COMILLAS
UNIVERSIDAD PONTIFICIA
ICAI

GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES (GITI)

TRABAJO FIN DE GRADO

DEVELOPMENT OF AN AUTOMATED DRAWER CONTROL SYSTEM FOR TOOL TROLLEYS USED IN FUTURE PRODUCTION: INDUSTRY 4.0.

Autor: Alejandro Gallardo Sánchez

Director: Sriram Badri M.Sc

Madrid

Agosto de 2019

AUTHORIZATION FOR DIGITALIZATION, STORAGE AND DISSEMINATION IN THE NETWORK OF END-OF-DEGREE PROJECTS, MASTER PROJECTS, DISSERTATIONS OR BACHILLERATO REPORTS

1. Declaration of authorship and accreditation thereof.

The author Mr. /Ms. _Alejandro Gallardo Sánchez

HEREBY DECLARES that he/she owns the intellectual property rights regarding the piece of work: _Development of an automated drawer control system for Tool trolleys used in future Production: Industry 4.0

that this is an original piece of work, and that he/she holds the status of author, in the sense granted by the Intellectual Property Law.

2. Subject matter and purpose of this assignment.

With the aim of disseminating the aforementioned piece of work as widely as possible using the University's Institutional Repository the author hereby **GRANTS** Comillas Pontifical University, on a royalty-free and non-exclusive basis, for the maximum legal term and with universal scope, the digitization, archiving, reproduction, distribution and public communication rights, including the right to make it electronically available, as described in the Intellectual Property Law. Transformation rights are assigned solely for the purposes described in a) of the following section.

3. Transfer and access terms

Without prejudice to the ownership of the work, which remains with its author, the transfer of rights covered by this license enables:

a) Transform it in order to adapt it to any technology suitable for sharing it online, as well as including metadata to register the piece of work and include "watermarks" or any other security or protection system.

b) Reproduce it in any digital medium in order to be included on an electronic database, including the right to reproduce and store the work on servers for the purposes of guaranteeing its security, maintaining it and preserving its format.

c) Communicate it, by default, by means of an institutional open archive, which has open and cost- free online access.

d) Any other way of access (restricted, embargoed, closed) shall be explicitly requested and requires that good cause be demonstrated.

e) Assign these pieces of work a Creative Commons license by default.

f) Assign these pieces of work a HANDLE (*persistent* URL). by default.

4. Copyright.

The author, as the owner of a piece of work, has the right to:

a) Have his/her name clearly identified by the University as the author
b) Communicate and publish the work in the version assigned and in other subsequent versions using any medium.
c) Request that the work be withdrawn from the repository for just cause.
d) Receive reliable communication of any claims third parties may make in relation to the work and, in particular, any claims relating to its intellectual property rights.

5. Duties of the author.

The author agrees to:

a) Guarantee that the commitment undertaken by means of this official document does not infringe any third party rights, regardless of whether they relate to industrial or intellectual property or any other type.

b) Guarantee that the content of the work does not infringe any third party honor, privacy or image rights.
c) Take responsibility for all claims and liability, including compensation for any damages, which may be brought against the University by third parties who believe that their rights and interests have been infringed by the assignment.
d) Take responsibility in the event that the institutions are found guilty of a rights infringement regarding the work subject to assignment.

6. Institutional Repository purposes and functioning.

The work shall be made available to the users so that they may use it in a fair and respectful way with regards to the copyright, according to the allowances given in the relevant legislation, and for study or research purposes, or any other legal use. With this aim in mind, the University undertakes the following duties and reserves the following powers:

a) The University shall inform the archive users of the permitted uses; however, it shall not guarantee or take any responsibility for any other subsequent ways the work may be used by users, which are non-compliant with the legislation in force. Any subsequent use, beyond private copying, shall require the source to be cited and authorship to be recognized, as well as the guarantee not to use it to gain commercial profit or carry out any derivative works.
b) The University shall not review the content of the works, which shall at all times fall under the exclusive responsibility of the author and it shall not be obligated to take part in lawsuits on behalf of the author in the event of any infringement of intellectual property rights deriving from storing and archiving the works. The author hereby waives any claim against the University due to any way the users may use the works that is not in keeping with the legislation in force.
c) The University shall adopt the necessary measures to safeguard the work in the future.
d) The University reserves the right to withdraw the work, after notifying the author, in sufficiently justified cases, or in the event of third party claims.

Madrid, on 19th….. of August…

HEREBY ACCEPTS

Signed………………………………………………

Reasons for requesting the restricted, closed or embargoed access to the work in the Institution's Repository.

I, hereby, declare that I am the only author of the project report with title:

DEVELOPMENT OF AN AUTOMATED DRAWER CONTROL SYSTEM FOR TOOL TROLLEYS USED IN FUTURE PRODUCTION: INDUSTRY 4.0

which has been submitted to ICAI School of Engineering of Comillas Pontifical University in the academic year 2018/19. This project is original, has not been submitted before for any other purpose and has not been copied from any other source either fully or partially. All information sources used have been rightly acknowledged.

Fdo.: Alejandro Gallardo Sánchez Date: 15/07/2019

I authorize the submission of this project

PROJECT SUPERVISOR

B. Sriram

Fdo.: Sriram Badri Date: 15/07/2019

GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES (GITI)

TRABAJO FIN DE GRADO

DEVELOPMENT OF AN AUTOMATED DRAWER CONTROL SYSTEM FOR TOOL TROLLEYS USED IN FUTURE PRODUCTION: INDUSTRY 4.0.

Autor: Alejandro Gallardo Sánchez

Director: Sriram Badri M.Sc

Madrid

Agosto de 2019

DESARROLLO DE UN SISTEMA AUTOMÁTICO DE CONTROL PARA LOS CAJONES DE TROLLEYS DE HERRAMIENTAS USADOS EN LA FUTURA PRODUCCIÓN: INDUSTRIA 4.0.

Autor: Gallardo Sánchez, Alejandro.

Director: Badri, Sriram.

Entidad Colaboradora: RWTH Aachen University.

RESUMEN DEL PROYECTO

1. Introducción

En el contexto de la Industria 4.0, y enmarcado dentro del proyecto europeo A4BLUE, este proyecto tiene como principal objetivo la automatización de un cajón de un trolley de herramientas autónomo a través de un sistema RFID que interactúa con un Arduino.

El objetivo del proyecto A4BLUE es la creación de un espacio de trabajo eficiente y sostenible, adaptado a los nuevos requerimientos de la Industria 4.0. Para ello, el uso de sistemas automatizados y su interacción con los seres humanos es una de las piezas clave. En este sentido, se está desarrollando un trolley de herramientas automatizado en el Instituto de Ingeniería de Producción de Componentes de Movilidad Electrónica (PEM) de la Universidad RWTH de Aquisgrán. Por lo tanto, esta tesis contribuye a la creación de este trolley, creando un sistema RFID para controlar el acceso a las herramientas en su interior, estableciendo un campo de seguridad para evitar accidentes no deseados y estableciendo un sistema de “cierre fácil" del cajón. Un actuador de cadena situado en el interior del cajón y fijado en la parte posterior del trolley es responsable del movimiento del cajón.

El protagonista es el sistema RFID. Consiste en un lector colocado en la parte superior del cajón y un cierto número de tarjetas, incluyendo una tarjeta maestra y una tarjeta de reinicio. La tarjeta maestra puede añadir o quitar cualquier tarjeta del sistema, mientras que la tarjeta de reinicio tiene la función de reiniciar el sistema.

Se crea un campo de seguridad alrededor del cajón por medio de dos sensores ultrasónicos. Cumplen la función de detener el sistema cuando el cajón está en peligro de colisionar con un objeto o una persona.

Finalmente, un sensor de presión situado en la parte delantera del cajón cierra el cajón con un ligero toque después de que el acceso al trolley haya sido concedido.

Palabras clave: RFID, ultrasonidos, FSR, trolley, Industria 4. 0.

2. Metodología

El sistema completo de control de acceso al cajón se compone de cuatro subsistemas claramente diferenciados: sistema de control de acceso por RFID, actuador de cadena y relé para el movimiento del cajón, sistema de seguridad por ultrasonidos y sistema de cierre fácil mediante un sensor de presión.

a. Sistema de control de acceso por RFID

Está formado por un lector de tarjetas RFID, colocado encima del trolley para proporcionar un fácil acceso al usuario, y una serie de tags. El funcionamiento de este está basado en la utilización de la memoria EEPROM de la placa Arduino, en la cual se van a almacenar los datos UID de las diferentes tarjetas. Habrá siempre una tarjeta maestra, responsable de la inclusión o eliminación de otras tarjetas en el sistema, y una tarjeta de reinicio, la cual será siempre predefinida y tiene la función de borrar toda la información disponible en la memoria EEPROM. La lógica del sistema se describirá en los siguientes párrafos.

En primer lugar, el sistema comprobará si ha sido definida ya una tarjeta maestra. En caso negativo, se requerirá al usuario la introducción de esta. Una vez ha sido definida la tarjeta maestra, comienza la lógica del sistema:

1. Si se introduce de nuevo la tarjeta maestra, entramos en lo que llamaremos “Program Mode”. En dicho modo podemos añadir o eliminar tarjetas del sistema. Una vez dentro, si escaneamos una tarjeta que no estaba previamente en el sistema, será incluida al mismo. En caso de escanear una tarjeta ya presente, será eliminada del sistema. Para salir de este modo, basta con volver a escanear la tarjeta maestra.
2. Si introducimos una tarjeta perteneciente al sistema, el sistema nos concederá acceso. En caso de que la tarjeta escaneada no sea reconocida, el acceso será denegado.
3. Finalmente, si se introduce la tarjeta de reinicio, la memoria EEPROM se reiniciará al completo y tendremos que introducir una nueva tarjeta maestra.

Ilustración 1: Lector RFID

b. *Actuador de cadena y sistema de relés*

El responsable del movimiento del cajón es un actuador de cadena comúnmente empleado para la apertura y cierre de ventanas. Es utilizado para movimientos que tienen un camino fijo, y se limitan a extraer o recoger cadena. Hemos elegido este tipo de actuador por su reducido tamaño y porque aporta la suficiente potencia para abrir y cerrar el cajón a una velocidad adecuada. Para conseguir el movimiento de apertura y cierre, fijamos el motor al cajón, y mediante un agujero en la parte trasera del mismo, hacemos pasar la cadena hasta la pared posterior del trolley, donde también la fijamos.

Por otro lado, para controlar la dirección de movimiento del motor utilizamos una placa de dos relés. De esta forma, mediante dos pines digitales de la placa Arduino podemos controlar fácilmente la dirección en la que se mueve el cajón.

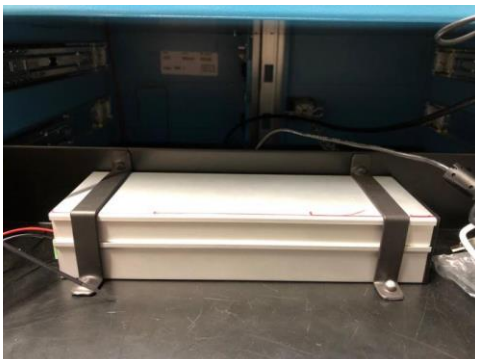


Ilustración 2: Posición del actuador dentro del trolley

c. *Sistema de seguridad mediante sensores ultrasonido*

Todo sistema mecánico necesita un sistema de seguridad para evitar posibles accidentes. Por ello, hemos desarrollado un sistema de seguridad mediante dos sensores ultrasonido situados a ambos lados del cajón, de forma que crean un campo de seguridad alrededor del mismo, tal como muestra la Ilustración 3. Los sensores cuentan con un Trigger, que manda una onda ultrasonido, la cual rebota en un hipotético objeto y es recogida por la Echo. El sensor proporciona el tiempo entre el disparo de la onda y la recogida de la misma. Utilizando la velocidad de propagación de la onda en el aire, podemos calcular la distancia a la que se encuentra el objeto.

Los sensores interaccionan con Arduino. La lógica detrás de los mismos es la siguiente: durante las operaciones de apertura/cierre, los sensores son activados, y permanentemente calculan la distancia con el objeto más cercano; se calcula la mínima de ambas (la más crítica) y se compara con un límite de seguridad impuesto; en caso de ser menor que dicho límite, el sistema se detendrá automáticamente.

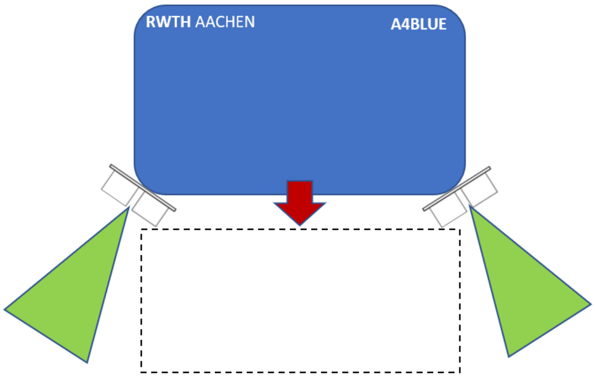


Ilustración 3: Esquema del campo de seguridad creado por los sensores ultrasonido

Ilustración 4: Disposición de los sensores en el trolley

d. *Sistema de cierre fácil mediante sensor de presión*

Este sistema ha sido ideado para facilitar el cierre del cajón una vez el acceso al mismo ha sido autorizado. Basta con un ligero toque del sensor para que el cajón se cierre. La lógica del mismo está basada en la imposición de una presión mínima, a partir de la cual el cajón procederá a la operación de cierre.

Ilustración 5: Sensor de presión en la parte frontal del cajón

3. *Resultados*

El conjunto de cuatro sistemas desarrollados e implementados en el trolley proporciona un sistema seguro de control de acceso y automatización del cajón. El funcionamiento del sistema en su conjunto ha sido verificado en aplicación real, simulando con una tarjeta maestra la figura del director de sección, y con dos tarjetas más las posiciones de un operador con acceso autorizado al cajón y de otro que no está incluido en el sistema. El sistema de inclusión y eliminación de tarjetas dentro del sistema RFID ha sido probado con éxito. La operación de apertura no dura más de 7 segundos, alcanzando una velocidad adecuada y segura para esta aplicación. Al mismo tiempo, se han simulado situaciones de emergencia en las que un objeto interrumpe el movimiento de apertura y el motor consigue reaccionar a tiempo y detener el sistema inmediatamente. Finalmente, el sistema de cierre fácil mediante el sensor de presión demuestra su eficacia a la hora de cerrar el cajón. Pudimos comprobar la utilidad y la mejora del sistema en términos de accesibilidad y facilidad de uso.

4. *Conclusiones*

Este proyecto encuentra su mayor utilidad en la implementación a gran escala. La tesis sienta las bases para una idea que, en su producción en serie, incluirá mejoras e innovaciones. Su implementación en la industria incluiría la automatización de los seis cajones del trolley. Para ello se necesitan seis actuadores, uno para cada cajón. El fabricante debe contar con un espacio al final de cada cajón para poder incluir el accionador. Por otro lado, una placa Arduino, o posiblemente un microcontrolador más potente, sería suficiente para controlar el sistema. Esto sería posible a través de un complejo sistema de relés, manteniendo el resto de la estructura. El fabricante debe tener en cuenta la salida del cable para conectar el lector RFID al sistema, que se colocará en la parte superior del carro. El sistema de control de acceso registrará las tarjetas incluidas en el sistema de cada uno de los cajones, dividiendo el almacenamiento de la memoria EEPROM en seis módulos diferentes. Cada operador tendrá acceso sólo a su cajón asignado. De esta manera, se evitará el extravío de las herramientas y se garantizará una correcta distribución de las herramientas en los cajones. Por último, se necesitan paredes huecas del carro a ambos lados del cajón, de modo que los sensores de ultrasonidos queden fijados en el interior. Los sensores FSR en la parte delantera de cada cajón permiten un cierre fácil.

DEVELOPMENT OF AN AUTOMATED DRAWER CONTROL SYSTEM FOR TOOL TROLLEYS USED IN FUTURE PRODUCTION: INDUSTRY 4.0.

Author: Gallardo Sánchez, Alejandro.

Director: Badri, Sriram.

Collaborating Entity: RWTH Aachen University.

PROJECT SUMMARY

1. Introduction

In the context of Industry 4.0, and framed within the European project A4BLUE, this thesis finds its main objective in automatizing a drawer of an autonomous tool trolley through an RFID system which interfaces with an Arduino.

