**THE UNIVERSITY OF DANANG**



**UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**FACULTY OF MECHANICAL ENGINEERING**

A blue and yellow logo

Description automatically generated A logo with a red and blue gear

Description automatically generated

**PBL 3: DESIGNING EMBEDDED SYSTEMS WITH MICROCONTROLLERS AND SENSORS**

**PROJECT: DIGITAL WEIGHING SCALE FOR SELLING GOODS**

**Lecturers: Ph.D DANG PHUOC VINH**

**Student: LE VAN TRUNG**

**NGUYEN PHU LINH**

**Student ID Number: 101200251**

**101200274**

**Class: 20CDT1 & 20CDT2**

**Danang, May 2024**

|  |  |
| --- | --- |
| **DA NANG UNIVERSITY OF SCIENCE**  **AND TECHNOLOGY** | **THE SOCIALIST REPUBLIC OF VIETNAM**  Independent – Freedom - Happiness  **––––––––––––––––––––––** |
|  |  |

**MISSION OF PROJECT PBL3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Num | Name | ID | Class | Industry |
| 1 | Nguyen Phu Linh | 101200274 | 20CDT2 | Mechatronic Engineering |
| 2 | Le Van Trung | 101200251 | 20CDT1 | Mechatronic Engineering |

**1.Name of project:** Digital Weighing Scale For Selling Goods

**2.The subject is subject to:**  Having signed an agreement on intellectual property for the results of implementation

**3. Content of the explanations and caculations:**

a) General part:

|  |  |  |
| --- | --- | --- |
| Num | Name | Content |
| 1  2 | Nguyen Phu Linh  Le Van Trung | - Get an overview about Digital Weighing Scale For Selling Goods  - Hardware and software design caculation  - Prepare presentation, slides and drawings as required |

b) Private part:

|  |  |  |
| --- | --- | --- |
| Num | Name | Content |
| 1 | Nguyen Phu Linh | - Calculation and design of electronic  - Design HMI interface to monitor system operation  - System control programming |
| 2 | Le Van Trung | - Calculation and design control parts  - Calculation of mechanical design |

**4. Drawing, graphs (specify types and sizes of drawings):**

a) General part:

|  |  |  |
| --- | --- | --- |
| Num | Name | Content |
| 1  2 | Nguyen Phu Linh  Le Van Trung | - Make the overall drawing of the machine (1 – A0) |

b) Private part:

|  |  |  |
| --- | --- | --- |
| Num | Name | Content |
| 1 | Nguyen Phu Linh | - Make an algorithm flowchart drawing |
| 2 | Le Van Trung | - Make a drawing of an electrical circuit diagram |

**5. Lectures: Ph.D DANG PHUOC VINH**

**6. Project assignment date:** 11/2/2024

**7. Project completion date:** 14/6/2024

|  |  |
| --- | --- |
|  | *Da Nang, June 13, 2024* |
| **Head of subject**  **Ph.D VO NHU THANH** | **Lectures** |

**CONTENTS**

[**INTRODUCTION 7**](#_Toc169104607)

[**CHAPTER 1: INTRODUCTION TO ELECTRONIC SCALES AND COMPUTER COMMUNICATION 8**](#_Toc169104608)

[I. General Introduction: 8](#_Toc169104609)

[1. Enhanced Accuracy: 8](#_Toc169104610)

[2. Increased Operational Efficiency: 8](#_Toc169104611)

[3. Enhanced Brand Image: 9](#_Toc169104612)

[4. Effective Inventory Management: 9](#_Toc169104613)

[5. Supports Diverse Features: 9](#_Toc169104614)

[II. Reasons for Choosing the Topic: 10](#_Toc169104615)

[1. Some electronic scales on the market: 10](#_Toc169104616)

[2. Urgency: 11](#_Toc169104620)

[3. Objectives: 12](#_Toc169104621)

[**CHAPTER 2: HARDWARE DESIGN 14**](#_Toc169104622)

[I. Block Diagram 15](#_Toc169104625)

[II. Voltage Conversion Circuit from 12VDC to 5VDC 15](#_Toc169104626)

[1. Schematic Diagram: 15](#_Toc169104627)

[2. Important components: 16](#_Toc169104628)

[III. Microcontroller Circuit 19](#_Toc169104629)

[1.Schematic Diagram: 19](#_Toc169104630)

[2.Loadcell and Module HX711 25](#_Toc169104633)

[3.UART-RS232 Communication: 29](#_Toc169104635)

[4.LCD Display (Liquid Crystal Display) 32](#_Toc169104636)

[5.Camera and Barcode: 34](#_Toc169104638)

[**CHAPTER 3: SOFTWARE DESIGN 39**](#_Toc169104642)

[Programming for the system 39](#_Toc169104643)

[**CHAPTER 4 DESIGN THE SYSTEM'S SALES MANAGEMENT INTERFACE AND DATABASE**](#_Toc169104644)

[1.Visual Studio 2022 Software 41](#_Toc169104646)

[2.Microsoft SQL Server 2022 Software 41](#_Toc169104647)

[3.User Interface Software 42](#_Toc169104648)

[4.Code C# 46](#_Toc169104650)

[**CHAPTER 5 LESSON LEARNED AND CONCLUSION 64**](#_Toc169104651)

[1. Overall Evaluation 64](#_Toc169104652)

[2. Advantages and disadvantages of the product 64](#_Toc169104655)

[3. Future Development Directions 64](#_Toc169104656)

[**REFERENCE 65**](#_Toc169104657)

# **INTRODUCTION**

In today's rapidly evolving technological landscape, the fields of mechanical and electronic engineering play pivotal roles in driving economic progress. As students hailing from the Department of Mechatronics at the University of Science and Technology, we've traversed through several years of academic rigors, enriched by the mentorship of seasoned professors. Armed with a wealth of knowledge and honed skills, we've resolved to undertake a project that is not only intellectually stimulating but also socially impactful: the development of an integrated electronic weighing system coupled with advanced sales management functionalities.

Our motivation for selecting this particular project extends beyond the realms of academic pursuit; it's deeply rooted in our aspiration to address real-world challenges and cater to the demands of contemporary society. In an era dominated by digital transformation, the creation of an intelligent electronic weighing system promises not just to streamline sales processes but also to revolutionize the shopping experience, offering consumers unparalleled convenience and efficiency.

Acknowledging the hurdles that lie ahead – from our own limited technical expertise to the pressing demands of time – we approach this endeavor with steadfast determination. We recognize that navigating through these challenges necessitates the guidance and expertise of seasoned professionals. Thus, we eagerly welcome and value the insights and feedback from our esteemed professors, knowing well that their mentorship will be instrumental in our journey towards success.

Lastly, we wish to express our heartfelt gratitude to all the faculty members within the Department of Mechatronics, with a special mention to Mr. Dang Phuoc Vinh, whose unwavering support and encouragement have been the cornerstone of our academic and professional growth. It is through their guidance that we not only aim to accomplish this project but also aspire to emerge as influential engineers poised to make meaningful contributions to the future of our field and society at large.

Finally, we would like to express sincere gratitude to all the teachers in the Department of Mechatronics, especially to Mr. Dang Phuoc Vinh, for their continuous support and encouragement throughout our journey. Your guidance has not only helped us complete this project but has also shaped us into influential engineers for the future.

Danang, June 13, 2024

Author

NGUYEN PHU LINH

LE VAN TRUNG

# **CHAPTER 1: INTRODUCTION TO ELECTRONIC SCALES AND COMPUTER COMMUNICATION**

## **I. General Introduction:**

In the era of booming digital technology, the application of advanced solutions to business operations is essential to increase efficiency and affirm brand position. And electronic sales scales are one of the indispensable powerful tools for businesses, especially in the retail sector.

**The Essential Role of Electronic Scales:**

Electronic scales are not just ordinary measuring devices; they offer a multitude of significant benefits to businesses, contributing to sustainable development. Here are the reasons why electronic sales scales have become the key to success:

### **1. Enhanced Accuracy:**

Electronic scales boast far greater accuracy than traditional scales, eliminating any errors in the measurement process. This ensures fairness for both buyers and sellers while also boosting customer confidence in the business's products and services.

**Examples:**

* **Selling fruits and vegetables:** Instead of estimating by eye, electronic scales help calculate the exact weight, allowing for fair pricing of products and increasing store revenue.
* **Selling meat, fish, and seafood:** Ensure customers receive the correct quantity of purchased products, avoiding shortages that lead to disputes and loss of customers.

