Performance optimization and reliability improvement of railroad signaling in Kazakhstan with FPGAs

Sultan Kabi, Master's degree

Submitted in fulfilment of the requirements for the degree of Master of Science in Electrical and Computer Engineering

School of Engineering and Digital Sciences
Department of Electrical and Computer Engineering
Nazarbayev University

53 Kabanbay Batyr Avenue,
Astana, Kazakhstan, 010000

Supervisor: Nursultan Kabylkas
Co-supervisor: Akhan Almagambetov

Abstract

In this study the software implementation of a simulation of the railroad station and optimal pathfinding between two points(segments) was modeled for the railroad systems. The prototype processing unit will be created and evaluated on a FPGA equipment, and Java algorithm specifically designed for efficient hardware implementation and simulation of the track design. The autonomous system uses a linear approach to manage the system, considering inputs that include the train's current states, segment's current state(track, junctions etc.), coordinates and type. This system, which is linked to each individual train controller, determines the best option for all operating locomotives by sending data to the Java application that simulates the railway layout(design) and showing real-time position of the train. Sensors that are positioned at different points on the rails serve as the inputs, while the switches which regulate the pathway serve as its output. Remote transferof data is a crucial resource for a firm that requires immediate access to information as a factor of strategy in making choices. Monitoring specific characteristics in train industry might assist avoiding incidents and enhance customer services.

Key words: FPGA, automation, railroad system

Acknowledgements

First and foremost, I want to express my gratitude to my supervisors, Prof. Nursultan Kabylkas and Prof. Akhan Almagambetov. Because of their oversight and direction throughout my academic career, I was able to successfully complete this Master's program. I appreciate the many hours of discourse they have offered for me, especially in the areas of communications and embedded systems.

Second, I'd want to thank the administrative personnel at Nazarbayev University for their efforts, which are often overlooked. I'd also like to thank the Student Affairs staff for their counsel and prompt responses during my two years of study.

Third, I want to convey my gratitude to my family, partner, and friends for their support and advice.

Table of Contents

Chapter 1 - Introduction

1.1 - General

Nowadays, transportation networks rely significantly on railroad automation. It allows for the efficient and rapid operation of the train system, and its implementation has encouraged the development of various novel technologies. Many countries, for example, have started employing digital monitoring devices to monitor the location of trains on the system in addition to the condition of railroads. The network of devices keeps track of the variables under control by transmitting data acquired by each tracking equipment to a central computer, where it is analyzed. The rapid development in engineering has resulted in electronic gadgets that employ sensors to track train action throughout its path. Those very dependable electronic technologies have reduced the odds of train collisions to almost none, and it is currently being utilized to replace the old signaling systems. Monitoring equipment are installed alongside the railway line to avoid trains from taking the wrong direction, therefore minimizing collisions and losses. This progress has lately contributed to a more automated railroad system, in which a train that mimics a remotely operated automobile may be constructed, reducing the demand of personnel and its functioningis entirely managed by a computer.

1.2 – Problems

- The main problem Kazakhstan has in present is that most of the railroad systems are based on outdated software and hardware. Most of the systems work using a relay operation which in practice is not efficient that leads to lateness and errors, that is why it needs a constant monitoring.
- Overbudgeting and space usage, every relay used in railways costs around 30,000tenge.

1.3 - Techniques

To implement those methods the Field Programmable Gate Arrays (FPGA) hardware integration as well as Java based simulation was chosen. This method has a number of benefits, FPGA application enables quick production and minimal experimental expenses. FPGAs additionally present cost-effective and useful alternatives for particular uses, opening up new avenues for creating adaptable computer networks. FPGAs intests provide architects the option to modify elements of the circuitry for optimized performance, as well as do additional tests to enhance the concept.

Modeling performance is a key benefit of hardware-based implementation over software execution. The use of hardware-driven simulations allows the simulation process to capitalize on the benefits of simultaneous command processing. Other advantages of programmable hardware include the ability to do bit-level manipulations on non-powers-of-two-word sizes and the flexibility to allocate only a specific number of bits to storing variables within it. Therefore, the Field Programmable Gate Array (FPGA) combines the adaptability of software with the speed of hardware.

Chapter 2 - Background

2.1 - Railway history in Kazakhstan

Figure 2.1: Timeline of railroad history in Kazakhstan

The very first railroad line opened in 1894 in Kazakhstan, “Pokrovskaya Svoboda”,which runs 130 kilometers, completed on October 25th. In 1901, work on a railroad connecting important towns in Kazakhstan commenced in the country's west. First locomotive between Orenburg and Kubek terminal was opened in January 1904.

In 1926 development begins on the Turkestan-Siberian Railway, which will takeapproximately five decades to finish and is one of the largest-scale constructions of the freshly formed Soviet Union.

In 1991 the first operational cargo train traveled over the railway connecting China and Kazakhstan's then-capital of Almaty in July of 1991.

Kazakhstan Temir Zholy (KTZ) formed in 1997 to own and operate the national railway network as well as provide public transportation.

The Kazakhstan's railway transportation reformation initiative begins in 2001. This trend started with the construction of the 187-kilometer-long Aksu-Degelen railroad line, which provided the quickest route from the Semey area to Pavlodar.

2008 - The new TRANSKAZAKHSTAN lines Shar - Ust-Kamenogorsk, an estimated 150-kilometer-long railroad line linking Kazakhstan's national railways system within its boundaries, is built, expanding Kazakhstan's export and transit possibilities.

The Korgas Pass rail link has been constructed in 2012, connecting the China-Kazakhstan borders with China's inland trains.

The Shanghai Lianyungang station, an agreement involving Kazakhstan and China, is unveiled during Kazakh President Nursultan Nazarbayev's official travel to Shanghai in 2014.

The construction of the FEZ Khorgos entryway. The Khorgos terminal, known as the"Silk Road Economic Belt," is the world's largest dry dock, where East connects West, allowing products to transit more effectively from China to Europe. The 214-kilometer-long railroad route Arkalyk-Shubarkol was additionally inaugurated, cutting transit distances between Central Kazakhstan and northern parts by atotal of 550 kilometers.

2.2 - FPGA

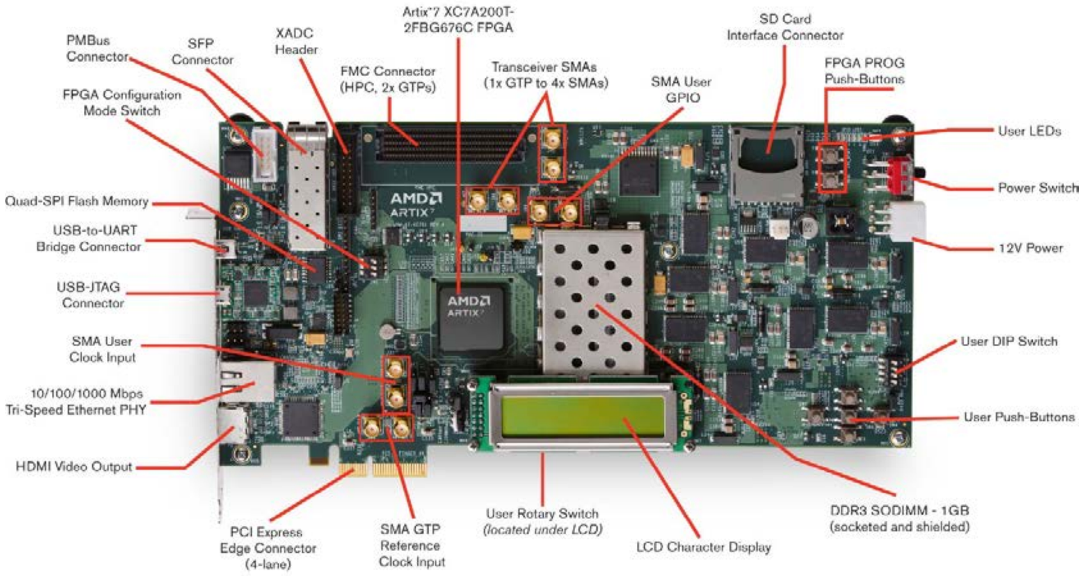

Figure 2.2: Xilinx Artix 7 FPGA AC701 Evaluation Kit specification [19]

A brief explanation of the Xilinx software suite is provided. Understanding the technology utilized in industry to build FPGA prototypes is crucial for developing efficient models and finished solutions. Xilinx is the major FPGA supplier, and its software is meant to assist designers with FPGA design challenges. Xilinx provides a comprehensive suite of integrated system architecture products to help improve the design workflow and minimize working hours. "Integrated application instruments" are widely used to describe tools for creating, editing, compiling, linking, loading, and debugging high-level coding, usually in C or C++, for execution on a processing engine. Xilinx provides a design product called the Software Development Kit (SDK).

SDK provides a comprehensive solution for developing interconnected, configurable systems. The setup package includes the previously mentioned design goods, documentation, and intellectual property. Chip Scope Pro analyzes all intrinsic FPGA communication, including embedded CPU busses. Chip Scope Pro uses FPGA setup to identify flaws and change designs in hours or minutes, rather than months or even years as in traditional ASIC design. The inbuilt logic analyzer supports in problem detection and debugging by providing advanced triggering, filtering, and displaying options.

FPGA application provides rapid production with low experimental costs. FPGAs also offer cost-effective and useful alternatives for specific applications, expanding the possibilities for constructing adaptive computer networks. FPGAs in testing enable architects to fine-tune circuitry components for enhanced performance while also running more tests to improve the idea. FPGAs (Field-Programmable Gate Arrays) can provide a flexible foundation for developing unique electrical logic in railroad system control. Their versatility, multitasking capabilities, and ability to handle a wide range of duties make them valuable assets in a variety of rail sector applications. FPGAs can be utilized in railroad control systems for real-time signal processing, protocol communication, and train control, specifically Positive Train Control (PTC) executions. They contribute to the efficiency of interlocking systems, provide safe train motion within rail terminals, and play a significant role in managing railroad signaling infrastructure. FPGAs are crucial for handling and interpreting massive volumes of data generated by railroad systems in order to optimize operations and maintenance. Their versatility enables the development of fault-tolerant principles that ensure the system's reliability in the face of breakdowns. Custom hardware acceleration, enabled by FPGAs, increases the efficiency of specific train control algorithms. In addition, FPGAs enable the development of adaptive control systems that may be modified to respond to structural or operational changes. Their high-speed parallel processing capabilities make them ideal for applications that require rapid information and response processing, which is vital for maintaining the security and efficacy of train operations.