The objective of the A4BLUE project is the creation of an efficient and sustainable workspace, adapted to the new requirements of Industry 4.0. For this, the use of automated systems and their interaction with humans is one of the key parts. Within this purpose, an automated tool trolley is being developed at the Institute for the Production Engineering of E-Mobility Components (PEM) of the RWTH Aachen University. Therefore, this thesis contributes to create this trolley, by creating an RFID system to control the access to the tools in it, setting up a security field to avoid unwanted accidents and establishing an 'easy closing' system of the drawer. A chain actuator located inside the drawer and fixed to the back of the trolley is responsible for the drawer's movement.

The main character is the RFID system. It consists of a reader placed on top of the drawer and a certain number of tags, including a master tag and a wiping tag. Master tag can add or remove any card from the system, while the wiping tag has the function of reseting the system.

A security field is created surrounding the drawer by means of two ultrasonic sensors. They fulfill the function of stopping the system whenever the drawer is in danger of colliding with an object or person.

Finally, a force resistive sensor placed at the front of the drawer closes the drawer by a slight touch after a granted access to the trolley.

Key words: RFID, ultrasonic, FSR, trolley, Industry 4.0.

2. *Methodology*

The complete drawer access control system consists of four clearly differentiated subsystems: RFID access control system, chain actuator and relay for drawer movement, ultrasonic safety system and easy closing system by means of a pressure sensor.

a. RFID based access control system

It consists of an RFID card reader, placed on top of the trolley to provide easy access to the user, and a series of tags. The operation of this is based on the use of the EEPROM memory of the Arduino board, in which the UID data of the different cards will be stored. There will always be a master card, responsible for the inclusion or deletion of other cards in the system, and a reset card, which will always be predefined and has the function of deleting all the information available in the EEPROM memory. The logic of the system will be described in the following paragraphs.

First, the system checks whether a master card has already been defined. If not, the user will be prompted to enter it. Once the master card has been defined, the system logic begins:

1. If the master card is inserted again, we enter what we call “Program Mode". In this mode we can add or remove cards from the system. Once inside, if we scan a card that was not previously in the system, it will be included in the system. If an existing card is scanned, it will be removed from the system. To exit this mode, simply rescan the master card.
2. If we introduce a card belonging to the system, the system will grant us access. If the scanned card is not recognized, access will be denied.
3. Finally, if the wipe card is inserted, the EEPROM memory will reboot completely and a new master card will have to be inserted.

Figure 1: RFID reader

b. Chain actuator and relay system

Responsible for the movement of the drawer is a chain actuator commonly used for opening and closing windows. It is used for movements that have a fixed path and are limited to extract or pick up chain. We chose this type of actuator because of its small size and because it provides enough power to open and close the drawer at an appropriate speed. To get the opening and closing movement, we fix the motor to the drawer, and through a hole in the back of it, we pass the chain to the back wall of the trolley, where we also fix it.

On the other hand, to control the direction of movement of the motor we use a two-relay plate. In this way, using two digital pins on the Arduino board, we can easily control the direction in which the drawer moves.

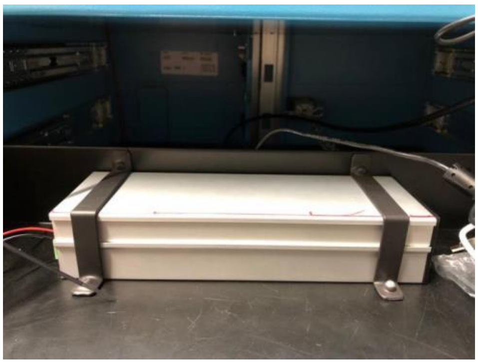


Figure 2: Position of the chain actuator inside the trolley

c. Ultrasonic based security system

Every mechanical system needs a safety system to avoid possible accidents. Therefore, we have developed a security system using two ultra-sound sensors located on both sides of the drawer, so that they create a security field around it, as shown in Figure 3. The sensors have a Trigger, which sends an ultrasound wave, which bounces off a hypothetical object and is picked up by the Echo. The sensor provides the time between triggering the wave and picking it up. Using the velocity of propagation of the wave in the air, we can calculate the distance at which the object is located.

Sensors interact with Arduino. The logic behind them is as follows: during opening/closing operations, the sensors are activated, and permanently calculate the distance to the nearest object; the minimum of both (the most critical) is calculated and compared with a imposed safety limit; if it is lower than this limit, the system will stop automatically.

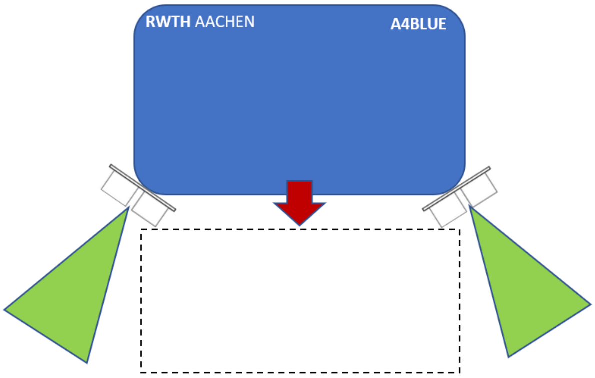


Figure 3: Sketch of the security field

Figure 4: Real implementation of the sensors

d. Easy closing of the drawer

This system has been designed to facilitate the closing of the drawer once access to the same has been authorized. A light touch of the sensor is enough to close the drawer. Its logic is based on the imposition of a minimum pressure, from which the drawer will proceed to the closing operation.

Figure 5: Force resistive sensor on the front part of the drawer

3. Results

The set of four systems developed and implemented in the trolley provide a safe and secure system of access control and automation of the drawer. The functioning of the system as a whole has been verified in real application, simulating with a master card the figure of the section director, and with two more cards the positions of an operator with authorized access to the drawer and of another who is not included in the system. The system of inclusion and elimination of cards within the RFID system has been successfully tested. Opening operation lasts no longer than 7 seconds, performing a suitable and safe speed for this application. At the same time, emergency situations have been simulated in which an object interrupts the opening movement of the drawer and the drawer manages to react in time and break the system immediately. Finally, the system of easy closing by means of the pressure sensor proves effectiveness at the time of closing the drawer. We were able to verify the usefulness and improvement of the system in terms of accessibility and ease of use.

4. Conclusions

This project finds its greatest utility in large-scale implementation. The thesis lays the foundations for an idea which, in its series production, will include improvements and innovations. Its implementation in the industry would include an automation of all six trol-ley drawers. For this, six actuators are required, one for each drawer. The manufacturer should count on a space at the end of each drawer in order to be able to include the ac-tuator. On the other hand, an Arduino board, or possibly a more powerful microcontroller, would suffice to control the system. Through a complex system of relays this would be possible, maintaining the rest of the structure. The manufacturer must consider the cable outlet to connect the RFID reader to the system, which will be placed on top of the trol-ley. The access control system will register the cards included in the system of each of the drawers, by dividing the storage of the EEPROM memory into six different modules. Each operator will have access only to his/her assigned drawer. This way, the misplace-ment of the tools will be avoided and a correct distribution of the tools in the drawers is ensured. Finally, hollow trolley walls are required at both sides of the drawer, so that the ultrasonic sensors are clamped inside. FSR sensors on the front part of each drawer will provide an easy closing.

Abstract

In the context of Industry 4.0, and framed within the European project A4BLUE, this thesis finds its main objective in automatizing a drawer of an autonomous tool trolley through an RFID system which interfaces with an Arduino.

The objective of the A4BLUE project is the creation of an efficient and sustainable workspace, adapted to the new requirements of Industry 4.0. For this, the use of automated systems and their interaction with humans is one of the key parts. Within this purpose, an automated tool trolley is being developed at the Institute for the Production Engineering of E-Mobility Components (PEM) of the RWTH Aachen University. Therefore, this thesis contributes to create this trolley, by creating an RFID system to control the access to the tools in it, setting up a security field to avoid unwanted accidents and establishing an 'easy closing' system of the drawer. A chain actuator located inside the drawer and fixed to the back of the trolley is responsible for the drawer's movement.

The main character is the RFID system. It consists of a reader placed on top of the drawer and a certain number of tags, including a master tag and a wiping tag. Master tag can add or remove any card from the system, while the wiping tag has the function of reseting the system.

A security field is created surrounding the drawer by means of two ultrasonic sensors. They fulfill the function of stopping the system whenever the drawer is in danger of colliding with an object or person.

Finally, a force resistive sensor placed at the front of the drawer closes the drawer by a slight touch after a granted access to the trolley.

Key words: RFID, ultrasonic, FSR, trolley, Industry 4.0.

Table of Contents

List of Figures

List of Tables

Abbreviations

AGV	Autonomous Guided Vehicle
RFID	Radio Frequency Identification
WZL	Werkzeugmaschinenlabor (Machine Tool Laboratory of RWTH Aachen)
PEM	Production Engineering of E-Mobility Components (Institute of RWTH Aachen)
FSR	Force Sensitive Resistor
EEPROM	Electrically Erasable Programmable Read-Only Memory
UID	User Identification
DTS	Driverless Transport System

1 Introduction and Objectives

Modern production systems as well as todays range of products challenge assembly workers. The aim of the A4BLUE project is to develop a new generation of sustainable, adaptive workplaces dealing with evolving requirements of manufacturing processes in cooperation with strong international partners. Automation mechanisms are to be introduced that are suitable for flexible and efficient task execution in interaction with human workers.

One of the major contributions of the WZL and PEM is the development and implementation of a use-case scenario, in which assembly operators are supported by augmented reality solutions and an automated tool trolley during the assembly of an electric vehicle. In preparation of the implementation of Industry 4.0 software solutions, a visualization model will be designed to document the production resources in a transparent and understandable manner. Furthermore, an assessment framework will be compiled to enable continuous monitoring of modern assembly workplaces in order to validate the implemented solutions. In addition, a methodology for sustainable design of future assembly systems will be developed, that identifies and realizes individually optimal and adaptive levels of automation. (14)

In the production of small series as well as customer-configured products, tools for different work tasks are used on different workstations at various in-stances of time. Appropriate scheduling optimizes the utilization of production resources and enables them to be used jointly in different areas of assembly or production. However, the transport of the tools associated with the requirement in different locations means additional organization and work effort. An AGV, in form of an autonomous tool trolley, is suitable for reducing this additional effort by making the production equipment available directly at the required location.

The aim of the thesis is to develop an RFID access control to automatically control the Tool Trolley drawers. For this purpose, a robust opening and closing mechanism has to be designed and constructed suitable for the Trolley, which interfaces with an Arduino to provide easy interface for the user. Furthermore, a security system will be implemented by means of several ultrasonic sensors which will create a safety area around the trolley in order to avoid unintended impacts when opening or closing operation. Lastly, an easy closing of the drawer will be achieved using an FSR sensor, after the sensor detects certain pushing pressure.

Fig. 1.1: A4BLUE Tool Trolley

2 State of the art

2.1 Industry 4.0. Nine pillars

Since the developments of the first manufacturing plants, new technologies and better practices have helped the manufacturing industry to grow. Known as the “fourth industrial revolution”, Industry 4.0 signifies the new wave of technology to mass produce more products easier and faster. Real time data can be downloaded at any point in the production process, which can identify any irregularities or setbacks. Connecting different machines with one another can vastly improve production, lower costs, and manage the products being created. (3)

Industry 4.0 is a transformation powered by nine foundational technology advances (as Figure below shows). In this transformation, sensors, machines, workpieces, and IT systems will be connected along the value chain beyond a single enterprise. These connected systems (also referred to as cyberphysical systems) can interact with one another using standard Internet-based protocols and analyze data to predict failure, configure themselves, and adapt to changes. Industry 4.0 will make it possible to gather and analyze data across machines, enabling faster, more flexible, and more efficient processes to produce higher-quality goods at reduced costs. This in turn will increase manufacturing productivity, shift economics, foster industrial growth, and modify the profile of the workforce—ultimately changing the competitiveness of companies and regions. (18)

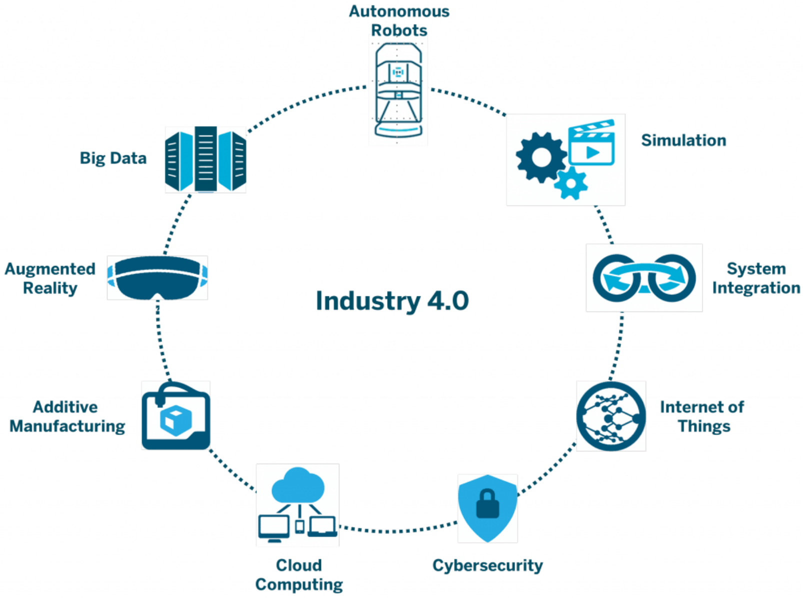


Fig. 2.1: Sketch of the nine pillars of Industry 4.0 (3)

2.2 RFID based security systems

The basic working principle of RFID systems is based on a reader, which sends an unmodulated signal to a tag. This tag reads an internal memory of data and changes the loading on the tag antenna in a coded manner corresponding to the stored data. The reflected signal, which has been modulated with the code information, is received by the reader again.

RFID is an integral part of contemporary life already used in a multitude of everyday situations: payments in means of transport (tolls, public transport…), inventories and warehouses, identification of animals, libraries, museums, timing in sport events, providing ski lift access, etc. Improved data accuracy, faster data collection, improved inventory/asset tracking, and reduced labor costs are just a few of the many improvements that RFID generates. (3)

This chapter will focus on RFID based access control systems. In manufacturing facilities, secure laboratories, company entrances, and public buildings, access rights must be controlled. RFID systems are a reliable and maintenance-free option to control access rights. Authorized personnel gain access to an area by passing an RFID tag over a reader. Location-specific access rights can be granted and modified when necessary, and lost tags can be blocked. (19)

Controlling access with RFID-enabled ID badges allows to:

- Deter theft
- Track personnel and visitors
- Limit access to restricted areas based on role

Compared to the traditional access control solution based on magnetic stripes, barcodes or proximity reads, RFID access control provides the following advantages:

- Increased reading distance
- Ability to update badge holder data without issuing a new ID
- Ability to read multiple IDs at once
- Cheap and easily programmable

The typical RFID-based controlled access solution consists of tags that contain unique identification data capable of granting or denying access to restricted areas, and a mechanism for reading the RFID tags at the access control points. These RFID tags can be placed on objects, or they may be embedded into paper or plastic ID cards. When the tags are read at the access control points, their data are validated against a database in a centrally controlled security system, and access is either granted or denied.

In addition to the security benefits of controlling access with RFID, such an application has economic benefits as well. Relative to other RFID applications, the access control application is well understood, and system components like tags and hardware are widely available. Prices of tags and equipment are falling, too, which is encouraging to organizations needing to upgrade their security systems.

However, this lock system has drawbacks which need to be considered too. To begin with, it is important to note that the system can be bypassed if an unauthorized person tailgates and authorized one, i.e., takes advantage of his/her authorized access to enter the building. However, this fact means also a disadvantage in other access control systems. A second disadvantage is that the system can be defeated: RFID tags can be cloned with readily available equipment. Anyone with an RFID reader can “skim” the data from a tag of interest and make a new ID badge with the desired access permissions. Because of these warnings, it is highly recommended that any RFID access control is compliment by a video surveillance system to prevent tailgating or ID theft. (4)

2.3 Smart trolleys

In comparison to fully automated systems, manual assembly work systems offer high flexibility but notably lower productivity. In order to increase productivity maintaining flexibility, future systems need to incorporate greater levels of automation which complement the capabilities of the human operators.

