MAGIC

Alexander Whitfield [1], Tom Wildoer [2], Nathan Fiddes [3], Vinay Panicker [4], Finn Pickering [5], and Miles Nathan [6]

[1,2,4,6] Department of Robotics and Mechatronics Engineering

[3,5] Department of Electrical and Computer Systems Engineering

Abstract

The Magnetic Automated Gantry for Intelligent Chess (MAGIC) project is a robotic chess board that lets users play on a physical board, either against a chess AI or someone online. Using a 2-axis gantry and Hall effect sensors, the board autonomously moves pieces and tracks gameplay. The board is fully integrated, with all subsystems (mechanical, electrical and software) working together simultaneously. The following paper outlines the steps taken that resulted in the success of this integration, providing proof of concept for a continuously sensing boardtop. Unlike previous versions of chess-playing robots that rely on large and intrusive mechanics, the MAGIC project incorporates a fully functional system beneath the board that maximises space efficiency. The board paves a path for a new evolution of long distance communication and interactive gameplay, while still maintaining the classic board game experience.

1 Introduction

MAGIC is an interactive, automated chess system designed to foster meaningful, in-person and remote social interaction through a tangible, physical board. Users can play against online opponents, friends, or a chess AI, creating a shared, immersive experience that contrasts with the impersonal nature of screen-based communication. This project is a response to the growing sense of disconnection in an increasingly digital world. While modern platforms enable instant connectivity, they often fall short in nurturing intentional, emotional interactions. MAGIC seeks to reintroduce the warmth and mindfulness of physical presence through the familiar, universally accessible medium of chess. The goal is to create a shared interactive space that encourages thoughtful engagement and casual connection, even asynchronously. This can be especially meaningful for individuals who are most vulnerable to isolation, such as older adults or people with physical disabilities, by offering them an intuitive and inclusive way to stay connected.

Compared to commercial robotic boards, MAGIC aims to provide more precise tracking, smoother piece movement, and greater flexibility for customisation. The platform also opens the door to future enhancements like voice commands, adaptive play modes, and other accessibility-focused features. Ultimately, MAGIC explores how interactive physical interfaces can counteract the emotional distance of virtual life, one chess move at a time.

2 Methodology - Mechanical

To enable autonomous piece movement, a mechanical subsystem was designed. The motion system, or gantry, forms the core of the MAGIC platform and must operate with high reliability, durability, and precision to consistently replicate the movements of an external player. The gantry requirements were complicated by its need for portability.

The gantry, which consisted predominantly of linear rails, bearings and stepper motors, took many forms over the duration of the project. In initial sketches (Figure 1a) it was envisioned that the belt could be manoeuvred by a complex idler system in the hope to maximise the usable space to volume ratio (USV). The USV in initial concept drawings was optimised by using the technique of belt management along with nesting the joining points of rail mounts into the linear rails (carbon spars; Figure 1b).

This version (V1), however, suffered from its own complexity. The belt system was too complex to work consistently and the rail mounts were not stiff enough to prevent racking (Collins, D. n.d.). Moreover, versions one and two opted for a single linear bearing on the longer axis, meaning they could not be pretensioned (also a contributor to racking).

Version two (V2) aimed to solve the problems with stiffness by moving to a more ‘foundational’ approach. The complex belt tensioning mechanism was removed and the press-fit nesting mounts seen in the first iteration were swapped for more conventional pinch bolts. In a smaller, but still meaningful way - the racking persisted.

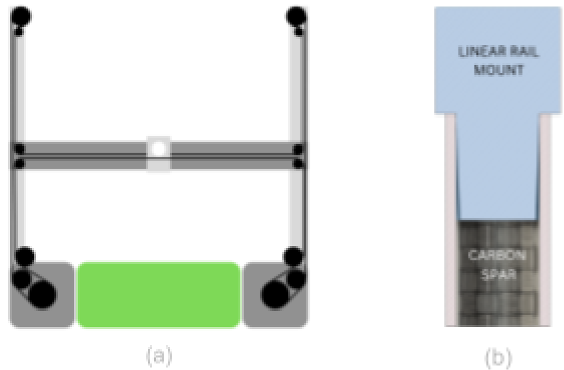


Figure 1a: Initial Concept Gantry (top down). Figure 1b: Pressfit Rail Mount

Version three (V3) was the first iteration to tackle the problem at the source. The carbon spars were replaced with precision ground steel rods and USV was sacrificed by implementing two preloaded bearings on each side of the longer axis with contact patch shown in blue (Figure 2). The effective width of the carriage doubled to accommodate the bearings, however the racking was now tunable and could be eliminated entirely. This is the case because the distance of the contact patch to the normal (red hatched line) can be adjusted by rotating the bearing off-axis and bringing the patch closer or further tuning the effective diameter of the bearings relative to the rail. The inherent play in each linear bearing could be removed by intentionally introducing this artificial racking into each carriage. This would counteract the looseness of the bearings and minimise overall system racking. V3 was the first entirely mechanically functional version of the gantry and coincided with the start of semester two teaching period.