### **2. Increased Operational Efficiency:**

Electronic scales help optimize the sales process, saving time and manpower. With the ability to measure quickly and accurately, sales staff can serve customers faster, reduce queues, and boost store revenue.

**Examples:**

* **In supermarkets:** Electronic scales with built-in printing capabilities save cashiers time and expedite customer checkout.
* **In retail stores:** Electronic scales automatically calculate prices based on product weight, eliminating manual tasks and minimizing errors.

### **3. Enhanced Brand Image:**

Using electronic scales demonstrates a business's professionalism and modernity, building customer trust and enhancing brand value. Customers appreciate the transparency and accuracy of measurements, encouraging them to return and recommend the store to friends.

**Examples:**

* **Using a reputable brand of electronic scales** will help businesses elevate their market position and attract potential customers.
* **Electronic scales with clear displays** allow customers to easily follow the measurement process, increasing trust and creating a positive impression of the store.

### **4. Effective Inventory Management:**

Electronic scales help accurately track the quantity of goods in stock, ensuring that enough products are always available to meet customer demand. This allows businesses to avoid stockouts or overstocking, saving costs and increasing profits.

**Examples:**

* **Using inventory management software connected to electronic scales** enables businesses to automatically track the quantity and cost of each item, optimizing ordering and minimizing losses.
* **Portable electronic scales** allow for weight checks of goods directly in the warehouse, saving time and effort for employees.

### **5. Supports Diverse Features:**

Modern electronic scales are equipped with various advanced features such as printing receipts, connecting to computers, managing sales data, and more. This helps businesses manage their operations more effectively, increase productivity, and make informed decisions.

**Examples:**

* **Analyzing sales data from electronic scales** helps businesses identify which products are selling well and which are not, allowing them to adjust their business strategies accordingly.
* **Using electronic scale data to forecast market demand** helps businesses plan their procurement effectively, avoiding stockouts or overstocking.

Electronic sales scales are not just measuring tools; they are a smart investment in business success. Upgrade to electronic scales now to enhance operational efficiency, increase profits, and affirm your market position.

Therefore, we have approached and implemented the manufacturing of electronic scales with a system using the PIC16F877A microcontroller and communicating with the computer using QR code technology to identify and quantify items for payment. To develop the software, we use CCS for microcontroller programming and Visual Studio 2022 with the C# programming language to develop the software component of the system.

## **II. Reasons for Choosing the Topic:**

1. **Some electronic scales on the market:**





Figure 1.1: Commercial Electronic Scale

1. **Urgency:**

In today's fiercely competitive market, enhancing operational efficiency and increasing profits is the top priority for every business. And electronic sales scales are the essential solution to help businesses achieve this goal.

Using electronic scales offers numerous significant benefits. However, many businesses are still hesitant and do not fully recognize the urgency of adopting this technology. Here are some reasons why electronic sales scales have become more crucial than ever before:

**Increasing Customer Demands:**

Today's customers increasingly demand accuracy, transparency, and convenience when shopping. They want their product prices to be calculated precisely based on weight and to check out quickly without waiting long. Electronic scales perfectly meet these needs, helping to enhance the customer’s shopping experience and increase customer retention.

**Intensifying Competition:**

In today's market, businesses face intense competition from their rivals. To attract customers and elevate their competitive position, businesses need to improve operational efficiency and provide better service. Electronic scales help businesses optimize the sales process, save costs, and increase profits, thereby strengthening their market competitiveness.

**Technology Adoption as an Inevitable Trend:**

Digital technology is rapidly evolving and permeating every aspect of life. Applying technology to business operations is an inevitable trend for businesses to enhance productivity, efficiency, and make informed decisions. Electronic scales are a prime example of technological application, helping businesses digitally transform and elevate their business operations.

**Reasonable Investment Costs:**

Nowadays, the cost of electronic scales has significantly decreased compared to the past. With a wide range of prices to suit every budget, businesses can easily choose the product that aligns with their needs. Compared to the immense benefits electronic scales bring, this is just a small investment that helps businesses increase revenue and profits in the long run.

**Risks of Losses Due to Inaccuracies:**

Using traditional measurement methods such as hand scales or visual estimation carries numerous risks of inaccuracies, leading to business losses such as:

* **Loss of customer trust:** Customers will be dissatisfied if they are charged the wrong price, leading to **lost customers** and **damage to brand reputation**.
* **Product loss:** Inaccuracies in measurement can lead to **product loss**, causing **financial losses** for the business.
* **Customer disputes:** Inaccuracies in measurement can lead to **customer disputes**, affecting **business harmony** and the company's image.

1. **Objectives:**

**Design and develop an electronic scale and sales management system with the following functions:**

* + **Weight measurement:** Utilize a load cell connected to the HX711 module to convert analog data to digital data and amplify the signal.
  + **Functional buttons:** Include four function buttons: a reset button to zero the weight value, a hold button to retain the weight value, a count button to count the number of weighing, and a unit conversion button (kg, g, oz).
  + **Display:** Show the weight value and mode on a 16×2 LCD screen.
  + **Sales management system:** The system will process QR codes to read the code and obtain the weight value from the electronic scale to calculate the price.
  + **Print sales receipts:** Print sales receipts after the scanning process is complete.
  + **Product management:** Include functions to add new products, delete and update items, and manage other information within the system.

**Expected Outcomes:**

* **Improved operational efficiency:** The system will streamline the sales process, reducing time and labor costs.
* **Enhanced accuracy:** Electronic scales will eliminate measurement errors, ensuring fair pricing and customer satisfaction.
* **Increased customer satisfaction:** The system will provide a faster and more convenient checkout experience for customers.
* **Better inventory management:** The system will track product inventory levels, helping to prevent stockouts and overstocking.
* **Increased sales and profits:** The combination of these benefits will lead to increased sales and profits for the business.

**Target Users:**

The system is primarily designed for retail businesses, particularly those that sell products by weight. However, it can also be adapted for use in other industries, such as warehouses, logistics, and manufacturing.

**Scope of Work:**

The project scope includes the following tasks:

* **Hardware design:** Design the electronic scale circuit using the load cell, HX711 module, and microcontroller.
* **Software development:** Develop the microcontroller program to read the weight data from the load cell, process the data, and control the LCD display and function buttons.
* **Sales management system development:** Develop the sales management system software to process QR codes, calculate

# **CHAPTER 2: HARDWARE DESIGN**

****

**A black and white electronic device with a wire

Description automatically generated**

Figure 2.1**:** Actual model of the product

## **I. Block Diagram**

The electronic scale is composed of a load cell force sensor, an HX711 module, a PIC 16F877A microcontroller, and a computer communicating via UART-RS232.

A diagram of a computer system

Description automatically generated

Figure 2.2: Main block diagram

**Operating Principle:**

The PIC16F877A microcontroller processes data and connects to other devices. The load cell is used to measure weight through the HX711 module transmitted to the microcontroller. The computer with the Visual Studio interface receives data from the LCD and PIC and processes it.

## **II. Voltage Conversion Circuit from 12VDC to 5VDC**

### **1. Schematic Diagram:**

A 12V power supply is used to power the IC7805, which has a relatively stable output of 5V, to power the microcontroller and other circuit components.

A B688 BJT is used to amplify the current for the circuit.

A 100Ω / 5W resistor is used to protect the B688 BJT.

Electrolytic capacitors are selected to filter out ripple noise at the input and output of the circuit.

**Components:**

* IC7805: 5V voltage regulator
* B688: NPN transistor
* R1: 100Ω / 5W resistor
* C1, C2: 100µF electrolytic capacitors
* D1: 1N4001 diode

**Operation:**

The 12V power supply is connected to the input of the circuit. The IC7805 regulates the voltage to a stable 5V, which is then supplied to the microcontroller and other circuit components. The B688 BJT is used to amplify the current for the circuit, if necessary. The 100Ω / 5W resistor protects the B688 BJT from overcurrent. Electrolytic capacitors C1 and C2 filter out ripple noise at the input and output of the circuit. The diode D1 protects the circuit from reverse polarity.

A diagram of a circuit

Description automatically generated

Figure 2.3: 5-12V Power Supply

### **2. Important components:**

**a. IC7805**

In this project, we are using a PIC16F877A microcontroller which operates at 5V, therefore we choose the IC7805 to design the power supply circuit.