Despite an increasing digitalization and automation, manual labor is still required in assembly systems because many human qualities, such as the ability to cognition and problem solving, are still irreplaceable.

For this reasons, future assembly work systems will involve high levels of socio-technical integration and reconfigurability. Combine people and technology together to benefit from each other’s strengths, which not only requires consideration of technical specifications but also the needs of human operators. (20)

One actual application of conjunction of both human and digital work are the driverless transport systems (DTS), such as the smart trolley concerning this thesis. DTS, mostly used for physical logistics operations, differ in terms of vehicle designs and system structures according to sector and items being carried. They include Automated Guided Vehicles (AGV) for which the main issues are guide-path design, vehicle routing, vehicle requirements, idle-vehicle positioning, battery management, vehicle scheduling and deadlock resolution. AGV are commonly used in the automotive industry:

- At Daimler all required parts are picked/sorted and then brought to the final assembly line by driverless transportation vehicles.
- Audi have progressed even more and established DTS to transport finished vehicles automatically through the plant.

A benchmarking survey conducted by (20) has shown that the importance of DTS across the automotive and aerospace sectors, as well as in Information and Communications

Technology (ICT) and manufacturing and industrial process sectors, is evenly distributed at a medium-high level. Moreover, results on current use and usefulness conclude high importance of DTS in broadly every industry, except for participants from smaller organizations (SME).

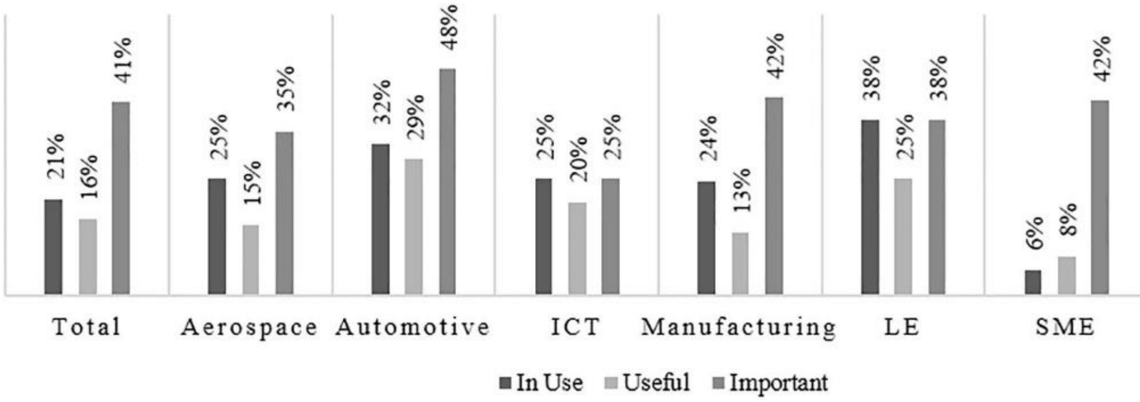


Fig. 2.2: Benchmarking survey conducted by (20)

The survey consists of two sections. The first consists of nine multiple choice questions that aim to obtain profile information to identify the sector and type of organization they represent, to ascertain what/how existing automation and technology is currently used in that domain, and to describe current limitations. The second section seeks opinion ratings about adaptive workplaces and related technologies. Three questions of this section in particular are used to reflect key indications of opinion: one asks what technologies are currently ‘in use’ within the participants’ company, another asks which technologies have provided most improvement to processes and are therefore most ‘useful’, and the third asks the respondent to indicate which technology they consider to be most ‘important’ in the future. (20)

2.4 Inventory tracking using RFID

Today, RFID technology is a fast-evolving market with clear future expectations. Within manufacturing processes in Industry 4.0 RFID is a key technology. For instance, RFID sensors work through radio transmissions and sensors that can provide information in real time. These gadgets can improve manufacturing logistics in several ways, and it takes minimal human effort to maintain control over the units. Not including costs, RFID can greatly improve the manufacturing stage of many businesses and improve production. (3)

While many manufacturers use some sort of system to catalog their products, RFID makes managing products more efficient. There are numerous ways RFID continues to work in sync with manufacturers including producing customized products, making cataloging more effective, and producing better results. All the information from the production floor is managed, and there are some plants that can run entirely on the machines. (3)

RFID inventory tracking has shown great improvement of production times and costs. The following are two examples of actual implementation of this technology: in a car and truck seat supplier in California and in the International Air Transport Association (IATA).

2.4.1 Johnson Controls

Johnson Controls (JC), located in California, supplies seats for car and truck models to New United Motor Manufacturers Incorporated (NUMMI), a joint production facility of Toyota and General Motors, housed in Fremont, California. JC receives orders every hour and delivers car seats 12 times a day directly to the production line of NUMMI, which requires that JC pay careful attention to delivery: its production system is designed to let JC unload car seats directly to its conveyors for immediate assembly. Human errors frequently occurred in the initial manual system causing NUMMI to be dissatisfied with JC. To reduce human errors, JC installed an RFID system on the truck seat production line and attached writable and readable tags on the bottom of the truck seat pallets that contained recipe information. The RFID technology was integrated with a touch-screen programmable logic controller (PLC) that operators used in each work process. As the pallet went through each station, the RFID reader read the information in the RFID tag and transmitted the information to the PLC, through which operators could check whether the car seat required any alteration. The PLC allowed the conveyor to move down only if the car seat did not need any work in the workstation or if necessary, work had been executed. When tasks were completed, the PLC sent the information to a RFID writer and it added the information to the tag underneath the car seat. Then the PLC would let the conveyor move down the line. This process repeated at the final station, where the PLC verified the seat's recipe information with the actual modification record. This combined application of RFID with the manufacturing system made a new production-control process possible. JC reports that RFID implementation saved time and dramatically reduced human errors. (17)

2.4.2 The International Air Transport Association (IATA)

The IATA has been using RFID technology to manage the huge costs of mishandled baggage. Approximately 1.7 billion bags pass through its system every year and roughly 17 million bags, are misallocated. On average, it costs the industry $100 to cope with one mishandled bag, resulting in $1.7 billion loss per year. In 2004, it chose RFID over the bar-code technology because of its accuracy and speed. The expected benefits were that RFID would significantly reduce the number of misplaced items and that the overall savings would amount to $1 billion a year. (17)

3 Hardware specifications

This chapter has the purpose of defining and explaining the basic features and functioning of each of the hardware and software components taking part in the project.

3.1 Arduino UNO

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online (4). To communicate with the board, Arduino programming language and the Arduino Software are used. The software can be easily downloaded from (1).

It consists of a microcontroller ATmega328, which stores the information introduced by the user through the Arduino Software. The board contains 14 digital pins, which can be defined either as input or output, and be set either high ('1') or low ('0').

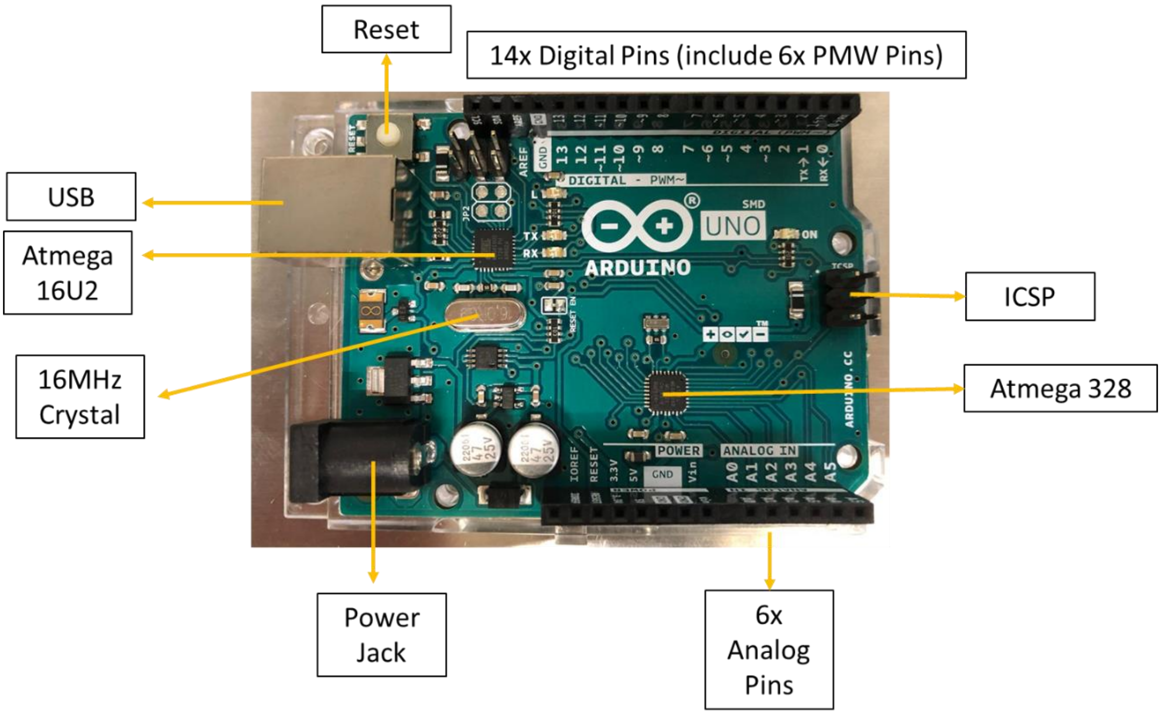


Fig. 3.1: Parts of Arduino UNO

The basic characteristics of the Arduino UNO board are summarized in the following table:

Tab. 3.1: Technical features Arduino UNO board

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

3.1.1 EEPROM Memory

EEPROM memory represents a key part of the RFID system functioning, since it is responsible for storing the information about the RFID tags. The initials EEPROM stand for Electrically Erasable Programmable Read Only Memory. The EEPROM is a form of non-volatile memory (the information persists after the power is turned off) where individual bytes of data can be erased and reprogrammed.

Its development came out of the EPROM memory, which was widespread in the late 1970s. These EPROM memories could be programmed, typically with machine software, and then later erased by exposing the chip to UV light if the software needed to be changed. (9)

A further development of this technology was achieved, and later in 1983 the EEPROM memory appeared. It could erase the stored data electrically. The data writing/erasing is performed byte per byte.

Arduino UNO board has an EEPROM storage memory of 1024 bytes (1kB). This memory can be accessed using the library EEPROM. This library enables you to read and write those bytes, as well as erasing and restarting the data stored in the memory.

3.2 Relay module

A relay is an electromagnetic switch operated by a relatively small electric current that can turn on or off a much larger electric current. The heart of a relay is an electromagnet (a coil of wire that becomes a temporary magnet when electricity flows through it). The relay can be understood as a kind of electric lever: switch it on with a tiny current and it switches on ("leverages") another appliance using a much bigger current. Relays work not only as a switch, but also as a security element in some electric circuits. (27)

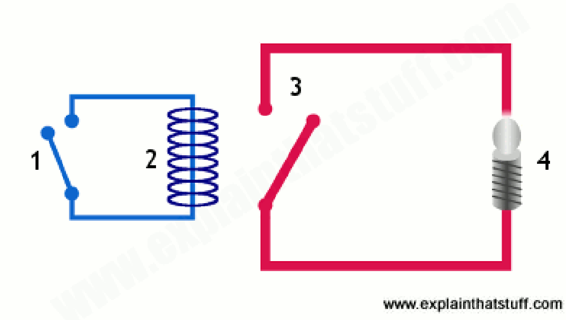


Fig. 3.2: Relay NO (normally open) mode (27)

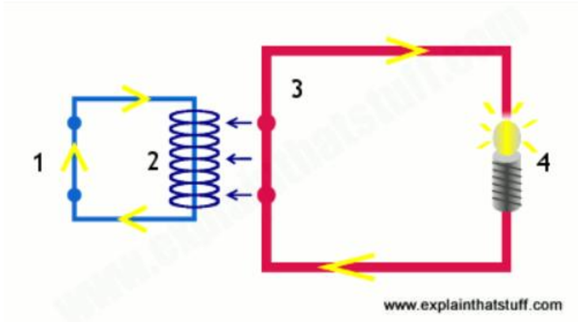


Fig. 3.3: Relay NC (normally closed) mode (27)

For this application, a two-channel relay will be implemented, for controlling each of the poles of the DC actuator.

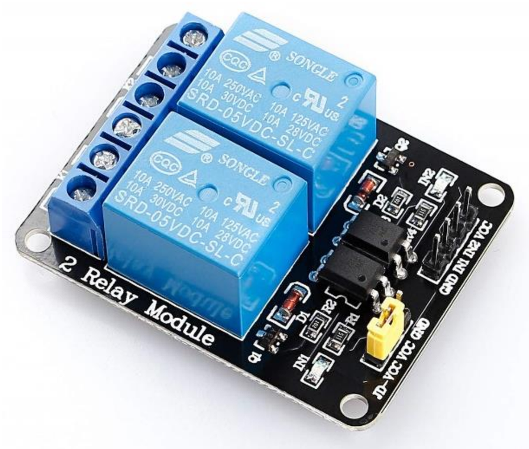


Fig. 3.4: Two-channel relay module 5V/220V

3.3 FSR sensor

FSR sensors measure a resistance change according to the pressure applied. They consist of two membranes separated by a thin air gap. The air gap is maintained by a spacer around the edges and by the rigidity of the two membranes. One of the membranes has two sets of interdigitated fingers that are electrically distinct, with each set connecting to one trace on a tail. The other membrane is coated with FSR ink. When pressed, the FSR ink shorts the two traces together with a resistance that depends on applied force. (16)

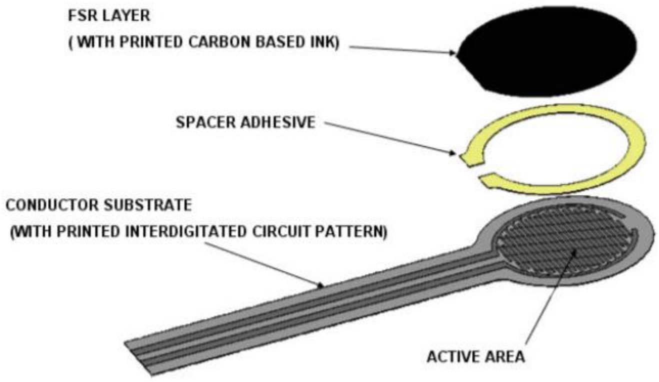


Fig. 3.5: Depicted Force Sensitive Resistor sensor (16)

Around the perimeter of the sensor is a spacer adhesive that serves both to separate the two substrates and hold the sensor together. This spacer typically has a thickness between 0.03mm and 0.15mm. This spacer may be screen printed of a pressure sensitive adhesive, may be cut from a film pressure sensitive adhesive, or may be built up using any combination of materials that can both separate and adhere to the two substrates. (16)

In Figure 3.6, the characteristic curve resistance-force of an FSR sensor is shown. Although the graph shows a quasi linear behavior, it should be noticed that a logaritmic scale is used. Therefore, force applied and resistance measured don't follow a proportional pattern.

As explained above, as force on the sensor increases, the electrical resistance of the FSR decreases. If there is no pressure applied, there is no contact between both FSR layer and conductive substrate and consequently, the resistance of the sensor will be infinite (open circuit). On the other hand, the more force is applied, the more active area comes in contact with FSR semiconductive area. This results in a reduction of the resistance.

