الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

UNIVERSITE BADJI MOKHTAR - ANNABA
BADJI MOKHTAR – ANNABA UNIVERSITY

جامعة باجي مختار – عنابـــــة

Faculté : TECHNOLOGIE

Département : ELECTRONIQUE

Domaine : SCIENCES ET TECHNIQUES

Filière : ELECTRONIQUE

Spécialité : Electronique des Systèmes Embarqués

Mémoire

Présenté en vue de l'obtention du Diplôme de Master

Thème:

Development of a control system prototype for surgical operating table

Présenté par : *MERAH Zakaria*

FALEK Khalil Mohamed Tayeb

Encadrant : *YAHI Amira* *M.C.B* *UBMA*

Jury de Soutenance :

ZERMI Narima	M.C.A	UMBA	Président
YAHI Amira	M.C.B	UBMA	Encadrant
AMARA Fethi	M.C.B	UBMA	Examinateur

Année Universitaire : 2021/2022

acknowledgements :

I want to thank my supervisors, Pr. Benouaret Mouhamed and Dr. Yahi Amira, for your patience, guidance, and support. I have benefited greatly from your wealth of knowledge and meticulous editing. I am extremely grateful that you took me on as a student and continued to have faith in me over the years.

Thank you to my committee members, Dr. Zermi Narima and Dr. Amara Fethi. Your encouraging words and thoughtful, detailed feedback will be very important to me.

Thank you to friends and family, for your endless support. You have always stood behind me, and this was no exception.

Abstract :

The current report was created at the conclusion of the implementation of a prototype of an operating table controlled by Android via Bluetooth for our end-of-study project. It is meant to assist any student or aspiring engineer in all aspects of operating tables. This report initially covers the table's history, but it also describes its many varieties and all conceivable placements of the universal table, as well as how to govern it. In this work we use the ESP32 microcontroller to control a stepper motor to raise the table or lower it in order to match the ideal position for the surgical operation. we describe the technical aspects of our project as well as the equipment required for our job. Finally, we illustrate the findings of our research as well as the processes we took to get there. Accompanied with a list of recommendations and suggestions for further development.

Keywords : ESP32, microcontroller, stepper motor, surgical table, android.

Résumé :

Le présent rapport a été réalisé à l'issue de la mise en place d'un prototype de table d'opération pilotée par application Android pour notre projet de fin d'études du Master. Il est destiné à aider tout étudiant ou ingénieur en herbe dans tous les aspects des tables d'opérations. Ce rapport couvre d'abord l'histoire des tables opératoires, mais il décrit également ses nombreuses variétés et tous les emplacements imaginables de la table universelle, ainsi que la manière de la gouverner. nous utilisons l'ESP32 pour commander un moteur pas à pas pour but de remonter la table ou la faire descendre afin de correspondre à la position idéale pour l'opération chirurgicale. Nous décrivons ensuite les aspects techniques de notre projet ainsi que les équipements nécessaires à notre travail. Enfin, nous illustrons les résultats de nos recherches ainsi que les processus que nous avons suivis pour y arriver. Accompagné d'une liste de recommandations et de suggestions pour le développement ultérieur.

Mots clés : ESP32, microcontrôleur, moteur pas à pas, table chirurgicale, android.

الملخص :

تم إنشاء التقرير الحالي في ختام تنفيذ نموذج أولي لطاولة طبية يتم التحكم بها بواسطة تطبيق اندرويد كمشروع نهاية الدراسة الخاص بشهادة الماستر. إنه يهدف إلى مساعدة أي طالب أو مهندس طموح في هذا المجال. يغطي هذا التقرير في البداية تاريخ الطاولات الطبية ، ولكنه يصف أيضًا تنوعه العديدة وجميع المواضع التي يمكن تطويرها ، فضلاً عن كيفية التحكم فيه. نستخدم **ESP32** للتحكم في محرك متدرج لغرض رفع الطاولة أو خفضها لتتوافق مع الوضع المثالي للعملية الجراحية ثم نصف الجوانب التقنية لمشروعنا بالإضافة إلى المعدات المطلوبة لعملنا. أخيرًا ، نوضح نتائج بحثنا بالإضافة إلى العمليات التي اتخذناها للوصول إليها. مصحوبة بقائمة من التوصيات والاقتراحات لمزيد من التطوير.

الكلمات المفتاحية: **ESP32**، متحكم ، محرك متدرج ، طاولة جراحية ، أندرويد.

Abbreviations list

Esp	Espressif Systems
SoC	System on chip
RF	Radio-frequency
TSMC	Taiwan Semiconductor Manufacturing Company
SRAM	*Static* Random Access Memory
ROM	*Read Only Memory*
RTC	real-time clock
BLE	*Bluetooth Low Energy*
GPIO	General Purpose Input/Output
ADC	analog-to-*digital* converter
SAR	Specific Absorption Rate
DAC	*digital*-to-analog converter
UART	Universal Asynchronous Receiver Transmitter
SPI	Serial Peripheral Interface
LAN	local area network
SD	Secure Digital
SDIO	Secure Digital I/O Card
MMC	Modular Multilevel Converter
PWM	Pulse width modulation
LED	light-emitting diode
AES	Advanced Encryption Standard
Hash	short-length alphanumeric string created from a document, sort of its digital fingerprint
RSA	Rivest–Shamir–Adleman

ECC	Error-Correcting Code
RNG	random number generator
DTR	Data Terminal Ready
RTS	Return from Subroutine
MCU	microcontroller unit
CNC	Computer numerical control
DVD	digital video disc
3D	three-dimensional
NEMA	National Electrical Manufacturers Association
DSP	Digital signal processing
EDF	Electronic Document Filing
IDE	Integrated development environment
JSON	Java script object notation
XML	Extensible Markup Language
URL	Uniform Resource Locator
MIT	Massachusetts institute of technology
OS	Operating system
GUI	Graphic user interface
pul	Pulse
Dir	direction
ena	enable
MPU	Microprocessor Unit

Table of contents :

CHAPTER III : Implementation and Results

List of tables :

Table of figures:

General Introduction

An operating table, sometimes called operating room table, is the table on which the patient lies during a surgical operation. This surgical equipment is usually found inside the surgery room of a hospital.[1]. Since Lithotomies were considered one of the most common and dangerous operations of the 18th century, so surgical tables were developed for hospitals to aid in these procedures. These tables had adjustable backrests to hold the patient in a semi-vertical position and restrained his arms and hands. Lithotomy tables were one of the first specialized furniture developed for surgical purposes.[1]

The objective of this project was to develop a device that will help the surgeon operate at a good height while having the ability to modify it at any time

Our project is based on the choice of a scissor lift table for its stability that ensures a certain safety for the patient and on the other hand this report will focus on the electronic side of our project and how to control the surgical table.

This brief contains 3 chapters:

The first chapter will contain a bit of history, some definitions, possible positions, and how to control them,

The second chapter will focus on the material used while going through the programming side of our work ,

To finish the third and final chapter will highlight the results of our research and the prospects for the future.

Chapter I: Overview of the surgical tables

1-Introduction :

The history of dental and surgical procedures reaches back to the Neolithic and pre-Classical ages. The first evidence of a surgical procedure is that of trepanation. Trepanation was the practice of drilling or cutting a hole through the skull to expose the brain. This was thought to cure mental illness, migraines, epileptic seizures and was used as emergency surgery after a head wound.[2]. This procedure was practiced as early as 3000 BC and continued through the Middle Ages and even into the Renaissance. [2]

Sushruta (c. 600 BCE) is considered as the "founding father of surgery". His period is usually placed between the period of 1200 BC - 600 BC. known as the main author of an important early medical text and the first text to represent the process of rhinoplasty **The Sushruta Samhita** . One of the earliest documented plastic surgeons, he also described procedures for treating hemorrhoids and fistulae, as well as cataract surgery[3]

And to this day all operations required a surgical table/bed which is improving over time.

2-Definition of a surgical bed /table :

A surgical bed is a bed equipped with mechanisms that can elevate or lower the entire bed platform, flex or extend individual components of the platform, or raise or lower the head or the feet of the patient independently. [4]

3-Surgical table/bed history :

- 1815: It’s estimated that sometime between 1815 and 1825 that beds with adjustable rails began to appear in Britain. They used a mechanical crank to move the side rails up and down. [5]
- 1874: Andrew Wuest and Son, a mattress company out of Cincinnati, Ohio, registered a patent for a mattress frame that could be elevated. This is considered the first iteration of the modern hospital bed we see today. [5]
- 1909: Willis D. Gatch, former chair of the Department of Surgery at the Indiana University School of Medicine, provided the next innovation in hospital beds. He invented the 3 segment adjustable bed, which today is often referred to as the Gatch Bed. It allows for the head and the feet to be elevated. [5]
- 1945: Push button hospital beds were invented by General Electric. One interesting feature about this design is it included a built-in toilet. The idea was to eliminate the bedpan. [5]

- 1946: Billionaire Howard Hughes was in a plane accident. Unhappy with the type of equipment available, he created a hospital bed to suit his needs. It had 6 sections and 30 electric engines. [5]
- 1949: Control and basic functions start to appear. [5]
- 1950s: In 1952, the Hill-Rom Company, developed a bed with an electric engine. In 1956, the company its first bed with full electrical functioning. [5]
- 1958: The Circ-O'lectric Bed was developed by Dr. Homer Stryker. Many other alternative models to address complex medical conditions were developed. [5]
- 1961: Beds with a basic bed controller were introduced. [5]
- 1974: Control panel on side rails was invented. [5]
- 1978: Side rails with remote control were added to the functionality of hospital beds. [5]
- 1980s: The 1980s was more focused on the development of the mattresses used in hospital beds. Therapeutic mattresses were developed. Some had position detectors and mechanics to weigh patient while still in bed. Patent exit monitoring devices were also developed. Other developments included a device to call the nurse and permanent cardiovascular monitoring became standard bed features. [5]
- 1983: An electric bed for home use outside of the hospital was introduced. [5]
- 1990s: Beds with more advanced functionality are developed. Mechatronic beds become a reality and developments are focused on more intelligent design. [5]
- 1993: Specific industry standards for electronic hospital beds are developed to protect against mechanic and electric hazards. [5]
- Today: Researchers continue to develop and build upon current hospital bed designs. New industry standards are evolving helping to make beds more functional, intelligent, comfortable and easy to use for patients and caregivers in a hospital, care facility, or private home setting.[5]

4-Type of bed :

People who are immobile or spend most of the day in bed may require medical beds that have more positions and options available to prevent pressure or bed sores. There are 3 main types of medical beds (manual, semi-electric and full electric) and others that are more specialized (such as bariatric beds). [6]

