الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Département : Génie électrique Et Informatique Industrielle

End-of-study dissertation with a view to obtaining the diploma of state engineer in electronics

Speciality

Embedded systems

- Theme –

Design and construction of an electronic system for a Thruster test bench

Realized by

Abdenour ABDELLAOUI , Moussa BAKHOUCH

Jury members:

M. REZOUG Amar **M.C.A**	President
M. ZAOUI Abdelhalim **Prof**	Supervisor
M. BENSIKADDOUR ELHABIB	Supervisor
M. BOUDIAR Toufik **M.C.A**	Examiner
Mme BELHACHAT Faiza **M.C.A**	Examiner

College year 2021 –2022

Dedications 1

I dedicate this work,

To my dear parents, to my mother for her love and encouragement,

To my father for his support and his trust in me.

To all the members of my family who have supported me throughout my university career, and to all my friends.

Thank you for always being there for me may this report be the best gift I can give you.

Abdenour ABDELLAOUI

Dedications 2

I dedicate this work,

To my dear parents, to my mother for her love and encouragement,

To my father for his support and his trust in me.

To all the members of my family who have supported me throughout my university career, and to all my friends.

Thank you for always being there for me may this report be the best gift I can give you.

Moussa BAKHOUCH

Acknowledgments

First of all, we thank God who gave us this property and has the strength and patience to complete this work.

We would like to express our thanks to our supervisors Mr. Bensikaddour and Mr. Zaoui who directed this work.

We would like to thank the president of the jury, as well as the members of the jury for agreeing to judge this work.

This work that we have done owes a lot to certain people whom we would like to sincerely thank. Finally, we would like to thank all the people who have contributed from near or far, near or far, to the realization of this work.

Satellite Development Center (CDS) Special acknowledgments

The Satellite Development Center (CDS), inaugurated on February 23, 2012, the CDS is located in the municipality of Bir El Djir, in the wilaya of Oran, and is made up of modern infrastructures (workshops and laboratories) dedicated to the design and assembly of satellites as well as environmental testing facilities. The Satellite Development Center is responsible for the design, development, and construction of the space systems provided for under the national space program.

Our end-of-studies project was held inside the CDS during our internship. And that is thanks to my internship manager, Mr. Si Mohammed AREZKI, head of the RMS (Research department in Space Mechanics) department within the CDS for having received me at the most inconvenient time.

I would like to thank the Director of the satellite development center Mr. Mohammed KAMECHE for welcoming me to his center.

I am very grateful to Mr. Elhabib Bensikaddour that who not only helped me learn several concepts but also devoted his limited time to guide and support me through my stay at CDS.

I thank the team in charge at CDS for their patience and their warm help.

Table of Contents

List of figures and tables. 1
Abbreviations list 5
General Introduction 6
Chapter I Measurement acquisition chain 7
Introduction 7
I.1 Role and constitution of the Acquisition chain 7
I.2 Elements of a data acquisition system 9
I.2.1 Sensors 9
I.2.2 Analogue to digital converters 14
I.2.3 Amplifiers 18
I.2.4 Multiplexer 21
I.2.5 Data Acquisition Software 24
I.3 Conclusion 24
Chapter II Design and architecture of the electronic board 25
Introduction 25
II.1 Development plan 25
II.1.1 Product requirements 25
II.1.2 System architecture 27
II.1.3 Hardware design 39
II.1.4 Software design 62
II.2 Conclusion 67
Chapter III Results and testing 68
Introduction 68
III.1 PCB design 68
III.2 The final form of the data acquisition board 70
III.3 The scale and the temperature sensors 72
III.3.1 Load cell Sensor 72
III.3.2 Thermocouple sensor 74
III.4 Test and result 75
III.4.1 Firing test and result 75
III.4.2 Weight scale sensor test and result 76
III.4.3 Thermocouple sensor test and result 78
III.5 Conclusion 79
General conclusion 80
Annexes 81
Bibliography list. 82

List of figures and tables.

Figure I-1: Position of the data acquisition chain in a process measurement-control assembly…...……7

Figure I-2: Exemple of an acquisition Chain structure…...……8

Figure I-3: Errors in sensors: (a) gain error, (b) offset, (c) terminal linearity error, (d) independent linearity error, (e) zero-based linearity error…..……10

Figure I-4: Wheatstone bridge representations: a) rectangular, b) rhomboidal, c) square…..……..13

Figure I-5: Wheatstone bridge…..…….13

Figure I-6: Quantification errors and thresholds…..…….15

Figure I-7: Block diagram of ΔΣ ADC (Baker, 2016)……..16

Figure I-8: Input signal to the ΔΣ modulator…...……...…..17

Figure I-9: First-order, low-pass averaging filter…...……...….17

Figure I-10: Differential amplifier. Schematic diagram…..……..18

Figure I-11: Instrumentation amplifier……..20

Figure I-12: Multiplex a) unipolar b) differential…….21

Figure I-13: Chain structure and associated multiplexer……...……...23

Figure II-1: Design process of an embedded system……25

Figure II-2: Different propositions for the data acquisition chain……27

Figure II-3:Functional Block Diagram AD7794/AD7795…..……...……....28

Figure II-4: Functional Block Diagram AD7191…...……...……29

Figure II-5: K-Type Thermocouple Measurement Using Integrated Temperature Sensor for Cold-Junction Compensation…...…….31

Figure II-6: Simplified Block Diagrams of ADS1115…...……...32

Figure II-7: Simplified Block Diagrams of CS5509…… ……..33

Figure II-8: Typical digital control system. …..…….. 35

Figure II-9: PIC18F452 …..……..39

Figure II-10: KidCad logo…….39

Figure II-11: Different blocks circuit of the electronic board ……40

Figure II-12: Different blocks circuit of the electronic board on KiCad software……..……40

Figure II-13: 5 Volte Power supply block circuit…….41

Figure II-14: 9V battery……41

Figure II-15: LM7805 – Pin Diagram…….....42

Figure II-16:PIC18f452 microcontroller pins and all it connection……..42

Figure II-17: Minipro programmer for all PIC microcontrollers…….....43

Figure II 18: Loading a Bootloader into a MCU 43

Figure II-19: Microcontroller in Circuit Serial Programming (ICSP) with pickit 3……44

Figure II-20: ICSP connection with pic microcontroller. …...……45

Figure II-21: Buzzer passive 5v……45

Figure II-22: Rotary Switch keylock……46

Figure II-23: Simple firing circuit…….47

Figure II-24: Firing circuit using transistor……...47

Figure II-25: Firing circuit using transistor and a relay……48

Figure II-26: Firing circuit using a transistor, relay, and optocoupler for testing……48

Figure II-27: Final firing circuit……49

Figure II 28: Display circuit …….49

Figure II-29: Pin Configuration and Typical Application Circuit of the MAX7219……50

Figure II-30: SPI master-slave relationship……..50

Figure II-31: Data acquisition circuit ……..51

Figure II-32: 74HC4067 Pinout (instrument)……...52

Figure II-33: 74HC4067 module……..52

Figure II-34: AD620 circuit connection diagram …….53

Figure II-35: Instrumentation Amplifier Diamond Plot Tool……...54

Figure II-36: Pin Configurations for the ICL7660S……..55

Figure II-37: Simple negative converter and its……55

Figure II-38: Pin Configuration for the ADS1118……55

Figure II-39: Input Multiplexer .....……..57

Figure II-40: DOUT/DRDY Behavior without Data Retrieval in Continuous Conversion Mode……..……..58

Figure II-41: SD card and RS232 pin connections……...58

Figure II-42: Mirco SD Card Interface Module……60

Figure II-43: UART interfacing with PIC circuit ……61

Figure II-44: DB9 Connector pinout ……..61

Figure II-45: TTL to RS232 convertor module…….62

Figure II-46: Pycharm and python ……...62

Figure II-47: Serial communication root……..64

Figure II-48: Extended root GUI …….65

Figure II-49: Graph popup root ……65

Figure II-50: The organization chart of the program produced in PIC18F45273…….66

Figure III-1: Final PCB design……..…….68

Figure III-2: 3D module of the front electronic board ……..……69

Figure III-3: The 3D module of the back electronic board ……..…….70

Figure III-4: Data acquisition board and it different blocks……71

Figure III-5: 10KG Load cell Scale Sensor……...…72

Figure III-6: Graph of weights scale sensor with rated output voltage……..…73

Figure III-7: Weight scale sensor…….73

Figure III-8: K Type Thermocouple…….74

Figure III-9: Graph of Temperature measured in °C with the output voltage in mV of Type K Thermocouple……...74

Figure III-10: Cheap igniter using match head with thin copper strands…….....75

Figure III-11: Data acquisition using an electric board with a load of 1.5 Kg…….76

Figure III-12: Screenshot of the pc during the test with the 1.5Kg of weight…….77

Figure III-13: Data acquisition using an electric board with a load of 5.5 Kg…….77

Figure III-14: Screenshot of the pc during the test with the 5.5Kg of weight…….……...…….78

Figure III-15: Data acquisition using an electric board with a K type temperature sensor…….…….78

Figure III-16: Screenshot of the pc during the test with the K-Type temperature sensor.…….……...79

Table I-1: Common thermocouple materials ……..……..…..……12

Table I-2: Value of the quantum q according to the number n of bits (Vpe: 10V)……..…….14

Table II-1: ADC specification comparison table……..……..……..…34

Table II-2: MCU under consideration ……..38

Table II-3: Pin Functions of ADS1118……..…56

Table II-4: Pin Configuration of Micro SD…….……59

Abbreviations list

ADC (Analog-Digital Converter)

CPU (Central Processing Unit)

DAS (Data Acquisition Systems)

EDA (Electronics Design Automation)

EMF (Electromotive Force)

IC (Integrated Circuits)

ICSP (In Circuit Serial Programming)

IOT (Internet of Things)

LSB (Less Significant Bit)

MCU (Microcontroller Unit)

MR (Measuring Range)

PC (Personal Computer)

PCB (Printed Circuit Board)

PGA (Programmable Gain Amplifier)

PLD (Programmable Logic Devises)

RAM (Random-Access Memory)

ROM (Read-Only Memory)

SMD (Surface Mounting Technology)

SPI (Serial Peripheral Interface)

SPS (Samples Per Second)

TCL (Tool Command Language)

TTL (Transistor-Transistor Logic)

TUI (Text-based User Interface)

USB (Universal Serial Bus)

VDU (Video Display Unit)

General Introduction

Globalization today increasingly pushes companies and various industry organizations to engage in a fiercely competitive battle.

On this basis, there is a rapid development of technology in all fields, especially in the fields of measurement.

One of the fields of use of measurement is the field of the test bench for solid rocket engines. and this test can only be carried out by integrating two fields, the mechanical field, and the electronic field, the latter requires us to study and develop electronic acquisition board, which can gain strength through the force sensor and the temperature sensor.

Thus, we aim in this research to obtain an acquisition board capable of acquiring forces with multiple force sensors and temperature with multiple temperature sensors. and also, being cable of lighting the igniter inside the rocket engine (firing).

This board will be able to function in 2 modes. The local mode and the remote mode. In the local mode, the firing will be held mainly on the board, and the acquisition data will be saved on the SD board. on the other hand, in the remote mode, a graphic user interface is going to handle the firing and the data acquisition.

In the first chapter, we introduced the data acquisition system, as well as its configuration structure, which consisted of sensors and their different types, static performance, for each force and temperature sensor. analogue to digital converter and their general characteristics. Differential amplifiers. Multiplexer and a computer to display the data acquired.

While we were devoting the second chapter to the design and production of the electronic board, by using a structural method in embedded systems design that consists of mutable steps from product requirements when we define the main requirement of our electronic board which mainly is the local and the remote mode, System architecture we choose the best suitable proposition for the board design, Hardware design by designing the electronic circuits and Software design by building the necessary software.

Lastly, the third chapter contains the development and the test of the board, after careful testing of components in the architecture it is time to design the final PCB board. after that, we print it and then solder all components on the board. Then comes the testing part when we integrate the software and hardware part together testing everything from force sensors and temperature sensors to the firing system. And then visualizing the result and data being acquired.

Chapter I Measurement acquisition chain

Introduction

Data Acquisition Systems (DAS) are the main instruments used in laboratory research from scientists and engineers; in particular, for test and measurement, automation and so on. [1] Data acquisition systems are systems designed to measure a physical phenomenon and convert those measurements into a form that can be viewed and manipulated on a computer. These systems have become essential for measuring and analyzing almost all forms of physical systems, such as Pressure, Temperature, Flow, etc. The objective of this chapter is to present the architecture of data acquisition systems, as well as the different proposition for the individual blocks that make up the DAS.

I.1 Role and constitution of the Acquisition chain

An acquisition chain collects the information needed to know and control a process; it delivers this information in a form suitable for use (Figure I-1).

An industrial process brings together various elements whose physical or (and) chemical interactions contribute to a manufacturing or transformation goal; the process is instrumented and controlled for this purpose. The state of a process is, at any time, characterized by the values of a certain number of variable physical or (and) chemical quantities: these values are the information that the acquisition system must provide. The assignment of a value to a physical or chemical quantity is a measurement operation, the object quantity of the measurement being called the measured. [1]

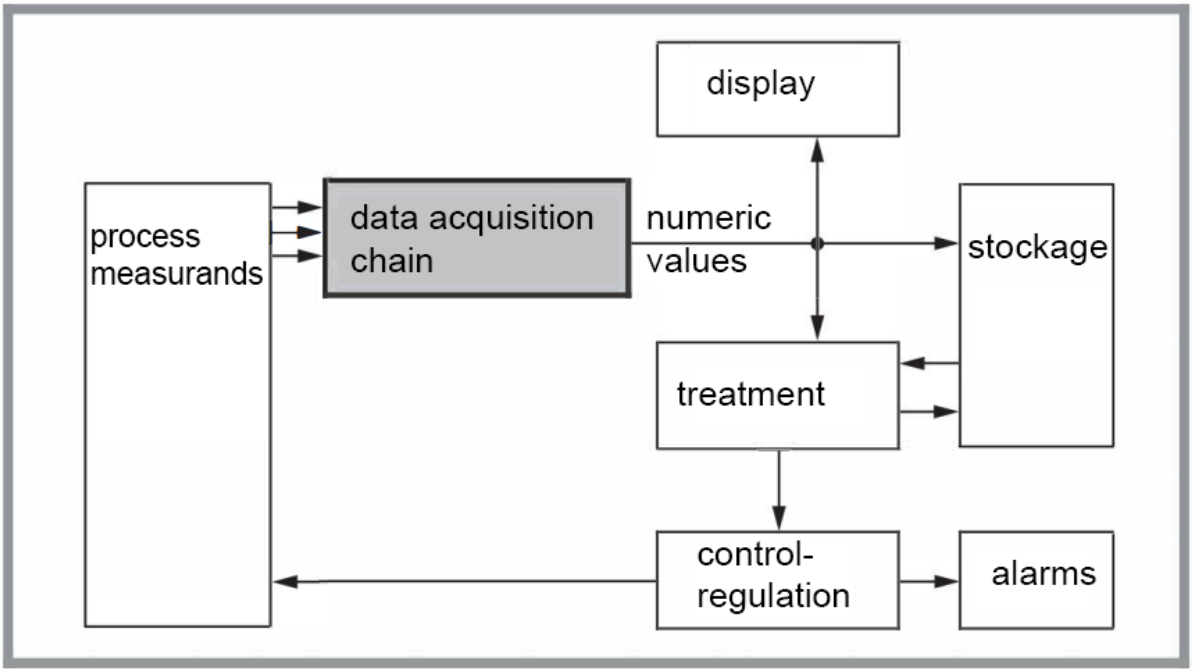


Figure I-1: Position of the data acquisition chain in a process measurement-control assembly. [1]

The analysis, for its fundamental magnitudes, of the complex set of interactions that constitutes a process is obtained by a succession of instrumental operations, each having its own function: the acquisition chain is formed by the ordered and coordinated set of various devices performing these functions (Figure I-2).

In its basic structure, an acquisition chain must be able to ensure, by means of the appropriate devices, the following functions : [1]

- Extraction of information concerning each of the physical quantities to be known and translation into an electrical signal by means of sensors and conditioners.
- Analog signal processing intended in particular to avoid degradation by noise and parasites: amplification, filtering.
- Selection from all the available signals of the single required using the multiplexer.
- Conversion of the signal into digital form adapted to the computer in charge of exploiting it, by means of the sample-and-hold device and the analog-to-digital converter.
- The coordination of the preceding operations is ensured by a computer which, can be a way to display all information acquired from the acquisition system, in addition, can be responsible for various processing operations on the digital signals that it has acquired.

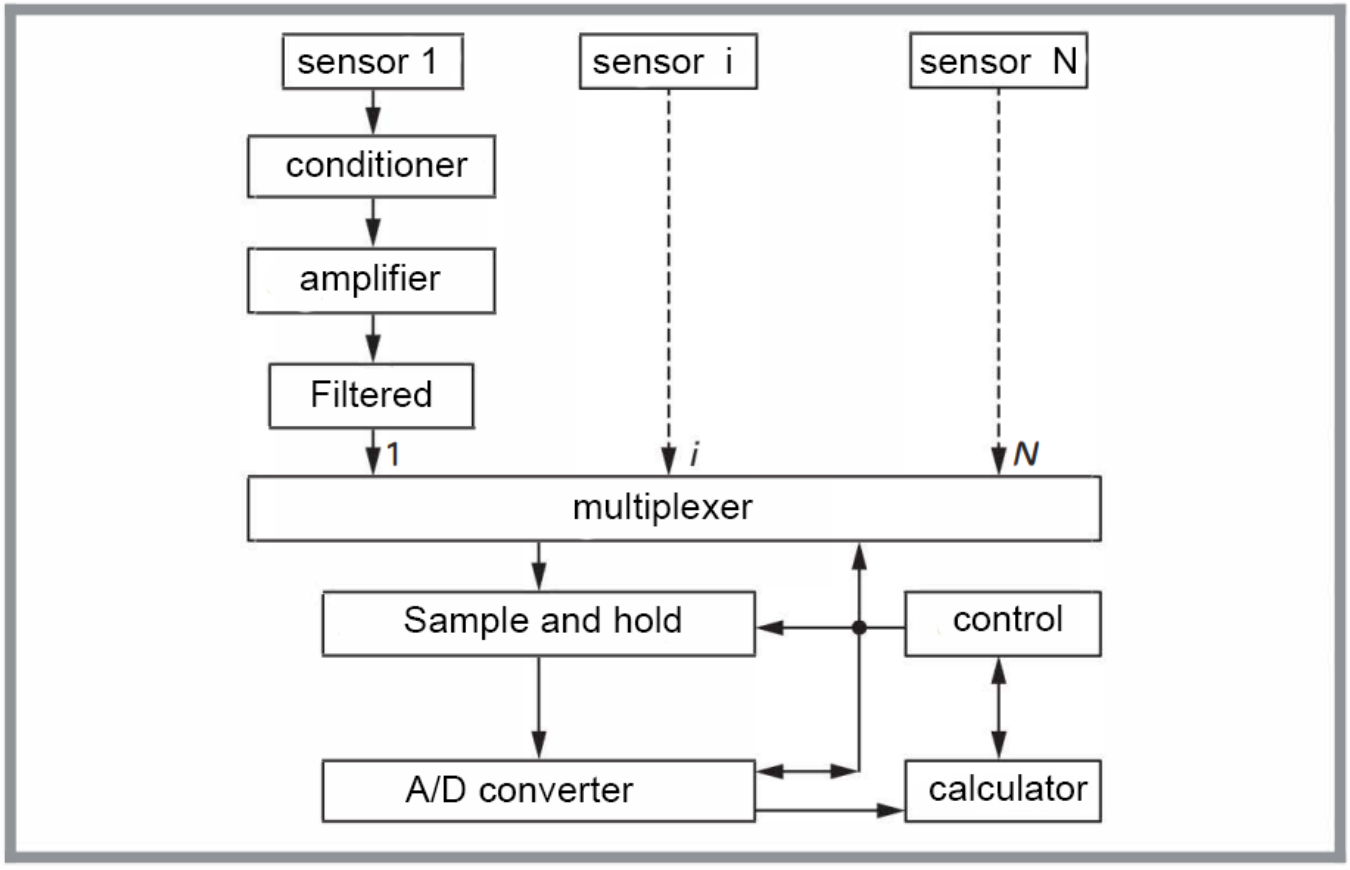


Figure I-2: Example of an acquisition Chain structure. [1]

I.2 Elements of a data acquisition system

The following elements are commonly found in data acquisition systems:

I.2.1 Sensors

Sensors are the first step in the acquisition process. Each measurand has a sensor whose job it is to convert the measurand's m-value into an electrical quantity$s(m)$, which is a known function of the measurand, so that each value s of this electrical quantity may be uniquely linked to the measurand's m-value. The origin of the electrical signal that the acquisition system processes is this electrical quantity $s(m)$.

I.2.1.1 Active and passive sensors

On the same rationale as filters, sensors are classed as active or passive. The energy sent to the next stage, signal conditioning, is obtained entirely from the measurand by a passive sensor, which has no power supply. Self-generating sensors are another name for passive sensors. Early record player pick-ups, for example, derived all of their acoustic energy from the kinetic energy of the record. Long-playing records with narrower grooves could not be developed until the mass of the pick-up was lowered by several orders of magnitude, which was accomplished with the introduction of amplification.[2]

As a modulator, an active sensor can provide more energy to the next step than it consumes from the measurand. The output is modulated by the measurand and has the same frequency if the power supply is d.c. The output is the carrier frequency with sidebands at the signal frequency if the supply is a.c.[2]

I.2.1.2 Static performance of sensors

a) Accuracy

The sensitivity is the slope of the output vs input graph. This can be estimated in some circumstances, but most sensors are calibrated by applying a known input value and viewing the output. The static calibration is the outcome of performing this test slowly. A specification for mass-produced sensors includes a sensitivity rating and a tolerance band based on limited numbers of carefully tested samples. A sensor's sensitivity must be within the tolerance for it to be acceptable.[2]

The tolerance is viewed as the uncertainty that restricts the accuracy of this type of sensor, which is assumed to have the given sensitivity. Individual calibration certificates are included with more expensive sensors. The thin line in Figure I-3 reflects the true or nominal sensitivity. Gain error can be accounted for through software or signal conditioning, and on some amplifiers, the user adjusts a dial to the sensor sensitivity as specified in the calibration certificate, and the sensor and amplifier's total sensitivity is then standardized.[2]

b) Offset

The offset is the output when the input is zero (or whatever input should give zero output). Signal conditioning and software can both adjust for offset.[2]

c) Linearity

A sensor's output is a function of its input, although it is not always proportionate. A radiation temperature sensor is an example of an extremely nonlinear sensor. Thermocouples are linear for minor temperature changes but nonlinear across their whole range. Most other sensors, on the other hand, are close enough to linear that any deviation from linearity is considered an error.[2]

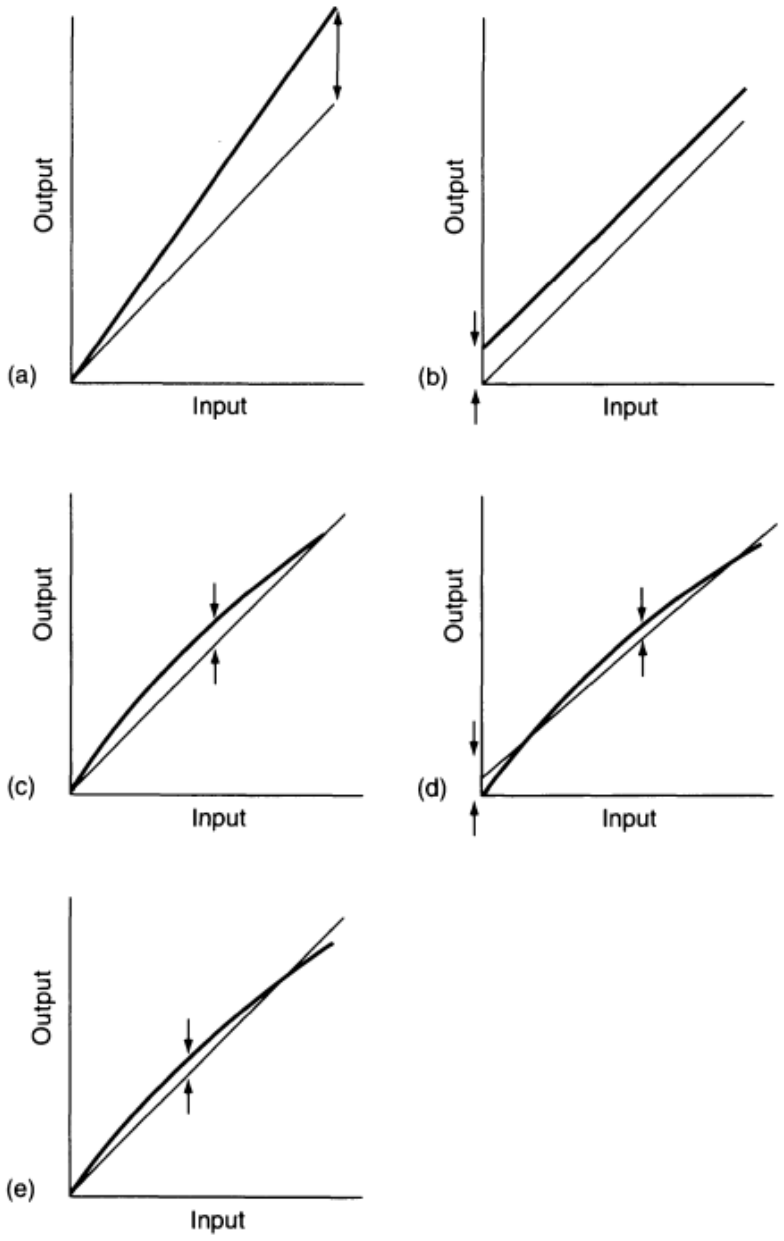


Figure I-3: Errors in sensors: (a) gain error, (b) offset, (c) terminal linearity error, (d) independent linearity error, (e) zero-based linearity error.[2]

The same curve is shown in Figure I-3 (c)-(e) with the nonlinearity given in three different ways. In (c), the actual output is compared to a straight line connecting the two curve endpoints, with the biggest difference between the two lines indicating the terminal linearity defect. This is the same definition that DIA and AID converters utilize. The output is compared to the best straight line through all of the calibration points in step (d).[2]

A least squares fit calculation can be used to find the best straight line. This is the error in independent linearity. This is a common specification for sensors, and it yields an error that is around one-third of the terminal linearity error for the same curve. Check which method was utilized while reading a linearity specification. Even though there is no offset, the best straight line may not pass through zero, as shown in (d). The best straight line that passes through zero is used in the third approach shown in (e).[2]

I.2.1.3 Temperature sensors

Temperature is one of the most widely measured quantities in industry and elsewhere, and many physical properties are temperature related.

a) Thermocouples

A potential gradient exists whenever a temperature gradient exists along a conductor. The **Seebeck** effect is what causes this. The magnitude of the potential gradient is determined by the conductor's substance. The emf between the cold part and the hot part on one side is perfectly matched by the emf on the other side of the circuit from hot to cold if a circuit constructed of a uniform material is not at a uniform temperature. If the circuit is made up of two materials A and B, with one junction hot and the other cold, the emf in A from cold to hot is not equal to the emf in B. The net emf that results can be used to determine the temperature difference. This is a thermocouple. Heat is absorbed or released at the junction as current flows from one substance to another, depending on the direction of current flow. 2]

Material pairs that are typically used for practical thermocouples are recognized by code letters. Their properties are summarized in TableI-1. The thermoelectric or Seebeck emf is not strictly proportional to the temperature difference, as can be shown, and full tables linking emf to temperature difference for all of these materials have been published (Quinn, 1990; BS 4937 (1981) and IEC 584-1; Kaye, 1986).[2]

Thermocouples come in a variety of shapes and sizes to suit diverse applications. The quickest response comes from bare wire couples, but they are also the most vulnerable. The thermocouple can be encased in a steel sheath, just like a resistance thermometer, to measure the temperature of a liquid, and its response time is then controlled as mentioned above. Two strips of appropriate metals are welded together side by side to form a tab that may be pressed against a solid surface to detect its temperature.[2]

Table I-1: Common thermocouple materials. [2]

Letter code	Materials	Usable range (℃)	emf, cold junction 0℃ (mV)		
			20 ℃	100 ℃	400℃
T	Copper-Constantan Cu-(60% Cu, 40% Ni)	-200 to 400	0.79	4.28	20.87
J	Iron-Constantan Fe-(60% Cu, 40% Ni)	-200 to 1200	1.02	5.27	21.85
K	Chromel-Alumel (90% Ni, 10% Cr)-(94% Ni, 2% AI etc.)	-300 to 1300	0.80	4.10	16.40
S	Platinum-Platinum 10% Rhodium	-100 to 1700	0.11	0.65	3.26
R	Platinum-Platinum 13% Rhodium	-100 to 1750	0.11	0.65	3.41

I.2.1.4 Force Sensor

Force is a measure of how objects interact, mass is a measure of how much matter is in an object, weight is the force of gravity acting on an object, and load is the force acting on a surface or an object. Short-range atomic force, electromagnetic force, and gravity are all examples. Force is a direction and magnitude vector. Newton (N) is the SI unit of force, defined as the force that transfers a kilogram of mass to 1 meter per second.[3]

a) Load Cells

For force measurement, load cells are often utilized. Flexible load-bearing components or component combinations are used in many load cells. The elastic element flexes as a result of the force applied to it, which is detected by the auxiliary sensor and converted into a measured output. Electrical signals can be used as the output.[3]

b) Wheatstone Bridges

The strain gauge bonded on the elastic component is a secondary sensor which converts the strain delivered by the force applied on the essential component into a resistance alters. The classic arrangement related to the strain gauges is the low-power, Wheatstone bridge.[3]

A Wheatstone bridge is typically made up of four strain gauges, while a half-bridge or multiples of four on complex-shaped flexible bodies can be made with only two SGs.[3]

In numerous graphic styles, the classic Wheatstone bridge can be represented as an input output device (two-port element), and three of which are shown in Figure I-4:

a) Rectangle – when the bridge is inserted into a complex measurement chain.
b) Rhomb – most useful and suggestive for strain gauges applications.
c) Square – rarely used, especially by American authors.