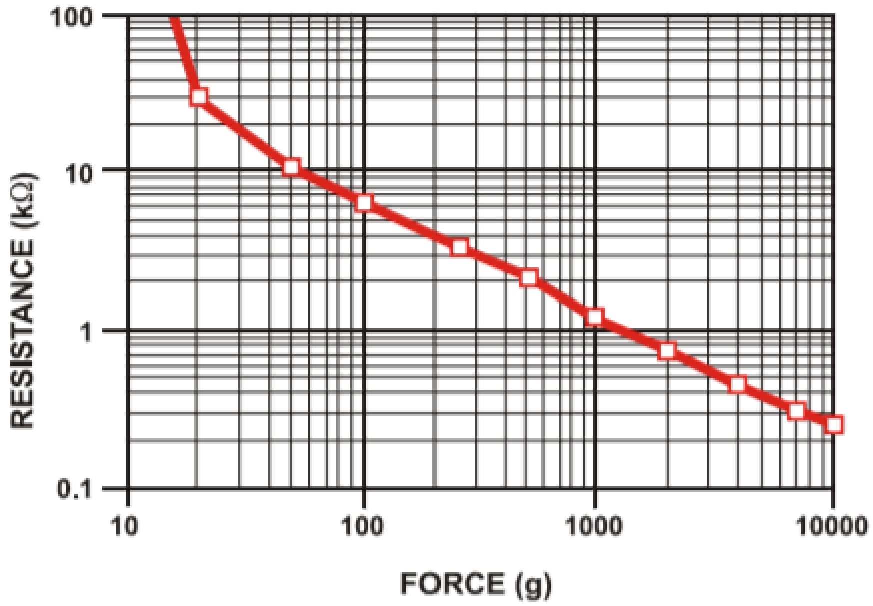


Fig. 3.6: FSR Force-Resistance curve (24)

3.4 Ultrasonic sensor HC-SR04

The HC-SR04 ultrasonic sensor consists of a transmitter and receiver. The transmitter emits 8 bursts of a directional 40KHz ultrasonic wave when triggered and starts a timer. Ultrasonic pulses travel outward until they encounter an object, which causes the wave to be reflected towards the unit. The ultrasonic receiver would detect the reflected wave and stop the timer.

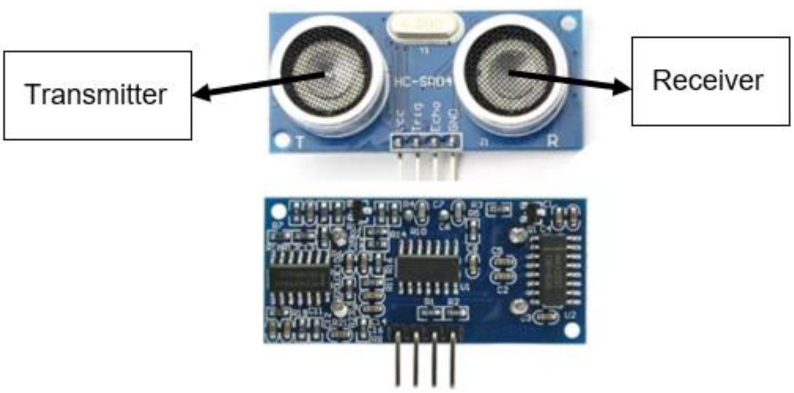


Fig. 3.7: HC-SR04 sensor (26)

The ultrasonic bursts are emitted by the HC-SR04 module in a maximum width angle of 30º. However, the effective angle may be less than 15º. This detection cone results in a measuring uncertainty. The receiver of the sensor will register the travelling time of the wave, but it will not give the angular position of the object as an output. Hence, it is not possible to determine by the output of the sensor whether the object is directly on the perpendicular line from the sensor or it slightly deviated. The object may be closer to the reference if it is not position exactly perpendicular to the sensor. For this reason, it is not possible to use this sensor for applications where distances must be precisely measured.

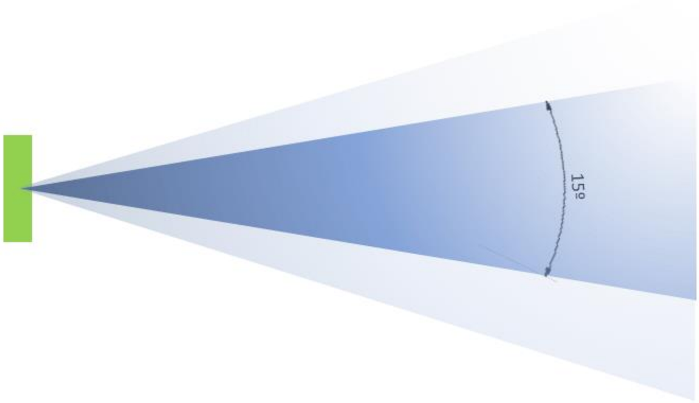


Fig. 3.8: Sketch of the sensor's detection angle

3.5 RFID reader

RFID (Radio Frequency IDentification) is a form of wireless communication that incorporates the use of electromagnetic or electrostatic coupling in the radio frequency portion of the electromagnetic spectrum to uniquely identify an object, animal or person. (25)

An RFID system consists of two main components, a transponder or a tag which is located on the object to be identified, and a transceiver or a reader. (12)

The RFID reader consists of a radio frequency module, a control unit and an antenna coil which generates high frequency electromagnetic field. On the other hand, the tag, usually a passive component, consists of just an antenna and an electronic microchip, so when it gets near the electromagnetic field of the transceiver, due to induction, a voltage is generated in its antenna coil and this voltage serves as power for the microchip. (12)

Fig. 3.9: Components of an RFID system (12)

Although the tag is usually a passive component, a distinction is made between active tags and passive tags:

1. Active tags and the chips contained in them need an external power source to be able to transmit. The RFID reader does not need a very high signal strength, since the greatest effort is made by the tag. The reading and writing capacities (up to 128 kb), as well as the range, are high.
2. Passive tags do not need an external power source, i.e., are activated with radio frequency emitted by the reader. In this case the signal strength must be very high. Reading and writing capabilities and range are far less than active systems.

Despite the apparent restrictions of passive RFID systems, these are widespread, mainly due to their lower unit cost of the tag, the lack of need for an external power supply and the consumption derived from it. (23)

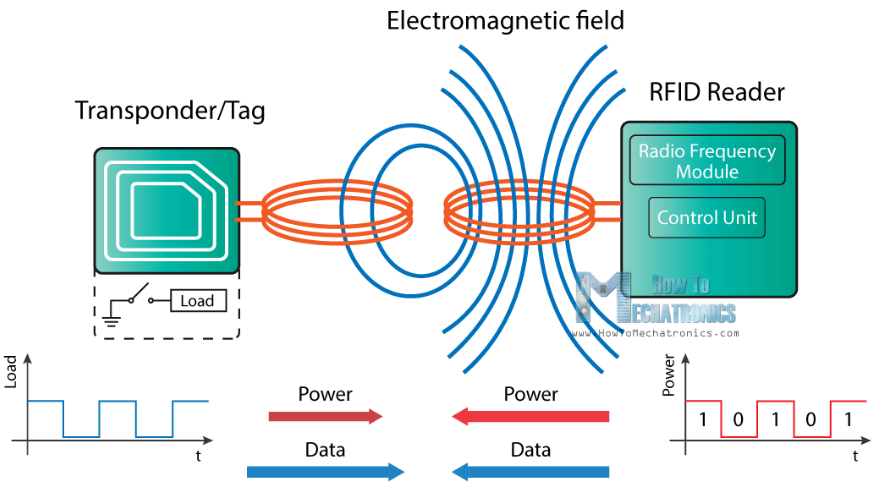


Fig. 3.10: Working principle of an RFID system (12)

As the tag is powered it can extract the transmitted message from the reader, and for sending information back to the reader, it uses a technique called load manipulation. Switching on and off a load at the antenna of the tag will affect the power consumption of the reader's antenna which can be measured as voltage drop. This changes in the voltage will be captured as ones and zeros and that's the way the data is transferred from the tag to the reader. (12)

On the other hand, each transponder/tag is distinguished by a four-byte UID, which is characteristic from each tag (assigned by the card manufacturer using a controlled database). The UID is transmitted to the RFID reader and can be printed on screen using an Arduino function. UID is the way to make a difference between different transponders.

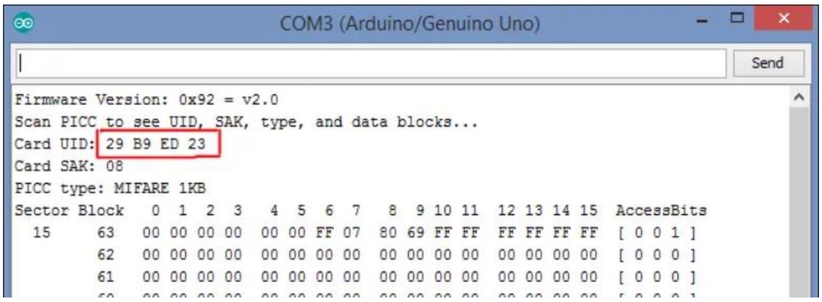


Fig. 3.11: Example of Card's UID

3.6 DC chain actuator

The DC chain actuator is the main character in the project. It is the real responsible of the opening and closing of the drawer. The motor works with 24V DC, and a maximum power of 55W. Thanks to its 380mm chain stroke length and a chain releasing speed of approximately 38mm/s, it can fully open the drawer in not more than 10 seconds. These motors have a compact design and are used to automate high level windows and light-weight roof vents and domes. A further datasheet is attached in Appendix I.

Fig. 3.12: Mingardi DC chain actuator

4 Set up and installation of hardware elements

4.1 Actuator and relay board

4.1.1 Relay configuration

At a first attempt, a motor driver (L298N) was used to control the change of the actuator direction. This driver would automatically change the polarity of the cables connecting with the actuator, whenever two input pins controlled by the Arduino changed from either “HIGH” to “LOW” or vice versa. However, the current through the driver was over the limit, and it overheated and burnt out.

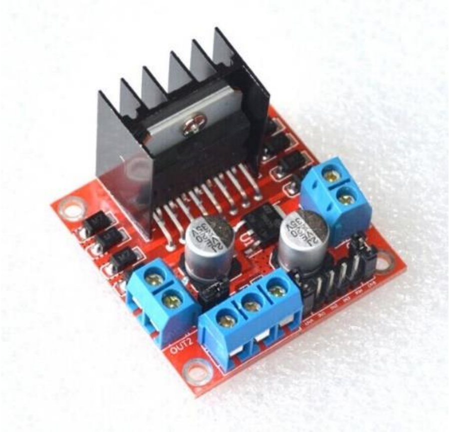


Fig. 4.1: L298N

As a secondary solution, a system with two relays was brought up. As a way to simplify the model, we used a board with both relays already integrated. The datasheet is attached in Appendix II.

Each relay switch has three inputs: NO (Normally open), COM (Common) and NC (Normally closed).

- Each of them would be connected to the positive pole and negative pole of the motor respectively: COM 1 to the motor positive pole and COM 2 to the motor negative.
- Both NO pins (Normally Open) will be connected with the positive pole of the 24V power supply.
- Both NC (Normally Closed) to the negative pole of the battery.

On the other hand, the connection between the relay board and the Arduino is the following:

- Arduino 5v pin - Relay module VCC pin

- Arduino GND pin - Relay module GND pin
- Arduino pin#7 - Relay module IN1
- Arduino pin#8 - Relay module IN2

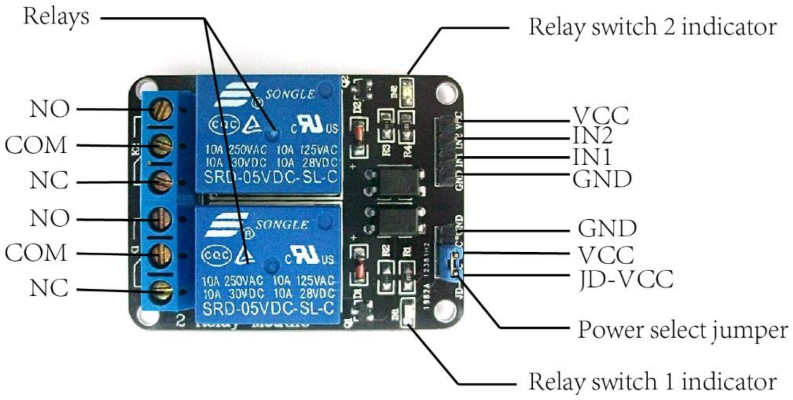


Fig. 4.2: Two-channel relay sketch

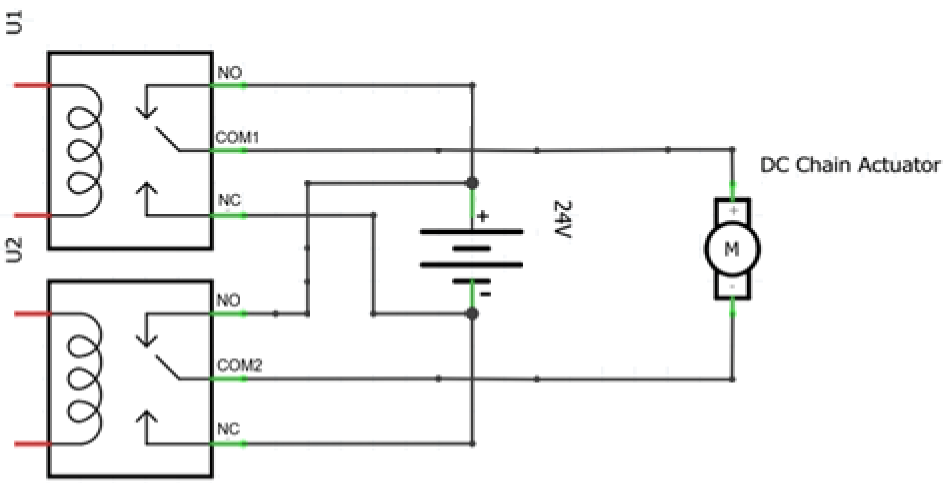


Fig. 4.3: Connection sketch of relay module, 24V battery and chain actuator

In this rough model, the connection between the relay module (represented as two separate relays U1 and U2), the 24V DC power supply and the actuator are shown.

The working principle is the following:

- If U1 is NO (no current flowing through the coil, controlled by an Arduino digital pin), and U2 is NC (current flow through the coil), the positive pole of battery and motor will be connected, as well as the negative poles of both devices. Consequently, the motor will operate in opening drawer direction.
- If U1 is NC and U2 is NO, the polarity of the motor is changed. Hence, the motor will change its working direction and the drawer will close.
- If both relays are either NO or NC, that would mean no motor movement.

Both in1 and in2 pins, which are connected to the digital pins 7 and 8 of Arduino respectively, are responsible for changing the relays' state from NC to NO, and thus, the motor polarity. In the initial condition, with both pins in 'HIGH' state, both relays remain normally closed. If a pin changes to 'LOW', current will no longer flow through the relay coil and its state will change to NO. This way, motor polarity can be controlled by two Arduino digital pins.

4.1.2 Chain actuator installation

The motor is clamped at the back of the trolley. A hole is drilled so the chain can go through. The end of the chain is clamped to the back wall of the trolley. That way, the functioning of the mechanism would be the following: if the chain is released, the drawer will open until the maximum elongation (380mm), and when the chain contracts, it would pull the drawer back insider of the trolley to its initial position.

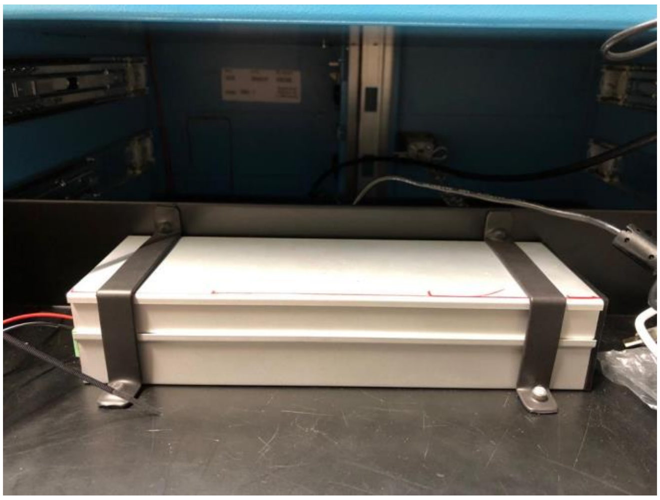


Fig. 4.4: Position of the motor inside of the drawer

4.2 Interaction with the user

In order to interact with the user, the complete system consists of:

1. FSR sensor for an easy closing after granted opening of the drawer.
2. Two ultrasonic sensors, to generate a security field around the trolley.
3. RFID reader and tags.

In this chapter, the installation of these three components will be explained, including details of the specific used devices, as well as their interaction with the Arduino UNO board.

4.2.1 FSR

The purpose of installing an FSR sensor on the drawer is to procure a fast and comfortable closing of the drawer, after its opening has been authorized. This means, once certain person has opened the drawer with their corresponding valid card, a simple light touch on the front of the drawer would be enough for it to automatically close.

To do this, a long thin strip of force sensitive resistor stuck on the front surface is required.

Fig. 4.5: Position of the FSR sensor on the drawer

Besides, the sensor will be connected to Arduino (placed inside of the drawer) by welding two cables to the positive and negative pins of the sensor respectively using a soldering iron.