4-1-Manual: Hand cranks are used to raise and lower the head and the foot of the bed as well as to adjust the height of the bed. These beds are the most economical and a good choice for people that do not require frequent repositioning. Hand cranks are typically found at the foot of the bed and require a person that is physically capable to operate**.** .[6]

4-2-Semi-electric: An electric motor is used to raise and lower the head and foot portions of the bed. Patients and caregivers adjust the positioning by pressing buttons on a hand pendant. The height of the bed is adjusted manually with a hand crank and will require someone that is physically able use it. Semi-electric beds are ideal for people that do not require the height of the bed to be adjusted often but will benefit from touch of a button positioning**.** [6]

4-3-Full Electric: Height and positioning of the bed is controlled by the patient and/or caregiver with a hand pendant and does not require the use of a hand crank (unless there is a power outage, but many beds now have a back-up battery that would power the bed in emergency situations). Full electric beds make it easier for patients to get in and out of bed or raise the bed to a comfortable height for the caregiver to tend to the patient or to change linens. Often full electric beds have more positions available such as Trendelenburg, reverse Trendelenburg and cardiac chair positions.[6]

4-4-The bariatric bed :

A bariatric bed is a specialized bed that is used in the care of bariatric patients. Their construction and design are tailored to the needs of obese and overweight patients. [7]

These beds are often used alongside other bariatric appliances such as bariatric chairs, bariatric walking aids and bariatric ceiling hoist systems. Their dimensions are larger than regular beds[7]

Typically, a hospital bed is approximately 95cm-100cm wide but a bariatric bed would be 110cm and 120cm wide. Some bariatric beds have an adjustable width also the weight capacity of bariatric beds is greater than regular beds. [7]

Figure 1 : the bariatric bed

5-The Smart Bed :

The Original Smart Bed™ was produced in 1988 by Craig Miller Sr., then owner of American National. This mattress was the first self-adjusting smart bed to feature dual-air adjustability. It was controlled by a dial indicator with a numbered sequence from 0-10. No matter what setting you adjusted the smart bed to it would constantly check and maintain that same pressure setting to accommodate any sleeping position.[8]

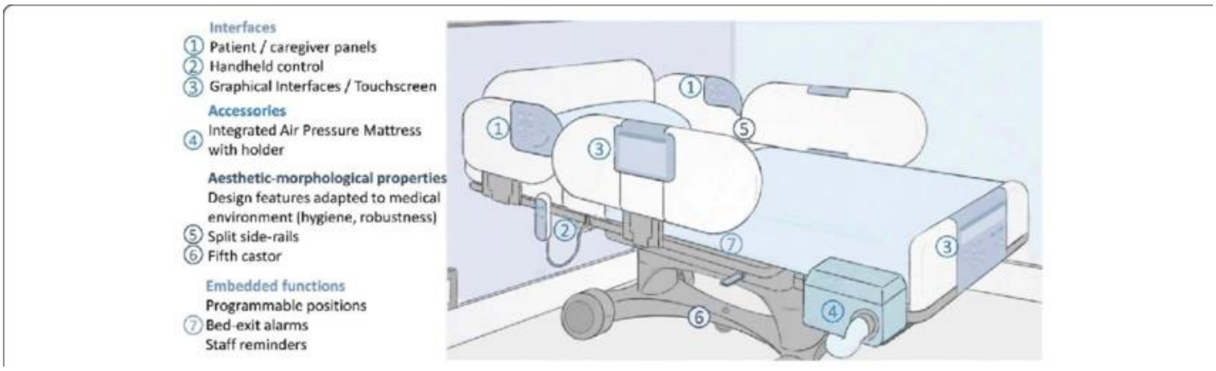


Figure 2 : the smart bed

6-Type of operating tables :

There are three major categories of operating tables: general surgical tables, orthopedic tables, and radiolucent imaging tables.[9]

6-1-General surgical tables :

They are used in a wide variety of surgical procedures like cardiovascular, pediatric, gynecology, gall bladder, and plastic surgery. A general surgical table does not have one specialty. Instead, it is designed for versatility and adaptability across a wide array of operations. General surgical tables can be adjusted for height and length, can be tilted to either side, and tilted horizontally. On most general tables, the head section is removable and can have a variety of attachable head rests.[9]

Figure 3 : general surgical table

6-2-The orthopedic table :

It is designed for easy manipulation and maneuverability required in orthopedic surgeries. To successfully perform an orthopedic surgery, surgeons need precise control and flexibility while maneuvering the patient. Orthopedic tables allow this access and flexibility of movement. [9]

Figure 4 : the orthopedic table

6-3-The radiolucent imaging table :

This type of table is designed for minimally invasive procedures that require fluoroscopy. Some of these procedures may be endovascular, vascular, or pain management. Radiolucent imaging tables are ideal for procedures that require clear, high-quality imaging.[9]

Figure 5 : the radiolucent imaging table

7-Two main classifications:

7-1-system and mobile :

An operating table system is basically made up of three components: an operating table column, the table top and the transporter. Modern operating table systems are available as both stationary and mobile units[11]

The advantage of the mobile operating table, on the other hand, is that the position of the table can be changed within the operating room. However, the foot of the table limits the leg space available to the surgical team. The individual segments of the table top can be easily removed and replaced. They also permit x-rays and conduct electricity.[11]

Another special feature of the operating table system is the ability to use appropriate interface modules to establish communication with diagnostics systems, for example, angiography, MR and CT. This is only possible with stationary columns since the systems require a fixed point.[11]

7-2-Comparison:

The operating table system has a number of advantages. Transport is easier since this unit is generally firmly secured to the floor and thus the foot and column of the unit no longer need to be transported. In addition, the transporter has light and large casters which are gentle not just to the floor. The entire operating area is more hygienic since the casters are not attached to the system, like on a mobile operating table. These are hard to clean and more unhygienic as a result. The operating table column can be rotated by 360° and offers ideal space for the feet of the team. The table top is, thanks to the use of x-ray-capable materials, almost completely radiolucent. The universal operating table is available as both stationary, mobile and moveable units. A mobile operating table is, however, used as a special table. The table top cannot be removed or replaced. Operation may, according to version, be manual, pedal or motorised.[11]

8- Common surgical positions for operating tables

Surgical procedures require proper patient positioning to keep the patient comfortable and safe during surgery, and to provide the surgeon with easy, unobstructed access to the surgical site. Many factors influence the decision of how to position a patient during a procedure:

-The patient's overall condition

-Length of procedure

-Techniques to be used during procedure

-Required exposure at operative site

-Expected anatomical and physiological changes associated with anesthesia

Furthermore, there are various risk factors that may lead to a patient's vulnerability to injury from improper positioning, such as:

.-Long procedures (3+ hours)

-Bone and joint conditions

-Skin breakdown due to aging

-Malnutrition, hypovolemia, anemia, paralysis, obesity, extreme thinness, or diabetes

The most common surgical positions are supine, Trendelenburg, reverse Trendelenburg, prone, lithotomy, sitting and lateral positions**.[9]**

-Supine:

This position is the natural position of the body at rest, making it the most common posture for surgery. Common complications associated with the supine positon are backaches and pressure-point reactions[9]

Figure 6 : supine position

-Trendelenburg:

The Trendelenburg position is a variation of the supine position. The upper torso is lowered and the feet raised, allowing for optimal visualization of the pelvic organs during laparoscopy and lower abdominal procedures[9]

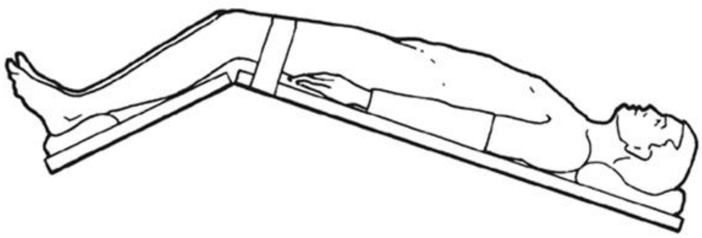


Figure 7 : trendelenburg position

-Reverse Trendelenburg:

Commonly known as the head-up and feet-down position, the reverse Trendelenburg is often used in head and neck procedures[9]

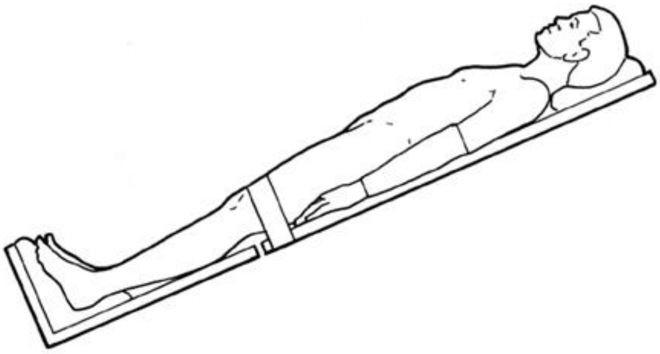


Figure 8 : trendelenburg position

-Prone:

In this position, the patient lies flat on their stomach and their head is turned to the side. This position is most commonly used for cervical spine, back, and rectal area procedures.[9]

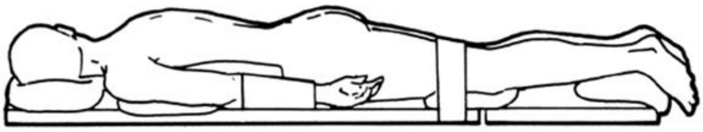


Figure 9 : Prone position

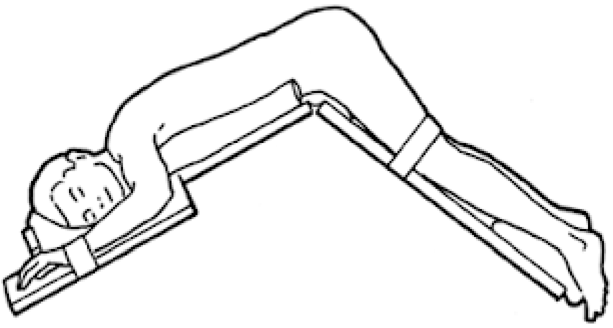


Figure 10 :Jacknife prone position

-Lithotomy:

While in the lithotomy position, the patient is in supine position and their legs are raised and abducted. Stirrups are needed for this position.[9]