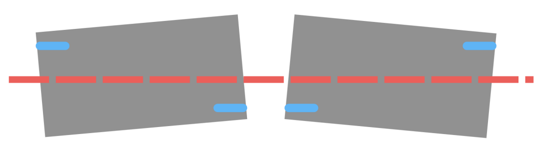


Figure 2: Amplified example of 'preload' on linear bearings'.

From this point, attention was focused on refinement. Version four (V4) included the standoffs for correct placement of the discrete Hall effect array (see section 3 'Methodology - Electrical') and also had arrangements for complete housing of the board (Figure 3b).

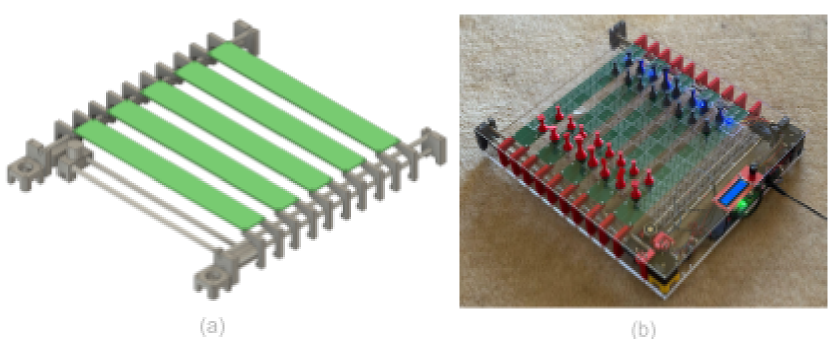


Figure 3a: Isometric view of version 4 of the gantry. Figure 3b: Image of final Gantry System.

Finally versions five and six (V5 and V6) tweaked geometry and playable area to improve USV whilst also properly implementing 'quality of life' upgrades like cable management, UI integration and noting of the chess board's file

and rank. Most notably however these versions were the first two to experiment with electromagnetic control.
The electromagnet in use for this project is a 12V 0.25A electromagnet with a rated holding force of 25N. It was initially assumed that a magnet would be too weak for the intended application. However, preliminary drag testing with a 3mm acrylic layer separating the magnet and the pieces revealed that the neodymium rare-earth magnets were too strong. Thus began a testing phase - 10 of both neodymium and ferrite magnets were purchased in sizes ranging from 5✕3 mm to 10✕5 mm. It was eventually determined, through cyclic drag testing, that the suitable choice was two ferrite magnets (iron oxide) of dimensions 5✕3 mm stacked on top of each other. This yielded the best characteristics for vertical pulling force and radial inertness (i.e. no attraction to other pieces). Integrating all the subsystems and refinement discussed above, the final product in its current form is displayed in Figure 3b.

3 Methodology - Electrical

The electrical team's primary goals were to design and implement an accurate, continuous detection system for the chess pieces and design the required hardware for computation. The system needed to detect when a piece is lifted and placed, as well as identify its origin and destination squares. Other electrical components were used for control of systems, such as motor movement or onboard chess bot computation.
After researching a variety of sensing technologies (see section 3.1), the Hall effect sensors demonstrated superior reliability and sensitivity, detecting the magnets from up to 10mm away. Therefore, it was selected for further research. Additional testing demonstrated that to have true continuous detection, the sensors would need to be placed at a much higher density and thus become less cost effective. As a result, the team focused on developing discrete detection boards with one Hall effect sensor per square, to ensure that the final integration of the MAGIC board was possible. Figure 4a and 4b show the discrete tiles used in the final version with integrated software displaying magnet placement. Once this was completed, the team continued to investigate the continuous board top as part of the MAGIC projects research (see section 3.3 for more information).