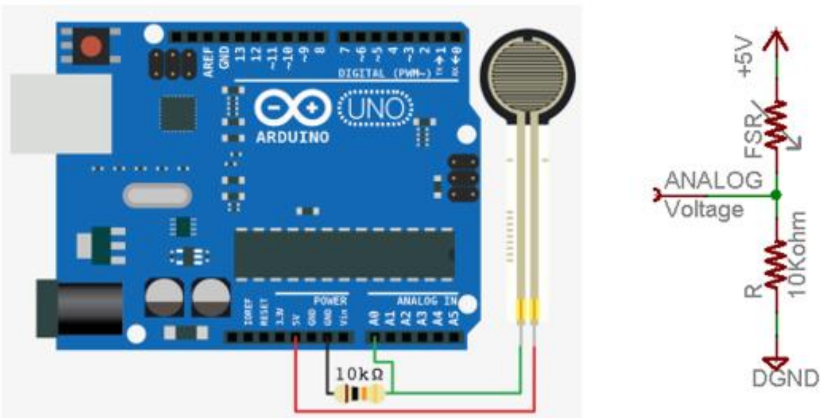


Fig. 4.6: Force Sensitive Resistor connection with Arduino (2)

The figure above shows the connection between the Arduino board and the sensor. However, in our case, the FSR would have a long and thin shape, contrary to the circular form the figure shows. The positive pole of the sensor receives the 5V power from the board. Negative pole connects, on the one hand with the analog pin A0, and on the other hand to the ground of the system through a 10k resistor.

$$V_{A0}[V] = 5\ [V] \cdot \frac{R}{FSR + R}$$

$$I[mA] = \frac{5\ [V]}{FSR + R\ [k\Omega]}$$

By setting up the above explained connection, A0 analog read voltage will vary as the FSR resistance value changes according to the equation above. The less force is applied on the sensor, the higher the FSR resistance, and consequently, the lower the voltage in pin A0.

Tab. 4.1 shows the current through the circuit and the measured voltage in A0 for different values of force (mass) applied on the sensor.

Tab. 4.1: Equivalence between force applied on the sensor and voltage measured in the analog pin A0.

Force [g]	Force [N]	FSR resistance [kΩ]	Current [mA]	A0 voltage [V]
0	0	Infinite	0	0
20	0,196	30	0,125	1,25
100	0,98	6	0,312	3,125
10^3	9,81	1,2	0,45	4,46
10^4	98,1	0,25	0,49	4,88

4.2.2 Ultrasonic sensors

Every mechanical based moving unit used in industry requires a security system to prevent unwanted accidents, that may cause damage either on the trolley or on any person interacting with it. The aim of the ultrasonic sensor unit in this project is to generate a safety zone surrounding the drawer during opening or closing operation. The movement of the drawer would be stopped if either an object or a person is detected within the safety zone.

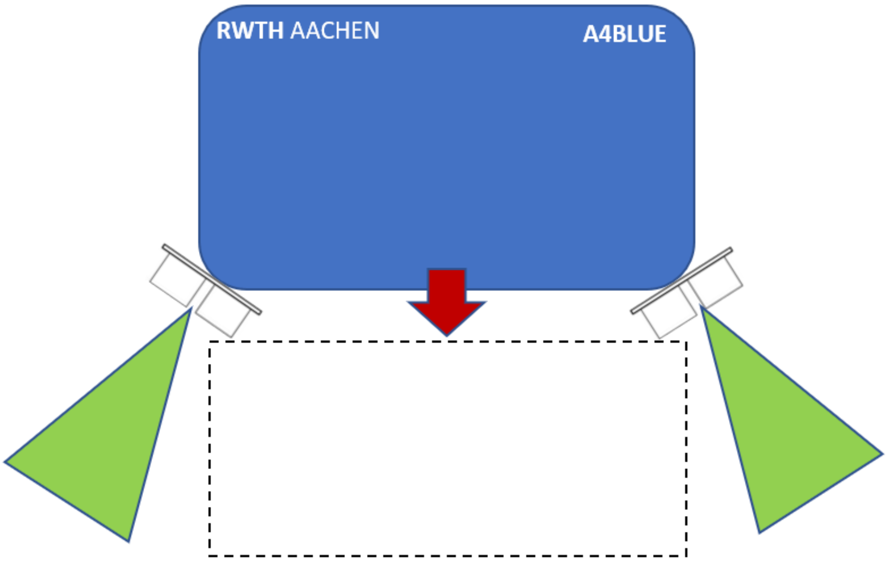


Fig. 4.7: Sketch of the security system.

The figure shows a diagram of the placement of both ultrasonic sensors in the trolley. Each of them generates a detection zone on each side of the trolley. The sensors, as explained in section 3, have a detection angle of approximately 15º, and can reach up to 30º in some cases.

The objective of this arrangement of sensors is to avoid the collision of the drawer with any object/person that may cross in front of the trolley during the opening or closing operation. The user will only be able to open the drawer if it is positioned in front of the trolley, so that he or she is outside the detection zone.

If either sensor receives a presence signal in its zone, the system will stop immediately and resume when the user reinserts a valid card.

Fig. 4.8: Ultrasonic sensors on the trolley

Figure 4. 8 shows the actual arrangement of the sensors on the trolley.

Concerning the connection between Arduino board and each of the sensors, each module has 4 pins: ground, V_{CC}, trig and echo. The ground and the V_{CC} pins of the module needs to be connected to the ground and the 5V pins on the Arduino Board respectively and the trig and echo pins to any Digital I/O pin on the Arduino Board. In order to generate the ultrasound, the trig is set on a high state for 10 µs. That will send out the 8-cycle sonic burst which will travel at the speed sound and, after hitting an object, it will come back and be received in the echo pin. The echo pin will output the time in microseconds the sound wave traveled. (13)

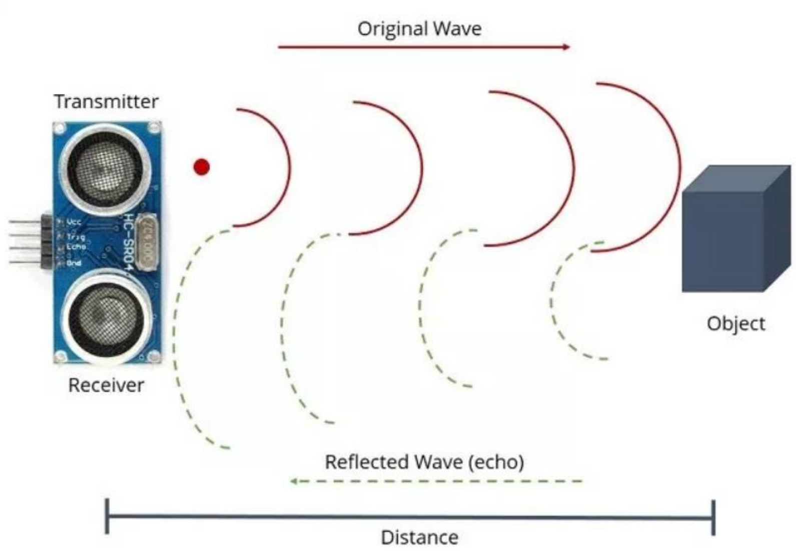


Fig. 4.9: Ultrasonic sensor HC-SR04 working principle sketch (15)

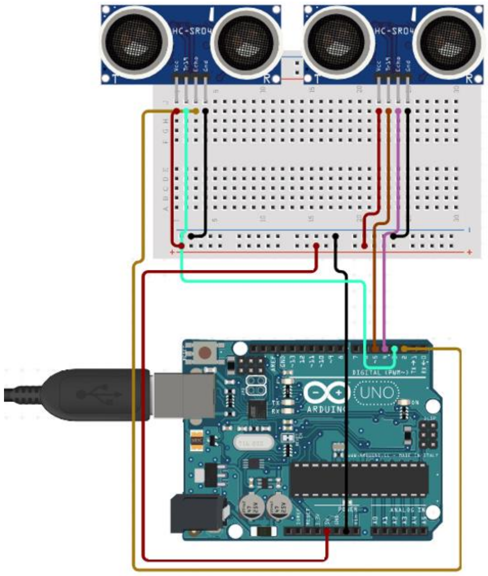


Fig. 4.10: Sketch of the connection of the ultrasonic sensors

4.2.3 RFID reader

The RFID system forms the brain of the trolley access control system. It consists of a reader, which will be placed outside the drawer above the trolley with the help of a small auxiliary bread board, and a set of tags, including a master card and a wiping tag, both previously defined.

The connection between RFID reader, RFID-RC522, and Arduino is standardized. The reader consists of several pins: SDA (SS), SCK, MOSI, MISO, IRQ, GND, RST and 3.3V. The following table and figure summarize the set up:

Tab. 4.2: Connection between Arduino UNO and RC522

RC522 module	Arduino UNO
SDA (SS)	10
SCK	13
MOSI	11
MISO	12
IRQ	Not connected
GND	GND
RST	9
3.3V	3.3V

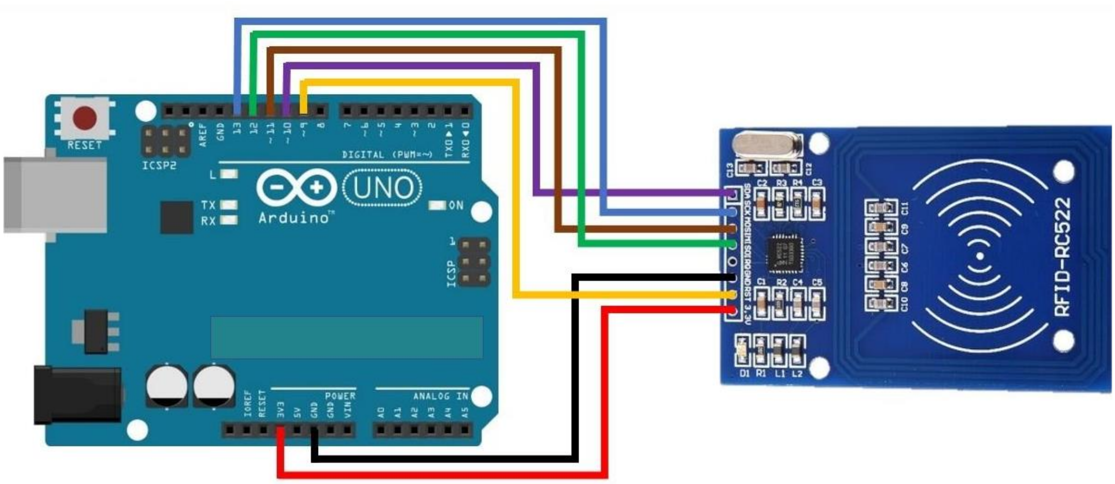


Fig. 4.11: RFID-RC522 Arduino connection (6)

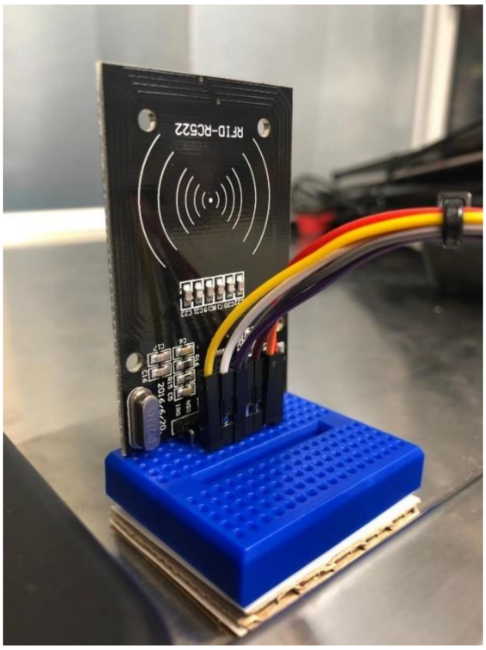


Fig. 4.12: RFID reader on top of the trolley

In order to make it accessible for the user, as Figure 4.12 shows, the RFID reader is placed on top of the trolley thanks to a small auxiliary bread board and connected to the Arduino board inside of the drawer.

4.3 Mounting diagram

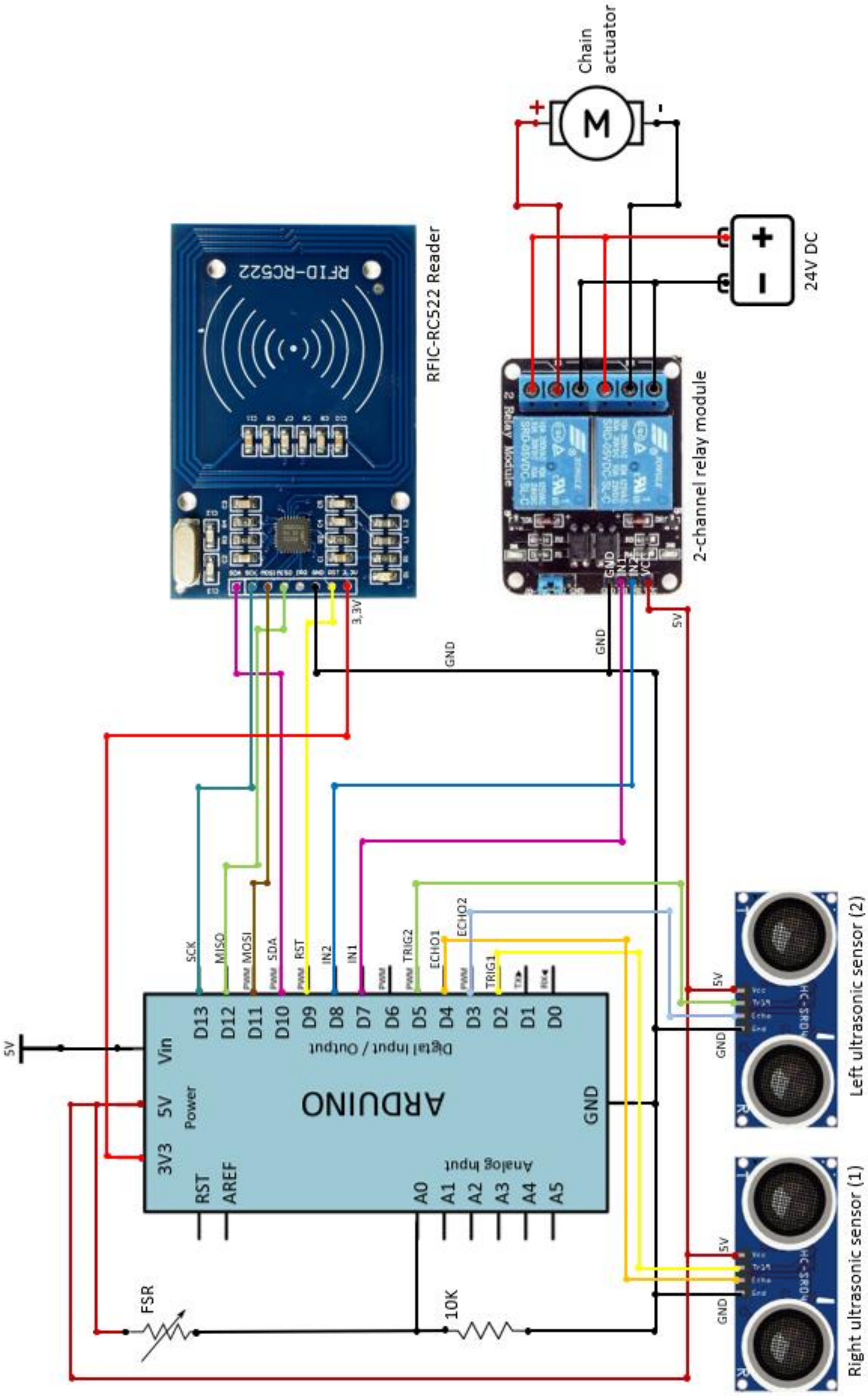


Fig. 4.13: Complete mounting diagram

5 Software programming

5.1 RFID system coding. EEPROM

For controlling the access to the drawer, an access control system by RFID is set up. This chapter provides an explanation of the logic flow of the RFID access control system and its actual implementation in Industry 4.0.

The coding is based on using the EEPROM memory, where the different cards/tags data are stored. The system principal is based on the establishment of a master card, which can on the one hand add new cards for new users or eliminate cards from the system on the other hand. Finally, a previously defined wiping card can erase the data stored in the EEPROM memory to restart the system.