FPGAs are expected to become increasingly important in the upgrading and optimization of railroad control networks as technology improves, providing an adaptive and effective tool for meeting the rail industry's complex demands.

Relays have played an important role in the advancement of electronic systems, acting as critical components for controlling and managing electrical circuits. As time passed, relay technology has changed dramatically, from traditional mechanical switches to modern solid-state alternatives. This essay goes into the historical evolution of relays, studying the issues faced by each technological advancement and assessing the current condition of relay technology.

2.3 - Relay evolution

The history of relay development may be traced back to the mid-19th century, when early telegraph systems used electromagnetic switches to boost signals that were weak for long-distance communications. These early relays used electromagnetic coils and mechanical contacts, signaling the beginning of electro-mechanical relays. As requests for advanced and dependable control systems grew, engineers worked to overcome the constraints of the first designs. Solid-state relays (SSRs) were introduced in the mid-twentieth century, and they revolutionized relay technology. These relays substituted mechanical components with semiconductor components, resulting in faster switching times, increased dependability, and lower maintenance requirements. SSRs offered advantages like as silent operation, a longer lifespan, and resistance to physical wear, which made them ideal for applications that required precision and efficiency. The late twentieth century saw the introduction of Programmable Logic Controllers (PLCs) and microprocessor-based control systems, which influenced relay technology and gave rise to intelligent or smart relays. These embedded microprocessors and programmable logic devices improve functionality, enabling advanced control techniques as well as simpler integration into complicated automation systems. Smart relays included sophisticated characteristics such as programmability, communication, and self-diagnosis. Field-Programmable Gate Arrays (FPGAs) recently revolutionized relay technology, giving engineers unprecedented flexibility in creating unique digital circuits. FPGAs in relay development allow for the development of complex algorithms for control, processing of signals tasks, and protocols for communication, making them suited for a wide range of applications from industrial automation to aerospace systems.

The present relay technology landscape includes a wide range of solutions adapted to specific industry requirements. While classic electromechanical relays are still used in some situations, solid-state relays dominate high-performance, high-reliability applications. Smart relays and FPGAs are at the leading edge, pushing the limits of modification, flexibility, and interaction.

2.3.1 Relay manufacturing process

The relay manufacturing process encompasses several steps, from design to distribution:

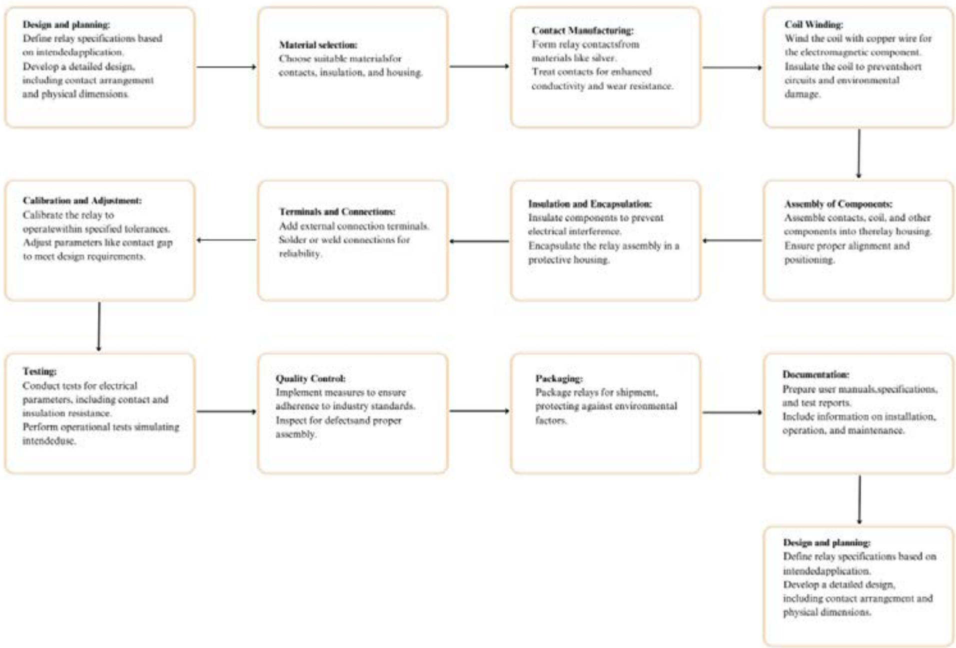

Figure 2.3 Relay Manufacturing process

Manufacturers may use automated technologies during the procedure, according to the level of detail and volume of production. Specialized relays that operate, which include solid-state or ones employed in certain industries, may have a distinct design and manufacturing process.

2.4 Terminal signaling and control in modern stations

Railroad terminals use an assortment of communication and control technology to guarantee that trains travel safely and efficiently. Semaphore signs with moving arms indicate track occupancy. Modern indicators have colored illumination. including red for breaks, yellow for caution and green for clear. Shift machineries position turnouts and toggle switches, enabling trains to change tracks. Derailers prohibit unlawful movement along specified lines.

Occupancy detectors and circuits set across rails monitor train activity to ensure safety. To avoid conflicts between train movements, interlocking systems sync switches and indicators. Centralized Traffic Control (CTC) systems track and regulate train movements through dispatcher-controlled panels.

Communication is facilitated through dispatching radios and train-to-train communication devices. Positive Train Control (PTC) systems manage train speed and safety automatically, with onboard equipment and centralized back-office systems. Electrification equipment like overhead wires or third rails provide power to locomotives in electrified terminals.

Information and data systems, including computer systems and terminal information displays, process and manage train actions, scheduling, and terminal operations. Safety measures include grade crossing warning devices and surveillance cameras, while access control systems limit terminal access.

The integration of computerized and automated technologies, driven by advancements, enhances overall terminal administration and safety. The selection of devices depends on terminal complexity, scale, and operator preferences.

Relays of various types are used in railroad systems for signaling and control. The precise sorts of relays used can vary depending on the devices and signaling networks used. The following are some frequent types of relays found in railroad systems:

Circuits for Tracks:

Occupancy Detectors - In track circuits, these relays identify the movement of a train on a specific piece of track. They are critical for signaling networks that guarantee safe trainseparation.

The specific relays used are determined by the railroad's signaling and control technology. Traditional relay-based systems continue to be used, but recent railways increasingly utilize digital and computerized communication systems, which may use microprocessor-based relays along with additional advanced technologies to improve effectiveness and dependability. It is important to take into account that signaling technology is constantly evolving, and new systems may emerge.

2.4.1 The objective of Railroad Relays

Railway relays play an important role in train operations. Security is required to ensure rail safety and prevent collisions. Signals operated by relays inform train operators if it is safe to proceed, stop, or slow down. Trains may move between tracks fast and reliably thanks to relays that regulate track switches. This allows for smoother train movement and more efficient railway capacity. Protection safeguards railway infrastructure by monitoring track conditions and issuing alarms or shutdowns in the event of difficulties such as track circuit faults or obstacles. To prevent mismatched train actions, interlocking systems rely largely on relays.

2.4.2 Working process of railway relays

Railway relays work on the concepts of electromagnetic and electrical control. They are basically electromagnetic gates that toggle between open and closed electrical circuits in accordance with predetermined conditions or input. Here's a short description of the way railway relays operate:

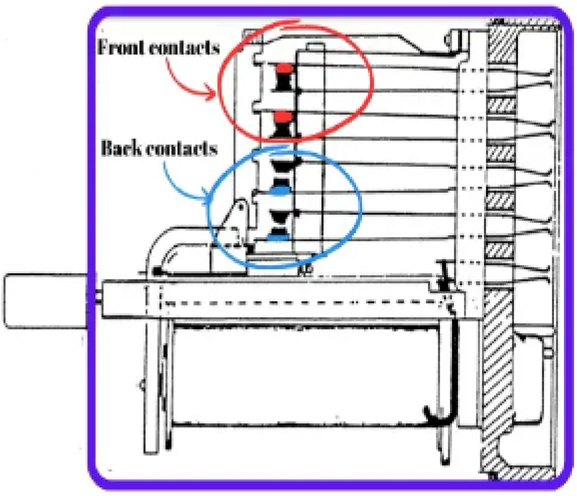

Figure 2.4: Relay energized, front contacts made, back contacts disconnected

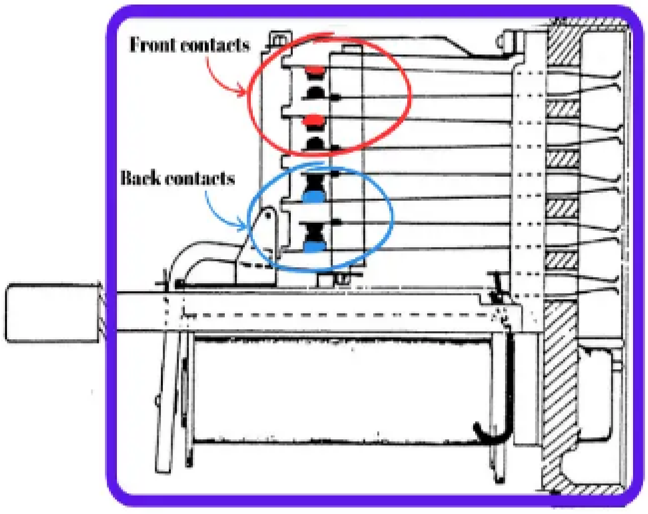

Figure 2.5: Relay de-energized, front contacts broken, back contacts connected

1. Input: Railroad relays are activated by input signals received from a variety of sources, including rail sensors, control panels, and dispatch stations. In the scenario of track circuit relays, a train approaching a piece of track finishes a circuit of electricity through its wheels, which serves as

an electrical signal to the relay.

2. Relay Activation: A electrical signal that stimulates the relay's electromagnetic coils. This magnetizes the iron core, which pulls the relay terminals into an open or shut position based on the relay's structure and function.

3. Control of Auxiliary Components: Opening or shutting relay contacts controls a variety of external devices, including signals, track switches, and alarms. For example, shutting a relay contact may trigger a red signal light indicating a stop, whereas opening a contact may cause a track switch to shift position.