Due to the perfect symmetry, the input (power supply or excitation) and output (measuring signal or sensing) diagonals are interchangeable. [3]

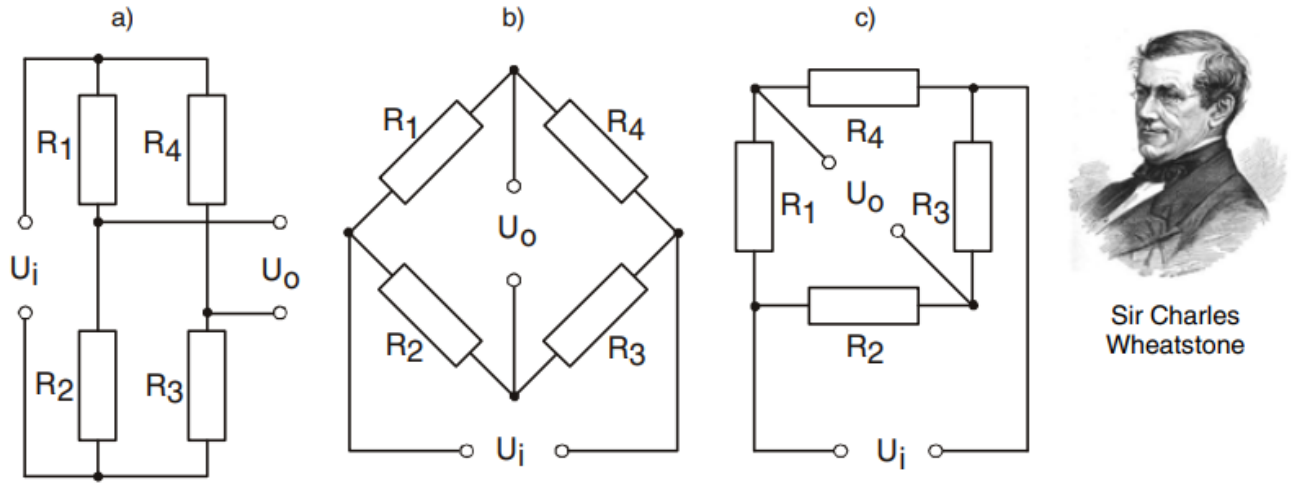


Figure I-4: Wheatstone bridge representations: a) rectangular, b) rhomboidal, c) square. [3]

c) Wheatstone bridge – fundamental properties

The simple bridge (Figure I 5.) is a full network consisting of four sides (with strain gauge values of 120Ω, 350Ω, or 1 $k\Omega$), four nodes (A, B, C, D), and two diagonals (U_A – supply, U_E – signal).[3]

Applying Kirchhoff's rules to the situation where the gauge is powered by a constant voltage source with little internal resistance and the load is an amplifier with almost infinite input impedance, the following results:

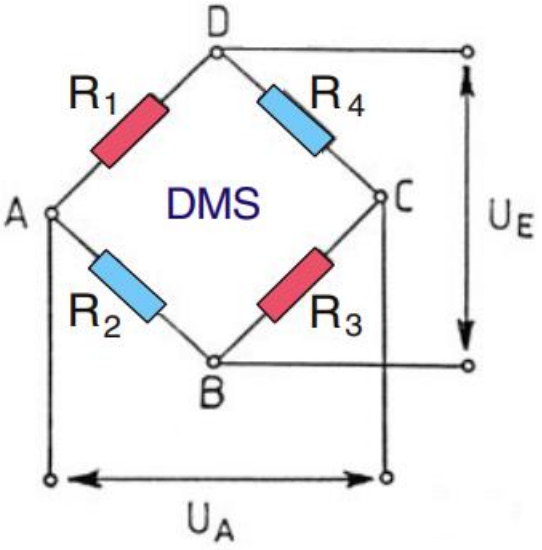


Figure I-5: Wheatstone bridge. [3]

$$\frac{U_E}{U_A} = \frac{R_1}{R_1+R_2} - \frac{R_4}{R_3+R_4}$$

The initial balance condition is:

$$R_1 \cdot R_3 = R_2 \cdot R_4$$

Meaning that the products of the resistances in the opposed arms should be equal and confirming the so-called

Golden rule of strain measurement by wheatstone bridge:

-The effects from two opposite arms is added**.**

-The effects from two adjacent arms are subtracted.

I.2.2 Analogue to digital converters

An analog-digital converter (ADC) receiving at its input an analog voltage vi outputs a word of n bits corresponding, according to a determined binary code, to the digital value N associated with vi.[1]

I.2.2.1 General characteristics

A CAN is characterized by:

-The convertible analog voltage range:

- For a unipolar converter, the range extends from $0V$ to V_{pe}called full scale voltage ($10V$ for example).
- For a bipolar converter, the range is symmetrical around $0V$: from $-V_{pe}/2$ to + $V_{pe}/2$, ($-5V$ to $+5\ V$ for example) the extension of the range being equal to V_{pe}.

The voltage V_{pe} is equal to a voltage E_{ref} applied to the ADC.

-The number n of bits of the output word (n = 8, 1 0, 12, 16, etc.)

-The time you need to perform a conversion.

The output word having n bits, the converter can deliver 2^ndistinct words (from 0 to $2^n - 1$) to digitize the extension analog voltage range V_{pe}. Each word is therefore associated with an elementary range q called the **quantum** (Table I-3):[1]

$$\boldsymbol{q = \frac{V_{pe}}{2^n}}$$

Table I-2: Value of the quantum q according to the number n of bits (Vpe: 10V) .[1]

$\boldsymbol{n}$	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
$\boldsymbol{q(mV)}$	625	312.6	156.3	78.2	39.1	19.6	9.8	4.9	2.4

$\boldsymbol{n}$	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**
$\boldsymbol{q(mV)}$	1.2	0.6	0.3	0.15	0.08	0.04	0.02	0.01	0.005

The digital value N of the binary word delivered by the ADC expresses the analog voltage v_i with the quantum q as unit, given that the same value of N corresponds to an analog input voltage range of width q, we have:

$$\frac{\boldsymbol{v_i}}{\boldsymbol{q}} = \boldsymbol{N} + \textbf{rest} \text{ Or } \boldsymbol{rest} < \boldsymbol{1}$$

If the output transition occurs when $^{v_i}/_{q} = N$, the set of analog values v_i corresponding to the same value of N is such that:

$$Nq \ll v_i < (N+1) \cdot q$$

The uncertainty on the analog value associated with N being able to reach q (Figure I-8).

Rather than an uncertainty of a single sign that can reach q (mean non-zero), we prefer a positive or negative uncertainty (mean zero) but limited to $\pm\frac{q}{2}$. To do this, we adjust the transition thresholds of the converter so that the digital output N appears for all values v_i such that (Figure I-6):

$$\left(N - {}^{1}/_{2}\right) \cdot q \ll v_i < \left(N + {}^{1}/_{2}\right) \cdot q$$

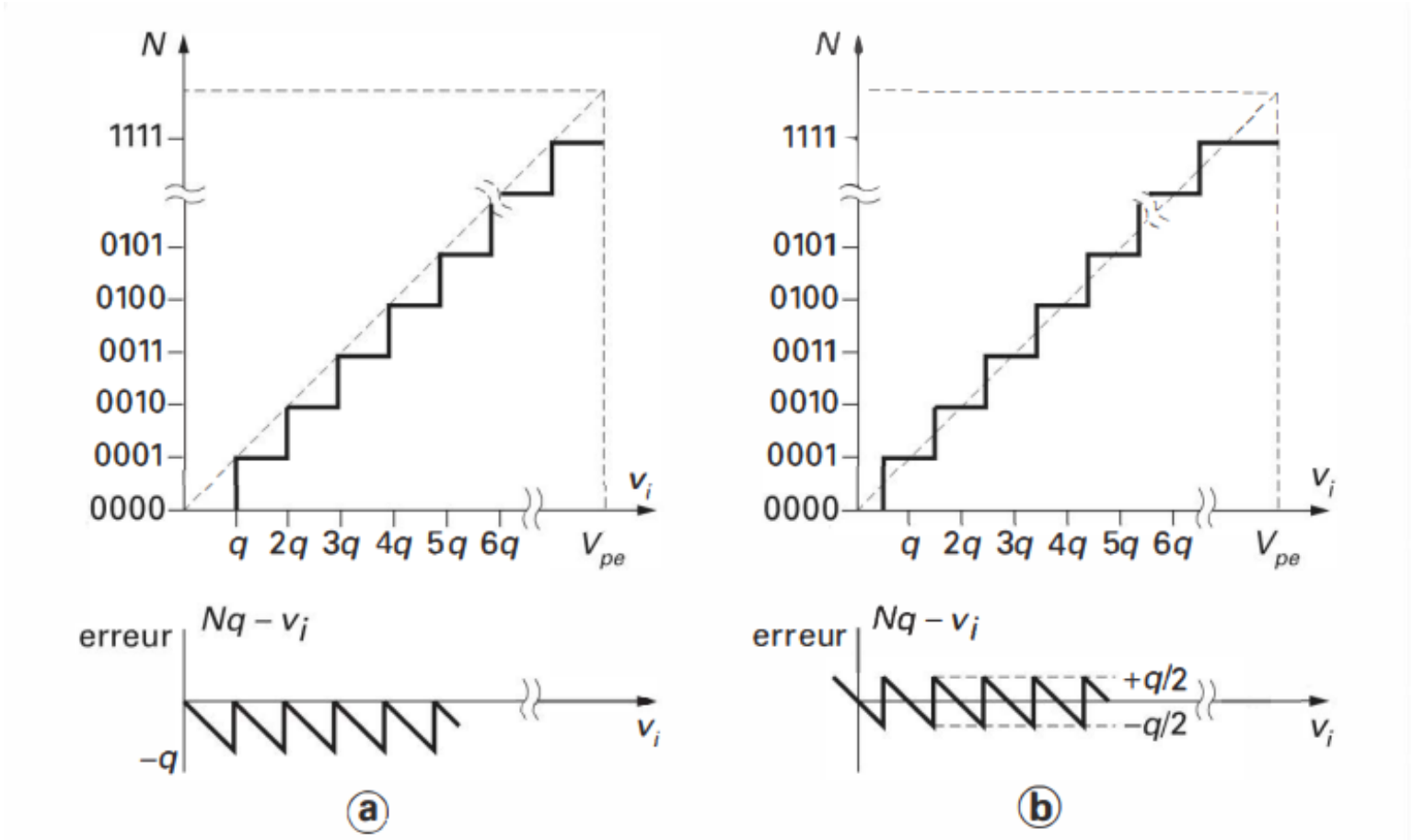


Figure I-6: Quantification errors and thresholds

a) thresholds at $v_i = Nq$ b) thresholds at $v_i = \left(N + {}^{1}/_{2}\right) \cdot q$

[1]

The uncertainty on the analog value v_i associated with N being limited to $\pm\, q/2$, the uncertainty on the digital value is $\pm\, 1/2$ least significant bit, is $\pm\, 1/2$ LSB (Less Significant Bit) we write:

$$\boldsymbol{N = \frac{v_i}{q} \pm \frac{1}{2}\,LSB}$$

$q/2$ Or $1/2$ LSB represent the maximum quantization error linked to the conversion process, error all the lower as$n$, the number of bits of the converter is higher.

I.2.2.2 Main conversion method

The rudimentary$\Delta\sum$ converter is a 1-bit sampling system. An analog signal sent to the converter's input must be slow enough for the converter to sample it numerous times, a process known as oversampling. The sampling rate is hundreds of times quicker than the output ports digital results. Through the digital/decimation filter, each individual sample is gathered over time and "averaged" with the other input-signal samples.[4]

The $\Delta\sum$ modulator and the digital/decimation filter are the key internal cells of the $\Delta\sum$ converter. Figure I-7 shows how the internal $\Delta\sum$ modulator coarsely samples the input signal at a very high rate into a 1-bit stream. After that, the digital/decimation filter turns the sampled data into a high-resolution, slower digital code. Unlike other converters, the $\Delta\sum$ converter has two sample rates: an input sampling rate (f_S) and an output data rate (f_D).[4]

a) The $\Delta\sum$ modulator

The heart of the $\Delta\sum$ ADC is the $\Delta\sum$ modulator. It's in charge of digitizing analog input signals and lowering noise at lower frequencies. At this point, the architecture employs a noise shaping function, which drives low-frequency noise up to higher frequencies, outside the region of interest. One of the reasons that $\Delta\sum$ converters are well-suited for low-frequency, high-accuracy measurements is noise shaping.[4]

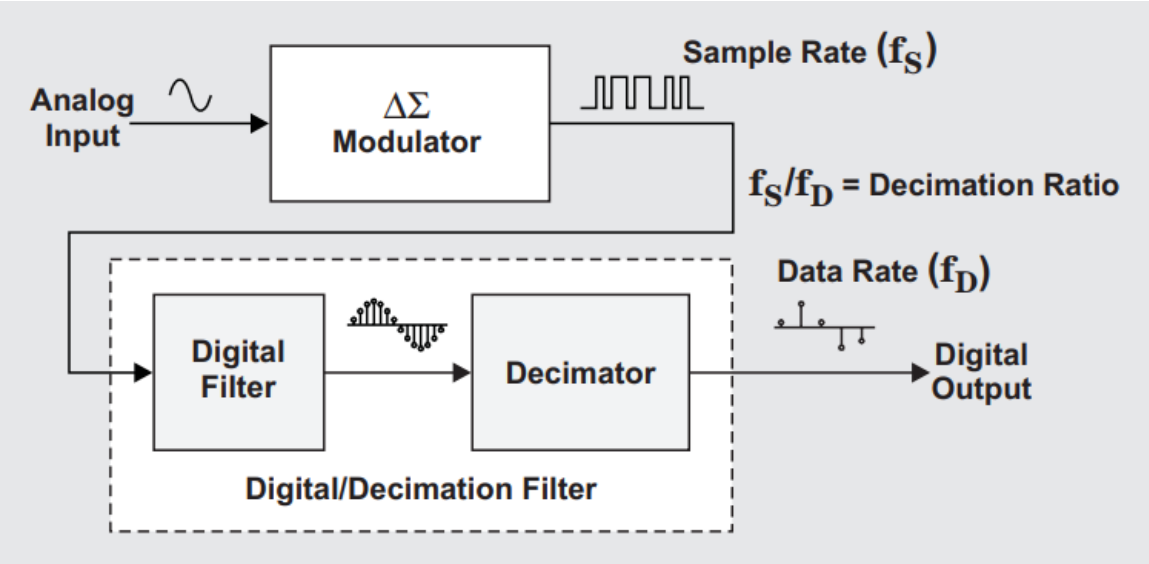


Figure I-7: Block diagram of $\Delta\sum$ ADC (Baker, 2016)

The ΔΣ modulator's input signal is a time-varying analog voltage. This input-voltage signal was used largely in audio applications where AC signals were relevant with early ΔΣ ADCs. Conversion rates include ΔΣ signals now that the focus has shifted to precision applications. A single cycle of a sine wave will be used to illustrate this concept.[4]

A single cycle of a sine wave for the input of a ΔΣ modulator is shown in Figure I-8 (a). The voltage amplitude of one single cycle varies throughout time. The frequency-domain representation of the time-domain signal in Figure I-8 (a) is shown in Figure I-8 (b). The continuous sine wave in Figure I-8 (a is represented by the curve in Figure I-8 (b), which appears as a straight line or a spur. [4]

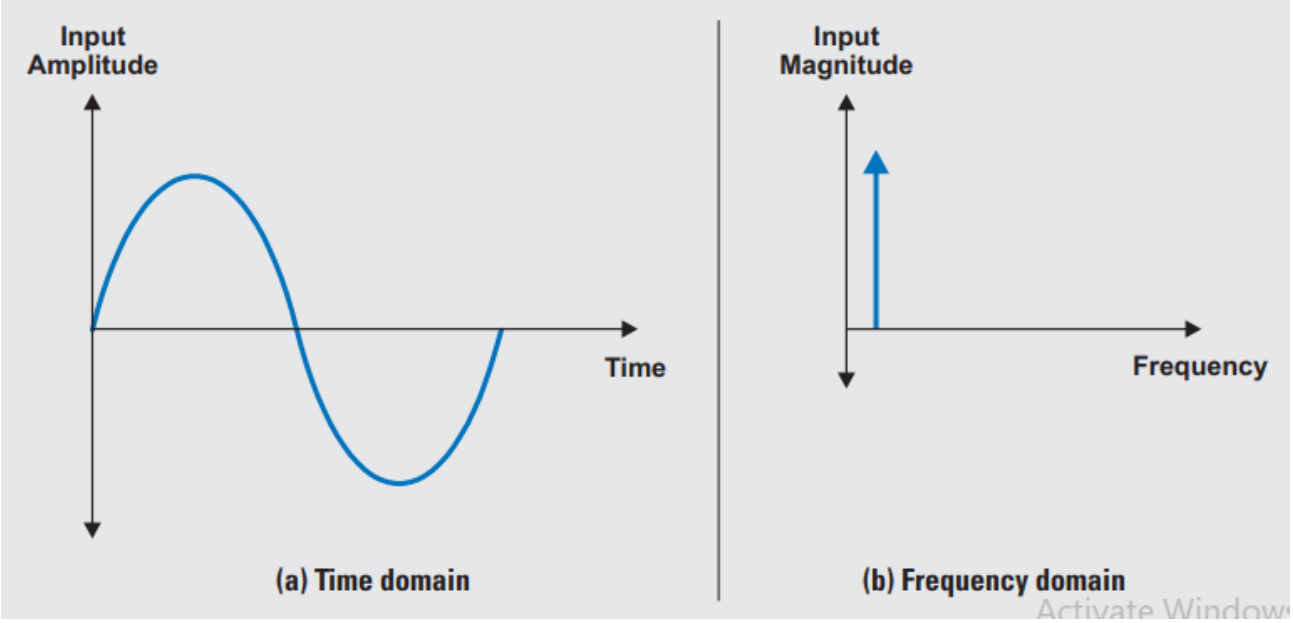


Figure I-8: Input signal to the ΔΣ modulator. [4]

b) The digital-filter function

By sampling the modulator stream of the 1-bit code first, the digital-filter function constructs a low-pass filter. A first-order low-pass averaging filter is shown in Figure I-4. In ΔΣ converters, the most frequent filter technique is an averaging filter.[4]

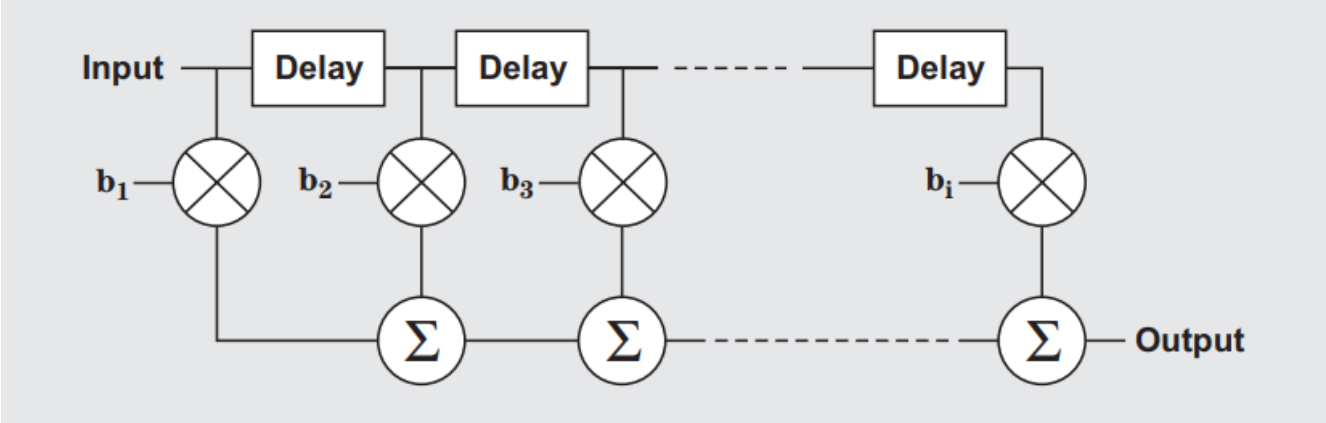


Figure I-9: First-order, low-pass averaging filter. [4]

I.2.3 Amplifiers

In general, an amplifier fulfills a triple function:

- By increasing its level, it ensures protection of the signal against interference, background noise and drifts of the following elements of the chain.
- By its high input impedance and its low internal impedance, it ensures optimal signal transfer between the devices it connects.
- It improves measurement accuracy by bringing the signal to the level required by the input scale of the final element of the chain (the ADC, analog-digital converter).

I.2.3.1 Differential amplifiers

A differential amplifier (Figure I-10) can be functionally considered as formed:

- A gain inverting amplifier: $-G_$.
- A non-gain inverting amplifier: G+.

An adder of the voltages from the two previous amplifiers.

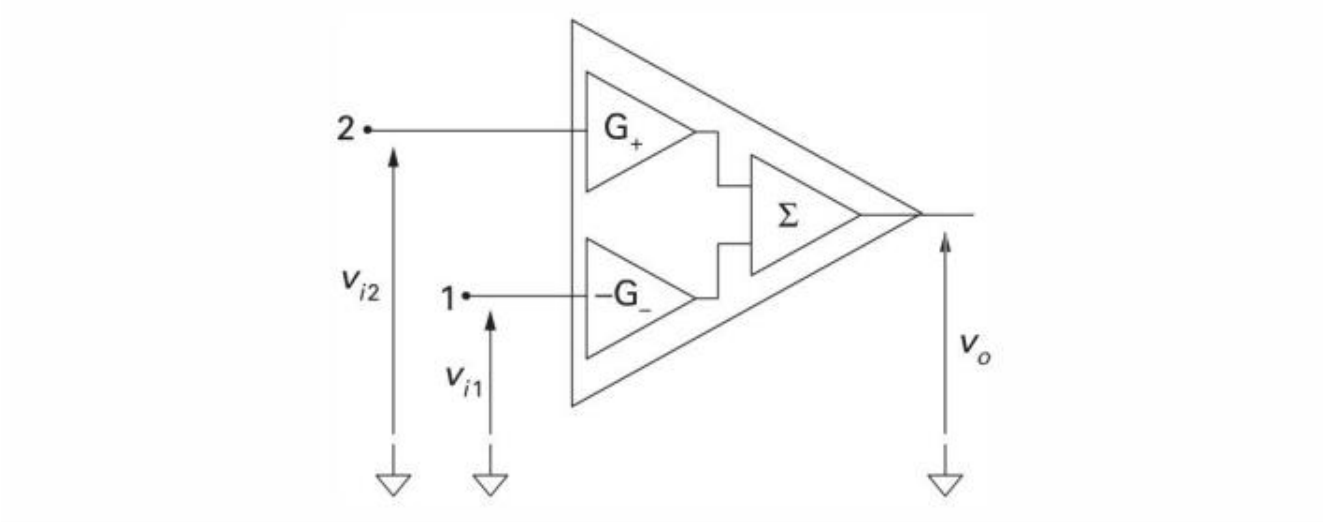


Figure I-10: Differential amplifier. Schematic diagram.[1]

The instrumentation amplifier (Figure I-11) has the following interesting characteristics:

- A large input impedance
- Gain adjustment by a single resistor $R_g$and requiring no modification of the paired resistors
- Perfect symmetry of the inverting and non-inverting pathways (in the ideal case).

We can distinguish on the assembly diagram (Figure I-11):

-An input stage formed by two operational amplifiers A.O.1 and A'.O.1 in non-inverting assembly.

-An output stage which is a differential amplifier with a single operational amplifier A.O.2.

The output voltages of the input stage have the expression:

$$\boldsymbol{v_{01} = v_{i1} + \frac{R_1}{R_g} \cdot (v_{i1} - v_{i2})} \quad \boldsymbol{v'_{01} = v_{i2} + \frac{R'_1}{R_g} \cdot (v_{i2} - v_{i1})}$$

Either, depending on $\boldsymbol{v_{d)i} = v_{i2} - v_{i1}}$ and $\boldsymbol{v_{mc)i} = (v_{i1} - v_{i2})/2}$:

$$\boldsymbol{v_{01} = v_{mc)i} - \left(\frac{1}{2} - \frac{R_1}{R_g}\right) \cdot v_{d)i}} \quad \boldsymbol{v'_{01} = v_{mc)i} - \left(\frac{1}{2} - \frac{R'_1}{R_g}\right) \cdot v_{d)i}}$$

The gain for the common mode of the first stage is equal to 1 which limits the risks of saturation. The output differential amplifier receives on its inputs the voltages $\boldsymbol{v_{01}}$and $\boldsymbol{v'_{01}}$which it amplifies:

- A difference with the gain: $\mathbf{G_{d2} = \frac{R_3}{R_2}}$.
- The common mode with the gain: $\mathrm{G_{mc2}} = \frac{4\varepsilon_R \mathrm{R_3}}{\mathrm{R_2}+\mathrm{R_3}}$, $\varepsilon_R = \mathbf{0.1\%} = \mathbf{10^{-3}}$

The output voltage of the instrumentation amplifier therefore has for express:

$$\boldsymbol{v_0 = \frac{R_3}{R_2} \cdot \left(1 + \frac{R_1 + R'_1}{R_g}\right) \cdot v_{d)i} + \frac{4\varepsilon_R R_3}{R_2 + R_3} \cdot v_{mc)i}}$$

From which we deduce:

- The differential gain: $\boldsymbol{G_d = \frac{R_3}{R_2} \cdot (1 + \frac{2R_1}{R_g})}$ for $\boldsymbol{R_1 = R'_1}$
- The common mode gain: $\mathrm{G_{mc}} = \frac{4\varepsilon_R \mathrm{R_3}}{\mathrm{R_2}+\mathrm{R_3}}$.

So that the gain $G_d$is variable from l, we choose R_3 =R_2, so that:

$$G_d = 1 + \frac{2R_1}{R_g} \text{ And } \mathrm{G_{mc}} = 2\varepsilon_R$$

We also note that the resistors $R_1$and R'_1 do not have to be matched with great precision since they only determine the differential gain and their possible difference has no effect on the common mode gain.[1]

The common mode rejection ratio of the instrumentation amplifier is:

$$\boldsymbol{\tau_r = \frac{G_d}{G_{mc}} = (1 + \frac{2R_1}{R_g}) \cdot (1 + \frac{R_3}{R_2}) \cdot \frac{1}{4\varepsilon_R}}$$

i.e., for $\boldsymbol{R_3}$ = $\boldsymbol{R_2}$:

$$\boldsymbol{\tau_r = (1 + \frac{2R_1}{R_g}) \cdot \frac{1}{2\varepsilon_R}}$$

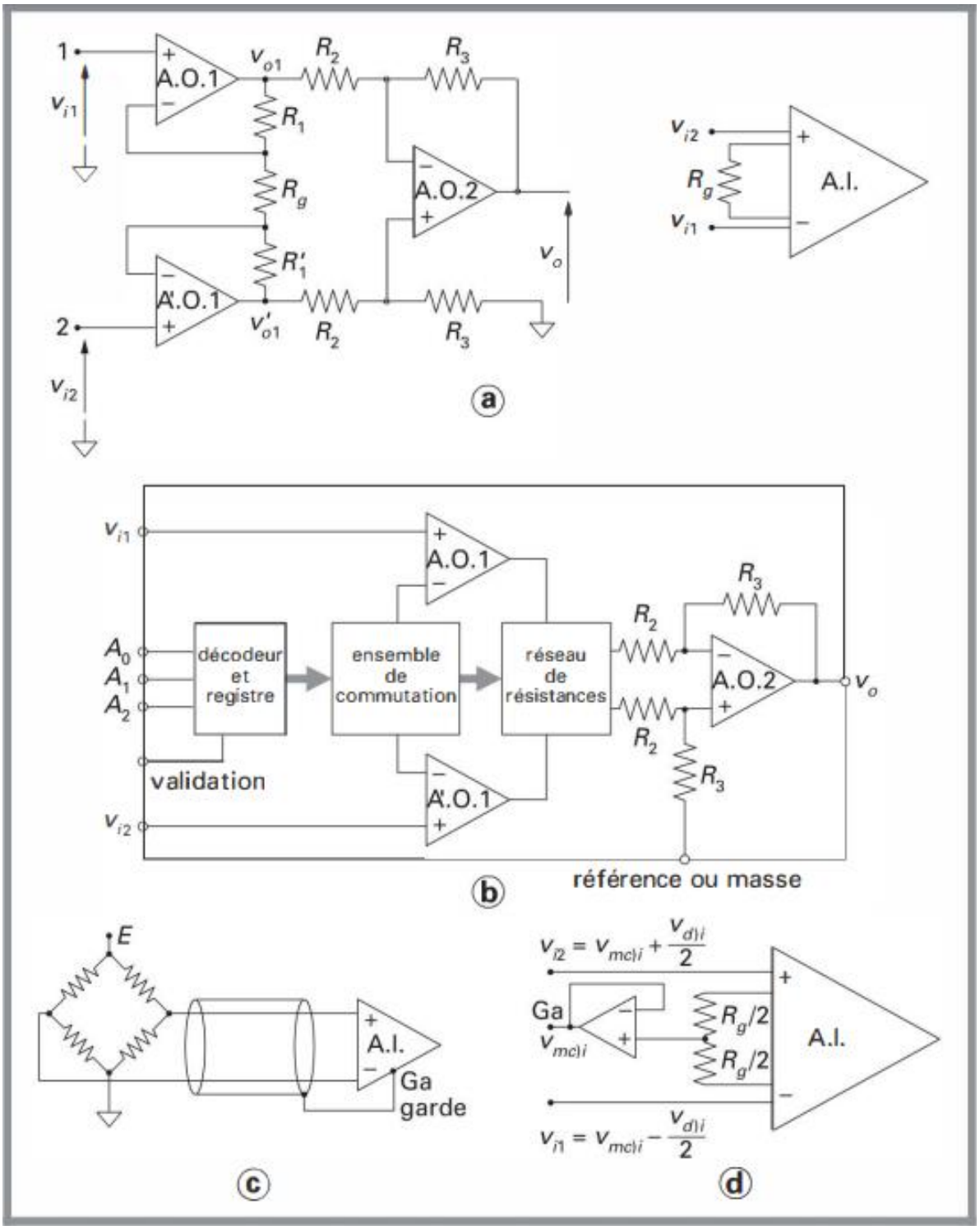


Figure I-11: Instrumentation amplifier: (a) constitution and representation; b) elements of a programmable amplifier; c) amplifier with guard output at common mode voltage; d) performing a guard exit.