The control of access to the trolley by RFID has the objective of facilitating, speeding up and controlling the access to the tools found in it. The section chief will be in possession of the master card and will be able to determine which employees have access to the tools. In addition, he will be able to authorize access, in a simple and practical way, to new employees or prohibit it to others, either because they leave the company or for any other reason. Therefore, the system provides great adaptability to changes and can be restored at any moment.

A future implementation would be to install the actuator-RFID system to the six trolley drawers, in order to determine to which drawer which employee has access depending on the station of the assembly line he or she oversees. In this way, we ensure that the right tools are used at every point and that they remain tidy, streamlining the production process. In addition, it will serve as a security system, preventing an operator from using, either by mistake or intentionally, a tool that does not correspond to his task.

In order to implement the system, a logic flow is defined and introduced in Arduino. In the following paragraphs and using Figure 5.1, the overview of the system functioning will be explained.

To begin with, the system checks whether a master card is defined. If not already entered, the system will send a message via the Monitor Serie, requiring the insertion of a master card.

Once the master card has been defined, the system is in stand-by. At this point, the logic defined in the flowchart in Figure 5.1 starts.

- The first option is for the user to insert the master card. In this case, the system enters what we have called “Program Mode". In this mode the user can add or remove cards from the system. If a previously valid card is scanned, it will be removed from the system and will no longer be valid (removed from the EEPROM memory). If, on the other hand, the scanned card was not among the cards giving access to the drawer, it will be included in the system (added to the EEPROM memory). Finally, to exit this mode, the user must rescan the master card.

- A second possibility is to scan a card other than the master card (and other than the wiping card, explained below). In this case, the system will check whether the card is currently in the EEPROM memory or not, allowing or denying access to the system respectively.
- Finally, the user may scan the wiping card. This card, as its name suggests, has the function of restoring the EEPROM memory by removing everything previously stored on it. After the system has been restored, the user will be prompted again to insert a new master card.

No matter which of the three different options has occurred, the system will be in stand-by again waiting for the user to scan a new card.

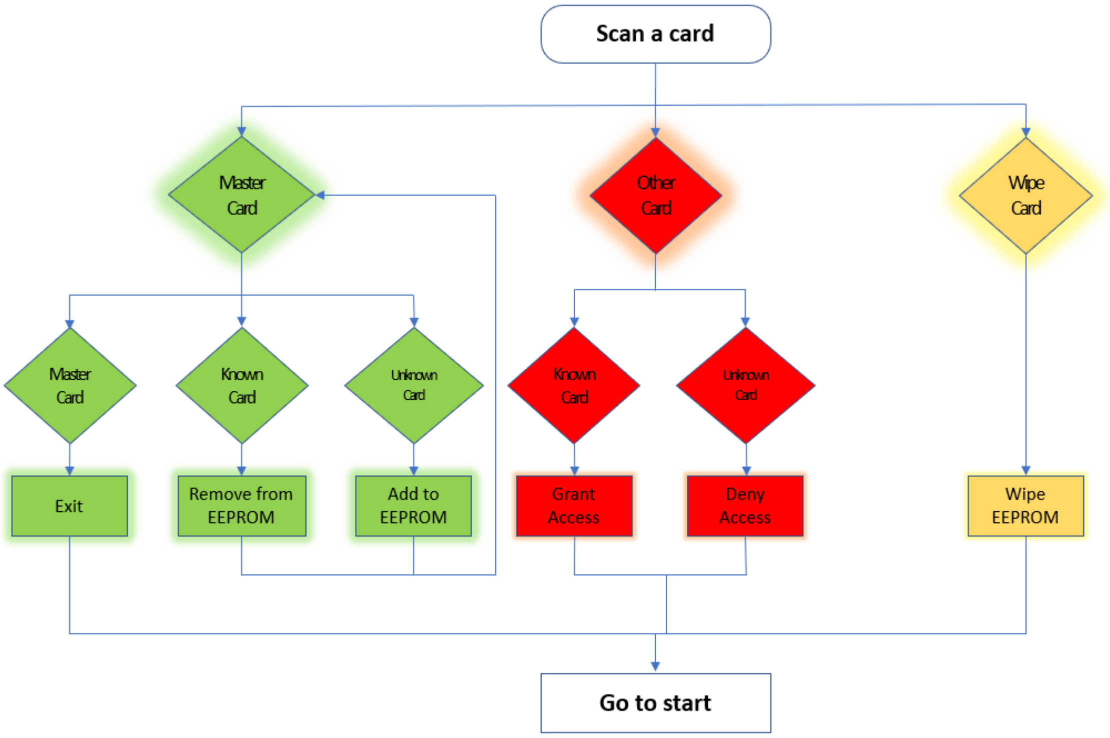


Fig. 5.1: Flowchart RFID access system

5.2 DC Motor controlling coding

The logic explained in chapter 4.1. is translated into programming language by means of the function “void granted()”. This function will be called in the void loop() every time the user scans a valid card in the RFID reader.

Figure 5.2 shows a flowchart explaining the logic used to program the opening and closing of the drawer. The logic flow is based on the use of an additional auxiliary variable: 'MotDirection' (MD).

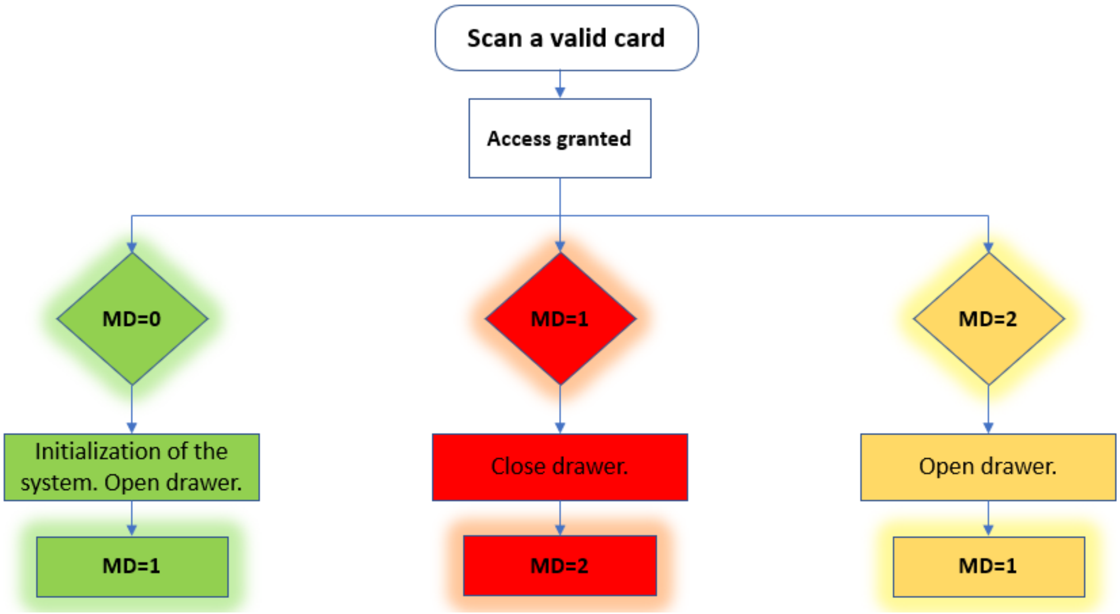


Fig. 5.2: Flowchart of the logic for the opening/closing of the drawer

This extra variable 'motDirection' is defined to register the actual state of the motor. It can take three distinct values:

- The variable is initialized with a 0. If a valid card is scanned, system will start and open the drawer. A '1' is stored, to show that the drawer state changes to open.
- If the stored value is '1', it means the drawer has been previously opened. Thus, whenever a valid card is scanned, the drawer proceeds to close and its value changes to '2'.
- Finally, the system enters the yellow branch whenever 'motDirection' stores a '2'. In that case, drawer will be opened, and MD updated to '1'.

In conclusion, '0', '1' and '2' mean respectively 'system not initialized yet', 'drawer open' and 'drawer closed'.

The chain actuator has an automatic system that stops the motor once the chain has reached its maximum elongation. For this reason, although the digital pins excite the relay so that it feeds the motor to open or close, there will be no movement. This fact, in addition to simplifying the code, serves as a security measure.

5.3 FSR Coding

The programming of the “easy closing after granted opening” is based on the changes of voltage read on pin A0 as a result of the variation of FSR resistance.

The microcontroller of the board has a circuit inside called an analog-to-digital converter or ADC that reads this changing voltage and converts it to a number between 0 and 1023. When there is no force applied on the sensor, there are 0 volts going to the pin, and the input value is 0. When the user applies the maximum force on the sensor, there are 5 volts going to the pin and the input value is 1023. In between, analogRead() returns a number between 0 and 1023 that is proportional to the amount of voltage being applied to the pin.

Using this principle, a border is set between what is considered a light touch to open the drawer or, on the other hand, an accidental touch. This threshold is set to 800. If the user presses softer than an analogRead() of 800, a message will be sent by means of the Monitor Serial, asking for a harder touch. If the touch has been hard enough, motor will proceed to close.

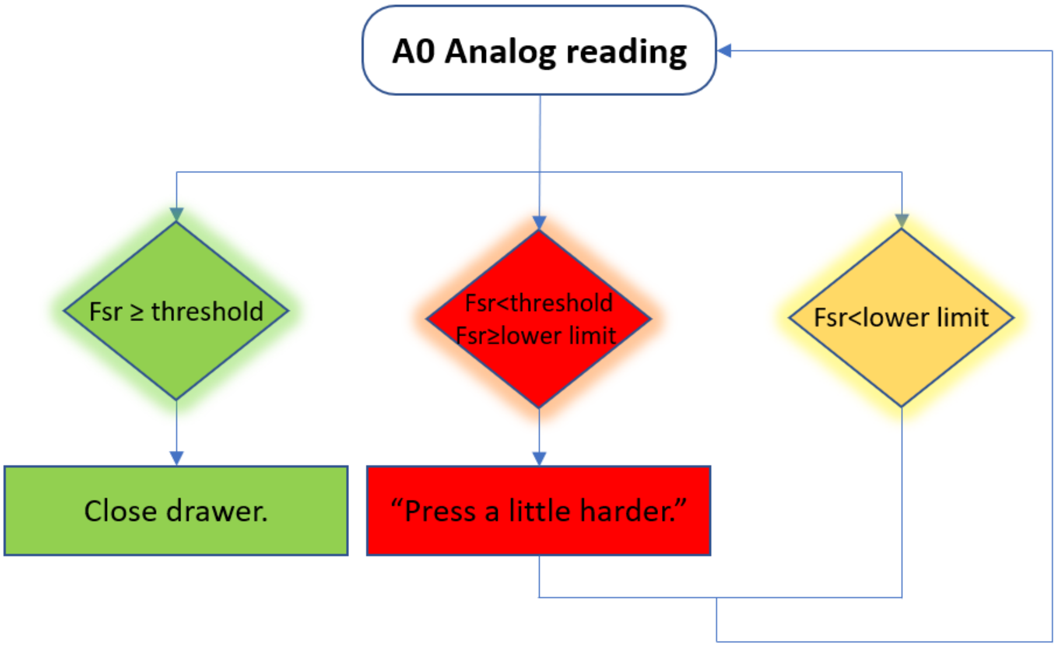


Fig. 5.3: Flow chart of the logic of the FSR closing system

5.4 Ultrasonic sensor coding

The principle of the functioning of the ultrasonic sensor is based on calculating the distance to the most proximate object, either on the right or the left sensor, and comparing it with a predetermined threshold distance. If the measured distance is smaller than the previously set distance, the system will immediately stop its motion.

During opening and closing operations, trigger and echo pins are permanently functioning, and the system receives the time traveled by the ultrasonic wave as an input. The velocity of the ultrasonic burst is 340 m/s. in air. Based on the time recorded by the sensor, and considering that the wave travels first from transmitter to object, and then back from object to receiver, the distance can be calculated using the following expression:

$$\mathrm{d} = \mathrm{c} \cdot \frac{\mathrm{t}}{2} \tag{5.1}$$

Where:

d: distance from sensor to object [m]

c: speed of the ultrasonic wave through air (340 [m/s])

t: time recorded by the sensor timer [s]

Once the distances corresponding to the times recorded by each of the two sensors have been calculated, both are compared, and the most critical (the smallest) is stored. The next step is to compare this value with the previously imposed limit. If the calculated distance is greater than the threshold, the entire process will start from the beginning again. If it is below the limit, the system will automatically stop to avoid the crash, or in the event of a crash, to avoid further damage.

The process goes on while the drawer is opening or closing and will stop when these operations are finished. To achieve this, the time it takes for the drawer to open/close was measured beforehand. Using a timer, we keep the security system working as long as the established time is not reached and there has not been an emergency stop.

In case of an emergency stop, the system will resume when the user scans a valid card again. In this case, the drawer will close if the emergency stop occurred during opening operation, and on the other hand, it will open if the system was interrupted while closing.

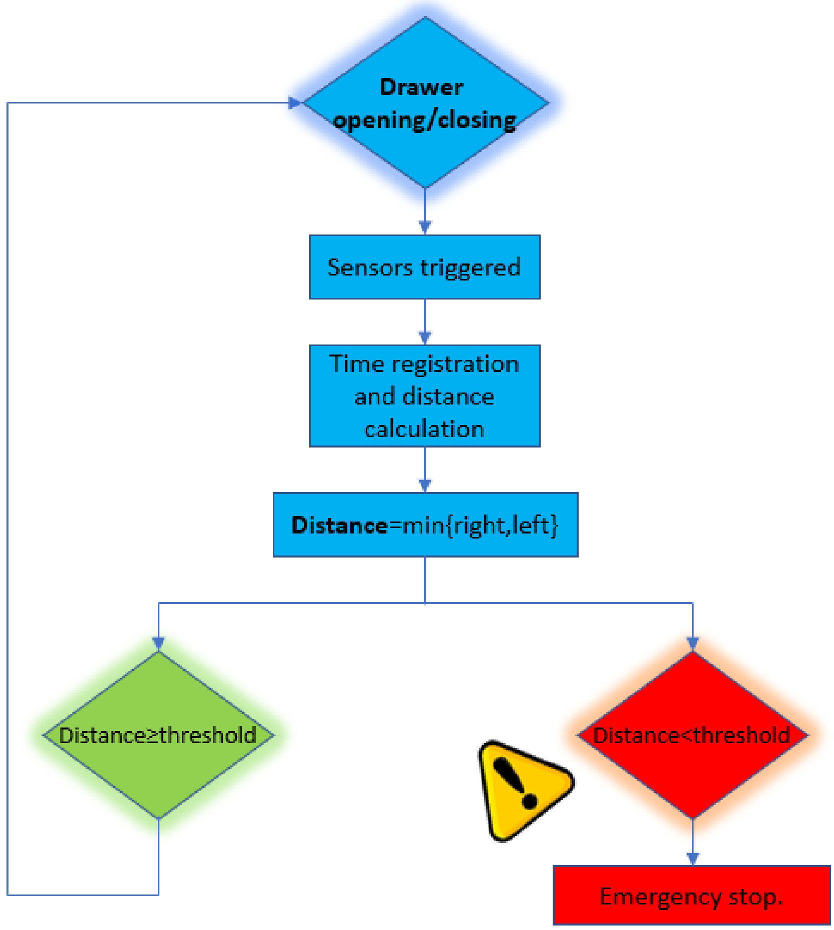


Fig. 5.4: Flowchart of the ultrasonic sensor coding

6 Conclusions

The set of four systems developed and implemented in the trolley provide a safe and secure system of access control and automation of the drawer. The functioning of the system as a whole has been verified in real application, simulating with a master card the figure of the section director, and with two more cards the positions of an operator with authorized access to the drawer and of another who is not included in the system. The system of inclusion and elimination of cards within the RFID system has been successfully tested. Opening operation lasts no longer than 7 seconds, performing a suitable and safe speed for this application. At the same time, emergency situations have been simulated in which an object interrupts the opening movement of the drawer and the drawer manages to react in time and break the system immediately. Finally, the system of easy closing by means of the pressure sensor proves effectiveness at the time of closing the drawer. We were able to verify the usefulness and improvement of the system in terms of accessibility and ease of use.