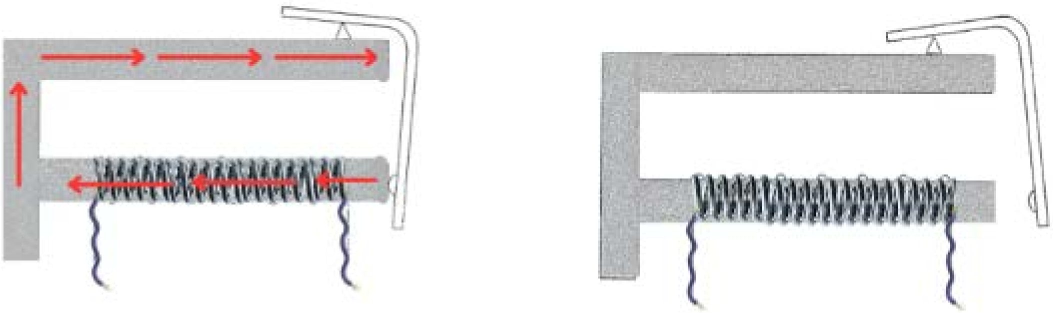

Figure 2.6: On the left: Relay energized and armature pulled

On the right: Relay de-energized and armature opens

2.4.2.1 Parts of railroad relays

Relays are made up of electrical parts that are intended to move small metal plates in a magnetic field to either close or open electrical circuits. Relays come in two varieties: DC and AC. For safety reasons, DC relays are utilized in AC traction regions and vice versa. The following elements comprise a DC relay:

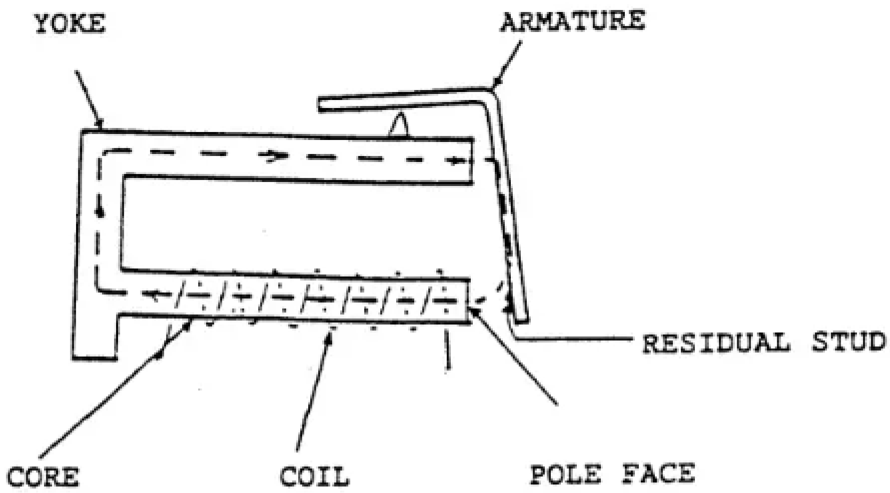

Figure 2.7: DC relay configuration (Neutral relay)

1. Electromagnetic Coils: An iron core is encircled by wire coils in railway relays. A magnetic field is produced when current is passed via these coils.

2. Armature: This is a portion of the relay that is attached to the iron core and rotated to the yoke. In the iron core, the magnetic field produces the south and north poles, which are opposite poles. The south pole effortlessly traverses the link to the armature, resulting in opposing poles at the other end of the armature. The armature travels in the direction of the iron core when its south pole pulls the iron core's north pole, in the same way that unlike poles attract.

3. Drive bar: A steel bar with several connections attached to it is mounted to the top of the armature. This drive bar rises upward, moving the contacts together with the armature as it draws closer to the pole.

4. Contacts: A few groups of electrical connections that might be open or shut are located inside the relay. External circuits are linked to these connections. There are two different kinds of contacting: front contact and back contact. There are opposing actions performed by the front and rear contacts. The rear contact breaks if the first contact is connected, and vice versa.

When the track is empty, the front contacts join; when the track is filled, they disengage. In the event that there is no train, the front contacts link, the magnetic field is energized by the electrical circuit, and vice versa. You can think of each set of contacts as a distinct switch. They can turn on and off sixteen circuits if there are sixteen contact pairs.

Front contacts are nearly always supplied with the back contacts. According to the signal design, the back connections can be utilized to create circuits that activate appropriate light signals or for other uses. For example, the rear contact is able to be employed to light up red if the top contact lights up green. As a result, numerous types of relays with varying back and front contacts counts and configurations exist, each intended for a particular purpose.

2.4.3 Railway Relay Types

Relays that are neutral:

Both orientations of the voltage's polarity might cause the current to flow through the coil. (see Figure 2.4)

Biased Relay

Through the coil, this kind of relay can only detect current flowing in one direction. On one of the ends of the armature is a permanent magnet. The armature is resisted if the electrical current travels in its direction. On the other hand, attraction rather than repulsive force results from the magnetic field and coil flux having the same directions while the current flows in the opposite way.

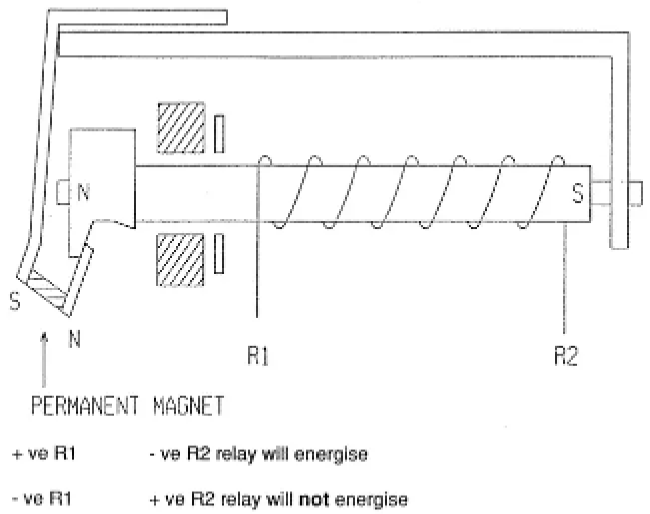

Figure 2.8 Biased Relay configuration

Slow Release/Time Delay Relays

These relays feature capacitor or copper covers that wrap around the coil. As the present state changes, the relay's release delays due to the capacitor's stored charge. After the capacitor's stored charge releases via the coil, the relays that control it are released. This method, the electricity is turned on or off after a specific time interval. The time interval can be custom set.

Polarized relays

This is a relay that permits current to flow in only one direction. It functions similarly to a biased relay however the operating principle is different. A diode has been linked in series with the relay coil. This stops the flow of electricity in the opposite direction.

2.5 - Interlocking system

A **station yard** consists of

- Signals - give information to the driver
- Track Circuits - detect presence of the train
- Points - sets or diverts the direction of train

These elements are the deciding factors in the safe movement of trains.

All these objects form input to a Centralized system, which monitors the state of these devices and based on the Interlocking rules and Commands given by the station master decide the safe movement of trains inside a station yard, thus the term INTERLOCKING comes in to existence.

2.5.1 - Types of Interlocking Systems

Cabin Interlocking System (Mechanical Interlocking)

Mechanical interlocking industries use a locking platform comprised of steel bars placed in a grid. The supports that link together in one direction have connections to the levers that operate switches, brakes, indications, and other appliances. The design of the bars prevents mechanical interference during junction locking among the two bars if one lever's function conflicts with another's purpose.

This, in turn, prevents the overlapping gear motion from occurring. In purely mechanical plants, levers regulate field appliances, such as signals, via wiring or manual rodding. Given that the levers need to offer a mechanical benefit to the operator, they are around shoulder level.

Panel Interlocking System (PI)

In many IR stations, the mechanism in use is called Panel Interlocking (PI). This is where each of the switches which regulate the points and communications are employed.

Route Relay Interlocking System (RRI)

In order to regulate train movements at intersections, crossings, and other crucial locations along the rail line, the Route Relay Interlocking System (RRI) is a contemporary electrical interlocking system that is extensively employed in railway signaling. Its versatility, dependability, and integration surpass those of conventional mechanical interlocking systems by a great margin.

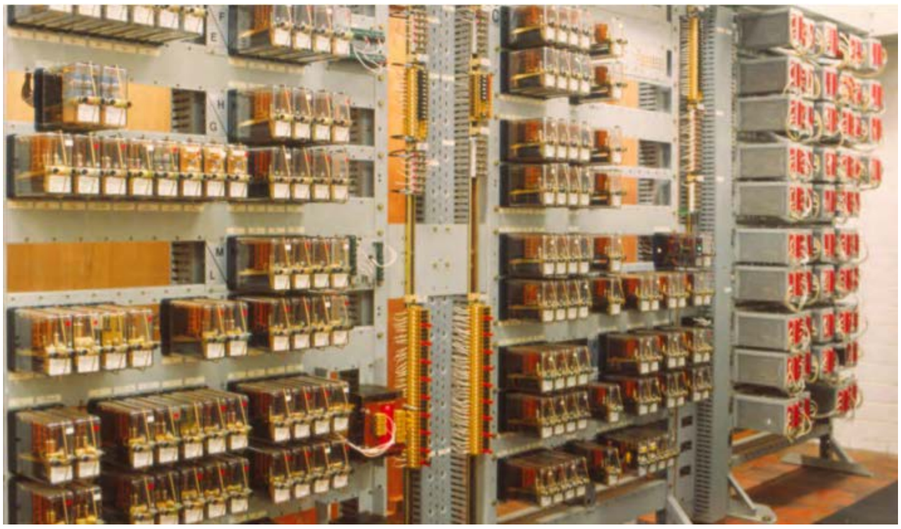

Figure 2.9: Route Relay Interlocking System implemented in train station

Solid Sate Interlocking System (SSI) or (CIS)

Solid State Interlocking (SSI), also known as the Computerized Interlocking System (CIS), represents a significant advancement in railroad signaling systems. With digitized electronic systems substituting traditional relay-based interlocking systems, train operations can be more flexible, dependable, and safe. Railways around the world have embraced SSI due to its several advantages over traditional interlocking technologies.

SSI controls and manages railway signaling systems using digital computer equipment. Software algorithms and microprocessors are used to evaluate signals obtained from railroad circuits, locomotive detecting devices, and operator instructions, as well as to determine train safety. Compared to relay-based structures, SSI's digital nature allows for faster processing, greater precision, and more complicated control logic.

Figure 2.10:Equipment in Solid State Interlocking

2.5.1.1 Relay Interlocking System (RRI)

In Route Relay Interlocking or popularly known as RRI, the Control Rules are implemented using Relays. In RRI the relay circuits are built using station Control Table as input and Interlocking rules as logic.

Relays: The fundamental building block of the traditional route-setting interlocking is the relay. There are two generic types of relays used for all interlockings.

