UNIVERSIDADE
BEIRA INTERIOR

eBike Energy Meter with Bluetooth and MQTT Data Transfer

Marcelo Aires Moreira

Dissertação para obtenção do Grau de Mestre em

Engenharia Eletrotécnica e de Computadores

(2º ciclo de estudos)

Orientador: Prof. Doutor António Eduardo Vitória do Espirito Santo

Outubro de 2021

Folha em branco

Acknowledgements

First, I would like to thank God for creating the world and for his grace and mercy.

Also, I would like to thank the project Urban-Air Project, cofounded by FEDER - Fundo Europeu de Desenvolvimento Regional (European Resources to Regional Development), through the POCTEC - Programa V-A Espanha-Portugal (Portugal-Spain V-A Program) for the scholarship and the commitment with the urban mobility sustainability.

I would like to thank my family for all the support and motivation they gave me during all the mastering process.

I would like to thank my advisor, Professor PhD António Eduardo Vitória do Espirito Santo, for all the guidance, commitment, help and support during all the mastering.

I would like to thank my friends for the encouragement, moments of fun and friendship.

I would like to thank people from the Instrumentation and Measurement Lab for the conviviality.

Last, I would like to thank all the professors and employees of Universidade da Beira Interior for all the teaching and support.

Folha em branco

Resumo Alargado

INTRODUÇÃO

A introdução aborda o contexto do problema e motivação, os objetivos, metodologia e estrutura da dissertação. As principais categorias de mobilidade elétrica são elevadores e veículos elétricos. Os elevadores são uma forma de transporte confiável e conveniente. Já os veículos elétricos estão ganhando importância global pela sua capacidade de reduzir os gases responsáveis pelo efeito estufa e pela sua alta eficiência, mais alta do que carros a combustão, hidrogênio, etanol e biodiesel.

O objetivo geral da dissertação é desenvolver um sistema embarcado capaz de medir tensão, corrente e potência de uma bateria de uma bicicleta elétrica, enviar os dados para um smartphone por Bluetooth, e do smartphone para um computador pela rede MQTT. O sistema deve ser desenvolvido de acordo com as normas IEEE1451.

A metodologia usada foi uma revisão de literatura de medidores de energia desenvolvidos ao redor do mundo para depois serem comparados com o sistema desenvolvido aqui; estudo das normas IEEE1451 para entender como aplicar os princípios no desenvolvimento do sistema; projeto do sistema baseado em aquisição de dados de sensores, transferência de dados com atuadores e a comunicação entre eles e os processadores; desenvolvimento de hardware usando princípios de IoT e o microcontrolador mbed, o sensor INA260, o modulo de Bluetooth HC05 e uma breakout board SD; desenvolvimento de firmware baseado em pooling, interrupção, maquinas de estado e Data Sheet Eletrônicos dos Transdutores (TEDS); Desenvolvimento de aplicativo para smartphone baseado nas redes Bluetooth e MQTT; Desenvolvimento da Aplicação de Gerenciamento baseado na conexão de nos usando o software Node-Red; e Finalmente realização de testes e analise de resultados.

MOBILIDADE ELÉTRICA

Este capítulo aborda uma revisão teórica sobre veículos elétricos, bicicletas elétricas, medidores de energia e as normas IEEE1451. Existem diferentes configurações de veículos elétricos usando diferentes baterias, conversores de potência, motores elétricos e marchas. As baterias mais usadas em veículos elétricos são as de Lítio-Íon porque elas possuem uma alta capacidade de armazenamento, alta tensão, carga rápida e um longo ciclo de vida.

As bicicletas elétricas são um meio de transporte conveniente porque são pequenas, podem reduzir o congestionamento de trânsito e podem melhorar a saúde do ciclista. As bicicletas elétricas podem ser classificadas em geral, por função, por recursos estruturais e por proposito a depender se tem assistência do pedal, se são tripuladas, os tipos de estruturas, número de rodas e onde são usadas.

Os medidores de energia estudados na literatura foram publicados entre 2015 e 2019 e apresentaram testes em bicicletas que tem o motor elétrico mas também tem assistência do pedal. Os testes foram realizados principalmente na Europa e os parâmetros analisados foram elétricos, dinâmicos, ambientais.

A IEEE1451 é uma família de 7 padrões para padronizar transdutores e permitir interoperabilidade. A IEEE1451 propõe a adição de TEDS para armazenar informações sobre os transdutores ao transdutor inteligente além da rede de comunicação, processor de aplicação, condicionamento de sinal, conversão de dados e transdutores presentes no transdutor básico.

BICICLETA ELÉTRICA EQUIPADA COM UM MÓDULO DE INTERFACE DO TRANSDUTOR (TIM)

Este capítulo aborda o projeto do sistema embarcado, o desenvolvimento de hardware e software e a fabricação da Placa de Circuito Impresso (PCB). O projeto do sistema foi feito através da modelagem dos componentes como canais de transdutores como propõem as normas IEEE1451. Os canais de transdutores são um transdutor e todos os componentes de condicionamento de sinal e conversão associados a ele. Os canais de transdutores utilizados foram Atuador Digital Embarcado, Sensor de Evento Digital Embarcado, Sensor de Entrada Analógico e Sensor Digital Embarcado.

O desenvolvimento de hardware foi feito através do desenvolvimento de circuitos usando as interfaces I2C, serial e SPI. A PCB foi modelada e fabricada usando desenvolvimentos de esquemático e layout com o software Design Spark.

O desenvolvimento de firmware foi feito em C usando o compilador online mbed. No programa, primeiro há a inclusão de bibliotecas, definição, configuração, declaração e inicialização de variáveis e funções. Depois numa função loop, há a leitura de comandos por Bluetooth e execução de uma das ações: acionamento do canal de transdutor, leitura de TEDS ou apagamento de arquivos. Por último há a implementação de máquina de estado de cada um dos canais de transdutores e funções de aquisição de dados de potência, tensão e corrente.

APLICATIVO MOVEL DO PROCESSADOR DE APLICACAO DE REDE (NCAP)

O aplicativo para smartphone foi desenvolvido em Java usando o software Android Studio. Há a troca de dados por Bluetooth entre o sistema e o smartphone, e a troca de dados por MQTT entre o smartphone e a internet. O desenvolvimento de Bluetooth foi feito através da pesquisa de dispositivos e conexão, enquanto que o desenvolvimento MQTT foi feito usando funções de publish e subscribe.

APLICATIVO DE GESTÃO

O aplicativo de gestão foi desenvolvido usando Flow Charts no software NodeRed. Os nós Button, MQTT out, MQTT in, inject, msg.payload, Split, delay e chart foram conectados para habilitar a conexão do aplicativo com o smartphone e gerar uma Interface. A interface do aplicativo fica disponível online e possui botões que ativam as funções do smartphone e do sistema. Os dados de texto são mostrados numa scrolling table e os parâmetros elétricos são mostrados em gráficos.

SISTEMA FINAL

Este capítulo explica o funcionamento final do sistema. O aplicativo de gestão manda comandos para um servidor na internet por MQTT. O servidor envia esses comandos para o smartphone por MQTT. O smartphone envia os comandos para o sistema acoplado a eBike por Bluetooth. O sistema acoplado a eBike realiza aquisição de dados de parâmetros elétricos e envia os dados para o smartphone por Bluetooth. O smartphone envia os dados para o servidor na internet por MQTT, e o servidor envia os dados para o aplicativo de gestão em um computador.

TESTES DE VALIDAÇÃO

Este capitulo apresenta a realização de testes e analise de resultados. Além dos testes de hardware, o sistema passou por 3 conjuntos de testes. No primeiro conjunto de testes houve a aquisição de dados de parâmetros elétricos de uma resistência de potência. O segundo conjunto de testes foram realizados através da medição de parâmetros elétricos de uma bateria de uma bicicleta elétrica com corrente aproximadamente constante. O terceiro conjunto de testes também foi feito na bicicleta elétrica mas com corrente variável. O sistema inteiro funcionou bem. A aquisição de dados pelo sistema teve a mesma tendência que os dados adquiridos por um osciloscópio.

A transferência de dados entre o sistema, o smartphone e o aplicativo de gestão foi testado separadamente e o sistema respondeu aos comandos enviados pela interface. As TEDS foram recebidas e mostradas corretamente na scrolling table. Os parâmetros elétricos foram recebidos e mostrados nos gráficos corretamente.

CONCLUSÕES

O sistema desenvolvido pode contribuir para a realização de testes para verificar o consumo elétrico, corrente, tensão e potência de uma bicicleta elétrica submetida a diferentes condições como variação da massa do ciclista, massa da bicicleta elétrica, aceleração, velocidade, condições do ar, grau de inclinação e rotas. Isso pode permitir a descoberta das rotas mais econômicas e o perfil dos ciclistas. Desse modo, juntamente com o monitoramento da poluição do ar e carregamento por painéis fotovoltaicos, este projeto pode contribuir para a sustentabilidade da mobilidade urbana e melhor qualidade de vida.

Em comparação com os energymetters revisados na literatura, o sistema desenvolvido nesse trabalho está integrado a bicicleta elétrica e tem pequeno peso, de modo que não interfere muito nas condições do ciclista. Assim, experimentos podem ser conduzidos de forma conveniente. Por outro lado, nos testes realizados nos sistemas estudados na literatura, o ciclista precisava carregar um laptop.

Abstract

The main categories of electric mobility are elevators and electric vehicles. The elevators are a reliable and convenient way of transport. The electric vehicles have been gaining global interest in the last years due to their capacity to reduce GHG and to their high efficiency. The electric bicycles are a convenient way of transport because they are small, reduce traffic jams and improve the rider's health.

The goal of the dissertation is to develop an embedded system capable of measure voltage, current and power from the electric battery of an electric bicycle, send data to a smartphone by Bluetooth, and from the smartphone to a computer using MQTT network. The system had to be developed under the IEEE1451 Standard.

The methodology used was literature review of energymetters developed around the world; study of IEEE1451 standards; design of the system based on sensors, actuators and the communication between them and the processor; hardware development using mbed microcontroller, INA260 sensor, HC05 Bluetooth module and a SD breakout board; firmware development based on pooling, interruption, state machines and TEDS; Smartphone app development based on Bluetooth and MQTT networks; Management application development based on node connections using software NodeRed; Finally tests performance and results analysis.

The system developed here may contribute to perform tests to verify the electric consumption, current, voltage and power of eBike in different conditions such as varying the cyclist mass, eBike mass, acceleration conditions, speed conditions, air conditions, slope degree and routes. This may allow to discover the most economical routes or trace the user profile. In this way, together with air pollution gas tracking and Photovoltaic eBike charging, this project may contribute to urban mobility sustainability and better quality of life. In comparison to the energy meters revised in the literature, the system developed here is integrated with the eBike and has lightweight. This makes the experiments easy to be conducted.

Keywords

Electric Mobility; Electric Bicycles; Bluetooth; MQTT; IEEE1451; Energy meters.

Folha em branco

Table of Contents

Folha em branco

List of Figures

Folha em branco

List of Tables

Folha em branco

List of Acronyms

eBike	Electric Bicycle
MQTT	Message Queuing Telemetry Transport
FEDER	Fundo Europeu de Desenvolvimento Regional
POCTEC	Programa V-A Espanha-Portugal
TEDS	Transducer Electronic Data Sheet
NCAP	Network Capable Application Processor
TIM	Transducer Interface Module
TII	Transducer Independent Interface
NI	Network Interface
IML	Instrumentation and Measurement Laboratory
TCh	Transducer Channel
IoT	Internet of Things

Folha em branco

Chapter 1

Introduction

1.1. Problem Context and Motivation

Electric Mobility is the term used to refer to all the electric equipment and systems used to transport human beings from one point to another. Electric mobility is essential for efficient use of energy for mobility, global warming, air pollution in cities and employment shifts in important energy sectors due to structural changes in the supply chains of energy sectors. [32]

The main categories of Electric Mobility are elevators and electric vehicles. Elevators ease human access and material transportation in a fast and reliable manner and they have become a necessary tool for the improvement of life in the urban environment. They are present in private or public buildings, inside shopping centers and in sky-scrapers, in airports, in rail and metro stations and in tourist places. [35]

Electric vehicles are vehicles that are driven by an electric motor. There are various kinds of electric vehicles such as cars, aeroplanes, scooters, buses and bicycles. The electric vehicles were invented at the beginning of the XIX century. Still, they were not used on a large scale during the XX century due to a combination of factors, including the invention of the Self-Starter to combustion vehicles and the low range of the electric batteries. [1],[2],[3],[4], [5]

In the XXI century the interest in electric vehicles returned due to environmental and economical issues. The environmental issue is that electric vehicles do not emit pollutant gases to nature. The economical issue is that electric vehicles have a high efficiency (approximately 80%). [1],[2],[3],[4], [5]

This mastering Dissertation was developed within the Urban-Air Project. FEDER - Fundo Europeu de Desenvolvimento Regional (European Resources to Regional Development) cofounded the Urbain-Air project through the POCTEC - Programa V-A Espanha-Portugal (Portugal-Spain V-A Program) to improve the urban environment and to reduce the air pollution through sustainable mobility solutions in cities of both countries. [6]

Some ways to improve urban air quality are to track the city pollutant gases such as ozone and carbon dioxide, monitor the electric bicycle electric parameters, such as energy, power, current and voltage, and detect if the cyclist falls. Therefore, in Portugal, the Urban-Air project has 50 electric bicycles, four charging stations integrated into photovoltaic generation panels, strategically distributed through Covilhã city. The main aim is to develop a bicycle sharing system using smartphones where the bicycles must be instrumented with power sensors, gas sensors and detect fall. [6]

Electric bicycles instrumentation is compatible with the family of standards IEEE1451, what allows interoperability. The IEEE1451 family of standards results from seven standards founded by the IEEE Standard Association to standardize smart transducers. [7]

1.2. Goals

The dissertation aims to develop an embedded system capable of measure voltage, current, and power from the electric battery of an electric bicycle. The microcontroller mbed NXP LPC1768 connects with Bluetooth using HC05 module. Current, voltage and power values acquired by an INA260 sensor are recorded in a MicroSD module and simultaneously sent to a smartphone. Data is sent to a database using Bluetooth and MQTT networks.

The specific goals are:

- Develop an embedded system to realize data acquisition of voltage, current and power of the electric battery of the electric bicycle using the microcontroller MBED and the INA260 sensor;
- Transmit voltage, current and power values to a smartphone using the Bluetooth HC05 module, while the cyclist is pedalling;
- Store voltage, current and power in a microSD card using a MicroSD module;
- Transmit voltage, current and power data to a data base stored in the cloud using MQTT;
- Analyze the data on a computer.

1.3. Methodology

- Study the family of standard IEEE1451 and how to apply in the development of the system;
- Study the INA2060 for voltage, current and power values acquisition;
- Elaborate the working scheme of the INA260 according to the IEEE1451;
- Develop and test an embedded system to be programmed in C++ to realize data acquisition of voltage, current and power of a power resistance using the microcontroller MBED, protoboard and a portable computer;
- Produce the necessary Transducer Electronic Data Sheets (TEDS);
- Develop and test an embedded system to send voltage, current and power data to a portable computer to be programmed in C++ using Bluetooth HC05 module;
- Develop and test an embedded system to store voltage, current and power data in a MicroSD module to be programmed in C++;
- Integrate the three previous embedded systems in one;

- Design and assembly and printed circuit board of the integrated embedded system;
- Test the integrated embedded system to measure electric parameters of the electric battery of the electric bicycle using a computer as the data receiver;
- Develop an application to work as an interface between the web system/data base and the integrated embedded system programmed in JAVA. This means that the application must be capable of receive trigger commands from a web system, send the trigger commands to the integrated embedded system, receive data from the integrated embedded system and send data to the database;
- Test the whole system in an electric bicycle;
- Create charts in convenient formats to analyze data;
- Analyze data and results.

1.4. Structure of the Dissertation

This first chapter introduces the dissertation's context: electric mobility, electric bicycles, Urban-Air Project and IEEE1451. The rest of the dissertation is structured as follows. Chapter 2 talks about Electric Mobility. A State-Of-Art of different electric systems used in urban mobility, their implementation and applied standards will be done. The development of the embedded system is presented in Chapter 3. The user interface developed to the smartphone is described in Chapter 4. The Management Application is described in Chapter 5. The overall system is described in Chapter 6. Validation tests and results analysis are described in Chapter 7. Finally, Chapter 8 provides the conclusions. References and Appendices are listed in the end.

Chapter 2

Electrical Mobility

2.1. Electric Vehicles

Electric motors drive electric vehicles. The electric motor is powered by a set of rechargeable energy storage devices, which generally consists of electrochemical accumulators, using a power converter. The energy stored in the batteries comes from another source of primary generation, such as electric power plants, solar energy and wind energy.

The output of the electric motor depends on the current and voltage applied to it. Therefore, the maximum torque is available at all the speeds of the motor. This chain of energy processing does not involve the emission of hazardous gases to the environment; they are called "Zero Emission" and ensure a low noise level. [21]

The first electrical vehicle was created in 1842 by the Scottish Rober Anderson. This vehicle could drive at an average speed of 6.4km/h and used primary batteries of zinc. Between 1842 and 1920 the electric vehicles were used mainly in France, England and United States. The electric vehicles brought attention to the consumers because they did not emit noise, have vibrations, or smell, and did not need gear. [22], [11]

The year of 1920, although, marked the beginning of a period of shadows to the electric vehicles. Among others, the reasons were:

- Due to low cost, combustion vehicles attracted more interest;
- Disinterest in electric vehicles due to their low range of operation;
- Infra-structures to charge electric vehicles were not available.

In this way, the interest in develop and produce electric vehicles was resumed only in 1947 from i) the increase of the price of gas; ii) the invention of the transistors in 1925, enabling the power regulation of electric motors and consequent increase rheostat efficiency. [22]

Today, the electric vehicles are gaining global importance because of their high energy efficiency and because of their potential to reduce carbon emission, mainly when associated with primary electricity with low CO2 emission. [12] The most basic configuration of a purely electric vehicles is composed of a battery, a power converter, electric motor and gears, organized according to Figure 1.

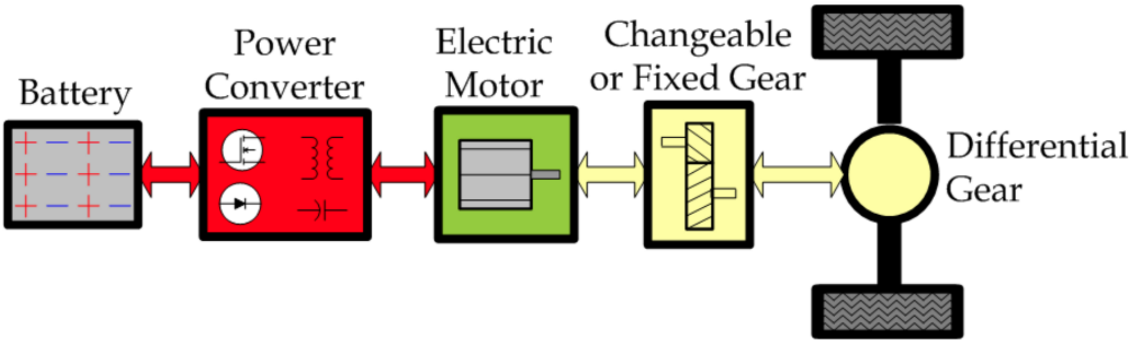

Figure 1: Basic Architecture of purely electric vehicles. [14]

The battery stores electrical energy. The power converter converts power from DC into AC if the vehicle uses AC motors or regulates the current DC if the vehicles uses DC motors. The electric motors AC or DC generate mechanical energy to transmit to the gears. The gears transmit mechanical power to the wheels. Next, a detailed description of each of the subsystems will be provided.

The energy storage system of the electric vehicles is generally composed of a set of hundreds of fuel cells grouped in series or parallel to satisfy the power demand of the electric system of each vehicle. The average power of electric vehicles varies from 30 to 120 kW. [15]

Some of the equivalent circuits used in the battery modelling of electric vehicles are a) one power source in series with one resistor; b) one power source in series with a connection of a resistor with a capacitor in parallel, in series with another resistor and c) one power source in series with a connection of a resistor with a capacitor in parallel, in series with another connection of a resistor in parallel with a capacitor, in series with another resistor.

The storage capacity of the rechargeable batteries is characterized by two parameters: Energetic Density by volume unit (Wh/l) and Energetic Density by mass (Wh/Kg). A comparison between storage capacity from the most used kinds of rechargeable batteries in the world in shown in Figure 2.

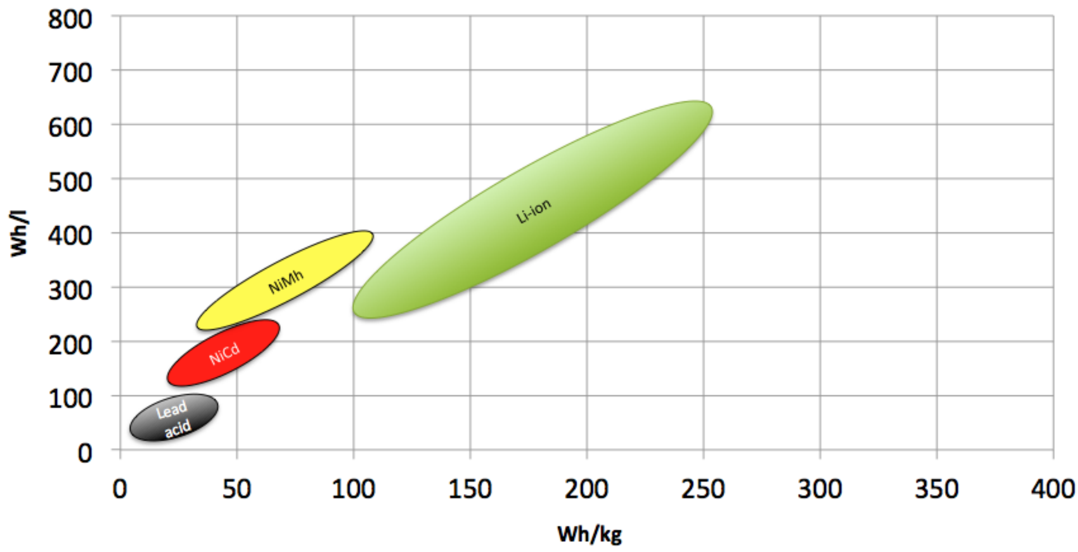

Figure 2: Energetic Density of the most used rechargeable batteries in the world.

From the chart analysis, it can be seen that from the batteries considered, the Li-Ion battery is the one that presents the best energy storage capacity. Because of this reason, a considerable amount of electric vehicles use this technology.

Some of the main characteristics of the Li-Ion batteries are:

- They have high voltage (3.6V), which is three times higher than NiCd or NiMh (1.2V);
- They have high energy density by volume unit (250 to 650 Wh/l) and by mass (100 to 250 Wh/Kg);
- They have a quick charging;
- They have a long life cycle;
- They have a high cost;
- They need an specific design;
- They need a protection circuit. [16]

The power converters can be DC-DC, DC-AC, unidirectional or bidirectional. Two examples of power converters are Full bridge DC-DC bidirectional converter and Two level DC-AC inverter converter connected to a permanent magnet brushless synchronous motor.