This project finds its greatest utility in large-scale implementation. The thesis lays the foundations for an idea which, in its series production, will include improvements and innovations. Its implementation in the industry would include an automation of all six trolley drawers. For this, six actuators are required, one for each drawer. The manufacturer should count on a space at the end of each drawer in order to be able to include the actuator. On the other hand, an Arduino board, or possibly a more powerful microcontroller, would suffice to control the system. Through a complex system of relays this would be possible, maintaining the rest of the structure. The manufacturer must consider the cable outlet to connect the RFID reader to the system, which will be placed on top of the trolley. The access control system will register the cards included in the system of each of the drawers, by dividing the storage of the EEPROM memory into six different modules. Each operator will have access only to his/her assigned drawer. This way, the misplacement of the tools will be avoided and a correct distribution of the tools in the drawers is ensured. Finally, hollow trolley walls are required at both sides of the drawer, so that the ultrasonic sensors are clamped inside. FSR sensors on the front part of each drawer will provide an easy closing.

7 Bibliography

1. (n.d.). Retrieved from https://www.arduino.cc/en/Main/Software. Last accessed on 15/06/2019.
2. *Adafruit*. (n.d.). Retrieved from https://learn.adafruit.com/force-sensitive-resistor-fsr/using-an-fsr. Last accessed on 04/06/2019.
3. Advanced Mobile Group. (2017, March 27). *RFID's Role in Industry 4.0*. Retrieved from https://www.advancedmobilegroup.com/blog/rfids-role-in-industry-4.0
4. American Barcode and RFID. (n.d.). Retrieved from https://www.abr.com/rfid-access-control/. Last accessed on 05/06/2019.
5. Arduino. (2019). *What is Arduino?* Retrieved from https://www.arduino.cc/en/Guide/Introduction. Last accessed on 14/06/2019.
6. circuits4you.com. (n.d.). *Interfacing of RFID RC522 with Arduino UNO*. Retrieved from https://circuits4you.com/2018/10/03/interfacing-of-rfid-rc522-with-arduino-uno/. Last accessed on 18/05/2019.
7. Corday, R. (2014, April 24). Retrieved from https://robohub.org/the-evolution-of-assembly-lines-a-brief-history/
8. EEE Projects. (2018, April 22). *ARDUINO UNO BOARD – FULL DESCRIPTION*. Retrieved from https://eeeproject.com/arduino-uno-board/
9. Electronics notes. (n.d.). *What is EEPROM Memory Technology*. Retrieved from https://www.electronicsnotes.com/articles/electronic_components/semiconductor-ic-memory/eeprom-e2prom-technology.php. Last accessed on 04/06/2019.
10. European Commission. (2019). *A4BLUE*. Retrieved from http://a4blue.eu/
11. Hopeland. (2019). *Industry 4.0. Smart Manufacturing.* Retrieved from https://www.hopelandrfid.com/industry-4-0-smart-manufacturing_n40
12. How to mechatronics. (n.d.). *How RFID Works and How To Make an Arduino based RFID Door Lock*. Retrieved from https://howtomechatronics.com/tutorials/arduino/rfid-works-make-arduinobased-rfid-door-lock/. Last accessed on 04/06/2019.
13. How to mechatronics. (n.d.). *Ultrasonic Sensor HC-SR04 and Arduino Tutorial*. Retrieved from https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/. Last accessed on 04/06/2019.
14. Institute for the Production Engineering of E-Mobility Components. (2017, September 29). *Adaptive Automation in Assembly For BLUE collar workers*

satisfaction in Evolvable context. Retrieved from http://www.pem.rwth-aachen.de/cms/PEM/Forschung/Projekte/~opwk/A4BLUE/?lidx=1

15. Instructables Circuits. (n.d.). Retrieved from https://www.instructables.com/id/ARDUINO-With-Ultrasonic-Sensor-distance-Measuremen/. Last accessed on 29/05/2019.
16. Interlink Electronics. (n.d.). *FSR® Integration Guide.* Retrieved from https://produktinfo.conrad.com/datenblaetter/500000-524999/503372-an-01-en-DRUCKSENSOR_FSR_408.pdf. Last accessed on 04/06/2019.
17. Jungbae Roh, J., Kunnathur, A., & Tarafdar, M. (2009, July 18). *Classification of RFID adoption: An expected benefits approach.* Retrieved from https://www.sciencedirect.com/science/article/pii/S0378720609000767
18. Michael Rüßmann, M. L. (2015, April 09). *Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries.* Retrieved from http://www.inovasyon.org/pdf/bcg.perspectives_Industry.4.0_2015.pdf
19. PEPPERL+FUCHS AG. (n.d.). Retrieved from https://www.pepperl-fuchs.com/global/en/23763.htm. Last accessed on 04/08/2019.
20. R. Fletcher, S., Johnson, T., Adlon, T., Casla, P., Larreina, J., Parigot, L., . . . Otero, M. (2019, March 23). *Adaptive automation assembly: Identifying system requirements for technical efficiency and worker satisfaction.* Retrieved from https://www.sciencedirect.com/science/article/pii/S036083521930172X
21. Random Nerd Tutorials. (n.d.). *Complete Guide for Ultrasonic Sensor HC-SR04 with Arduino*. Retrieved from https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/. Last accessed on 04/06/2019.
22. Roberti, M. (2005, January 16). *The History of RFID Technology.* Retrieved from https://www.rfidjournal.com/articles/view?1338
23. Sancho, J. S. (n.d.). *Industria 4.0. Aplicación de Tecnologías RFID para la mejora de procesos logísticos.* Retrieved from https://repositorio.unican.es/xmlui/bitstream/handle/10902/13400/SOTRESSANCHOJAVIER.pdf?sequence=1&isAllowed=y. Last accessed on 03/08/2019.
24. Sang Chul Ahn, H.-G. K.-M. (2004, January). *Smart Location Tracking System using FSR (Force Sensing Resistor).* Retrieved from https://www.researchgate.net/publication/253026507_Smart_Location_Tracking_System_using_FSR_Force_Sensing_Resistor
25. Tech Target. (n.d.). *RFID*. Retrieved from https://internetofthingsagenda.techtarget.com/definition/RFID-radio-frequency-identification. Last accessed on 04/06/2019.

26. *Trive blog*. (n.d.). Retrieved from https://blog.gotrive.com/el-primer-coche-electrico-1832/trive_blog_cochelectrico_ford-model-t-assembly-line/#.XPaftYgzZPZ. Last accessed on 04/06/2019.

27. Woodford, C. (2018, April 20). *Explain that stuff!* Retrieved from https://www.explainthatstuff.com/howrelayswork.html

8 Appendix I

[Linea **Micro 02**] · MINGARDI
THE REFERENCE FOR WINDOW AUTOMATION

Kettenantriebe für:
Klappfenster, Kippfenster
Chain actuator for:
top-hinged windows,
bottom-hinged windows

150/300N

- Wählbarer Hub
  Selectable stroke
- Gehäuse Silbereloxiert
  Anodised aluminium body
- Wartungsfrei
  Maintenance free

- Schutzart: IP 20 / *Protection degree: IP 20*
- Endabschaltung Auf/Zu: Reed-/Microschalter / *Limit switch closing-opening: Microswitch-Reed*
- Farbe: Silbereloxiert (A) - Schwarz (N) - RAL 9010 (B) - RAL 9016 (C)
  Colour: Anodised silver (A) - black (N) - RAL 9010 (B) - RAL 9016 (C)

TECHNISCHE DATEN Micro 02 / *TECHINAL FEATURES* Micro 02

BEST-NR *CODE*	BEZEICHNUNG *DESCRIPTION*	STEUERSPANNUNG *VOLTAGE*	SCHUBKRAFT ZUGKRAFT *THRUST TENSILE*	HUB *STROKE*	LEISTUNG MAX *CONSUMPTION MAX*	GESCHWINDIGKEIT MIN. *SPEED MIN.*
2700462	MICRO 02 250/380 mm - A	230 V~ / 50 Hz	150 N / 300 N	250/380 mm	130 W	ca. 38 mm/s
2700469	MICRO 02 250/380 mm - N	230 V~ / 50 Hz	150 N / 300 N	250/380 mm	130 W	ca. 38 mm/s
2700463	MICRO 02 250/380 mm - B	230 V~ / 50 Hz	150 N / 300 N	250/380 mm	130 W	ca. 38 mm/s
2700331	MICRO 02 250/380 mm - C	230 V~ / 50 Hz	150 N / 300 N	250/380 mm	130 W	ca. 38 mm/s
2700471	MICRO 02 250/380 mm - A	24 V =	150 N / 300 N	250/380 mm	55 W	ca. 38 mm/s
2700477	MICRO 02 250/380 mm - N	24 V =	150 N / 300 N	250/380 mm	55 W	ca. 38 mm/s
2700472	MICRO 02 250/380 mm - B	24 V =	150 N / 300 N	250/380 mm	55 W	ca. 38 mm/s
2700332	MICRO 02 250/380 mm - C	24 V =	150 N / 300 N	250/380 mm	55 W	ca. 38 mm/s

DIE IN DER TABELLE ANGEFÜHRTE WERTE WURDEN BEI EINER BETRIEBSTEMPERATUR ZWISCHEN -10°/+40° UND EINER RELATIVEN LUFTFEUCHTIGKEIT VON 60% ERMITTELT.
THE VALUES INDICATED ON THE TABLE WERE RECORDED IN A CLIMATIC TEST CHAMBER AT OPERATING TEMPERATURES BETWEEN -10°C/+40°C WITH MAX 60% RELATIVE HUMIDITY

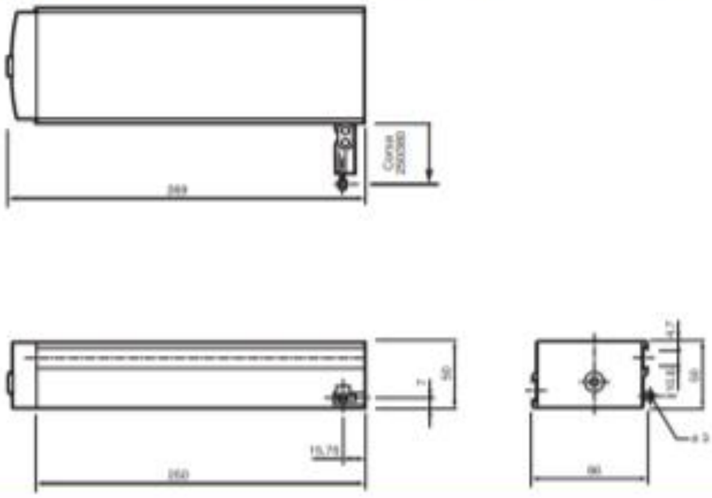
PRODUKTBESCHREIBUNG:

Elektrischer Linearantrieb mit Gelenkkette.
Betriebsspannung: 230V~ 50 Hz oder 24V DC. Entspricht den Richtlinien 73/23 EGW Niederspannung - 89/336 EWG (EMV) gemäß Änderungen 93/68 EWG (für den Betrieb bei 24V DC entsprechend der Richtlinien 89/336 EWG (EMV) gemäß Änderungen 93/68 EWG). Ausgestattet mit Reedschaltern bei der Öffnung und Mikroendschaltern bei der Schließung. Hublänge wählbar mittels Jumper. Möglichkeit der Parallelschaltung. Erhältlich in den Farben: Silbereloxiert, Schwarzeloxiert oder Weiß RAL 9010 RAL 9016. Komplett mit Installationszubehör für Klappflügelmontage.

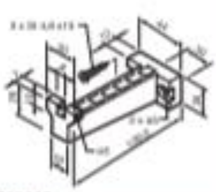
DESCRIPTION OF SPECIFICATIONS

MINGARDI linear electric actuator with sproket chain inside the actuator body it operates either on 230V~ 50 Hz. or 24V DC. In conformity with Directives 73/23 LVD - 89/336 EMC as modified by 93/68 CE (compliant with Directives 89/336 EMC as modified by 93/68 CE for 24V DC operation). Outfitted with end-of-stroke microswitch in closed position / reed contact in open position system. Selectable stroke by means of external selector. Can be connected in parallel with signal outputs. Available colours: anodised silver / anodised black / white 9010 Ral / 9016 Ral. Complete with universal supports.

Micro 02 Masszeichnungen / *Micro 02 Scale drawing*

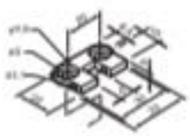


Zubehör und Ersatzteile / *Accessories - Spare parts*



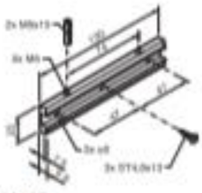
2700695

2700705



2700114

2700111

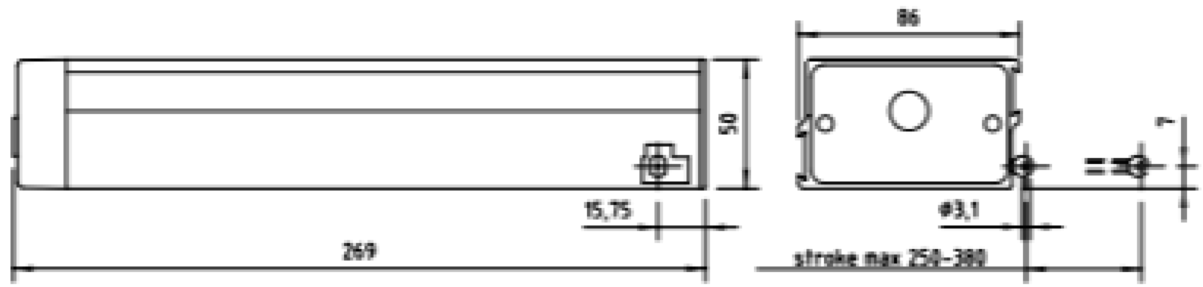


2700105

MONTAGEZUBEHÖR / *FIXING ACCESSORIES*	
BEST-NR *CODE*	**BEZEICHNUNG** *DESCRIPTION*
2700695	Fensterflügelkonsole für Kippfenster *Bracket for bottom-hinged windows*
2700705	Verlängerte Augenschraube für PVC-und Holz-Fenster *Extended front head for windows in pvc/wood*

ERSATZTEILE / *SPARE PARTS*	
BEST-NR *CODE*	**BEZEICHNUNG** *DESCRIPTION*
2700114	Fensterflügelkonsole für Klappfenster *Front bracket for window frames*
2700111	Splint *Split pin*
2700105	Befestigungsschiene für Antrieb *Motor support bracket*

STEUERZUBEHÖR: Für weitere Informationen werden Sie sich bitte an unseren technischen Vertriebsservice oder besuchen Sie unsere Internetseiten www.mingardimotor.com
CONTROL ACCESSORIES: please contact our technical support service for further information or consulting our web-site www.mingardimotor.com



Kettenantrieb Micro 02
230V, 150 N / 300 N,
Hub 250/380mm

10.003462.3

9 Appendix II

5V/220V 2 Channel Optocouplers Relay Shield for Arduino

- Technical data:
- Supply voltage (VCC): 5V
- Current consumption control input approx. 4 mA
- Current consumption coil approx. 70 mA
- Coil resistance approx. 70 Ohm
- High current relay: AC250V 10A; DC30V 10A
- Standard interface for Arduino, AVR, PIC, ARM, DSP,8051, ARM, TTL logic
- LEDs show the status of the relays

