figure 7) contains the ATMEGA328P microcontroller, Raspberry Pi CM3L, and connections for the SD card, touchscreen display, fingerprint scanner, motors, and speaker. The second PCB (Figure 8) contains the power jack for the 20V power supply and all of the voltage regulation circuits.

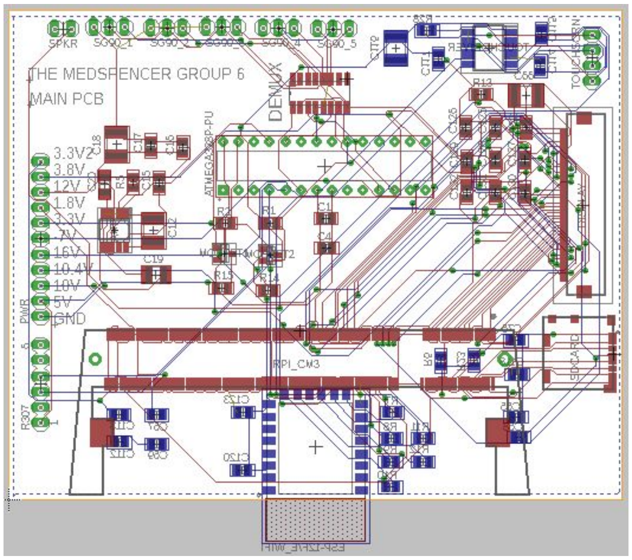



FIGURE 7. Main PCB layout

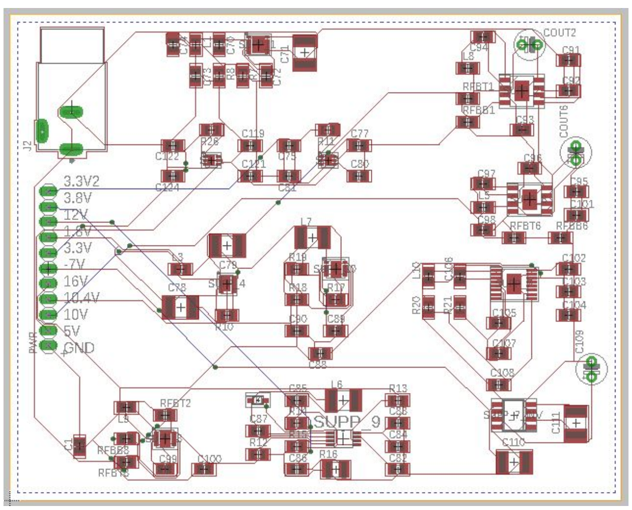


FIGURE 8. Voltage regulation PCB layout

During the initial design phase, the various hardware components and electrical circuits were tested in the UCF Senior Design Lab using development boards, breadboards, through-hole components, and the provided multimeter, oscilloscope, and DC power supply. Once the schematics and board layouts were finalized, the PCBs were ordered and received. The surface mount components and connectors were soldered onto the PCBs at UCF's Texas Instruments Innovation Lab.

## III. SOFTWARE

### *A. Overview*

The software of the Medspencer is broken into two parts: the software running on the Raspberry Pi CM3L, and the software on the ATMega328P microcontroller. The CM3L contains the code for the components that the user interacts with. It controls the display, reacts to the touchscreen. The ATMega328P manages the peripherals, meaning it controls the servo motors, speaker, and fingerprint scanner. This means that the ATMega328P does not react to user input on its own; it receives commands from the CM3L. This communication between the two processors define the project functionality.

### *B. Frame System*

The most critical part of the CM3L software is the frame system. Each frame consists of both a display object and event code. The display portion of a frame contains an arrangement of text, buttons, and other interactable components for the touchscreen. The event code is what reacts to user actions, and triggers changes in the system state, such as sending a command to the ATMega328P or moving to a new frame.

The first frame that is loaded is the Time frame, which simply shows the time. This is the beginning of the frame tree, where a user is presented with the opportunity to scan their fingerprint. If the user is registered as a patient, they enter the Patient frame, and if they are are recognized as an admin (or 'caretaker'), they are taken to the administrator frame. If their print is not recognized, they are refused entry. This process is shown in Figure 9.