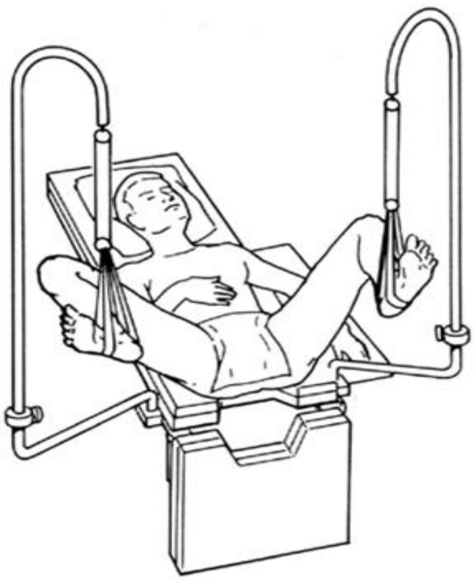


Figure 11 : lithotomy position

-Sitting:

Also known as Fowler's position, the patient in this position is sitting at a 90 degree angle. The knees are slightly flexed and the feet are placed against a padded foot board. This position is ideal for neurosurgery, facial operations, and some shoulder surgeries.[9]

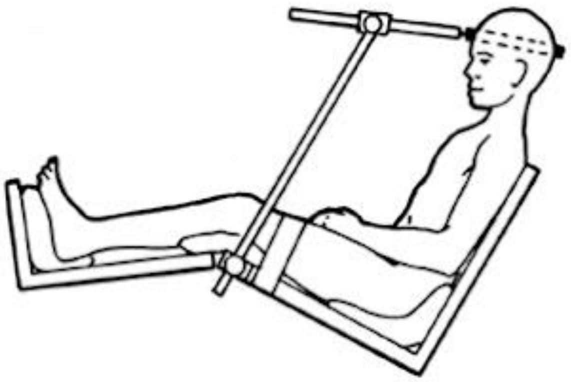


Figure 12 : sitting position

-Lateral:

The lateral position places the patient on the non-operative side to that surgery can be performed on the hip, chest, or kidney.[9]

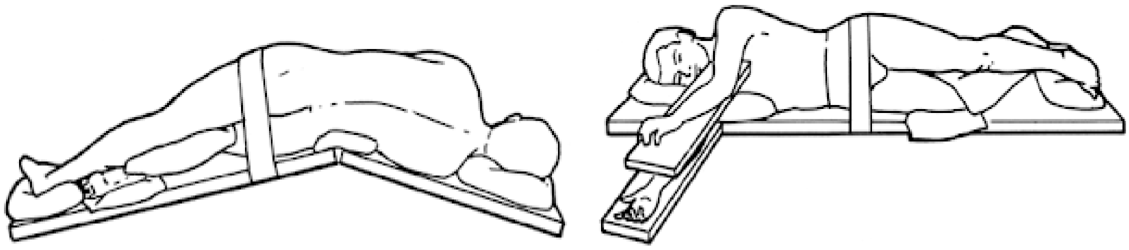


Figure 13 : lateral position

9-The movements performed by the universal operating tables :

1/ The adjustment of the total height, to allow the surgeon to work sitting or standing while having an ergonomic posture. the height adjustment of the universal operating tables varies between 23inches and 42".

2/The up-down vertical inclinations of the table, which make it possible to carry out the Trendelenburg and reverse Trendelenburg positions. On average, the angle vertical inclination of operating tables varies between 20 degrees and 30 degrees

3/The left and right lateral inclinations, which improve the field view of the surgeon, for example, to better observe the interior of the cavities. In general, the left and right tilt angle of operating tables varies between 20 and 30 degrees.

4/The individual adjustments of the table top sections upwards and down to achieve different surgical positions. Angles adjustment vary from one operating table model to another. in addition to other different positioning shapes shown in the figure below :

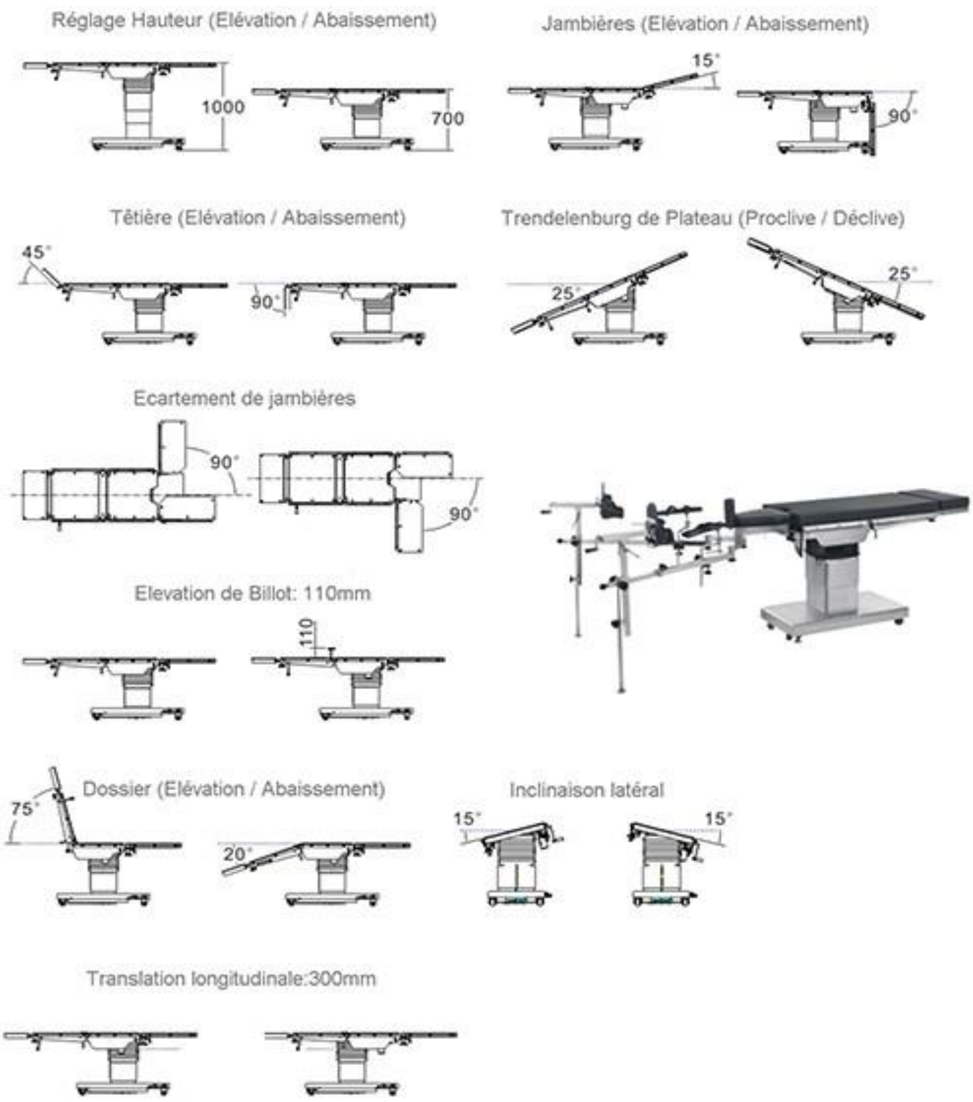


Figure 14 : The movements performed by the universal operating tables

10-Control of the movements carried out by the operating tables:

The control of the movements of the operating table can be carried out in 3 ways, as shown in the Figure: by a wired or wireless remote control, by a pedal and by a remote control panel allows, with the help of a single button, to carry out the desired adjustments on the operating table.[10]

The foot pedal on the operating table allows the surgeon to easily make small adjustments himself without using his hands. The functionality of the pedal is basic:

-Table height adjustment

-Table adjustment in Trendelenburg / Reverse Trendelenburg position

- Lateral inclinations of the table

As for the control panel located on the pillar of the operating table, it is used as emergency control in the event of remote control malfunction. It is positioned from securely on the abutment and can be locked to prevent accidental activation.[10]

Figure 15 : Control system of the table

11-Conclusion:

In this chapter we have presented the operating tables their types their movements we will now talk about the components necessary for our design.

Chapter II: Surgical tables Surgical table prototype requirements

I.Introduction

In this chapter, a framework is presented. We were able to understand the design's needs and generate numerous possibilities by researching previous projects, sketch the parts and put them together to make the final platform; this chapter also includes information about the electrical components we used and the electrical circuit diagram for the project.

II.hardware

1. ***Esp32*** : This device is the main control unit and the intermediate between the user and the surgical bed. The ESP32, is an excellent SoC with integrated Wi-Fi and Bluetooth connectivity. And this is a brief introduction about the ESP32 Microcontroller, its specifications, development board layout and a brief pinout as well.

Figure 16 : ESP 32

ESP32 is a low-cost System on Chip (SoC) Microcontroller from Espressif Systems, the developers of the famous ESP8266 SoC. It is a successor to ESP8266 SoC and comes in both single-core and dual-core variations of the Tensilica's 32-bit Xtensa LX6 Microprocessor with integrated Wi-Fi and Bluetooth.The good thing about ESP32 is its integrated RF components like Power Amplifier, Low-Noise Receive Amplifier, Antenna Switch, Filters and RF Balun. This makes designing hardware around ESP32 very easy as you require very few external components.

Another important thing to know about ESP32 is that it is manufactured using TSMC's ultra-low-power 40 nm technology. So, designing battery operated applications like wearables, audio equipment, baby monitors, smart watches, etc., using ESP32 should be very easy.

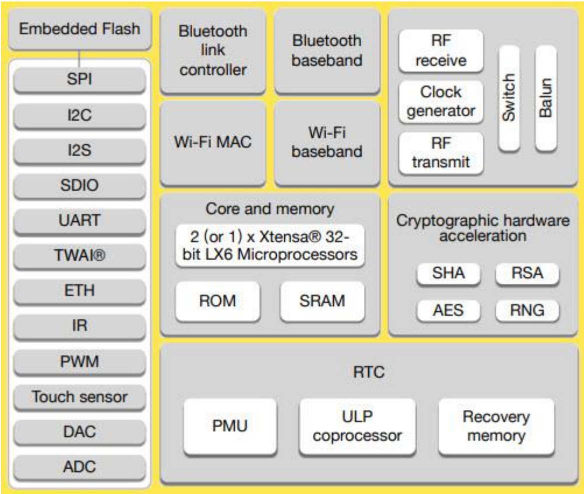


Figure 17 : ESP 32 representation

1.2. Specifications of the ESP32

ESP32 has a lot more features. So, this is a list of some of the important specifications of ESP32.