1) Fail-safe: This type of relay is known generically as type N in UIC standard 736i. The best-known family of signaling relays in this category is the BR930 series

2) Type C: Known as type C in UIC standard 736i. It is not guaranteed to behave in the inherently fail-safe manner described above.

Track Circuit Relays: They are devices that detect the activity of trains on specified sections of track. They are used to regulate signaling and track trains for signaling and safety reasons. Track circuit first appeared in 1872, its purpose was to tell the signalman about the train's location. Now, track circuits signals for control, locking stages, and other operations. The working mechanism of railway circuit relays are examined in the figure below.

When there are no trains on the track circuit, the current flows as follows:

1) Through the feed resistor, connecting the (+ve) charge onto the positive rail.

2) Going across the railroad relay coils and then switching back over to the negative track from the positive rail.

3) From the negative rail towards the battery's negative terminal.

The circuit is notably a "series" system that is "closed" or finished.

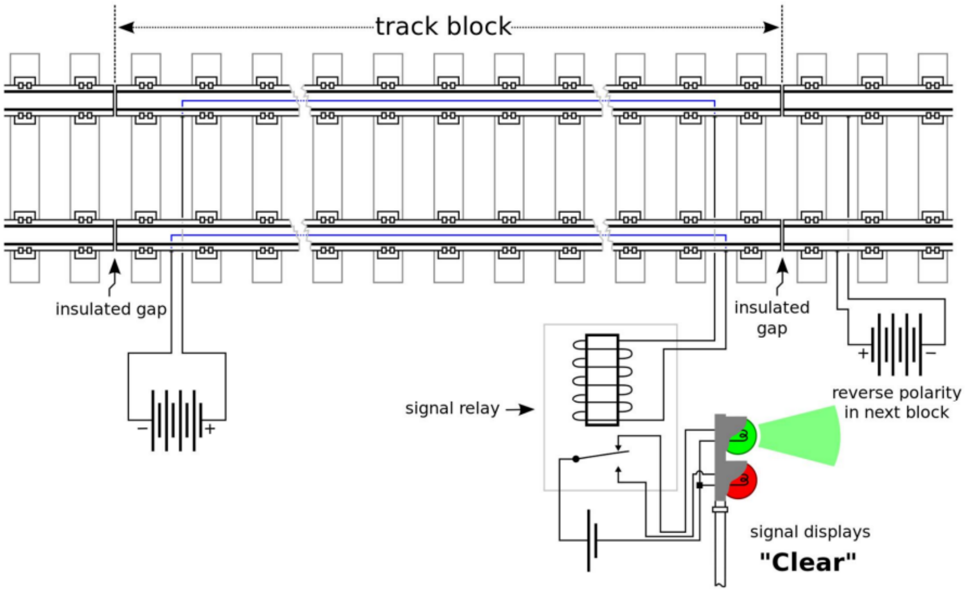

Figure 2.11: A relay function in track circuit when a track is not occupied

When a train is operating on the track circuit, the current flow changes as follows:

1) From the battery (+ ve), via the positive rail and the supply resistor

In 2) 3), the train's wheels and axles cause a short circuit among the (+) and (-) tracks. The rims and axle, the Lowest Resistant Path, the negative track, and finally the battery will get most of the electrical current.

A very tiny quantity of current will continue to pass through the rails and reach the track relay. The relay's function is deactivated when the current is redirected.

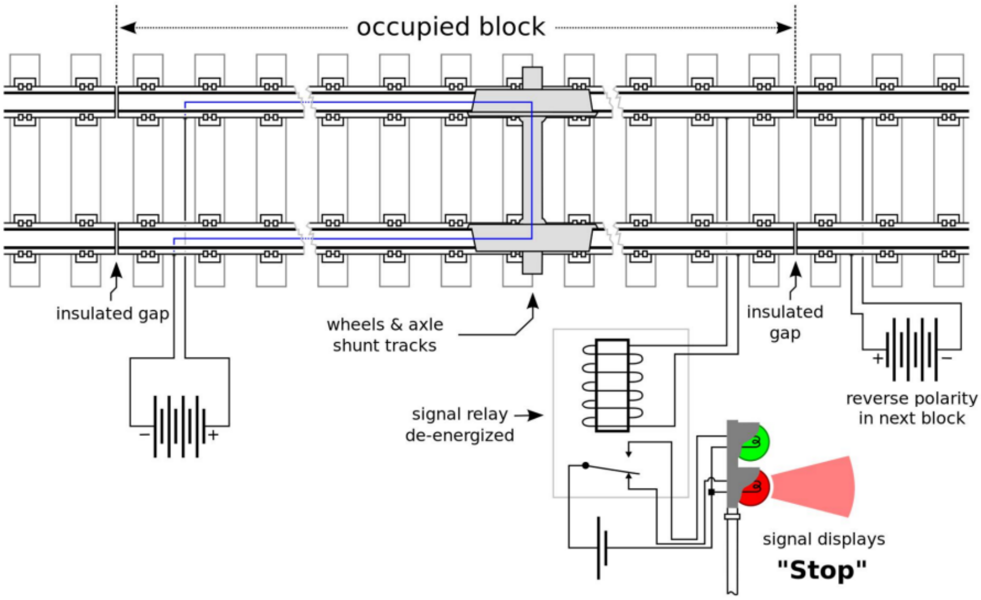

Figure 2.12: A relay function in track circuit when track is occupied

Track circuit indication relay.

This is a secondary relay for the track relay, which is used to signal track occupancy. This relay is operated by connecting the track relay's back contact to it. This relay is vital because it provides a delay time during voltage pickup, to make sure the track does not display indicators of clear in the event that the track circuit charge briefly rises up and drips. This ensures that traffic moves safely.

Three-aspect light signaling relays

Color rail light signaling relays regulate the many signal elements provided to train operators. These relays assess if a signal should be appropriate color signaling whether it is allowed to carry on, approach cautiously, or stop. A number of relays is utilized to operate these signals that are programmed to function in a specific manner. The relay's accurate timing and shifting functions are critical for ensuring safe train operations.

2.6 - Railroad terminal overview

Rail line processes, storing facility procedures, and gate operations are the three main components that are interconnected in intermodal terminal management. Because a delay in one process may affect the others, it is important to make sure that they all work together effectively. For example, issues with storage yard operations can affect the efficiency and standard of services provided by the terminal, as well as cause delays at the railway line and gate operations. Typically, an intermodal rail containers terminal consists of the following components, each of which serves a distinct purpose:[2]

The **intermodal yard** is the station's core area where unit trains are unloaded and loaded using side-loaders or cranes with tire treads. Carrier unit trains can be more than 2 kilometers long due to their large size. Unit trains are typically broken into many portions, particularly when the infrastructure is of older construction. This enables for the movement of chassis across the yard at a midway juncture; otherwise, transfers between space for storage and unit trucks would take much longer. Hostlers transport cargo to the storage facility or trackside.

A type of yard is not common in current intermodal rail terminals, particularly if the facility has been upgraded from a conventional cargo to a container terminal. The major aim of a sort of yard is to assemble and disassemble freight trains carrying diverse types of cargo. This is critical since each railcar may be bound for various places and can be moved several times, either at the starting point, location, or intermediate points.

Intermodal operations require less shunting since it is easier to assign and arrange packaging on well vehicles than transferring the well vehicles themselves. Classification terminals are often placed at separate locations from intermodal yards. Sophisticated intermodal rail terminals are designed to handle containerized cargo efficiently; hence the old classification yard is less prevalent in this setting.

The **storage space** within an intermodal yard act as an intermediary among the road systems and an intermodal site. It often occupies a space comparable to the intermodal area, requiring an enormous space because of the spatial demands of current rail intermodal yards. Storage alternatives include grounded storage, which stacks containers on top of one another, and wheeled storage, which places containers on a chassis. Cargoes are transported directly to awaiting chassis for shipment, rendering the chassis a vital element in terminal operations.

Chassis storage is an established location wherein unoccupied chassis are maintained while they await assignment to a truck or holster. Some storage spaces are located inside the terminal itself, while others are in adjacent chassis pools. Chassis play an important function since they act as the link among railway and trucking.

The **repair/maintenance** area is an established spot for routine maintenance of the terminal's bulky machinery. It involves periodic servicing and maintenance to ensure that the terminal's equipment runs smoothly and lasts as long as possible.

Chapter 3 - Literature Review

The contemporary commuter rail network began decades ago, when the city's growing population required efficient public transportation. Aside from the railway system's power system, technological advancements in security measures such as signaling and train identification devices, as well as fully automated systems, have propelled the railway industry to the status of a highly sophisticated sector for transportation. Advances in technology have resulted in technological gadgets that use sensors to monitor train movements throughout their route [1]. This very reliable electronic technology has reduced the odds of train collisions to zero, and it is currently being utilized to replace the traditional signaling technology. Sensors are installed along the track in order to avoid any trains from taking the wrong direction, hence avoiding collisions or damage. Recent advancements have led to a more automated railway system, where a train mimics a remotely operated vehicle, decreasing the need for humans and totally operated via a computer [2]-[3]. Initially, a VHDL and Java applications was used to model an automated train station management network that controls trains in a single stop. Following that, a framework for system hardware realization using FPGAs is proposed [4]. There have been various attempts to construct a similar method, most notably from Chang et al [1] and Cataldi et al [2]. Both techniques selected software as the platform for their techniques. However, in this research, a hardware solution using an FPGA was used. There are various benefits to this method. FPGA integration enables quick execution and zero prototype expenses. FPGAs also provide cost-effective and practical options for specific applications, as well as new opportunities in developing reconfigurable computer systems. In evaluation, FPGAs give designers the option to modify elements within the system for optimization, as well as execute additional tests to enhance the design [5]. Main advantage of implementing hardware against software is simulation speed. Hardware-based simulation enables a modeling procedure to benefit from concurrent execution of commands. Additional benefits of programmable technology include the ability to dedicate only a particular number of bits to storing internal parameters. As can be seen, the FPGA blends software freedom with high-speed hardware [6].

Embedded systems exist in all sectors, from aviation to everyday use. New improvements in method's architecture enable the implementation of sophisticated applications.

Throughout the design and creation of an embedded system, particular procedure approaches are applied. These models typically require creating a working version for the finished system. Embedded systems are resource and cost-constrained, flexible, and offer immediate effect. Embedded issues can be tackled utilizing a variety of ways. [7]

While developing a system that is embedded, numerous design issues arise. These tasks determine which sort of chip will be used. Aside from system functionality and safety, the following factors must be addressed when designing a near-optimal system: Performance, cost, Power, Upkeep.