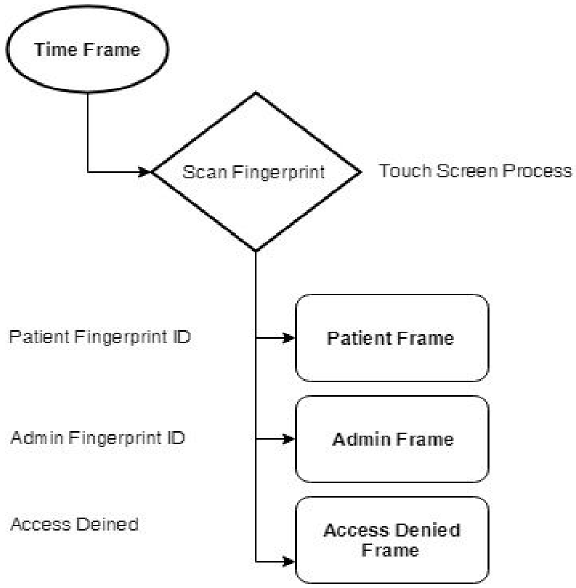



FIGURE 9. Fingerprint Frame

After the user scans their fingerprint, the system identifies them and moves them to the corresponding frame. In the Patient frame, the user will have access to their demographics, which contains the user's name, date of birth, address with zip code, phone number and email. From here, the patient can select a medication to dispense.

If the medication is listed as “As Needed”, then it is available at any time. However, some medications have a regulated dose, so they are only available according to a schedule. On this page, patients can either choose a medicine to dispense, or return to the Time frame. The structure of the Patient frame is shown below in Figure 10.

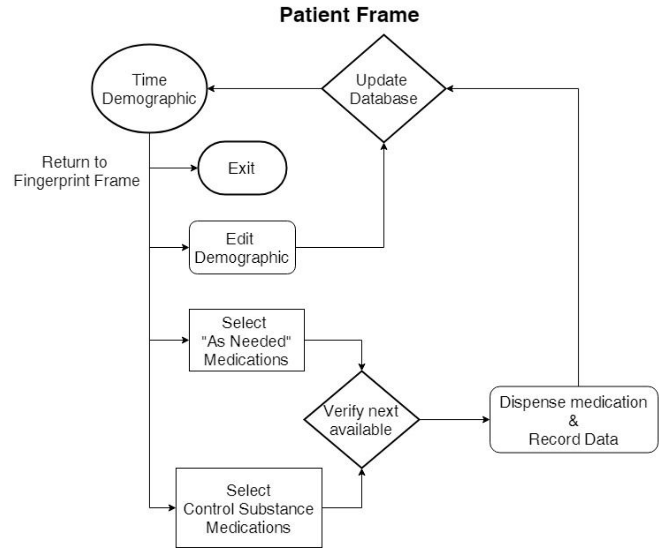



FIGURE 10. Patient Frame

If a user's fingerprint is registered as an administrator, they are taken to the Admin frame upon scanning. In the Admin frame, as seen in Figure 11, options exist for viewing the admin's and the patients' demographics, much like the patient; the important difference is that administrators are allowed to edit this information. Admins can view, update, add, and delete patients, as well as the patients' demographics and medication prescriptions. Administrators also manage prescribing doctors' profiles. These profiles contain demographic information, much like patient and admin profiles.

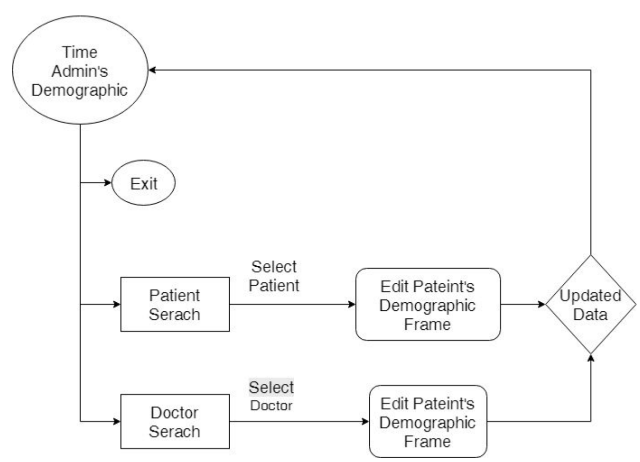



FIGURE 11. Administrator Frame

All user data is contained in an SQL relational database. This database maps patient, administrator, and doctor data to the frame when it is opened, and it also contains the fingerprint IDs of the patients and administrator. For every patient, the database will contain a unique alphanumeric sequence; a chart number, in medical terms. This will be used to identify and protect the patient's information. For every prescription, the database will record the medication name (brand name, not generic), dosage, timing restrictions, and number of dosages..