The electric motors receive electric energy from the battery through controllers and convert them to the needed mechanical energy to turn their axis. There are four kinds of motors viable to electric vehicles: Conventional DC Motor, Permanent Magnet Brushless DC Motor, Induction Motor and Variable Reluctance Motor. [17], [23], [27]

The conventional DC motors present orthogonal arrangement of armature and field, average efficiency between 85% to 95% under full load, maximum average rotation between 9000 and 15000 rpm and simple control. [19], [13] The Permanent Magnet DC Brushless Motors has high power density, good torque characteristics, wide velocity range, low maintenance and high efficiency, but they have complex control. [25], [26], [28]

The Induction motors present low cost, high efficiency, high reliability, easy cooling, they are robust and have the feature of regenerative break, where during the brake, the motor starts to operate as a generator and transmits energy back to the batteries. This can increase the vehicle range by 8% to 25%. [18], [17], [27], [25]

The Variable Reluctance Motors present good torque characteristics associated to velocity, low production cost, high efficiency and they are light. However they have high torque ripple and vibration and noise problems. [23], [24]

The gear system of electric vehicles varies according to the number of motors (from one to four) and from the traction system (frontward, backward or 4x4) used and can have fixe or changeable and/or differential gear or none of them.

The fixed gear has the function to transmit mechanical energy from the motor to the differential gears. They can be avoided if an efficient control system is used. On the other hand, the differential gears transmit mechanical energy from the fixed or changeable gears to the wheels when the electric vehicle uses one or two motors. The use of four motors connected straight to the wheels and an efficient control system dispenses the utilization of any gear, significantly decreasing the mechanical losses. [14], [19]

According to Martin Ebehard, cofounder of Tesla Motors, electric cars are more efficient than combustion engine cars, hydrogen cars, ethanol cars and biodiesel cars. Martin Ebehard studies take into consideration the production efficiencies, the fuel energy efficiencies and the vehicle efficiencies. The results from his study show that:

- The most efficient combustion engine car consumes 2.41MJ/km, while the most efficient electric car consumes 1.14MJ/km. In this way the most efficient combustion engine car has a consumption two times higher than the most efficient electric car;
- With 1MWh of electric energy, the most efficient hydrogen car would run for 1769 Km, while the most efficient electric car would run 5760 Km. In this way the most efficient electric car would run three times longer than the most efficient hydrogen car;
- With 1 ton of biomass, the most efficient ethanol car would run for 3520 Km, while the most efficient electric car would run for 5760 Km. In this way, the most efficient electric car would run a distance 64% higher than the most efficient hydrogen car;
- With 3.8 litres of biodiesel, the most efficient biodiesel car would run for 60.8 Km, while the most efficient electric car would run for 104 Km. In this way, the most efficient electric car would run a distance 74% higher than the most efficient biodiesel car. [20]

2.2. Electric Bicycles

Next, the text explains the main eBikes' categories, their history and the benefits of eBikes. Electric Bicycles can be classified in general, by function, by structural features and by purpose. In general they can be classified as pure, power-assisted or pure and power-assisted. The pure eBikes are driven by a motor without the user's pedaling force. The user controls a handle bar throttle that transfers electric power from the battery to the motor.

The power-assisted eBikes have a motor that assists the user with power only when he pedals. This category of eBike is also called a pedelec and has a sensor to detect the pedaling speed or force. Figure 3 shows an example of a Pedelec. [29]

Figure 3: Example of Pedelec. [29]

The power-assisted and pure is an eBike that works either in the pure mode or power-assisted mode. An example of a power-assisted or pure eBike is showed in Figure 4.

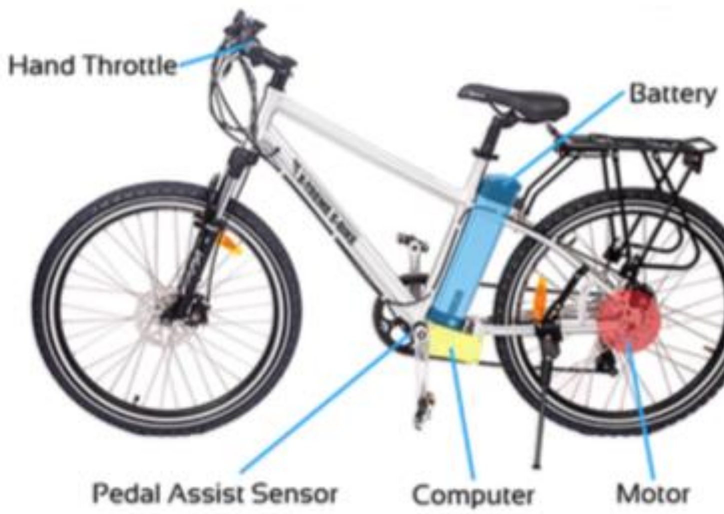

Figure 4: Example of a power-assisted or pure eBike. [29]

In the classification by function the eBikes can be autonomous, manned or unmanned. Structural frames they can be classified according to the frame types such as fixed frame and folding frame, according to the frame materials and the number of wheels.

On the other hand, in the classification by purpose they can be classified in hill-climbing, urban transport and travelling. eBikes used for hill climbing usually have a light and secure frame and strong motor power to help the rider resist the hilly terrain. eBikes used for urban transport usually have high mobility features such as smaller size and folding frame. Travelling eBikes can have storage compartment for necessities.

The first versions of eBikes were produced in the 1890's, as shown by U.S. patents. In 1895 Ogden Bolton invented and eBike integrated with a six-pole DC hub motor in the rear wheel to handle a 100A, 10V battery. Then in 1897, Hosea W Libbey invented an eBike with a double motor. In 898, Gordon John Scott designed an eBike that used a generator instead of a battery

From 1900 to 1990, the eBikes continued their development and in 1900 Albert Hänsel invented an eBike with a non-hub motor, mounted in the centre of the eBike. In this period, also Thomas M. McDonald developed the first eBike with a motor mounted in the front wheel. From the 1990s to the end of the twentieth century, the eBikes had many significant improvements such as solar charging apparatus and manpower torque detection.

From the early 21st century to the present, the eBikes continue to be investigated and developed with advanced technologies to increase their performance. An example of those technologies was the invention of an eBike that could be folded and wheeled by hand.

Some of the benefits of eBikes are: i) they are usually small and can transverse various terrains such as hilly, rugged and flat; ii) move quickly in crowded cities, helping to reduce traffic jams; iii) usually cheaper than other electric vehicles; iv) in most countries they do not require insurance, road taxes or license to ride; v) have low maintenance costs; vi) can improve the health of the rider. [29]

2.3. eBike Energy Meters Developments

This section reviews some developments in eBike's energy meters found in the literature. Section 6.2. will compare these developments with he system developed in this dissertation. In [30], HUNG and LIM evaluate electric bicycles' dynamic and electric consumption under different operation conditions and structural parameters. First, they simulate the electric bicycle dynamic in Matlab considering Newton's second law, propulsion force, rolling resistance force, wind resistance force, and motor and battery models.

Then they analyze the results and perform an experimental study with an eBike equipped with a lithium-ion battery, a DC electric motor, signal processing and measurement devices, a photoelectric sensor, an Electronic Control Unit, a Laptop and a voltage input module. The setup of the experimental study is shown in Figure 5.

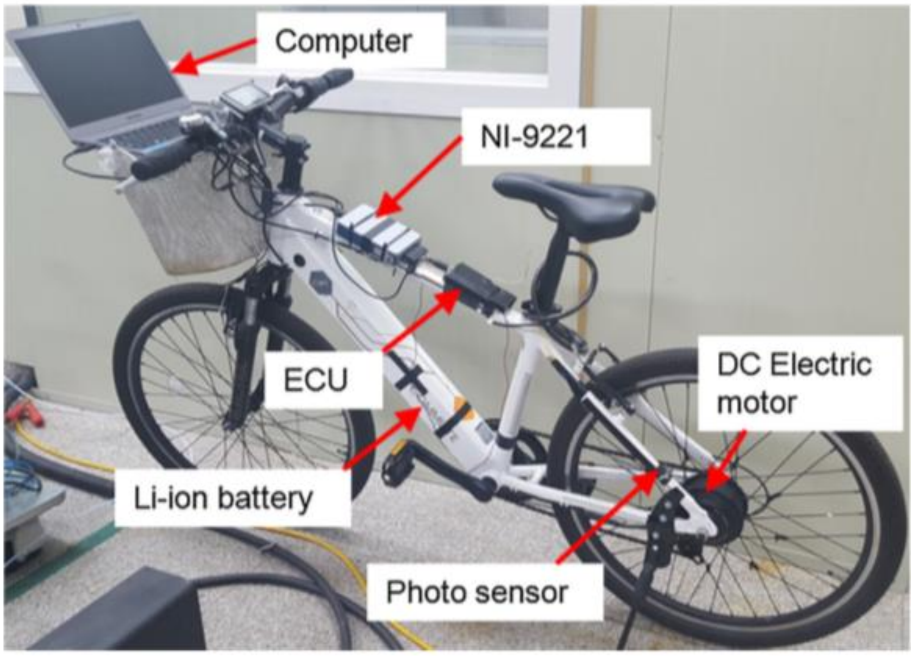

Figure 5: Experimental Study Setup.

The results of the simulation are compared with the experimental study present the same trends. The conclusion was that the dynamic performance and electric consumption of the eBike are improved by reducing the bicycle mass, wheel radius and increasing the sprocket ratio.

In [31], HUNG and LIM use the same methodology of the paper [30], but now they evaluate the dynamic performance and electric consumption of the eBike under another conditions. The conclusion was that as the velocity of an eBike increases, the air density reduces, increasing the distance run. Also, the increase of air density increases the electric consumption of the eBike. The increase of slope grade reduces the eBike velocity and, therefore, distance. Also, the increase of the slope grade increases the eBike electric consumption.

In [32], MENDES et all quantify the energy consumption, trip travel and driving dynamics of electric and conventional bicycles and motorcycles on specific routes in Lisbon, Portugal. The experimental setup used was a GPS that could monitor location, vehicle speed and barometric altimetry, a National Instruments DAQ with voltage and current probes and a laptop. The equipment was carried inside a backpack and weighted 4Kg. The vehicles tested and the experimental setup used in the experiment are shown in Figures 6 and 7.

Figure 6: Vehicles tested in [32].

Figure 7: Experimental Setup in [32].

The energy consumption was calculated using two methods: Vehicle-specific Power, which combines speed, acceleration and road grade, and measured by the National Instruments DAQ. It was discussed that comparing the electric bicycles with the conventional ones. The electric bikes allow travelling at high speeds on higher slopes. The average speed of the Electric Bicycle was 16% higher than the one in conventional bicycles. The two methods of energy consumption estimation were compared and the differences ranged from -7.3% to 18.2%.

In [33], the electric-assisted bike's power demand and environmental analysis were made in Naples, Italy. The methodology of power estimation was based on the experimental kinematic parameters including the power to exceed the air strength, the power necessary to overtake the hill, the power required to overcome the rolling resistances and the power to overcome the vehicle inertia during the acceleration phases. The technical details of the pedelec used in the experiment are shown in Table 1.

Table 1: Technical Characteristics of the tested pedelec.

Total Weight	26Kg
Motor Type and Power	Brushless DC Machine, 250W
Motor Assembly	Hub
Motor Placement	Real Wheel
Levels of Electrical Assistance	Four Modes
Battery Type	Li-Po, 48V, 10Ah
Charging time	5h
Cycles of Charge/Discharge	Up to 800
Tire Type	26x2.3^{n}

In the analysis, the significant factors describing the driving range of the electric bicycles were the road orography (gradient of the hills), the weight, the speed of the bike and the energy provided by the cyclist. With a high total weight, the total energy for a specific trip increases. Also, a steep road requires more energy than flat road. The frequency of start and stops is crucial to estimate the required electrical power. The results presented are of driving range as a function of the four working modes of the eBike. The driving range decreases with the increase of electrical assistance.

In [34] experimental studies of the dynamics and power of an electric bike are done. The acceleration characteristics were measured only with electric traction drive and with traction drive and muscular cyclist effort. The dynamics of the acceleration at different degrees of charge of the accumulation battery were estimated. Also speed, voltage, current, distance travelled, and energy were recorded.

The results showed that when the acceleration was measured only with electric traction drive, the acceleration time to a speed of 5m/s falls as the battery discharges. This is due to a decrease in the output power of the battery; as the voltage decreases, it discharges and the internal resistance increase. Results also showed that acceleration with electric traction drive and pedalling significantly reduces the energy expended for acceleration to a given speed. At the same time, the path traversed by the electric bike during the acceleration time increases.

2.4. Standard IEEE1451

The IEEE1451 standard family is composed of a set of smart transducers interface standards that define a set of common, open and independent communication interfaces to connect transducers to microcontrollers, instrumentation systems and control and field networks. This family has protocols to applications wired and wireless to distributed monitoring and control. [7]

A smart transducer integrates an analogue or digital sensor or actuator to a processing unit and a communication interface. The basic architecture of a smart transducer is shown in Figure 8.

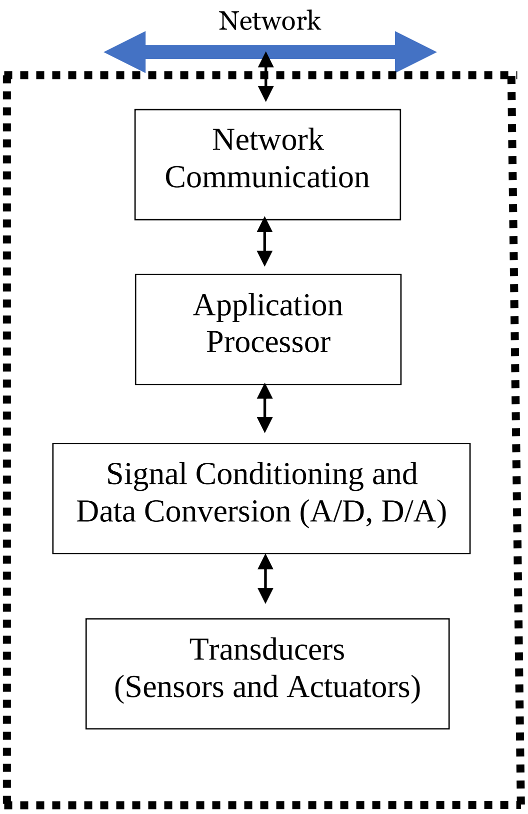

Figure 8: Basis architecture of a smart transducer. [7]

The IEEE1451 family of standards propose smart transducer standardization, describing the primary architecture, with the addition of RAM and ROM memories to the transducer containing general information about the transducer. Figure 9 shows the architecture of a smart transducer according to IEEE1451.

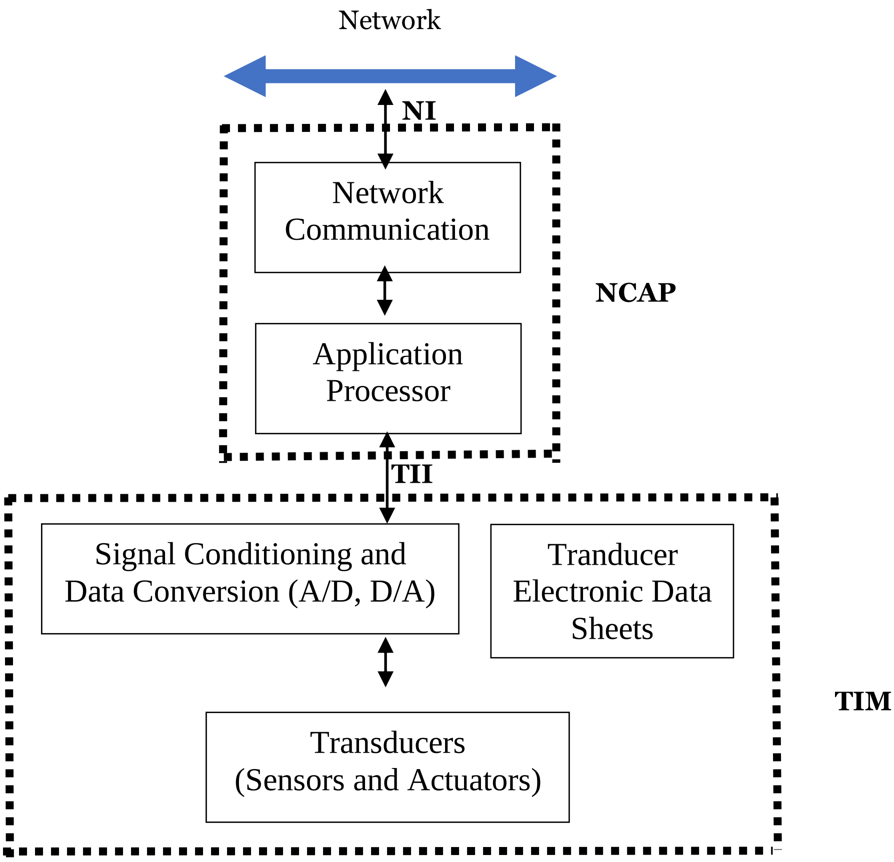

Figure 9: Smart transducer architecture according to IEEE1451. [7]

The Network Capable Application Processor (NCAP) is a network node that performs network processing and communication functions. The Transducer Interface Module (TIM) consists of signal conditioning, data conversion and a set of sensors and actuators with a maximum of 255 units. The Transducer Independent Interface (TII) defines a communication way and a protocol to information transfer between the TIM and the NCAP using operations, messages and read, write and read/write answers. The Transducer Electronic Data Sheets have RAM and ROM memories with the transducer information. [7]

IEEE1451.0

The IEEE1451.0 defines a set of standard functionalities, commands and TEDS to the IEEE1451 family. The functionalities are independent of the physical media between the transducer and the NCAP. They include essential functions of reading and writing to the transducers, reading and writing to the TEDS, configuration, control, and operation of the TIM. The commands are used to access the sensors and actuators, while the TEDS are divided in mandatory and specific. Some mandatory TEDS are:

- TransducerChannel TEDS;
- User's transducer name TEDS;
- PHY TEDS.

Some specific TEDS are:

- Calibration TEDS;
- Frequency Response TEDS;
- Transfer Function TEDS;
- Text based TEDS;
- End user application specific TEDS;
- Manufacturer-defined TEDS.

IEEE1451.1

The standard defines an object model and specifies the interface between the smart transducer and the network components. This standard focus mainly on the communication between different NCAPs and between an NCAP and another nodes of the system. [7]

The interface specifications in software are made through classes that form blocks, components and services to the transducer model. The object model of the smart transducer with the network encapsulates the details of the hardware implementation through a simple programming model. On the other hand the network service interfaces encapsulate the details of different network protocol implementations through a small set of communication methods. [8]

IEEE1451.2

The standard defines an interface between the transducers and the NCAPS and TEDS to the configuration peer-to-peer. This standard has a communication line based on the SPI interface and hardware lines for control and timing, with a total of ten lines. The standard defines the functionalities of the TIM, here called STIM, and the mode how they become accessible to the NCAP, and the TII. [7], [9]

IEEE1451.3

The standard defines an interface between the NCAP and the transducer and TEDS using a multi-drop communication protocol. This standard allows transducers organized in a multi-drop transducer network arranged with nodes sharing one pair of wires. [7]

IEEE1451.4

This standard defines a mix-mode interface to analogue transducers with analogue and digital operation modes. Besides that, a TED was added to the traditional two wired sensors with continuous current excitation containing a FET amplifier. [7]

The mix-mode interface may work through two levels. In the first level the analog signals and the data share the same bus, saving wires. In contrast, the analogue signals and the data are transmitted through separated connections in the second level. Examples of devices that use the first level are accelerometers and microphones. Examples of devices that use the second level are high impedance sensors, sensors and actuators of kind 4-20mA, RTDs and thermistors. [9],[10] The data transfer is implemented through four commands: reset, write-one, write-zero and read. The transfer always begins with reset, followed by a ROM command composed of a sequence of reads and writes, followed by a RAM command also composed of a sequence of reads and writes. [10]

IEEE1451.5
This standard defines and specifies specific radio protocols to a interface between wireless transducers and NCAPS and it defines TEDS to wireless transducers. In this case the NCAPs have one or more wireless radios (802.11, Bluetooth and Zigbee, for example) capable of communicating with one or more interface modules of wireless transducers (WTIM) or with an external network. Else each WTIM has a wireless radio, signal conditioning, A/D or D/A converters and the transducers. [7]

IEEE1451.6
The standard defines an interface between the transducers and the NCAPs and TEDS using the high-speed CANopen interface. This standard defines the TEDS 1451 mapping to inputs of the CANopen dictionary, communications messages, process data, a communication parameter and diagnose information. [7]

IEEE1451.7
This standrd defines an interface and a communications protocol between transducers and RFID systems. The goals of the standard are: provide interface and methods to the interface between transducers and RFID; and data communication inside the RFID systems. [7], [9]

Chapter 3

eBike Equipped With a TIM

3.1. TIM Structure

In order to develop the TIM of the system, first, the IEEE1451.0 Standard was studied, mainly the chapter 5, since that to develop the system under the standard rules was one of the project's requirements. Also, two articles were revised. [1],[2] Then, based on the standard, the working diagrams of the system were drawn in order to fulfill the requirements of the goals of the work, discussed in section 1.2. The diagrams are shown in Figures 10, 11, 12, 13, 14 and 15.

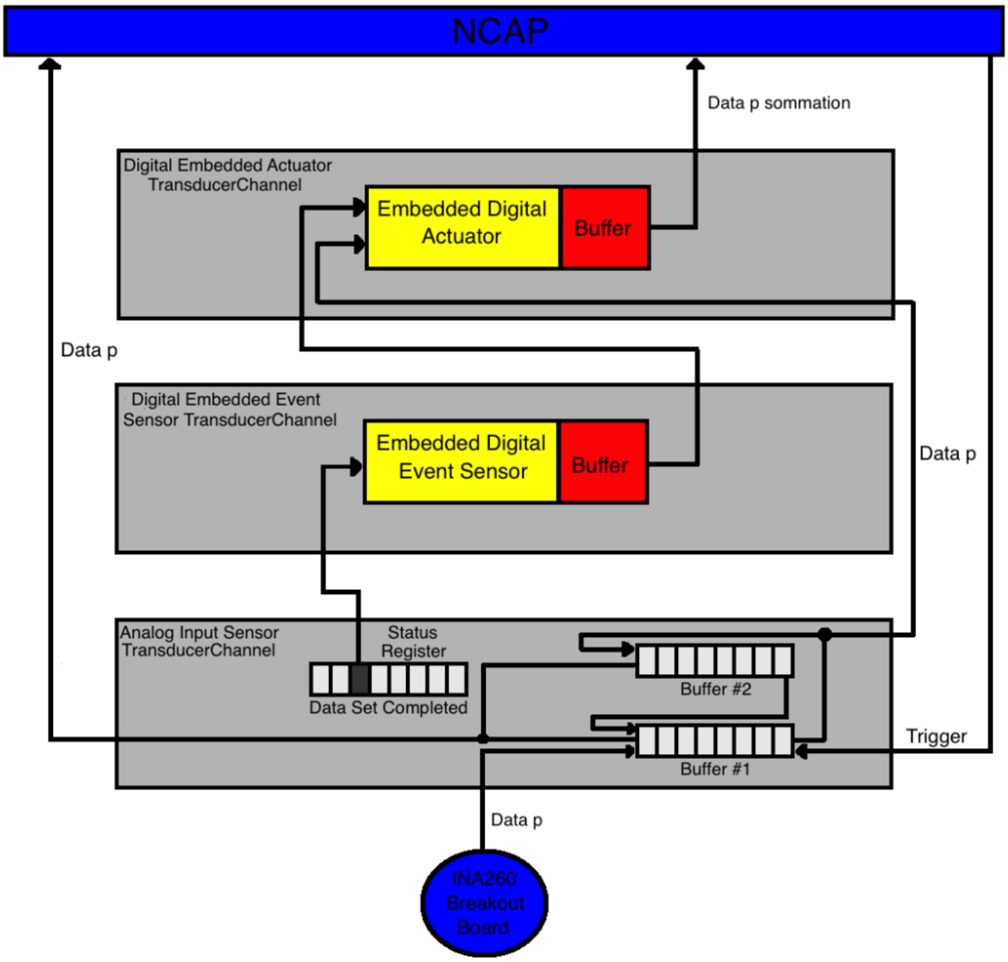

Figure 10: Working diagram of power data transfer between NCAP and INA260 sensor.