* **IC7805:** The IC7805 is a **voltage regulator IC**, a type of electronic component that maintains a constant output voltage even when the input voltage fluctuates. It is commonly used in electronic circuits to provide a stable power supply to other components.
* **78xx series:** The 78xx series is a family of **positive voltage regulators** manufactured by various companies, including Texas Instruments, On Semiconductor, and STMicroelectronics. The "xx" in the series name indicates the nominal output voltage of the regulator. For example, the IC7805 has an output voltage of 5V, while the IC7812 has an output voltage of 12V.
* **Input voltage range:** The input voltage range is the range of voltages that can be applied to the input pins of the IC without damaging it. For the IC7805, the input voltage range is 7V to 18V. This means that the input voltage must be between 7V and 18V for the IC to function properly.
* A diagram of a circuit board

  Description automatically generated**Output voltage:** The output voltage is the voltage that the IC provides to its output pin. For the IC7805, the output voltage is a stable 5V. This means that the voltage at the output pin will always be 5V, even if the input voltage fluctuates.

Figure 2.4: IC7805

A picture containing text, number, line, font

Description automatically generated

Figure 2.5: Datasheet IC7805

**b. B688**

In this project, we use IC7805 to generate a 5V output in the power supply circuit, but the output current of IC7805 is only 1A, which is not enough to meet the load, so we use BJT B688 to amplify the current.

There are many types of PNP switching BJTs on the market, and B688 is commonly used in power supply circuits.

A table with text on it

Description automatically generated

Figure 2.6: Parameter B686

**c. Capacitors**

Capacitors C9 and C10 are used to filter the voltage applied to the Vi pin of the IC 7805. Capacitor C9 is used to provide temporary voltage to the Vi pin when the power supply experiences a sudden drop in voltage. Capacitor C10 is a ceramic capacitor, and therefore has a high impedance. C10 helps to prevent the input voltage from rising suddenly, which can cause the input voltage waveform to become sawtooth shaped.

Capacitors C11 and C12 are used to filter the voltage supplied to the load from the Vo pin of the IC 7805. Capacitor C11 provides temporary voltage to the load when the load voltage drops suddenly. Capacitor C12 has a high impedance and helps to filter out output voltage noise (noise is an unwanted voltage that can cause the output voltage waveform to become distorted).

**III. Microcontroller Circuit**

1. **Schematic Diagram:**

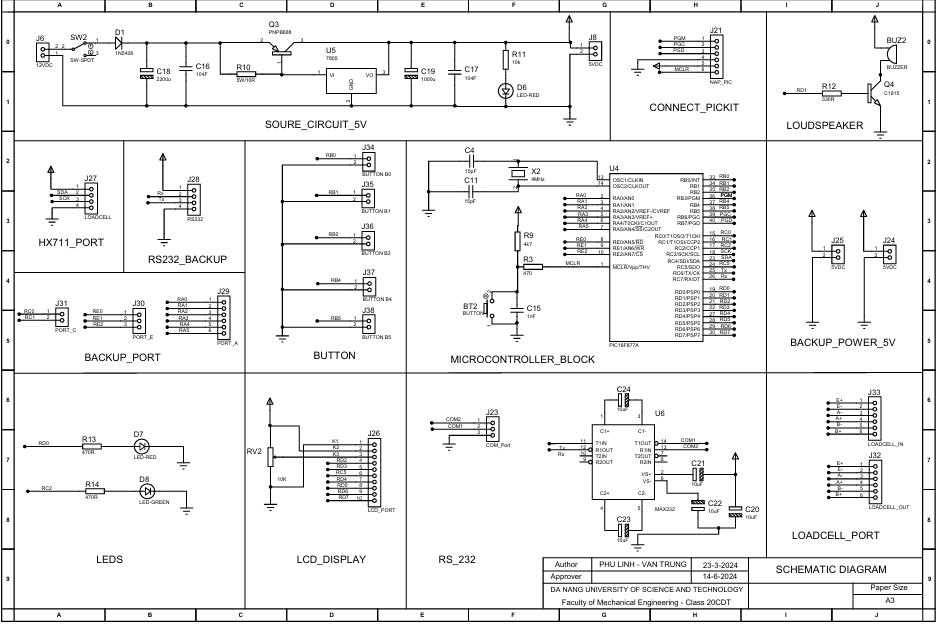


Figure 2.7: **Schematic Diagram**

A blue circuit board with many different colored lines

Description automatically generated

A computer chip with many different components

Description automatically generated

Figure 2.8: PCB Layout and 3D Modeling of Microcontroller Circuits in Proteus

A close-up of a circuit board

Description automatically generated

Figure 2.9: Actual microcontroller circuit

**a. PIC16F877A**

The PIC16F877A microcontroller is one of the PIC16F family of microcontrollers manufactured by Microchip. It has a 35-instruction set with a 14-bit instruction length. It is a very popular chip in Vietnam and around the world.

A black microchip with white text

Description automatically generated

**A circuit board with many letters and numbers

Description automatically generated**

## Figure 2.10: **PIC16F877A and** Pins diagram of the PIC 16F877A

## **Key Features of the PIC16F877A Microcontroller**

The PIC16F877A is a versatile and powerful microcontroller that is well-suited for a wide range of applications. Here are some of its key features:

* **8/16-bit CPU, built on a modified Harvard architecture:** This architecture provides efficient memory management and simplifies programming.
* **Flash and ROM memory options from 256 bytes to 256 Kbytes:** This flexibility allows you to choose the right amount of memory for your application.
* **I/O ports with normal logic levels from 0V to 5.5V, corresponding to logic 0 and logic 1:** This makes it easy to interface with a variety of external devices.
* **Synchronous serial peripheral communication standards USART, AUSART, EUSART:** These standards enable reliable communication with other devices.
* **Operates at a variety of clock frequencies, up to 20 MHz with a 200ns instruction cycle:** This allows you to adjust the performance of the microcontroller to meet your needs.
* **10/12-bit ADC converter:** This high-resolution converter allows you to accurately measure analog signals.
* **Voltage comparator:** This component can be used to compare two voltages for various applications.
* **Two CCP blocks: Capture/Compare/PWM:** These blocks provide flexible options for signal processing and control tasks.
* **Part of the PIC16F family with 35 instructions, each 14 bits long:** This family is known for its ease of use and affordability.
* **All instructions are single cycle, except for subroutine calls (two cycles):** This contributes to the microcontroller's efficiency.
* **8K × 14-bit Flash program memory, readable/writable up to 100,000 times:** This allows you to store and modify your program code as needed.
* **368 bytes of RAM:** This volatile memory provides temporary storage for data during program execution.
* **256 bytes of EEPROM memory, readable/writable up to 1,000,000 times and data retention for up to 40 years:** This non-volatile memory is ideal for storing configuration data.
* **Supply voltage: 4.0 to 5.5 VDC:** This wide range makes the microcontroller compatible with various power sources.
* **SLEEP mode for power saving:** This mode allows you to conserve power when the microcontroller is idle.
* **Five I/O ports with 33 pins (A, B, C, D, E):** These ports provide ample connectivity options for external devices.
* **Supports USB, CAN, LIN, IrDA:** This enables communication with various peripherals and networks.

**b. 4MHz Crystal Oscillator**

A crystal oscillator circuit is an electronic oscillator circuit that uses a quartz crystal as its frequency-determining element. The quartz crystal vibrates at a specific frequency when an electric voltage is applied to it. This vibration can be used to generate a stable and accurate clock signal for a microcontroller or other electronic device.

The 4MHz crystal oscillator in the PIC16F877A power supply circuit is used to generate a 4MHz clock signal for the microcontroller. The crystal is connected to two pins on the microcontroller (OSC1 and OSC2) through two capacitors (C1 and C2). The capacitors help to stabilize the oscillation of the crystal and provide a clean clock signal to the microcontroller.

Here is a more detailed explanation of how the 4MHz crystal oscillator works:

* The quartz crystal is connected to two pins on the microcontroller (OSC1 and OSC2).
* Two capacitors (C1 and C2) are connected in series with the crystal.
* An electric voltage is applied to the crystal.
* The crystal vibrates at a specific frequency.
* The vibration of the crystal generates an electric current.
* The capacitors filter the electric current and provide a clean clock signal to the microcontroller.A close-up of a transistor

  Description automatically generated

*A picture containing text, diagram, line, font

Description automatically generated***Figure 2.11:** Quartz 4Mhz

Figure 2.12: Quartz 4 MHz is connected to two pins OSC1 and OSC2

**c. Reset Circuit**

* The reset circuit is an important part of the PIC16F877A microcontroller circuit. It is responsible for resetting the microcontroller to its initial state when it is powered on or when a reset signal is applied. This ensures that the microcontroller starts up in a known and stable state.
* The reset circuit in the PIC16F877A microcontroller circuit is a low-active reset circuit. This means that the reset signal is active when it is low voltage. The reset signal is connected to the MCLR/VPP pin of the microcontroller.
* A diagram of a circuit

  Description automatically generatedThe reset circuit also includes a noise filter. The noise filter is used to prevent false reset signals from being generated by noise in the power supply or other parts of the circuit. The noise filter typically consists of a capacitor and a resistor. The capacitor filters out high-frequency noise, while the resistor limits the current that can flow through the circuit.