- Single or Dual-Core 32-bit LX6 Microprocessor with clock frequency up to 240 MHz.
- 520 KB of SRAM, 448 KB of ROM and 16 KB of RTC SRAM.
- Supports 802.11 b/g/n Wi-Fi connectivity with speeds up to 150 Mbps.
- Support for both Classic Bluetooth v4.2 and BLE specifications.
- 34 Programmable GPIOs.
- Up to 18 channels of 12-bit SAR ADC and 2 channels of 8-bit DAC
- Serial Connectivity include 4 x SPI, 2 x I^2C, 2 x I^2S, 3 x UART.
- Ethernet MAC for physical LAN Communication (requires external PHY).
- 1 Host controller for SD/SDIO/MMC and 1 Slave controller for SDIO/SPI.
- Motor PWM and up to 16-channels of LED PWM.
- Secure Boot and Flash Encryption.
- Cryptographic Hardware Acceleration for AES, Hash (SHA-2), RSA, ECC and RNG.

1.3. Layout of the ESP32

To see what a typical ESP32 Development Board consists of by taking a look at the layout of one of the popular low-cost ESP Boards available in the market called the ESP WROOM32NODEMCU Board.The following image shows the layout of an ESP32Board used.the board used has 38 Pins (19 pins on each side).

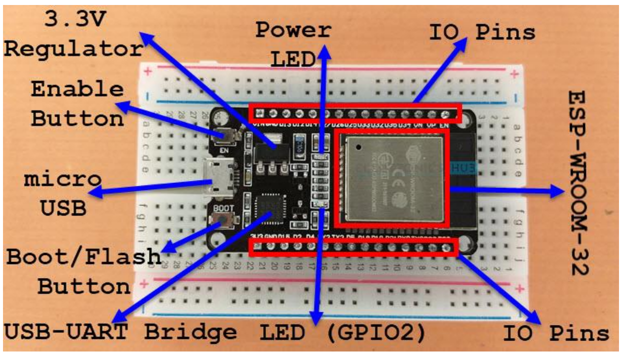


Figure 18 : ESP 32 components

the ESP32 Board consists of the following:

- ESP-WROOM-32 Module
- Two rows of IO Pins (with 19 pins on each side)
- CP2012 USB – UART Bridge IC
- micro–USB Connector (for power and programming)
- AMS1117 3.3V Regulator IC
- Enable Button (for Reset)
- Boot Button (for flashing)
- Power LED (Red)
- User LED (Blue – connected to GPIO2)
- Some passive components

An interesting point about the USB-to-UART IC is that its DTR and RTS pins are used to automatically set the ESP32 in to programming mode (whenever required) and also rest the board after programming.

1.4. *Pinout of the ESP32 board*

This is the pinout diagram of the ESP WROOM32 NodeMCU Board.

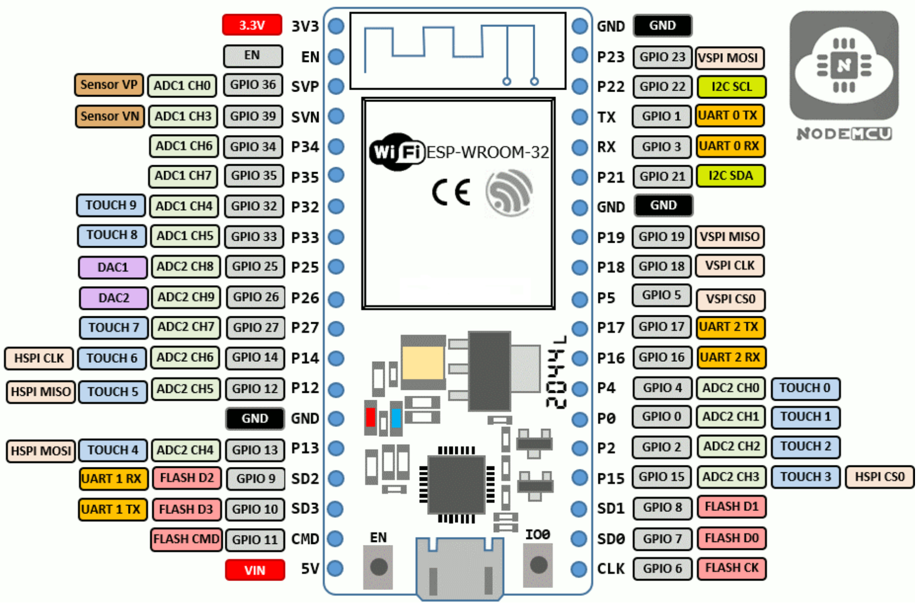


Figure 19 : ESP 32 pinout

This pinout is for the 38 – pin version of the ESP Board.

2. Stepper *motor*

This motor is responsible for regulating the height of the table by executing the commands coming from the ESP. Stepper motors can be found in a wide range of equipment, including

3D printers and CNC machines, as well as DVD drives, heating ducts, and analogue clocks. Despite their ubiquity, many experimenters are hesitant to use stepper motors because they appear to necessitate complicated wiring and programming.

Figure 20 : The stepper motor

We intend to dispel that myth in this thesis by demonstrating how simple it is to use a stepper motor with an ESP32. So, stick with us; we'll walk you step by step through all of this "complicated" stepper theory.

Stepper motor, a brushless DC and synchronous electric motor that divides a full rotation into a number of steps is known as a stepper motor. It takes one step at a time, each one equal in size. This allows us to precisely rotate the motor to a precise location. It can rotate in either a clockwise or counterclockwise direction.

They provide exact position control according to the number of steps per revolution for which the motor is designed, unlike other DC motors.They come in handy when you need to position something precisely. They are used in 3D printers to accurately position the printhead, as well as CNC machines to accurately position the cutting head. unlike DC motors, are controlled by applying DC electrical pulses to their internal coils. Each pulse advances the motor one step or a fraction of a step, the latter being referred to as "microstepping."

2.1. How a stepper motor works

A magnetized geared core is surrounded by a number of coils that operate as electromagnets in stepper motors. In a stepper motor, despite the number of coils on the surface, there are usually only two coils, separated into a number of little coils.The motor shaft can be made to move in discrete steps by precisely manipulating the current in the coils, as shown in the images below:

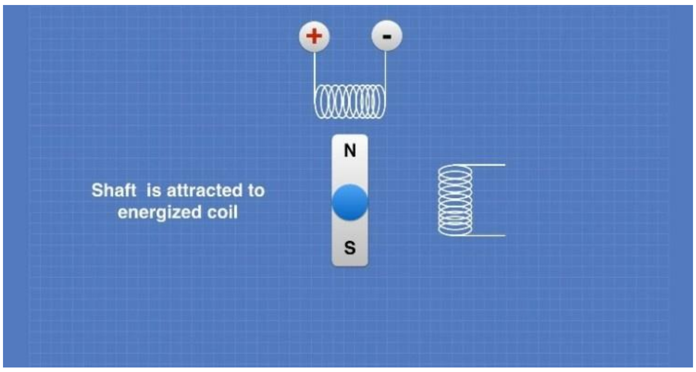


Figure 21 : diagram of the inner structure of a stepper motor

The coil at the top of the first diagram is energized by applying electricity in the polarity depicted. This coil attracts the magnetized shaft, which subsequently locks into place.When the power from the upper coil is withdrawn and applied to the other coil. The shaft is drawn to the second coil, where it locks into place.

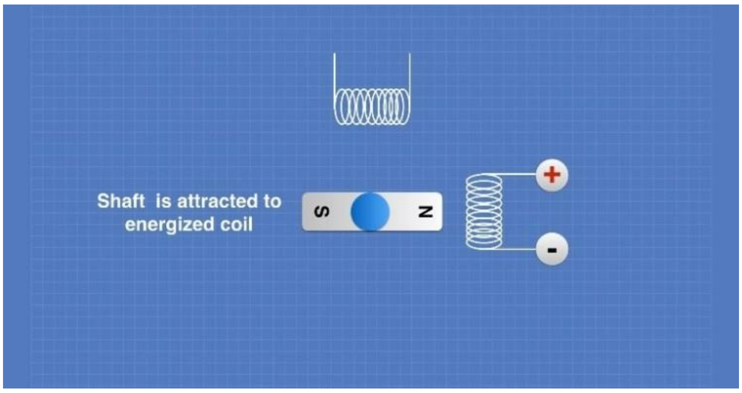


Figure 22 : diagram of the inner structure of a stepper motor

The difference between the two places is one step (a step in this instance is 90 degrees, but a stepper motor normally steps only a fraction of that). For clarity, the diagrams have been simplified).

2.2. Stepper motor specifications

Many first-time users are put off by the large number of specs given with some stepper motors. In reality, they aren't that tough to comprehend.Here are a few of the most important requirements:

Phase: The groups of the various coils in the stepper motor are referred to as phase. Even though a stepper motor contains multiple coils, they are all linked together and regulated in phases. Stepper motors with two, four, or five phases are widespread. A phase diagram is commonly included with a stepper motor, indicating the order in which the motor phases are driven.

Step Angle: This is the amount that the shaft of the motor will spin for each individual full step, measured in degrees, in some stepper motors this is referred to as Steps Per Revolution and the two figures are just different ways of expressing the same thing. A 1.8-degree step angle, for example, is a common rating for a stepper motor. Because a full rotation has 360 degrees, this equates to 200 steps per revolution (1.8 x 200 = 360).

Voltage: This refers to the voltage rating of the motor coils. It is also a function of the current rating and the coil resistance, which can be calculated using Ohm's Law.

current: Maximum current at the specified voltage. This is an important specification because it will help you choose a suitable driver and power supply for your stepper motor.

Resistance: The resistance of the coil, measured in ohms.

Inductance: The inductance of each motor coil is measured in millihenries. This is an important specification because inductance limits the maximum speed at which your stepper can be driven efficiently. Unipolar stepper motors typically have an advantage in this regard because they only use half a coil and thus have lower inductance than their bipolar counterparts.

Holding Torque: The amount of force produced when the stepper motor is activated.