3.1 Existing solutions

Open Rails project offers a train simulator for the world's greatest collection of digital information - paths, locomotives, and events - which was originally produced by Microsoft's Train Simulator program.

Microsoft Train Simulator, that appeared in 2001, was a huge success, but Microsoft had multiple unsuccessful attempts in developing it further, which came to an end in 2009.

Open Rails is the replacement to Microsoft Train Simulator, with trains running on lines designed specifically for the game. However, there are a few distinctions worth noting.

Open Rails employs more latest technologies, allowing it to take advantage of newer graphics processors. This results in faster frame rates, more vibrant hues and specifics, larger screens, and cleaner action.[20]

The AnyLogic Company has become an international business based in the United States and Europe, having a worldwide community of collaborators. The company designs and creates modeling and simulation software for applications in business. AnyLogic(AL) is a simulation software suitable for various purposes. AL Cloud provides an online platform for execution of models and interaction. AnyLogistix is application for chain of custody modeling.

AL has transformed the way people create simulation models and increased their use in complicated corporate situations. Its unparalleled flexibility enables customers to record the complexities of almost any system at any degree of depth.[21]

3.2 Comparison

The existing solutions gives the user to monitor the simulation and give discrete pointers or timetables of train routes. The solutions mentioned do not consider usage of the simulations in real train stations. Most of them are for educational purposes or only to check the validity of the schematical view of the station. Whereas system proposed in the thesis offers connection to the FPGA which in turn can be implemented into the in real world stations.

Also, the use java language in the simulation proposed enables users to use it in any platform that has java, whether it is Windows or Linux, IOS or Android.

3.3 Open-source software

In today's digital landscape, when technology affects almost every part of our lives, open-source software has become known as a disruptive force. Unlike traditional proprietary software, which is built securely, open-source software promotes cooperation, accountability and innovation driven by communities. From computer operating systems to internet browsers, libraries to coding languages, open-source technology is nowadays a staple of modern computing, allowing individuals, programmers, and businesses to capitalize on its inventiveness, adaptability, and affordability. Open-source software is used in this thesis work to build a software simulation with broad functionality. The name of the open-source is “DEVS-Suite Simulator”. The simulation framework, built on DEVS and Constrained DEVS, provides flexible design and testing capabilities. It makes Component and Cellular Automata (CA) analysis easier, with distinct experiment creation, assessment and visualization capabilities. Model creation and experimenting are simplified by following software architectural principles. The system enables automated, controlled surveillance and troubleshooting, which benefits scholars, professionals, and instructors in the simulation and modeling field. It also allows automatic, adjustable design of experiments, DEVS-FMU integration, and Action Level Real-Time simulation. It also includes features like linear and superdense time-based information path calculation, multilevel continuous element representation. It ensures synced conceptual and soft immediate execution, as well as real-time animations and information trajectory display. In the figure below a project sample can be seen:

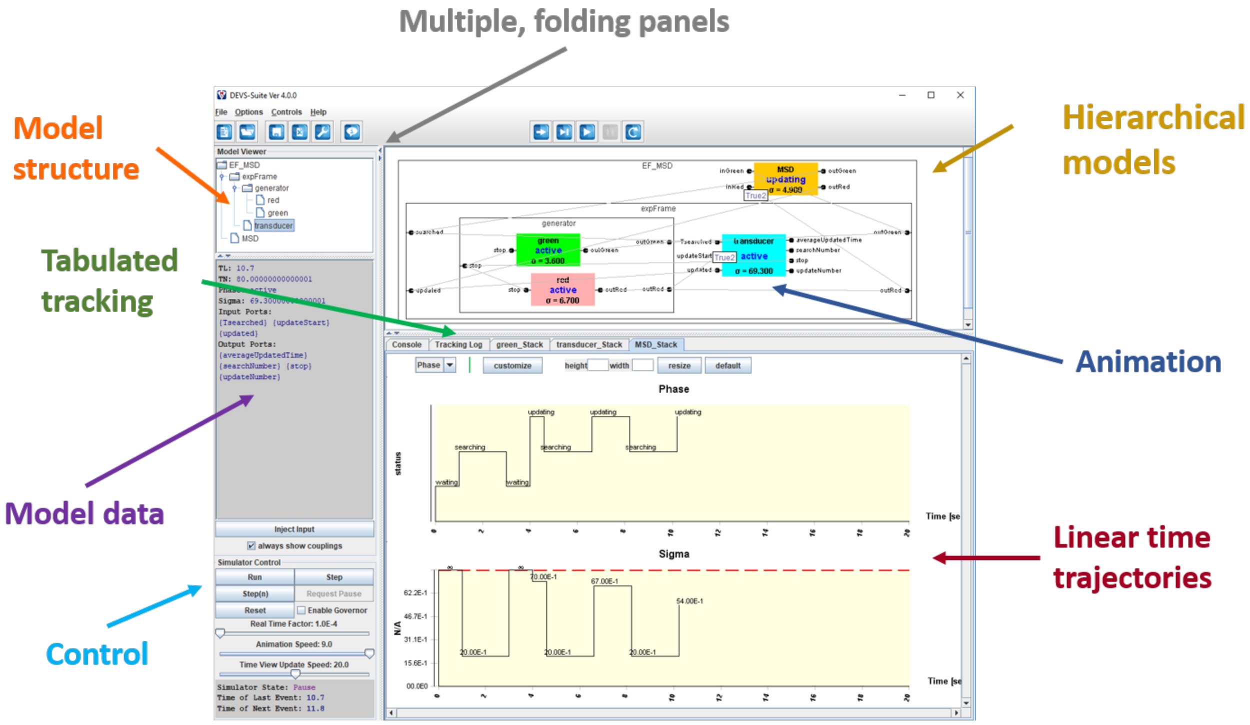

Figure 3.1: GUI of the open-source DEVS-Suite Simulator

DEVS-Suite Simulator uses "GNU Library or Lesser General Public License version 3.0 (LGPLv3)" license which can be accessed using a link: https://www.gnu.org/licenses/lgpl-3.0.txt, in full text, which is used in the Railroad simulation software built in this thesis work.

Chapter 4 - Methodology

The system is divided into two parts. The first one being a Java based software, where the visuals and simulation are shown. Previously explained design overview can be implemented in a software simulation. Various railway softwires are presently in use in several countries; nevertheless, these applications limit users to a constrained framework in which manual management for every railroad system component is required. These include set up routes and integrated algorithms within the code, giving consumers limited interaction options. In essence, those railway programs are software that focus primarily on train operating rather than providing a larger range of activities. The second part is calculations(rules), all the calculations in a modern railroad station are calculated using relays, which is outdated.

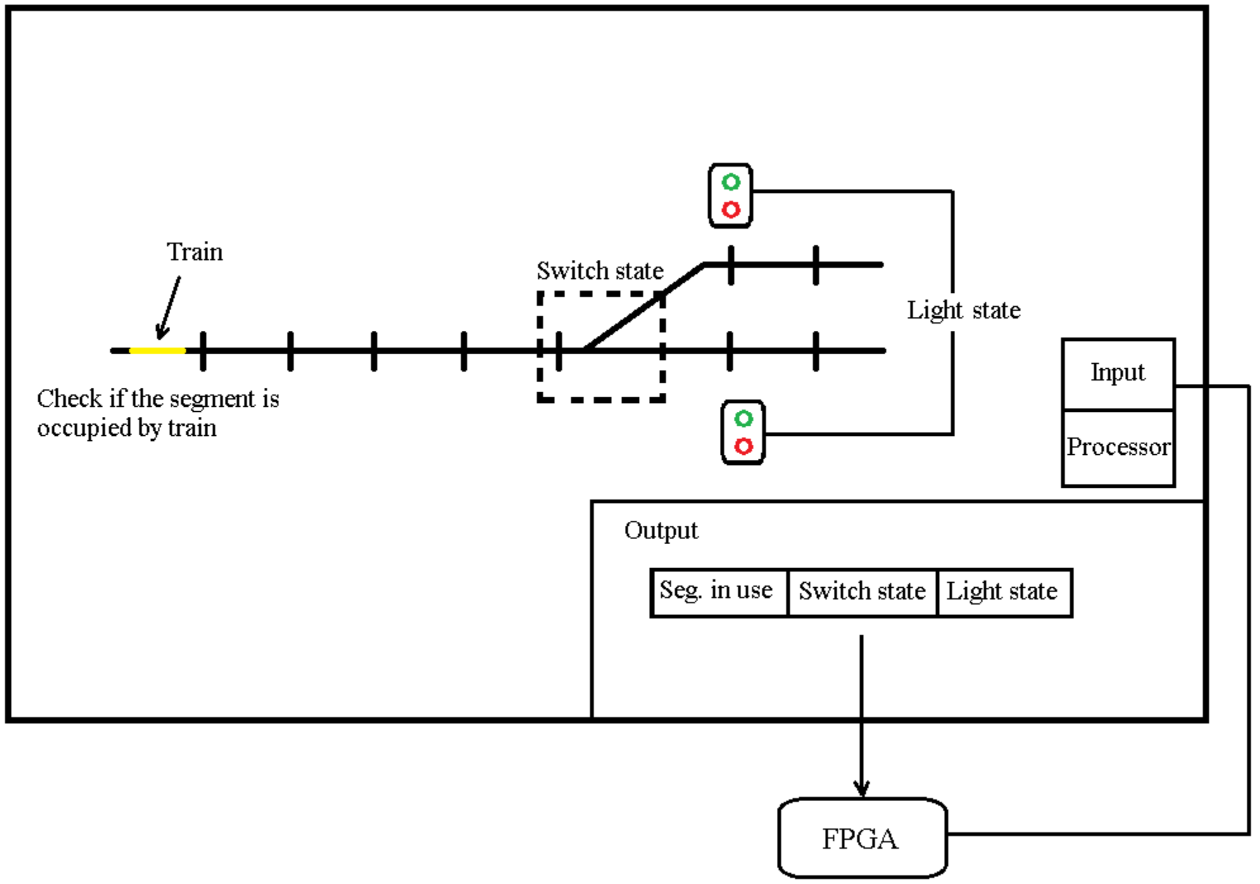

Figure 4.1: Working process of the simulation and connection of the FPGA

As seen in Fig 4.5, The simulation comprises of a working space in which all actions are displayed; the major function of this is to record the current positions of the switches and tracks, as well as to determine whether or not the train is occupying the track segments. This information is then sent to the FPGA, where all calculations take place. The FPGA, using the written code, can determine the optimum train motion to reach the destination B from position A. The FPGA then delivers the data returned to the software, where all of the computed actions are handled and compiled to demonstrate the process of implementing the rules supplied to the user. The data from the program to the FPGA is sent via Universal Asynchronous Receiver(UART). There is also a synchronous variant with an additional clock signal line, although it is not relevant to the aims of this study. Each bit of each byte is delivered for an identical amount of time (known as a time slot).