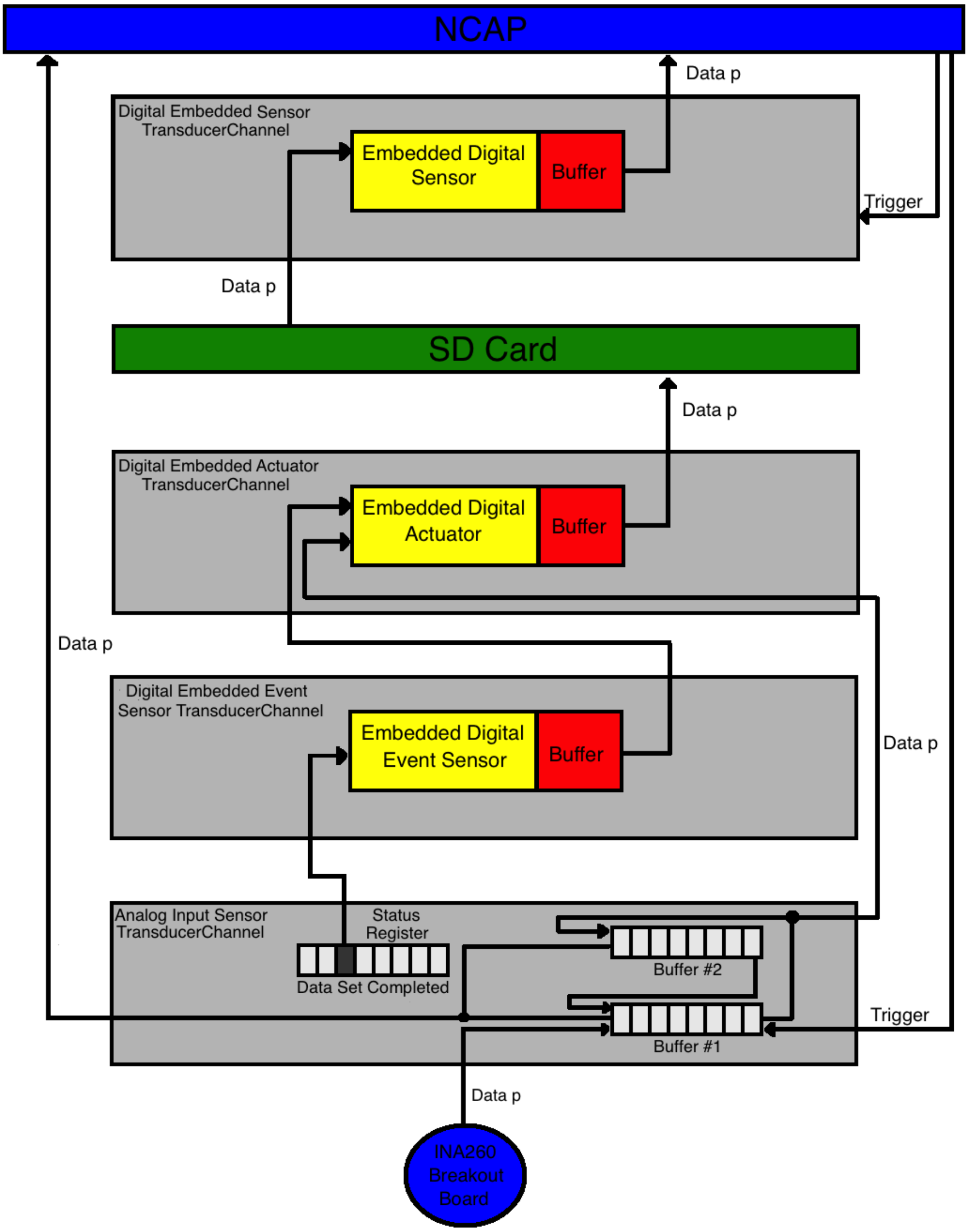

Figure 11: Working diagram of power data transfer between NCAP, INA260 sensor and SD Card.

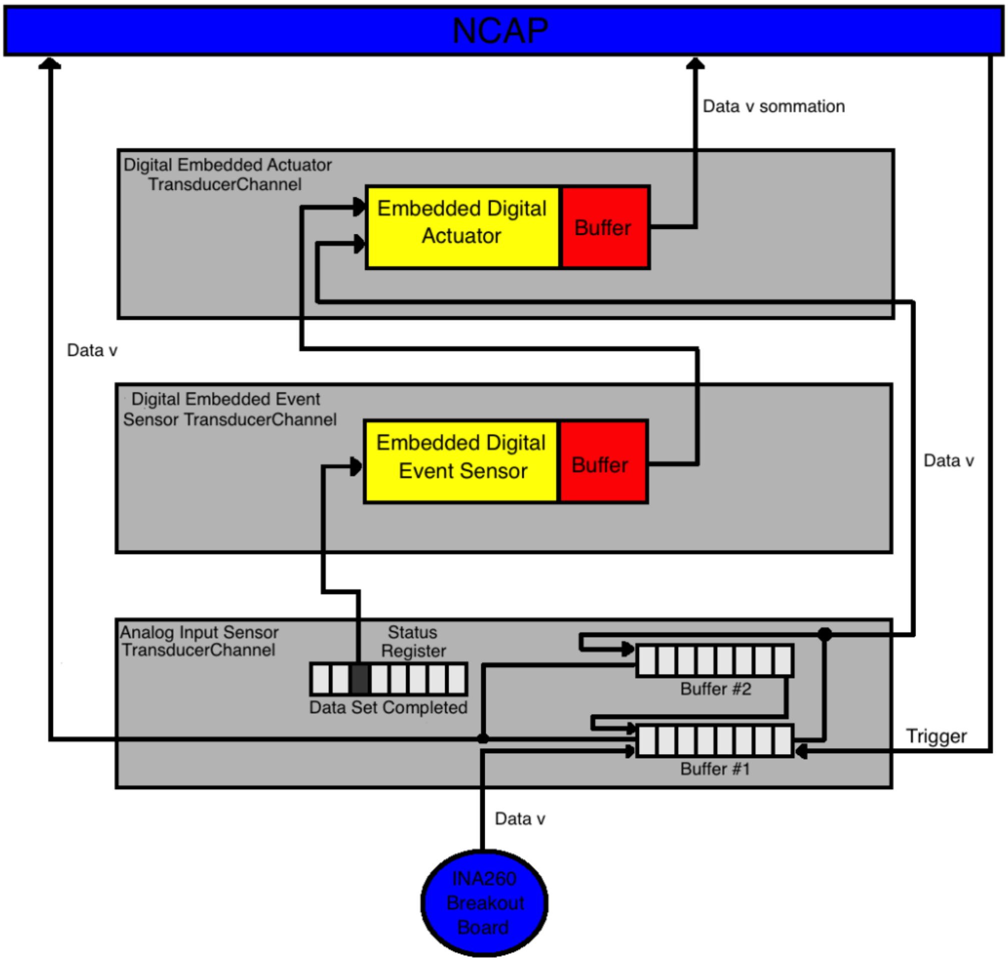

Figure 12: Working diagram of voltage data transfer between NCAP and INA260 sensor.

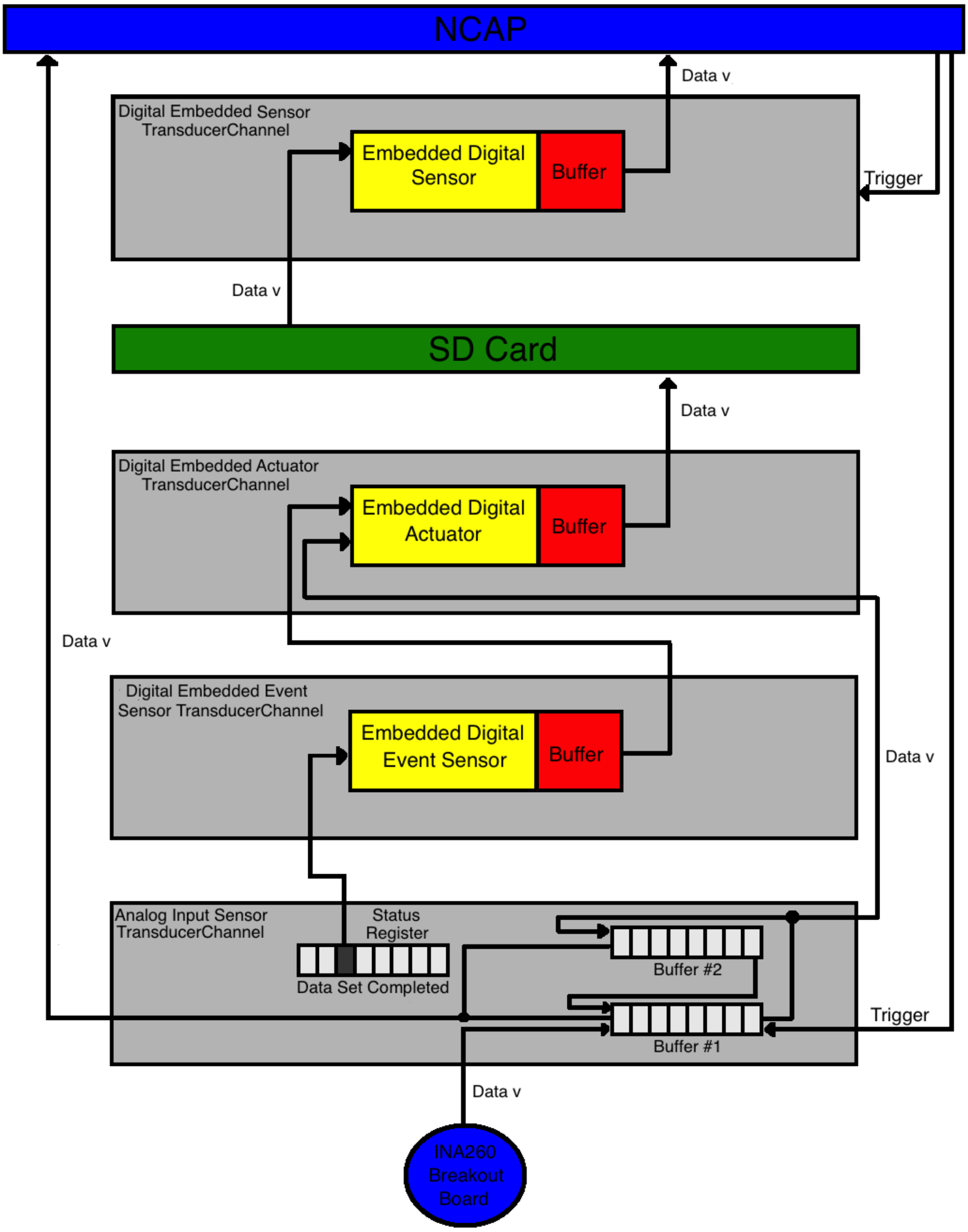

Figure 13: Working diagram of voltage data transfer between NCAP, INA260 sensor and SD Card.

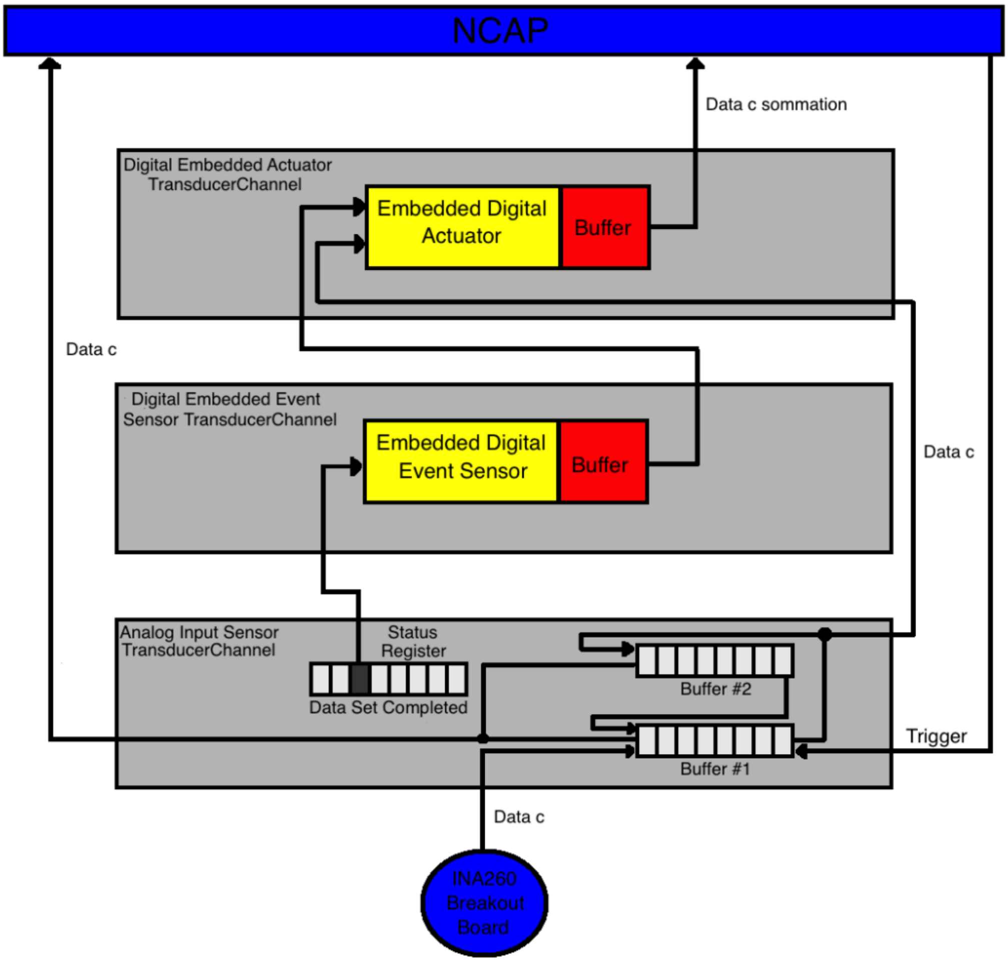

Figure 14: Working diagram of current data transfer between NCAP and INA260 sensor.

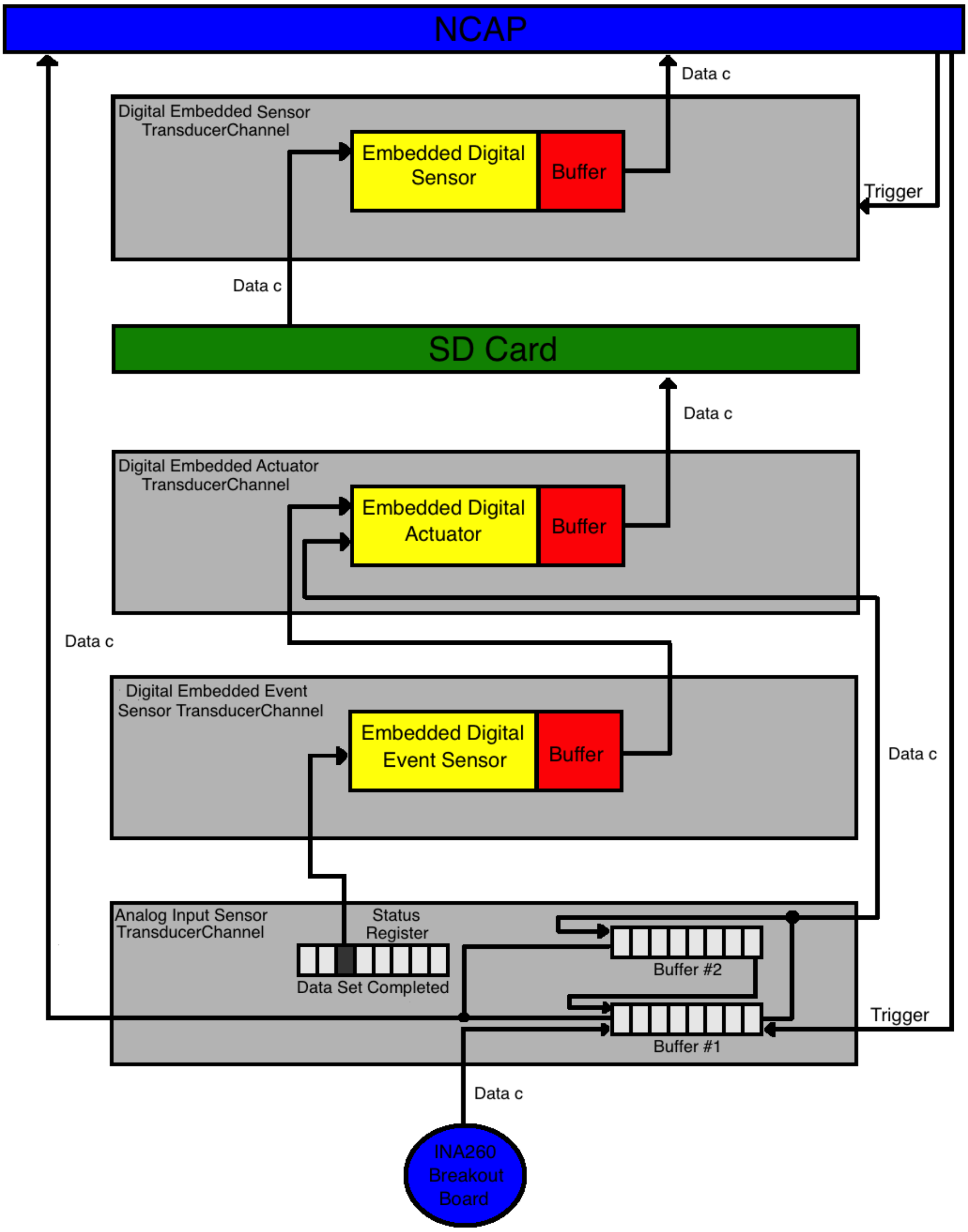

Figure 15: Working diagram of current data transfer between NCAP, INA260 sensor and SD Card.

Figure 10 represents the exchange of information between the NCAP and the INA260 sensor for power data. According to the standard IEEE1451, the TransducerChannel is "a transducer and all of the signal conditioning and conversion components associated with that transducer." In this way, the power sensor is considered and ANALOG INPUT SENSOR TRANSDUCERCHANNEL that has two buffers.

The data acquisition works in the Continuous Sampling Mode. In this mode, the sensor performs analogue voltage and current acquisition from the output of the electric bike battery once it receives a trigger from the NCAP. Then it converts the two analog signals in digital signals and calculates instantaneous power during time intervals Δt.

The instantaneous power data are stored in one of the buffers of the ANALOG INPUT SENSOR TRANSDUCERCHANNEL. The data transfer works in the Only When Commanded Mode. In this mode, the data are transmitted to the NCAP only when the NCAP sends the TransducerChannel Dataset Segment Read Command.

However, when the first buffer of the ANALOG INPUT SENSOR TRANSDUCERCHANNEL is complete, the Register Status Bit is set indicating that the data set is complete. Once the Register Status Bit is set, it triggers the DIGITAL EMBEDDED EVENT SENSOR TRANSDUCERCHANNEL, whose bit has changed from 0 to 1 to indicate the event's occurrence of the event "first buffer full". This sensor is called EMBEDDED because it is internal of the TIM and does not deal with samples straight from the sensor.

The EMBEDDED EVENT SENSOR TRANSDUCERCHANNEL triggers a DIGITAL EMBEDDED ACTUATOR TRANSDUCERCHANNEL that takes instantaneous power samples $p_1, p_2, p_3, \dots, p_n$ acquired in time intervals $t_1, t_2, t_3, \dots, t_n$ from the full buffer and sum them. This sum of the power data indicates the cumulative power during the time t_n. This sum is sent to the NCAP.

The cumulative power during the time t_n can be multiplied to Δt in the NCAP to obtain the energy spent during the time t_n. As soon as the first buffer is filled, the sensor keeps acquiring new data and this time stores it in the next available buffer of the ANALOG INPUT SENSOR TRANSDUCERCHANNEL (the second buffer).

When the two buffers of the ANALOG INPUT SENSOR TRANSDUCERCHANNEL are full, the data from the first buffer must be discarded whether if they have been transmitted to the NCAP and this buffer starts to acquire new data. In this way, it must be guaranteed that the time of data transmission from one buffer to the NCAP is lower than the data acquisition time of one buffer in case of the NCAP request data transmission after the filling of the first buffer.

Figure 11 represents the exchange of information between the NCAP, INA260 sensor and the SD Card for power data. The working principle of this diagram is similar to the Diagram of Figure 10. The ANALOG INPUT SENSOR TRANSDUCER CHANNEL shown in Figure 11 is the same as one shown in Figure 10. In Figure 11, the power sensor does the acquisition of voltage and current when it receives a trigger from NCAP. Then it calculates the power and fills buffers one and two in the same way as in the Figure 10.

When one buffer is full, a Status Register bit is set and it triggers the DIGITAL EMBEDDED EVENT SENSOR TRANSDUCERCHANNEL. This even sensor triggers the DIGITAL EMBEDDED ACTUATOR TRASDUCER CHANNEL and this time this actuator takes data p and send to the Sd Card. This is used to write to the SD Card. On the other way, if the NCAP wants to read from the SD Card, it sends a trigger to one DIGITAL EMBEDDED SENSOR

TRANSDUCERCHANNEL. Then the digital sensor reads power data from the SD Card and sends it to the NCAP.

The working principle of Figure 12 is analogue of Figure 10;, the only difference is that the data now is voltage. In the same way, the working principle of Figure 13 is analogue of Figure 11; the only difference here is that the data is voltage. Figure 14 has the same principle of Figure 10 but with current data and Figure 15 has the same principle of Figure 11 but with current data.

3.2. Hardware Design

The hardware modules and components were chosen through a discussion between the student and the advisor, and research. The hardware chosen to develop the system were the microcontroller mbed NXP LPC1768, sensor INA260, HC05 Bluetooth module and SD Card breakout board.

The mbed NXP LPC1768 was chosen because it is suitable for IoT solutions, it supports low power and low energy consumptiom. A picture of the microcontroller selected is shown in Figure 16 and a scheme of its inputs and outputs pins is shown in Figure 17.

Figure 16: Microcontroller mbed NXP LPC1768.