The values of the rejection ratio are generally of the order of 90 db for G_d = 1 and 120 db for G_d = 10^3 for common mode voltages with a frequency of less than 60 Hz and a line imbalance limited to 1 kΩ. For higher frequencies or greater line imbalances, the rejection rate decreases.[1]

The programmable instrumentation amplifier allows the choice of a gain G_d from a set (generally 4 or 8) of possible gains. This choice is made by selection of resistors R_g and R_1 by means of a binary word $A_1\ A_0$, (1 among 4) or $A_2\ \ A_1\ A_0$, (1 among 8) applied by the controller to the gain selection inputs (Figure I-11 b).[1]

I.2.4 Multiplexer

I.2.4.1 General

When the data acquisition relates to multiple measurands, each having its own channel (sensor, conditioner, filter, etc.), the multiplexer allows the selection of a determined channel in order to route its specific signal to the processing devices located downstream.[1]

The selection of a channel is carried out by its address, binary word delivered by the controller of the chain and representative of its order number; it's decoding commands the closure of switches connecting the chosen channel to the output of the multiplexer. The switch is usually a field-effect transistor, more rarely a relay, changing from off to on when selected.[1]

According on the one hand to the voltage references of the signals of each channel, and on the other hand to the organization of the chain upstream of the multiplexer, each channel presents its signal on one or two lines. In the first case, the associated multiplexer has one channel per channel and is said to be unipolar (Figure I-12 a)

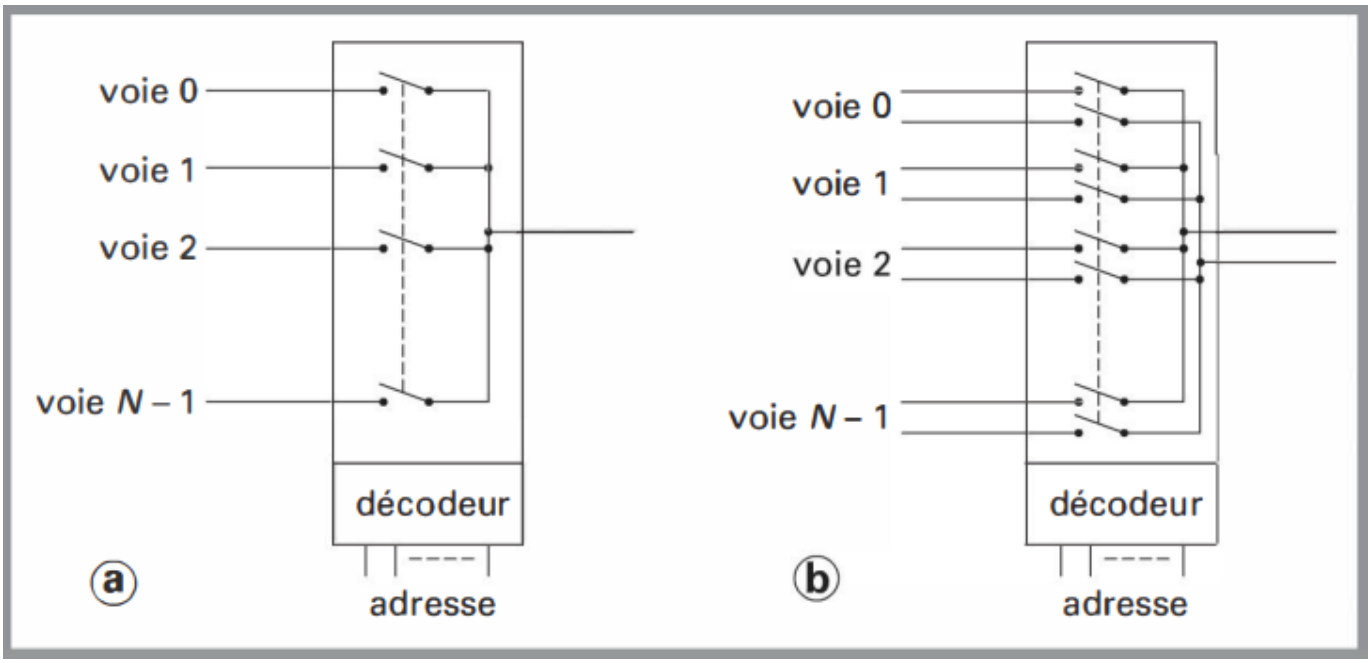


Figure I-12: Multiplex a) unipolar b) differential.

In the second case the multiplexer must comprise two channels per channel and it is designated as differential (Figure I-12b). It should be noted that a unipolar multiplexer with N input channels can be used, if it has the necessary controls, as a differential multiplexer with N/2 input channels. [1].

I.2.4.2 Measurement chain and choice of a multiplexer

Different configurations of the acquisition chain are possible depending on the voltage references (grounds) of the signal sources (sensors and any conditioners). This then results, depending on the case, in the use of unipolar or differential multiplexers.[1] Three cases can be distinguished:

- **The signal sources are all referenced to the same ground which is also ground M_i instrumentation (Figure a)**

In this case, there is no ground potential difference between the signal and the instrumentation. The signal referenced to the common ground can be transmitted by a single wire allowing the use of a unipolar multiplexer and an asymmetrical amplifier, possibly with programmable gain, when placed after the multiplexer. [1]

- **The signal sources are all referenced to the same ground which is distinct from ground M_i instrumentation (figure b)**

In this case, there is a ground potential difference v_M between the common ground of the sensors and the instrumentation. This potential difference appears as a common mode voltage if the signal is transmitted by two wires to an instrumentation differential amplifier. However, one of the wires, which is connected to the single ground of the sensors, is common and does not have to be multiplexed. It is therefore possible to use a unipolar multiplexer followed by a differential amplifier, one input terminal of which is permanently connected to the common ground of the sensors (pseudo-differential multiplexing).[1]

- **The sources of the signals are referenced to different masses distinct from the mass M_i instrumentation.**

Two solutions are possible in this case:

- There is in each channel an instrumentation amplifier which, receiving a differential signal from the sensor, delivers at its output an amplified signal referenced to the ground of the instrumentation. The multiplexer then only receiving signals referenced to the ground of the instrumentation can be unipolar (figure c).
- The differential signal from each sensor is selected by a differential multiplexer and delivered to an instrumentation amplifier, possibly programmable, and placed after the multiplexer (figure d) In this multiplexer, the conductors and the switches of each channel have capacitors interference with respect to the mass of the instrumentation. These capacitors which are in parallel on the inputs of the instrumentation amplifier risk degrading its rejection rate. This is why certain differential multiplexers are fitted with guard shielding, brought to the common mode potential of the channel scanned by a connection to the “guard” output of the amplifier. The potential difference due to the common mode voltage at the terminals of the parasitic capacitances then being cancelled, their influence is neutralized.

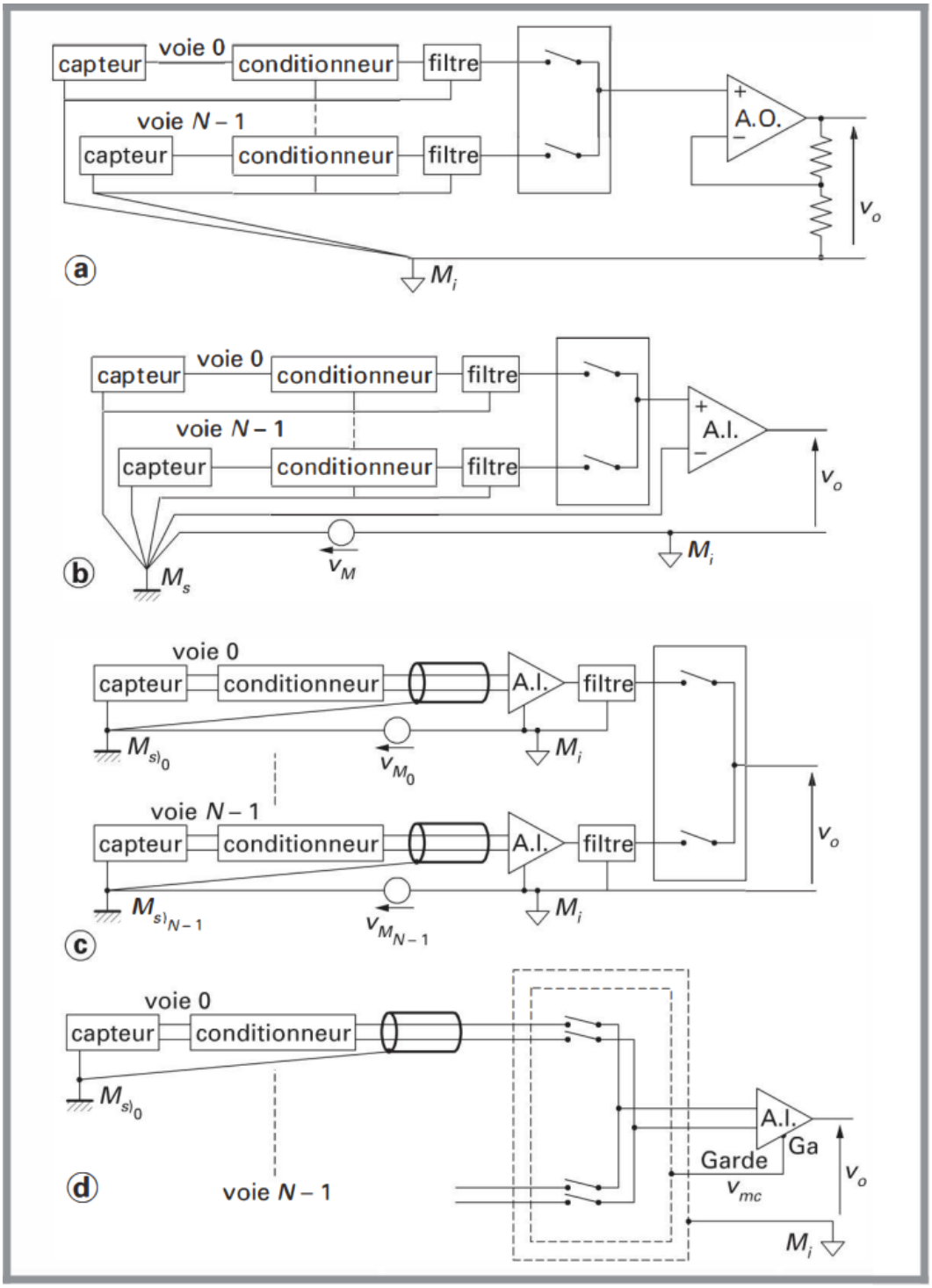


Figure I-13: Chain structure and associated multiplexer:

a) Sources and instrumentation have a common ground;

b) The sources have a common ground distinct from the ground of the instrumentation;

c) An instrumentation amplifier per channel and unipolar multiplexing;

d) Differential multiplexing and a single instrumentation amplifier;

I.2.5 Data Acquisition Software

DAQ software is the most important part of the DAQ system since it allows the DAQ hardware to communicate with a computer. Software for data collection.

Can be developed in a variety of languages (for example, C) and can be tailored to a specific design application. Alternatively, there are a variety of proprietary data gathering software programs available that can be used (see National Instrument with Labview). DAQ software typically consists of a text-based user interface (TUI) consisting of an ASCII configuration file and a graphic user interface (GUI) accessible via any Web browser, for example.[5]

The GUI interacts with the ASCII configuration file at a higher level, assisting the operator in compiling the file and controlling the acquisition. The Web interface eliminates grammatical errors by requiring no knowledge of the configuration file syntax. When it comes to changing the DAQ setup, the operator has the option of using the TUI or the GUI.[5]

Data Acquisition Software performs the following functions:

- To provide a more user-friendly working environment for the user, in which the user is prompted or directed interactively through the processes of operations without risk of becoming confused.[6]
- To aid in the establishment of a connection between the DAS and the host computer. The required interaction environment (e.g. handshaking needs, baud rate, etc.) will be automatically set up using this feature; the communication process will be transparent to the user.[6]
- To plot the data in a graph that can be seen in real time on the VDU. Some more advanced software can even display a "process" on the screen using real-time graphics. They offer a wonderful man-machine interface that rivals that of the most expensive centralised control systems.[6]
- Controlling the process. The process engineer can extract control actions straight from these software without having to get into the complexities of implementing the control algorithms.[6]
-

I.3 Conclusion

This chapter has introduced the fundamentals of DAS Systems, including A/D and D/A, architectures, signal conditioning, and driver software. DAS products come in all shapes and sizes, from very inexpensive and simple single-function devices to sophisticated high-end measurement systems that have been designed to deliver performance, flexibility, and reliability across a wide range of environmental conditions. These high-performance systems may incorporate advanced features and functionality not covered in this article. However, the basic concepts discussed here are generally universal in the world of DAS and form the very basis of the most sophisticated DAS systems.

Chapter II Design and architecture of the electronic board

Introduction

The objective of this work is to realize a solid rocket propellant test bench with an electronic system capable of controlling firing and acquiring the necessary measurements. To achieve this objective, we made a design and realization of this system. The realized system is composed of two parts: software part and hardware part. In the rest of this chapter, we will describe the design and construction of these two parts. A development of a GUI to control and visualize the measurements is also requested.

II.1 Development plan

Embedded systems design can be overwhelming if you're tackling it without a good plan. However, a structured method, on the other hand, can assist reduce the design cycle down into manageable steps, allowing for effective planning, execution, and cooperation.

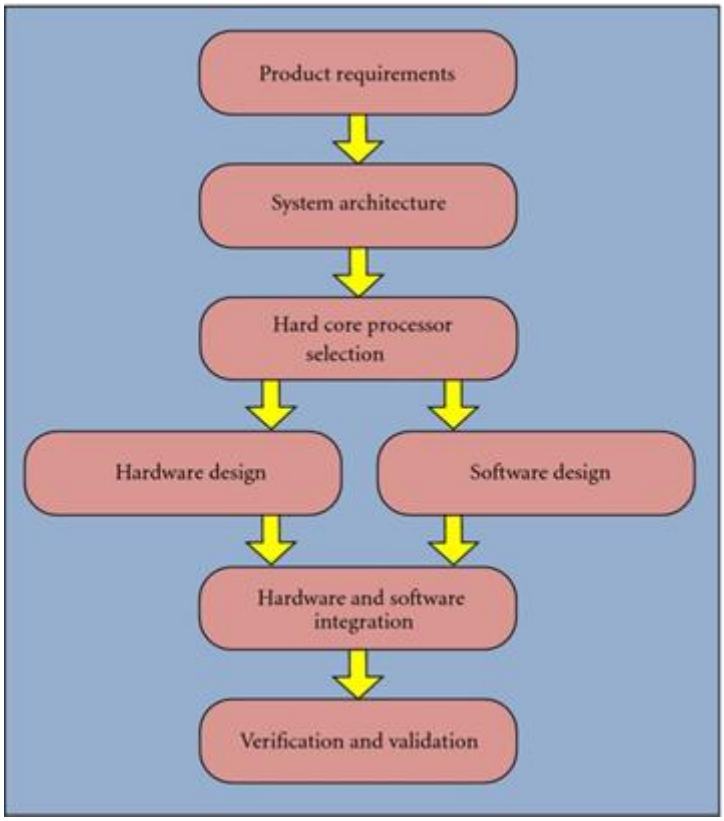


Figure II-1: Design process of an embedded system

II.1.1 Product requirements

The main focus of this project is to create a low-cost firing/ acquisition board that has 2 main modes locale, and remote mode.

Local mode: is when the engineer has the ability to start firing within the board, when the firing start all data acquired is going to be saved in an SD cart.

Remote mode: is when the engineer has the ability to start firing remotely using a user graphic interface along with viewing all data acquired through the sensors.

Several functional requirements must be met by this system:

II.1.1.1 A high level of Security

To ensure full Security when dealing with firing beforehand it is required to have a set of mechanisms and steps to avoid any accident during the firing. The following techniques are necessary:

- **Current continuity check**: once turning on the board the first-ever check has to be the courant continuity check to the ignition circuit if that is continuity then we are set to start the process of firing, if not we have to check what is wrong.
- **Light to ensure firing**: during the process of firing, it is crucial to have a light that indicates different steps of the process. For example when having good courant connectivity in the ignition circuit a green light would light up.
- **Lock switch**: firing button has to be handled carefully so having a lock switch with a specific key before accessing the firing button help a lot to avoid any accident during the process.
- **Time delay**: after pressing the firing button a reasonable delay of 50 to 100 seconds is needed to empty the room and leave the test bench area and get all the engineers ready for firing.

II.1.1.2 A high level of reliability

When choosing and purchasing the appropriate electrical components from a variety of options, digital system developers usually take into account a number of aspects. Cost, market share, maturity, dimension, and performance are some of these characteristics. Capability, efficiency, and dependability are all aspects of performance. A component's capability is its capacity to perform its needed or intended purpose. The component's efficiency refers to how readily and successfully it can perform its required function or achieve its objectives. The ability of a component to continue running without failure is characterized as reliability.[7]

One of the most important characteristics that affects the quality of electronic components for safety-critical applications is reliability. Electronic component producers and customers must agree on how to define and anticipate reliability. The practice of predicting a component's capacity to execute its specified function without failure over time is known as reliability prediction.[7] During the concept, design, development, operation, and maintenance phases, reliability predictions are made. It's also used for other things:

- To assess whether reliability goals can be reached
- To compare alternative designs
- To identify potential design weakness and potential improvement opportunities
- To plan logistics support strategies, such as spare parts provisioning and calculation of warranty and lifecycle costs
- To provide data for system reliability and safety analysis
- To estimate mission reliability

II.1.1.3 Saving and transmitting data

Saving data through acquisition is a highly must do when dealing with a test bench system

It's necessary to save data through an SD cart when working on the local mode, those measurements can be shown on the user graphic interface. On the other hand, when operating the test bench remotely using remote mode the data is going to be transmitted to the user interface, using protocols that must be selected from known standards such as: RS 422, RS485, etc.

II.1.2 System architecture

In the design phase, we must define the diagrams as well as the adequate resources for this project. The design in this project is composed of two parts, the firing part, and the data acquisition part. The selection of requirements to be used for both parties is completed based on two criteria cost and availability.

II.1.2.1 Possible proposals for the design of data acquisition

As we discussed in the previous chapter the data acquisition chain consists of multiple elements. These elements are sensors, analog to digital converters, amplifiers, and multiplexers. Choosing what component you can buy to reduce costs while maintaining reliability can be hard taking into account the availability of the component. Due to this case we have multiple propositions.

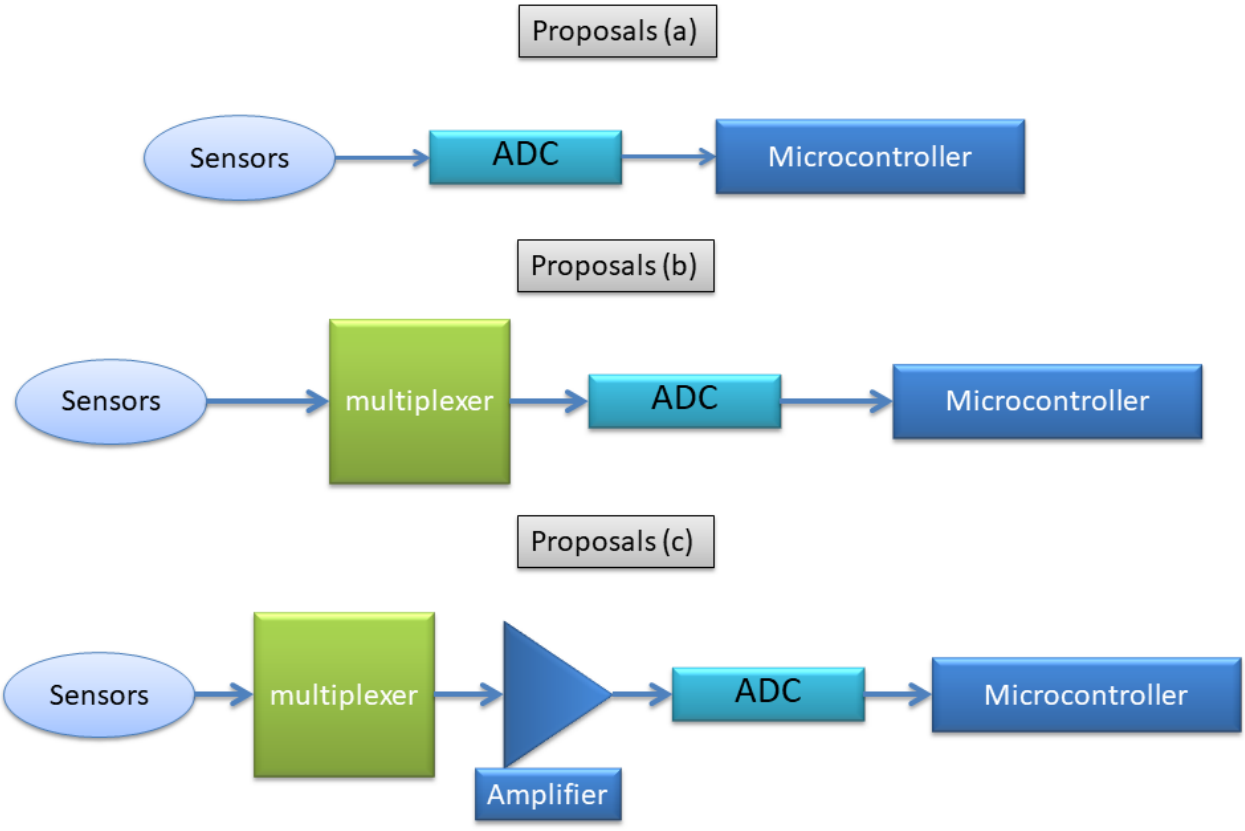


Figure II-2: Different propositions for the data acquisition chain

a) Proposition (a)

This is the best Proposition, It consists of multiple sensors connected to the ADC (Analog to digital converter) and then going to the microcontroller. The ADC in this proposition has to have a multiplexer for the different chains to have multiple measurements from different sensors. In addition, this ADC has to have an amplifier to boost the small sign we get from all sensors.

Those specifications can be applied on these ADC:

AD7794/AD7795:

The AD7794/AD7795 are full analog front ends with low power and low noise for high precision measuring applications. They have six differential inputs and a low-noise 24-/16-bit ADC. Signals of tiny amplitude can be interfaced directly to the ADC thanks to the on-chip low noise instrumentation amplifier.[8]

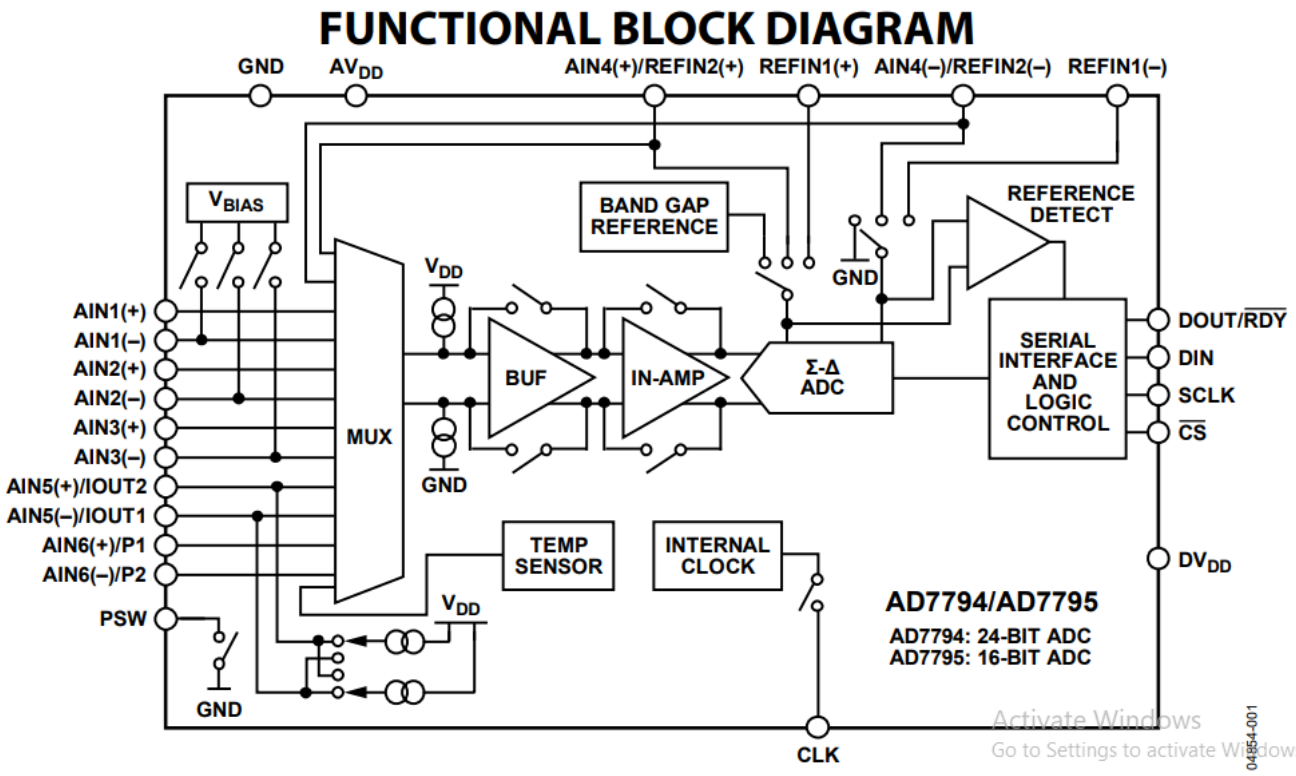


Figure II-3:Functional Block Diagram AD7794/AD7795 [8]

Features:

- Up to 23 effective bits
- Six differential analog inputs
- Power supply: 2.7 V to 5.25 V
- Temperature range: B grade: –40°C to +105°C C grade: –40°C to +125°C
- 3-wire serial interface SPI®, QSPI™, MICROWIRE™, and DSP compatible

Applications:

- Temperature measurement
- Pressure measurement
- Weigh scales

Having this type of ADC makes this proposition low cast and more official however, due to the unavailability of those ADC, Proposition (a) is not going to be taken into account.

b) Proposition (b)

This is the second best Proposition, It consists of multiple sensors connected to a multiplexer the output of this multiplexer is directly connected to the ADC and then goes to the microcontroller. The ADC in this proposition has to an amplifier to boost the small sign we get from all sensors.

Those specifications can be applied to this ADC:

AD7191:

The AD7191 is a low noise, complete analog front end for high precision measurement applications. It contains a low noise, 24-bit sigma-delta (Σ-Δ) ADC. The on-chip low noise gain stage means that signals of small amplitude can be interfaced directly to the ADC. It contains two differential analog inputs. The part also includes a temperature sensor that can be used for temperature compensation. The on-chip PGA has a gain of 1, 8, 64, or 128. [9]

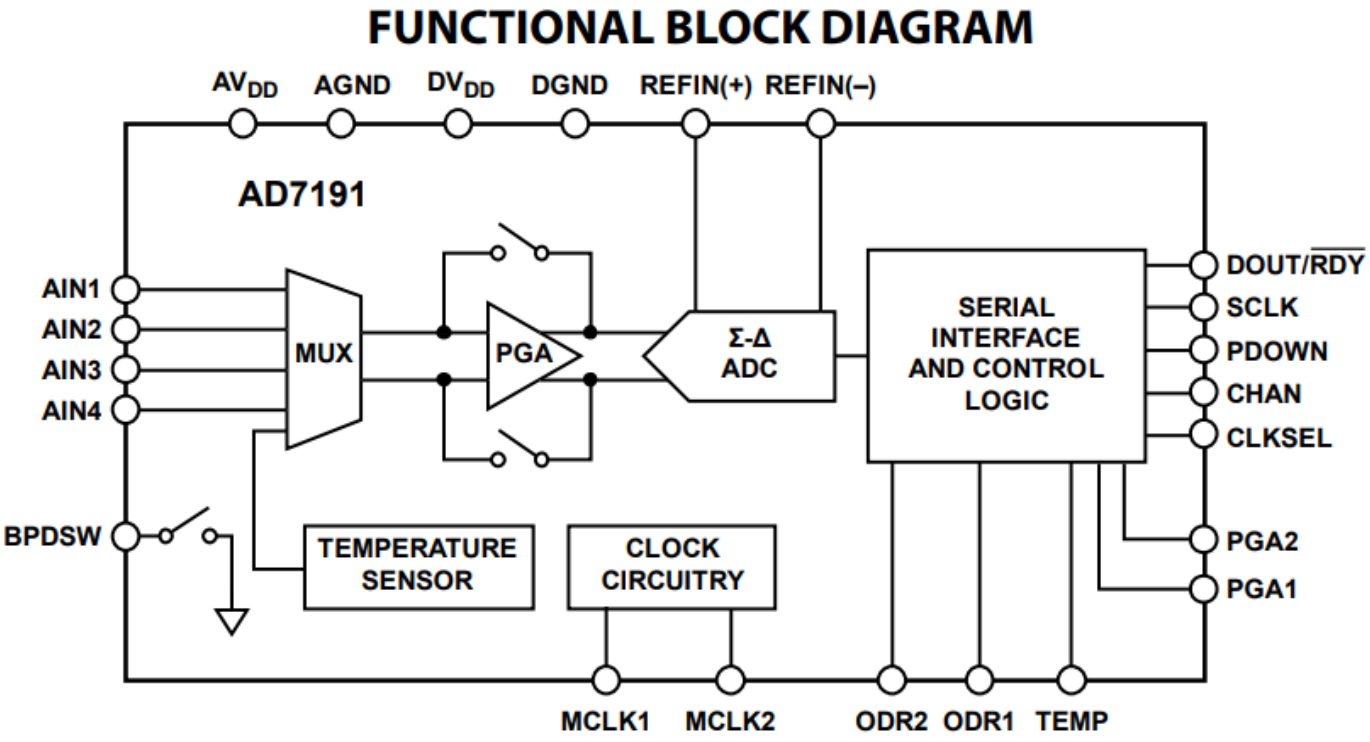


Figure II-4: Functional Block Diagram AD7191 [9]

Features:

- Pin-programmable output rate
- Pin-programmable PGA
- Gain: 1, 8, 64, 128
- Internal temperature sensor
- Temperature range: –40°C to +105°C
- Power supply: 3 V to 5.25 V

Interface:

- 2-wire serial
- SPI, QSPI™, and MICROWIRE™ compatible
- Schmitt trigger on SCLK

Applications:

- Weigh scales
- Pressure measurement

Having this type of ADC makes this proposition low cast and more official however, due to the unavailability of those ADC, Proposition (b) also is not going to be taken into account.

c) Proposition (c)

This is the third-best Proposition, It consists of multiple sensors connected to a multiplexer, the output of this multiplexer is directly connected to an amplifier then to the ADC, and then goes to the microcontroller. In this case the ADC chosen does not have to have multiple channels and specific amplifier, the main feature to look for is the ADC being sigma-delta and has more than 16 bit in resolution.

There are plenty of ADCs that fit these criteria.