Detent Torque: The amount of holding torque that can be expected when the motor is not turned on.

shaft style: The physical shape of the motor shaft is referred to as the shaft style. This is necessary in order to connect your stepper motor to gears, pulleys, and other external connections such as shaft couplers. There are several popular shapes, and the shaft length is important for obvious reasons.The following are some examples of common shaft types:

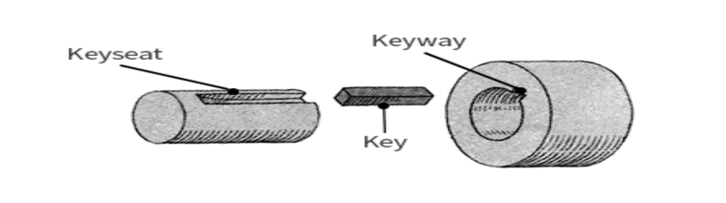
Style	description	picture
Round Shaft	pretty much sums it up!	
"D" Shaft	a "D-shaped" shaft that can be used to mount gears with set screws.	
geared shaft	has a gear etched into it.	
Lead-Screw Shaft	A screw-shaped shaft used in the construction of linear actuators.	
Shaft with Featherkey Notch		

Table 01 : Shaft types

Size: NEMA (<u>*National Electrical Manufacturers Association)*</u> is an international standards committee. A lot of confusion was created as you often heard people refer to a motor simply as a “NEMA 34”, which really only designates the size of the motor and not its other specifications such as voltage, current, step angle or even if it is bipolar or unipolar.The “34” in “NEMA 34” is the faceplate size, in the NEMA standard, the faceplate is the NEMA “number” divided by 10 in inches. So, a NEMA 34 motor has a faceplate approximately 3.4

inches wide. We're using is a 2-phase stepper motor so it has 4 wires each two connected to a coil. Red and Green connected to one coil; black and Blue connected to other.

2.3. How to use NEMA 34 Stepper Motor

As this motor draws high current is it advised to use an appropriately powerful stepper motor driver instead of directly controlling it. for our project we're using the DMA860H microstep driver.

2.4. Microstep driver DMA860H

The DMA860H is a fully digital stepper drive based on the latest motion control technology and built with an innovative DSP control algorithm. It has reached a high level of system smoothness, resulting in ideal torque and the elimination of mid-range instability. Its capability of motor auto-identification and parameter auto-configuration allows for speedy establishment of optimal modes with various motors. When compared to typical analogue drives, the DMA860H can drive a stepper motor with significantly less noise, temperature, and movement smoothness. DMA860H is a good solution for high-requirement applications because to its unique features.

DMA860H			
	Min	Max	Unit
Output current	2.4	7.2	A
Supply voltage	18	80	VAC
Logic signal current	7	16	MA
Plus input frequency	0	200	KHZ
Isolation resistance	50		MΩ

Figure 23: The DMA 860H **Table 02 : DMA 860H specification**

CURRENT TABLE

REF Current	PK Current	SW1	SW2	SW3
2.00A	2.40A	on	on	on
2.57A	3.08A	off	on	on
3.14A	3.77A	on	off	on
3.71A	4.45A	off	off	on
4.28A	5.14A	on	on	off
4.86A	5.83A	off	on	off
5.43A	6.52A	on	off	off
6.00A	7.20A	off	off	off

PULES/REV TABLE

Pules/rev	SW5	SW6	SW7	SW8
400	on	on	on	on
800	off	on	on	on
1600	on	off	on	on
3200	off	off	on	on
6400	on	on	off	on
12800	off	on	off	on
25600	on	off	off	on
51200	off	off	off	on
1000	on	on	on	on
2000	off	on	on	off
4000	on	off	on	off
5000	off	off	on	off
8000	on	on	off	off
10000	off	on	off	off
20000	on	off	off	off
40000	off	off	off	off

Table 03 : DMA 860H current table **Table 04 : DMA 860H pules /rev table**

2.5.Pin Assignment and Description:

There are two input connectors on the DMA860H: connector P1 for control signals and connector P2 for power and motor connections. The two connections are described briefly in the tables below.

Connector P1 Configurations

Pin Function	Details
PUL+	Pulse signal: In single pulse (pulse/direction) mode, this input represents pulse signal, each rising or falling edge active (set by inside jumper J1); 4.5-5V when PUL-HIGH, 0-0.5V when PUL-LOW. In CW/CCW mode (set by inside jumper J2), this input represents clockwise (CW) pulse. For reliable response, pulse width should be longer than 2.5μs.
PUL-	
DIR+	DIR signal: In single-pulse mode, this signal has low/high voltage levels, representing two directions of motor rotation; in CW/CCW mode (set by inside jumper J2), this signal is counter-clock (CCW) pulse. For reliable motion response, DIR signal should be ahead of PUL signal by 5μs at least. 4.5-5V when DIR-HIGH, 0-0.5V when DIR-LOW. Please note that rotation direction is also related to motor-driver wiring match. Exchanging the connection of two wires for a coil to the driver will reverse motion direction.
DIR-	
ENA+	Enable signal: This signal is used for enabling/disabling the driver. High level (NPN control signal, PNP and differential control signals are on the contrary, namely low level for enabling.) for enabling the driver and low level for disabling the driver. Usually left **UNCONNECTED (ENABLED)**.
ENA-	

Table 05 : Connector P1 configuration

Connector P2 Configurations

Pin Function	Details
VDC	Power supply, 20~70 VDC or 24V—70VAC, Including voltage fluctuation and EMF voltage.
GND	Power Ground.
A+, A-	Motor Phase A
B+, B-	Motor Phase B

Table 06 : Connector P2 configuration

3. OPTOCOUPLER :

Figure 24 : The optocoupler

An opto-coupler is a device that consists of two electrically independent but optically coupled elements contained within a perfectly sealed envelope.An opto- coupler's role is to provide galvanic isolation (no electrical connection) between two electrical systems for a variety of applications such as:

- Data transmission interface.
- Control of Low Voltage structures (EDF Sector).
- Power variation.
- Either to capture data through light:
    - Obstacles are present.
    - Following a path.

This device has two components: an input photo-emitter that emits visible or infrared light, and an output photo-receiver, photodiode, or, more commonly, phototransistor which is what we're using in this project.

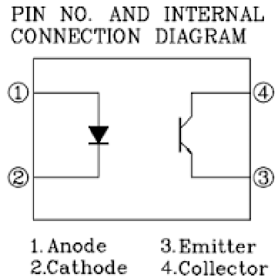


Figure 25: Optocoupler internal connection

4. Limit switch

A limit switch is an electromechanical device that is activated by the physical force exerted on it by an item.Limit switches are used to determine whether an object is present.These switches were initially employed to establish an object's limit of travel, thus the term Limit Switch.

2.1.How does a limit switch work?

Limit switches are types of automatic sensors that detect the position of an object through physical contact. An object moves the actuator that opens or closes a set of electrical contacts housed in the switch body, which are connected to equipment circuits by the connection terminals

Figure 26 : The limit switch

The main components of a limit switch are the switch body, the connection terminals, and the actuator.The **switch body**, or **contact block**, includes the enclosure and electrical contacts.The **connection terminals** are where you connect the input and output wires. In most cases, a limit switch is connected to some sort of control circuit (e.g. relays or programmable logic controllers).The **actuator** physically touches the target object and will open or close the switch contacts. Most limit switch actuators are either plungers or levers, which are also called "arms," and may have roller operating heads.

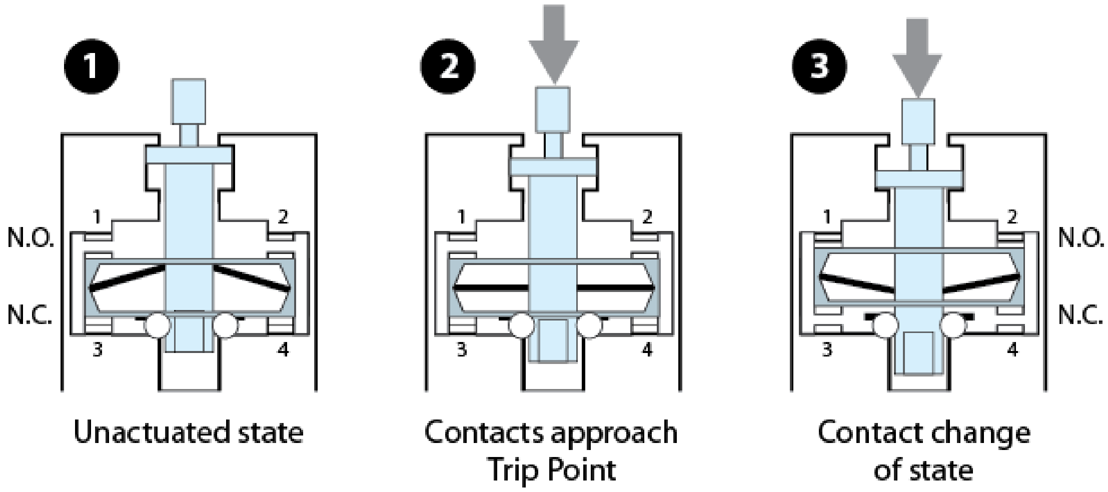


Figure 27 : limit switch different states

III.Software

The approach of software design is presented in this chapter. The entire software implementation, as well as the Arduino IDE settings to adapt with the ESP32.

1. Arduino IDE:

1.1. Programming the ESP32

Many alternative development environments may be used to program the ESP32. Code may be written in either C++ (as with the Arduino) or MicroPython.Espressif released the Espressif IoT Development Framework, or ESP-IDF, to enable the usage of all ESP32 functionalities.The familiar Arduino IDE is a good place to start for beginners. While this is not the ideal environment for working with the ESP32, it does have the advantage of being a known program, which reduces the learning curve.For our project, we will use the Arduino IDE.

1.2. Getting started with Arduino IDE

Before we can utilize the Arduino IDE with an ESP32 board, we must first add them to the Arduino IDE Board Manager. You may use the following instructions, which are significantly easier, as long as you are using a current version of the Arduino IDE.

To use the ESP32, you must first add an extra source to the Arduino IDE Board Manager and then install the ESP32 boards. Espressif have provided a link to a JSON file that takes care of almost everything for you.

If you are not acquainted with JSON, a text file format allows structured data to be transferred between computers. It is comparable to XML in this regard.

To install the ESP32 boards into the Arduino IDE, follow these steps:

- Launch the Arduino IDE. Check that you are on version 1.8 or higher; if not, update your IDE to the most recent version.
- On the top menu bar, select the File menu.
- Select Preferences from the menu. This will bring up the Preferences dialog box.
- By default, you should be on the Settings tab in the Preferences dialog box.
- Look for the "Additional Boards Manager URLs" textbox.
- If there is already content in this area, put a comma at the end of it before proceeding to the next step.
- Copy and paste the URL below into the text box https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
- Click the OK button to save the setting.