4.1 Java software

To simulate railroad stations a software was built. The software was coded in java language and built in Eclipse IDE.

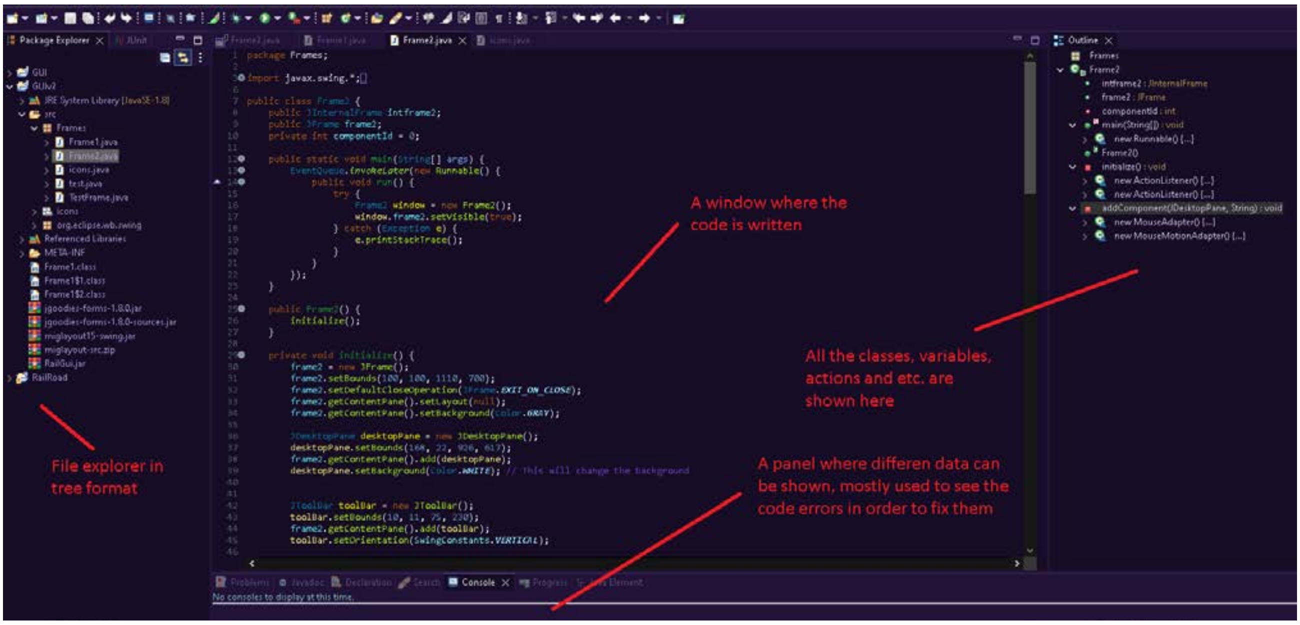

Figure 4.2: Eclipse environment explained

The reason a java language was used is because it has more pros that cons, for example:

- Procedural with object-oriented support

Java is a procedural programming language with object-oriented capabilities, despite the fact that it is frequently referred as an OOP language. It's really a hybrid language, which makes it very flexible and easy to use.

- Heightened security

While no programming language is totally immune to assaults or guaranteed to be devoid of vulnerabilities, Java delivers a level of greater security by protecting against specific security flaws seen in other languages. For example, the absence of references may prevent unauthorized access, hence resolving a common issue when programming in C. In the same line, employing conceptualization, transmission, and encapsulation can improve security even further.

- Platform-independent

Java programming is portable across platforms. It supports the write-once, run-anywhere (WORA) paradigm, which allows cross-platform Java applications to run without requiring code changes.

- Multithreaded

When it comes to managing numerous activities at once, a multi-threading CPU is significantly more effective; but, for best performance, software built to harness this capability must be used. Multithreading is supported by Java. Code that makes full use of multiple threads can be created in Java application creation, leading to more effective CPU resource utilization and improved performance.

- APIs

Application programming interfaces (APIs) are collections of commands or processes that define how distinct activities connect with one another. A variety of APIs are accessible, which usually simplifies Java development. Furthermore, because the features that are required do not need to be constructed from scratch, the implementation of numerous apps and present software becomes easier to manage.

- Memory management

Java memory management relies on a certain level of isolation. Memory space is made up of stack and heap spaces, and the JVM may assign memory in the correct places to aid in storing, managing, and restoring. These memory allocation characteristics result in increased operating efficiency.

- A distributed language

Another advantage of Java is that it is distributed. Java supports system-to-system information and code sharing, effectively enabling remote device collaboration for increased performance. Furthermore, it has Remote Method Invocation (RMI), which enables parallel computing in Java.

- Simple general-purpose language

One of Java's distinguishing advantages is its ability to function as a programming language for all purposes alongside C, PHP, and Python. As a result, programming in Java is quite adaptable and may be utilized for a variety of reasons.

4.2 GUI

The GUI of the software is mostly built using classes JFrame, which are frames or windows of the program. There are in total two frames the first being a “WelcomeWindow” and the second is working panel.

The working panel consists of ToolBox and JPanel. ToolBox is a box that displays all the models/objects that can be placed on a blank canvas which is a JPanel. Using only two of those elements a user can draw/build schematics of the railroad systems. Possibly creating a new custom-made design for the train stations.

Each segment is assigned a node number and is stored in map. Other than that, the segment information like: name of the segment(if it is a straight line or a junction etc.), coordinates and connection hierarchy.

There is a huge number of possible inputs(starting and ending points of the path). The “container” with an orange dot is a finishing point, application takes the first put element to be a starting point. This was implemented to broaden the functions of the software. The path searching is found using Breadth first search(BFS). BFS follows the concept of “go wide, bird's eye-view”. Instead of following a specific path to the end, BFS involves moving forward one neighbor at a time. This means the following:

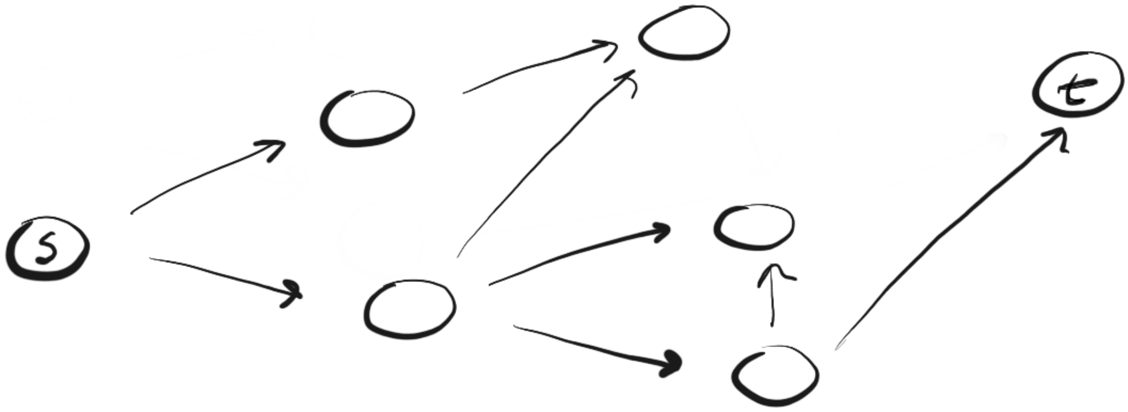

Figure 4.3: BFS explained.

Instead of following a path, BFS involves visiting the neighbors closest to s in one action (step), then visiting the neighbors' neighbors, and so on until “t” is found.

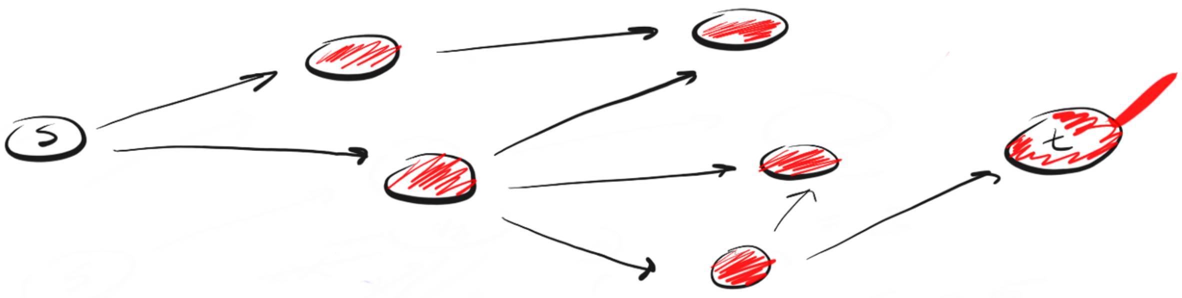

Figure 4.4: BFS logic of visiting nodes.

The next question is: how do you know which neighbors to visit first?

To do this, we can use the first-in-first-out (FIFO) concept of a queue. We put the vertex closest to us into the queue first, then its unvisited neighbors, and continue this process until the queue is empty or until we find the desired vertex.

In our case, the peaks will be the middle of the path and edges are the distance between points.