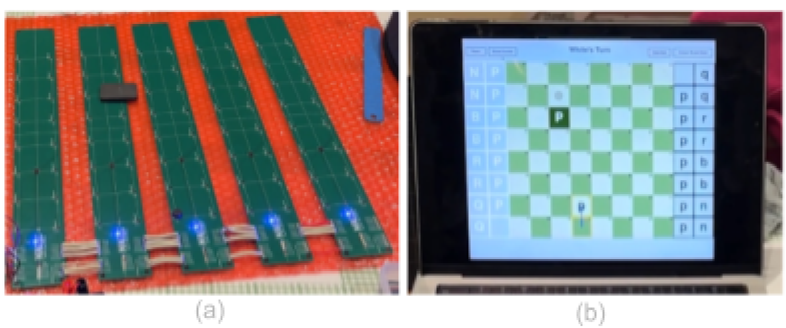


Figure 4a: Five interconnected Hall Effect sensor PCBs. Figure 4b: Corresponding software GUI.

3.1 Sensing Technologies Research

Identifying a cost-effective and minimally intrusive method to detect the presence and movement of pieces was a significant challenge. The system requires a sensor that is accurate, reliable, cost-effective, low profile and minimally wired. This resulted in investigating a variety of sensing methods, including reed switches, radio-frequency identification (RFID), near-field communication (NFC), Hall effect sensors, inductive coils, and touch foil sensing (Cook, 2023; Blackler, 2024). A formal research investigation was also conducted to test the capabilities and effectiveness of inductive coils and Hall effect sensors, see Figure 5 for the experimental circuits used.
Reed switches were inexpensive but did not have the flexibility required for continuous detection. Inductive coils had interesting results but lacked repeatable accuracy and the ability for continuous piece detection through triangulation algorithms. Their discrete detection capability was also investigated but proved inefficient and inaccurate. Capacitive touch foils were dismissed due to the high cost for detecting 40 pieces, while RFID and NFC technology were also not pursued. This was due to the fact that these technologies are able to detect sensors well over the 35mm square length (Cao et al., 2019) and therefore multiple sensors outside of the detection zone could be triggered resulting in incorrect position calculations.
Following this research and experimentation, Hall effect sensors proved to be the best option. They offered the best balance of performance, cost and implementation complexity, while also allowing continuous piece detection

through triangulation algorithms. Using magnets embedded in each piece, the Hall effect sensors could detect changes in the local magnetic field and provide analogue outputs that correspond to the pieces' proximity (Paun, Sallese, Kayal, 2013).

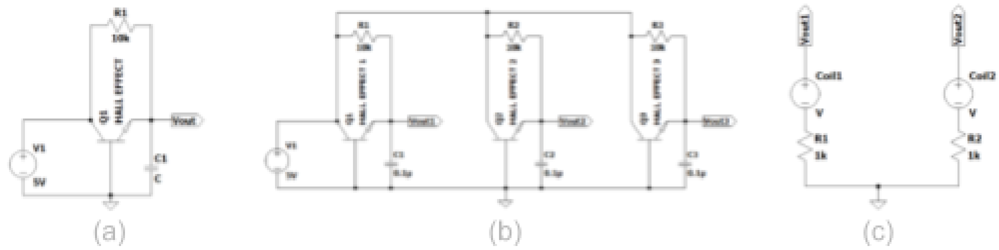


Figure 5a: Single hall effect sensor output circuit. Figure 5b: Mutli-sensor array. Figure 5c: Coil/load driver stage circuits used to test hall effect and coil sensing capabilities.

3.2 PCB Design

A printed circuit board (PCB) was designed with the intention of housing the processing unit of the system. This board housed a Raspberry Pi CM5 and received inputs from all sensors, including the Hall effects and sensors in the gantry system. It also drove the gantry motors and controlled the option menu display with the software written for the Raspberry Pi. A schematic diagram of the motor driver circuit and Raspberry Pi connections is shown in Figure 6.

During testing, several issues were encountered with the board, and multiple redesigns were required before achieving a final functional version. The first version of the board (V1) was mostly a proof-of-concept to check for basic functionality. Unfortunately, due to some poorly placed components V1 demonstrated limited performance. The Raspberry Pi was unable to correctly output voltage readings on the GPIO pins, which was later determined to be caused by a lack of bulk capacitance near the power source. A ground plane was also added in subsequent versions for greater stability in voltages.