Figure 2.13: Recommended circuit

1. **Loadcell and Module HX711**
2. **Loadcell**

A load cell, also known as a force sensor, is a transducer that converts a force or weight into an electrical signal. Load cells are commonly used in electronic weighing scales, strain gauges, and other applications where it is necessary to measure force or weight.

Load cells work by measuring the strain that is induced in a metal element when a force is applied to it. The strain causes a change in the resistance of the metal element, which can be measured by an electrical circuit. The amount of change in resistance is proportional to the force that is being applied.

Load cells are available in a variety of sizes and capacities. The size and capacity of a load cell will depend on the application for which it is being used. For example, a load cell that is used to measure the weight of a person will be much larger and have a higher capacity than a load cell that is used to measure the weight of a small object.

Load cells are precision instruments that are typically made from high-quality materials. They are also very durable and can withstand a lot of abuse. As a result, load cells are a reliable and versatile tool for measuring force and weight.

## **Load Cell Specifications 10kg**

Figure 2.14: Specifications of loadcell 10kg

**Construction:**

A load cell is composed of two main components: strain gages and a load element.

* **Strain gages:** These are specialized resistors, typically the size of a fingernail, that change their resistance when compressed or stretched. They are powered by a stable voltage source and are bonded firmly to the load element.
* **Load element:** This is a resilient metal bar that bears the applied load.

**Working Principle:**

* A diagram of a strain gauge

  Description automatically generated with medium confidenceLoad cells operate based on the principle of the balanced Wheatstone bridge. The applied force value is proportional to the change in resistance of the strain gages within the bridge, resulting in a proportional voltage output (see Figure 2.13: Wheatstone Bridge).

Figure 2.15: Wheatstone bridge

* As shown in the diagram above, an excitation voltage is applied to the load cell inputs, R1 and R4, of the Wheatstone bridge, and the output voltage is measured between the other two corners, R2 and R3.
* When no load is applied, the output signal voltage is approximately zero. When a weight is placed on the scale, the load cell body is compressed, causing the resistance of the strain gages to change according to the length and cross-sectional area of the metal wire in the resistor.

1. **HX711 Loadcell ADC Conversion Module**

The HX711 24-bit ADC conversion module is designed to read the changing resistance values from a load cell sensor (which are typically very small and cannot be read directly by a microcontroller) with 24-bit ADC resolution and convert it to a 2-wire (Clock and Data) communication interface for sending data to the microcontroller. It is suitable for use with load cells in mass measurement applications.

**Key Features:**

* **High-precision ADC:** 24-bit ADC resolution for accurate load cell readings.
* **2-wire communication:** Simple and efficient communication interface using only Clock and Data lines.
* **Low power consumption:** Ideal for battery-powered applications.
* **Wide gain range:** Supports a wide range of load cell capacities.
* **Easy to use:** Simple setup and programming with minimal coding required.

**Working Principle:**

* The HX711 module utilizes a 24-bit analog-to-digital converter (ADC) to convert the analog voltage signal from the load cell into a digital value. The module also incorporates a gain amplifier to adjust the sensitivity of the ADC to match the specific load cell being used.
* Communication with the HX711 module is achieved through a 2-wire serial interface, consisting of a Clock (CLK) line and a Data (DAT) line. The microcontroller sends clock pulses to the module, and the module responds by sending data bits on the data line.
* The module's internal circuitry processes the load cell data and converts it into a 24-bit digital value, which is then transmitted to the microcontroller through the serial interface. The microcontroller can then interpret the digital value and calculate the corresponding weight or force measurement.

**Technical Specifications:**

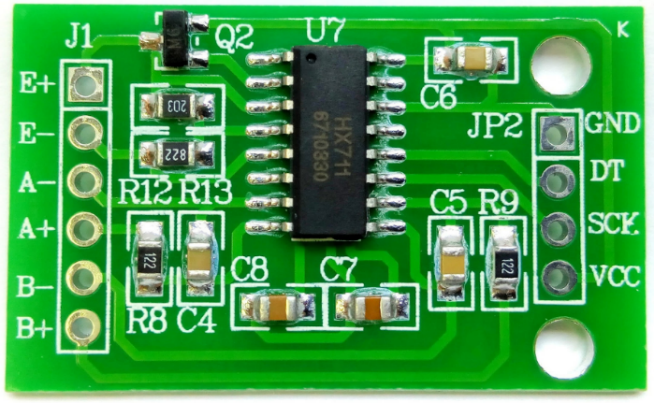
* **Operating Voltage:** 2.7 - 5V
* **Current Consumption:** < 1.5 mA
* **Sampling Rate:** 10 - 80 SPS (Configurable)
* **Resolution:** 24-bit ADC
* **Voltage Resolution:** 40mV
* **Dimensions:** 38 x 21 x 10 mm

Figure 2.16: Module HX711

### **UART-RS232 Communication:**

1. **UART**

UART stands for Universal Asynchronous Receiver Transmitter. It is a communication protocol for serial data transmission and reception. UART is commonly used for communication between microcontrollers, computers, and other devices.

**Key Characteristics of UART:**

* **Asynchronous:** Data is transmitted one bit at a time, without the need for a clock signal. Each bit is framed by start and stop bits.
* **Half-duplex:** Communication can occur in either direction, but not simultaneously. Devices take turns transmitting and receiving data.
* **Simple implementation:** UART requires minimal hardware and software overhead, making it easy to implement in microcontrollers.

**Typical UART Pins:**

* **TX (Transmit):** Sends data from the device to another device.
* **RX (Receive):** Receives data from another device.

**UART Voltage Levels:**

* **Microcontroller (UART):**
  + Logic "0": 0 V
  + Logic "1": 3.3 V to 5 V
* **Computer (RS232):**
  + Logic "0": +3.5 V to +25 V
  + Logic "1": -3.5 V to -25 V

1. **RS232**

RS232 (Recommended Standard 232) is a serial communication standard developed by the Electronic Industries Association (EIA). It defines the physical and electrical characteristics for data exchange between devices.

**Key Characteristics of RS232:**

* **Asynchronous:** Data is transmitted one bit at a time, without a clock signal. Each bit is framed by start and stop bits.
* **Half-duplex:** Communication can occur in either direction, but not simultaneously. Devices take turns transmitting and receiving data.
* **Point-to-point:** RS232 typically connects two devices directly.
* **Longer cable lengths:** RS232 supports cable lengths up to 15 meters (50 feet).

**Typical RS232 Pins:**

* **DCD (Data Carrier Detect):** Indicates whether the modem is ready to communicate.
* **RD (Received Data):** Receives data from the other device.
* **TD (Transmitted Data):** Sends data to the other device.
* **DTR (Data Terminal Ready):** Indicates that the device is ready to transmit data.
* **SG (Signal Ground):** Reference ground for the RS232 signals.
* **DSR (Data Set Ready):** Indicates that the modem is ready to receive data.
* **RTS (Request to Send):** Requests the other device to start transmitting data.
* **CTS (Clear to Send):** Indicates that the other device is ready to receive data.
* **RI (Ring Indicator):** Indicates that a call is incoming.

**UART-RS232 Conversion:**

Due to the difference in voltage levels between UART and RS232, a level shifter circuit is required to convert the signals for proper communication. The level shifter converts the low-voltage UART signals to the higher-voltage RS232 signals and vice versa.

A diagram of a circuit board

Description automatically generated

Figure 2.17: Connection RS232 and Max232

A close-up of a computer chip

Description automatically generated

Figure 2.18: RS232 and Max232 reality

**Function of UART-RS232 Pins:**

* **Pin 1 (DCD):** Not typically used for UART-RS232 communication.
* **Pin 2 (RD):** Connected to the UART RX pin.
* **Pin 3 (TD):** Connected to the UART TX pin.
* **Pin 4 (DTR):** Not typically used for UART-RS232 communication.
* **Pin 5 (SG):** Connected to the UART ground pin.
* **Pin 6 (DSR):** Not typically used for UART-RS232 communication.
* **Pin 7 (RTS):** Connected to the UART RTS pin.
* **Pin 8 (CTS):** Connected to the UART CTS pin.
* **Pin 9 (RI):** Not typically used for UART-RS232 communication.