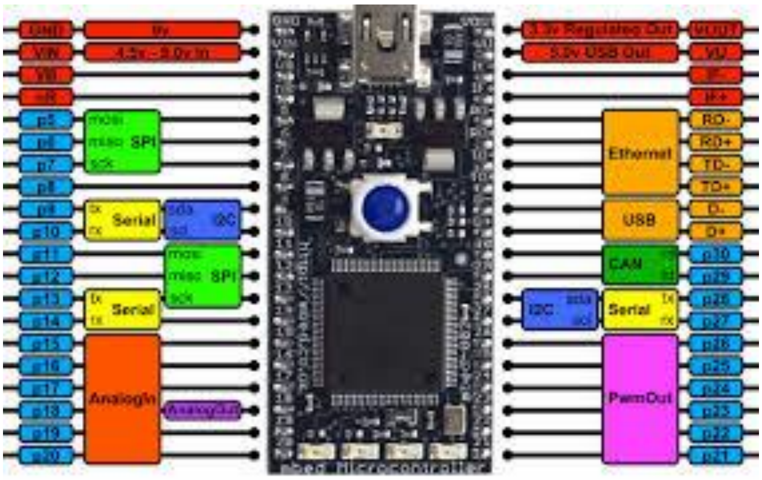

Figure 17: Scheme of Microcontroller mbed NXP LPC1768 pins.

The sensor INA260 was chosen because it can to measure power, current and voltage, and it has I2C interface which is compatible with the mbed microcontroller. The picture of the INA260 sensor is shown in Figure 18.

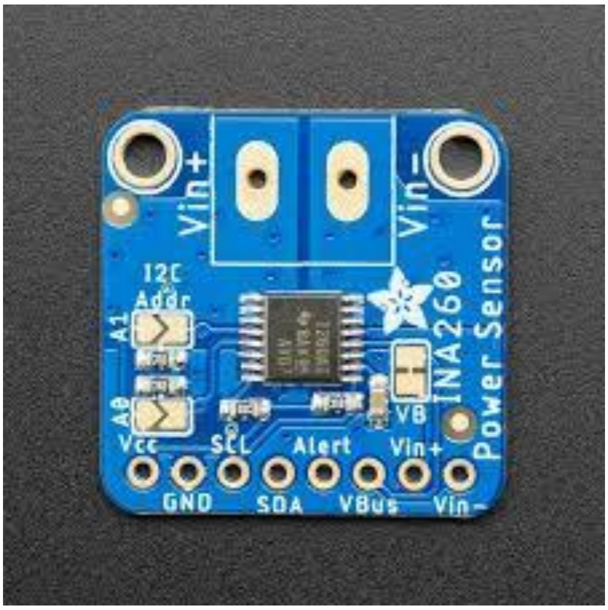

Figure 18: Sensor INA260.

The sensor senses voltages up to 36V, which can measure the voltages from the eBike battery voltage and it senses current up to 15A, which is compatible with the eBike battery current. Other characteristics of the iNA260 sensor are:

- It has operation temperature from -40°C to +85°C;
- The analogue voltage is converted to digital using an DAC of 16 bits;
- Its internal resistance is 2mΩ;

- The current resolution is 1.5mA;
- It has maximum offset of 5mA;
- It has 16 programmable addresses;
- Its power supply is 2.7V to 5.5V;
- Its conversion time is from 140μs to 8.244ms;
- According to the manufacturer, to achieve the highest accuracy measurement possible and reduce noise, a combination of longest allowable conversion times and highest number of averages shall be used based on the timing requirements of the system.

The HC05 Bluetooth module was chosen because it has a serial interface which is compatible with the mbed microcontroller. A picture of the HC05 Bluetooth module is shown in Figure 19.

Figure 19: HC05 Bluetooth module.

The SD card breakout board was chosen because it has an SPI interface which is compatible with the mbed microcontroller. A picture of the SD breakout board is shown in Figure 20.

Figure 20: SD Card breakout board.

After choosing the hardware components of the systems, the components were bought and tested one by one. In order to test the INA260 sensor the circuit shown in Figure 21 was set up.

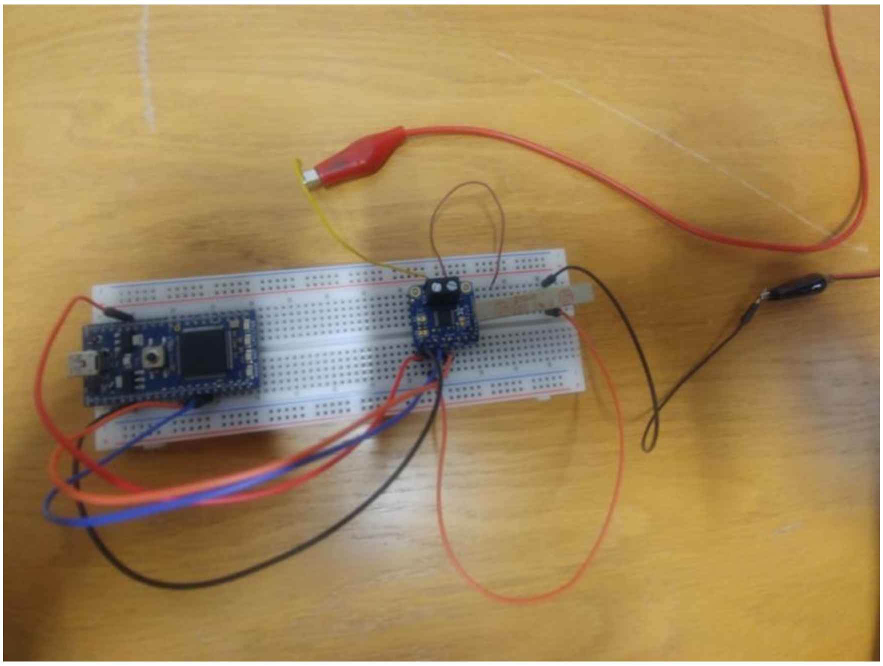

Figure 21: Eletronic circuit to test the INA260 sensor.

In the circuit the microcontroller mbed is connected to the INA260 sensor through the I2C interface. This connection was made with the following wired connections:

Table 2: Connections Between INA260 and mbed.

INA260	MBED
PIN 1 (VOUT)	PIN 40 (VOUT)
PIN 2 (GND)	PIN 1 (GND)
PIN 3 (SCL)	PIN 10 (SCL)
PIN 4 (SDA)	PIN 9 (SDA)

The INA260 is connected to a power resistance. The circuit was supplied by a power supply and connected to a MAC BOOK pro through a USB cable. The firmware of the embedded system is a C program were the SPI interface is first configurated and them the INA260 sensor starts to read power, current and voltage continually and print it to a serial monitor.

In order to test the system, the voltage, and therefore current and power was varied and the voltage from the power supply was compared to the one measure by the embedded system. The voltages matched, which showed that INA260 worked well.

In order to test the Bluetooth module, the Bluetooth module was connected to mbed microcontroller via a serial interface. This connection was made with the following wired connections:

Table 3: Connections Between Bluetooth Module and mbed.

Bluetooth Module	MBED
PIN 2 (RXD)	PIN 28 (TXD)
PIN 3 (TXD)	PIN 27 (RXD)
PIN 4 (GND)	PIN 1 (GND)
PIN 5 (VOUT)	PIN 40 (VOUT)

The embedded system's firmware was a C program were the Bluetooth module and the Serial monitor were configurated and then inside an infinite loop if a character from the Serial Monitor was received it was sent again to the Serial Monitor and to the Bluetooth Channel. Also, inside the same infinite loop if a character was received via Bluetooth, it was sent via Bluetooth again and sent to the Serial Monitor. The system worked well.

In order to test the SD Card breakout board, the SD Card breakout board was connected to the mbed using SPI interface. This connection was made with the following wired connections:

Table 4: Connections Between SD Breakout Board and mbed.

SD Card Breakout Board	MBED
PIN 1 (VU)	PIN 39 (VU)
PIN 3 (GND)	PIN 1 (GND)
PIN 4 (CLK)	PIN 7 (CLK)
PIN 5 (DO)	PIN 6 (DO)
PIN 6 (DI)	PIN 5 (DI)
PIN 7 (CS)	PIN 8 (CS)

The embedded system's firmware was a C program were first the SD Breakout board was configurated and the functions to write, read and delete files to the SD Card were written. The system worked well.

In this way, the schematic of the final embedded system, were the mbed, the INA260 sensor, the Bluetooth Module and the SD Card Breakout board were put together in the same circuit is shown in Figure 22.

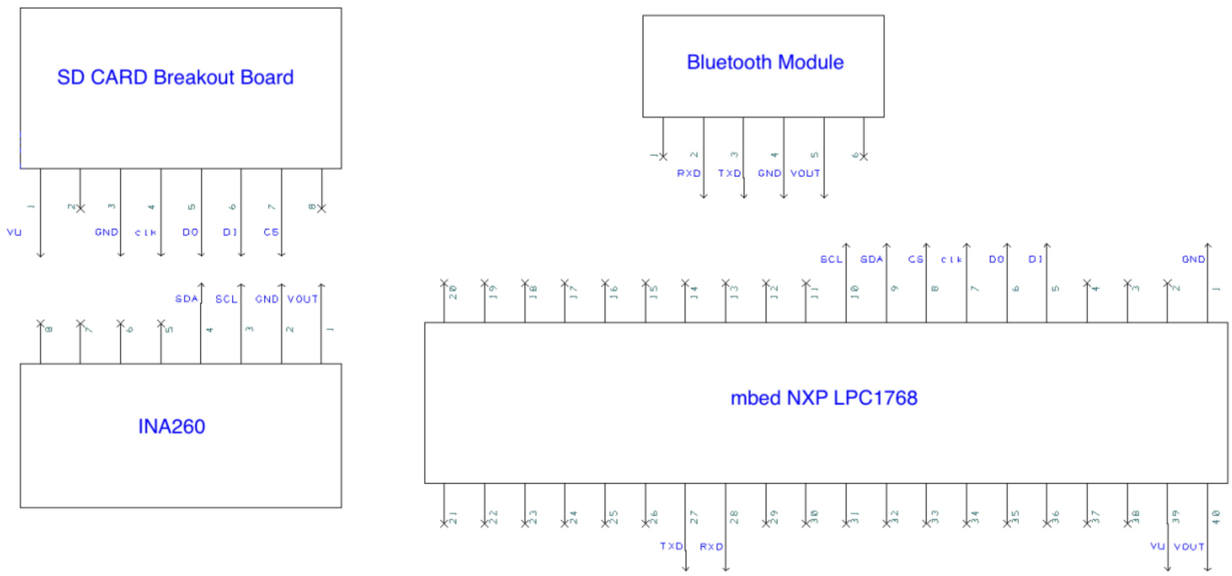

Figure 22: Schematic of eBike Battery Power, Energy, Current and Voltage meter.

3.3. Firmware Implementation

In order to implement the embedded system's firmware, first the TEDS of the whole system had to be written since that once the eBike TIM receives specific commands from the NCAP asking for each of the Transducer Channels TEDS, the TIM needs to answer with the corresponding set of TEDS. These answers are set of hexadecimal numbers that need to be generate according to the IEEE1451 standard.

In order to write the Transducer Channel TEDS a program online was developed by the Instrumentation LAB team. This program allows the user to fill the Transducer Channel TEDS and automatically get the TEDS in the Hexadecimal format, as requested by the standard. The processes of TEDS writing using the online program is showed in Figures 23.

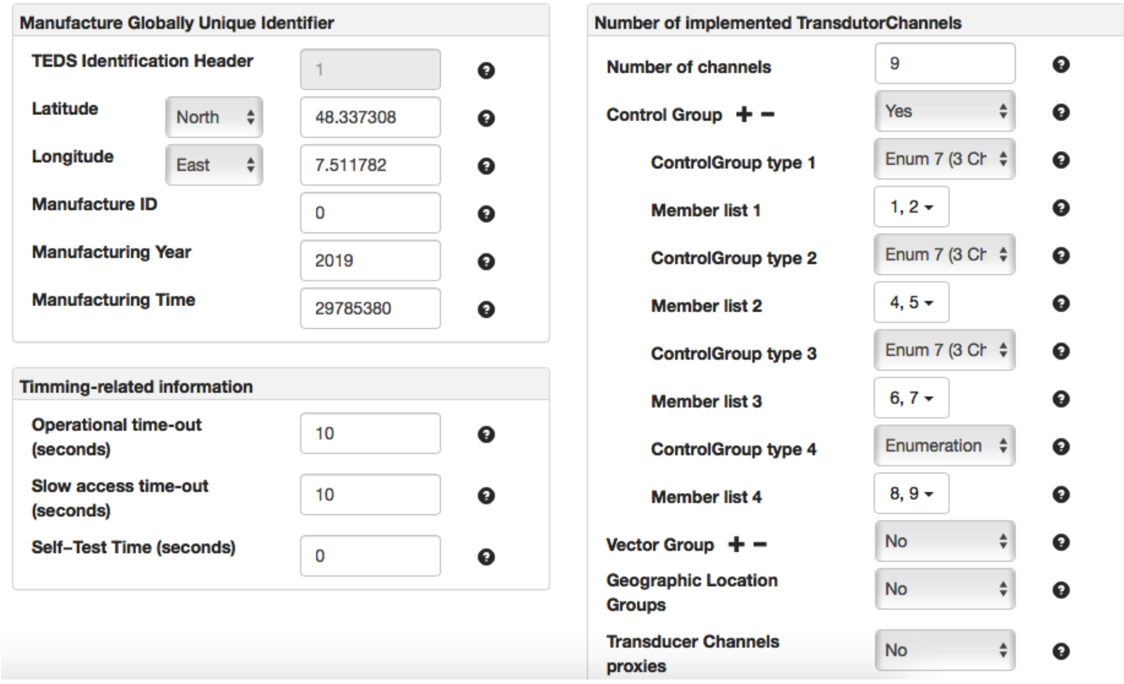

Figure 23: Meta TEDS Fill-in according to TEDS online software.

The TEDS Identification Header is already fixed to 1. The Latitude and Longitude are set according to the location of the assembly of the system. Manufactured ID is chosen according to the user's desire. Manufacture year and time are the year and time of the system assembly. The operational time-out contains the time interval, in seconds, after an action for which the lack of reply following the receipt of a command may be interpreted as a failed operation. The slow access time-out is the same as the operational time-out but may be used if one time-out is not enough.

The self-test time contains the maximum time, in seconds, required to execute the self-test. If no self-test is implemented, this field is zero. The number of transducer channels contains the number of transducer channels implemented. ControlGroups identify transducers that are used to control the operation of other transducers and they utilization are explained in the standard.

The geografic location implements a specialized VectorGroup type that is used to provide dynamic geographic location information. VectorGroups identify the relationships between the data sets within a transducer module, and their work are explained in the standard. A TransducerChannel proxy is an artificial construct used to combine the outputs of multiple sensors or the input to numerous actuators into a single structure and its work is explained in the standard. The MetaTEDS in the hexadecimal format generate by the online software are shown below.

****

META-TEDS

****

0x00 0x00 0x00 0x2C
0x03 0x04 0x00 0x00 0x00 0x01
0x04 0x0A 0x95 0x3D 0xF4 0x1A 0x68 0x81 0xF8 0xF8 0xCF 0xA4
0x0A 0x04 0x41 0x20 0x00 0x00
0x0B 0x04 0x41 0x20 0x00 0x00
0x0C 0x04 0x00 0x00 0x00 0x00
0x0D 0x02 0x00 0x09
0x0E 0x30 0x15 0x01 0x07 0x16 0x03 0x01 0x02 0x00 0x15 0x01 0x07 0x16 03 0x04 0x05 0x00 0x15 0x01 07 0x16 0x03 0x07 0x08 0x00
0x15 0x01 0x07 0x16 0x03 0x0A 0x0B 0x00 0x15 0x01 0x07 0x16 0x03 0x0D 0x0E 0x00 0x15 0x01 0x07 0x16 0x03 0x10 0x11 0x00
0xF6 0x5B

****

The hexadecimal format of all TEDS are shown in appendix 1. Later the eBike TIM firmware was written using mbed online compiler. The program was implemented in the C language and the pseudo-code of the program is shown in Algorithm 1 below.

Algorithm 1: eBike TIM Embedded System Firmware Pseudo-Code

```
mbed, INA260, Buffer and SD File library includes
Buffers size definition
Serial monitor and Bluetooth configuration
Function and variables declaration
Variables Initialization
Main
    Serial Monitor and Bluetooth Communication rate set
    while (1)
        Bluetooth commands read
        if receives command asking to trigger a transducer channel
```

```
            Trigger this transducer channel
        end if
        If receives command asking to read TEDS
            Send TEDS
        end if
        If receives command asking to delete file
            Delete file
        end if
        State machine of Analog Input Sensor TransducerChannel for power data
//This state machine attach the interrupt function "Aquisicao1()"
        State machine of Digital Embedded Sensor Transducer Channel from SD Card for power Data
//This state machine reads power data from the SD Card
        State machine of Digital Embedded Sensor Transducer Channel from SD Card for voltage Data
//This state machine reads voltage data from the SD Card
        State machine of Digital Embedded Sensor Transducer Channel from SD Card for current Data
//This state machine reads current data from the SD Card
        State machine of Analog Input Sensor TransducerChannel for voltage data
//This state machine attach the interrupt function "Aquisicao2()"
        State machine of Analog Input Sensor TransducerChannel for current data
//This state machine attach the interrupt function "Aquisicao3()"
        State Machine for the Digital Embedded Actuator Transducer Channel for power data
//This state machine calculates power data sum and average and send the average to the NCAP
        State Machine for the Digital Embedded Actuator Transducer Channel for voltage data
//This state machine calculates voltage data sum and average and send the average to the NCAP
        State Machine for the Digital Embedded Actuator Transducer Channel for current data
//This state machine calculates current data sum and average and send the average to the NCAP
       State Machine for Digital Embedded Actuator Transducer Channel for power data
//This state machine writes power data to the SD Card
       State Machine for Digital Embedded Actuator Transducer Channel for voltage data
//This state machine writes voltage data to the SD Card
       State Machine for Digital Embedded Actuator Transducer Channel for current data
//This state machine writes current data to the SD Card
    end while
end main
Aquisicao1
    Power data acquisition and storage in buffers
end Aquisicao1
Aquisicao2
    Voltage data acquisition and storage in buffers
end Aquisicao2
Aquisicao3
    Current data acquisition and storage in buffers
end Aquisicao3
```

At the beginning of the code some libraries are included, the size of the buffers that will be used is defined, serial monitor and Bluetooth are configured, functions are declared, and variables are declared and initialized.

Then, the program enters the **main()** function. Inside the while, Bluetooth commands are read. If the command refers to a transducer channel trigger, the program triggers the corresponding transducer channel. Also, if the command refers to TEDS read, the corresponding TED is sent. And third, if the command refers to a file delete request, this file is deleted.

After that, the program enters into a series of State Machines implemented with switches functions. Each state machine represents one of the transducer channels from the diagrams in section 3.1, where the transducer channel corresponds to either a sensor or an actuator. According to the standard IEEE1451, the sensor follows the state machine in Figure 24.

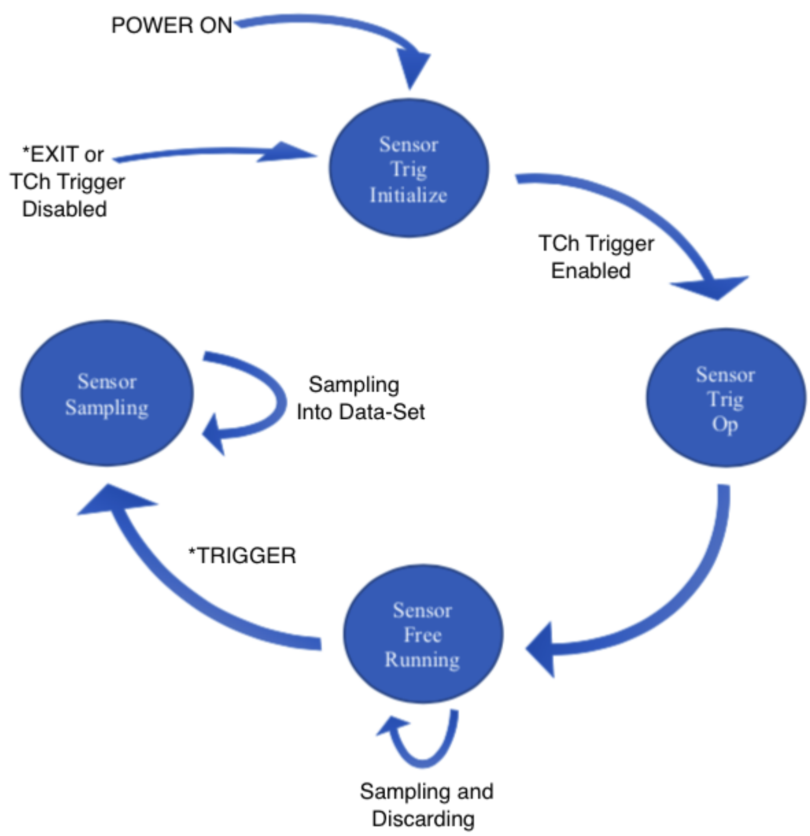

Figure 24: Sensor in the Continuous Sampling mode State Machine.

In our case of study, the sensor is in the Continuous Sampling mode. In this case, after the TIM is turned on, the system enters the Sensor Trig Initialize State. In the Sensor Trig Initialize State, the sensor is initialized and configurated. If the command TCh Trigger is enabled, the state machine goes to Sensor Trig Op State. In the Trig Op state, the sensor is ready to start working. In the Continuous Sampling mode, the sensor goes straight to the Sensor Free Running State. In the Sensor Free Running State, the sensor is sampling and discarding and it is waiting for a trigger coming from the NCAP to start acquiring data to a data set. [36]

In the Sensor Sampling State, the sensor is sampling and storing data in its buffers. If the sensor is in any of the states except for in the Trig Initialize state and Tch Trigger is enabled or the Reset, Device Clear, Abort Trigger or Write TCh Trigger State are disabled the state machine goes back to the Sensor Initialize State. [36] On the other hand, according to the standard IEEE1451, the actuator follows the state machine in Figure 25.

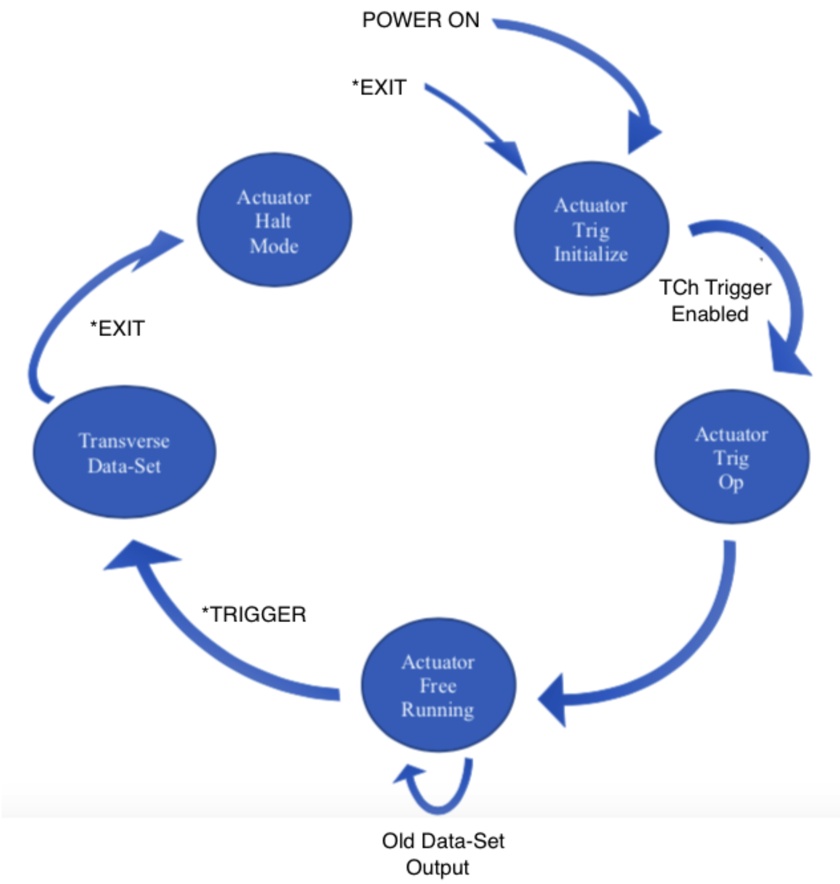

Figure 25: Actuator in the Continuous Sampling mode State Machine.