ADS1118:

> The ADS1118 is a 16-bit precision, low-power analog-to-digital converter (ADC) with all the characteristics needed to measure the most common sensor signals in a leadless X2QFN-10 or VSSOP-10 packaging. A programmable gain amplifier (PGA), voltage reference, oscillator, and high-accuracy temperature sensor are all included in the ADS1118. These properties, together with a wide power supply range of 2 to 5.5 V, make the ADS1118 appropriate for sensor measurement applications that are power and space constrained.[10]

Features:

- Wide Supply Range: 2 V to 5.5 V
- Programmable Data Rate: 8 SPS to 860 SPS (Samples Per Second)

- Four Single-Ended or Two Differential Inputs

Applications:

- Temperature Measurement:
    - Thermocouple Measurement
    - Cold-Junction Compensation
    - Thermistor Measurement
- Portable Instrumentation
- Factory Automation and Process Controls

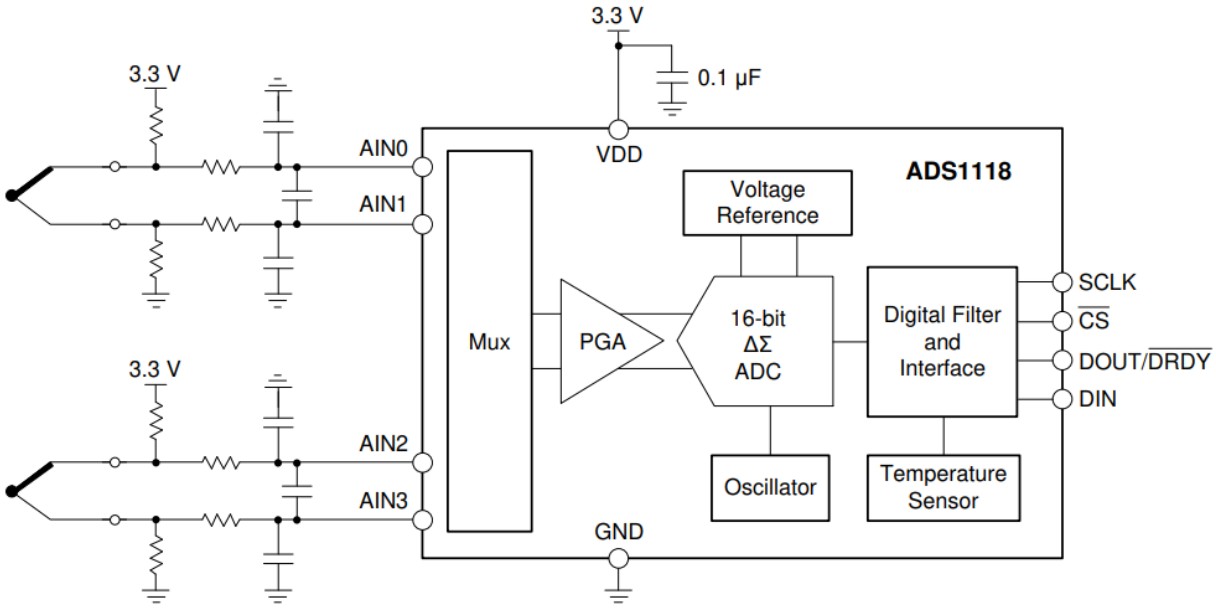


Figure II-5: K-Type Thermocouple Measurement Using Integrated Temperature Sensor for Cold-Junction Compensation [10]

ADS1115:

The ADS1115 is a 16-bit, low-power, precision analog-to-digital converter (ADC) available in an ultra-small, leadless X2QFN-10 package as well as a VSSOP-10 device. A low-drift voltage reference and an oscillator are included in the ADS1115 device. A programmable gain amplifier (PGA) and a digital comparator are also included in the ADS1115. These qualities, together with a wide working supply range, make the ADS1115 ideal for sensor measurement applications that are power and space constrained. [11]

The ADS1115 can convert data at speeds of up to 860 samples per second (SPS). The PGA allows precision large- and small-signal measurements with input ranges ranging from 256 mV to 6.144 V. An input multiplexer (MUX) on the ADS1115 provides for two differential or four single-ended input measurements. [11]

Features:

- Wide Supply Range: 2 V to 5.5 V
- Programmable Data Rate: 8 SPS to 860 SPS
- Internal Temperature Sensor: 0.5°C (Maximum) Error: 0°C to 70°C
- Programmable Data Rate: 8 SPS to 860 SPS
- I2C Interface: Four Pin-Selectable Addresses
- Four Single-Ended or Two Differential Inputs

Applications:

- Portable Instrumentation
- Temperature Measurement Systems

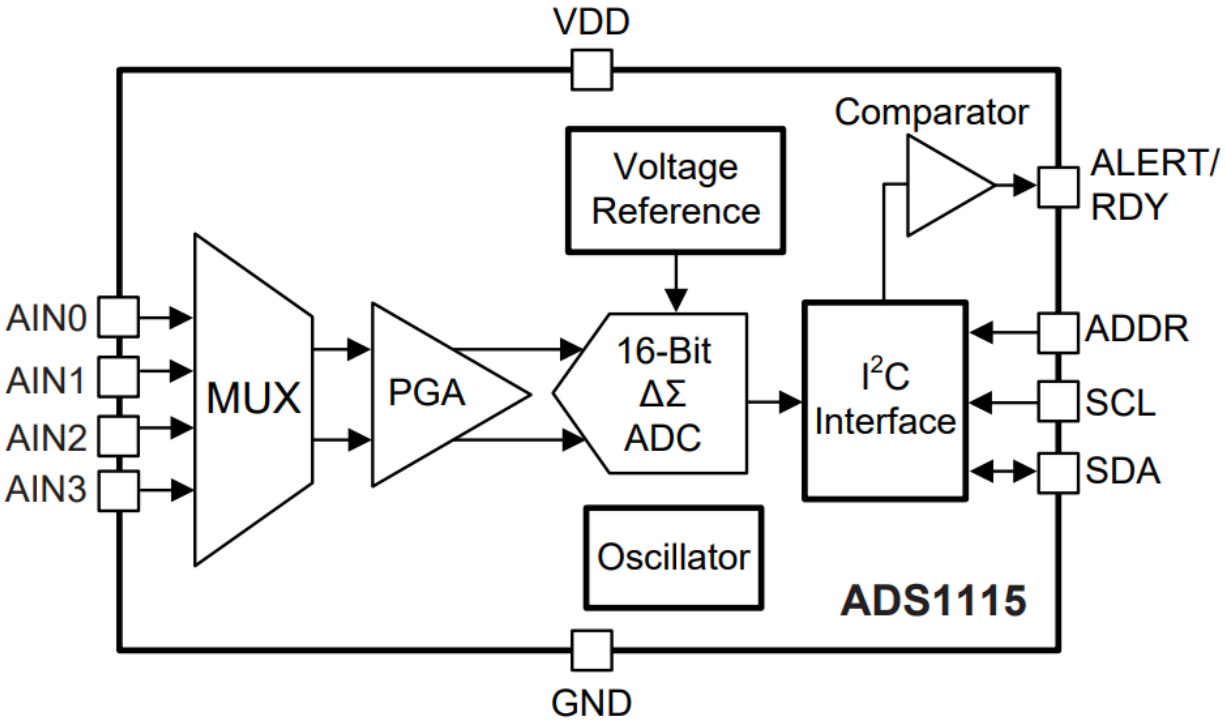


Figure II-6: Simplified Block Diagrams of ADS1115[11]

CS5509:

The CS5509 is a 16-bit serial-output CMOS (Complimentary Metal-oxide Semicodictor) A/D converter with a single supply. The CS5509 provides low-cost, high-resolution measurements at output word rates up to 200 samples per second using charge-balanced (delta-sigma) approaches.[12]

Features:

- Delta-sigma A/D Converter 16-bit, No Missing Codes
- Differential Input - Pin-selectable Unipolar/Bipolar Ranges
- Either 5V or 3.3V Digital Interface

- On-chip Self-calibration Circuitry
- Output Update Rates up to 200/second

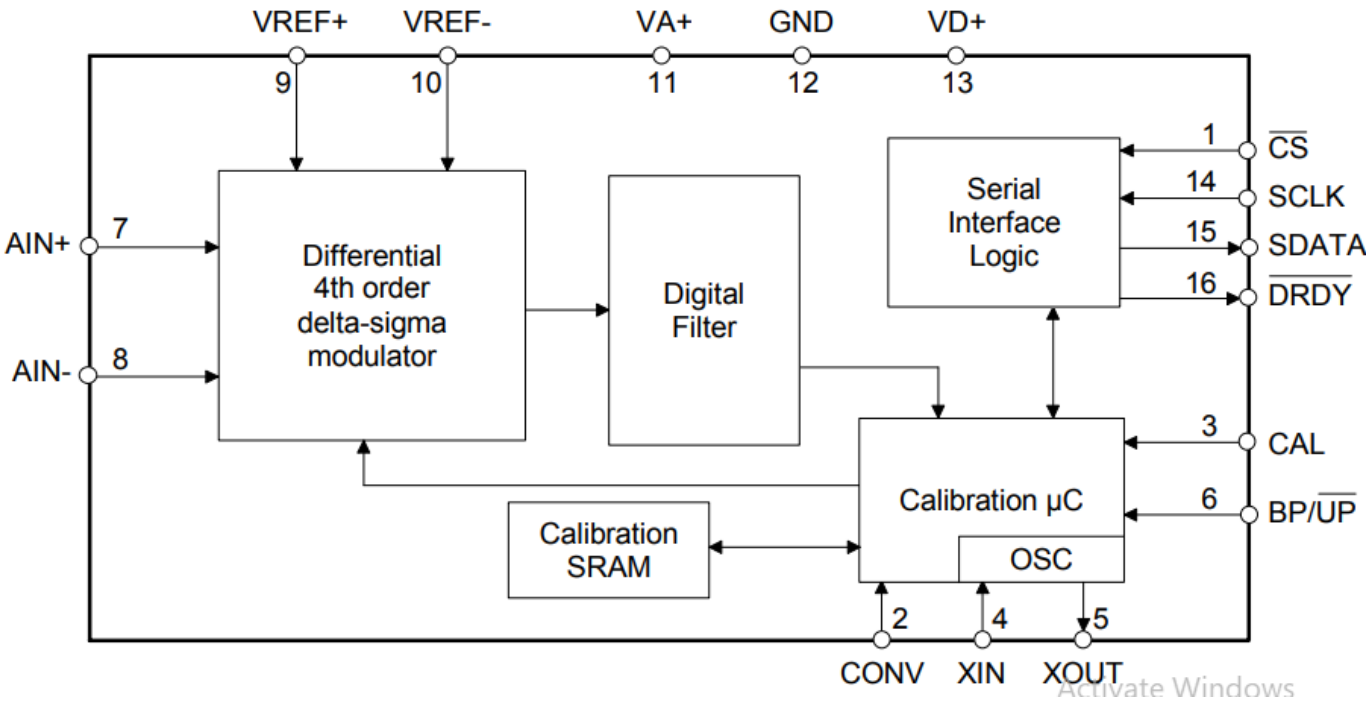


Figure II-7: Simplified Block Diagrams of CS5509. [12]

MCP3421:

In a tiny SOT-23-6 packaging, the MCP3421 is a single channel low-noise, high-accuracy A/D converter with differential inputs and up to 18 bits of resolution. The on-board precise 2.048V reference voltage allows for a differential input range of 2.048V (voltage = 4.096V). The gadget is powered by a single 2.7V to 5.5V power source and has a two-wire I2C compliant serial interface. [13]

Using the two-wire I2C serial interface, the MCP3421 device converts at rates of 3.75, 15, 60, or 240 samples per second (SPS), depending on the user adjustable configuration bit values. This device features a customizable gain amplifier on board (PGA). Before the analog-to-digital conversion, the user can select PGA gain of x1, x2, x4, or x8.[13]

Features:

- 18-bit ΔΣ ADC in a SOT-23-6 package
- Differential Input Operation
- Self-Calibration of Internal Offset and Gain Per Each Conversion
- On-Board Programmable Gain Amplifier (PGA): Gains of 1,2, 4 or 8
- Programmable Data Rate Options: - 3.75 SPS (18 bits) - 15 SPS (16 bits) - 60 SPS (14 bits) - 240 SPS (12 bits)

- Supports I2C Serial Interface: - Standard, Fast and High Speed Modes
- Single Supply Operation: 2.7V to 5.5V
- Extended Temperature Range: -40°C to +125°C

Applications:

- Portable Instrumentation
- Weigh Scales
- Temperature Sensing with RTD, Thermistor, and Thermocouple
- Bridge Sensing for Pressure, Strain, and Force

Table II-1: ADC specification comparison table

ADC	Resolution (bit)	Channels	Gain	Communication interface	Price (da)
ADS1118	16	2	1, 2, 4, 8, 16	SPI	2000
ADS1115	16	2	1, 2, 4, 8, 16	I2C	1200
CS5509	16	1	x	SPI	2000
MCP3421	18	1	1,2, 4 or 8	I2C	1000

All of the 4 previous ADCs are good pairs in this proposition, however, 2 of them are hard to find due to unavailability, and so what we have left is the 2 Texas instruments ADCs 1118 and 1115. We could use the 1118 and 1115 together in one board by separating the data acquisition chain and having one for temperature sensors and the other for force sensors. Or we could use one ADC for both types of sensors.

SPI protocol is common in various other electronic components we could use in this project, for example, an SD cart. by choosing an ADC that uses one it is very easy to control other SPI components with the same protocol. So choosing ADS1118 is the right pick.

II.1.2.2 Hard core processor selection

The majority of control engineering applications nowadays are computer-based, with the controller being a digital computer or a microcontroller. A typical computer-controlled system is depicted in the diagram (Figure II-8). The error signal is presumed to be analog in this case, and an A/D converter is employed to convert it to digital form so that the computer can read it.[14]

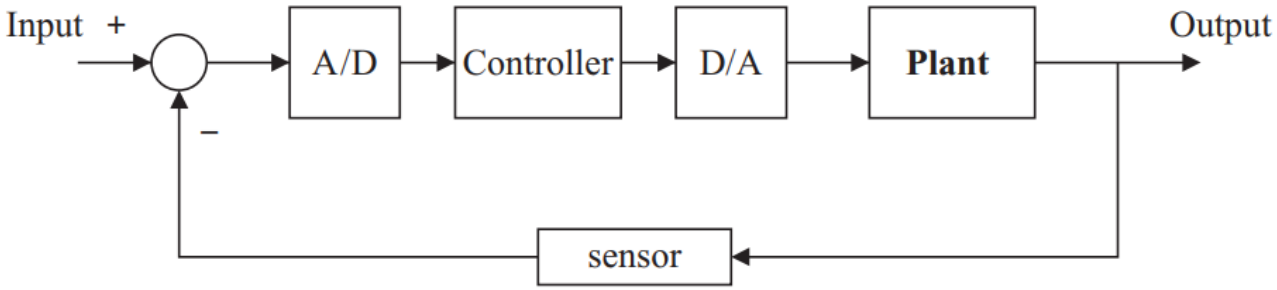


Figure II-8: Typical digital control system. [14]

The signal is sampled on a regular basis, and the samples are converted into a digital word that can be processed by the digital computer. The computer uses a controller algorithm (software) to carry out the appropriate activities, ensuring that the plant's output responds as expected. A digital computer's output is a digital signal, which is often transformed to analog via a digital-to-analog (D/A) converter. A zero-order hold transfer function is commonly used to imitate the behavior of a D/A converter.[14]

II.1.2.3 Microcontrollers

A microcontroller is a single-chip microprocessor designed specifically for control purposes in embedded computers. These devices are inexpensive and simple to use in digital control applications. The circuits required for computer control applications are incorporated into most microcontrollers. A/D converters, for example, may be included in a microcontroller so that external signals can be sampled. They also contain parallel input–output ports for reading and writing digital data to and from the microprocessor. Some devices include built-in D/A converters, and the converter's output can be utilized to control the plant via an actuator (e.g. an amplifier). A timer and interrupt logic may be integrated into microcontrollers. We can program the microcontroller to accurately implement the control algorithm by using the timer or interrupt features.[14]

Microcontrollers have typically been programmed in the target device's assembly language. As a result, the assembly languages of microcontrollers manufactured by various companies varied drastically, requiring the user to learn a new language before programming a new device. High-level languages such as BASIC, PASCAL, and C can now be used to program microcontrollers. When compared to assembly language [14], high-level languages have various advantages:

- It is easier to develop programs using a high-level language.
- Program maintenance is much easier if the program is developed using a high-level language.
- Testing a program developed in a high-level language is much easier.

- High-level languages are more user-friendly and less prone to making errors.
- It is easier to document a program developed using a high-level language.

In addition to the benefits listed above, high-level languages have significant drawbacks. When a high-level language is used, for example, the length of the code in memory is usually higher, and programs written in assembly language execute quicker than those written in a high-level language.[14]

In this project, PIC microcontrollers are used as digital controllers. The microcontrollers are programmed using the high-level C language.

II.1.2.4 Selection of the Microcontrollers

The process of selecting the right MCU goes through the following steps.

Step 1: Define the System Requirements

If you decide to employ MCU (Microcontroller Unit), the designer must first answer the question, "What does the MCU need to do in my system?" The answer to this basic question determines the system's necessary MCU features and, as a result, is the deciding factor in the selection process.[15]

Step 2: Make a list of all candidates for MCUs

Look for MCUs that satisfy all of the system's needs. This usually entails a literature search as well as consultations. If the search yields more than one MCU that meets all of the system criteria, this stage is considered successful. If this is the case, the designer will proceed to the following stage.[15]

Step 3: Make a final decision

Make an effort to narrow down the list of eligible MCUs to a single option.

II.1.2.5 Define the System Requirements

We offer an outline of the thought process that will guide the designer in compiling a comprehensive list of system requirements in this part. This is the most crucial step toward completing the selection procedure. The suggested method takes the shape of a series of questions that the designer must answer. Some of the questions the designer must ask and find answers to include. [15]

a) Questions related to the system hardware requirements:

To achieve a higher level of integration and reliability at a cheaper cost, all MCUs have on-chip resources. An on-chip resource is a piece of circuitry integrated into the MCU that performs a useful function under the MCU's control. Built-in resources are more reliable because they don't require any additional circuitry to function. Because the circuitry is integrated into the MCU, they are pre-tested by the manufacturer and save board space. On-chip resources are passed from one MCU to the next. Then, during the analysis phase, it's critical to precisely specify the system hardware requirements in order to select the appropriate MCU. [15]

Some of the questions to be asked and answer are:

How many devices/bits (I/O pins) need to be controlled? Among the many possible types of I/O devices to be controlled/monitored in this project are: RS232C terminals, switches, relays, keypads, sensors (temperature, pressure, light, voltage, etc.), audible alarms, visual indicators (LCD displays, LEDs), analog-to-digital (A/D), digital-to-analog (D/A), liquid crystal display drivers (LCD), and SD board reader. [15]

What is the expected capacity of the ROM and RAM that are required to store the programs and the data? To store and run all code required to accomplish all the tasks the board has to do it is necessary to have the right amount of ram and ROM in the processing unit in our case for this project having too much to control from data being acquired and saved also systems firing control having around 1.5 KB of SRAM and 256 Bytes of ROM, also Program Memory Size 32 KB is the sweet spot.

How many timers the application needs? Timers include both real-time clocks and periodic interrupt timers.

What is the length of the words? Based on the size of its arithmetic and index registers, MCUs can be divided into three groups: 8-bit, 16 bit, and 32 bit (s). A lower-cost 8 -bit MCU cannot be able to handle the requirements of the system, a higher-cost 16- bit or 32-bit MCU can do so. The 16- bit range for each instruction cycle is 0 to 65535.

b) Questions related to the performance:

As previously stated, performance can be measured by the amount of time it takes to execute a task. The clock speed is one of the elements that affects throughput. The clock speed, or more precisely the bus speed, controls how much processing the MCU can do in a given length of time. Some microcontrollers have a limited clock speed range, whereas others can run at zero. For example, for serial baud rates, a specific clock frequency is sometimes specified to generate another clock required in the system. Higher clock frequencies increase processing power, power consumption, and system cost in general. System costs increase with frequency because not only does the MCU cost more, but so do all the support chips required, such as RAMs, ROMs, PLDs (programmable logic devices), and bus drivers.[15]

c) Questions related to MCU interrupts

Examining the interrupt structure is a necessity when constructing a real-time system. The designer should look at:

How many interrupt lines or levels are there versus how many does your system require?

The project we are designing has a lot of element that works in fast great condition when it is as an interrupt element such as the reset button as the first priority in the electronic board then start firing button that allows the firing to start, stop firing as well is needed to stop. It when the counting to start

firing start, it's optional to stop it using an interrupt button as well, switching between modes can be very necessary working on local and remote mode having it as an interrupt switch could help a lot.

II.1.2.6 Finalizing the Selection

This Table II-2 listing each MCU under consideration on one axis and the important attributes on the other axis. After a considerable look at the MUCs available in the market, the content of the table is what we end up with. Those attributes are strictly based on the requirements our project needs:

Table II-2: MCU under consideration

Microcontroller	Max CPU Speed (MHz)	RAM (Bytes)	ROM (Bytes)	Timer(s)	I/O pins	Interrupts pins	Communication Protocols	Price (da)
STM32f411ceu6	100	128K	512K	6(16 bit) 2(32 bit)	36	All digital pin	3 UART, 3 I2C, 5 SPI	2000
STM32f103c8t6	72	20 K	128K	3(16 bit)	37	All digital pin	1 UART, 1 I2C, 2 SPI, 1 CAN	1500
PIC18f452	40	1.5 K	256	3(16 bit) 1(8 bit)	34	8	1 UART, 1 SPI, 1 I2C	1200
PIC16f877a	20	368	256	1(16 bit) 2(8 bit)	33	4	1 UART, 1 SPI, 1 I2C	1000

II.1.2.7 Make a final decision

Looking at **STM32f411ceu6** and **STM32f103c8t6** those two are very powerful microcontrollers with fast CPU clocks large RAM and ROM, various timers, and a bunch of different Interrupts pins, and different Communication Protocols. on the other hand, **PIC16f877a** is a less powerful microcontroller in the CPU clock speed less RAM and ROM capacity, and fewer Interrupts pins however, the **PIC18f452** is exactly what fits our project the CPU clock speed can easily hand the task of the project ram and ROM are perfectly fine with the data handling program loading, timers are enough, the number of communication protocol this PIC fit well in the board.

PIC18f452

This powerful 10 MIPS (100 nanosecond instruction execution) yet easy-to-program (only 77 single word instructions) CMOS FLASH based 8-bit microcontroller packs Microchip's powerful PIC® architecture into an 40 pin package and is upwards compatible with the PIC16C5X, PIC12CXXX, PIC16CXX and PIC17CXX devices and thus providing a seamless migration path of software code to higher levels of hardware integration. The PIC18F452 features a 'C' compiler friendly development environment, 256 bytes of EEPROM, Self-programming, an ICD, 2 capture/compare/PWM functions, 8 channels of 10-bit Analog-to Digital (A/D) converter, the synchronous serial port can be configured as either 3-wire Serial Peripheral Interface (SPI™) or the 2-wire Inter-Integrated Circuit (I²C™) bus and

Addressable Universal Asynchronous Receiver Transmitter (AUSART). All of these features make it ideal for manufacturing equipment, instrumentation and monitoring, data acquisition, power conditioning, environmental monitoring, telecom and consumer audio/video applications.[16]



Figure II-9: PIC18F452 [16]

II.1.3 Hardware design

In this section, we will present the full-on architecture of the electronic board. Going in from part to part each block of instruction has its own demonstration, in addition we will add on the flowchart of the program carried out in the PIC microcontroller.

II.1.3.1 KiCad

KiCad is an electronics design automation (EDA) package that is free and open-source. It includes schematic capture, integrated circuit simulation, PCB layout, 3D rendering, and plotting/data export to a variety of formats. KiCad also comes with a comprehensive component library that comprises hundreds of symbols, footprints, and 3D models. KiCad operates on Linux, Windows, and macOS and has minimum system requirements.[17]

KiCad 6.0 is the most recent major release. It includes hundreds of new features and bug fixes.

This open-source program will hold our schematic and the PCB design



Figure II 10: KidCad logo[17]

II.1.3.2 Circuit board architect

The following figures demonstrates the different blocks our board consists of.

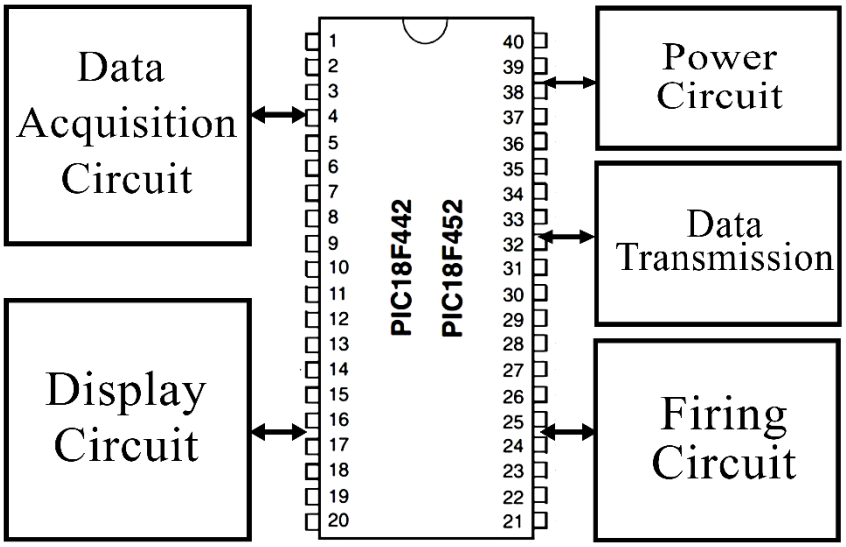


Figure II-11: Different blocks circuit of the electronic board

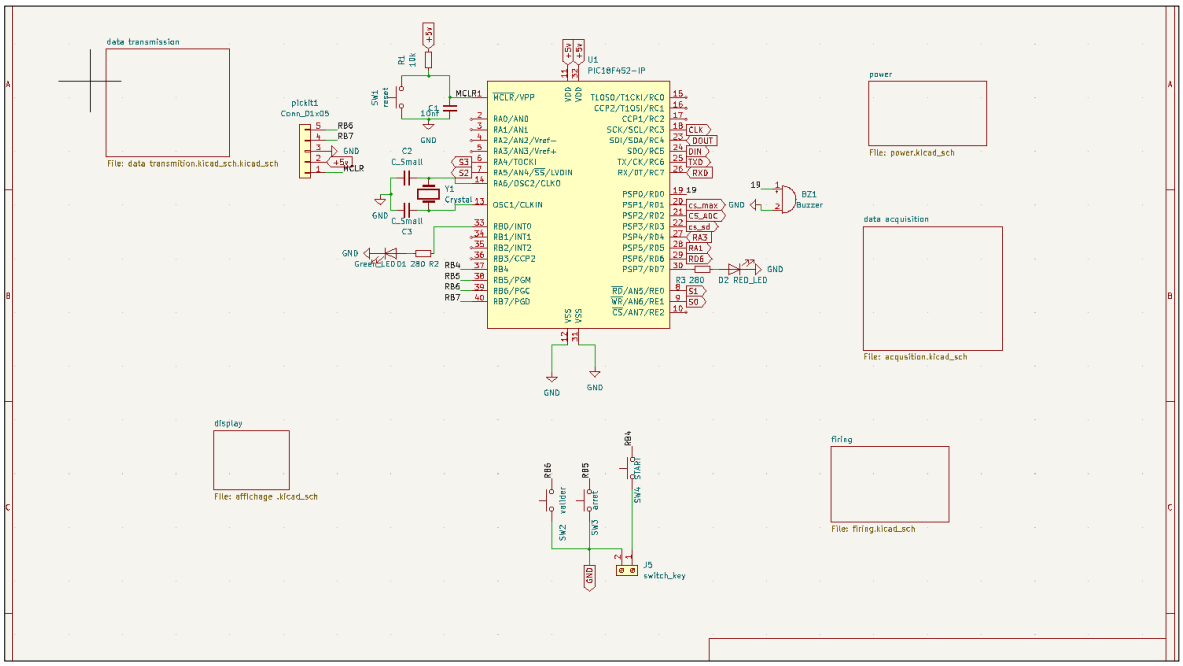


Figure II-12: Different blocks circuit of the electronic board on KiCad software

a) Power supply block circuit

Each component must be designed before a power supply can be built. In this block, we are going to demonstrate what the circuit of a 5V power supply consists of. The (Figure II-9) shows the power supply circuit we are about to use.

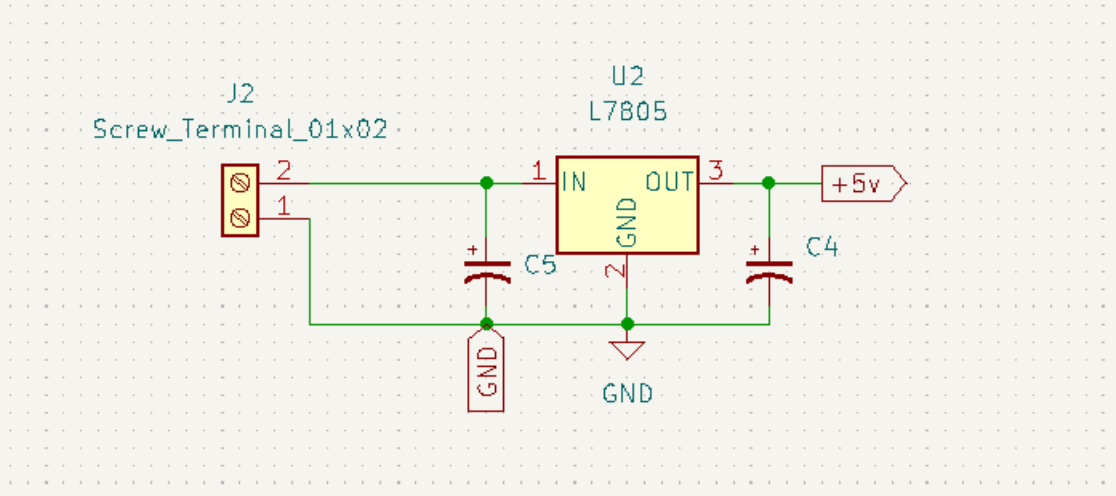


Figure II-13: 5 Volte Power supply block circuit

To begin with, we are going to use a 9V battery (Figure II-10). This 9V battery helps power up the board, which then goes to a voltage regulator long with some capacitors in order to remove any residual noise or ripples. As we require a 5V we need LM7805 Voltage Regulator IC.



Figure II-14: 9V battery

The LM7805 monolithic 3-terminal positive voltage regulator employs internal current limiting, thermal shutdown, and safe-area compensation, making them essentially indestructible. If adequate heat sinking is provided, they can deliver over the 1.5-A output current. They are intended as fixed voltage regulators in a wide range of applications including local (on- board) regulation for the elimination of noise and distribution problems associated with single-point regulation. In addition, to use as fixed voltage regulators, these devices can be used with external components to obtain adjustable output voltages and currents.

LM7805 Feature

- **Input voltage range 7V- 35V**
- **Current rating Ic = 1A**

- **Output voltage range** V_{MAX}**= 5.2V,** V_{MIN}**= 4.8V**
- **Current Limiting Feature:** Each device has an internal current limit to prevent it from damage in the event of a short circuit at the regulator's output. The LM7805 has a typical current limit of 2.4 A.

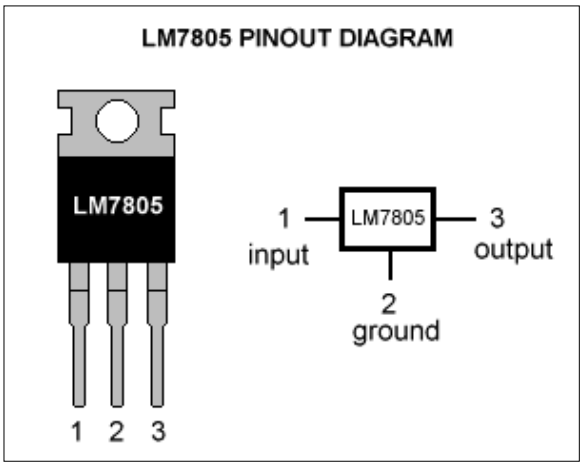


Figure II-15: LM7805 – Pin Diagram

b) Microcontroller connection

This block circuit consists of the main processing unit controlled we have selected precisely **PIC18F452**. This microcontroller is connected to all 5 block circuits of the board and works on dealing with all input/output coming out of and in of the blocks, the (Figure II-11) shows the **PIC18F452** microcontroller alongside all of its connection pins.

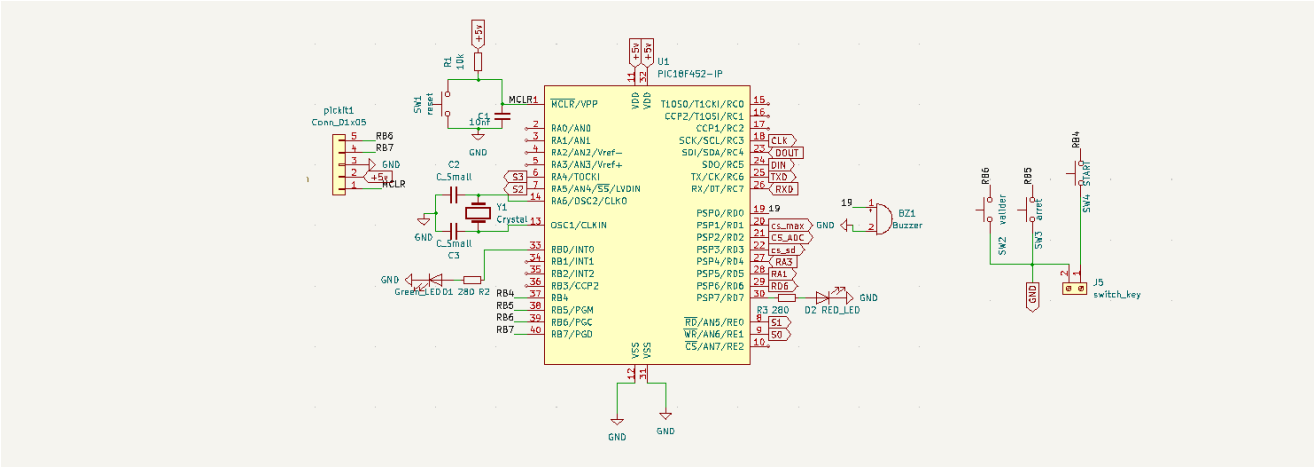


Figure II-16:PIC18f452 microcontroller pins and all its connection

Right on the first pin, we will have the MCLR1 pin, which is the reset pin. This last one is connected to a button to reset the microcontroller in case of having a long-lasted infinity loop or some errors.