**Advantages of UART-RS232:**

* **Simple and widely supported:** UART and RS232 are well-established standards with broad support in various devices and software.
* **Reliable and robust:** UART-RS232 communication is relatively resistant to noise and interference.
* **Long cable lengths:** RS232 supports cable lengths up to 15 meters (50 feet), making it suitable for longer-distance connections.

**Disadvantages of UART-RS232:**

* **Limited speed:** UART-RS232 typically supports data rates up to 115 kbps, which may be insufficient for high-speed data transfer applications.
* **Larger connectors:** RS232 connectors are larger and bulkier compared to newer interfaces like USB or Ethernet.

### **LCD Display (Liquid Crystal Display)**

LCD (Liquid Crystal Display) is an external device used for visual data display. The displayed data can be images or video characters.

**16x2 LCD Module**

In this topic, we use a special 16x2 LCD module to display characters according to ASCII encoding. Since this module can only display characters, it is often referred to as a Text LCD.

**Characteristics of the 1602 LCD Module:**

* Operates at 5 VDC voltage.
* Includes 32 cells distributed over 16 columns and 2 rows.
* Each cell displays one character according to ASCII encoding.
* Includes 16 pins with the order and corresponding functions as detailed in Figure 2.18.

## Technical Specifications of the 1602 LCD

* **Operating Voltage:** 5VDC
* **Display:** 16 lines, 2 characters
* **Dimensions:** 58.0 x 32.0 x 14.0 mm (2.28 x 1.26 x 0.55 inches)
* **Character Size:** 2.45 x 5.00 mm (0.096 x 0.197 inches)
* A picture containing text, screenshot, electronics, display

  Description automatically generated**Pixel Size:** 0.45 x 0.50 mm (0.018 x 0.020 inches)

Figure 2.19: LCD 1602A

A screenshot of a computer

Description automatically generated

Figure 2.20: Function of LCD pins

### **Camera and Barcode:**

1. **Camera and DroidCam Software:**

DroidCam is a free and open-source software that allows you to turn your Android smartphone into a webcam for your computer. It works by connecting the smartphone to the computer via Wi-Fi or USB and then providing a virtual webcam driver to the computer.

****

Figure 2.21: Droidcam

**Key Features of DroidCam:**

* **Free and open source:** DroidCam is completely free to use and its source code is available under the GPLv3 license.
* **Easy to use:** DroidCam is very easy to install and use. Simply install the DroidCam Server app on your smartphone and the DroidCam Client app on your computer, and then connect the two devices together via Wi-Fi or USB.
* **Multi-platform support:** DroidCam supports various operating system platforms, including Windows, macOS, Linux, and Android.
* **Multiple features:** DroidCam offers several useful features, such as the ability to adjust video resolution, audio quality, turn the flash on/off, etc.
* **Supports multiple connection types:** DroidCam can connect your smartphone to your computer via Wi-Fi or USB.

**Benefits of Using DroidCam:**

* **Cost-effective:** DroidCam saves you the cost of buying a webcam.
* **Portable:** You can use your smartphone as a webcam anywhere you have a Wi-Fi or USB connection.
* **Good image quality:** DroidCam provides good image quality, especially when using a USB connection.
* **Multiple features:** DroidCam offers many useful features that a regular webcam does not have.

**Drawbacks of Using DroidCam:**

* **Smartphone dependency:** DroidCam relies on the smartphone to function, so you need to make sure the phone is charged and has a stable internet connection.
* **Image quality may be affected:** Image quality can be affected by lighting, noise, and network speed.
* **Latency:** There may be a slight delay between the image displayed on the smartphone and the image displayed on the computer.

**In this project, we will use a smartphone camera in conjunction with DroidCam software and barcode/QR code to create an automated product recognition system.**

1. **Barcode:**

A barcode is a computer-readable representation of numbers and characters. It usually consists of parallel black and white bars of different widths that a barcode scanner can read.



Figure 2.20: Barcode

**Components of Barcodes:**

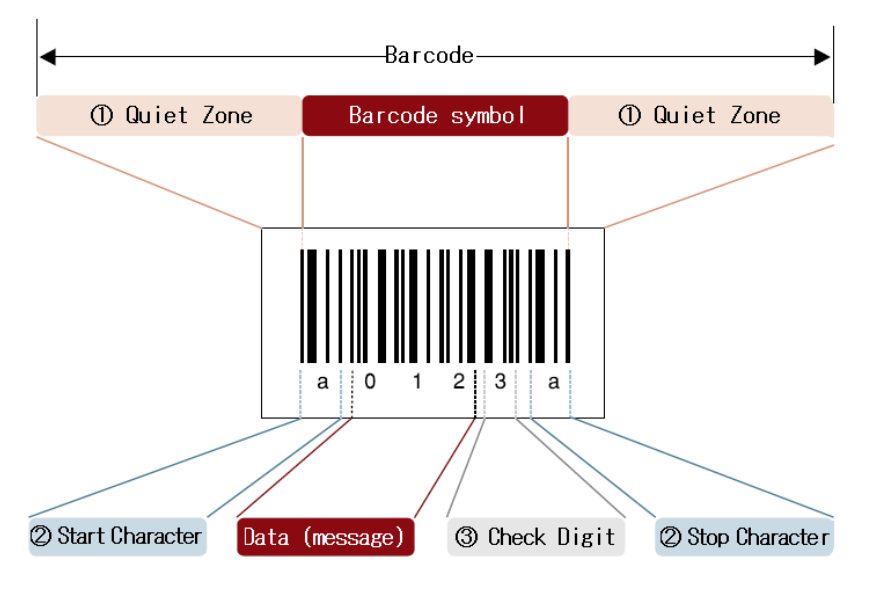


Figure 2.21: Components of Barcodes

### Quiet zone (margin)

A quiet zone is a blank margin located at either end of the barcode. If the quiet zone isn’t wide enough, it’ll be difficult for the scanner to read the barcode. Therefore, the minimum margin distance from the outermost bar of one symbology to the outermost bar of another symbol is 2.5 mm.

### Start/Stop character

Start/Stop characters are characters that represent the start/end of the data. These characters will vary slightly depending on the barcode type.

### Check digit (symbol check character)

Check digit is a digit to validate the barcode. Thanks to its symbol check character, you can check if the barcode data encoded in the barcode is correct.

**Working Principle of Barcodes**A barcode scanner and barcode scanner

Description automatically generated

Figure 2.22: Working Principle of Barcodes

* First, barcode scanners will use an incandescent light bulb or laser to shine light through the barcode and then shine it into a super-sensitive light detector.
* The black lines on the barcode absorb light, and the white parts of the barcode will shine through and be reflected.
* The scanner detects the amount of light that is then converted into computer-readable data. For example, the black parts that do not reflect well are recorded as 1 and the white parts recognized by the light scanner as 0.
* Inventory management software will receive and decode this information on the system.

All barcodes will have 12 digits, usually printed underneath as a precaution for possible complications. Here’s what the barcode numbers represent:

* First number: Product type
* 5 following numbers: Manufacturer Code
* 5 following numbers on the right: Product code
* Last number: Barcode number according to a self-policing system

# **CHAPTER 3: SOFTWARE DESIGN**

## Programming for the system

1. Algorithm flowchart

A diagram of a diagram

Description automatically generated

Figure 3.1 Program flowchart

1. Working principle

Initially, the weighing scale's display reads 0 when no object is present. Upon placing an object on the scale, the load cell detects the applied force, altering the strain gauge resistance. The sensor then transmits a voltage signal, which the HX711 amplifies and converts from analog to digital. This signal is subsequently processed by the microcontroller and displayed on the LCD screen.