The textbox with the JSON link in it is illustrated here:

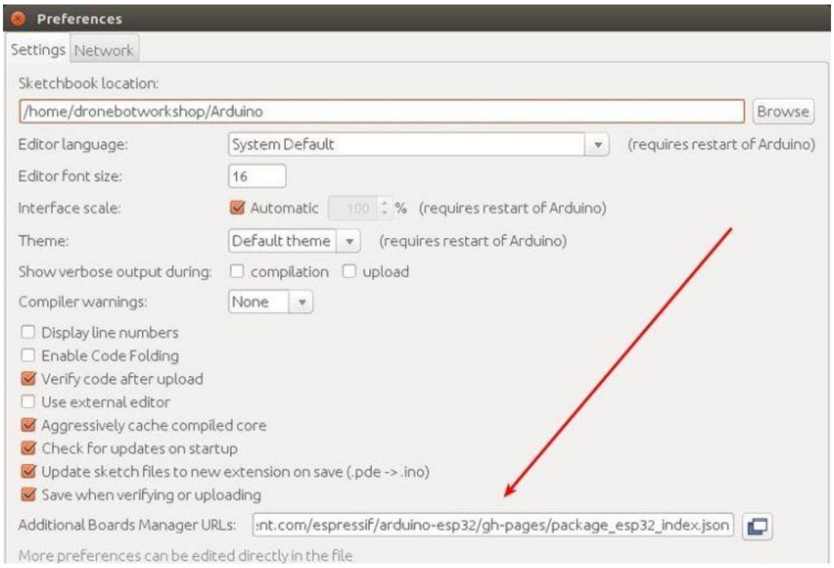


Figure 28 : setting up the JSON link with the Arduino IDE

Then, use the new entry to actually add the ESP32 boards to your Arduino IDE. You accomplish this by following the steps below:

- Within the Arduino IDE, On the top menu bar, select the Tools menu.
- Scroll down to the entry Board: (for example, Board: Arduino/Genuino Uno).
- When you highlight the Board: entry, a submenu will appear.
- Boards Manager is at the top of the submenu. Clicking on it will launch the Boards Manager dialog box.
- In the Boards Manager, type "esp32" into the search box.
- There should be a listing for "esp32 by Espressif Systems." Select this item and press the Install button.

- This will add the ESP32 boards to the Arduino IDE.

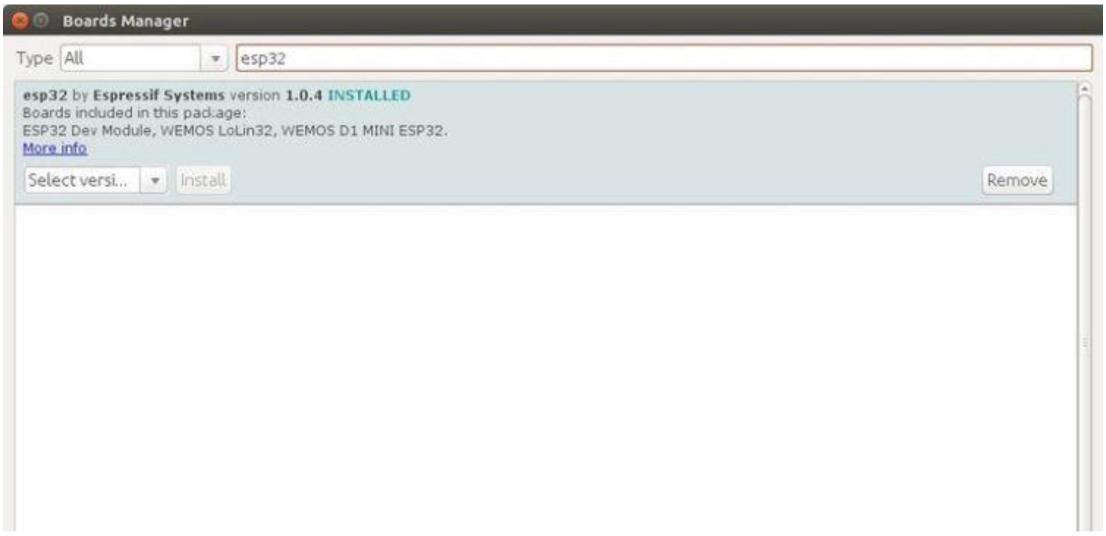


Figure 29 : library download window for Arduino IDE

Return to the Boards: submenu, and you should see a variety of ESP32 boards. You must choose a board that matches (or is equal to) the ESP32 board you purchased.

2. App inventor:

Figure 30 : MIT app inventor logo

MIT App Inventor is a web application integrated development environment originally provided by Google and now maintained by the Massachusetts Institute of Technology (MIT). It enables novice programmers to construct application software (apps) for two operating systems (OS): Android and iOS. It is open-source and free.

It employs a graphical user interface (GUI) that is very similar to the programming languages Scratch and StarLogo, allowing users to drag and drop visual items to construct an application that can run on Android smartphones.

App Inventor and the other projects are founded and guided by constructionist learning theories, which stress that programming may be used to engage powerful concepts through active learning.

3. Conclusion

This chapter provides a full explanation of our project, its components and the configurations made. This accomplishment required a significant amount of effort. We had to evaluate and overcome numerous challenges at the same time, including the mechanical difficulties of the platform, the design and programming of the electrical pieces, and the interface between them, all in a very short but intense period dedicated completely to make our project work.

Chapter III: Implementation and Results

I. Introduction

In this chapter we will go through the steps of building our prototype in details, from the frame design to the app/interface conception on the phone. we will walk step by step though the development of this project from sketches to tryouts to final product. We will also try to attempt comparing our design choices to others out there and break down inconveniences and benefits of each as best as we can. finally, we will go through costs and reflect on what to improve on our prototype in future.

II. Project Implementation

II.1. Bed design

We chose to go with a unique design inspired by a mixture of "scissor" lifting mechanism and screw/nut system. Our improved leveling mechanism topples the conventional scissor design via the following table:

Enhanced prototype	scissor
Better stability at all levels of height ✓	Loses stability due to moving parts ✗
Superior weight distribution ✓	Balances leans to one side the higher we go ✗
Complex and in-depth calculations ✗	Simple and straight forward ✓

Table 07: Comparative table 01

We went with the screw/nut system over the various others for the following reasons:

screw/nut	Hydraulic	Compressed air
No high pressure contained substances ✓	Possibility of leaks ✗	
One stepper motor ✓	Need hydraulic pump ✗	Need air compressor ✗
Simple and affordable components ✓	High maintenance and expensive parts ✗	
/	Expensive special oil ✗	Air ✓
High resolution stepper setting ✓	Low precision ✗	
Can be set to travel a certain height and automatically stopping and holding designated position ✓	Manual only (always needing an operator) ✗	

Table 08: Comparative table 02

The following figure shows the prototype of the surgical table made with different positioning angles.

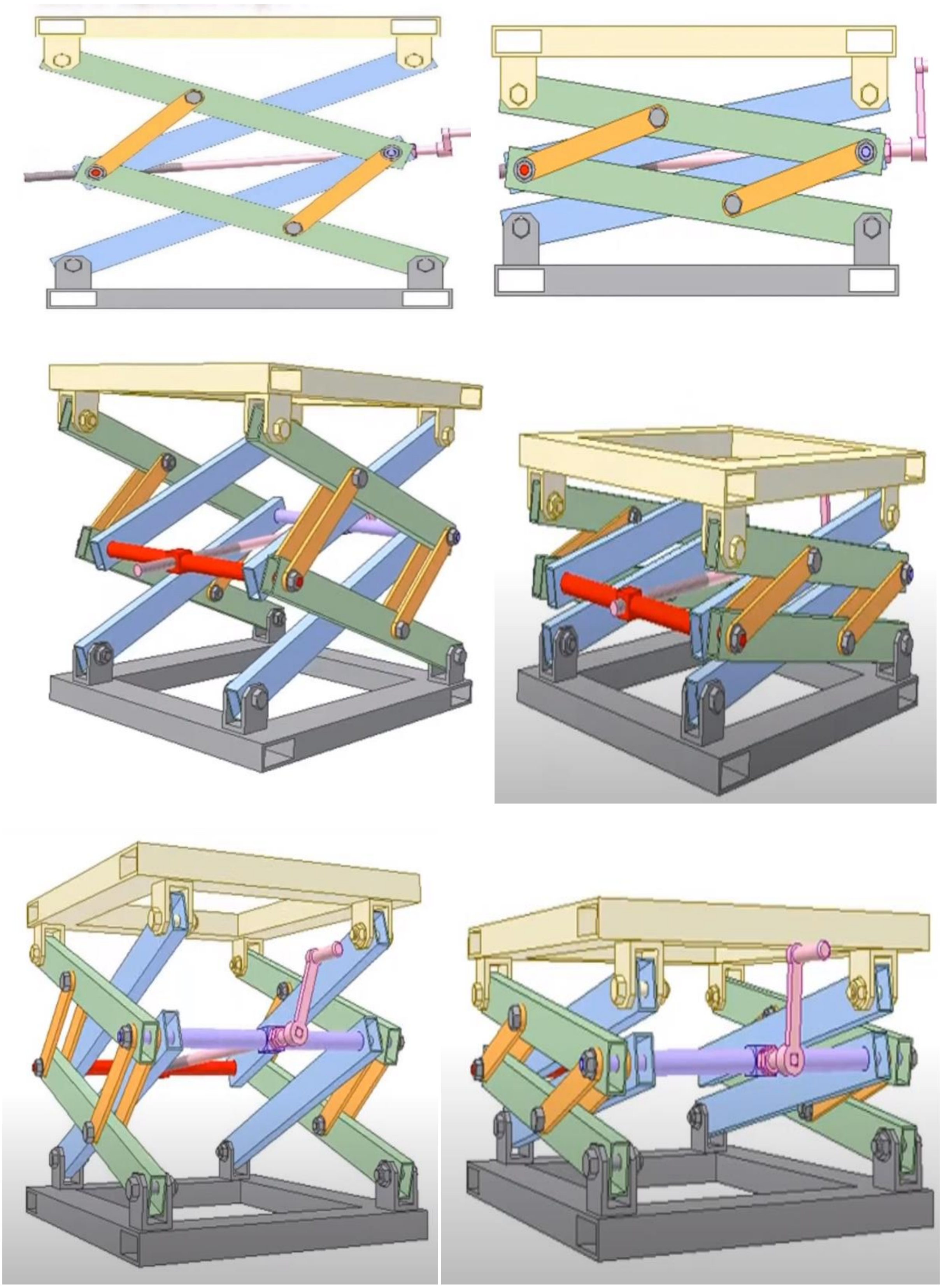


Figure 31: Different position of the operating table

II.2. Hardware configuration

In order to make the prototype for the surgical table, we followed the following steps:

Step 1: **getting familiar with stepper motors**. Stepper motors require special drivers to be operable. Stepper drivers have three sections that we need to familiar with.

a-wiring

- A High voltage section where a power supply and the stepper motor's phases are wired as follows:

Figure 32 : wiring the high voltage section

- A PA setting with switches to certain resolution and amperage parameters (check chapter 2)
- A Signal section where a µcontroller is plugged according to the pins initialized in the code.