Jumping on the pins RA6 and RA7 which are oscillator pins OSC2 and OSC1 pins for the external crystal clock, in this case, we used a 16 MHz clock rate crystal oscillator and resistor-capacitor (RC). the reason behind picking this exact clock rate is mainly the UART Baud rates, according to the baud rates for the asynchronous mode table in the PIC18F452 datasheet, to lower data transfer error using the

UART module with the 9.2 Kbps transfer baud rates it's highly recommended to use a 16 Mhz frequency oscillator to reduce error down to +1.73%

As embedded devices become more capable and connected, the necessity for remote application updates grows. Embedded connection, which isn't confined to a single communication protocol, comes in a variety of shapes and sizes and is critical in emerging sectors such as the Internet of Things (IoT). Remote firmware updates make it easy to upgrade an existing program while also extending the product's useful life. Fresh features can be added to a product to preserve a competitive advantage versus new market offerings or to unlock existing fee-based premium content. Furthermore, software fixes can be swiftly disseminated to guarantee that the system runs smoothly. [18]

For example, you can take it out of the slut and reprogram it using an outside programmer as shown in the (Figure II-17) or, the easiest way to do so is by loading a bootloader into the microcontroller.



Figure II-17: Minipro programmer for all PIC microcontrollers

To take advantage of this connectivity, bootloader firmware must exist in memory, allowing the microcontroller to self-program (MCU).

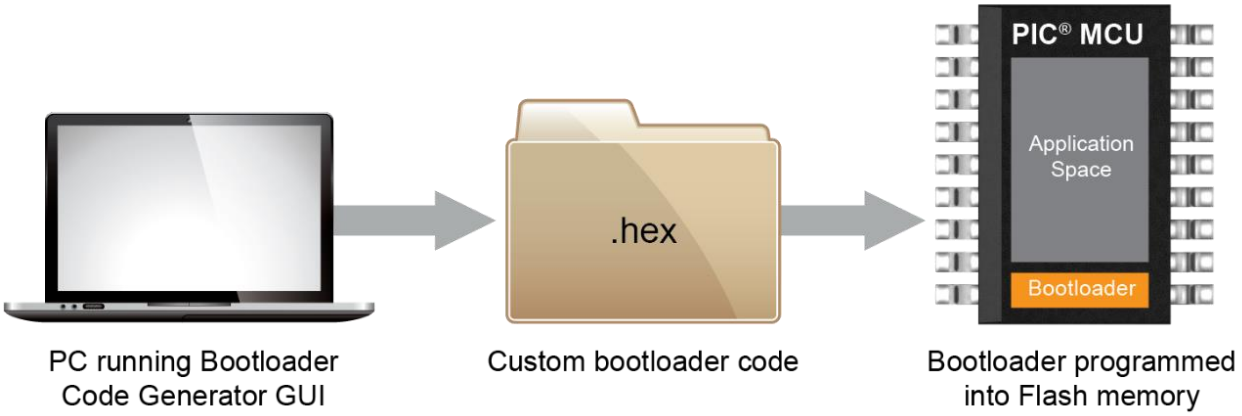


Figure II-18: Loading a Bootloader into a MCU

Microchip has also offered a Bootloader Host program to go along with the Bootloader Generator. This tool is used to transfer fresh application code to the embedded device's bootloader code. This GUI application supports UART, I2C, USB, and Ethernet protocols for communication with the target device. The Bootloader Host application can program the Flash memory of 8-bit PIC® MCUs and can also write to EEPROM.[18]

In our project we use a 16 Mhz external clock, the chance of having the same exact bootloader for this clock speed is not a choice due to the unavailability of this kind of bootloader. so the alternative solution is the following:

Microcontroller In-Circuit Serial Programming (ICSP) with Microchip PIC

In Circuit Serial Programming is a technique for programming a Microchip PIC while it is attached to a circuit, rather than programming the chip first and then connecting it to the circuit. ICSP has numerous advantages, but it also has some critical design issues, which I will try to emphasize.

In circuit programming has many advantages:

- It shortens the development cycle – it is really annoying to take a chip out of the board each time it needs to be reprogrammed, and it’s even much harder with SMD.
- Allows customer and in field firmware upgrades
- Calibrate the system during manufacturing or in the field
- Assign unique ID \ serial number to product

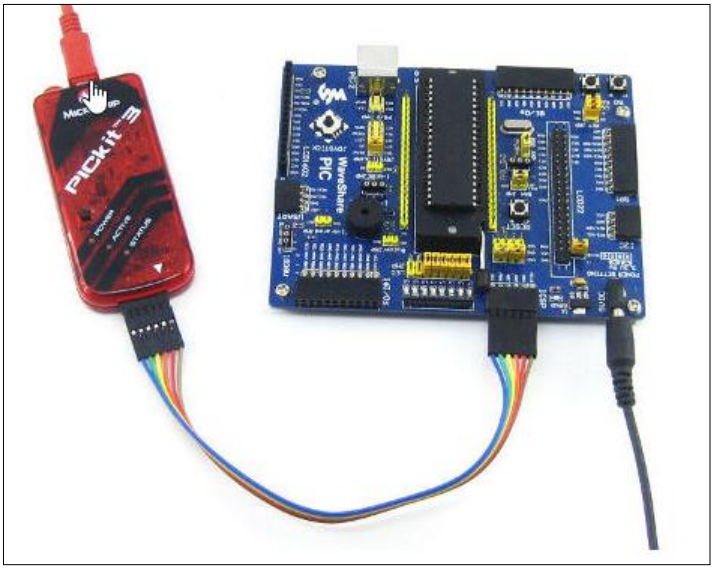


Figure II-19: Microcontroller in Circuit Serial Programming (ICSP) with pickit 3

How to prepare your circuit to work with ICSP?

In Circuit Serial Programming is a technique for programming a Microchip PIC while it is attached to a circuit, rather than programming the chip first and then connecting it to the circuit. ICSP has numerous advantages, but it also has some critical design issues, which I will try to emphasize.[19]

Use dedicated ICSP pins whenever the application allows it. It will save you a lot of time and effort. By not sharing a pin for both ICSP and I/O, for example, the amount of preparation work required to enable ICSP is reduced. Often, and especially true with low pin devices, it is not possible to dedicate the 3 needed pins just for ICSP, and when want them to have dual functionality.[19]

In this case:

1. Use a Schotkey diode and an R/C to isolate Vpp from the circuit.

2. Isolate the ICSP Clock and ICSP Data circuits from the rest of the system. Because the isolation approach is application specific, there is no ready-to-use formula. A resistive isolation is frequently sufficient. 1k to 10k resistor values are recommended.

3. To reduce attenuation, place the ICSP header as close as feasible to the programmed chip.

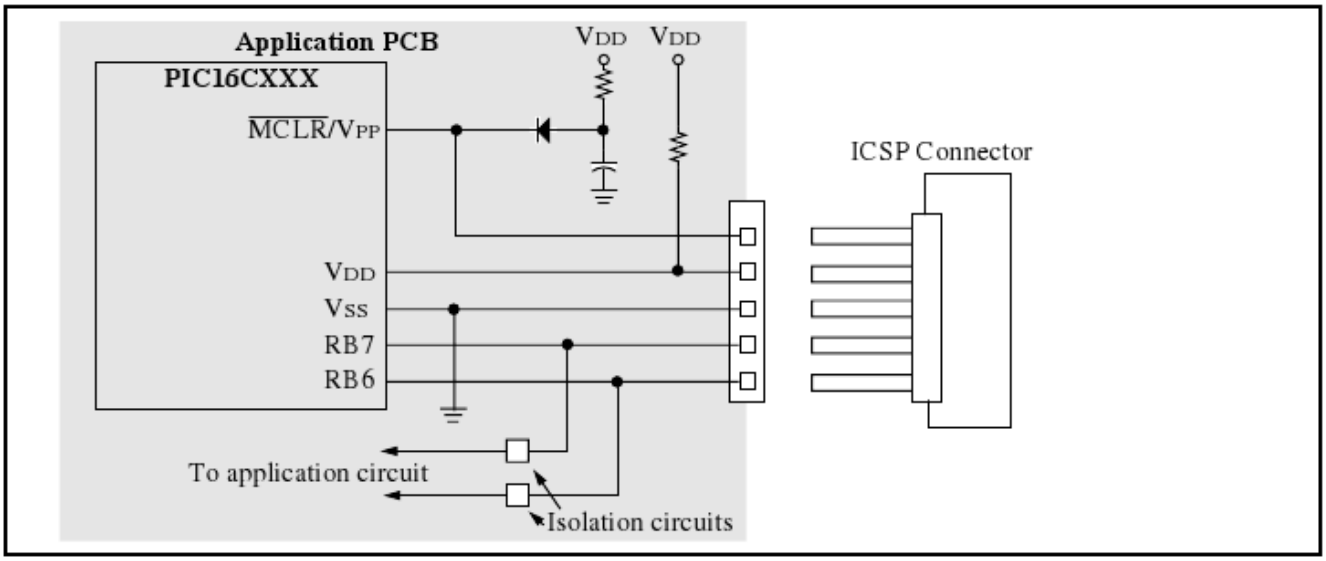


Figure II-20: ICSP connection with pic microcontroller. [19]

In addition to the other connections, the **RB0** and **RD7** are for the green and red LEDs (Light-emitting diode), the green one indicates everything is good for firing and the red indicates an error occurring on the step of current testing around the igniter.

As for safety and security, it is necessary to have a buzzer when the ignition is happening. Pin 19 **RD0** holds the passive buzzer. The passive buzzer is a type of electromagnetic squeaker that produces sound signals at various frequencies.



Figure II-21: Buzzer passive 5v

3 main buttons are connected to the microcontroller which is START, STOP, and Validation buttons.

The start button starts up the counting down toward the firing, and the stop buttons stop the counting, before the countdown starts you have to validate by clicking on the validation button. This validation button also works as Mode switcher, holding it for a litter bit will change the mode from local mode to remote mode. For security reasons, a switch in form of a Rotary Switch keylock, is a lock to the start ignition button.

Those buttons validation, stop, and start are successively connected to **RB6**, **RB5**, and **RB4**.



Figure II-22: Rotary Switch keylock

c) Firing circuit

In order to read the data from sensors that are placed around the thruster, we should first make sure that the thruster was fired up and that can be done by giving it the firing conditions O_2, border, and an electric signal which is what our firing circuit will provide. The following circuit shows the simplest way to generate this electric signal:

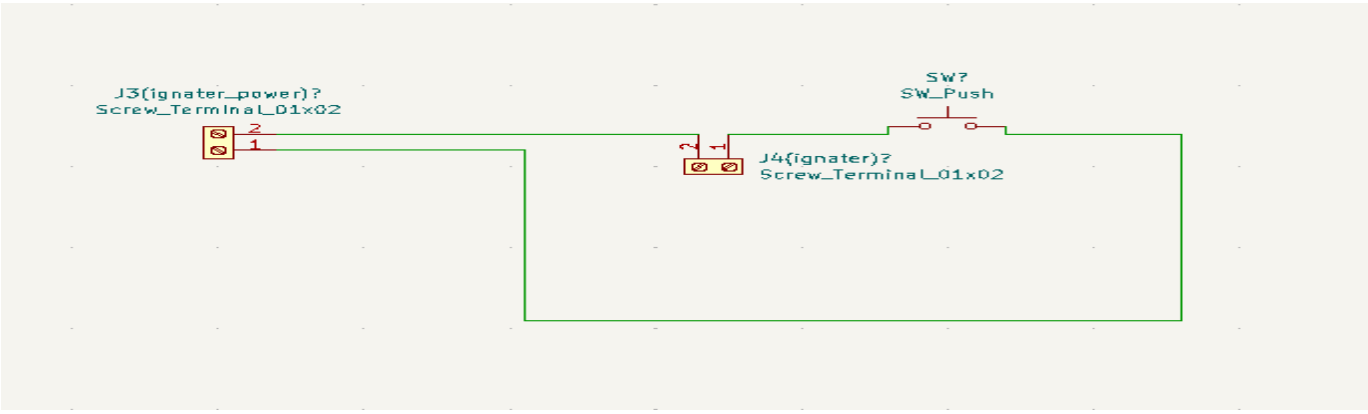


Figure II-23: Simple firing circuit

The 9A of generated by the battery will be flowing through the wires when pressing the button, which is going to fire up the igniter. And that is because there is no resistance put in series with the igniter.

However, the thing that is missing here is that we want to control the circuit automatically and not with a button, one of the solutions was to add a transistor to close or open the circuit as shown in the following image:

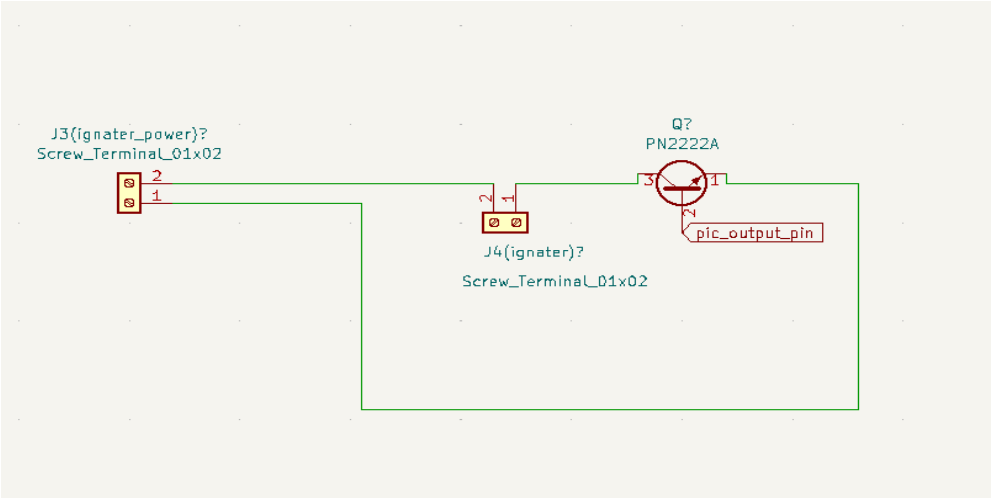


Figure II-24: Firing circuit using transistor

However, adding a transistor this way will not allow enough current to flow throw wires. So the better way is to add a rely which will be feeding with 0.060A from the transistor. Here is how it going to be.

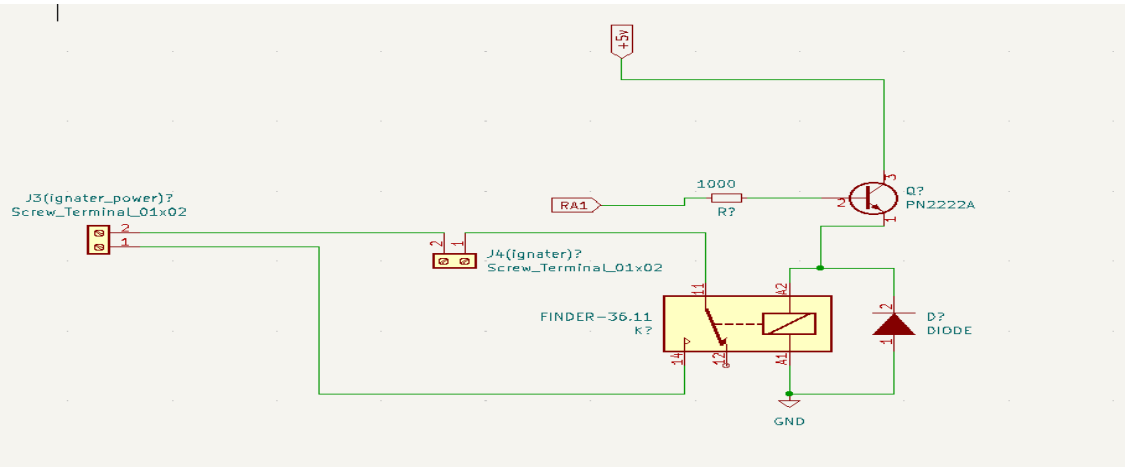


Figure II-25: Firing circuit using transistor and a relay

Another benefit of using the rely, is to separate the igniter Ground from the microcontroller Ground.

And the reason why adding the diode in parallel whit the coil is to discharge the coil when the current suddenly stopped so the high current going out from the coil will not damage any component.

Until now the circuit works perfectly, but we may want to use it for more than firing the igniter,

Testing the connectivity of the igniter whether it is working or damaged in the first place, so we don't lose time waiting for counting down to be finished and figuring out that the igniter is damaged.

For so, we used the normally closed relay pin to wire it whit an optocoupler which will give our microcontroller a 0 digital signal if it is perfectly working or 1 digital signal in the opposite case.

And of course, adding the optocoupler would mean less if we do not prevent the 9A of the current firing of the igniter. Cause, remember we want to test it not firing it so adding a 500hom-200hom resistor in series with the optocoupler is necessary.

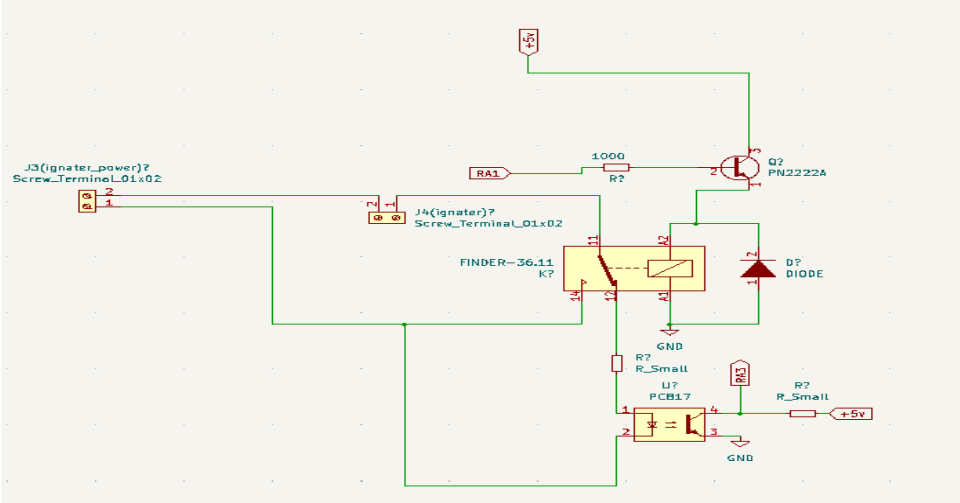


Figure II-26: Firing circuit using a transistor, relay, and optocoupler for testing

Now the last modification we should add is to control the testing circuit, if you at it you will notes that the circuit is always closed. Which means it always consumes the energy of the battery, to prevent that, we preferred to add another optocoupler as a switch controlled by the microcontroller.

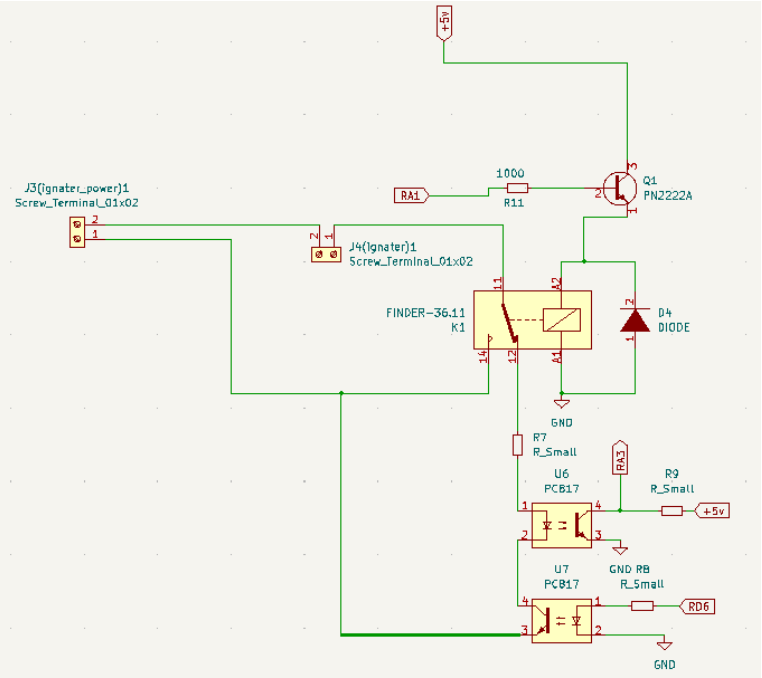


Figure II-27: Final firing circuit

d) Display circuit

Before launching firing, and burning the igniter, and for safety and security reasons it is necessary to have a count down before the firing happens. All that is due to the display circuit, the following circuit uses the MAX7219 module with a 4 of 7 segment display to help show the count down.

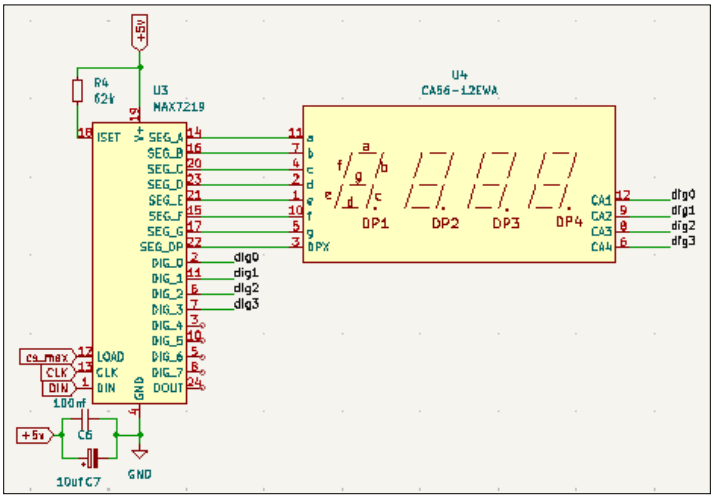


Figure II-28: Display circuit

The MAX7219 is a serial input/output common cathode display driver that connects microprocessors to 7 segment numeric LED displays with up to 8 digits, bar-graph displays, or 64 individual LEDs. A

BCD code-B decoder, multiplex scan circuitry, segment and digit drivers, and an 8x8 static RAM that saves each digit are all included in the device. To set the segment current for all LEDs, only one external resistor is necessary. The MAX721 is SPI (Serial Peripheral Interface) compatible.[20]

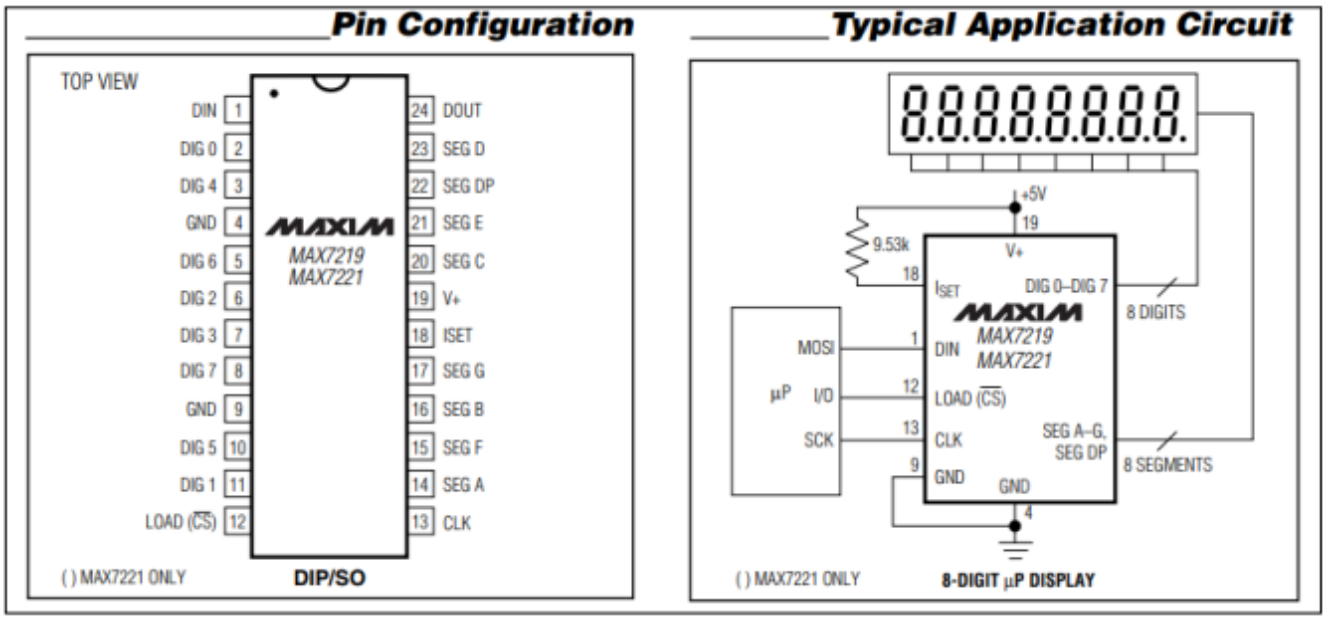


Figure II-29: Pin Configuration and Typical Application Circuit of the MAX7219[20]

The display brightness of the MAX7219 may be regulated by connecting an external resistor (RSET) between V+ and ISET. The peak current drawn by the segment drivers is 100 times the current drawn by ISET. To allow brightness modification from the front panel, this resistor can be fixed or variable. Its minimum value should be 9.53k, which sets the segment current to 40mA in most cases. The intensity register can also be used to alter the display brightness digitally. [20]

Introduction to SPI communication

SPI is a widely used communication protocol that is utilized by a variety of devices. SPI is used to connect with microcontrollers by SD card reader modules, RFID card reader modules, and 2.4 GHz wireless transmitter/receivers, for example.[21]

SPI-connected devices are in a master-slave relationship. The master (typically a microcontroller) is the controlling device, while the slave (generally a sensor, display, or memory chip) receives instructions from the master. A single master, single slave system is the most basic configuration of SPI, yet one master can govern several slaves (more on this below).[21]

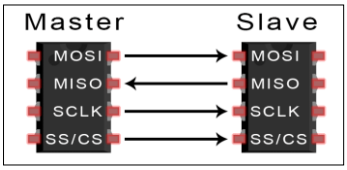


Figure II-30: SPI master-slave relationship

MOSI (Master Output/Slave Input) – Line for the master to send data to the slave.

MISO (Master Input/Slave Output) – Line for the slave to send data to the master.

SCLK (Clock) – Line for the clock signal.

SS/CS (Slave Select/Chip Select) – Line for the master to select which slave to send data to.

In our project, the **DIN** in the MAX7219 is connected to the **PIN 24** which is an SDO SPI pin on the microcontroller, the **CLK** is connected to the **PIN 18** which is an SCK SPI pin on the microcontroller, the **LOAD** is connected to the **PIN 20** which is an RD1 on the microcontroller that works as the Slave Select/Chip Select.

The use of MISO (Master Input/Slave Output) is not necessary in this case of MAX7219, because that is no data transfer from the max to the MCU.

e) Data acquisition

As described in the system architecture, for several reasons we chose the third proposition (d) as a proposition for the data acquisition system. This data architecture structure is as follows.

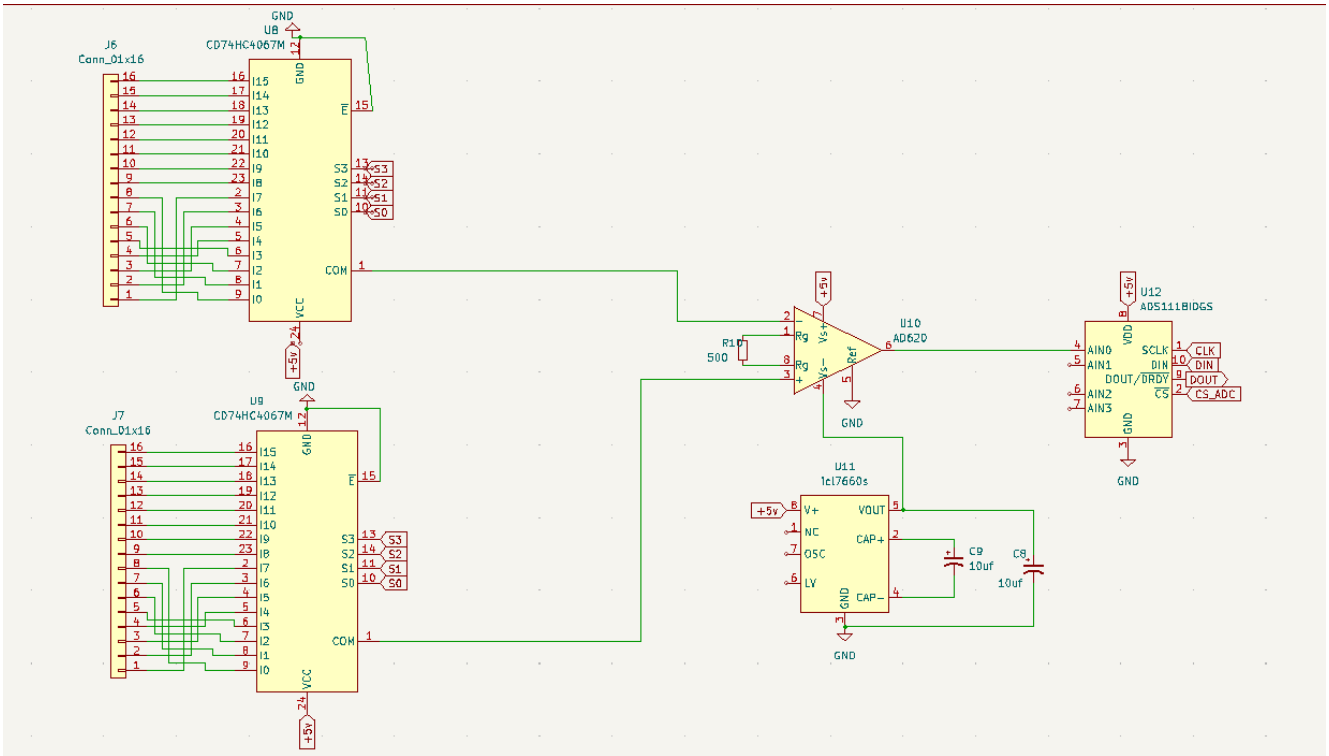


Figure II-31: Data acquisition circuit

The data acquisition system contains 2 analog multiplexers going in right into an amplifier; the output of this last one is connected straight to the ADC responsible for translating the analog output into a digital one.

Multiplexers

The **74HC4067** is a digitally controlled analog switch that uses silicon-gate CMOS technology to provide LSTTL-like operating speeds with typical CMOS integrated circuits' low power consumption.