### *C. ATMega328P Microcontroller*

The user interacts directly with the Python program on the CM3L, so for their actions to affect the peripherals, a command must be sent from the CM3L to the ATMega328P controller. There are four commands that can be recognized by the ATMega328P: to verify a fingerprint, to enroll a fingerprint, to dispense a medicine, or to sound the alarm. These actions are triggered by the business logic in the CM3L program, which causes a command to be sent over I2C to the ATMega328P. This command is nothing more than a byte of data, indicating one of these options. The only command that requires an accompanying parameter is the dispense command: it needs a second byte to indicate which medicine cylinder to dispense from. The fingerprint commands must return some data: either the ID of the verified user, or the new ID of the enrolled user. Once this return data has been gathered by the ATMega328P, it is held until the CM3L sends a read request.

## IV. CONCLUSION

For our senior design project at UCF, we decided to create the Medspencer. Through completing this project, we learned how to apply the knowledge and skills from university and gained experience in designing and producing the electronic hardware and software that goes into a working product. We also gained experience in working in a team, listening to requests and advice from sponsors, and presenting our product professionally.


## ACKNOWLEDGEMENT

The authors would like to acknowledge the guidance and support of Dr. Fredesvinda Jacobs-Alvarez, MD, head physician of Esperanza Behavioral Health and Services. A special appreciation to all professors that challenge our knowledge pushing us to the limit and invite us to do research and solve real-life problem with the knowledge achieved during college. Without them, we are only numerical solving engineers rather than highly skilled professionals.

REFERENCES

[1] "ATMEGA328/P Datasheet." Atmel, Nov. 2016, http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

[2] "R307 Fingerprint Module User Manual." Hangzhou Grow Technology Co., Ltd., Feb. 2011, https://www.dropbox.com/sh/orprmb3bgb6lqb6/AACpiIXOF91R7-RQ9OkD4JXha?dl=0

[3] D. Lee, J. Yu, J. Huang, C. Chou, J. Lin. "LCD Module Specification" (Model Name: AT070TN92 V.1; Remark: Touch Screen Panel). Innolux Display Corporation, Feb. 2009.

[4] "ESP8266EX Datasheet," Espressif, Feb. 2018, https://www.mouser.com/pdfdocs/Espressif_ESP8266EX_Datasheet.pdf

[5] "R307 Fingerprint Module User Manual." Hangzhou Grow Technology Co., Ltd., Feb. 2011, https://www.dropbox.com/sh/orprmb3bgb6lqb6/AACpiIXOF91R7-RQ9OkD4JXha?dl=0

[6] E. Sabaté, editor. "Adherence to Long-Term Therapies: Evidence for Action." World Health Organization, 2003, http://www.who.int/chp/knowledge/publications/adherence_full_report.pdf

[7] "Compute Module Datasheet." Release 2. Raspberry Pi (Trading) Ltd., June 2018, https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM_2p0.pdf

[8] "WEBENCH Power Designer." Texas Instruments, 2018, http://www.ti.com/design-tools/webench-power-design/power-designer.html

[9] "Servo Motor SG90 Datasheet." Tower Pro, http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf

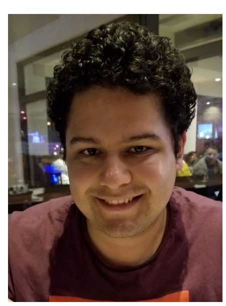
[10] "EAGLE." Version 9.0.0. Autodesk, Inc., 2018.


GROUP 6
SUMMER 2018

**Matthew Hoover** is a senior Computer Engineering student from the University of Central Florida. Currently, he is working as a Software Engineer at Riptide Software in Orlando, Florida.

**Sakeenah Khan** is a senior Electrical Engineering student from the University of Central Florida. She is a Burnett Honors College student, and is the vice president of Eta Kappa Nu, the honor society of IEEE. She conducted research on GaN power devices as an undergraduate student. Upon graduating, she will begin working at Texas Instruments in Tucson, Arizona.

**Gustavo Morales Burbano** is a senior Electrical Engineering student from the University of Central Florida. He has an Avionics Technician certificate from UCF Continuing Education. He started his major at Escuela Superior Politecnica del Litoral in Ecuador. He will join Naval Air Systems Command in Maryland as an electrical engineer upon graduation.



**Ivan Alvarez** is a senior Electrical and Computer Engineering student with a Minor in Secure Networking and Computing from the University of Central Florida. Since 2008, he started working as an Information Technician for Esperanza Mental Behavioral Health Clinic in Orlando, Florida, where he is now the head of the Technical Department.