b- testing: to avoid risking damaging expensive components of our projects, initial tests and tryouts were conducted with a NEMA17 stepper motor and an Arduino NANO

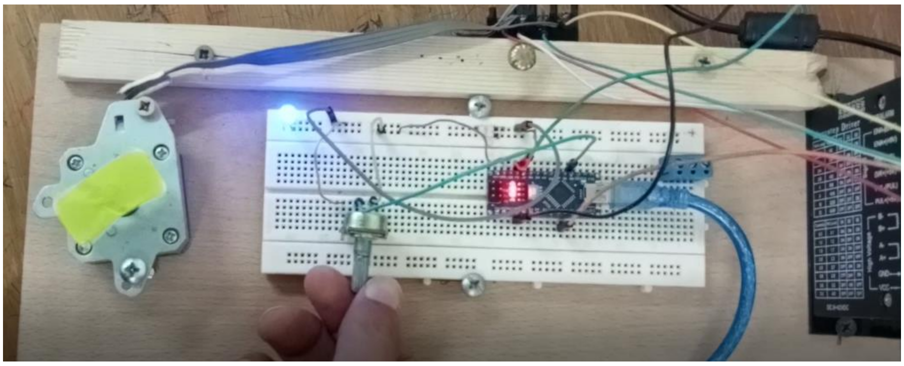


figure 33: testing circuit

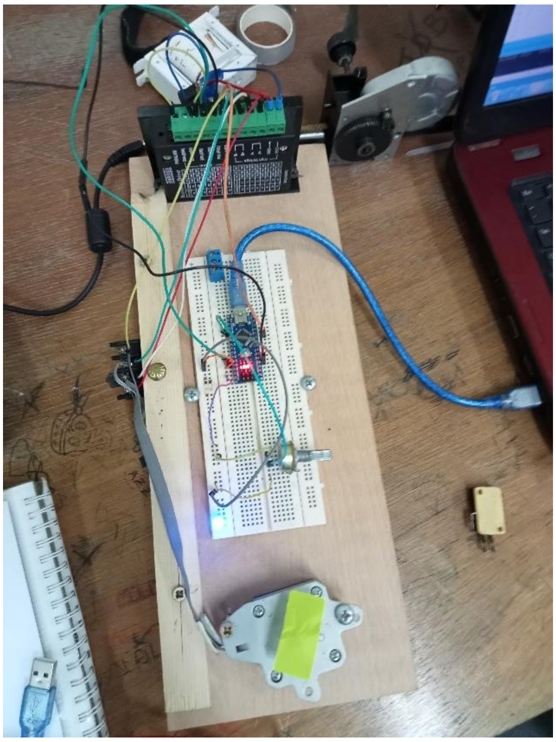


figure 34: testing circuit

step 2: setting up the bleutooth

- To test the Bluetooth on our ESP32 we will use a simple demonstration sketch that is included in the examples, one called *SerialToSerialBT*. The sketch is found in the *BlueToothSerial* submenu. The function of this program is to exchange data between the Arduino IDE Serial Monitor and an external Bluetooth Serial Terminal. What is written to one device is read by the other, and it works in both directions.
- Plug in the power cable to the ESP32, uploading the sketch, and pressing the enable button to start the program

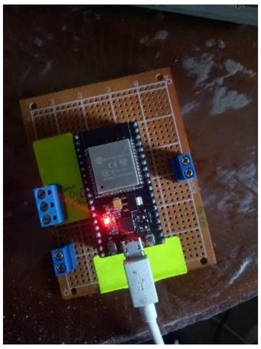


Figure 35: ESP32 circuit

- In order to test the sketch you'll need to have a serial Bluetooth terminal of some so58rt. If you have an Android device you can use the

"Serial Bluetooth Terminal App", which is free on the Google Play store.

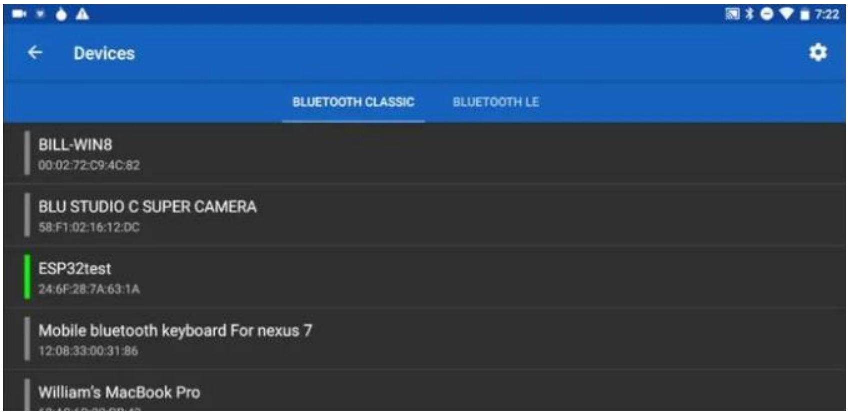


Figure 36: connecting interface of the serial bleutooth terminal app

- After pairing the devices a communication link is established to exchange texts as shown in the picture bellow:

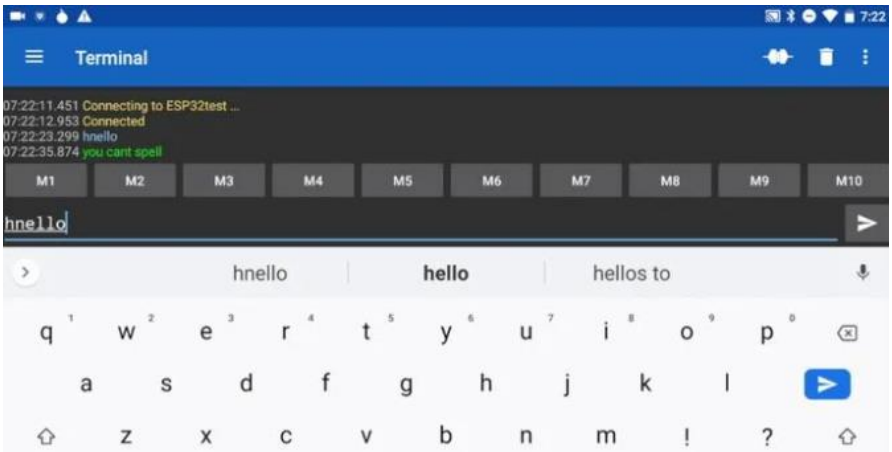


figure 37: example of data exchange

Step 3: controlling the big stepper motor with the ESP32.

- The wiring required to control the stepper motor goes as follows
  P25 = pul+ (step)
  P26 = dir+ (direction)
  P27= ena+ (enable)
  - Pul+, dir+, ena+ are slots in the configuration section of the stepper driver. Pul-, dir-, ena- go to the ground.

figure 38: fully wired circuit with NEMA34 and ESP32

1= 24V power supply

2= stepper motor driver

3= stepper motor NEMA 34

4= ESP32 µcontroller

- Despite the motor turning and functioning as it should we ran into a slight inconvenience: the signals emitted by the ESP32 are 3.3V and the stepper driver functions with 5V signals optimally. As a safety measure we opted to use an intermidiate circuit using optocuplers to ensure the smoothness and avoid any mishaps that might occur.

II.3. MIT app inventor

A google account is required for this step. Setting up the environment for before starting to work is pretty straight forward as we need to follow simple steps once we're in the https://appinventor.mit.edu/ site.

Figure 39: MIT app inventor home page

Clicking on the orange “create Apps!” button we’ll be asked to name a new project (provided that we don’t have prior projects open already), then we will be taken to our working space.

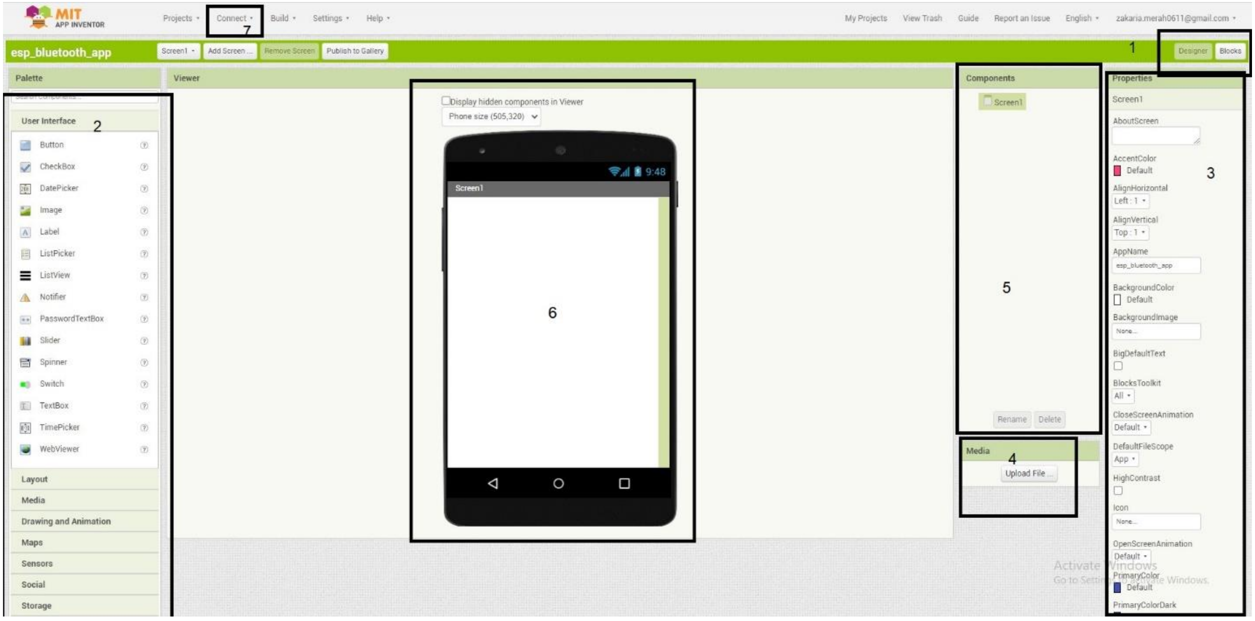


Figure 40: MIT app inventor interface

1.Designer/blocks selection: it’s close to what is called in the web development domain “front end/back end”. The **designer** window is where we build our app interface (buttons, scales, illustrations, status messages..etc).

1. User interface is the list of settings we choose from to design our app.
2. Properties is the parameter window of each setting from the previous menu.
3. The media added from the programmer (button shapes/colors, headers..).
4. This window is what has been already added from all the menus above in the app.
5. A virtual display or simulation where you can adjust what goes where
6. A dropdown menu where we can link this workspace to a device (phone/tablet) and see how the app looks like.