These analog multiplexers/demultiplexers regulate analog voltages that change depending on the voltage supply. Because they are bidirectional switches, they may be used to convert any analog input to an output and vice versa. Minimal "on" resistance and low "off" leakages characterize the switches. Furthermore, these devices contain an enable control that, when set to high, disables all switches and sets them to "off."[22]

Features

- Wide Analog Input Voltage Range
- Fast Switching and Propagation Speeds
- Wide Operating Temperature Range . . . -55℃ to 125℃
- 16 INPUT/OUTPUT chain
- Balanced Propagation Delay and Transition Times

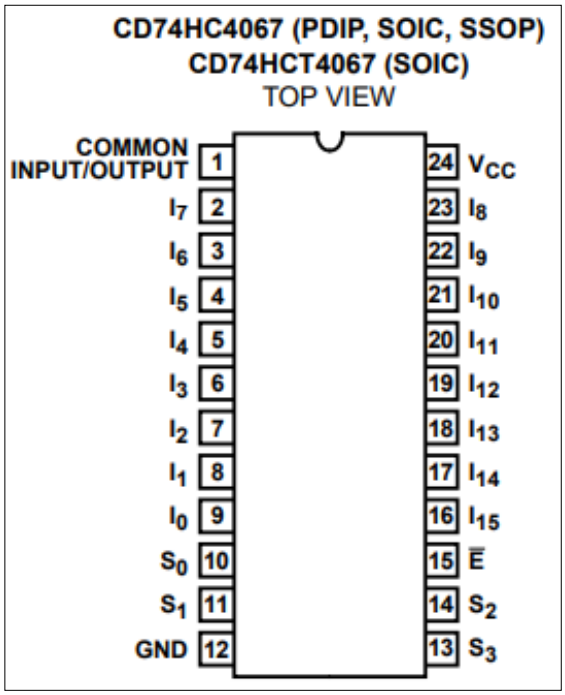


Figure II-32: 74HC4067 Pinout (instrument)

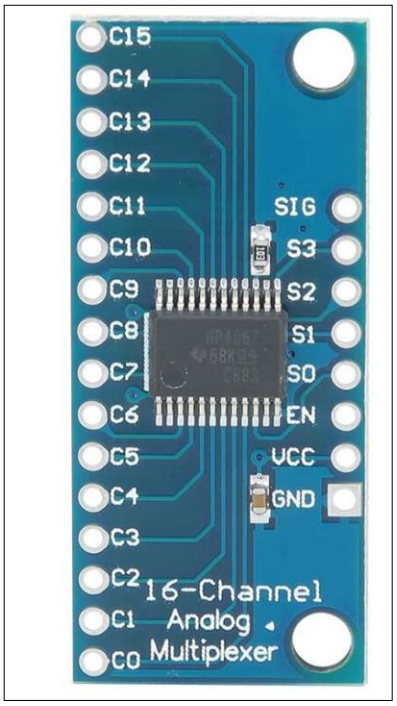


Figure II-33: 74HC4067 module

The following pins (**C0 to C15**) are 16 chan's all reserved to the differential sensor inputs, pins **(S0 to S3)** are all selective pins allow the microcontroller to select from all 16 different channels, **EN/COM** Pin the enable pin is low activated that why it’s connected to the ground, **VCC** and **GND**, **SIG/COM** wich is the output pin going in right to the ADC module.

Amplifier

The AD620 is a low-cost, high-accuracy instrumentation amplifier with only one external resistor required to establish gains ranging from 1 to 10,000. The AD620 also has an 8-lead SOIC and DIP

package that is smaller than discrete designs and consumes less power (just 1.3 mA max supply current), making it ideal for battery-powered (or remote) applications.

The AD620 is suited for use in precision data acquisition systems, such as weigh scales and transducer interfaces, because to its high accuracy of 40 ppm maximum nonlinearity, low offset voltage of 50 V max, and offset drift of 0.6 V/°C max. In addition, the input bias current is low and noise is minimal.[23]

Features

- Gain set with one external resistor (Gain range 1 to 10,000)
- Wide power supply range (±2.3 V to ±18 V)
- Low power, 1.3 mA max supply current
- Low noise 9 nV/√Hz @ 1 kHz, input voltage noise 0.28 µV p-p noise (0.1 Hz to 10 Hz)
- Excellent ac specifications 120 kHz bandwidth (G = 100)

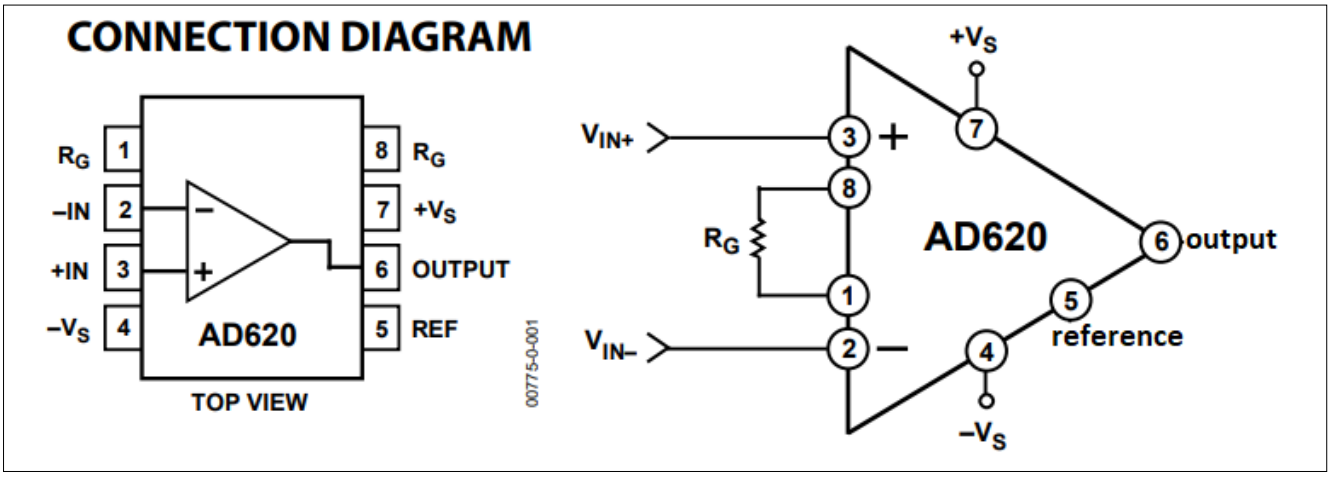


Figure II-34: AD620 circuit connection diagram [23]

Gain selection

The AD620 gain is resistor-programmed by RG, or more precisely, by whatever impedance appears between Pins 1 and 8. The AD620 is designed to offer accurate gains using 0.1% to 1% resistors. Note that for G = 1, the RG pins are unconnected (RG = ∞). For any arbitrary gain, RG can be calculated by using the formula: [23]

$$R_G = \frac{49.4k\Omega}{G-1}$$

Instrumentation Amplifier Diamond Plot Tool

Instrumentation amplifier datasheets typically show a graph (or several variations) of the Output Swing vs Input Common-Mode Voltage, also known as the Diamond Plot, which is a comprehensive graph of all external and internal headroom limits. The Diamond Plot can be changed for varied Supply Voltages, Gains, and Reference Pin Voltages if enough fundamental information about the circuit is available. Although this adjusted plot is useful information in and of itself, the actual usefulness comes from assessing whether the signal of interest is inside the Diamond Plot limitations. This necessitates

the knowledge of the input signal. The range of the input signal is plotted directly on the Diamond Plot, and any errors where the input signal crosses the Diamond Plot borders are displayed on the plot and reported by the tool.[23]

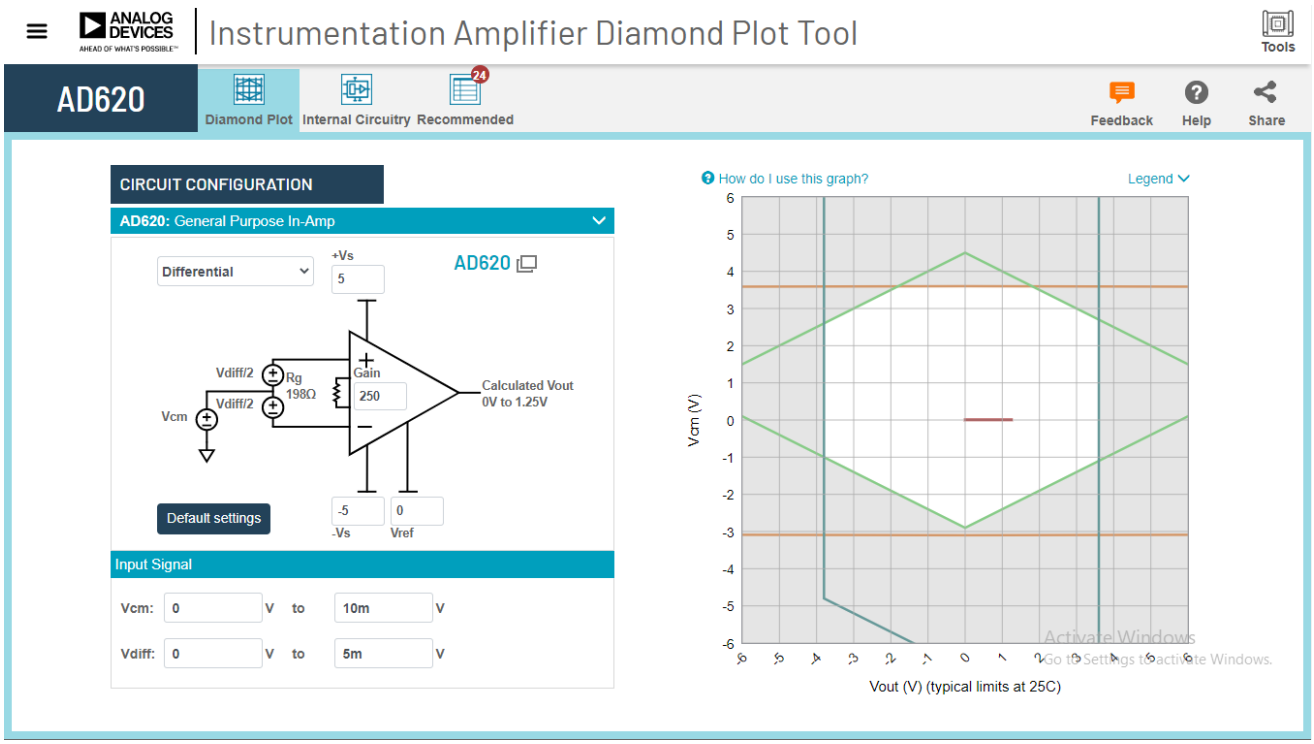


Figure II-35: Instrumentation Amplifier Diamond Plot Tool

Setting up the Circuit Configuration Panel, choosing a product from the drop-down list in our case the AD620, and entering the circuit conditions, such as supply voltage, reference pin voltage, and gain. Enter. Voltage supply, we chose 5volt +Vs and -5volt -Vs, reference pin voltage at the ground, and the 250 gain.[23]

The input signal range, including any expected variations due to component tolerance and common-mode or differential offsets that must be accounted for in the design. Most force sensors/temperature sensors have a Rated output of 2mV/V±10%. In this study case, we will use an estimated measure of the 0-2mV Vdff and 0-5mV Vcm. The calculated output of the AD620 is around 0V to 1.25V differential voltage.[23]

To generate the -5 volts necessary for the ad620 to work it is easy and necessary to have a separated IC (integrated circuit) to do so. In our cases, the **icl7660** fits this requirement perfectly.

The ICL7660S Super Voltage Converters are monolithic CMOS voltage conversion ICs that outperform similar devices. They're direct replacements for the industry-standard ICL7660, with a higher operational supply voltage range of up to 12V and reduced supply current. A Frequency Boost pin has been added to allow users to achieve reduced output impedance even with smaller capacitors. The ICL7660S converts positive to negative supply voltages for an input range of 1.5V to 12V, yielding

complementary output voltages of -1.5V to -12V. For the charge pump and charge reservoir functions, just two noncritical external capacitors are required.[24]

Pin Configurations

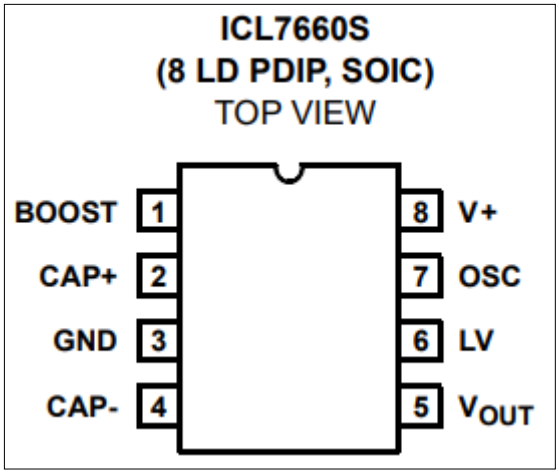


Figure II-36: Pin Configurations for the ICL7660S

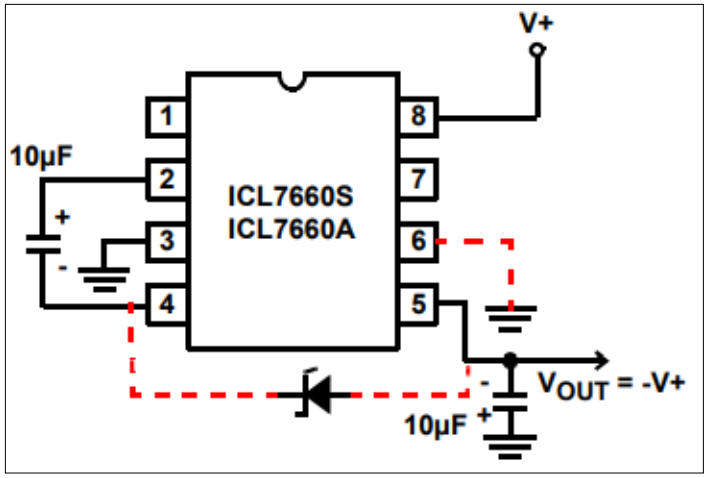


Figure II-37: Simple negative converter and its

Simple Negative Voltage Converter

The ICL7660S will mostly certainly be used to generate negative supply voltages in the vast majority of applications. Where a positive supply of +1.5V to +12V is available, the Figure II-37 displays conventional connections to generate a negative supply. Remember that for supply voltages below 3.5V, pin 6 (LV) is connected to the supply negative (GND).[24]

The Vout is directly connected to the **PIN4** (-Vs) of AD620.

The amplified output of the AD620 is connected to an ADC (analog to digital converter) which is ADS1118.

The ADS1118 Ultra-Small Analog-to-Digital Converter (ADC) from Texas Instruments has a resolution of 16 bits and is designed for precision, power, and ease of use. The ADS1118 uses a single power supply that ranges from 2 to 5.5 volts and can convert data at up to 860 samples per second (SPS).[10]

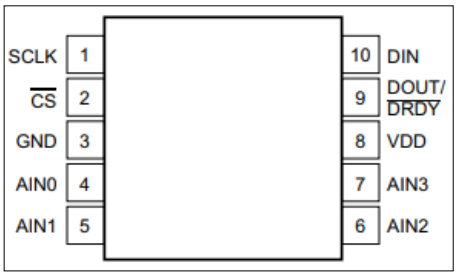


Figure II-38: Pin Configuration for the ADS1118

Table II -3: Pin Functions of ADS1118

PIN		TYPE	DESCRIPTION
NO.	**NAME**		
1	SCLK	Digital input	Serial clock input
2	$\overline{CS}$	Digital input	Chip select; active low. Connect to GND if not used.
3	GND	Supply	Ground
4	AIN0	Analog input	Analog input 0. Leave unconnected or tie to VDD if not used.
5	AIN1	Analog input	Analog input 1. Leave unconnected or tie to VDD if not used.
6	AIN2	Analog input	Analog input 2. Leave unconnected or tie to VDD if not used
7	AIN3	Analog input	Analog input 3. Leave unconnected or tie to VDD if not used.
8	VDD	Supply	Power supply. Connect a 100-nF power supply decoupling capacitor to GND.
9	DOUT/$\overline{DRDY}$	Digital output	Serial data output combined with data ready; active low
10	DIN	Digital input	Serial data input

The ADS1118 ADC core measures a differential signal, V_{IN} that is the difference of $V_{(\mathrm{AINP})}$ and $V_{(\mathrm{AINN})}$. The converter core consists of a differential, switched-capacitor ΔΣ modulator followed by a digital filter. This architecture results in a very strong attenuation in any common-mode signals. Input signals are compared to the internal voltage reference. The digital filter receives a high-speed bitstream from the modulator and outputs a code proportional to the input voltage.[10]

The ADS1118 has two available conversion modes: single-shot mode and continuous conversion mode. In single-shot mode, the ADC performs one conversion of the input signal upon request and stores the value to an internal conversion register. The device then enters a power-down state. This mode is intended to provide significant power savings in systems that require only periodic conversions or when there are long idle periods between conversions. In continuous conversion mode, the ADC automatically begins a conversion of the input signal as soon as the previous conversion is completed. The rate of continuous conversion is equal to the programmed data rate. Data can be read at any time and always reflect the most recently completed conversion.[10]

Feature Description

As illustrated in Figure II-39, the ADS1118 has an input multiplexer (mux). It is possible to measure either four single-ended or two differential signals. AIN0, AIN1, and AIN2 can also be measured differently from AIN3. Bits MUX [2:0] in the Config register are used to control the multiplexer. When measuring single-ended signals, a switch inside the multiplexer connects the ADC's negative input to GND.[10]

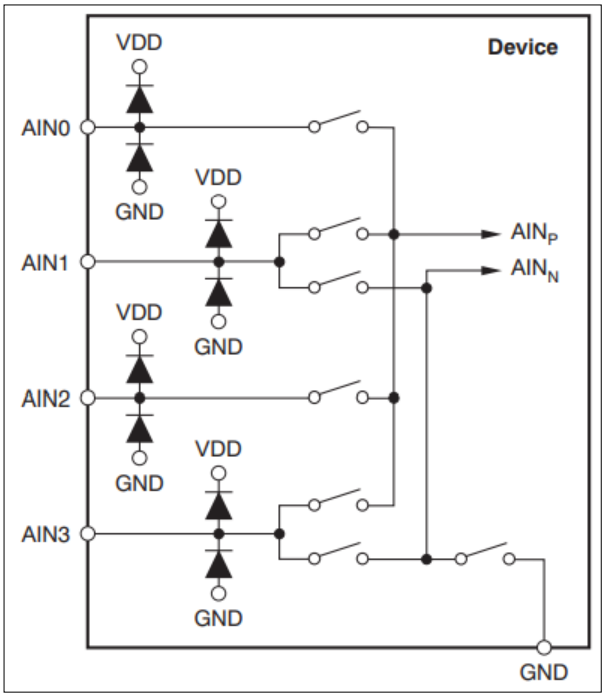


Figure II-39: Input Multiplexer. [22]

Programming

- **Serial Interface:** The SPI-compatible serial interface consists of either four signals (CS, SCLK, DIN, and DOUT/DRDY), or three signals (in which case CS may be tied low). The interface is used to read conversion data, read and write registers, and control device operation.[10]
-
- **Chip Select ($\overline{CS}$):** The chip select pin (CS) selects the ADS1118 for SPI communication. This feature is useful when multiple devices share the same serial bus. Keep CS low for the duration of the serial communication. When CS is taken high, the serial interface is reset, SCLK is ignored, and DOUT/DRDY enters a high-impedance state. In this state, DOUT/DRDY cannot provide data-ready indication. In situations where multiple devices are present and DOUT/DRDY must be monitored, lower CS periodically. At this point, the DOUT/DRDY pin either immediately goes high to indicate that no new data are available, or immediately goes low to indicate that new data are present in the Conversion register and are available for transfer. New data can be transferred at any time without concern of data corruption. When a transmission starts, the current result is locked into the output shift register and does not change until the communication completes. This system avoids any possibility of data corruption.[10]
- **Serial Clock (SCLK):** features a Schmitt-triggered input and is used to clock data on the DIN and DOUT/DRDY pins into and out of the ADS1118. Even though the input has hysteresis, TI recommends keeping SCLK as clean as possible to prevent glitches from accidentally shifting the data. If SCLK is held low for 28 ms, the serial interface resets and the next SCLK pulse starts a new communication cycle. This time-out feature can be used to recover communication when a serial interface transmission is interrupted. When the serial interface is idle, hold SCLK low.[10]

- **Data Input (DIN):** The data input pin (DIN) is used along with SCLK to send data to the ADS1118. The device latches data on DIN on the SCLK falling edge. The ADS1118 never drives the DIN pin. [10]
- **Data Output and Data Ready (DOUT/ $\overline{DRDY}$):** The data output and data-ready pin (DOUT/$\overline{DRDY}$) is used with SCLK to read conversion and register data from the ADS1118. Data on DOUT/$\overline{DRDY}$ are shifted out on the SCLK rising edge. DOUT/$\overline{DRDY}$is also used to indicate that a conversion is complete and new data are available. This pin transitions low when new data are ready for retrieval. DOUT/$\overline{DRDY}$ is also able to trigger a microcontroller to start reading data from the ADS1118. In continuous-conversion mode, DOUT/$\overline{DRDY}$ transitions high again 8 µs before the next data ready signal (DOUT/$\overline{DRDY}$ low) if no data are retrieved from the device. This transition is shown in FigureII-12. Complete the data transfer before DOUT/$\overline{DRDY}$ returns high. [10]

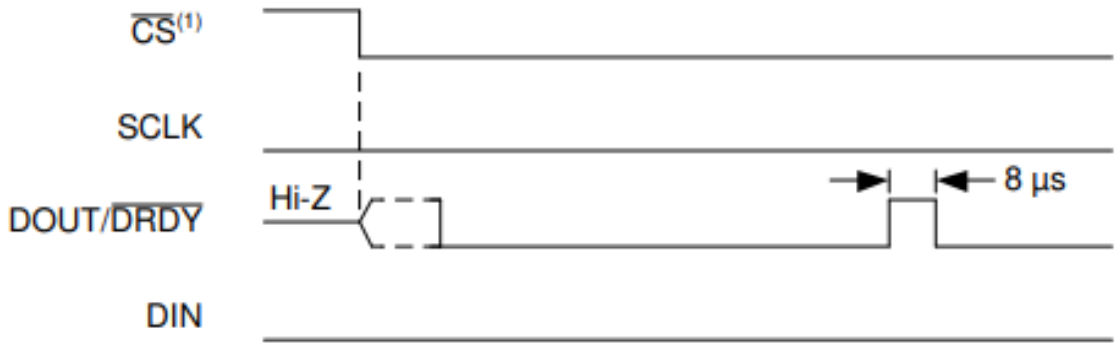


Figure II-40: DOUT/DRDY Behavior without Data Retrieval in Continuous Conversion Mode. [10]

f) Data transmission

Data being acquired has to be transferred or saved depending on the mode. As we described previously this board has 2 modes. Local mode and remote mode.

In the **local mode** the data acquired has to be saved right on the SD card. Using an SD card model this activity works as follows:

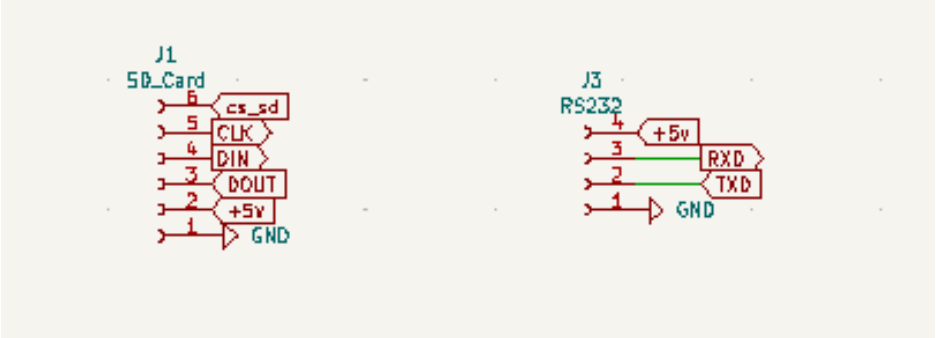


Figure II-41: SD card and RS232 pin connections

Micro SD Card Micro SDHC Mini TF Card Adapter ReaderModule

SD cards, sometimes known as Micro SD cards, are widely utilized in a variety of applications, including data logging, data visualization, and more. Micro SD Card Adapter Modules make it much easier to access these SD cards. The Micro SD Card Adapter module is a simple to use module with an SPI interface and an integrated 3.3V voltage regulator to ensure that the SD card is properly powered.[25]

Features and Specifications of Micro SD Card Adapter Module

- Operating Voltage: 4.5V - 5.5V DC
- Current Requirement: 0.2-200 mA
- 3.3 V on-board Voltage Regulator
- Supports FAT file system
- Supports micro SD up to 2GB
- Supports Micro SDHC up to 32GB

Pin Configuration of Micro SD Card Adapter Module

Power and communication with the microcontroller are provided by six pins on the module. The pin type and function of each pin on the module are described in the table II-4 below.

Table II-4: Pin Configuration of Micro SD [25]

Pin Type	Pin Description
GND	Ground
VCC	Voltage Input
MISO	Master In Slave Out(SPI)
MOSI	Master Out Slave In(SPI)
SCK	Serial Clock(SPI)
CS	Chip Select(SPI)

Connecting Micro SD Card Adapter Module to an MCU/MPU

A Micro SD Card adapter module can be easily connected to the **PIC18F452**. Since the module communicates via the SPI protocol, we need to connect the MISO, MOSI, SCK, and CS of the module to the MCU's. Figure II-13 describes the pin connection between the SD cart module and the pic18f452. The regular SPI pins are all connected alongside the CS enable pin(cs_sd) which is the RD1 in this case. With the SD card holder, the module has a 3.3V voltage regulator, along with a 74LVC125A Level shifter IC.[25]

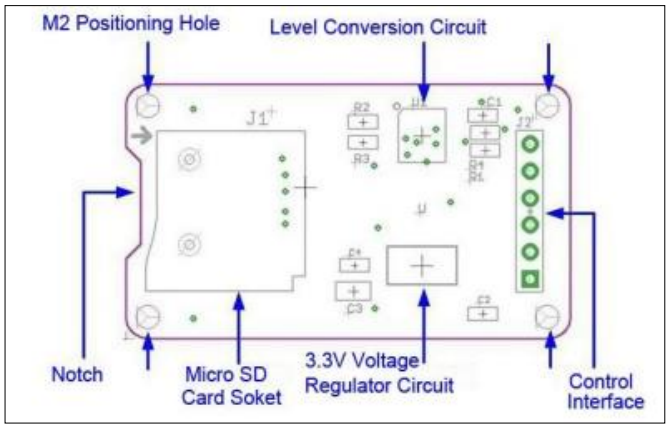


Figure II-42: Mirco SD Card Interface Module

The **remote mode** Transmits and sends the data to the operation station, which in most cases could be a PC (Personal Computer). This activity works as follows:

UART Interfacing with PIC Microcontroller

The term UART stands for Universal Asynchronous Receiver/Transmitter, and it is a commonly used hardware module for serial communication using protocols such as RS 232.

UART is a commonly used hardware module for serial communication using protocols such as RS 232. Asynchronous or synchronous serial communication is possible. In synchronous communication, a common clock will direct both the receiver and the transmitter, which is not the case in asynchronous communication. The most widely used technique of asynchronous serial communication between two devices is the UART. Tx (transmission line) and Rx (receiver line) are used in UART communication (reception line). The UART interface allows for simultaneous transmission and reception of data (full duplex). The inherent USART hardware on PIC microcontrollers (most of them) can be adjusted in three different modes.[26]

1. USART asynchronous (full duplex)
2. USART synchronous master (half duplex)
3. USART synchronous slave (half duplex)

In our cases, we have worked with the transmit and receive data simultaneously (full-duplex) using the UART interface.

Rs-232 Level Converter

RS-232 is a serial communication standard that allows two devices to communicate with each other. A logic 0 has a voltage between +3V and +15V, and a logic 1 has a voltage between -3V and -15V, according to the protocol. Because the voltage levels for logic 0 and logic 1 are 0V and 5V, this voltage level is incompatible with a CMOS or TTL (Transistor-Transistor Logic) logic family device. A voltage level shifter is required to convert RS-232 logic levels to TTL logic levels and vice versa due to the voltage level discrepancy. For this function, the MAX232 is the most often used voltage level shifter IC.[26]

UART Interfacing with PIC Microcontroller

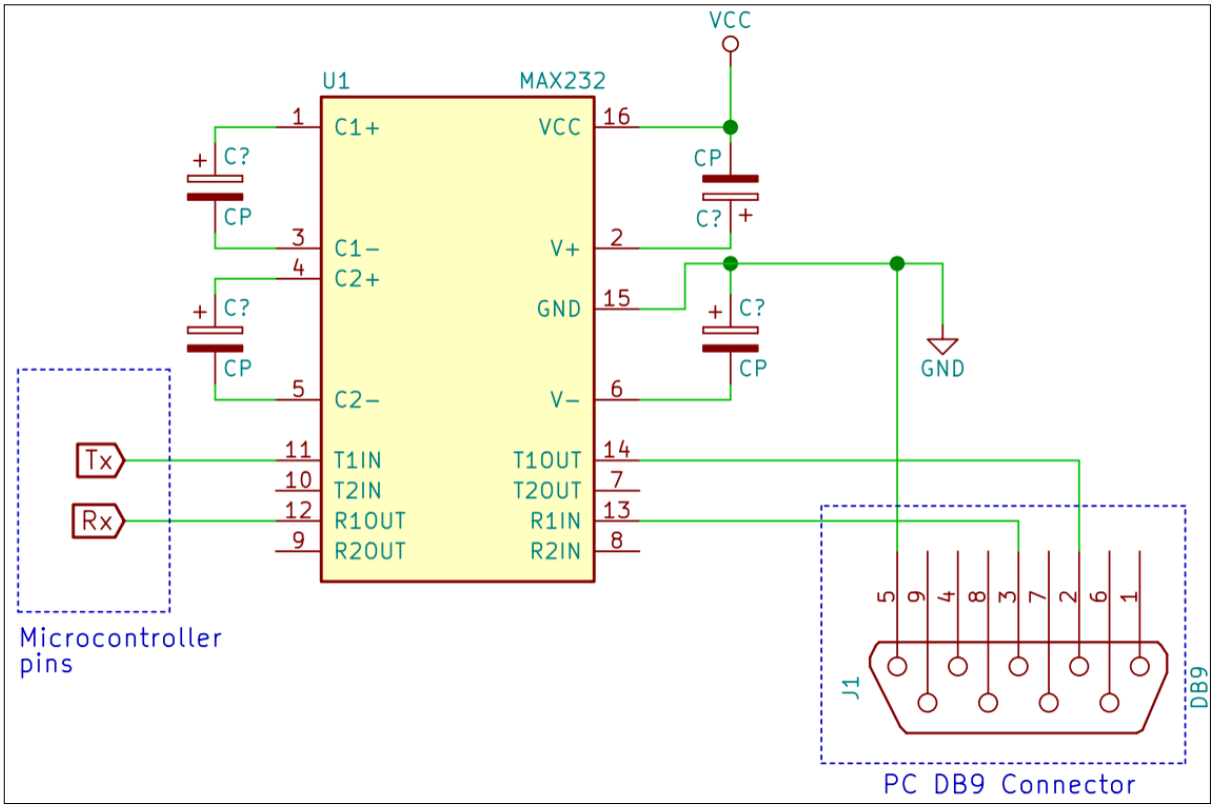


Figure II-43: UART interfacing with PIC circuit. [26]

The DB9 connector is an analog connector that is mostly used for serial communications with the RS-232 standard for asynchronous data transmission. The pinout for the PC DB9 connector is shown below.