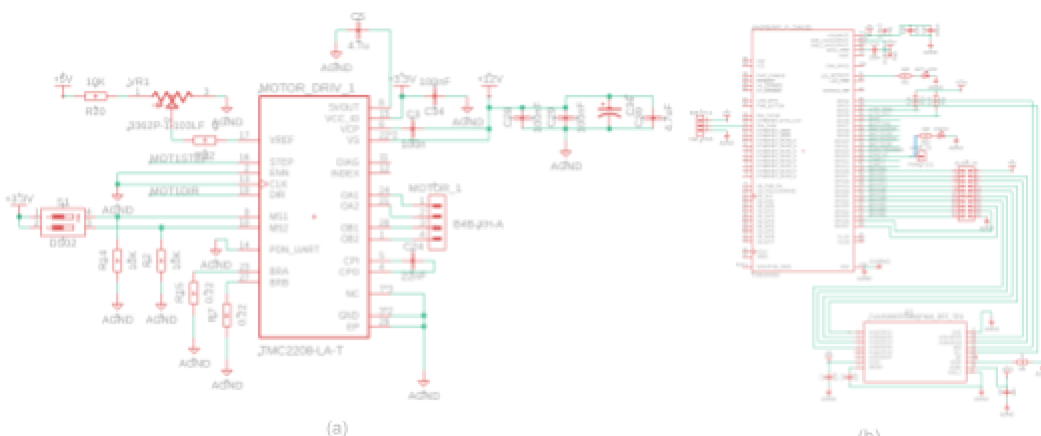


Figure 6a: Motor circuit schematic on control board with motor driver. Figure 6b: Raspberry Pi connections, including ADC and labelled external connections.

The second version of the board (V2) yielded much better results. In V2, more development features were added to easily troubleshoot issues, such as LEDs, pads connected to some GPIO outputs and switches for motor control. The GPIO pins in V2 worked as expected, giving the desired output when measured with a multimeter and turning on components as required. However, while one motor worked well when connected to the PCB, the other would turn with very low torque or not at all. The problem was identified as a 'brown out' issue with the motor driver and the capacitors that worked in conjunction with it, as they were too far apart for it to work as intended.

The third version of the board (V3) was still not quite functional, as the electromagnet would not work in conjunction with the motors, but this was fixed in the fourth and final version (V4), which functions as required. The 4 versions of the manufactured PCBs are shown in Figure 7 below.

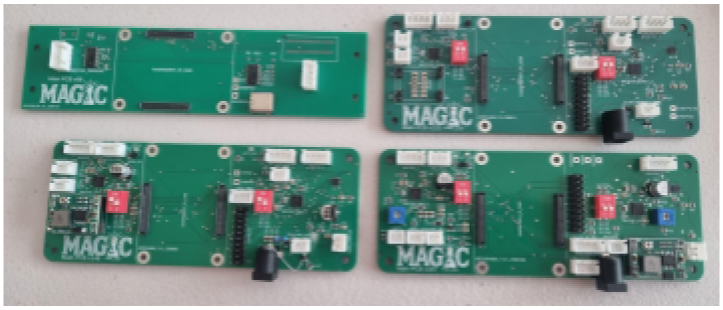


Figure 7: PCB versions left to right top to bottom: V1, V2, V3, V4.

To ensure reliable sensor readings, a multiplexer system was implemented with an analogue to digital converter (ADC) device. The ADC converted multiplexer outputs into I2C signals that are read by the Raspberry Pi to pinpoint which pieces have been moved and the square they moved to. Each square was polled approximately every 100ms to track piece movement and verify piece locations.

The sensor system PCBs were mounted along the underside of the playing surface, with one Hall effect sensor under each square, except for the outer graveyard squares. The PCBs were made to be identical to each other to save costs on ordering, each one sensing a 2×8 square area on the board, with the 5 boards covering 80 of the 96 total chess board squares. The external graveyard squares were intentionally excluded, as the pieces there could be inferred without the need for explicit sensing.

An LCD screen was integrated into the design as a user interface, allowing players to change settings and select a gamemode for playing on the board. A rotary button on the side of the screen controlled a menu and could be used to select the desired settings. This screen was controlled via the GPIO pins on the Raspberry Pi that can handle I2C connections.

3.3 Continuous Detection Research

As introduced earlier, the electrical team initially explored a continuous piece-tracking system using Hall effect sensors. However, due to the limited sensing range of affordable sensors and the resulting need for a highly dense circuit layout, this approach proved impractical. The team then continued with a discrete detection board to test the software and mechanical subsystems, which were further ahead in development. In parallel, work began on a smaller prototype continuous board, designed to be a cost-effective proof of concept to investigate whether continuous tracking could be achieved within a constrained budget.

Initial prototyping on Veroboard using through-hole Hall-effect sensors, illustrated in Figure 8a and 8b, proved cumbersome and resulted in inconsistent measurements, including unexpected sensor responses. These experimental limitations motivated a transition to a simulation-based methodology for characterising magnetic-field interactions. The sensor selection was therefore limited to Texas Instruments Hall effect sensors, allowing the use of the TI Magnetic Sense Simulator (TIMSS) (Figure 8c) for virtual testing and validation.