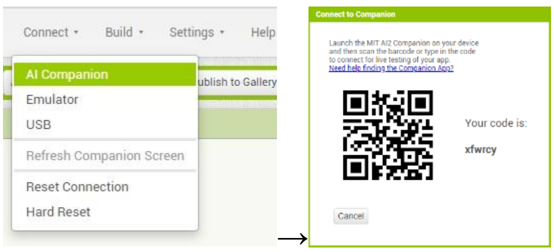


Figure 41: QR code of the application

Scan the QR code with the MIT AI2 companion app on the phone

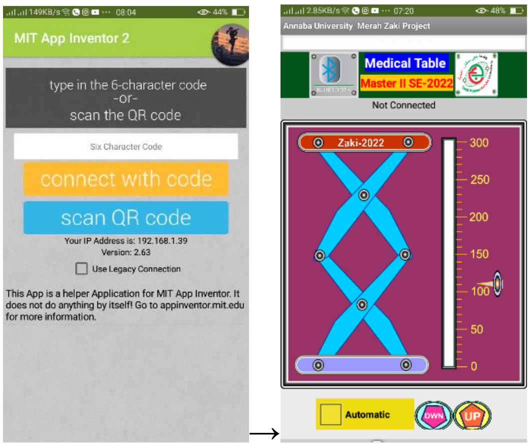


Figure 42: Operating table app

Example from our project:

Figure 43: button creation

Click and drag a button from the user menu to the phone screen

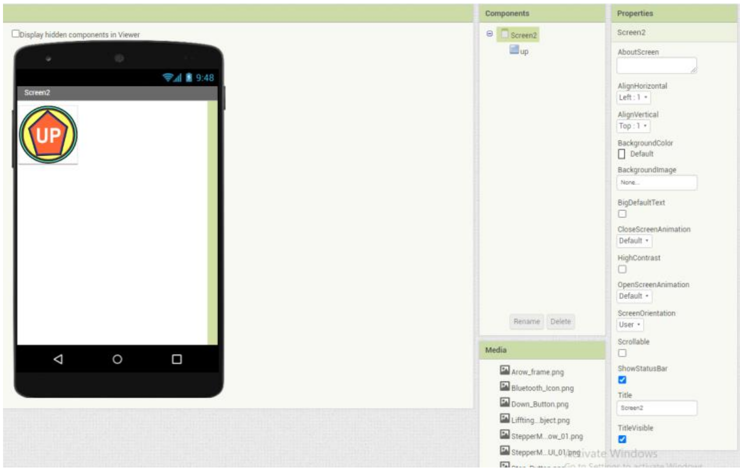


Figure 44: Button UP creation

Now we have our button in the component window and we added an image to it from the media that we previously uploaded.

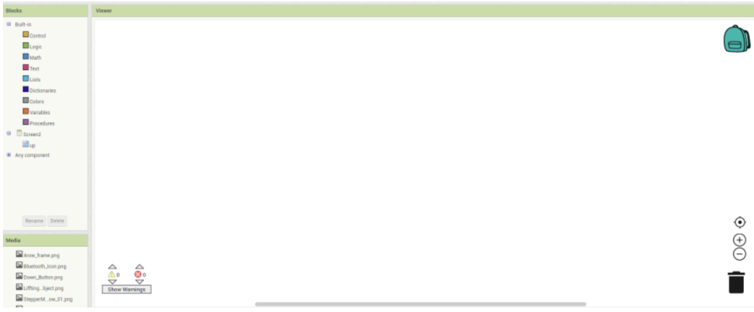


Figure 45: blocks part of the app

After that we go to the blocks window to associate a function to our button.

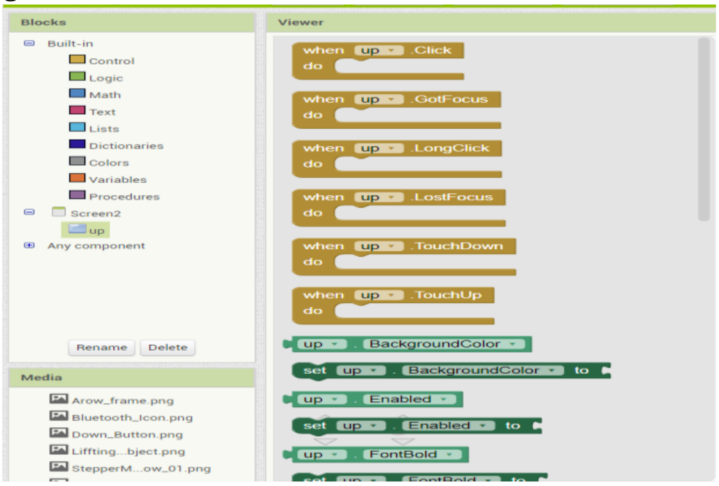
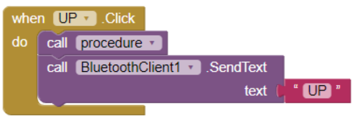


Figure 46: Configuration part of the app

Click on the button and you’ll get a pop up menu with all the build in function you can use, click and drag the one you need, in our case the top one.



this block is communicating with this line of code

```
if (dirRotation == HIGH) {
  dirRotation = LOW;
}
else {
  dirRotation = HIGH;
}
digitalWrite(dirPin,dirRotation);
delay(10);
state = "";
```

Figure 47: correspondence between app inventor code and Arduino code

III. Assembly

The following pictures are the final technical assembly of our prototype.

Figures 48 49 50 51: prototype assembly

IV. costs

Object	Quantity	Unit price	Total price
Stepper motor NEMA34	1	14.000 DA	14.000 DA
Stepper driver DMA	1	18.000 DA	18.000 DA
ESP32	1	2.200 DA	2.200 DA
Power supply 24V	1	2.500 DA	2.500 DA
PCB plate	3	250 DA	750 DA
Optocoupler	3	25 DA	75 DA
HW-130 power supply	1	750 DA	750 DA
Metal	/	/	2.800 DA
Copper ring (serves as a coupling)	1	2.500 DA	2.500 DA
threaded shaft + nut (16ø)	1	1.100 DA	1.100 DA
Component case	/	/	2.000 DA
Misc	/	/	1.500 DA
TOTAL	48.175 DA		

Table 09: Costs of our project

V. Conclusion

After numerous attempts and extensive research into all elements of this project, including the material selection, control, and mechanical design, we were able to construct this platform according to the specifications. As shown in the pictures above, the platform's operation and control operate effectively. Of course, this system has its limitations; for instance, if the speed is so high, the system will fail. As previously stated, the system will react in time to do the appropriate adjustments.

General conclusion

To summarize, this research had a beneficial influence and has had a major impact on us. It enabled us to expand and consolidate our skills in a variety of disciplines, including electronics, microcontrollers, programming, and mechanics. It was also an excellent way to learn about Esp32 and how to work with it.

From Creating the mechanical design and choosing the exact measurements, creating the circuit diagram, to coordinating the software to the hardware, we had to go back and forth to reach a desirable outcome

Great efforts have been undertaken to ultimately bring this project to a successful conclusion. Throughout this project, we encountered a number of technical challenges that we had to overcome. Some of these problems occurred during system construction and other problems occurred during system's programming. Despite the challenges and difficulties that occurred this year, the system was able to execute as expected.

VI. Perspective

our prototype has a lot of potential and room for improvement, as some perspectives we mention:

- retractable self-leveling tray for tools and utensils using MPU6050 module

Figure 52: MPU 6050 module

- folding extensions on both ends of the bed
- one adjustable folding side extension for headrest
- built-in charging stations for power tools (instead of dentists using compressed air tools that are prone to leaks and fails)
- built-in weight scale and medical devices (blood pressure meter/ heart rate monitor)
- on-board height control panel (buttons and display attached to bed)
- motorized wheels with obstacle detection/avoidance.

Bibliography

[1] Y. (2022, June 4). MEJA OPERASI (Operating Table). BLOGGER. Retrieved November 13, 2021, from http://yoyopiyo29.blogspot.com/2012_01_01_archive.html

[2] History of Surgery. (n.d.). Hartford Stage. Retrieved January 12, 2022, from https://www.hartfordstage.org/stagenotes/ether-dome/history-of-surgery

[3] Sushruta. (2021, June 12). In Wikipedia. https://en.wikipedia.org/wiki/Sushruta

[4] surgical bed. (2009). TheFreeDictionary.com. https://medical-dictionary.thefreedictionary.com/surgical+bed

[5] M. (2019, July 24). The History of the Hospital Beds and Their Development. MedPlus. Retrieved 02-22-01 from https://www.medplushealth.ca/blog/the-history-of-the-hospital-beds-and-their-development/

[6] Silver Cross. (2021, September 16). Find info about types of home medical beds & mattresses. Retrieved January 3, 2022, from https://silvercross.com/medical-beds/medical-beds-types/

[7] Rupp, S. (2021, April 27). Choosing bariatric beds: What you need to know. Electronic Health Reporter. Accessed November 23, 2021, from https://electronichealthreporter.com/choosing-bariatric-beds-what-you-need-to-know/

[8] The Original Smart Bed. (n.d.). Personal Comfort. Accessed January 3, 2022, from https://www.personalcomfortbed.com/the-original-smart-bed

[9] The ultimate guide to operating tables | knowledge center. (2017, February 28). STERIS. Accessed February 3, 2022, from https://www.steris.com/healthcare/knowledge-center/surgical-equipment/ultimate-guide-to-operating-table

[10] Acquisition of operating tables for the operating room. (2016).

[11] Operating table. (2020, July 24). In Wikipedia. https://en.wikipedia.org/wiki/Operating_table

[12] http://algo.tn/esp32/introduction/

[13]https://www.tutorialspoint.com/esp32_for_iot/esp32_for_iot_introduction.htm?fbclid=IwAR2RRL0CUMf2AKxBqQ1DAiYMf8YWhNZFQQ989eo_32ECllDFCcaoMQUSck#

[14] http://www.iotsharing.com/2017/05/introduction-to-esp32.html?m=1&fbclid=IwAR22HX5Jy7jR10ja2SZlIWA6C5RbCATFXxChv05ndhzAX5AXh3O9FvUH3_8

[15] https://dronebotworkshop.com/stepper-motors-with-arduino/

[16] https://dronebotworkshop.com/esp32-intro/

[17] http://www.leadshine.com/product_three/pro-2-12-19.html

[18] https://temcoindustrial.com/product-guides/switches-and-relays/electrical-switch-overview

[19] https://appinventor.mit.edu/