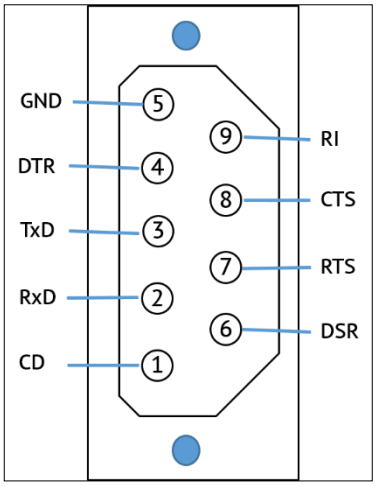


Figure II-44: DB9 Connector pinout

Figure II-14 shows the shows the TTL to RS232 convertor module. This unit is a simple low-cost RS232 adapter board for converting TTL level logic to RS232 levels for connecting with PCs, dumb

terminals, or other standard devices that use an RS232 serial connection. It was designed for embedding into third-party products.[26]



Figure II-45: TTL to RS232 convertor module

This module has 4 pins. UART TX and RX pins are connected successfully to the TX (pin 25) and RX (pin 26) of the microcontroller. The rest are VCC and GND pins. This unit can be powered from 3.0V to 5.5V.

II.1.4 Software design

In this part, we present the graphic interface realized within the framework of this project. This interface can communicate with the hardware part in order to collect the measurements and display them in numerical form and in graph form.

The python language was used to create this interface. The choice of python is justified by its simplicity and the global trend towards this language.



Figure II-46: Pycharm and python

For this development, we used: Pycharm and python 3.10 with Tkinter (Figure II-46). The first serves as a python development environment. In addition, the Tkinter is a module of python that provides an interface.

II.1.4.1 Pycharm software

PyCharm is a tool developed by the company JetBrains, a specialist in the development of IDES for different programming languages. The version used is the free version.

II.1.4.2 Python

Python is a powerful and easy to learn programming language. It features high-level data structures and allows a simple yet effective approach to object-oriented programming. Because its syntax is elegant, its typing is dynamic, and it is interpreted, Python is an ideal language for scripting and rapid application development in many areas and on most platforms.

The Python interpreter and its extensive standard library are freely available, either as sources or as binaries.

The Python interpreter can be easily extended with new functions and data types implemented in C or C++ (or any other language callable from C). Python is also suitable as an extension language for customizing applications.[27]

It is widely used in bioinformatics and in general data analysis. All of these features mean that Python is now taught in many courses.

II.1.4.3 Tkinter concept

Tkinter, sometimes known as "Tk interface," is a Python module that allows access to the Tk GUI toolkit. This toolkit was created in TCL (Tool Command Language), and it is multiplatform, supporting Linux, MAC OS, and Microsoft Windows. Although Tk is not a component of Python, it is native to Linux, MAC OS, and can be simply installed on Microsoft Windows. Tkinter, sometimes known as "Tkinter" in Python versions before 3 and "tkinter" after that, is a component of the language.[28]

Inside a Python script, widgets are objects, or instances of classes that represent mentioned window components. To instantiate an object on Tk, and then on Tkinter, is necessary to indicate its parent, what maintain a window hierarchy between all elements. On that hierarchy, the main window is the root. Each widget has a set of configuration options which control how they are displayed or how they behave, like a “text” option for components that display some text, as a label, or “command” option when they accept events, as the mouse click of a button.

II.1.4.4 Graphic interface

In this part, we present the design and the different parts of our GUI (Graphical User Interface) alongside the way our graphical user interface functions. Using Python and Tkinter.

When you launch the GUI a small window will appear. This root is named serial communication; it has inside a widget named Com Manager. This widget controller the serial communication between the MCU and the GUI in which you can select the available port depending on the right port the RS232 cable is on. Baud rate is the measure of the speed of data transfer in serial communication in units of symbols per second. It may also be considered as the number of bits per second under some circumstances. You can choose from the drop-down menu the speed you would like. In our project, we have chosen 9600 bits per second, the reason behind choosing such a board rate is this one works best with the crystal 16 MHz clock we are using for he microcontroller according to the Baud rate error calculating.[16]

Desired Baud Rate = FOSC / (64 (X + 1))

Solving for X:

X = ((FOSC / Desired Baud Rate) / 64) – 1

X = ((16000000 / 9600) / 64) – 1

X = [25.042] = 25

Calculated Baud Rate = 16000000 / (64 (25 + 1))

= 9615

Error = (Calculated Baud Rate – Desired Baud Rate) /Desired Baud Rate

= (9615 – 9600) / 9600

= 0.16%

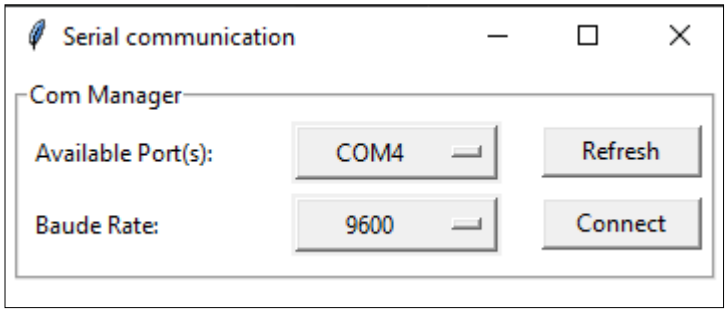


Figure II-47: Serial communication root

After clicking on connect button a security step will get checked, the GUI will send a code to the connected MCU and the MCU will send a code back to the GUI, the two communicating sides exchange messages to acknowledge each other, verify each other, and agree on session keys.

If the connection is successfully establish, the small root GUI will extend into a bigger one having more widgets on it.

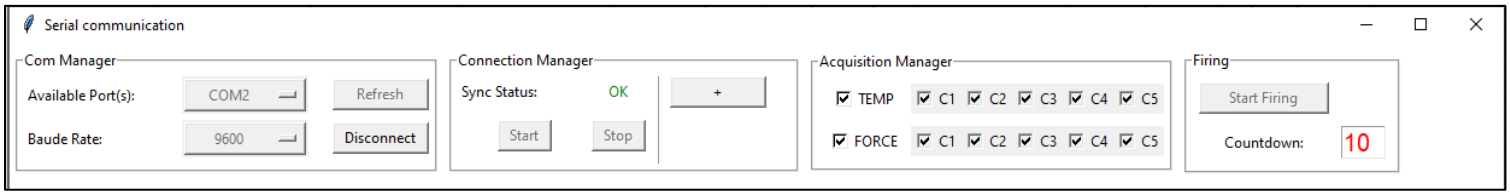


Figure II-48: Extended root GUI

This extended GUI contained 3 other widgets, Connection Manager, Acquisition Manager, and Firing widget. The Acquisition Manager widget is set to select the main sensor to acquire from, you can choose from 5 different Temperature sensors and the other 5 different force sensors. After choosing the right sensor to acquire data from, now is the time to head into the connection manager widget, this widget contains a “+” button that allows a pop-up window to appear this window contains the data graph on the right and the selected sensors on the left. You can check and uncheck the selected sensors from appearing in the graph. The “+” button allow you to add as many popup graph windows as you want for better visualization of different sensor data.

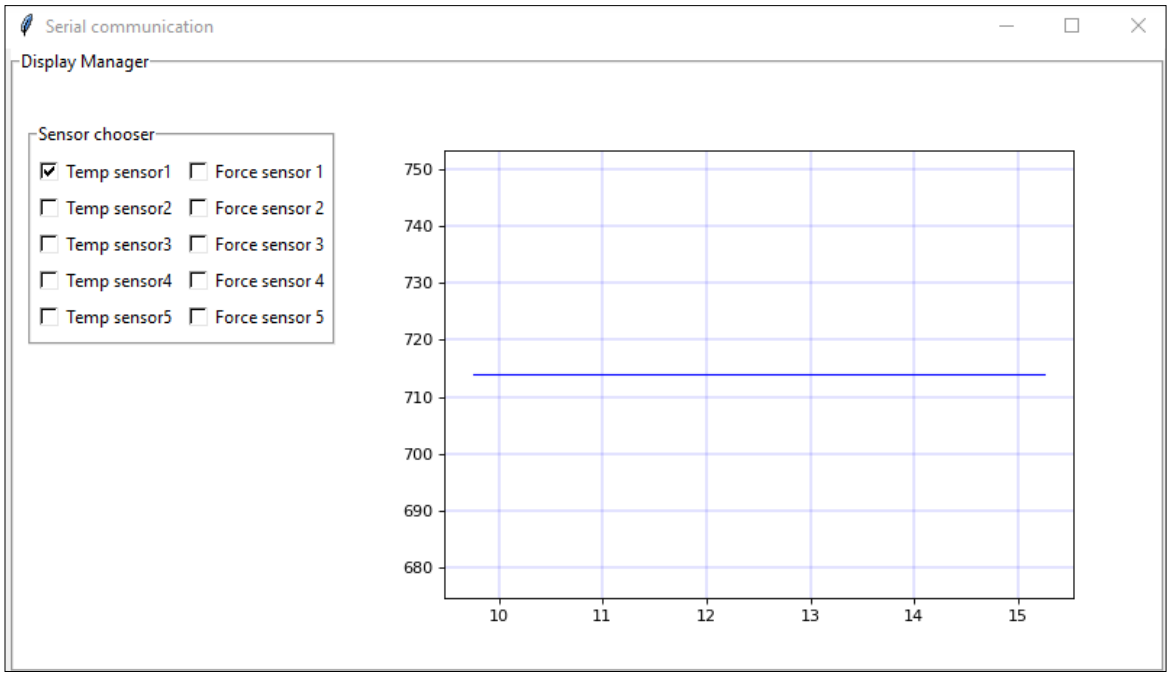


Figure II-49: Graph popup root

The start and stop button allow you to start streaming data from the microcontroller and stop it as well. it is necessary to start streaming data before starting the firing so you can be ready once the firing start and not miss any data.

Sync status assures the connection with the microcontroller if the connection is stable "OK" will show otherwise a "Sync...” symbol will be shown.

After having everything set and going it is time for firing, the last widget is the firing one clicking on the firing button will allow you to start the countdown of how many second you want in our case is 300

second, as soon as the countdown end a firing code will be sent to the MCU, this code will start immediately the firing by igniting the igniter.

II.1.4.5 The organization chart of the program produced in PIC18F452

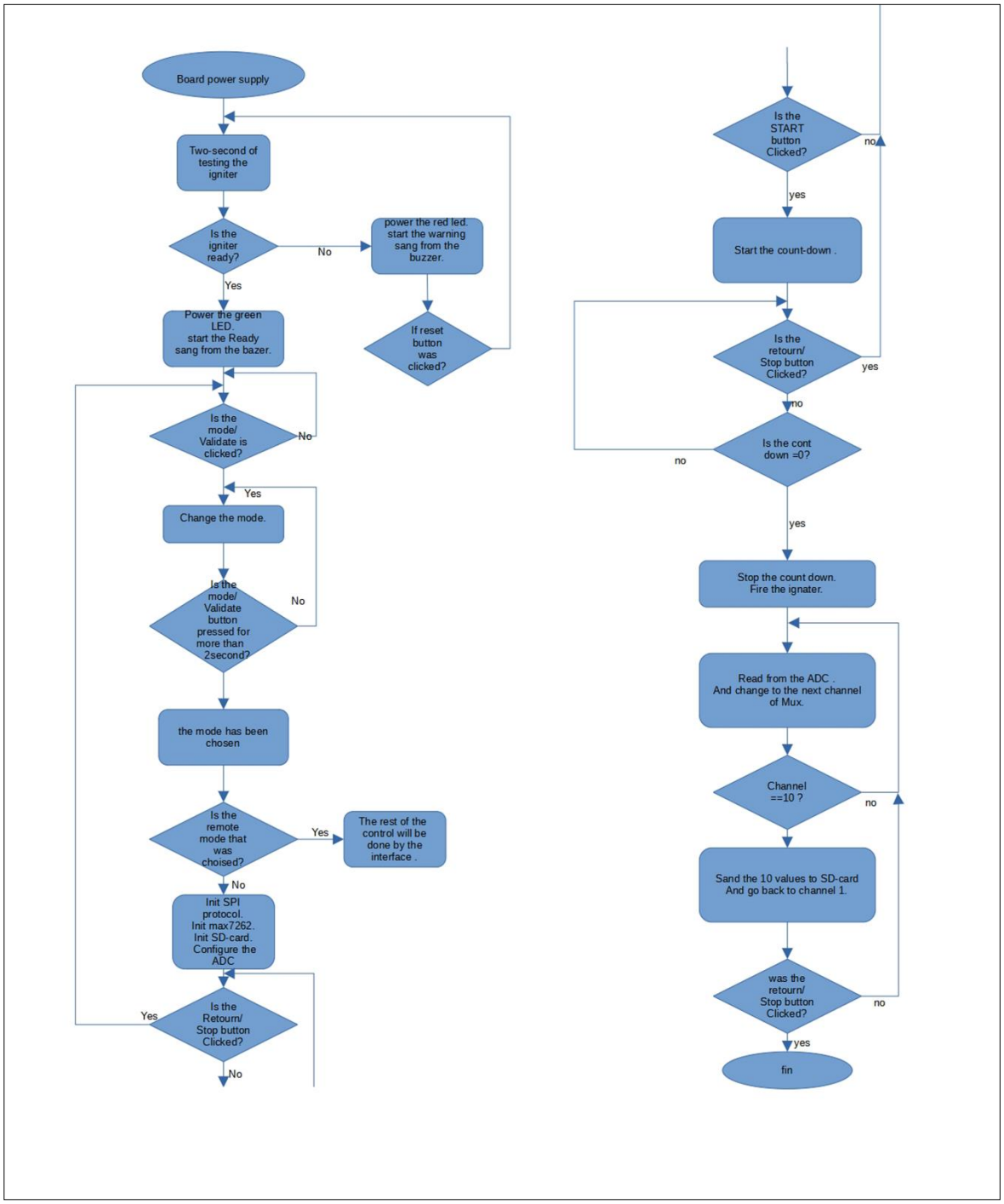


Figure II-50: The organization chart of the program produced in PIC18F452

II.2 Conclusion

In this chapter, we have detailed the different stages and the workflow of the design and realization our system has gone throw, from different conceptions, and hardware design, to software design and many more. This chapter illustrates the different stages of our practical work and the final layout.

Chapter III Results and testing

Introduction

In this chapter, we will go through the final PCB design of the electronic board. Hardware and software integration, by connecting the board with the graphic interface, and the final step will be the Verification and validation, it contain looking through some errors and trying to make everything acceptable.

III.1 PCB design

After the step of picking up the necessary components and being sure they are all working together without any problems, now we go forward to the next step where we will build a board so all the components can be placed on it, we call it PCB (Printed Circuit Board), and That was done using the software KiCad (a free open source application).

The PCB is composed of two-layer of copper separated by a substrate layer, the second layer helps when it is impossible to attach the wired pads.

There are several conditions and requirements to follow in order to make this PCB capable of being printed by the PCB making company. In our cases, we choose the almitech-dz the PCB production company. SNC ALMITech is an Algerian company, with great experience since 1991, established in Algiers; it has forged a solid reputation in professional electronic manufacturing.[29]

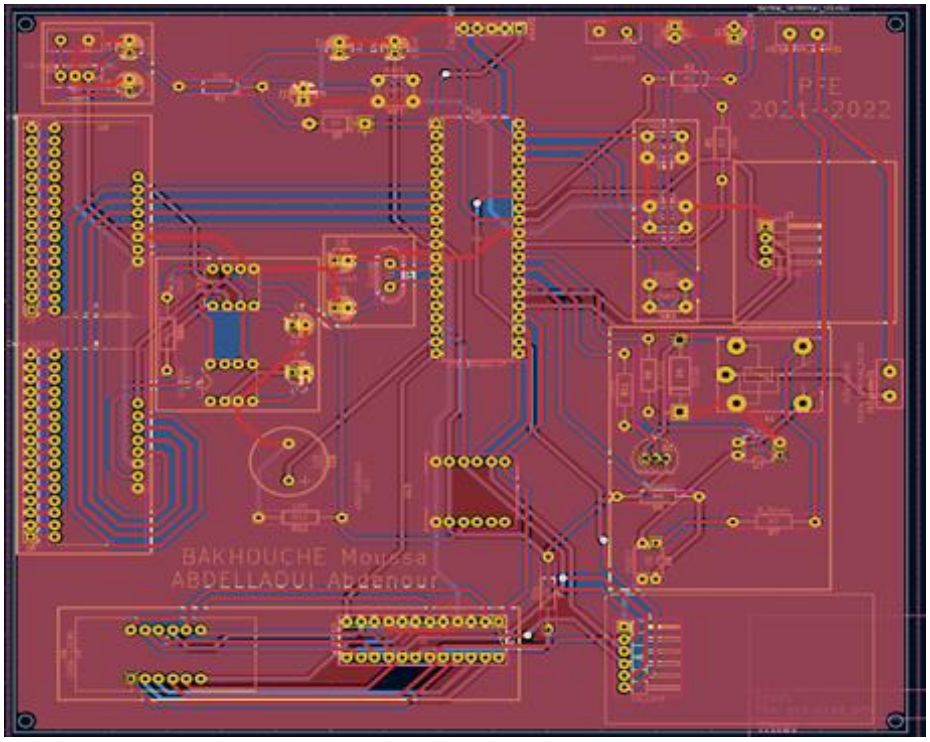


Figure III 1: Final PCB design

Those requirements are as follows:

- The minimum pad diameter is 1.6mm and the minimum pad hull diameter is 0.85mm.
- The minimum via diameter is 1.2mm and the minimum via hull diameter is 0.6mm.
- The minimum track width is 0.8mm and the minimum distance between the track and another track and cupper layer is 0.3mm.

The number of placed components is 51 elements with 292 pads wired by 116 nets.

The 3d model shows a good-looking image of the electric board.

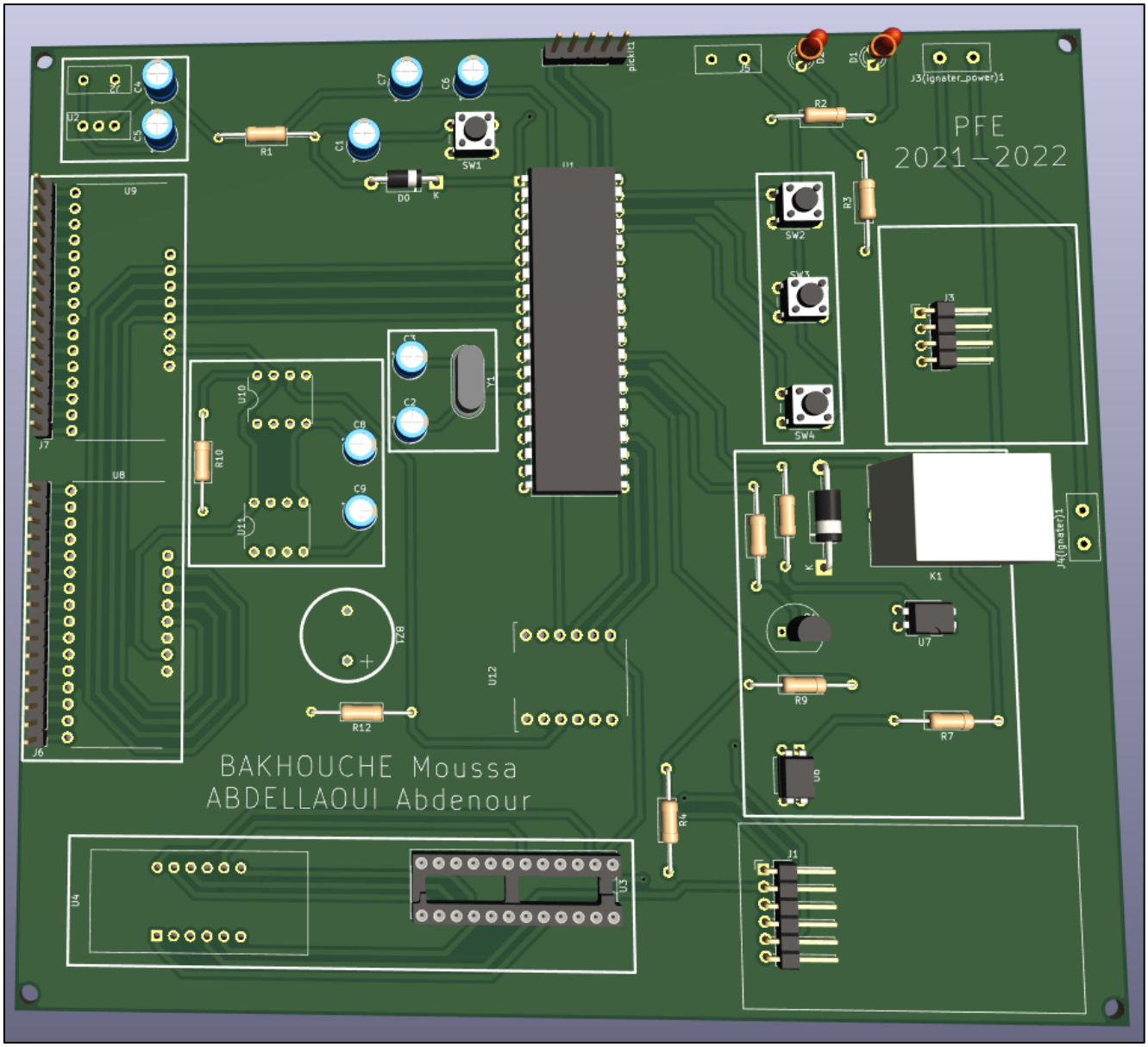


Figure III 2: 3D module of the front electronic board

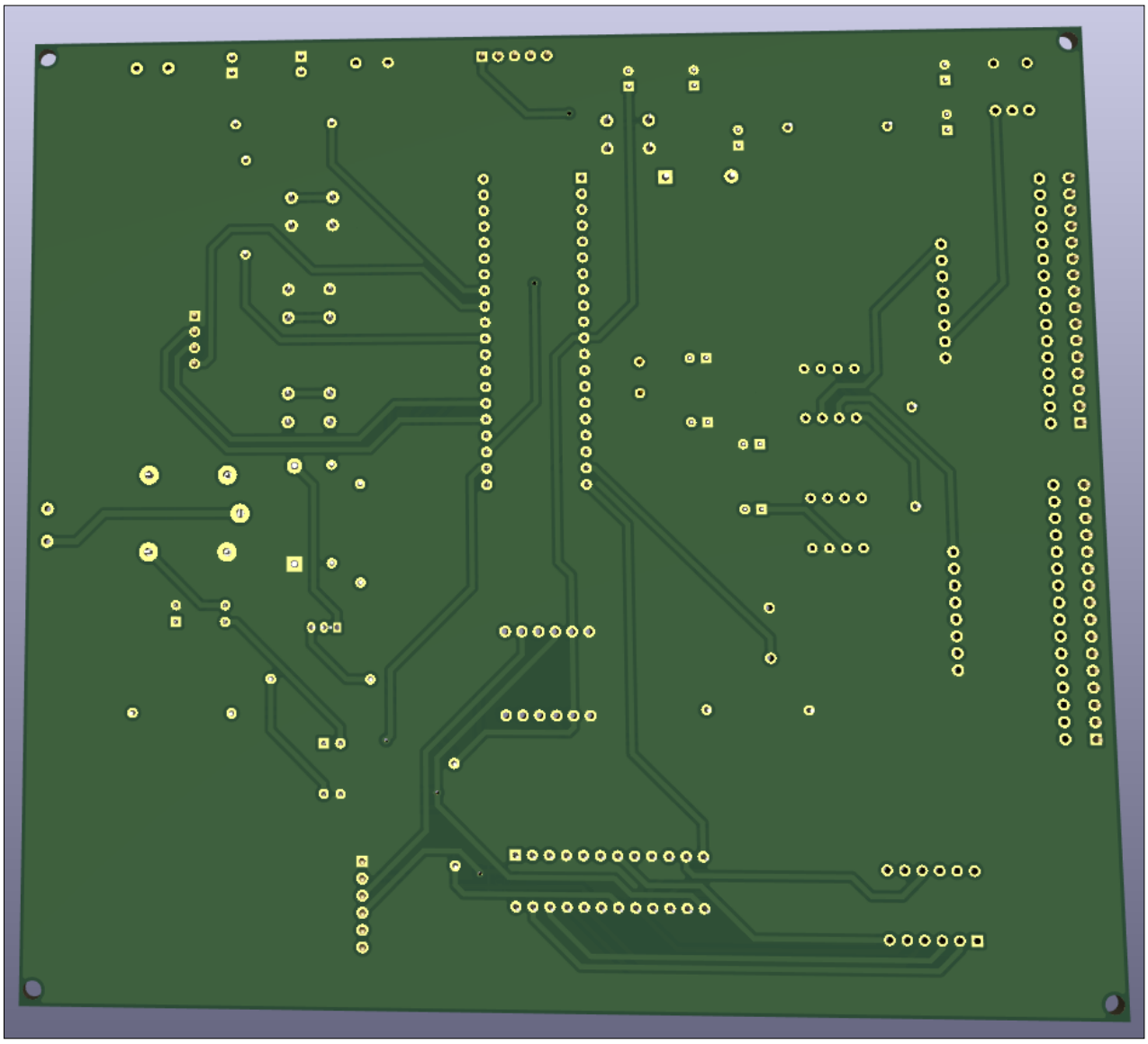


Figure III 3: The 3D module of the back electronic board

III.2 The final form of the data acquisition board

After printing the board and soldering all the components in it the board is now ready for the test. This figure down shows the final form of the data acquisition board. This board is divided into 8 different blocks. The **first block** is the 5V power supply circuit. The **second block** is the data acquisition consisting of the 2 multipliers, amplification (AD620) and ADC (ADC1118).

The **third block** is the display block that has the MAX7219 module along a 4 of 7 segment display to display the countdown and the type of mode we using either local or remote.

The **fourth block** is the firing block. It is responsible for lighting the igniter and other security.

The **fifth block** is the UART to RS232 module hock up to the RS232 to USB cable for transmitting data to the station in our case is a computer.

Block six contains 3 buttons CHANGE, STOP, and START. CHANGE button changes between modes local mode and remote mode. START starts the firing countdown and STOP stops the firing countdown.

Block seven has 2 wire entries; one is for the key lock security to prevent anyone from starting the firing. The second is the power supply for the firing circuit surely to light up the igniter. That is also 2 LED green one indicate that the firing circuit is all good; the red one is the opposite some error may occur.

The **eighth block** is for Pickit 3 programmer to hook up with the board in case we need to reprogram the PIC microcontroller.

The last **ninth block** is the SD cart block to save data in our cast we couldn't find a reliable SD cart to use with this module.

The rest is basically the PIC microcontroller, reset button, clock, and a buzzer to sound when the firing started.

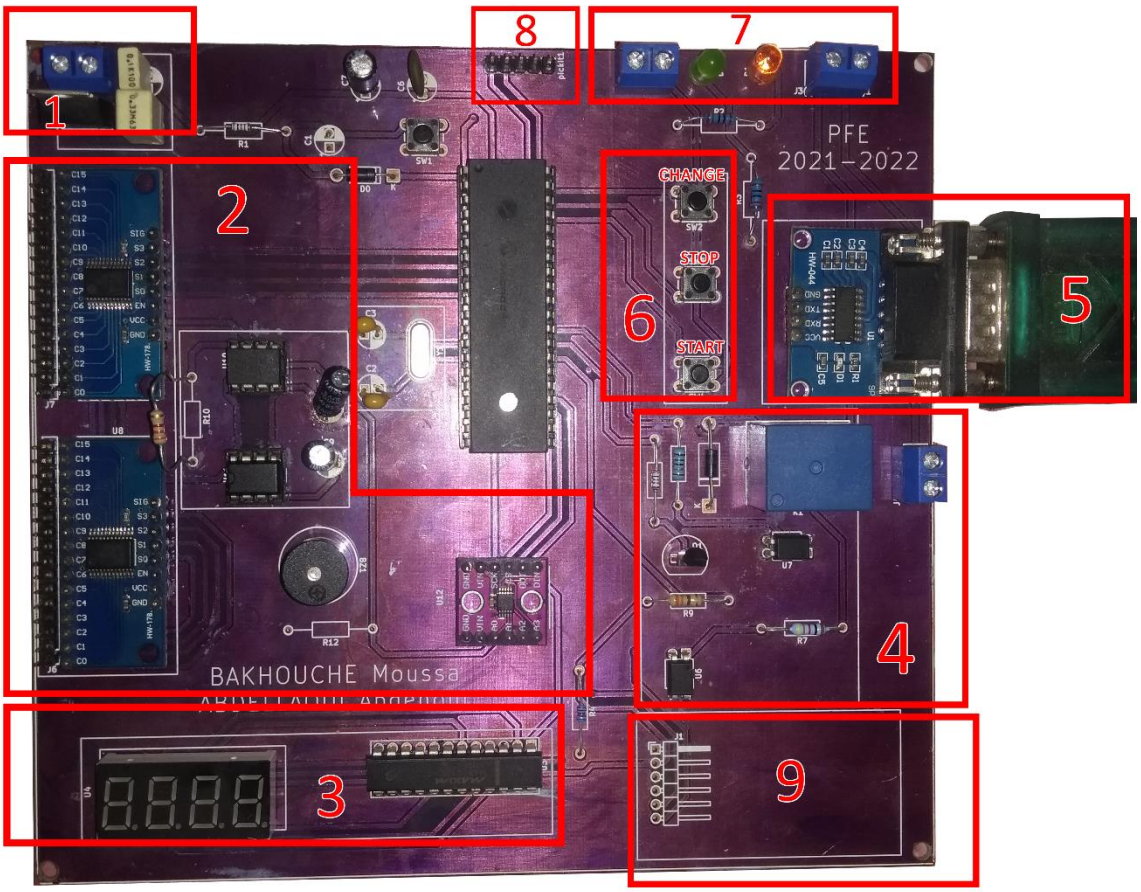


Figure III 4: Data acquisition board and it different blocks

III.3 The scale and the temperature sensors

III.3.1 Load cell Sensor

Standard load cells like this one can measure weight up to 10 kg. This sensor is most frequently used in weighing scales. This conversion takes place in two steps and is indirect. A strain gauge is deformed by the force being perceived thanks to a mechanical setup. Because the strain modifies the wire's actual electrical resistance, the strain gauge detects the deformation (strain) as an electrical signal. Four strain gauges are often arranged in a Wheatstone bridge configuration to form a load cell. Before it can be used, the electrical signal output, which is normally in the range of a few millivolts, must be amplified by an instrumentation amplifier. Through multiplexer channels, the transducer's output is connected to the board.



Figure III 5: 10KG Load cell Scale Sensor

III.3.1.1 Sensor calibration

Due to using a non-datasheet available sensor we had to use the next best step to adjust the sensors to our correct output this step called sensor calibration

An algorithm to calculate the force applied to the transducer is necessary to display the amount of force in order to do that we have set a couple of weights into the scale sensor to calibrate the load cell with the microcontroller.

The graph below (**Figure III-6)** shows the different weights put into the sensor on Kg with the rated voltage output of the sensor after amplification in Volte. This graph is done using the program **Excel**.