10 Appendix III

```
#include <EEPROM.h>     // We are going to read and write Tag's UIDs from/to
EEPROM
#include <MFRC522.h>
#include <Servo.h>
#include <SPI.h>
#include <Wire.h>
// Create instances
int trigPin1 = 2;     // Trigger1
int echoPin1 = 4;     // Echo1
int trigPin2 = 5;     // Trigger2
int echoPin2 = 3;     // Echo2
int RightSensor, LeftSensor;
int in1 = 7;
int in2 = 8;
long timer, timer1;
int emergency;
int cm;
int cm1;
int cm2;
int distance;
int duration1, duration2;
int motDirection;
int fsrPin = 0;     // the FSR and 10K pulldown are connected to a0
int fsrReading;     // the analog reading from the FSR resistor divider
const int RST_PIN = 9;         // Pin 9 reset of RC522
const int SS_PIN = 10;         // Pin 10 SS (SDA) of RC522
MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create instance of MFRC522
//Servo myServo;  // create servo object to control a servo
   // Button pin for WipeMode
boolean match = false;          // initialize card match to false
boolean programMode = false;  // initialize programming mode to false
boolean replaceMaster = false;
uint8_t successRead;    // Variable integer to keep if we have Successful Read
from Reader
byte storedCard[4];   // Stores an ID read from EEPROM
byte readCard[4];   // Stores scanned ID read from RFID Module
byte masterCard[4];   // Stores master card's ID read from EEPROM
byte wipeCard[4] = { 0xC1, 0x44, 0xD0, 0x83 };

void ShowReaderDetails();
uint8_t getID();
boolean isMaster( byte test[] );
boolean findID( byte find[] );
void deleteID( byte a[] );
void writeID( byte a[] );
void granted ();
int security ();
void SonarSensor (int trigger, int echo);


//Function to compare two vectors
bool isEqualArray(byte* arrayA, byte* arrayB, int length)
{
  for (int index = 0; index < length; index++)
  {
    if (arrayA[index] != arrayB[index]) return false;
  }
  return true;
}

////////////////////////////////////////// Setup
////////////////////////////////////
void setup() {
```

```
  //Protocol Configuration
  Serial.begin(9600);
  SPI.begin();           // MFRC522 Hardware uses SPI protocol
  mfrc522.PCD_Init();    // Initialize MFRC522 Hardware

  ShowReaderDetails();  // Show details of PCD - MFRC522 Card Reader details

  //Pin modes
  pinMode(in1,OUTPUT);
  pinMode(in2,OUTPUT);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);

  //Set initial motor direction (no motion, both in1 and in2 high)
  digitalWrite(in1,HIGH);
  digitalWrite(in2,HIGH);

  // Check if master card defined, if not let user choose a master card
  // This also useful to just redefine the Master Card
  // You can keep other EEPROM records just write other than 143 to EEPROM ad-
dress 1
  // EEPROM address 1 should hold magical number which is '143'
  if (EEPROM.read(1) != 143) {

    Serial.println("\nNo Master Card defined. ");

    Serial.println("\nScan A Tag to define as Master");
    do {
      successRead = getID();            // sets successRead to 1 when we get
read from reader otherwise 0
    }
    while (!successRead);                  // Program will not go further
while you not get a successful read
    for ( uint8_t j = 0; j < 4; j++ ) {        // Loop 4 times
      EEPROM.write( 2 + j, readCard[j] );  // Write scanned Tag's UID to
EEPROM, start from address 3
    }
    EEPROM.write(1, 143);                  // Write to EEPROM we defined Mas-
ter Card.

    Serial.println("Master Defined");
    //delay(2000);
  }
  for ( uint8_t i = 0; i < 4; i++ ) {          // Read Master Card's UID from
EEPROM
    masterCard[i] = EEPROM.read(2 + i);    // Write it to masterCard
  }

  Serial.println("Mastercard UID: ");

      for (byte i = 0; i < 4; i++) {
              Serial.print(masterCard[i] < 0x10 ? " 0" : " ");
              Serial.print(masterCard[i], HEX);
}
}
////////////////////////////////////////// Main Loop
///////////////////////////////////
void loop () {

  do {
    successRead = getID();// sets successRead to 1 when we get read from
reader otherwise 0
  }
```

```
  while (!successRead);   //the program will not go further while you are not
getting a successful read
 {

  {
  if (isEqualArray(readCard, wipeCard, 4)) {    // If wiping card is tagged,
wipe EEPROM
     Serial.println("\nWipecard UID: ");

      for (byte i = 0; i < 4; i++) {
              Serial.print(wipeCard[i] < 0x10 ? " 0" : " ");
              Serial.print(wipeCard[i], HEX);
}
      Serial.println("\nWiping EEPROM...");
      for (uint16_t x = 0; x < EEPROM.length(); x = x + 1) {    //Loop end of
EEPROM address
        if (EEPROM.read(x) == 0) {              //If EEPROM address 0
          // do nothing, already clear, go to the next address in order to
save time and reduce writes to EEPROM
        }
        else {
          EEPROM.write(x, 0);       // if not write 0 to clear, it takes 3.3mS
        }
      }

      Serial.println("\nWiping Done");
   }
   // Finish current card reading
      mfrc522.PICC_HaltA();
       }
}
//Check again if there is a master card defined.
   if (EEPROM.read(1) != 143) {
     {
      Serial.println("\nNo Master Card defined. ");
      Serial.println("\nScan A Tag to define as Master");

    do {
      successRead = getID();            // sets successRead to 1 when we get
read from reader otherwise 0
    }
    while (!successRead);                  // Program will not go further
while you not get a successful read
    for ( uint8_t j = 0; j < 4; j++ ) {         // Loop 4 times
      EEPROM.write( 2 + j, readCard[j] );  // Write scanned Tag's UID to
EEPROM, start from address 3
    }
    EEPROM.write(1, 143);                  // Write to EEPROM we defined Mas-
ter Card.

    Serial.println("Master Defined");
  }
  for ( uint8_t i = 0; i < 4; i++ ) {          // Read Master Card's UID from
EEPROM
    masterCard[i] = EEPROM.read(2 + i);    // Write it to masterCard
  }

  Serial.println("Mastercard UID: ");
      for (byte i = 0; i < 4; i++) {
              Serial.print(masterCard[i] < 0x10 ? " 0" : " ");
              Serial.print(masterCard[i], HEX);
}
}
  if (programMode) {
    if ( isMaster(readCard) ) { //When in program mode check First If master
card scanned again to exit program mode
```

```
      Serial.println("Exiting Program Mode");


      programMode = false;
      return;
    }
    else {
      if ( findID(readCard) ) { // If scanned card is known delete it

         Serial.println("Eliminating card from the system...");
        deleteID(readCard);

         Serial.println("Tag to ADD/REM or Master to EXIT");

      }
      else {                    // If scanned card is not known add it

        Serial.println("New Tag,adding...");
        writeID(readCard);

        Serial.println("Scan to ADD/REM");

        Serial.println("Master to Exit");
      }
    }
  }
  else {
    if ( isMaster(readCard)) {    // If scanned card's ID matches Master
Card's ID - enter program mode
      programMode = true;
       Serial.println("\nPROGRAM MODE");
      uint8_t count = EEPROM.read(0);   // Read the first Byte of EEPROM that
stores the number of ID's in EEPROM

      Serial.println("I have ");
      Serial.print(count);
      Serial.print(" registered.");

      Serial.println("\nScan a Tag to ADD/REMOVE or Mastercard to EXIT the
program mode");
    }
    else {
      if ( findID(readCard) ) { // If not, see if the card is in the EEPROM
        Serial.println("\nAccess Granted");
        granted();          // Grant access
        timer1=millis();
        do{
          emergency = security();
          timer=millis()-timer1;

          if(emergency < 10 && emergency != 0){
            digitalWrite(in1,HIGH);
            digitalWrite(in2,HIGH);
            Serial.println("\nEmergency stop.");

          }

        }while((emergency > 10 || emergency == 0) && timer < 7000 && ( !
mfrc522.PICC_IsNewCardPresent()));
//
           do {
        if(motDirection==1){
        fsrReading = analogRead(fsrPin);
        Serial.print("\nAnalog reading = ");
        Serial.print(fsrReading);     // the raw analog reading
        if(fsrReading >= 800){
          digitalWrite(in1,HIGH);
```

```
           digitalWrite(in2,LOW);
           Serial.print("\nEnough pressure made. Closing drawer");
           motDirection = 2;
           timer1=millis();
           do{
            emergency = security();
            timer=millis()-timer1;

             if(emergency < 10 && emergency != 0){
              digitalWrite(in1,HIGH);
              digitalWrite(in2,HIGH);
              Serial.println("\nEmergency stop.");

             }
            }while((emergency > 10 || emergency == 0) && timer < 7000 && ( !
mfrc522.PICC_IsNewCardPresent()));

         } else if (fsrReading <800 && fsrReading >=100){
           Serial.print("\nPuss a little harder");
          }
         }
     }while ((motDirection == 1) && ( ! mfrc522.PICC_IsNewCardPresent()));

       }
       else {      // If not, show that the Access is denied
         Serial.println("\nAccess Denied");
         //denied();

       }
 }
}


}
/////////////////////////////////////////  Sonar Sen-
sor    /////////////////////////////////////
//void SonarSensor (int trigger, int echo){
//  digitalWrite (trigger, LOW);
//  delayMicroseconds(5);
//  digitalWrite(trigger, HIGH);
//  delayMicroseconds(10);
//  digitalWrite(trigger, LOW);
//  duration = pulseIn(echo, HIGH);
//  distance = (duration/2) / 29.1;
//}
/////////////////////////////////////////  Secu-
rity    /////////////////////////////////////
int security () {
  // The sensor is triggered by a HIGH pulse of 10 or more microseconds.
      // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:

    digitalWrite (trigPin1, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin1, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin1, LOW);
    duration1 = pulseIn(echoPin1, HIGH);
    RightSensor = (duration1/2) / 29.1;

    Serial.print("\nRight Sensor: ");
    Serial.print(RightSensor);

    digitalWrite (trigPin2, LOW);
    delayMicroseconds(5);
    digitalWrite(trigPin2, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin2, LOW);
```

```
    duration2 = pulseIn(echoPin2, HIGH);
    LeftSensor = (duration2/2) / 29.1;

    Serial.print("\nLeft Sensor: ");
    Serial.print(LeftSensor);

      if(RightSensor != 0 && LeftSensor != 0){
      cm = min(RightSensor, LeftSensor);
      if (cm<0){
        cm=0;
      }
      }
      else if(RightSensor == 0 && LeftSensor != 0){
        cm = LeftSensor;
      }
      else if(LeftSensor == 0 && RightSensor != 0){
        cm = RightSensor;
      }else if (RightSensor == 0 && LeftSensor == 0){
        cm=0;
      }

      Serial.print("\n");
      Serial.print(cm);
       return cm ;

}


/////////////////////////////////////////  Access
Granted    ////////////////////////////////////
void granted () {
   if (motDirection == 0)
        {
          delay(200);
          Serial.println("Valid card. Opening drawer (initation of the sys-
tem).");
          digitalWrite(in1,LOW);
          digitalWrite(in2,HIGH);
          motDirection = 1;
        }

      else if (motDirection == 1)
        {
          delay(200);
          Serial.println("Valid card. Closing drawer.");
          digitalWrite(in1,HIGH);
          digitalWrite(in2,LOW);
          motDirection = 2;
        }
      else if (motDirection == 2)
      {
        delay(200);
        Serial.println("Valid card. Opening drawer.");
        digitalWrite(in1,LOW);
        digitalWrite(in2,HIGH);
        motDirection = 1;
        }
}
///////////////////////////////////////// Get Tag's UID
///////////////////////////////////
uint8_t getID() {
  // Getting ready for Reading Tags
  if ( ! mfrc522.PICC_IsNewCardPresent()) { //If a new Tag placed to RFID
reader continue
    return 0;
  }
```

```
  if ( ! mfrc522.PICC_ReadCardSerial()) {   //Since a Tag placed get Serial
and continue
    return 0;
  }
  // There are Mifare Tags which have 4 byte or 7 byte UID care if you use 7
byte Tag
  // I think we should assume every Tag as they have 4 byte UID
  // Until we support 7 byte Tags
  for ( uint8_t i = 0; i < 4; i++) {  //
    readCard[i] = mfrc522.uid.uidByte[i];
  }
  mfrc522.PICC_HaltA(); // Stop reading
  return 1;
}

///////////////////////// Check if RFID Reader is correctly initialized or not
/////////////////////
void ShowReaderDetails() {
  // Get the MFRC522 software version
  byte v = mfrc522.PCD_ReadRegister(mfrc522.VersionReg);
  // When 0x00 or 0xFF is returned, communication probably failed
  if ((v == 0x00) || (v == 0xFF)) {
    Serial.println("Communication Failure");
    Serial.println("Check Connections");

    while (true); // do not go further
  }
}

/////////////////////////////////////////// Read an ID from EEPROM
//////////////////////////////
void readID( uint8_t number ) {
  uint8_t start = (number * 4 ) + 2;     // Figure out starting position
  for ( uint8_t i = 0; i < 4; i++ ) {     // Loop 4 times to get the 4 Bytes
    storedCard[i] = EEPROM.read(start + i);   // Assign values read from
EEPROM to array
  }
}
/////////////////////////////////////////// Add ID to
EEPROM   ///////////////////////////////////
void writeID( byte a[] ) {
  if ( !findID( a ) ) {     // Before we write to the EEPROM, check to see if
we have seen this card before!
    uint8_t num = EEPROM.read(0);     // Get the numer of used spaces, posi-
tion 0 stores the number of ID cards
    uint8_t start = ( num * 4 ) + 6;  // Figure out where the next slot starts
    num++;                    // Increment the counter by one
    EEPROM.write( 0, num );     // Write the new count to the counter
    for ( uint8_t j = 0; j < 4; j++ ) {   // Loop 4 times
      EEPROM.write( start + j, a[j] );  // Write the array values to EEPROM in
the right position
    }

    Serial.println("Added");
    delay(1000);
  }
  else {

    Serial.println("Failed!");
    Serial.println("wrong ID or bad EEPROM");

  }
}
/////////////////////////////////////////// Remove ID from
EEPROM   ///////////////////////////////////
void deleteID( byte a[] ) {
```

```
  if ( !findID( a ) ) {      // Before we delete from the EEPROM, check to see
if we have this card!

    Serial.println("wrong ID or bad EEPROM");

  }
  else {
    uint8_t num = EEPROM.read(0);   // Get the numer of used spaces, position
0 stores the number of ID cards
    uint8_t slot;       // Figure out the slot number of the card
    uint8_t start;      // = ( num * 4 ) + 6; // Figure out where the next
slot starts
    uint8_t looping;    // The number of times the loop repeats
    uint8_t j;
    uint8_t count = EEPROM.read(0); // Read the first Byte of EEPROM that
stores number of cards
    slot = findIDSLOT( a );   // Figure out the slot number of the card to de-
lete
    start = (slot * 4) + 2;
    looping = ((num - slot) * 4);
    num--;      // Decrement the counter by one
    EEPROM.write( 0, num );   // Write the new count to the counter
    for ( j = 0; j < looping; j++ ) {         // Loop the card shift times
      EEPROM.write( start + j, EEPROM.read(start + 4 + j));   // Shift the ar-
ray values to 4 places earlier in the EEPROM
    }
    for ( uint8_t k = 0; k < 4; k++ ) {         // Shifting loop
      EEPROM.write( start + j + k, 0);
    }
    Serial.println("Eliminated");


  }
}
////////////////////////////////////////// Check
Bytes   ///////////////////////////////////
boolean checkTwo ( byte a[], byte b[] ) {
  if ( a[0] != 0 )      // Make sure there is something in the array first
    match = true;       // Assume they match at first
  for ( uint8_t k = 0; k < 4; k++ ) {   // Loop 4 times
    if ( a[k] != b[k] )     // IF a != b then set match = false, one fails,
all fail
      match = false;
  }
  if ( match ) {      // Check to see if if match is still true
    return true;      // Return true
  }
  else  {
    return false;       // Return false
  }
}
////////////////////////////////////////// Find
Slot   ///////////////////////////////////
uint8_t findIDSLOT( byte find[] ) {
  uint8_t count = EEPROM.read(0);       // Read the first Byte of EEPROM that
  for ( uint8_t i = 1; i <= count; i++ ) {    // Loop once for each EEPROM en-
try
    readID(i);                  // Read an ID from EEPROM, it is stored in
storedCard[4]
    if ( checkTwo( find, storedCard ) ) {   // Check to see if the storedCard
read from EEPROM
      // is the same as the find[] ID card passed
      return i;         // The slot number of the card
      break;          // Stop looking we found it
    }
  }
}
```

```
////////////////////////////////////////// Find ID From
EEPROM   ///////////////////////////////////
boolean findID( byte find[] ) {
  uint8_t count = EEPROM.read(0);     // Read the first Byte of EEPROM that
  for ( uint8_t i = 1; i <= count; i++ ) {    // Loop once for each EEPROM en-
try
    readID(i);          // Read an ID from EEPROM, it is stored in stored-
Card[4]
    if ( checkTwo( find, storedCard ) ) {   // Check to see if the storedCard
read from EEPROM
      return true;
      break;  // Stop looking we found it
    }
    else {    // If not, return false
    }
  }
  return false;
}

////////////////////// Check readCard IF is master-
Card   ///////////////////////////////////
// Check to see if the ID passed is the master programing card
boolean isMaster( byte test[] ) {
  if ( checkTwo( test, masterCard ) )
    return true;
  else
    return false;
}
```