1. Code CCS

#include <main.h>

#use rs232(baud=9600,parity=N,xmit=pin\_c6,rcv=pin\_c7,bits=8) //TX\_C6, RX\_C7

#define LCD\_ENABLE\_PIN PIN\_C5

#define LCD\_RS\_PIN PIN\_D2

#define LCD\_RW\_PIN PIN\_D3

#define LCD\_DATA4 PIN\_D4

#define LCD\_DATA5 PIN\_D5

#define LCD\_DATA6 PIN\_D6

#define LCD\_DATA7 PIN\_D7

#include <lcd.c>

#define DOUT PIN\_A0

#define PD\_SCK PIN\_A1

unsigned int32 ReadCount(){

unsigned int32 Count = 0;

unsigned int8 i, convert\_1, convert\_2, convert\_3;

output\_high(DOUT);

output\_low(PD\_SCK);

Count = 0;

while(input(DOUT));

for (i = 0 ; i < 24 ; i++){

output\_high(PD\_SCK);

Count = Count << 1;

output\_low(PD\_SCK);

if(input(DOUT)) Count++;

}

output\_high(PD\_SCK);

Count = Count|0x80;

output\_low(PD\_SCK);

convert\_1 = MAKE8(Count, 0);

convert\_2 = MAKE8(Count, 1);

convert\_3 = MAKE8(Count, 2);

convert\_2 = (convert\_2 & 0b11111000);

Count = MAKE16(convert\_3, convert\_2);

return(Count\*1.49);

}

unsigned int donvi, chuc, tram, nghin;

unsigned int16 KHOI\_LUONG, weigh;

void main(){

set\_tris\_d(0);

set\_tris\_c(0);

set\_tris\_a(0xFF);

lcd\_init();

lcd\_gotoxy(1,1);

printf(lcd\_putc, " Can kg ");

delay\_ms(1000);

weigh = ReadCount();

while(TRUE){

lcd\_gotoxy(1,2);

unsigned int32 KHOI\_LUONG = ReadCount() - weigh;

donvi = KHOI\_LUONG % 10;

chuc = (KHOI\_LUONG / 10) % 10;

tram = (KHOI\_LUONG / 100) % 10;

nghin = (KHOI\_LUONG / 1000) % 10;

printf(lcd\_putc, " Weigh: %d.%d%d%d kg" , nghin, tram, chuc, donvi );

printf("%d.%d%d%d\r", nghin, tram, chuc, donvi );

delay\_ms(1000);

}

}

# **CHAPTER 4 DESIGN THE SYSTEM'S SALES MANAGEMENT INTERFACE** **AND DATABASE**

## **Introduction to Visual Studio 2022 and SQL Server 2022 Software**

1. **Visual Studio 2022 Software**

- Visual Studio 2022 is a relatively new version of the Visual Studio software provided by Microsoft. This software can help in programming and creating user interfaces (software applications) in a relatively easy and convenient manner. In this project, the user interface for the electronic scale will be developed using Visual Studio software using the C# programming language.

A close-up of a logo

Description automatically generated

Figure 4.1 Logo Visual Studio

1. **Microsoft SQL Server 2022 Software**

- Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product that primarily functions to store and retrieve data as requested by other software applications. It can run on the same computer or on a different computer over a network (including the Internet).

A logo for microsoft server

Description automatically generated

Figure 4.2 Logo Sql Server

- In this project, we utilize a SQL database to store and retrieve product data, as well as store customer data. The database is running on a computer for efficient data management.

### **3. User Interface Software**

## **Software User Guide.**

* Scanning Interface:
* Utilizing a camera to scan QR codes to retrieve product information, calculate total weight, and display the corresponding price on the computer screen.
* Editing Interface:
* The system includes functions such as adding, deleting, searching, and editing product information within the system.
* Invoice Printing:
* Generating invoices with item details, purchase date, customer information, and pricing information.

**User Interface and Database Integration**

A close up of a cheeseburger

Description automatically generatedFigure 4.3 The home interface of the system

**A screenshot of a grocery store

Description automatically generated**

Figure 4.4 The main interface of the system for scanning QR codes

**A screenshot of a computer

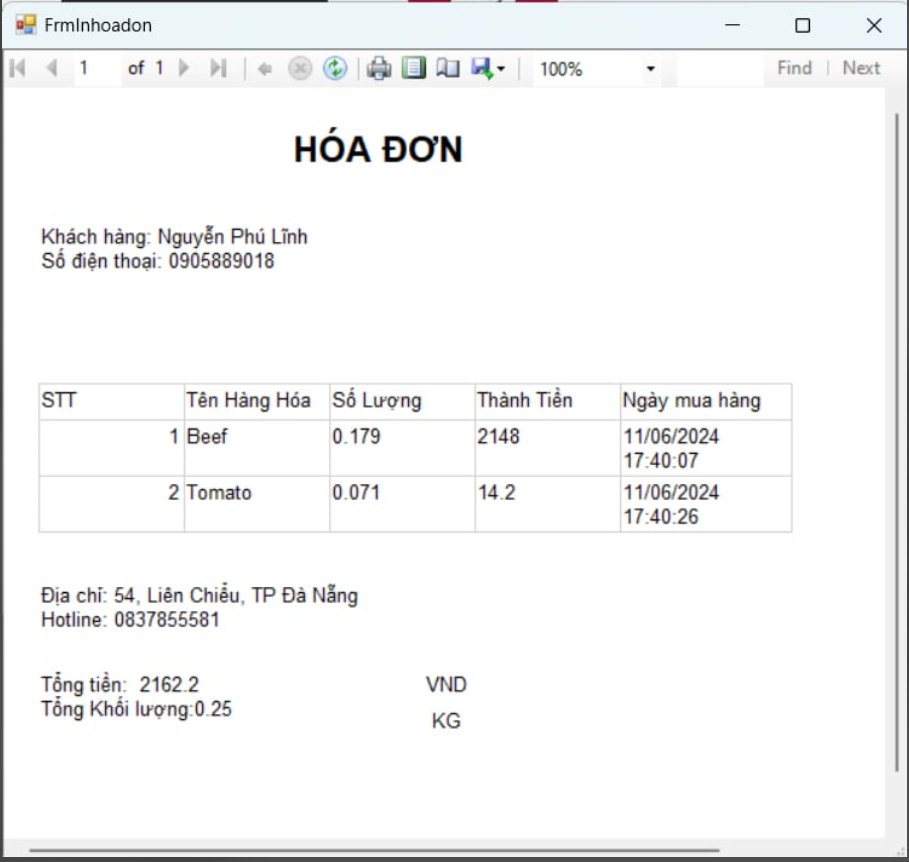
Description automatically generated**

Figure 4.5 The interface for adding, deleting, and editing on the system

A screenshot of a computer

Description automatically generated

Figure 4.6 The update interface

Figure 4.7 The interface for printing invoicesA screenshot of a computer

Description automatically generated

Figure 4.8 The system's product data

*A screenshot of a computer

Description automatically generated*

Figure 4.9 The sales data is stored in SQL Server

### **4. Code C#**

\* FormControl (Parent)

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace electronic\_scale

{

public partial class FormControl : Form

{

private Form activeForm;

public FormControl()

{

InitializeComponent();

SidePanel.Height = button1.Height;

SidePanel.Top = button1.Top;

}

private void OpenChildForm(Form childForm, object btnSender)

{

if (activeForm != null)

activeForm.Close();

activeForm = childForm;

childForm.TopLevel = false;

childForm.FormBorderStyle = FormBorderStyle.None; // loại bỏ đi các đường viền để cho form con giống với form cha

childForm.Dock = DockStyle.Fill; // thuộc tính dock fill lấp đầy toàn bộ không gian của form cha

this.panel\_Body.Controls.Add(childForm);

this.panel\_Body.Tag = childForm; // thêm form con vào trong panel 4

childForm.BringToFront(); // đẩy form con mới lên dể nó hiển thị phía trước

childForm.Show(); // hiển thị lên màn hình ở trong panel 4

}

private void button2\_Click(object sender, EventArgs e)

{

SidePanel.Height = button2.Height;

SidePanel.Top = button2.Top;

OpenChildForm(new SaleManager(), sender);

}

private void button1\_Click(object sender, EventArgs e)

{

SidePanel.Height = button1.Height;

SidePanel.Top = button1.Top;

OpenChildForm(new FormHome(), sender);

}

private void button3\_Click(object sender, EventArgs e)

{

SidePanel.Height = button3.Height;

SidePanel.Top = button3.Top;

OpenChildForm(new FormProduct(), sender);

}

private void button4\_Click(object sender, EventArgs e)

{

SidePanel.Height = button4.Height;

SidePanel.Top = button4.Top;

OpenChildForm(new FormUpdate(), sender);

}

private void button5\_Click(object sender, EventArgs e)

{

DialogResult result = MessageBox.Show("Do you want to close the application?", "Notification", MessageBoxButtons.OKCancel, MessageBoxIcon.Question);

if (result == DialogResult.OK)

{

Application.Exit();

}

}

}

}

\* FormAddProduct (Child)

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace electronic\_scale

{

public partial class FormProduct : Form

{

SqlConnection connection;

SqlCommand command;

string str = "Data Source=DESKTOP-O6OP19G\\WINCCPLUSMIG2014;Initial Catalog=saleManager;Integrated Security=True";