In our case of study, the actuator is in the Continuous Sampling mode. In this case, when the system is turned on, the state machine enters the Actuator Trig Initialize State. In the Trig Initialize State, the actuator is initialized and configured. If Tch Trigger is enabled, the state machine goes to the Trig Op state. In the Trig Op state, the actuator is ready to start working. In the Continuous Sampling mode the state machine goes straight to the Free Running state. In the Actuator Free Running, the actuator is outputting old data set data and it is waiting for a trigger coming from the NCAP to start outputting data from new data sets. [36]

In the Transverse Data-Set state, the actuator is outputting data from new buffers. In the Transverse Data-Set state if the Reset, Device Clear, Abort Trigger or Write Transducer Channel (TCh) Trigger State are disabled, the state machine goes to Actuator Halt Mode state. In the Actuator Halt Mode state, the actuator can hold the current state, finish applying the current data set, and hold the last output or ramp to a predefined state. Also if the state machine is in any state except for Trig Initialize or Transverse Data Set and the Reset, Device Clear, Abort Trigger or Write TCh Trigger State are disabled, the state machine goes back to the Trig Initialize state. [36]

The eBike TIM Embedded System program starts with a State machine of the Analog Input Sensor TransducerChannel for power data. In the Sensor Sampling State it sets an interruption function **Aquisicao1()**. This function is called once every 2 seconds. The function **Aquisicao1()** does Power data acquisition and store the data in buffers.

The second State Machine in the program is the State machine of Digital Embedded Sensor Transducer Channel from SD Card for power Data. In the sensor sampling state this state machine reads power data from the SD Card. Then, the State machine of Digital Embedded Sensor Transducer Channel from SD Card is repeated for voltage and current data. They read voltage and current data from the SD Card, respectively.

After that, the program implements States machine of Analog Input Sensor TransducerChannels for voltage and current data. In the Sensor Sampling State, these state machines set the interruption functions **Aquisicao2()** and **Aquisicao3()**. These functions do voltage and current data acquisition, respectively and store them in buffers.

Then the program implements a State Machine for the Digital Embedded Actuator Transducer Channel for power data. In the Transverse Data Set State, this state machine calculates power data sum and average and sends the average to the NCAP.

After that, the program implements States Machines for the Digital Embedded Actuator Transducer Channels for current and voltage data. They have analogue principles of the States Machines for the Digital Embedded Actuator Transducer Channels for power data.

The last three state machines are State Machines for Digital Embedded Actuator Transducer Channels for power, voltage and current data. In the Transverse Data Set state, each writes power, voltage and current data to the SD Card, respectively.

3.4. Printed Circuit Board Design

The Printed Circuit Design (PCB) was designed using the software Design Spark. First the schematic was drawn as shown in Figure 26. Then the equivalent schematic components were drawn as footprints for the layout design. The mbed, Bluetooth Module, INA260 and SD Card Breakout board footprints are shown in Figure 26 from the left to the right order.

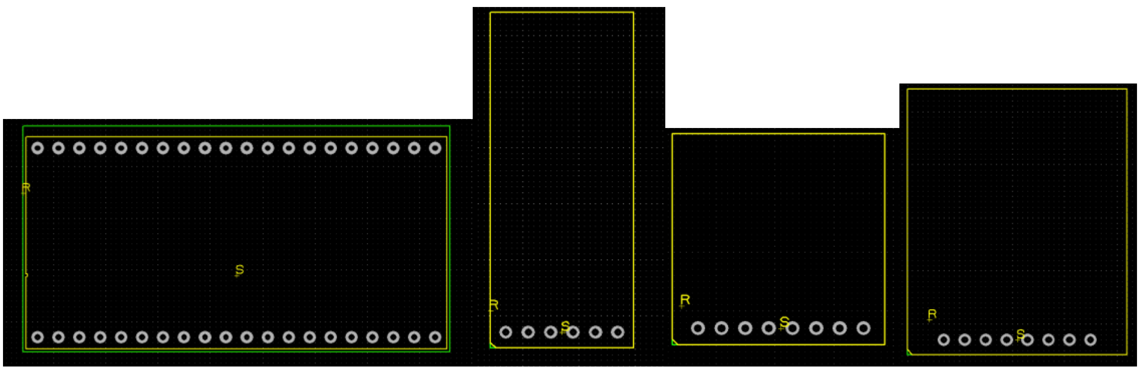

Figure 26: mbed, Bluetooth module, INA260 and SD breakout board footprints.

After that, the size of the PCB was chosen, the footprints were placed in the PCB in two faces, the rooting was done in both faces and the gerber files were generated. One face of the PCB rooted is shown in Figure 27.

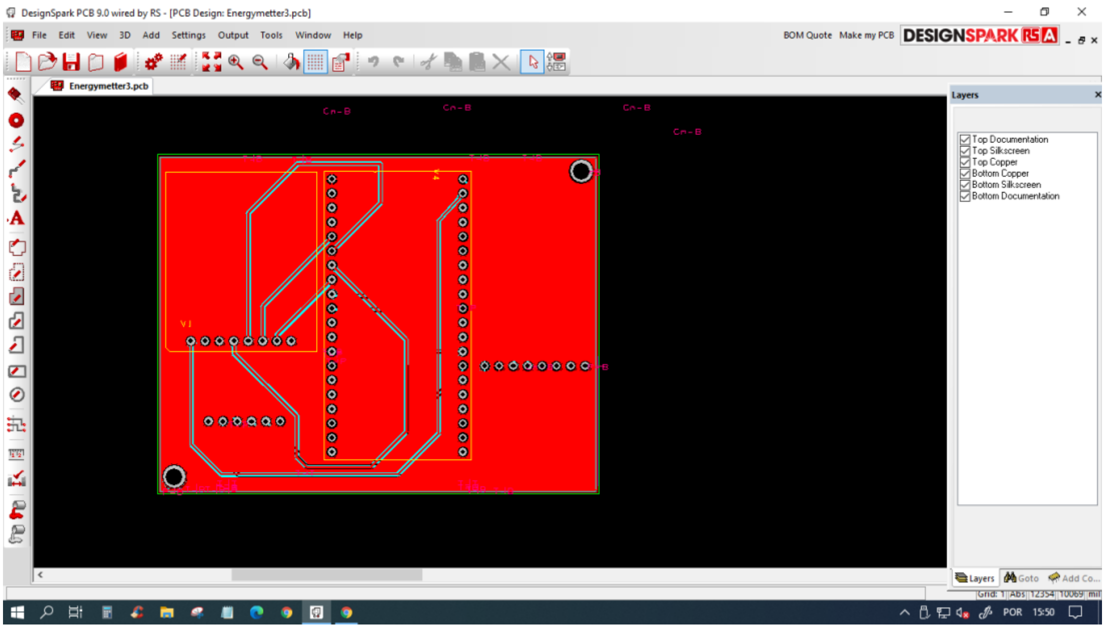

Figure 27: One face of the PCB rooted.

The PCB was then assembled and the components were soldered. The final device can be seen in Figures 28.

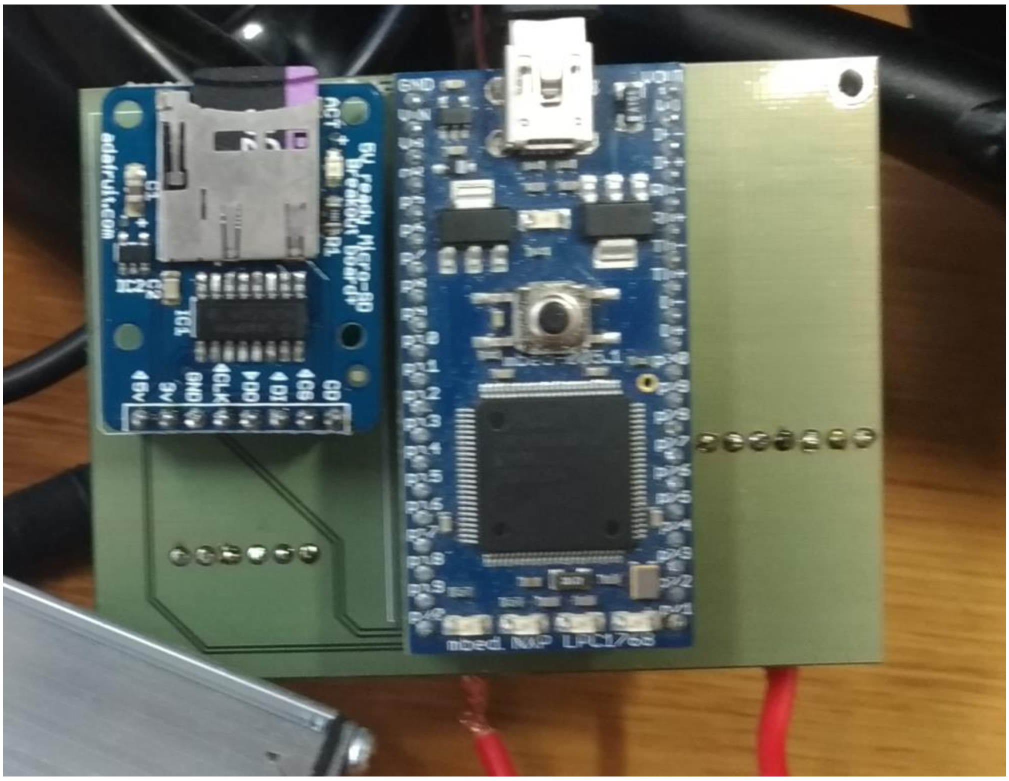

Figure 28: Energy meter.

Figure 28 shows one face of the PCB of the energy meter. The final dimentions of the PCB were 7.7 cm of length and 5.9 cm of width. On the right side of the Figure, the mbed is soldered and in the left side the SD Breakout Board is soldered. The Bluetooth Module and the INA260 sensor are on the other face of the PCB.

Chapter 4

Mobile NCAP Application

The desired NCAP application had to receive commands from the management system through the internet and send these commands via Bluetooth to the TIM. The TIM then send data back to the NCAP application via Bluetooth. The NCAP application must take these data and send them to the management system via the internet.

In order to exchange commands and data to the TIM, the NCAP application has to search for Bluetooth devices and connect to these Bluetooth devices. On the other hand, to exchange commands and data with the management system, the NCAP application must use the MQQT Protocol.

Then, to receive commands from the management system, the NCAP application must subscribe to the same topic that the management system will publish the command. This command will be sent first to the broker and the broker will send the command to the NCAP application. The NCAP application has to publish the data to a topic to send data to the management system. This data will be sent to the broker and then the broker will send this data to the management system if the management system subscribes to the same topic.

In order to develop the NCAP application, the first two separated applications were considered. The first one does the Bluetooth search, connection and message exchange. This app was download from the APP Store. The second app was used to publish and subscribe messages using MQTT protocol.

The two apps were put together in one app that communicates via Bluetooth and MQTT protocol with two clients. The app developed first search for the available Bluetooth devices to be connected and show them on the first screen, as shown in Figure 29.

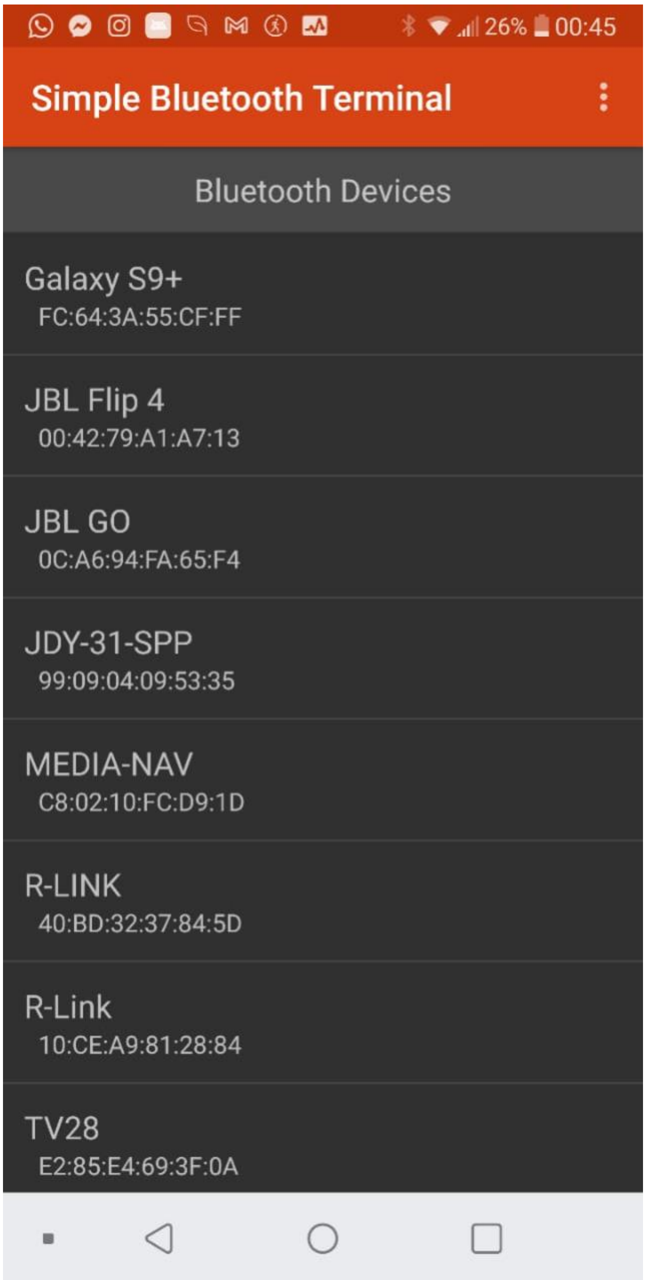

Figure 29: Bluetooth devices available to be connected.

When the user clicks on the device he wants to connect with, the app changes the screen to a screen showing if the Bluetooth connection is being established. Later, it shows if the devices are connected via Bluetooth and also indicates if the MQTT is connected. Then, if the NCAP application receives commands from the management system it sends them to the TIM via Bluetooth. After that, if it gets data from the TIM via Bluetooth, it shows the data on its screen and sends the data to the management system via MQTT protocol. The second screen of the NCAP APP, where this happens, is illustrated in Figure 30.

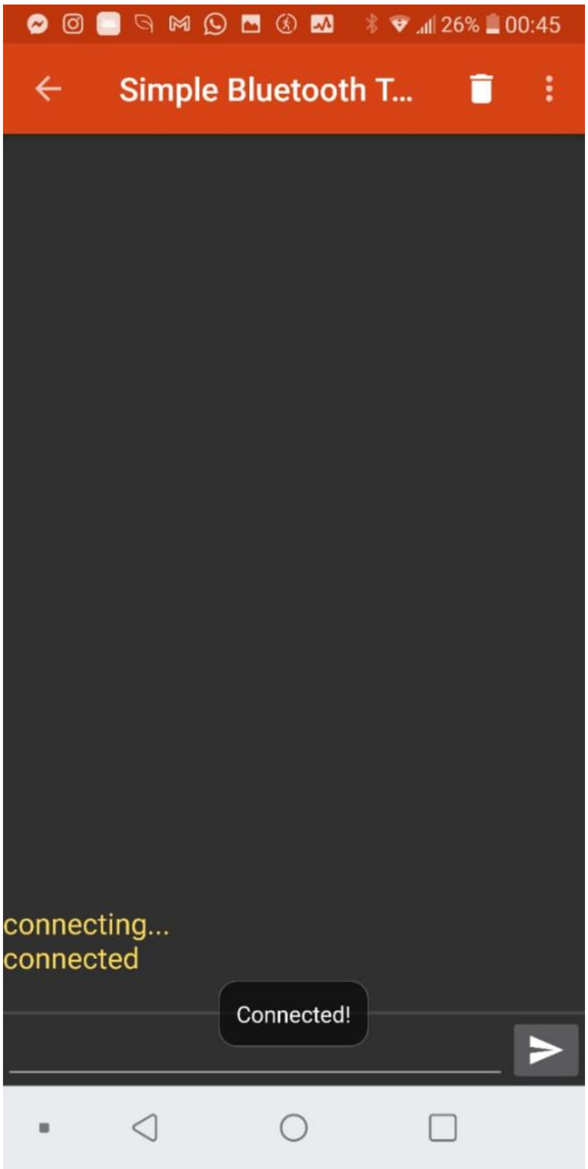

Figure 30: NCAP application second screen.

This screen shows the text “connecting...” when the APP tries to connect with the TIM via Bluetooth. It shows the word “connected” if the APP connected with the TIM via Bluetooth. Also, a message “connected” appears in the screen for a few milliseconds if the NCAP APP is successfully connected to the MQTT broker. The second screen of the NCAP APP if the when the app receives data from the TIM is shown in Figure 30.

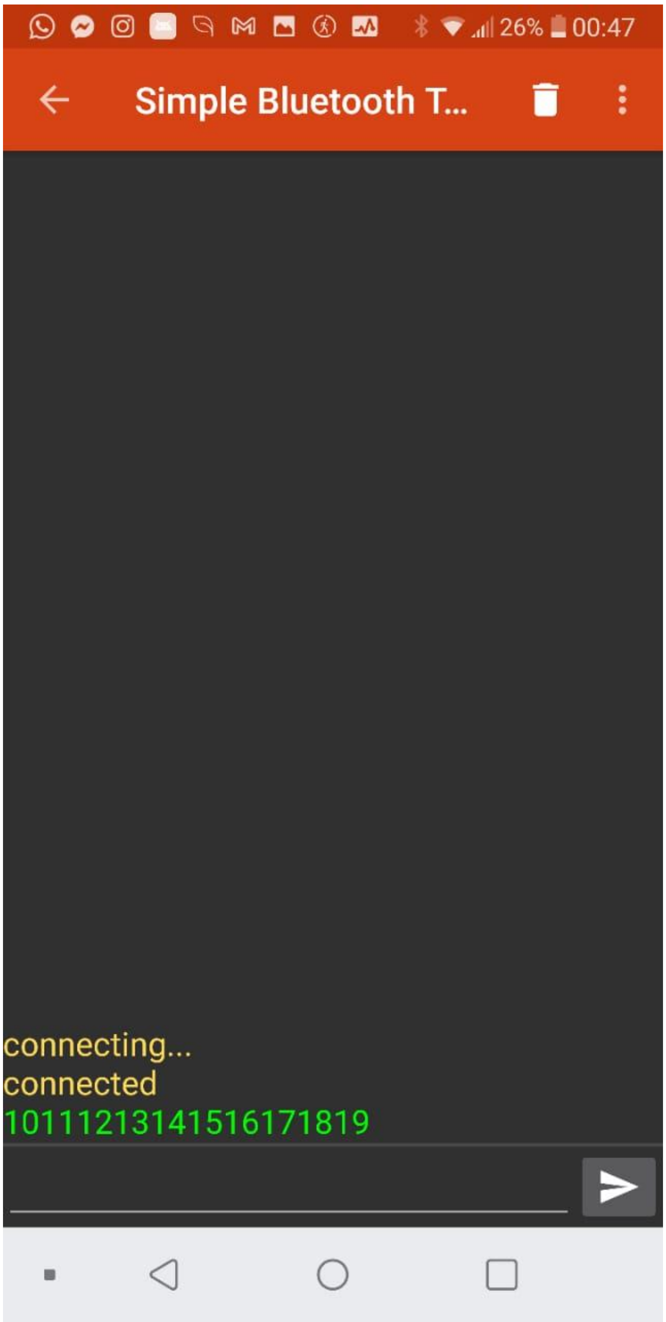

Figure 31: NCAP application second screen with data came from the TIM.

The numbers 10,11,12,13,14,15,16,17,18,19,20 are the data that the TIM sent to the NCAP APP.

Chapter 5

Management Application

5.1. Node-Red Flow Charts

The management application runs in a mac with the software node-red. The application is implemented by connecting nodes or blocks inside the Flows. The nodes may generate an User Interface called Dashboard that stay available online. Figures 32, 33, 34 and 35 show the central nodes used in the app development.

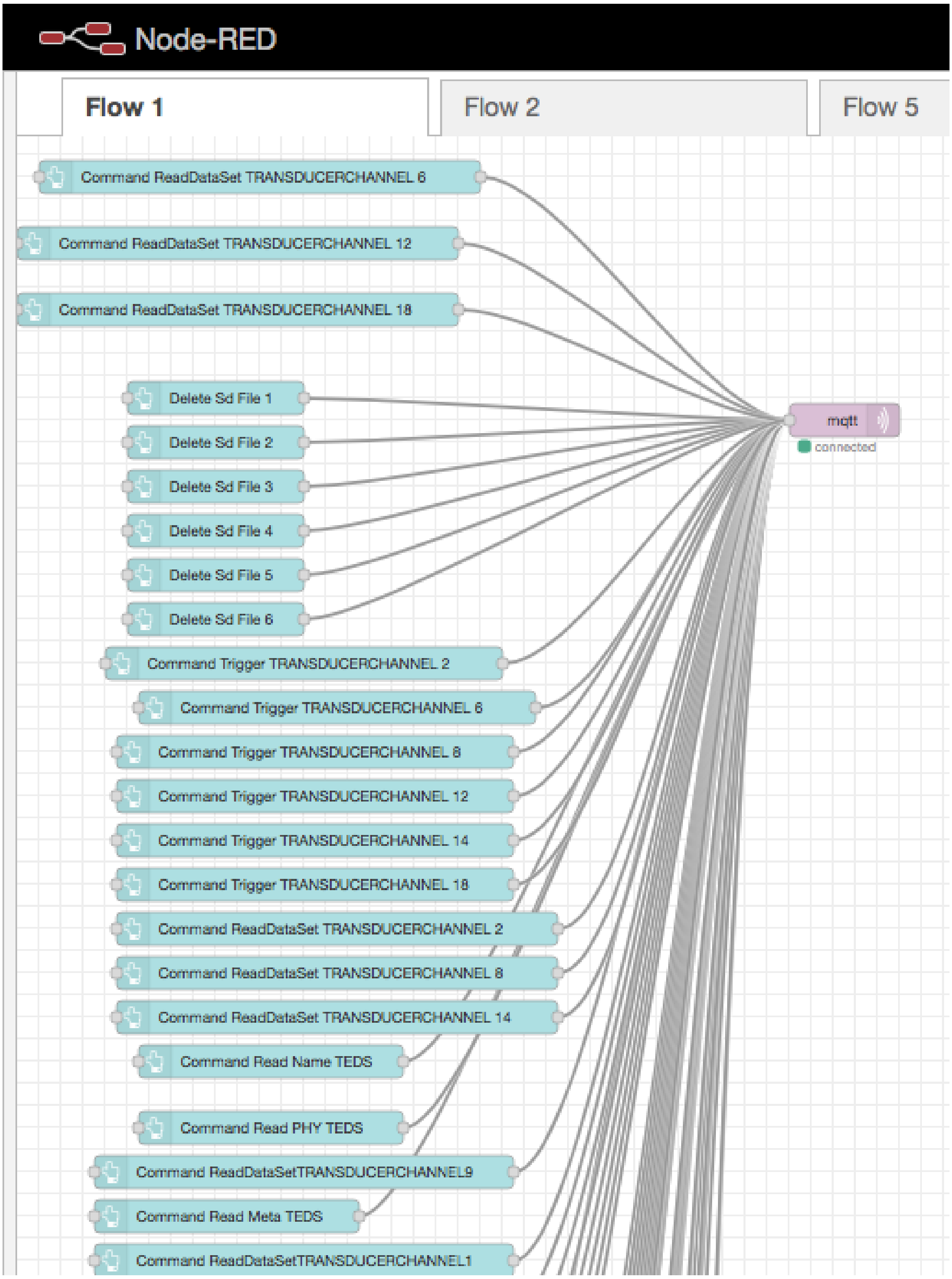

Figure 32: Part 1 of NodeRed App Development.