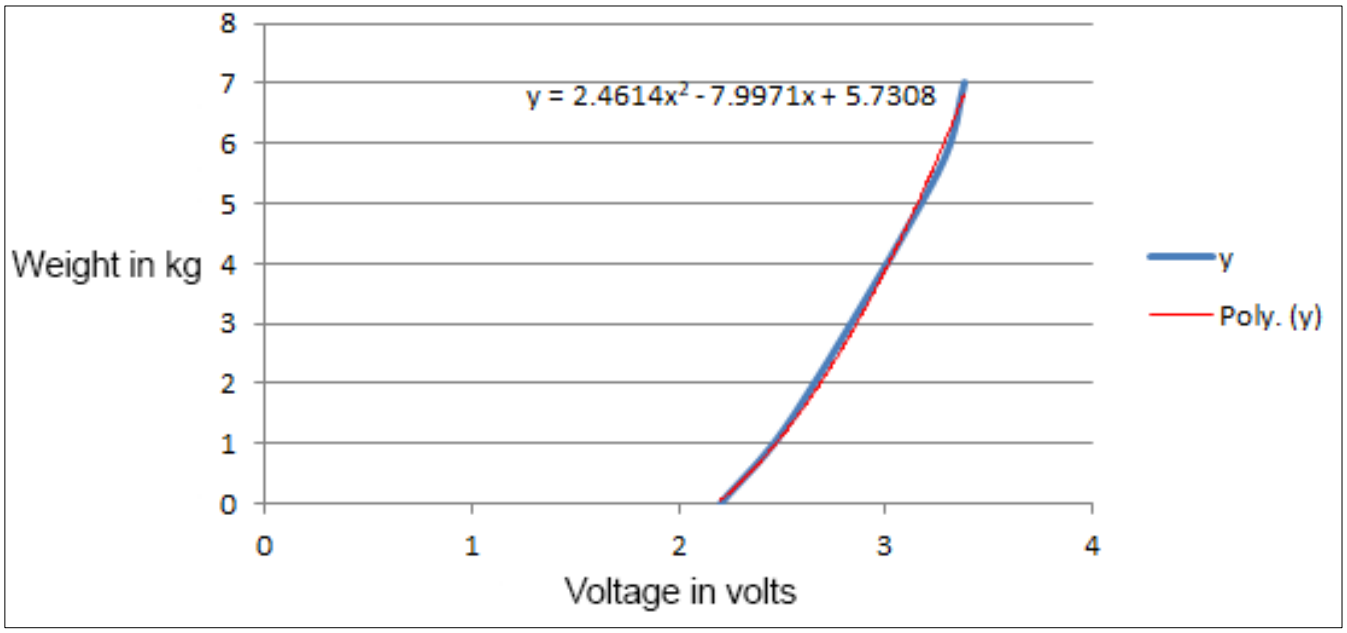


Figure III 6: Graph of weights scale sensor with rated output voltage

Using the said graph, we have concluded the following equation:

Y = 2.4614x2 - 7.9971x + 5.7308

This equation is embedded into the source code of the microcontroller giving accurate measurements associated with this weight scale sensor.



Figure III 7**: Weight scale sensor**

III.3.2 Thermocouple sensor

We decide to use the K Type Thermocouple mainly because of the perfect range of temperature it can handle/measure which is around 0-800°C. The following figure shows the temperature sensor.

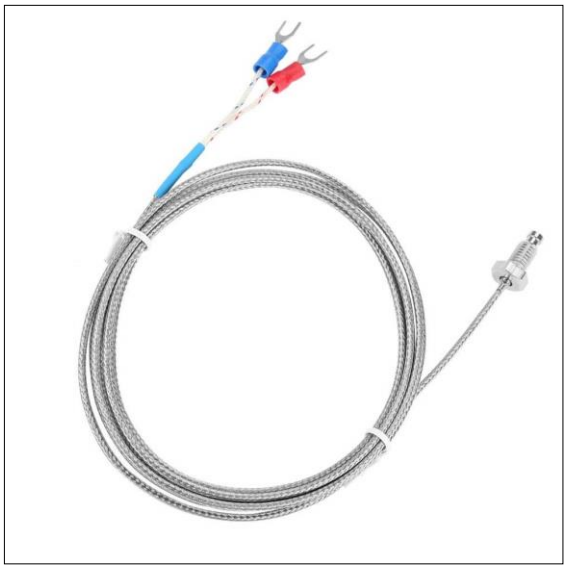


Figure III 8: K Type Thermocouple

III.3.2.1 Sensor calibration

In order to calibrate the thermocouple with the microcontroller, we hooked up the sensor to a fire-lighter and read the output voltage from the sensor directly and after amplification. And then we used the table of the K thermocouple to find out the temperature measured. Finally, we end up with a graph that displays the temperature measured in °C with the output voltage in mV (**Figure III-9)**. This graph is done using the program **Excel**.

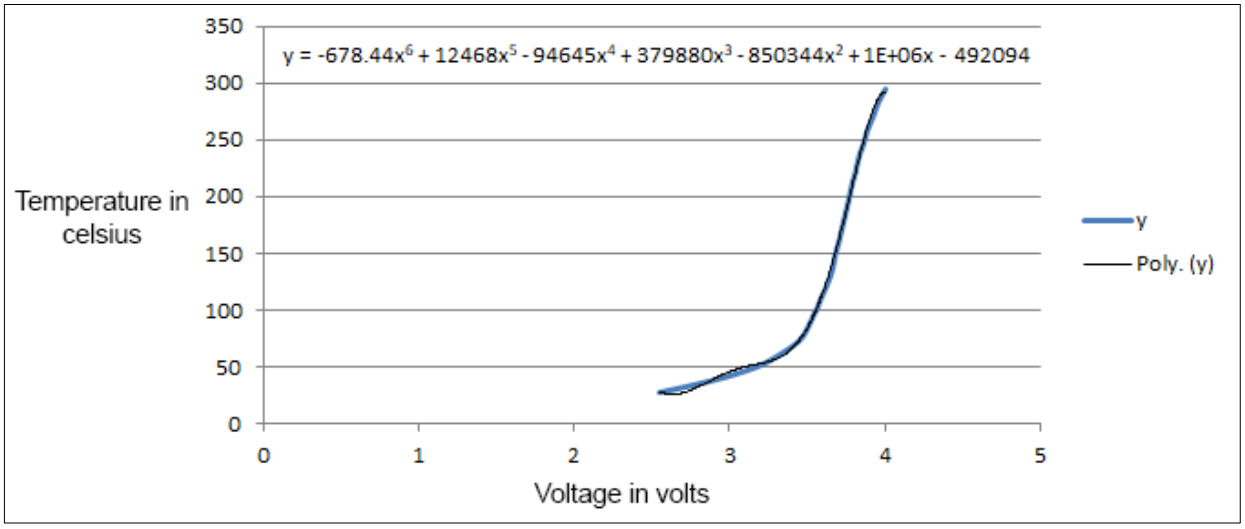


Figure III 9: Graph of Temperature measured in °C with the output voltage in mV of Type K Thermocouple

Using the said graph we have concluded the following equation:

$$\mathbf{Y = -678.44x^6 + 12468x^5 - 94645x^4 + 379880x^3 - 850344x^2 + 1E+06x - 492094}$$

This equation is embedded into the source code of the microcontroller giving accurate measurements associated with this Type K Thermocouple.

III.4 Test and result

III.4.1 Firing test and result

Firing can be done with two ways either with local mode: by hand in the board after going through the security steps of opening the key lock, having the green light which indicate that the firing circuit is ready and finally clicking on the start button that start a countdown which can be read in the 7 segments on the board. Or with remote mode which is simply clicking start firing on the graphic user interface, after the countdown is up the firing start.

This firing start through a large current voltage coming in straight from the wall outlet using a charger. This current is responsible for lighting up a thin cupper wire around a match head, the following (**Figure III-10)** demonstrate it.



Figure III 10: Cheap igniter using match head with thin copper strands

III.4.2 Weight scale sensor test and result

In this part, we carried out the installation of the weight scale sensor with the acquisition board and connected it to the computer. After that, we conducted some tests and got the following results attached in the figures below

Using the interface we are going to test the data acquisition board in the **Remote mode** which means we are going to pilot the board in distance using only the graphic user interface.



Figure III 11: Data acquisition using an electric board with a load of 1.5 Kg

This figure shows the electric board powered by a 5v power supply board instead of a 9v battery, with the load cell holding about 1.5 Kg of weight. This information is processed and sent to the PC using the local mode. The result is quite satisfying it jumps between 1.51 and 1.59 Kg this noise is due to various resonces sensors, cables, etc.

The next **Figure III-12** shows the screenshot of the pc during the test with the 1.5Kg of weight.

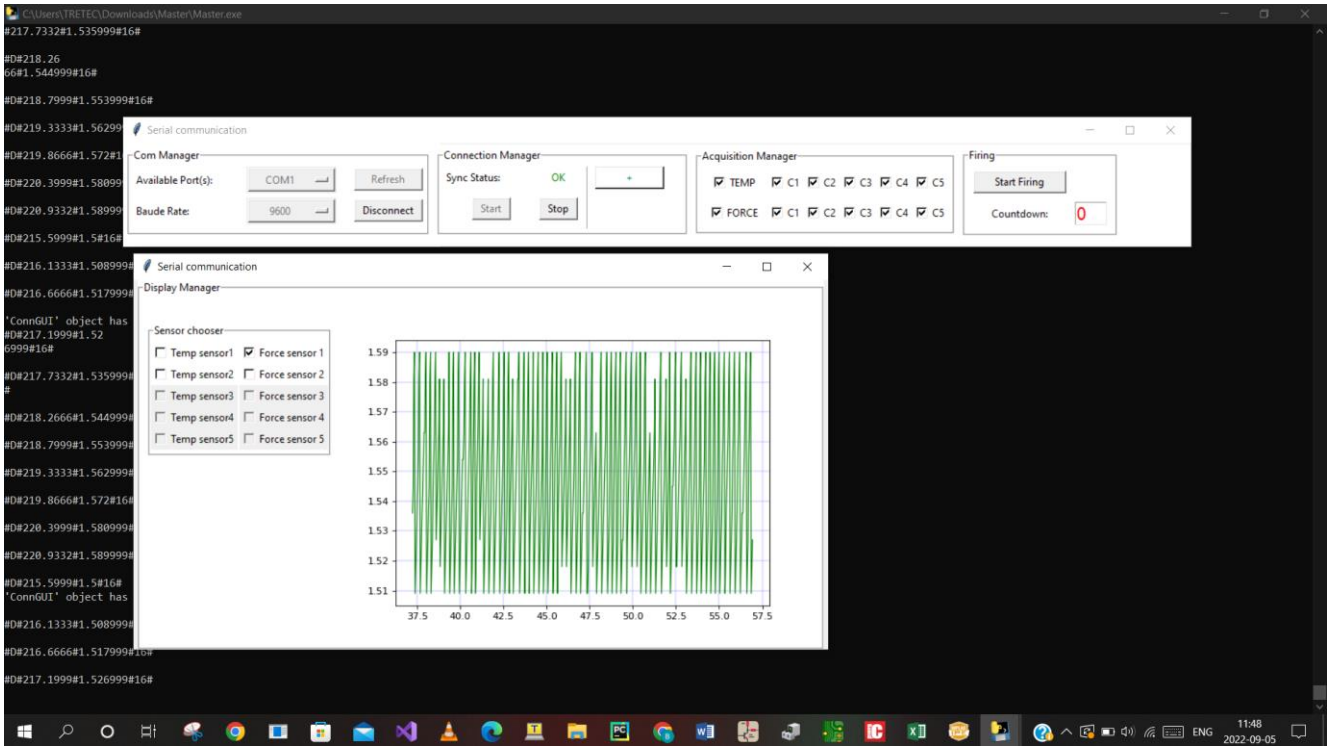


Figure III 12: Screenshot of the pc during the test with the 1.5Kg of weight

The same test now using 5.5Kg of weight

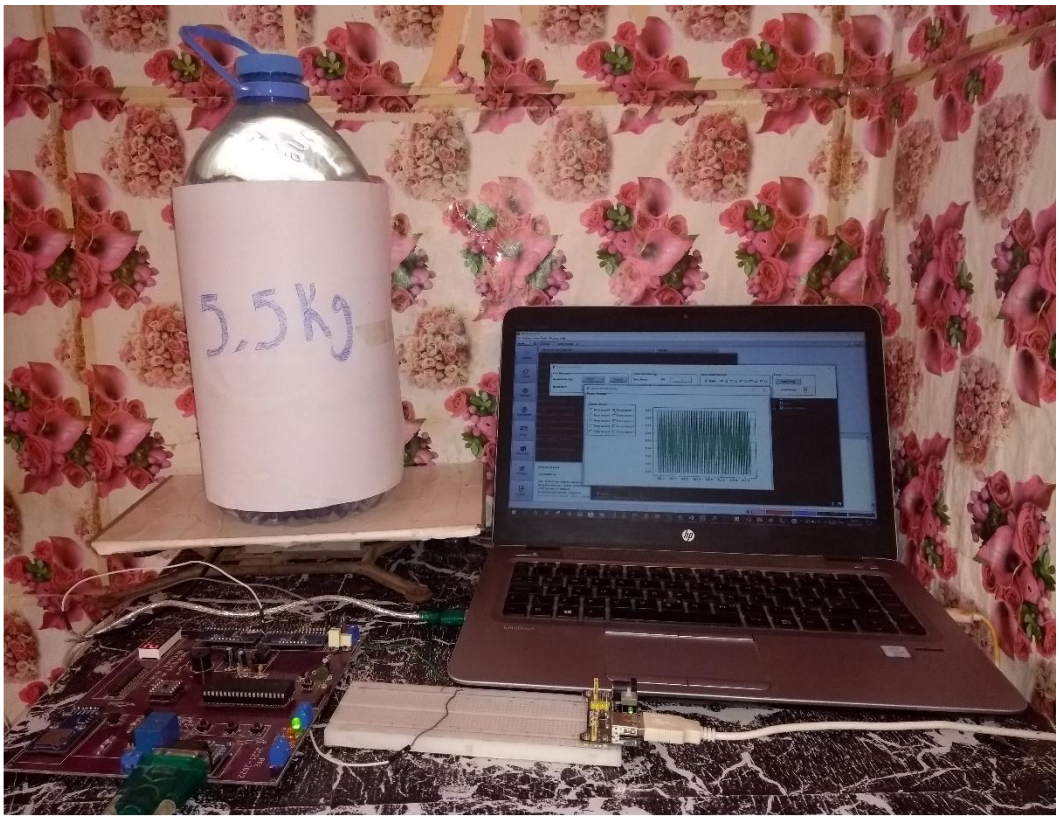


Figure III 13: Data acquisition using an electric board with a load of 5.5 Kg

The next **Figure III-14** shows the screenshot of the pc during the test with the 5.5Kg of weight.

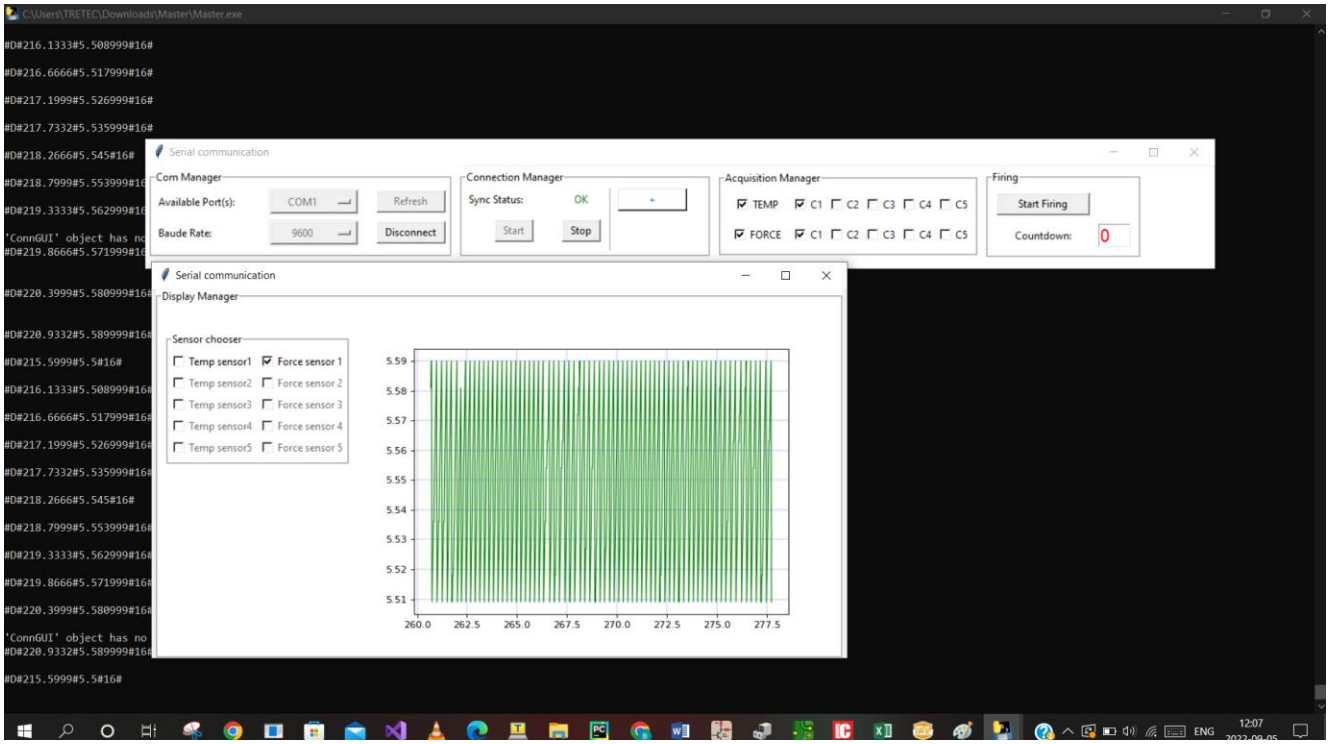


Figure III 14: Screenshot of the pc during the test with the 5.5Kg of weight

III.4.3 Thermocouple sensor test and result

We installed the Type K Thermocouple with the acquisition board in this section and linked it to the computer. Following that, we ran various tests and the results are shown in the figures below.

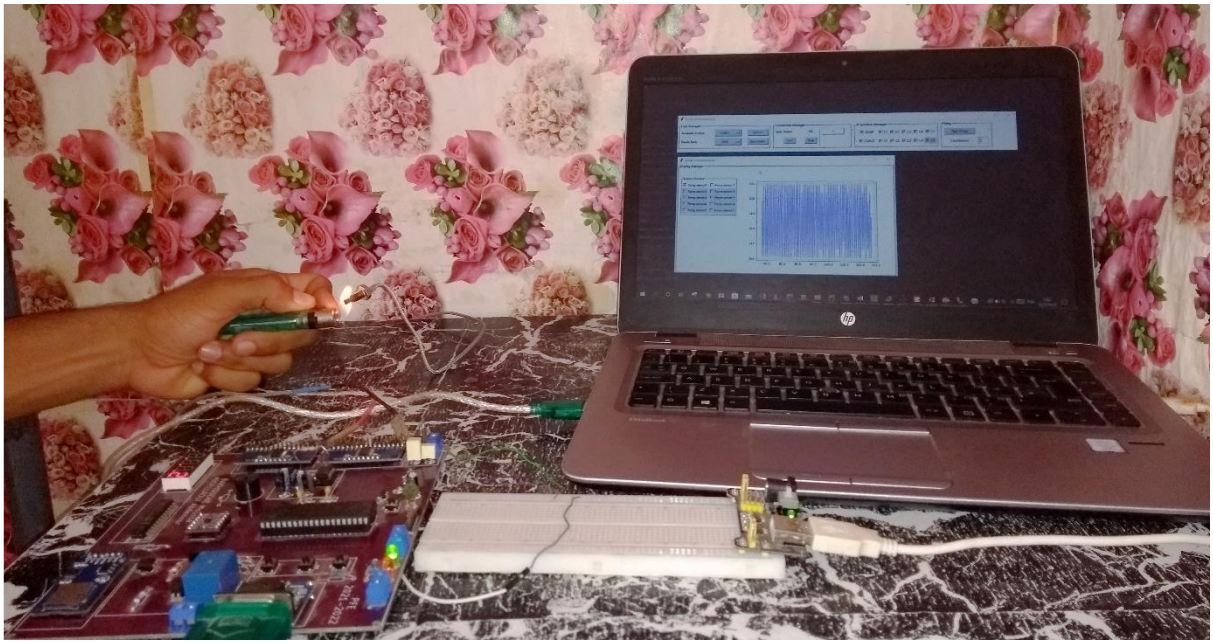


Figure III 15: Data acquisition using an electric board with a K type temperature sensor

We used a lighter to ignite a small flame to test the temperature sensor, for about 1 minute of holding the flame next to the temperature we end up with around 216 to 221 degrees Celsius. as shown in the **Figure III-16**

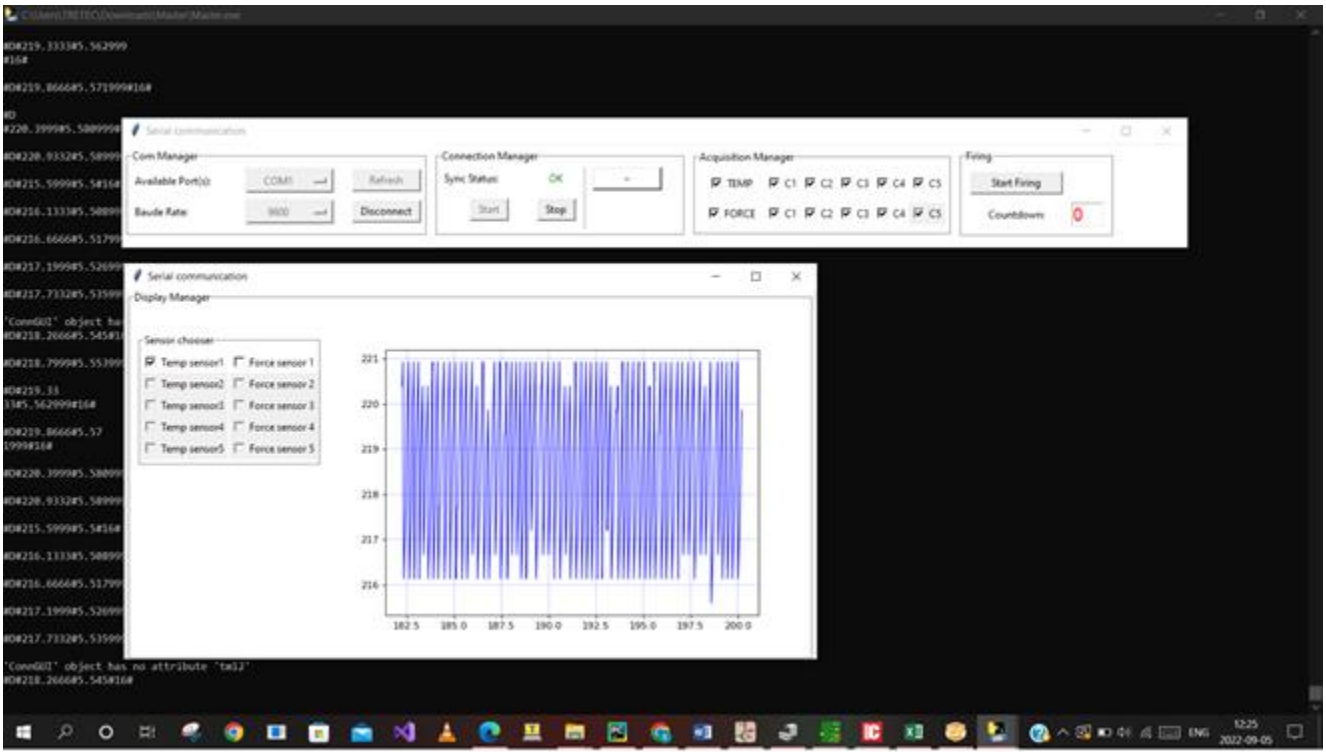


Figure III 16: Screenshot of the pc during the test with the K-Type temperature sensor.

III.5 Conclusion

After printing the board and soldering each component into it, it was time for testing each part of the board, the results of the testing concluded the following, firing which was used using a homemade igniter was perfectly fine to ignite a fire, data acquisition testing through remote using graphics interface as a commander of the board and the local mode when you use the board manually. The results were satisfying not precisely accurate around 5% error from each sensor temperature sensor and weight scale sensor, but nonetheless very relatable.

General conclusion

The work within the framework of the graduation project allowed us to deepen our theoretical knowledge in the field of electronics and to acquire a good experience in practical realization. The project gave us a better idea of the integration between the theoretical part and the practical part.

For this, we have created a data acquisition/firing board to acquire and display data from different force sensors and temperature sensors, and control firing by lighting the igniter inside the rocket engine, after going through multiple security systems. The results obtained are quite satisfactory, as we were able to launch firing through the board safely and save data locally through an SD cart. And in the same way remotely, we were able to pilot the board and start firing, following the transfer of data from the acquisition board through the different sensors to the computer, meaning we establish a connection between the acquisition board and the computer.

We have presented this work in three chapters. In the first chapter, we explained the nature of the data acquisition system, its components, and its structure.

Then, we discussed in the second chapter the architecture and the design process we went through in establishing the board. our structural method steps were complete. the definition of our product requirements (local and remote mode, security system, component reliability). the system architecture was the best we come up with (sensor plus multiplexer plus amplifier plus ADC to MCU). the hardware design was held on KiCad this program helps us create the schematic circuit of the different blocks of the board. And lastly, the software which consists of a graphical user interface was made by pythons with Tkinter library.

Finally, we end up with a final PCB design suitable for the board acquisition and firing capability using the same KiCad program. After we solder all components the testing will take place, in the testing phase we used only two sensors one for force and one for temperature the result of the testing was accurate in both force and temperature sensors. On the other hand, firing was successfully triggered using a hand-made igniter.

Overall the data acquisition/firing board was successfully operating in all its aspects and it is ready to be used in the field of measurement.

Annexes

Parameters of the electronic board:

Parameters	**Manufacture**	**Price (DA)**
LM7805	Continental Device India Ltd (CDIL)	~300
Pickit 3	Microchip	~5000
Switch keylock	Mouser electronic	~1500
Relay	TME electronic components	~ 400
MAX7219	Maxim Integrated	~300
74HC4067	Texas instruments	~500
AD620	Analog devices	~800
ICL7660S	Renesas	~ 200
ADS1118	Texas instruments	~ 2000
SD card module	Catalex	~ 350
RS232 module	Maxim integrated	~ 500
RS232 cable to USB	General	~ 900
PIC18f452	Microchip	~ 1500
PCB Production	ALMITech	~3000

Bibliography list.

[1]. Georges Asch, Patrick Renard, Pierre Desgoutte, Zoubir Mammeri, Éric Chambérod et al, Acquisition de données - 3ème édition,2011.

[2]. TAYLOR, H. ROSEMARY, Data Acquisition for Sensor Systems.

[3]. Vijay Kamble, Vasudev Dhondiram shinde, Overview of Load Cells.

[4]. Baker, Bonnie, How delta-sigma ADCs work, Part 1,2016.

[5]. Emilio, Maurizio Di PaoloEmbedded Systems Design for High-Speed Data Acquisition and Control,2015.

[6]. Poon, H.L, Applications of Data Acquisition Systems.

[7]. Jong-Gyun Choi, Poong Hyun Seong, Reliability of Electronic Components.

[8]. Analog Devices, IncAD7794/AD7795,2004-2007.

[9]. Devices, Analog, AD71912009.

[10]. Instruments, TexasADS1118 Ultrasmall, Low-Power, SPI™-Compatible, 16-Bit2021.

[11]. instruments, texas, ADS111x Ultra-Small, Low-Power, I2C-Compatible, 860-SPS, 16-Bit ADCs, 2022.

[12]. logic, cirrus, Single-supply, 16-bit A/D Converter, 2005.

[13]. Microchip, MCP3421, 2009.

[14]. Akhlaghi, Zhamak, Microcontroller Based Applied Digital Control, 2006.

[15]. Takawira F., Dawoud D.S, Selecting the right microcontroller unit.

[16]. Microchip, PIC18F452, microchip.

[17]. Kicad, kicad docs.

[18]. Microchip, 8-bit Bootloader.

[20]. MAXIM, Serially Interfaced, 8-Digit LED Display Drivers.

[22]. instrument, texas, CD74HC4067.

[23]. devices, analog, Low Cost Low Power Instrumentation Amplifier AD620.

[24]. Renesas, ICL7660S.

[25]. CATALEX, Datalogger - SD Memory Reader.

[26]. INT232-A, TTL TO RS232 CONVERTOR BOARD.

[27]. Python, The Python Tutorialhttps://docs.python.org/3/tutorial/#the-python-tutorial.

[28]. D. B. Beniz, A. M. EspindolamUSING TKINTER OF PYTHON TO CREATE GRAPHICAL USER INTERFACE (GUI) FOR SCRIPTS IN LNLS.

[19]. Lirtex, Microcontroller In Circuit Serial Programming (ICSP) with Microchip PIC and Atmel AVR,https://www.lirtex.com/embedded/microcontroller-icsp-in-circuit-programming/.

[21]. CAMPBELL, SCOTT, BASICS OF THE SPI COMMUNICATION PROTOCOL, https://www.circuitbasics.com/basics-of-the-spi-communication-protocol/.

[29]. ALMITech, SNC, About us, http://pcbproduction.almitech-dz.com/en/2020/03/about-us/.

Abstract

A data acquisition system is a collection of software and hardware that allows one to measure or control the physical characteristics of something in the real world. This work aims to equip a solid rocket propellant test bench with an electronic system capable of controlling firing and acquiring the necessary measurements. As well as developing a GUI to control and visualize the measurements. The results of this work gave satisfactory measurement results while saving /transmitting measurements (temperature, force, etc.). All that is done with a high level of security and reliability.

Keywords: Propellant test bench, firing, acquisition system

Résumé

Un système d'acquisition de données est un ensemble de logiciels et de matériel qui permet de mesurer ou de contrôler les caractéristiques physiques de quelque chose dans le monde réel. Ce travail vise à équiper un banc d'essai de propergol solide d'un système électronique capable de contrôler le tir et d'acquérir les mesures nécessaires. En plus de développer une interface graphique pour contrôler et visualiser les mesures. Les résultats de ce travail ont donné des résultats de mesure satisfaisants tout en sauvegardant/transmettant les mesures (température, force, etc.). Le tout avec un haut niveau de sécurité et de fiabilité.

Mots-clés : Banc de test propulseur, mise à feu, système d'acquisition

الملخص

نظام الحصول على البيانات عبارة عن مجموعة من البرامج والأجهزة التي تسمح للشخص بقياس أو التحكم في الخصائص الفيزيائية لشيء ما في العالم الحقيقي. يهدف هذا العمل إلى تزويد منصة اختبار وقود الصواريخ الصلبة بنظام إلكتروني قادر على التحكم في الإطلاق والحصول على القياسات اللازمة. بالإضافة إلى تطوير واجهة المستخدم الرسومية للتحكم في القياسات وتصورها. أعطت نتائج هذا العمل نتائج قياس مرضية مع توفير / نقل القياسات (درجة الحرارة ، القوة ، إلخ). كل ذلك يتم بمستوى عالٍ من الأمان والموثوقية.

الكلمات المفتاحية: منصة اختبار الدفع ، الحرق ، نظام الاستحواذ