SqlDataAdapter adapter = new SqlDataAdapter();

DataTable table = new DataTable();

void loadData()

{

command = connection.CreateCommand();

command.CommandText = "select \* from product";

adapter.SelectCommand = command;

table.Clear();

adapter.Fill(table);

dataGridView1.DataSource = table;

}

public FormProduct()

{

InitializeComponent();

}

private void FormProduct\_Load(object sender, EventArgs e)

{

connection = new SqlConnection(str);

connection.Open();

loadData();

}

private void btnAdd\_Click(object sender, EventArgs e)

{

try

{

command = connection.CreateCommand();

command.CommandText = "insert into product values('"

+ tbBarCode.Text + "', '"

+ tbName.Text + "', '"

+ tbUnit.Text + "', '"

+ tbPrice.Text + "') ";

command.ExecuteNonQuery();

loadData();

}

catch

{

MessageBox.Show("Can't add this Product");

}

}

private void btnDelete\_Click(object sender, EventArgs e)

{

DialogResult question = MessageBox.Show("Are you sure want to delete this product?", "Delete Product", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

// Ctrl + K + D

if (question == DialogResult.Yes)

{

command = connection.CreateCommand();

command.CommandText = "delete from product where BarCode='" + tbBarCode.Text + "'";

command.ExecuteNonQuery();

loadData();

}

}

private void btnUpdate\_Click(object sender, EventArgs e)

{

try

{

command = connection.CreateCommand();

command.CommandText = "update product set Name = '"

+ tbName.Text + "', Unit = '"

+ tbUnit.Text + "', Price = '"

+ tbPrice.Text + "' where BarCode = '"

+ tbBarCode.Text + "'";

command.ExecuteNonQuery();

loadData();

}

catch

{

MessageBox.Show("Can't update this Product");

}

}

private void btnFind\_Click(object sender, EventArgs e)

{

command = connection.CreateCommand();

command.CommandText = "SELECT \* FROM product WHERE Name LIKE '%' + @Name + '%'";

command.Parameters.AddWithValue("@Name", tbName.Text);

adapter.SelectCommand = command;

table.Clear();

adapter.Fill(table);

dataGridView1.DataSource = table;

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

int i;

i = dataGridView1.CurrentRow.Index;

tbBarCode.Text = dataGridView1.Rows[i].Cells[0].Value.ToString();

tbName.Text = dataGridView1.Rows[i].Cells[1].Value.ToString();

tbUnit.Text = dataGridView1.Rows[i].Cells[2].Value.ToString();

tbPrice.Text = dataGridView1.Rows[i].Cells[3].Value.ToString();

}

}

}

\* SaleManager (Child)

using AForge.Video.DirectShow;

using Microsoft.Reporting.WinForms;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using ZXing;

namespace electronic\_scale

{

public partial class SaleManager : Form

{

public double Tongtienhang { get; set; }

public double Tongkhoiluong { get; set; }

public string TenKhachHang { get; set; }

public string UID\_Khachhang { get; set; }

public SaleManager()

{

InitializeComponent();

btnConnect.Enabled = true; // an nut ngat ket noi

btnDisconnect.Enabled = false;

//txtQR.Enabled = false;

}

private FilterInfoCollection filterInfoCollection; // tạo biến thu thập có bao nhiêu camera kết nối với máy tính

private VideoCaptureDevice captureDevice; // biến chọn camera để ghi hình để sử dụng

private void SaleManager\_Load(object sender, EventArgs e)

{

filterInfoCollection = new FilterInfoCollection(FilterCategory.VideoInputDevice); // thao tác với camera

foreach (FilterInfo filterInfo in filterInfoCollection) cboDevice.Items.Add(filterInfo.Name); // đẩy danh sách các camera vào combox

cboDevice.SelectedIndex = 0;

khoi\_tao\_bang();

}

private void btnConnect\_Click(object sender, EventArgs e)

{

try

{

serialPort1.PortName = "COM6";

serialPort1.BaudRate = Convert.ToInt32(9600);

serialPort1.Open();

btnConnect.Enabled = false;

btnDisconnect.Enabled = true;

//txtQR.Enabled = true;

// ket noi va bat camera lên

captureDevice = new VideoCaptureDevice(filterInfoCollection[cboDevice.SelectedIndex].MonikerString);

captureDevice.NewFrame += CaptureDevice\_NewFrame;

captureDevice.Start();

timer1.Start();

timer2.Start();

timer3.Start();

}

catch

{

MessageBox.Show("Connect COM Port");

}

}

private void CaptureDevice\_NewFrame(object sender, AForge.Video.NewFrameEventArgs eventArgs)

{

//throw new NotImplementedException();

pictureBox1.Image = (Bitmap)eventArgs.Frame.Clone();

}

private void timer1\_Tick(object sender, EventArgs e)

{

if (pictureBox1.Image != null)

{

BarcodeReader barcodeReader = new BarcodeReader();

Result result = barcodeReader.Decode((Bitmap)pictureBox1.Image);

if (result != null)

{

txtQR.Text = result.ToString();

timer1.Stop();

serialPort1.Write("1"); // gui tin hieu xuong VDK de phat coi keu

if (captureDevice.IsRunning)

captureDevice.Stop();

//captureDevice.SignalToStop();

//captureDevice.WaitForStop();

}

}

}

private void btnDisconnect\_Click(object sender, EventArgs e)

{

serialPort1.Close();

btnConnect.Enabled = true; // an nut ngat ket noi

btnDisconnect.Enabled = false;

txtQR.Text = "";

timer2.Stop();

timer3.Stop();

if (captureDevice.IsRunning) // Neu may anh runing thi ta off capture\_device

{

captureDevice.Stop();

//captureDevice.SignalToStop();

//captureDevice.WaitForStop();

}

if (pictureBox1.Image != null) // Giai phong hinh anh da scaner QR de tiep tuc scan cac lan sau

{

pictureBox1.Image.Dispose();

pictureBox1.Image = null;

}

}

string strCon = "Data Source=DESKTOP-O6OP19G\\WINCCPLUSMIG2014;Initial Catalog=saleManager;Integrated Security=True";

SqlConnection sqlCon = null;

double sum = 0;

string donvi\_tam;

int tam1 = 0;

private void txtQR\_TextChanged(object sender, EventArgs e)

{

if (txtQR.Text != "")

{

if (sqlCon == null) // rong thi tao va dong thi mo

{

sqlCon = new SqlConnection(strCon);

}

if (sqlCon.State == ConnectionState.Closed)

{

sqlCon.Open();

}

string maBarcode = txtQR.Text.Trim(); // doi tuong thuc thi tri van lay csdl

SqlCommand sqlCmd = new SqlCommand();

sqlCmd.CommandType = CommandType.Text;

using (SqlCommand cmd = new SqlCommand("select \* from product where Barcode ='" + maBarcode + "'", sqlCon)) // gui truy van vao ket noi

{

using (SqlDataReader reader = cmd.ExecuteReader()) // thuc thidoc du lieu

{

while (reader.Read())

{

string tenhanghoa = reader.GetString(1);

string donvitinh = reader.GetString(2);

donvi\_tam = donvitinh;

string giaThanh = reader.GetString(3);

txtTenhanghoa.Text = tenhanghoa;

txtGiathanh.Text = giaThanh;

int a = Convert.ToInt32(reader.GetString(3)); // bien gia thanh

double b = Convert.ToDouble(txtKhoiluong.Text); // bien khoi luong

Tongtienhang = Tongtienhang + a \* b;

Tongkhoiluong = Tongkhoiluong + b;

sum = a \* b;

txtThanhtien.Text = Convert.ToString(sum);

txtTongkhoiluong.Text = Convert.ToString(Tongkhoiluong);

txtTongtienhang.Text = Convert.ToString(Tongtienhang);

//txtTonghoadon.Text = Convert.ToString(Tongtienhang);

Add\_dulieu();

}

reader.Close();

}

}

if (dataGridView1.Rows.Count - 1 > 0 && tam1 == 1)

{

using (SqlCommand sqlCmd2 = new SqlCommand("Insert into Sale(Macode,TenSanPham,Soluong,Giaban,Thanhtien,Ngaymuahang) values(@code,@tensanpham,@soluong,@giaban,@thanhtien,@ngaymuahang)", sqlCon))