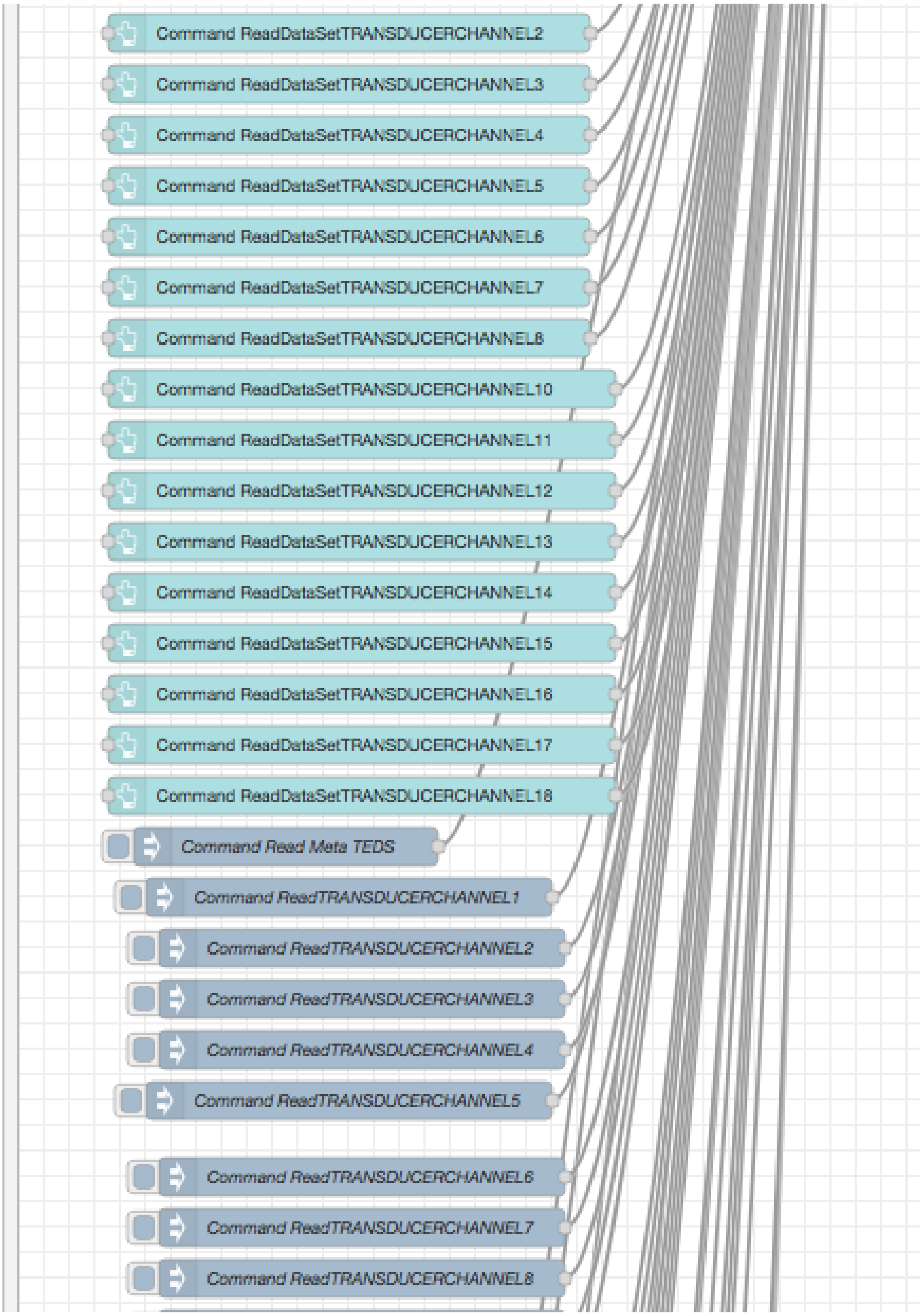

Figure 33: Part 2 of NodeRed App Development.

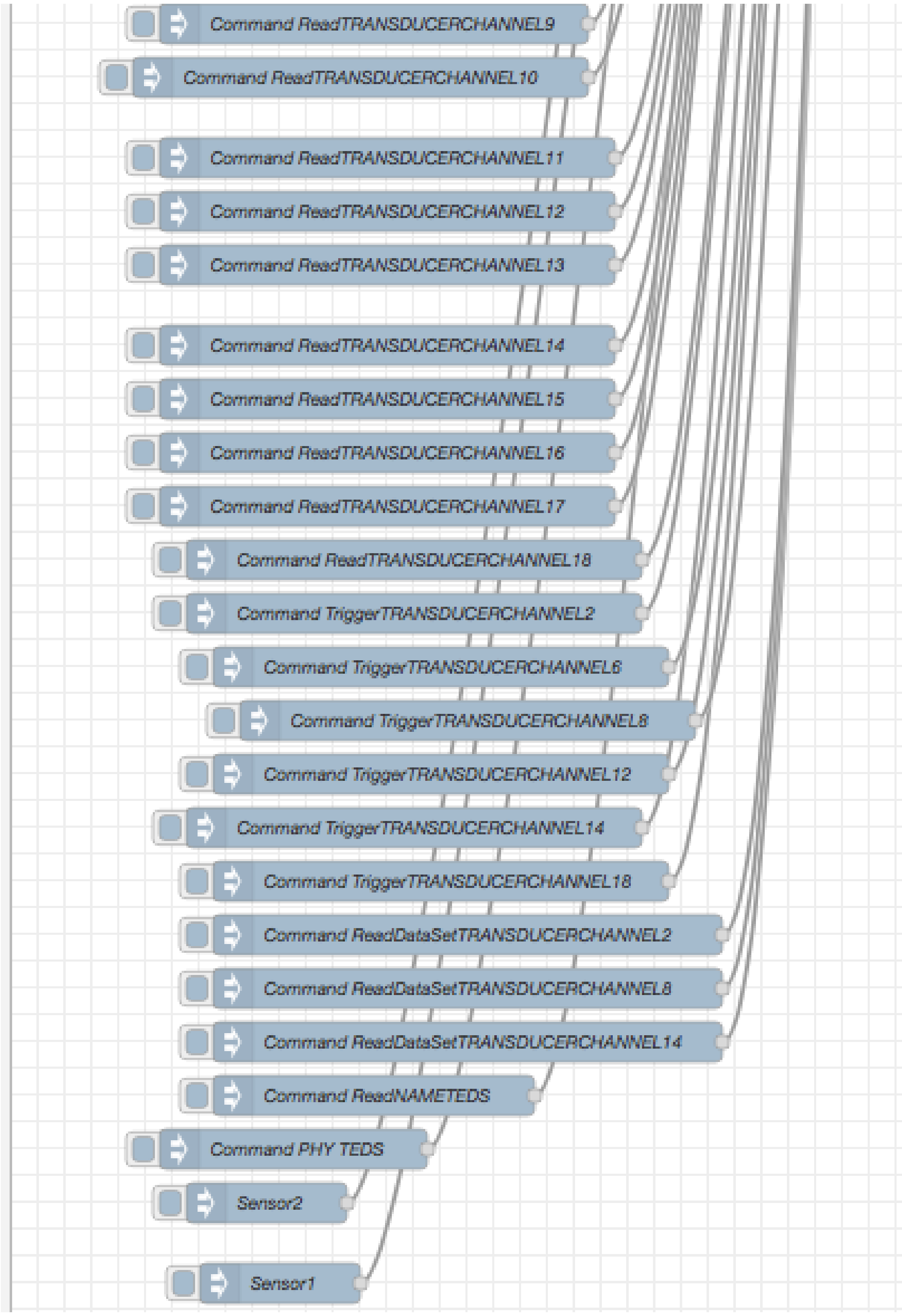

Figure 34: Part 3 of NodeRed App Development.

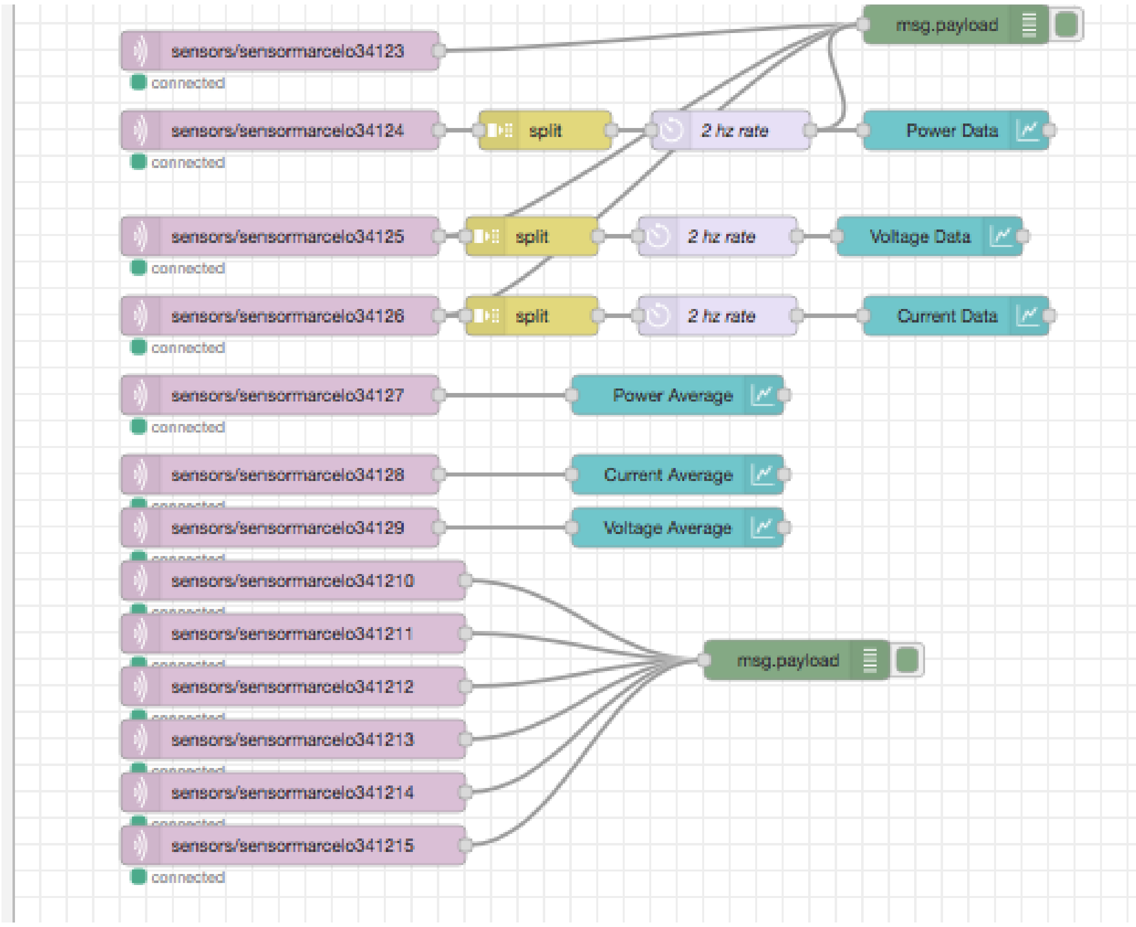

Figure 35: Part 4 of NodeRed App Development.

In Figure 32 and 33, the light blue nodes on the left side of the Figure are called button nodes. They generate a button on the node-red DashBoard and are linked to the node MQTT out labeled “mqtt”. This node publishes messages to the broker. Then, when the user pushes the buttons in the Dashboard, the messages inside them are published. In this way, all the commands of the diagrams in section 3.1. were put inside the button nodes.

In Figures 33 and 34, the light purple nodes are called inject nodes and have the same function as the light blue nodes, but in this case when the buttons are pressed inside the Flow, since they are connected to the MQTT node, the messages inside them are published to the broker.

In Figure 35, the light pink nodes are called MQTT in. They subscribe messages from the broker. The topic to which the node will subscribe is set inside the node. Then the MQTT in nodes are connected to msg.payload nodes, split nodes or chart nodes. The message payloads nodes print the messages subscribed to the debug screen for development purposes.

When the messages are received in the same line and with high frequencies, they pass to split and delay nodes before they are plotted to charts in the Dashboard. The split node splits the message into several lines. Then each message is configured in a frequency of 2Hz in the delay node labelled “2 Hz rate”. Later each message is plot in the chart nodes labelled “Power Data”, “Current Data” and “Voltage Data”. Other messages that have a lower frequency rate are connected straight to the chart nodes labelled “Power Average”, “Voltage Average” and “Current Average”. In this case, when the electric parameters data are sent by the smartphone to the broker and the node-red management app subscribe to those messages, they appear in the charts of the node-red Dashboard.

Another flow is not showed here that takes care of the text messages received. The messages pass through a delay node of 1Hz and then pass through other nodes to ensure the messages are shown in a scrolling table of 20 lines. This feature of the app will be clarified in the Node-Red DashBoard presented next.

5.2. Node-Red DashBoard

The Node-Red DashBoard is composed of two tabs. The first tab has the set of buttons that command the read of the Transducer Channels TEDS and a scrolling table of 20 lines. The scroll table present the TEDS received by the app. The second tab gives the buttons that command the data acquisition and offer the data in charts. The first tab of the DashBoard is shown in Figures 36 and 37.

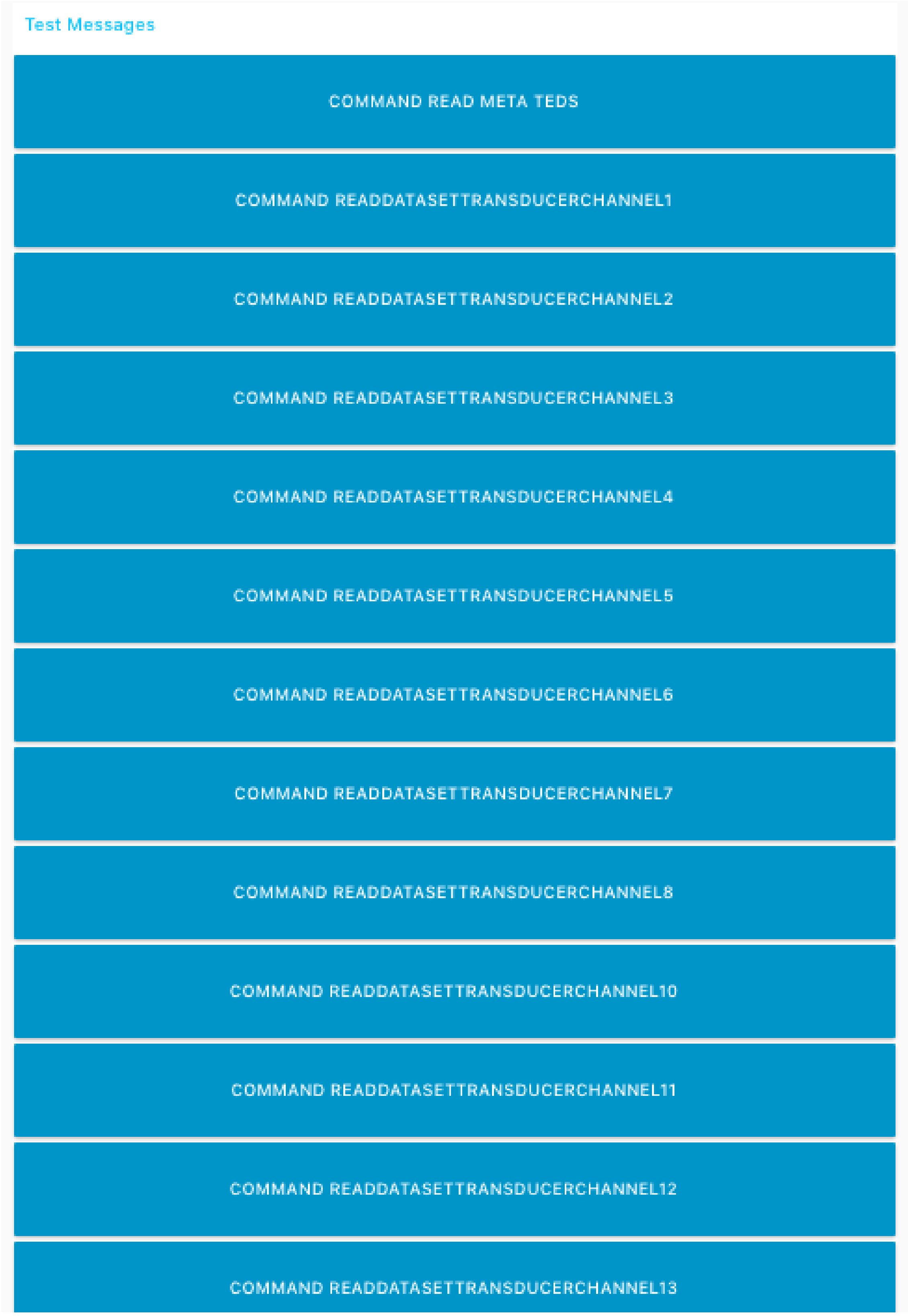

Figure 36: Part 1 of Text Tab of the DashBoard.

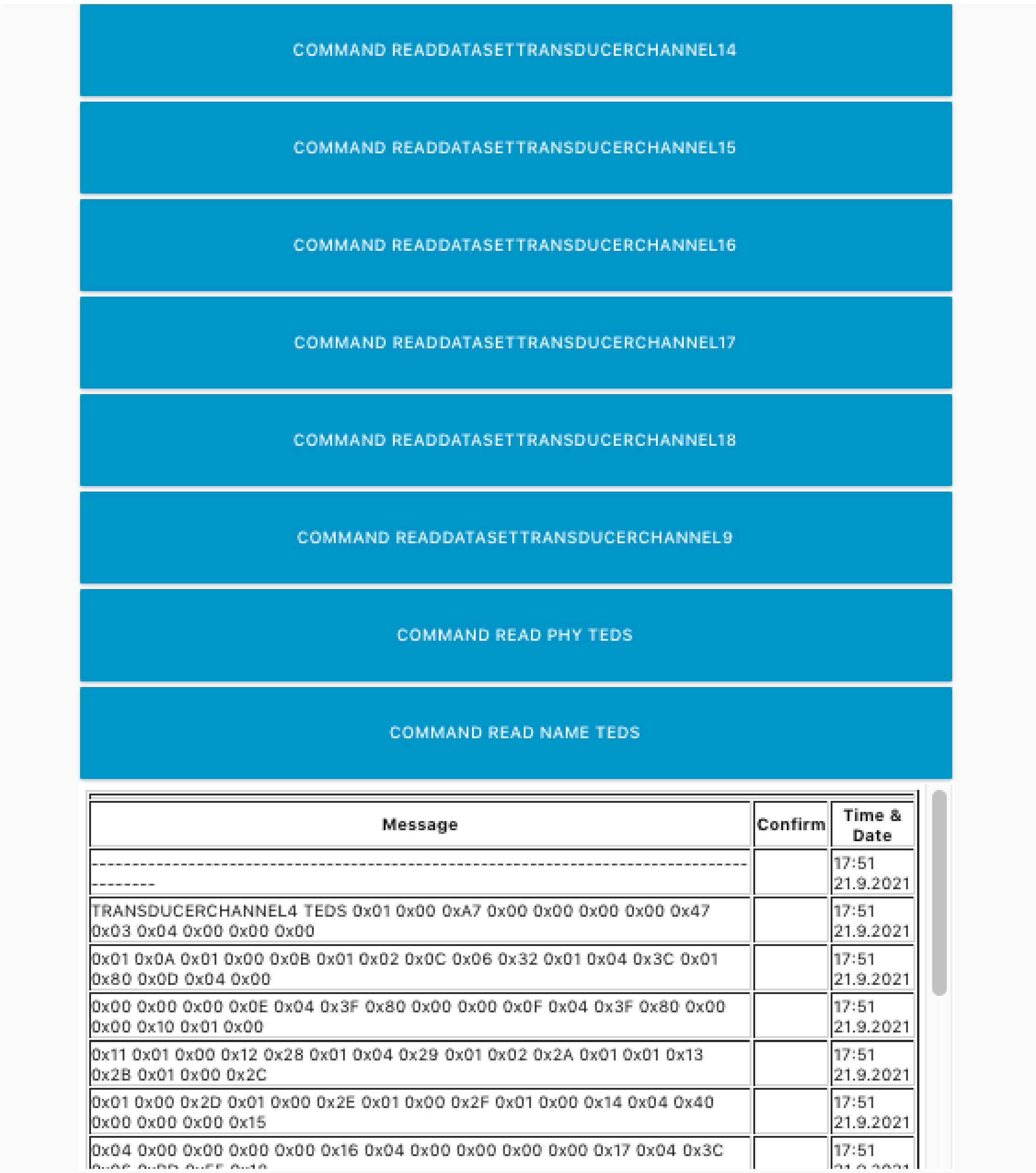

Figure 37: Part 2 of Text Tab of the DashBoard.

In Figures 35 and36, the button TransducerChannel TEDS 4 was pressed, and then the Transducer Channel TEDS appeared in the scrolling table. The Figures 37, 38, 39 and 40 present the Charts Tab of the DashBoard.