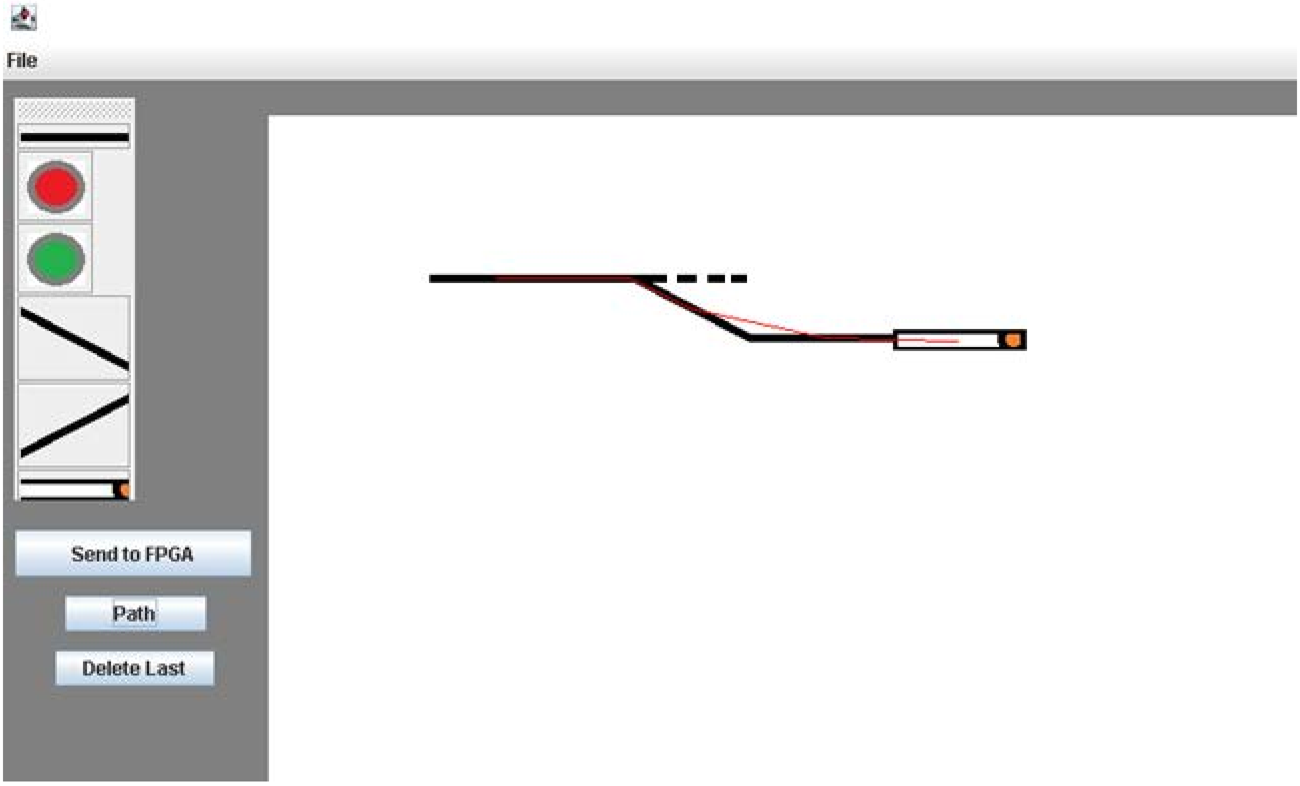

Figure 4.5: Working space of the software

But that does not mean any drawn schemes can be implemented into real stations, a simulation was added into the system in order to check for the errors or validity of the drawn scheme for the train stations.

4.2.1 Working principles of the software

Using proposed software, the user is able to design a scheme of the railroad system, basically building the rails using segments(tracks, junctions, lights). Each segment in the software is saved in map format and has an ID and a state(0 or 1), if the track is occupied it receives state of 1, otherwise 0. A state of a junction determines if it is open or closed, 0 or 1 accordingly. Initially every segment is in zero state, meaning all of the tracks are not occupied and junctions are open, the user can change the state of the segment inside the simulation to change the state to the opposite, showcasing the manual mode of the software. Visually tracks will change colors from black to red and junctions will switch the direction(see Graph 4.1).

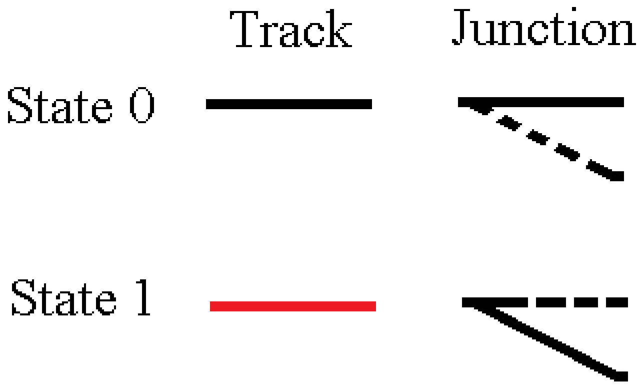

Figure 4.6: States of the segments and their visual representation

The stored data in map format can be formatted into bytes considering the largens of the system it could be either 8,16,32,64 and so on bits, which in turn are sent to the FPGA board.

4.3 FPGA configuration

The second part is calculations(rules), all the calculations in a modern railroad station are calculated using relays. Before building the system engineers create “rules”. Those rules look like a simple equation that consist of an ID of the segment and their states, for example: output = ID1*ID2+ID3, meaning output = 1*0+1(if ID1=1, ID2=0, ID3=1). In this case output = 1, which is an unsafe path. And the system consists of large number of rules, which are divided into smaller parts of the designed railroad system. Configuring the logic for the whole system sometimes might be hard, considering it is much easier to break it down to smaller rules and combining them if necessary. Those simple actions can be represented using logic gates in FPGA.

Let us consider example, where all the tracks are in a straight line. The specification would be:

Tracks: T1, T2, T3 in a straight line.

Train moves from T1 to T3 or backwards.

Output 0 indicates a safe passage; output 1 indicates an unsafe condition.

If any segment (T1, T2, or T3) is occupied, the output should indicate it's unsafe (output 1).

In this case we can use Inversion and AND gates to combine these inverted outputs to determine the safety condition: $S(safe) = T1' * T2' * T3'$

$T1', T2', T3'$ are 1 when their respective tracks are not occupied.

S(safe) will indicate safe passage only if $T1', T2'$ *and* $T3'$ are all 1(clear) and not safe if at least one track is occupied. The logic gate mechanism is fundamental in this straight-line case. Based on the occupancy of the tracks, it is possible to precisely calculate whether or not the train's passage from T1 to T3 is safe by utilizing NOT gates to invert the state of the tracks and an AND gate to combine those inverted outputs. By using this method, the system is guaranteed to accurately reflect the safety condition needed for the train to go on a passage.

And consider second example(Figure 4.2)

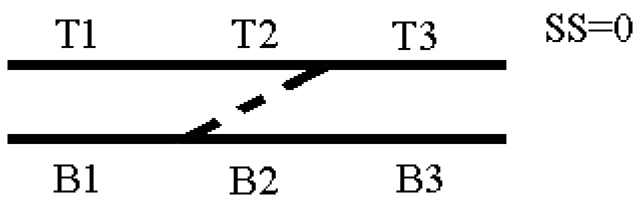

Figure 4.7: Track layout example SS = 0.

In this case switch state(SS) equals to zero, which is open position. If train moves on the top-line or bottom-line same logic applies as in the previous example, given the fact that the rules are divided into two smaller once, but the rules should be universal, so we should include all of the states into one equation, leading us to another logic:

$$Output = SS * (T1' * T2' * T3') + SS' * (B1' * B2' * B3')$$

Here we can see that the state of the switch is a deciding factor and we SS dictates the route of the train.

Consider SS = 1,

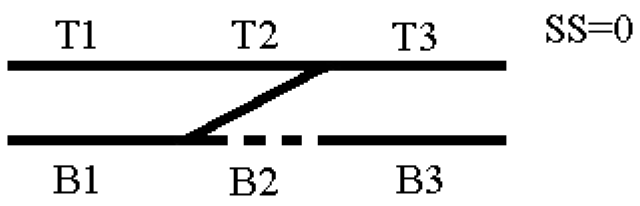

Figure 4.8: Track layout example SS = 1.

When SS = 1, we take the top route, but we do not need to know the state of T1(actually we need for safety, we need to know the states of the nearest track to the junction, but let us imagine that the T1 is longer than it looks, so there is no need). We can not just remove it from equation, since it need to be universal. So, the logic changes to:

$$Output = SS * (T1' * T2' * T3' + T1 * T2' * T3') + SS' * (B1' * B2' * B3')$$

All of this logic can be implemented using logic gates, multiplexers in FPGA, also to create universal logic, there need to be a rule on how to set the logic or machine learning can be initialized to automatically build the logic for the user, but it is for future works. There are existing researches and if asked from Artificial Intelligence(AI) for example ChatGPT it could create such a logic for any layout of the railroad system, thought it would be double checked(for safety futures).

Chapter 5 – Results and Discussion

5.1 Cost

In this chapter the results of the thesis work are discussed. Firstly, the cost of the system is considerably decreased which can be equally distributed in other global projects in a country. To realize the system proposed several devices are needed, such as FPGA, UART(any of the choice) the most expensive cost around 16 dollars and FPGA 1800-2000 dollars(see. Table 2), so to build the system from a start around 3000$ are spent which is not expensive compared to the system used right now. RRI systems used nowadays use relays. The cost of one relay is 24-50$(see. Table 3) and by using dozens of the relay's expenses can come up to 50,000 dollars or more of it is a larger train station. The use of the copper cables and etc. are not included in the calculations. So, the use of FPGA boards would significantly reduce the cost of the system. It is crucial in 3-d rate countries where villages, small cities are undergoing rapid development and turning into cities and towns where new railroads, railroad stations should be built. In the tables below we can clearly see the difference in material cost between FPGA based system and RRI with different components.

Table 2: Cost of the different FPGA boards(found from Internet)

Name	Cost
Basys 3 Artix-7 FPGA Trainer Board: Recommended for Introductory Users	$165.00
Arty A7-100T: Artix-7 FPGA Development Board	$299.00
Nexys A7: FPGA Trainer Board Recommended for ECE Curriculum	$349.00
AMD Artix 7 FPGA AC701	$1,678.00

Table 3: Cost of the different relays(found from Internet)

Name	Cost
SR2 with 2 force guided contacts according to IEC/EN 61810-3	$24/piece
SCHRACK Accessories Miniature Relay PT	$37.00/piece
SR6 with 6 force guided contacts according to IEC/EN 61810-3	$41.00/piece
RREC Railway Q Type Signaling Relay, 24V, DC	$50.00/piece

5.2 Performance

In terms of runtime Java code is slower than C but faster than Python. Because Python is an interpreted language whereas C is compiled, Python is comparatively quite sluggish. All at once, the compiler converts the C code to machine code. Conversely, each line of code must be read, understood, and executed by the interpreter in order to change the machine state, which adds a significant amount of overhead. The Java compiler, or JIT, is situated halfway between Python and C. The code is interpreted when run for the first time. On the other hand, code that is run a lot is compiled into computer code in real time and is then used for subsequent executions. That is why in this paper a Java code was chosen, since it can either simulate and give relatively fast output. For example, the runtime of the Dijkstra algorithm in C for 304 locations was 4.952 milliseconds, whereas the execution time of same algorithm’s implementation, but in FPGA board, was 0.544 milliseconds which is roughly 10 times faster than in C to add, the outputs are similar. The use of FPGA's parallel data processing greatly improves the speed of the algorithm.[23]

All the data that is used further for FPGA processing is stored in java application. There are several ways to store the information, but in this study “map” is used and this data is used not only to ensure if the path is safe or not, it is also used to show the ideal path to the finishing point. As it can be seen in a previous Figure 5.1, there is a red line which shows the pathing. The simulation shows the most ideal path to the station(end point).