{

sqlCmd2.CommandType = CommandType.Text;

sqlCmd2.Parameters.AddWithValue("@code", txtQR.Text);

sqlCmd2.Parameters.AddWithValue("@tensanpham", txtTenhanghoa.Text);

sqlCmd2.Parameters.AddWithValue("@soluong", txtKhoiluong.Text);

sqlCmd2.Parameters.AddWithValue("@giaban", txtGiathanh.Text);

sqlCmd2.Parameters.AddWithValue("@thanhtien", txtThanhtien.Text);

sqlCmd2.Parameters.AddWithValue("@ngaymuahang", lblClock.Text);

sqlCmd2.ExecuteNonQuery();

tam1 = 0;

}

}

}

}

private void timer2\_Tick(object sender, EventArgs e)

{

string tam = serialPort1.ReadExisting();

if (tam != "")

{

txtKhoiluong.Text = tam;

}

}

private void khoi\_tao\_bang()

{

dataGridView1.AutoGenerateColumns = false;

const int NumberColumn = 7;

dataGridView1.ColumnCount = NumberColumn;

string[] list = new string[NumberColumn] { "Barcode", "Tenhanghoa", "Donvitinh", "Soluong", "Giaban", "Thanhtien", "Ngaygiomua" };

string[] header\_name = new string[NumberColumn] { "BarCode", "Name", "Unit", "Quantity", "Price", "Cost", "DateTime" };

int[] width\_col = new int[NumberColumn] { 150, 120, 120, 120, 120, 120, 150 };

for (int i = 0; i < NumberColumn; i++)

{

dataGridView1.Columns[i].Name = list[i];

dataGridView1.Columns[i].DataPropertyName = list[i];

dataGridView1.Columns[i].HeaderText = header\_name[i];

dataGridView1.Columns[i].Width = width\_col[i];

dataGridView1.Columns[i].ValueType = typeof(String);

dataGridView1.Columns[i].Visible = true;

}

}

private void timer3\_Tick(object sender, EventArgs e)

{

DateTime currentDateTime = DateTime.Now;

lblClock.Text = currentDateTime.ToString("dd/MM/yyyy HH:mm:ss");

}

public void Add\_dulieu()

{

DataGridViewRow row = new DataGridViewRow();

row.CreateCells(dataGridView1);

row.Cells[0].Value = txtQR.Text;

row.Cells[1].Value = txtTenhanghoa.Text;

row.Cells[2].Value = donvi\_tam;

row.Cells[3].Value = txtKhoiluong.Text;

row.Cells[4].Value = txtGiathanh.Text;

row.Cells[5].Value = txtThanhtien.Text;

row.Cells[6].Value = lblClock.Text;

dataGridView1.Rows.Add(row);

tam1++;

}

private void btnXoa\_Click(object sender, EventArgs e)

{

sum = 0;

Tongkhoiluong = 0;

Tongtienhang = 0;

tam1 = 0;

txtTenKH.Text = "";

txtMaKH.Text = "";

txtQR.Text = "";

txtTenhanghoa.Text = "";

txtKhoiluong.Text = "";

txtGiathanh.Text = "";

txtThanhtien.Text = "";

txtTongkhoiluong.Text = "";

txtTongtienhang.Text = "";

dataGridView1.Rows.Clear();

}

ReportDataSource rs = new ReportDataSource();

private void btnInhoadon\_Click(object sender, EventArgs e)

{

if (txtMaKH.Text.Length > 0 && txtTenKH.Text.Length > 0)

{

TenKhachHang = txtTenKH.Text;

UID\_Khachhang = txtMaKH.Text;

List<Dulieu1> list = new List<Dulieu1>();

list.Clear();

for (int i = 0; i < dataGridView1.Rows.Count - 1; i++)

{

Dulieu1 dulieu1 = new Dulieu1

{

Tenhanghoa = dataGridView1.Rows[i].Cells[1].Value.ToString(),

Soluong = dataGridView1.Rows[i].Cells[3].Value.ToString(),

Thanhtien = dataGridView1.Rows[i].Cells[5].Value.ToString(),

Thoigianmua = dataGridView1.Rows[i].Cells[6].Value.ToString()

};

list.Add(dulieu1);

}

rs.Name = "DataSet1";

rs.Value = list;

FrmInhoadon frm2 = new FrmInhoadon(this);

frm2.reportViewer1.LocalReport.DataSources.Clear();

frm2.reportViewer1.LocalReport.DataSources.Add(rs);

frm2.reportViewer1.LocalReport.ReportEmbeddedResource = "electronic\_scale.Report1.rdlc";

frm2.ShowDialog();

}

else

{

MessageBox.Show("Vui long nhap ten KH", "Notify", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

}

\* FormInhoadon

using Microsoft.Reporting.WinForms;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace electronic\_scale

{

public partial class FrmInhoadon : Form

{

private SaleManager saleManager;

//private FormSale formSale;

public FrmInhoadon(SaleManager saleManager)

{

InitializeComponent();

this.saleManager = saleManager;

//this.formSale = new FormSale();

}

private void FrmInhoadon\_Load\_1(object sender, EventArgs e)

{

double tongtienhang = this.saleManager.Tongtienhang;

double tongkhoiluong = this.saleManager.Tongkhoiluong;

string tenkhachhang = this.saleManager.TenKhachHang;

string sodienthoai = this.saleManager.UID\_Khachhang;

//double tongtienhang = this.formSale.Tongtienhang;

//double tongkhoiluong = this.formSale.Tongkhoiluong;

//string tenkhachhang = this.formSale.TenKhachHang;

//string sodienthoai = this.formSale.UID\_Khachhang;

ReportParameter Tongtienhang = new ReportParameter("Tongtienhang", tongtienhang.ToString());

// tao cac paramerter hien thi dulieu vao bao cao report

this.reportViewer1.LocalReport.SetParameters(new ReportParameter[] { Tongtienhang });

ReportParameter Tongkhoiluong = new ReportParameter("Tongkhoiluong", tongkhoiluong.ToString());

this.reportViewer1.LocalReport.SetParameters(new ReportParameter[] { Tongkhoiluong });

ReportParameter TenKhachHang = new ReportParameter("TenKhachHang", tenkhachhang);

this.reportViewer1.LocalReport.SetParameters(new ReportParameter[] { TenKhachHang });

ReportParameter UID\_Khachhang = new ReportParameter("UID\_Khachhang", sodienthoai);

this.reportViewer1.LocalReport.SetParameters(new ReportParameter[] { UID\_Khachhang });

this.reportViewer1.RefreshReport();

}

}

}

\*Dulieu1.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace electronic\_scale

{

public class Dulieu1

{

public string Tenhanghoa { get; set; }

public string Soluong { get; set; }

public string Thanhtien { get; set; }

public string Thoigianmua { get; set; }

}

}

# **CHAPTER 5 LESSON LEARNED AND CONCLUSION**

## **Overall Evaluation**

## The Electronic Scale and Sales Management System is a product our team has designed, which is not just a prototype but also intended for practical, everyday use. While the market already offers various integrated barcode scanning scales, they tend to be quite costly. To address this, our team has focused on leveraging readily available components to develop a more affordable alternative. We are committed to continuously refining our product to enhance its functionality and affordability.

## Throughout the development process, we faced several challenges, particularly with printing and circuit fabrication. However, the etching process proved to be relatively straightforward, and the tasks of drilling holes and soldering components were manageable.

## **Advantages and disadvantages of the product**

a) **Advantages**

* Quick QR code scanning utilizing the built-in camera of the mobile Phone
* User-friendly interface
* Integration of multiple functions
* Simple installation, compact model
* Easy to use

b) **Disadvantages**

* Lower quality
* Lower accuracy
* Interface requires improvement
* Cannot weigh objects with too small mass
* Only works with Kg units
* There is still a lot of unused hardware on the circuit: Speaker, indicator lights, 5 function buttons

## **Future Development Directions**

* Strive to improve the accuracy of the electronic scale.
* Optimize the program algorithm.
* Optimize the design of the model.
* Improve the quality of the product by using sensors with higher accuracy.
* Create additional functions for hardware

# **REFERENCE**

[1] Document “Giáo trình vi điều khiển PIC” – Đặng Phước Vinh & Võ Như Thành, published in 2021.

[2] Document “Giáo trình kĩ thuật vi điều khiển – Đặng Phước Vinh, published in 2019

[3] Datasheet HX711 - Access link:https://pdf1.alldatasheet.com/datasheetpdf/view/1132222

/AVIA/HX711.html . Access on 23/3/2023.

[4] Datasheet PIC16F877A – Access link: <https://pdf1.alldatasheet.com/datasheet-pdf/view/82338/MICROCHIP/PIC16F877A.html> . Access on 23/3/2023