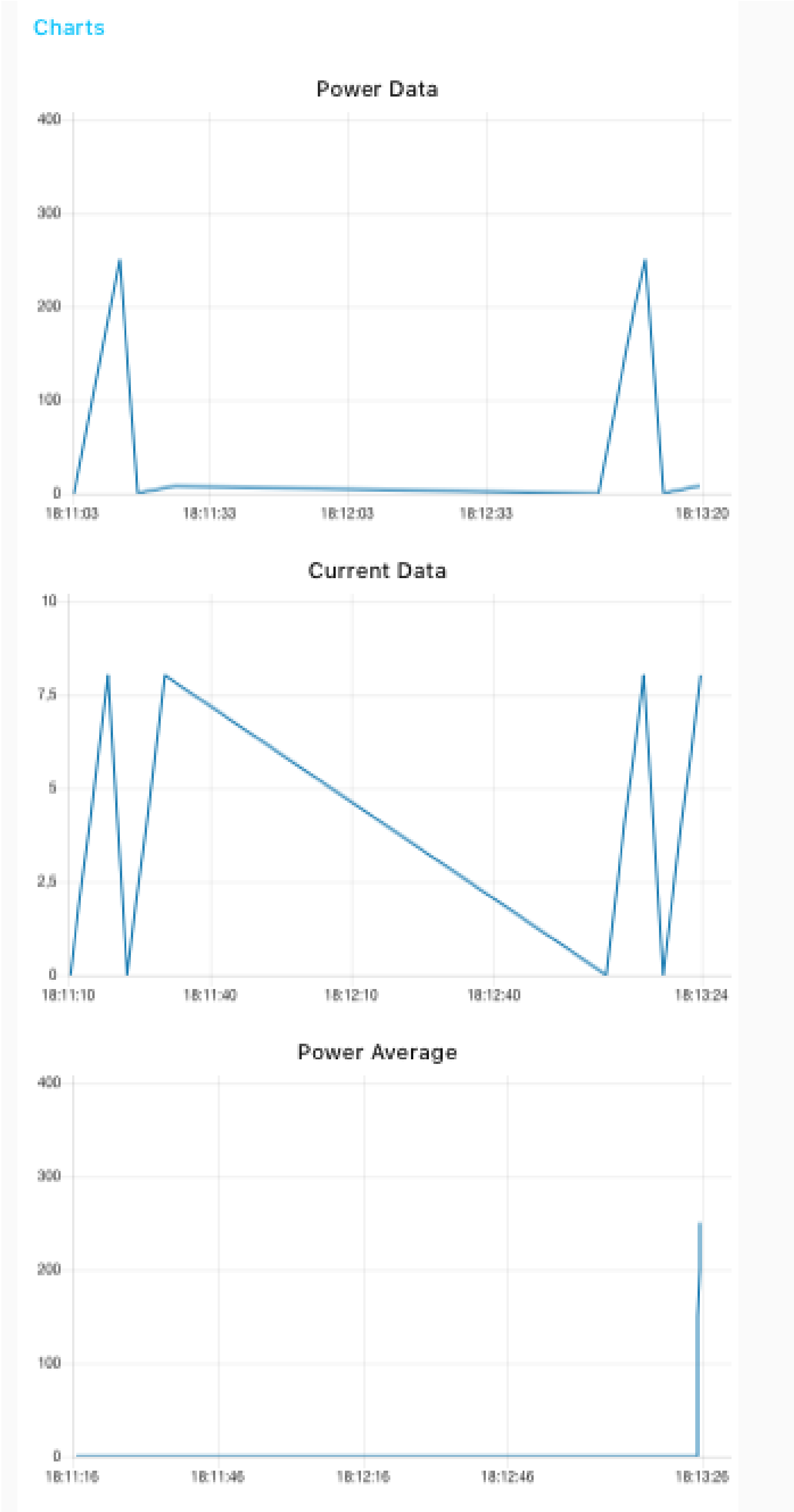

Figure 38: Part 1 of Chart Tab of the DashBoard.

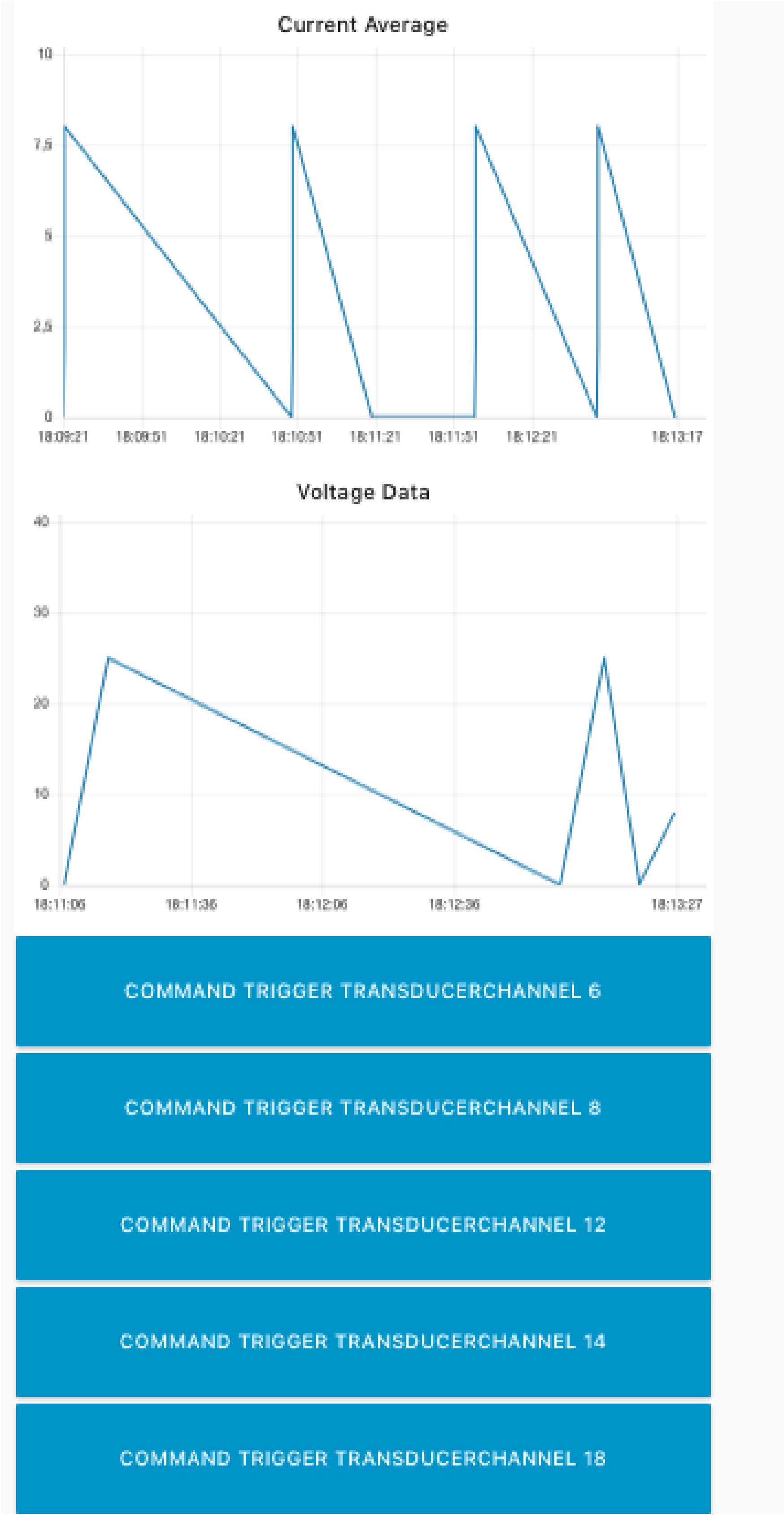

Figure 39: Part 2 of Text Tab of the DashBoard.

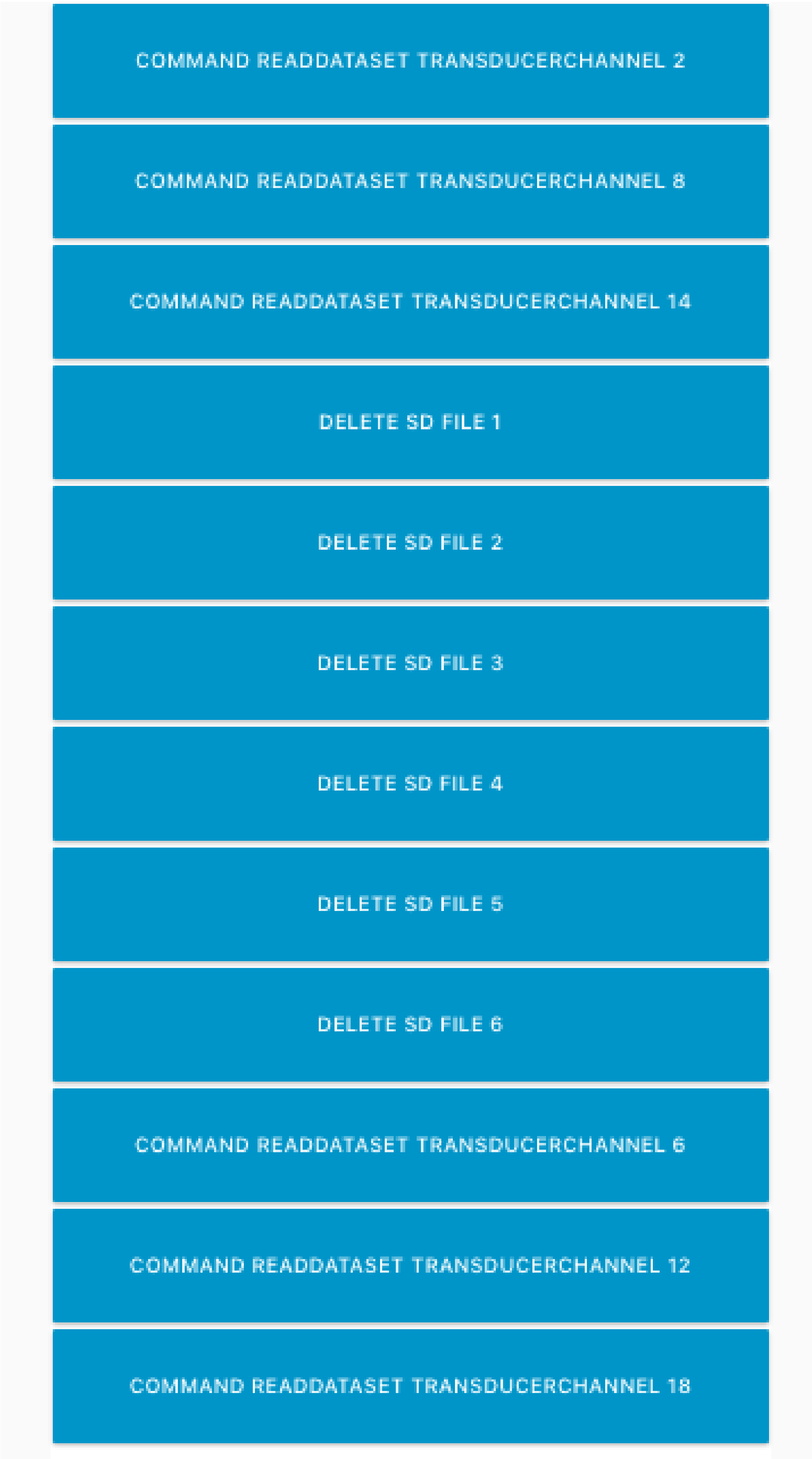

Figure 40: Part 3 of Chart Tab of the DashBoard.

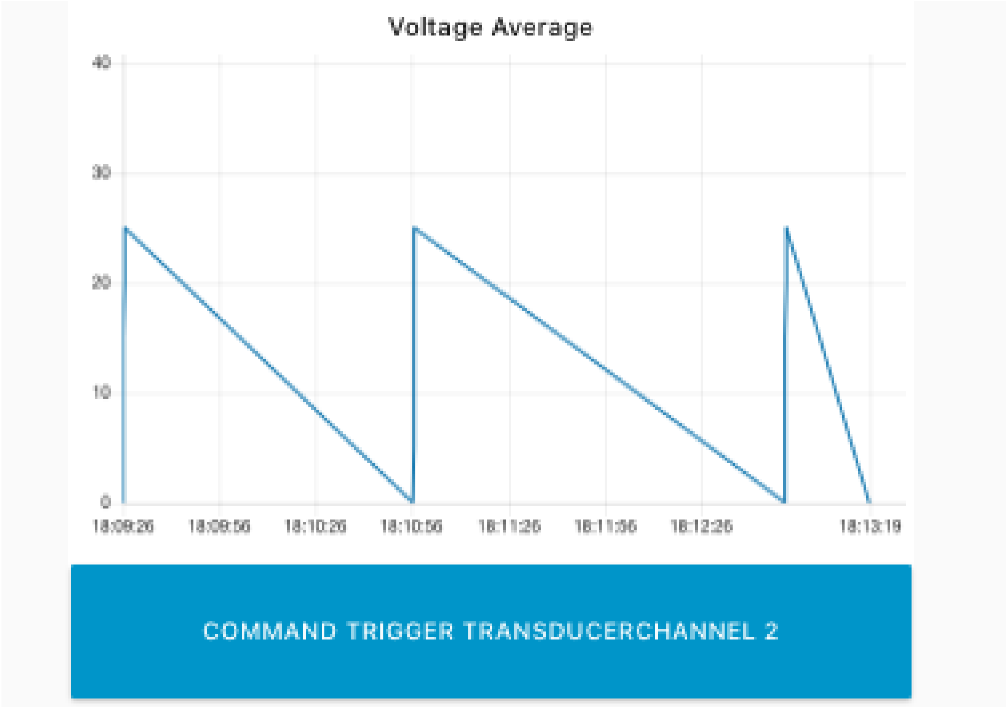

Figure 41: Part 4 of Text Tab of the DashBoard.

The buttons were pressed, and the charts show Power, Current and Voltage data and average data set in the C program in order to test the management app.

Chapter 6

Overall System

The final system developed is shown in Figure 41.

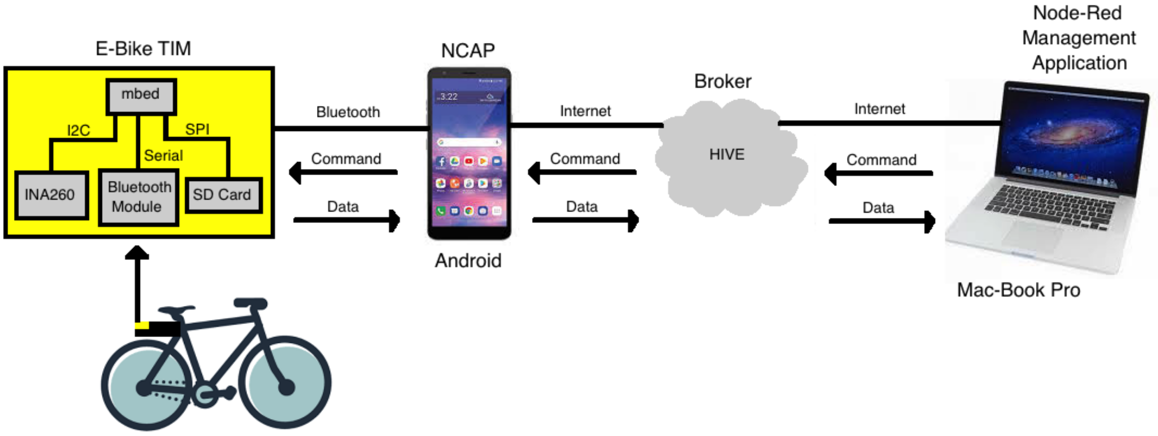

Figure 42: Working principle of eBike Power Management System.

The eBike Power Management System send commands from a Mac-Book to the HIVE MQTT broker through the internet using the Node-Red software. The HIVE broker sends commands to the Android NCAP through internet. The Android NCAP sends commands to the eBike TIM via Bluetooth.

Then the eBike TIM realize voltage, current and power data acquisition from the electric bicycle battery and sends the data back to the Android NCAP via Bluetooth. The Android NCAP sends the data to the HIVE Broker through the internet and the HIVE Broker sends the data to the Node-Red software in the Mac-Book through internet. In the Node-Red Management Application the data is analyzed.

Chapter 7

Validation Tests

7.1. Test of the system with MAC and power resistance

The first set of tests were performed using the system developed in PCB and a MAC Book in the Instrumentation and Measurement Laboratory (IML) in the Faculty of Engineering at Universidade da Beira Interior, Covilhã, Portugal.

A power source was used to power the INA sensor and a power resistor. The system developed was powered by the MAC book. The current, voltage and power of the power resistance were acquired by the INA sensor and stored in the SD card and also sent to the MAC book by Bluetooth. The current and voltage were changed, and therefore the power was changed and the data collected was analyzed in the MAC. The system worked well. The three parameters correspond to the ones provided by the power source.

7.2. Test of the system in an eBike with constant current

In the second set of tests, the system developed in PCB was tested using an eBike as the load in the Instrumentation and Measurement Laboratory (IML) in the Faculty of Engineering at Universidade da Beira Interior, Covilhã, Portugal. The experimental setup used is shown in Figure 43.

Figure 43: Experimental Setup for Test of the system in an eBike with constant current.

The system developed in PCB was powered by the battery of the eBike. The eBike current, voltage, and power were acquired by the INA sensor, stored in the system's SD card, and sent to the MAC Book by Bluetooth. The current was also measured by an oscilloscope. The data acquisition was performed for 1 minute, and we tried to keep the eBike pedal frequency constant to have an constant current. The results are shown in Figures 43 and 44.

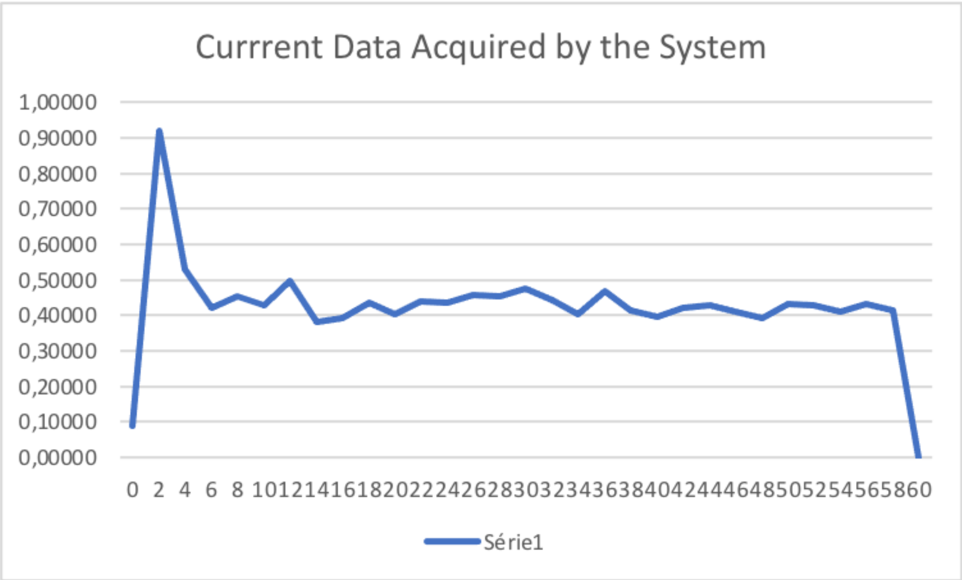

Figure 44: Chart of the eBike constant current acquired by the system developed.

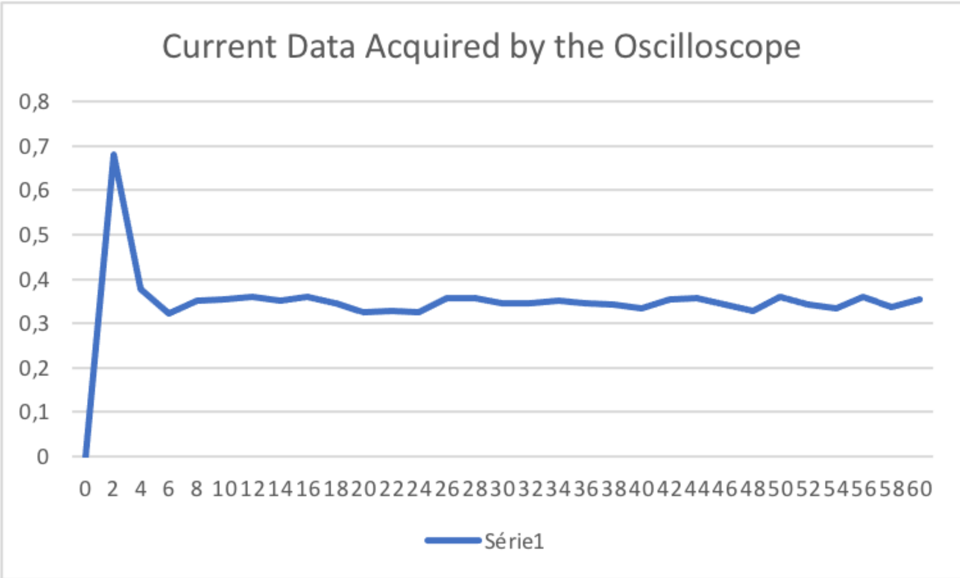

Figure 45: Chart of the eBike constant current measured by the oscilloscope.

It can be seen that the two charts present the same tendency with initial current zero and an almost constant current. The difference in current is probably because the current offset is not being corrected by the oscilloscope.

7.3. Test of the system in an eBike with variable current

The third set of tests were performed over the same conditions as the second set of tests but in this case the pedal frequency was variable and the current was intended to be variable. Figures 45 and 46 show the results of those tests.

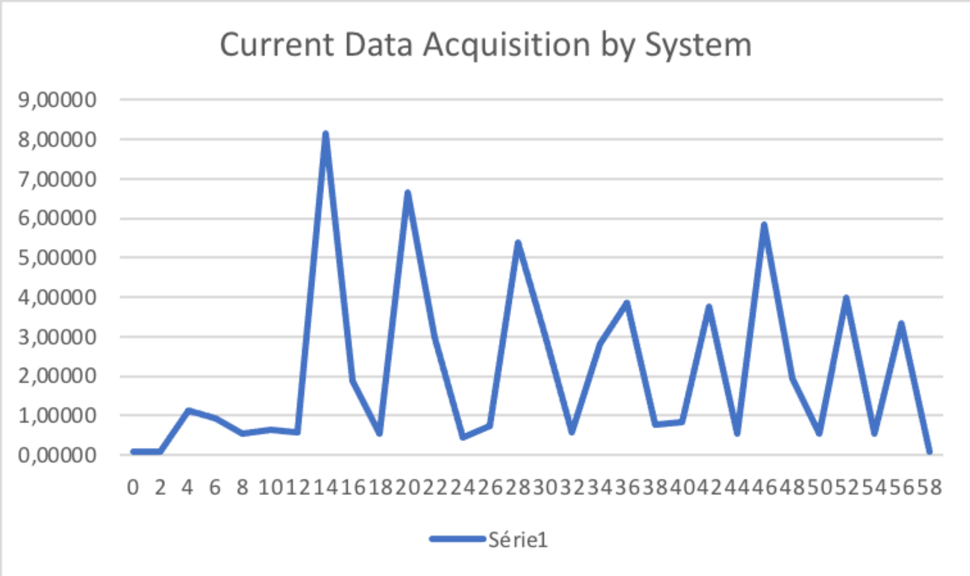

Figure 46: Chart of the eBike variable current acquired by the system developed.

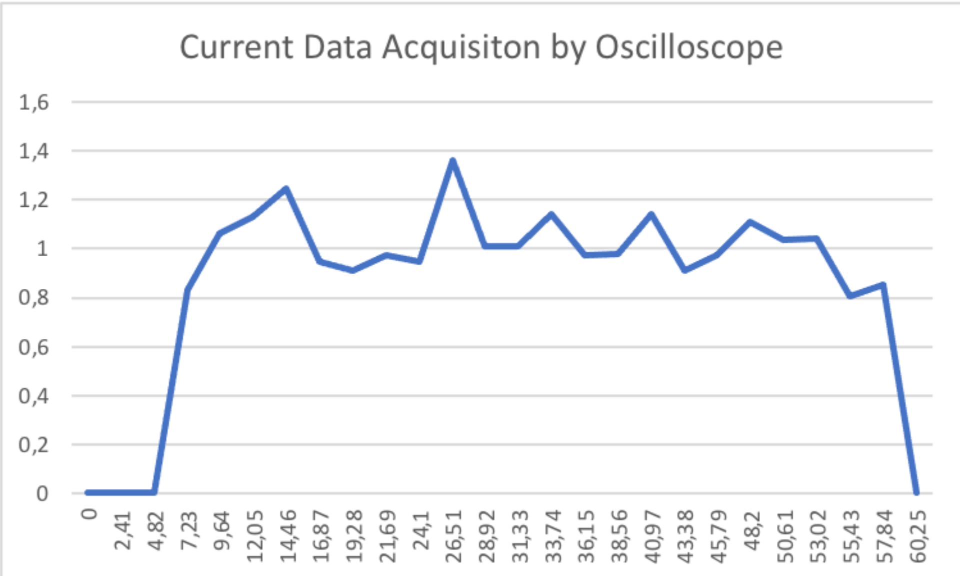

Figure 47: Chart of the eBike variable current measured by the oscilloscope.