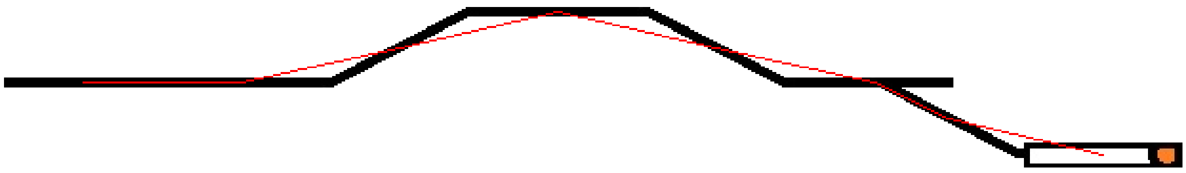

Figure 5.1: Track example1.

This is the visual part, as it can be seen it is robust, but those little bugs can be fixed in the future. Besides visually showing the path by highlighting it, software also gives an output:

```
Compo: track_1
dis: 536.3478348982123
dis: 454.3974031615938
dis: 415.0
dis: 332.48759375351136
dis: 251.7339865810733
dis: 166.97305171793442
dis: 83.0
At destination.
[{S=[finish_8, sidedowntrack_7, track_6, sidedowntrack_5, track_4, sideuptrack_3, track_2, track_1]}]
```

Figure 5.2: Output of the software for example1.

Here we can see the starting point(in most of the cases it would be track_1) and destination from every component to the ending point. Also a list from right to left the order of segments in the path.

5.3 Reliability

Better reliability is the next step of the innovation in the railroad industry. While the software and hardware are always not secure to use, there are a lot of security based addons and plugins for java applications. But reliability in case of errors in relays and human errors are excluded by the use of the system proposed. Proposed system automates the station to the point where no human interaction is need (only observing in case of outage and etc.).

There are several papers that explain and implement the reliability and security of the FPGA based systems. For example, there are BER(backward error recovery) based reliability methods. For SRAM-based FPGAs a straightforward bit flip might result in an aberrant calculation is an essential problem . In one of the papers authors introduced a novel approach to fault endurance that utilizes technology BER for protecting and rectify systems against the appearance of temporary defects. To guarantee hardware checkpoint, the limited dynamic reconfiguration provided by Xilinx Virtex FPGA was employed, and restoration upon fault identification should be employed. This approach has a number of benefits, including the fact that it doesn't require any internal system hardware module modifications, isn't dependent on redundant hardware assets and when implemented in a system, has a fixed area overhead ratio.[22]

Most of the suppliers of electronic companies employ very similar techniques to guarantee high dependability and facilitate the easy integration of safety mechanisms. The dependability of internal info transmission is increased by using ECC (Error Correction Code) on information and address buses.

Memories are subjected to the same methodology in order to reduce their mistakes. Two cores operating in lock-step mode are frequently integrated on a single chip in order to obtain high error coverage and quick error detection of processing units.

There is also a solution that could improve the security of the system. This solution requires the use of several FPGA boards with the same code and for same simulation to run, which is also called Threat monitoring and response systems(TMR). All of the system would be identical to each other and system itself will contain several outputs that are compared to each other. In case all of the outputs given by the FPGA board are equal or identical to each other they will pass, otherwise it will give an error. There are two possible ways, either “2 out of 3” TMR or “3 out of 3” TMR, where 2 out of 3 FPGA’s give same answer is enough or all of them should give the same output. For security to be on high level, the second choice is more reliable. In this case we can ensure that reliability of the system can be improved by the use of more than one board.

Chapter 6 - Conclusion

In this study an automated railway simulation was presented. With the usage of technologies such as: FPGA, relay and several sensors, the provided system can be automated to the point where the cost of the system will be reduces, the speed and effectiveness of the system improved, a real-time monitoring of the rail and train conditions streamed to the headquarters or nearby stations. Overall, the system can be realized as shown in methodology and overview of the system.

Bibliography

1. O. Cataldi and R. Alexander, "Train control for light rail systems on shared tracks", Railroad Conference 2001. Proceedings of the 2001 IEEE/ASME Joint, pp. 9-16, 17–19 April 2001.

2. Rodrigue, J-P (2015) “The Structuring Effects of Rail Terminals”, in C. Comtois and B.P.Y. Loo (eds) Sustainable Railway Futures: Issues and Challenges, Transport and Mobility series, London: Ashgate, pp. 23-37.
3. Rail Terminals, Dr. Jean-Paul Rodrigue and Dr. Brian Slack, online https://transportgeography.org/contents/chapter6/rail-terminals/
4. CS. Chang, C. S. Chua, H. B. Quek, X. Y. Xu and S. L. Ho, "Development of train movement simulator for analysis and optimization of railway signaling systems", Developments in Mass Transit Systems 1998. International Conference on (Conf. Publ. No. 453), pp. 243-248, 20–23 April 1998.

5. W. A. MacDonald, "Train controller interface design: factors influencing mental workload", Human Interfaces in Control Rooms Cockpits and Command Centres 1999. International Conference on, pp. 31-36, 21–23 June 1999.

6. R. Lipsett, C. Schaefer and C. Ussery, "VHDL: Hardware Description and Design" in , Dordrecht:Kluwer Academic Publishers, 1989.

7. S. Sjoholm and L. Lindh, "VHDL for Designers" in , Prentice Hall Europe, 1999.

8. S. D. Brown, R. J. Francis, J. Rose and Z. G. Vranesics, Field-Programmable Gate Arrays, Kluwer Academic Publishers, 1996.

9. L. H. Tsoukalas and R. E. Uhrig, Fuzzy and Neural Approaches in Engineering, Wiley-Interscience, 1997.

10. W. Chia, L. J. Jeganathan, M. B. Ibne-Reaz and F. Mohd-Yasin, "The FPGA approach in the design and development of an automated rail transit system controller," 2004 IEEE International Conference on Semiconductor Electronics, Kuala Lumpur, Malaysia, 2004, pp. 5 pp.-, doi: 10.1109/SMELEC.2004.1620897

11. Hernández, Álvaro & Pérez-Rubio, Carmen & García, etc. (2012). FPGA-Based

Track Circuit for Railways Using Transmission Encoding. Intelligent Transportation Systems, IEEE Transactions on. 13. 437-448. 10.1109/TITS.2011.2170976.

12. K. Rajan, S. Chouhan, 2017. An Efficient Implementation of Zigbee Transmitter onFPGA Using Verilog. Volume 4, Issue 3, July 2017, ISSN: 2348-8565 (Online)
13. Lavrukhin O., Schevcenko V., Kulova D., Kim K., Vernyhora R., Kyman A., & Shulika O. (2020). FORMING AN AUTOMATED TECHNOLOGY TO ACTIVELY MONITOR THE TRANSPORTATION OF DANGEROUS CAR GOES BY RAILROAD. East European Journal of Advanced Technologies, 3 (3-105), 78-85.
14. Butko, Tetiana and Prokhorov, Viktor and Kolisnyk, Alina and Parkhomenko, Larysa,Devising an Automated Technology to Organize the Railroad Transportation of Containers for Intermodal Deliveries Based on the Theory of Point Processes (February 24, 2020). Eastern-European Journal of Enterprise Technologies, 1(3 (103), 6-12. doi: 10.15587/1729-4061.2020.195071.
15. Ramezani, Reza & Ghavidel, Abolfazl & Sedaghat, Yasser, 2021. "Exact and efficient reliability and performance optimization of synchronous task graphs,"Reliability Engineering and System Safety, Elsevier, vol. 205(C).
16. Hoque, Khaza Anuarul & Ait Mohamed, Otmane & Savaria, Yvon, 2019. "Dependability modeling and optimization of triple modular redundancy partitioning forSRAM-based FPGAs," Reliability Engineering and System Safety, Elsevier, vol. 182(C),pages 107-119.
17. J. J. García, Á. Hernández, J. Ureña and E. García, "FPGA-Based Architecture for a Multisensory Barrier to Enhance Railway Safety," in IEEE Transactions on Instrumentation and Measurement, vol. 65, no. 6, pp. 1352-1363, June 2016, doi: 10.1109/TIM.2016.2534238.
18. R. Dobias and H. Kubatova, "FPGA based design of the railway's

interlocking equipments," Euromicro Symposium on Digital System Design, 2004. DSD 2004., Rennes, France, 2004, pp. 467-473, doi: 10.1109/DSD.2004.1333312.

19. P. Eh Kan, A. H. Abdul Halim, H. Othman, M. S. Mustafa, and C.Y. Lam, "An FPGA-Based Controller for High-Speed Train System" International Journal of Computer and Electrical Engineering, Vol. 4, No. 5, October 2012
20. Raj, A.A.B. (2017). FPGA-Based Embedded System Developer's Guide (1st ed.). CRC Press. https://doi.org/10.1201/9781315156200
21. AMD "The Artix™ 7 FPGA AC701 Evaluation Kit features the leading system performance per watt Artix 7 family to get you quickly prototyping for your cost sensitive applications." https://www.xilinx.com/products/boards-and-kits/ek-a7-ac701-g.html
22. Open Rails "Totally free train simulator that supports the world's largest range of digital content." https://www.openrails.org
23. AnyLogic Company "Rail simulation with AnyLogic delivers time and cost savings across the whole scope of rail logistics." https://www.anylogic.com
24. Fouad Sahraoui, Fakhreddine Ghaffari, Mohamed El Amine Benkhelifa, Bertrand Granado. An efficient BER-Based Reliability Method For SRAM-based FPGA. 7th IEEE International Design and Test Symposium IDT'2012, Dec 2012, Doha, Qatar. 6 p. ffhal-00854818f
25. Boyi Xu, Zhenyan Zhang and Chenxiao Zhu "Dijkstra algorithm on FPGA to find the shortest distance and route between two stations among the railway network of China", accessed 5 May 2024 https://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2018/cz382_zz488_bx64/cz382_zz488_bx64/cz382_zz488_bx64/index.html