The two charts show the same tendency, but the difference here may have come from the fact that the oscilloscope only captures the current when a threshold is reached. In this case only when the current was very different from its minimum value. Therefore, the system developed is better to measure current over a time than the oscilloscope.

7.4. Results Analysis

The Table next compares the systems explained in section 2.2.3 with the method developed here.

Table 5: Comparison between energymetters developed in the literature.

Paper/ System	Authors	Bicycles Tested	Local of Rides	Parameters Analyzed	Experimental Setup
[30]	HUNG at all (2019)	A pedelec with 21Kg, motor of 250W, Battery of Li-ion, 36V, 3Ah	Not Presented	electric consumption of the eBike according to bicycle mass, wheel radius and sprocket ratio.	Signal processing and measurement devices, a photoelectric sensor, an Electronic Control Unit, a Laptop and a voltage input module
[31]	HUNG at all (2019)	A pedelec with 21Kg, motor with 250W, Battery Li-ion, 36V,3Ah	Not Presented	Electric Consumption, Velocity, and distance run of the eBike according to air density and slope grade.	Signal processing and measurement devices, a photoelectric sensor, an Electronic Control Unit, a Laptop and a voltage input module
[32]	MENDES at all (2015)	A pedelec with several levels of electrical assistance, an eBike with different levels of assistance or electric assistance as the pedal intensity increases and a conventional bicycle	Lisbon, Portugal	energy consumption and speed	GPS, a National Instruments DAQ with voltage and current probes and a laptop
[33]	ABAGNALE at all (2015)	A pedelec with 26Kg, Brushless DC Machine of 250W, Battery of Li-Po, 48V, 10Ah	Naples, Italy	power requests, environmental parameters and driving range as a function of eBike working modes	Not Presented
[34]	ZARIPOV at all (2019)	Not Presented	Not presented	acceleration at different degrees of charge of the accumulation battery, speed, voltage, current, distance travelled and energy	Not presented
The system proposed in this dissertation.	MOREIRA, A. M.	A pedelec	Covilhã, Portugal	Battery Power, current and voltage	Microcontroller mbed, Bluetooth module HC05, INA260 sensor, MicroSD module, a smartphone and a laptop

It is observed that the energy meters were developed between 2015 and 2019; all the eBikes used were pedelecs, including the one used in this dissertation, or otherwise the eBike model was not presented; all the experiments were conducted in Europe or not presented; different parameters were analyzed such as electric parameters, dynamic parameters and environmental parameters and; the experimental setup varied a lot from one system to the other.

Three developments used a laptop that needed to be carried out along with the bicycle while the cyclist is pedalling the bicycle, making the experiments hard and dangerous. One contribution of the work proposed here is that the energy meter is coupled within the eBike and it has lightweight. In this way, the experiments can be conducted in convenient ways. The data can be processed away from the cyclist, in a lab or working centre, and the weight that the cyclist needs to carry is not increased much.

Chapter 8

Conclusions

The main categories of electric mobility are elevators and electric vehicles. The elevators are a reliable and convenient way of transport. Electric vehicles have been gaining global interest in the last years due to their capacity to reduce GHG and to their high efficiency, higher than ICV, Hydrogen, ethanol and biodiesel vehicles. There are different configurations of electric vehicles using various types of batteries, power converters, electric motors and gears. The batteries most used in EVs are Li-Ion batteries because of their high energy storage capacity, high voltage, quick charging and long life cycle.

Electric bicycles are a convenient way of transport because they are small, reduce traffic jams and improve the rider's health. The eBikes can be generally be classified in general, by function, structural features and purpose depending whether they are pedal-assisted, manned, the kind of frames, number of wheels, and where they are used.

The goal of the dissertation was to develop an embedded system capable of measure voltage, current and power from the electric battery of an electric bicycle, send data to a smartphone by Bluetooth, and from the smartphone to a computer using MQTT network. The system had to be developed under the IEEE1451 Standard.

The IEEE1451 is a family of 7 standards founded by IEEE to standardize smart transducers and allow interoperability. The IEEE 1451 standards suggest the addition of Transducer Electronic Data Sheets to the Smart transducer to store information about the transducers, besides the network communication, application processor, signal conditioning and data conversion and transducers from the basic smart transducer.

The methodology used in the dissertation was literature review of energy meters developed around the world to be compared to the system developed here later; study of IEEE1451 standards to understand how to apply the principles in the system development; design of the system based on data acquisition of sensors, actuators data push and the communication between them and the processor; hardware development using IoT principles and mbed microcontroller, INA260 sensor, HC05 Bluetooth module and a SD breakout board; firmware development based on pooling, interruption, state machines and TEDS; Smartphone app development based on Bluetooth and MQTT networks; Management application development based on node connections using software NodeRed; Finally tests performance and results analysis.

The system design was done modelling the components of the system as transducer channels according to the IEEE1451. The transducer channels are a transducer and all signal conditioning and conversion components associated with the transducer. The transducer channels used were Embedded Digital Actuator, Embedded Digital Event Sensor, Analog Input Sensor and Embedded Digital Sensor.

The hardware development was done by electronic circuit developments using I2C interface, serial interface and SPI Interface. Some tests were performed using a MAC Book Pro, power sources and power resistances. A PCB was modeled and assembled using schematic and layout developments with the Design Spark Software.

The firmware development was done in C language using the mbed online compiler. In the program, first, there are library inclusion, variable and function definition, configuration, declaration and initialization. Then in a loop function, there are the Bluetooth commands read to perform either transducer channels trigger, TEDS read or delete files. Later, state machines implement each transducer channel and acquisition functions to perform power, voltage and current data acquisition.

The smartphone application was developed in Java using Android Studio software. There is data exchange by Bluetooth between the system and the smartphone, and data exchange by MQTT protocol between the smartphone and the internet. The Bluetooth development was done through device search and connection and the MQTT development was made using publish and subscribe functions.

The Management Application was made using Flow Charts in the NodeRed software. The nodes button node, MQTT out node, MQTT in node, inject node, msg.payload node, split node, delay node and chart node were connected to communicate with the smartphone and generate a dashboard. The dashboard presents buttons to activate smartphone and system functions. The text data are shown in a scrolling table, and the electric parameter data are displayed in charts.

Besides the hardware tests, the system passed through 3 sets of tests. In the first set of tests there was electric parameters data acquisition from a power resistance. The data was sent to a MacBook by Bluetooth. The second set of tests were realized by measuring the electric parameters of a battery of an eBike with an approximately constant current. And the third set of tests were also done in an eBike but with varying current. The whole system worked well. The electric parameters data acquisition had the same tendency of the data acquired by an oscilloscope.

The data transfer between the system, the smartphone and the Management application was tested apart and the system responded to commands sent by the dashboard. The TEDS were received and showed correctly in a scrolling table. The electrical parameters data were received and showed in charts correctly.

The system developed here may contribute to perform tests to verify the electric consumption, current, voltage and power of eBike in different conditions such as varying the cyclist mass, eBike mass, acceleration conditions, speed conditions, air conditions, slope degree and routes. This may allow to discover the most economical routes or trace the user profile. In this way, together with air pollution gas tracking and Photovoltaic eBike charging, this project may contribute to urban mobility sustainability and better quality of life.

In comparison to the energy meters revised in the literature, the system developed here is integrated with the eBike and has lightweight. This almost does not interfere with the rider pedalling. In this way, the experiments can be conducted quickly and in convenient ways. On the other hand, the systems studied in the literature had to carry a laptop within the eBike while performing the tests.

Bibliography

[1] COX, D. "History of Electric Cars". Presentation at University of Nebraska – Lincoln. 2012.

[2] SEIXAS J., SIMÕES S., DIAS L., KNAUDIA A., FORTES P., M. Gargiulo, "Assessing the cost-effectiveness of electric vehicles in European coutries using integrated modeling," Energy Policy, 2015.

[3] NANAKI, E. A. KORONEOS, C. J. "Climate change mitigation and deployment of electric vehicles in urban areas" Renewable Energy 99 (2016) 1153e1160.

[4] ROBINSON, A. P. BLYTHE, P. T. BELL, M. C. HÜBNER, Y. HILL G. A. "Analysis of electric vehicle driver recharging demand profiles and subsequent impacts on the carbon content of electric vehicle trips" Energy Policy 61 (2013) 337–348.

[5] TEIXEIRA, A. C. R. SODRÉ, J. R. "Simulation of the impacts on carbon dioxide emissions from replacement of a conventional Brazilian taxi fleet by electric vehicles" Energy 115 (2016) 1617e16221618.

[6] PINTO, R., ESPÍRITO-SANTO, A., PACIELLO, V., "Proposal of a Sustainable e-Bike Sharing Infrastructure Based on the IEEE1451 Standard" IECON 2019 45th Conference of the IEEE Industrial Society.

[7] SONG, E. Y., LEE, K., "Understanding IEEE 1451 – Networked Smart Transducer Interface Standard." IEEE Instrumentation & Measurement Magazine 1094-6969/08/$25.00©2008IEEE.

[8] LEE, K., "IEEE 1451: A Standard in Support of Smart Transducer Networking." 0- 7803-5890-2/00/$10.000 2 000 IEEE.

[9] ROCHA NETO, J. S. "Introdução ao Estudo de Sensores/Atuadores Inteligentes Padrão IEEE 1451 Redes de Sensores Sem Fio" Power Point Presentation provided by the Author.

[10] MARK, J., HUFNAGEL, P., "The IEEE 1451.4 Standard for Smart Transducers." IEEE Standards Association.

[11] ANANDALAL ELECTRIC PVT LTD. "A Short History of Electric Vehicles." Available in: <http://www.anandalalelectric.com/short-history-of-electric-vehicles>. Accessed in July 2016.

[12] SEIXAS J., SIMÕES S., DIAS L., KNAUDIA A., FORTES P., M. Gargiulo, "Assessing the cost-effectiveness of electric vehicles in European coutries using integrated modeling," Energy Policy, 2015.

[14] Samuel E. de Lucena. "A Survey on Electric and Hybrid Electric Vehicle Technology, Electric Vehicles - The Benefits and Barriers", Dr. Seref Soylu (Ed.), InTech, DOI: 10.5772/18046. Available in: http://www.intechopen.com/books/electric-vehicles-the-benefits-and-barriers/a-survey- on-electric-and-hybrid-electric-vehicle-technology. Accessed in July 2016.

[15] XIANZHI G., RUI X., CHRIS M. Chunting, "Study of The Characteristics of Battery Packs in Electric Vehicles With Parallel-Connected Lithium-Ion Battery Cells." IEEE Transactions on Industry Applications, Vol 51, No. 2, March/April 2015.

[16] BUCKNER, M. "Lithium-ion Battery Technology for Vehicle Electrification". 2012.

[17] COX, D. EV Motors part 1. University of Nebraska – Lincoln. 2012.

[18] GUOQING X., KUN X., CHUNHUA Z., XINYE Z., TAIMOOR Z., "Fully Electrified Regenerative Braking Control for Deep Energy Recovery and Maintaining Safety of Electric Vehicles" IEEE Transactions on Vehicluar Technology, Vol. 65, NO.3, March 2016.

[19] The University of Tenessee. "Electric Vehicles". Available in: <http://www.utc.edu/college-engineering-computer-science/research- centers/cete/electric.php>. Accessed in July 2016.

[20] EBERHARD, M. "The Big Picture". Presentation at University of Nebraska, Lincoln. 2012.

[21] MOREIRA, Marcelo, A. Nayara, M. Lais, G. Aldo, "Veículos Elétricos," Essay presented at Federal University of Campina Grande. 2015.

[22] COX, D. History of Electric Cars. University of Nebraska – Lincoln. 2012

[23] SUN, W. LI, Y. HUANG, J. ZHANG, N. "Vibration effect and control of In- Wheel Switched Reluctance Motor for electric vehicle" Journal ofSoundandVibration338(2015)105–120.

[24] ABDEL-KADER, F. E. S. ELSHERIF, M. Z. ABDEL-RAHIM, Naser M. B. FATHY, Mohamed M. "Control methods of the switched reluctance motor in electric vehicle during acceleration" Journal of Renewable and Sustainable Energy 4 , 063142 (2012).

[25] PELLEGRINO, G. GUGLIELMI P. "Performance Comparison Between Surface-Mounted and Interior PM Motor Drives for Electric Vehicle Application" IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 59, NO. 2, FEBRUARY 2012.

[26] NIAN, X. PENG F. ZHANG, H. "Regenerative Braking System of Electric Vehicle Driven by Brushless DC Motor" IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 61, NO. 10, OCTOBER 2014.

[27] XIN, X. SHAN, H. YANG, Y. "Control Strategy for Motor Drive System of Electric Vehicle" IEEJ TRANSACTIONS ON ELECTRICAL AND ELECTRONIC ENGINEERING IEEJ Trans 2016; 11: 374–383.

[28] LEI, F. DU, B. LIU, X. XIE, X. CHAI, T. "Optimization of an implicit constrained multi-physics system for motor wheels of electric vehicle" Energy 113 (2016) 980e990.

[29] HUNG, N. B., LIM, O. "A review of history, development, design and research of electric bicycles" Applied Energy 260 (2020) 114323.

[30] HUNG, N. B., LIM, O. "The effects of operating conditions and structural parameters on the dynamic, electric consumption and power generation characteristics of an electric assisted bicycle." Applied Energy 247 (2019) 285–296286.

[31] HUNG, N. B., LIM, O. "A simulation and experimental study of dynamic performance and electric consumption of an electric bicycle" Energy Procedia 158 (2019) 2865–2871.

[32] ONAT, N. C., KUCUKVAR, M., ABOUSHAQRAH, NOUR N. M., JABBAR, R. "How sustainable is electric mobility? A comprehensive sustainabilityassessment approach for the case of Qatar" Applied Energy 250 (2019) 461–477462.

[33] ABGNALE, C., CORDONE, M., IODICE, P. STRANO, S., TERZO, M., VORRARO, G. "Power requirements and environmental impact of a pedelec. A case study based on real-life applications" Environmental Impact Assessment Review 53 (2015) 1–7.

[34] ZARIPOV, R., GAVRILOVS, P. "Study of dynamic characteristics of electric bicycles" *Procedia Computer Science 149 (2019) 307–313*.

[35] ZARIKAS, V., LOUPIS, M., PAPANIKOULAOU, N., KYRITSI, C., "Statistical survey of elevator accidents in Greece" Safety Science 59 (2013) 93–103.

[36] IEEE21450, "Information technology — Smart transducer interface for sensors and actuators — Common functions, communication protocols, and Transducer Electronic Data Sheet (TEDS) formats" First edition 2010-05-15.

Appendix 1: TEDS in Hexadecimal Format

****
* File: teds.txt
* Name: tim teds 4
* Description: INA260
* Created: 2021-04-04 00:56:30
* Autor: iml@ubi.pt
****

****
META-TEDS
****

0x00 0x00 0x00 0x2C
0x03 0x04 0x00 0x00 0x00 0x01
0x04 0x0A 0x95 0x3D 0xF4 0x1A 0x68 0x81 0xF8 0xF8 0xCF 0xA4
0x0A 0x04 0x41 0x20 0x00 0x00
0x0B 0x04 0x00 0x00 0x00 0x00
0x0C 0x04 0x00 0x00 0x00 0x00
0x0D 0x02 0x00 0x09
0x0E 0x18 0x15 0x01 0x07 0x16 0x03 0x01 0x02 0x00 0x15 0x01 0x07 0x16 03 0x04 0x05
0x00 0x15 0x01 07 0x16 0x03 0x07 0x08 0x00
0xF7 0xC7

****
TransducerChannel 1 TEDS (Digital Embedded Event Sensor Transducer Channel of Power)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x3
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01

0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x93

****
TransducerChannel 2 TEDS (Analog Input Sensor Transducer Channel of Power)
****

0x00 0x00 0x00 0x50
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x0F 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x80 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x44 0x07 0x00 0x00
0x0F 0x04 42 0xA2 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x90

****
TransducerChannel 3 TEDS (Digital Embedded Actuator Transducer Channel of Power)
****

0x00 0x00 0x00 0x50
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x0F 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x80 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x44 0x07 0x00 0x00
0x0F 0x04 42 0xA2 0x00 0x00

0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x90

****
TransducerChannel 4 TEDS (Digital Embedded Event Sensor Transducer Channel of SD and Power)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x06 0xBD 0xF5
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x01

0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x99

****
TransducerChannel 5 TEDS (Digital Embedded Actuator Transducer Channel of SD and Power)
****

0x00 0x00 0x00 0x50
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x0F 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x7A 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x44 0x07 0x00 0x00
0x0F 0x04 42 0xA2 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x90

****
TransducerChannel 6 TEDS (Digital Embedded Sensor Transducer Channel of SD and Power)
****

0x00 0x00 0x00 0x50
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x0F 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x80 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x44 0x07 0x00 0x00
0x0F 0x04 42 0xA2 0x00 0x00
0x10 0x01 00
0x11 0x01 00

0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x00
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x90

****
TransducerChannel 7 TEDS (Digital Embedded Event Sensor Transducer Channel of Voltage)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x3
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00

0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x93

****

TransducerChannel 8 TEDS (Analog Input Sensor Transducer Channel of Voltage)

****

0x00 0x00 0x00 0x53
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x12 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x80 0x3A 0x01 0x800x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x42 0x10 0x00 0x00
0x0F 0x04 40 0xAC 0xCC 0xCD
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x8D

****

TransducerChannel 9 TEDS (Digital Embedded Actuator Transducer Channel of Voltage)

****

0x00 0x00 0x00 0x53
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x12 0x0x32 0x01 0x0 0x35 0x01 0x84 0x36 0x01 0x82 0x37 0x01 0x80 0x38 0x01 0x80 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x42 0x10 0x00 0x00
0x0F 0x04 40 0xAC 0xCC 0xCD
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01

0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x8D

****
TransducerChannel 10 TEDS (Digital Embedded Event Sensor Transducer Channel of SD and Voltage)
****

0x00 0x00 0x00 0x53
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x12 0x0x32 0x01 0x0 0x35 0x01 0x84 0x36 0x01 0x82 0x37 0x01 0x80 0x38 0x01 0x80
0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x42 0x10 0x00 0x00
0x0F 0x04 40 0xAC 0xCC 0xCD
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00

0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x8D

****
TransducerChannel 11 TEDS (Digital Embedded Actuator Transducer Channel of SD and Voltage)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x06 0xBD 0xF5
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x01
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x99

****
TransducerChannel 12 TEDS (Digital Embedded Sensor Transducer Channel of SD and Voltage)
****

0x00 0x00 0x00 0x53
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x12 0x0x32 0x01 0x0 0x37 0x01 0x84 0x38 0x01 0x82 0x39 0x01 0x80 0x3A 0x01 0x800x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x42 0x10 0x00 0x00
0x0F 0x04 40 0xAC 0xCC 0xCD
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01

0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x3E8
0x20 0x01 0x3E8
0x21 0x01 0x00
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x8D

****
TransducerChannel 13 TEDS (Digital Embedded Event Sensor Transducer Channel of Current)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x3
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03

0xFF 0x93

****
TransducerChannel 14 TEDS (Analog Input Sensor Transducer Channel of Current)
****

0x00 0x00 0x00 0x4A
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x09 0x0x32 0x01 0x0 0x38 0x01 0x82 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x41 0x70 0x00 0x00
0x0F 0x04 40 0x10 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x96

****
TransducerChannel 15 TEDS (Digital Embedded Actuator Transducer Channel of Current)

****

0x00 0x00 0x00 0x4A
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x09 0x0x32 0x01 0x0 0x38 0x01 0x82 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x41 0x70 0x00 0x00
0x0F 0x04 40 0x10 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00

0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x96

****
TransducerChannel 16 TEDS (Digital Embedded Event Sensor Transducer Channel of SD and Current)
****

0x00 0x00 0x00 0x4A
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x1
0x0C 0x09 0x0x32 0x01 0x0 0x38 0x01 0x82 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x41 0x70 0x00 0x00
0x0F 0x04 40 0x10 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x40 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x00 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x02
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x01
0xFF 0x96

****

TransducerChannel 17 TEDS (Digital Embedded Actuator Transducer Channel of SD and Current)
****

0x00 0x00 0x00 0x47
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x2
0x0C 0x06 0x0x32 0x01 0x4 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x3F 0x80 0x00 0x00
0x0F 0x04 3F 0x80 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 04 0x29 0x01 02 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x40 0x00 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x06 0xBD 0xF5
0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x01
0x22 0x01 0x00
0x23 0x01 0x01
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x99

****
TransducerChannel 18 TEDS (Digital Embedded Sensor Transducer Channel of SD and Current)
****

0x00 0x00 0x00 0x4A
0x03 0x04 0x00 0x00 0x00 0x01
0x0A 0x01 0x0
0x0B 0x01 0x0
0x0C 0x09 0x0x32 0x01 0x0 0x38 0x01 0x82 0x3C 0x01 0x80
0x0D 0x04 0x00 0x00 0x00 0x00
0x0E 0x04 0x41 0x70 0x00 0x00
0x0F 0x04 40 0x10 0x00 0x00
0x10 0x01 00
0x11 0x01 00
0x12 0x28 0x01 02 0x29 0x01 00 0x2A 0x01 01
0x13 0x2B 0x01 0x00 0x2C 0x01 0x00 0x2D 0x01 0x00 0x2E 0x01 0x00 0x2F 0x01 0x00
0x14 0x04 0x3F 0x80 0x00 0x00
0x15 0x04 0x00 0x00 0x00 0x00
0x16 0x04 0x00 0x00 0x00 0x00
0x17 0x04 0x3C 0x07 0x11 0xD8

0x18 0x04 0x40 0x00 0x00 0x00
0x19 0x04 0x40 0x00 0x00 0x00
0x1A 0x04 0x00 0x00 0x00 0x00
0x1B 0x01 0x00
0x1C 0x04 0x00 0x00 0x00 0x00
0x1D 0x04 0x00 0x00 0x00 0x00
0x1E 0x04 0x00 0x00 0x00 0x00
0x1F 0x30 0x01 0x00
0x31 0x01 0x00
0x20 0x01 0x00
0x21 0x01 0x00
0x22 0x01 0x00
0x23 0x01 0x00
0x24 0x01 0x00
0x25 0x04 0x00 0x00 0x00 0x00
0x26 0x08 0x00 0x00 0x00 0x00 0x00 0x00 0x00 00
0x27 0x01 0x03
0xFF 0x96

****
Users TransducerChannel Name TEDS process
****

0x00 0x00 0x00 0x13
0x03 0x04 0x00 0x0C 0x01 0x01
0x04 0x01 0x00
0x05 0x06 0x49 0x4E 0x41 0x32 0x36 0x30 0x
0xFE 0x57

****
PHY TEDS
****

0x00 0x00 0x00 0xD
0x03 0x04 0x00 0x0C 0x01 0x01
0x04 0x01 0x00
0xFF 0xD8