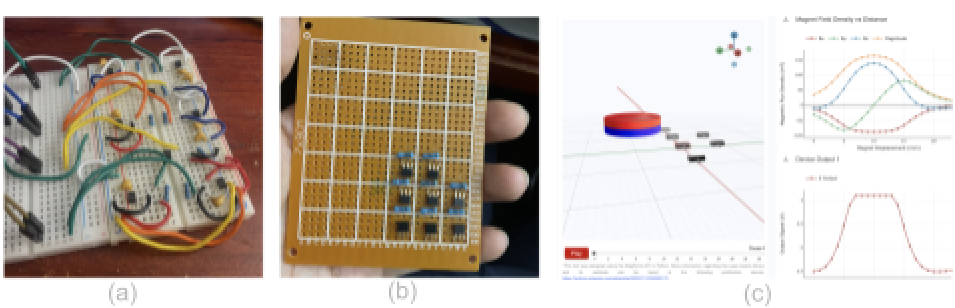


Figure 8a: Prototyping with breadboard. Figure 8b: Veroboard prototype. Figure 8c: TIMSS simulation results

The simulation-based approach offered two main advantages: it reduced prototyping costs by enabling evaluation of multiple layout configurations virtually, and it provided deeper insight into the non-linear magnetic-field behaviour of the sensors. Through these simulations, the limitations of the initial design strategy became evident. The design had aimed to estimate magnet position via triangulation using a sparsely distributed array of Hall effect sensors. Several geometric layouts – such as triangular and parallelogram arrangements – were assessed to balance positional accuracy with sensor economy. However, simulation results revealed two fundamental constraints that rendered this low-resolution approach infeasible.

The main issue was that the magnetic field intensity exhibited exponential decay, causing sensor readings to degrade sharply with even small displacements of the magnet. Consequently, when the magnet moved slightly, adjacent sensors received insufficient signal strength to maintain reliable triangulation with a sparse sensor arrangement. This behaviour is illustrated in Figure 9, which combines a schematic of the sparse sensor arrangement with a magnet positioned over one sensor (Figure 9a) and the corresponding sensor voltage response (Figure 9b) as the magnet sweeps across the array. The plot shows the exponential drop-off in sensor output as the magnet moves, demonstrating why small magnet displacements produce minimal detectable signals at neighbouring sensors.

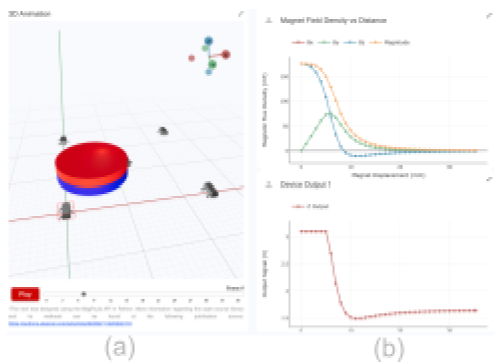


Figure 9a: TIMSS Screenshot: Sensor in red box. Figure 9b: Exponential drop shown in (Device Output 1).

These findings necessitated a shift from a sparse sensor layout to a high density layout. Iterative simulation analysis led to the selection of an 8✕8 mm square sensor grid combined with a 12✕3 mm cylindrical neodymium magnet (N42 grade). In this configuration, each magnet influenced only the 3–4 sensors directly beneath it, with negligible effect on neighbouring sensors, as illustrated in Figure 10 sensors S1-4 receive a high response whereas S5 and S6 are not influenced. Meaning the magnet's position can be independently resolved from its local sensor cluster.

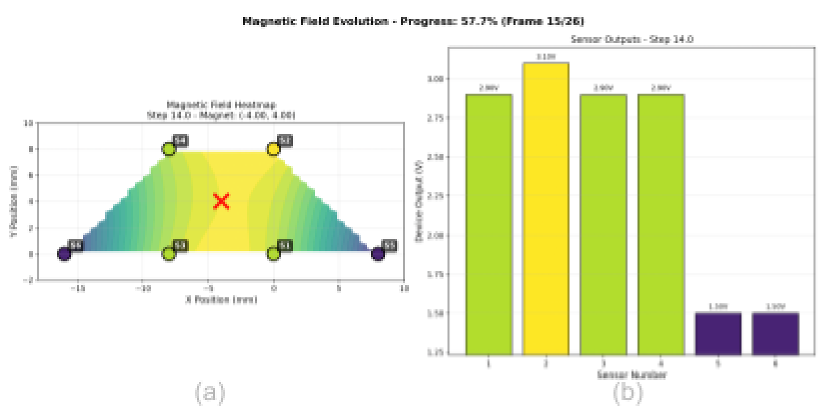
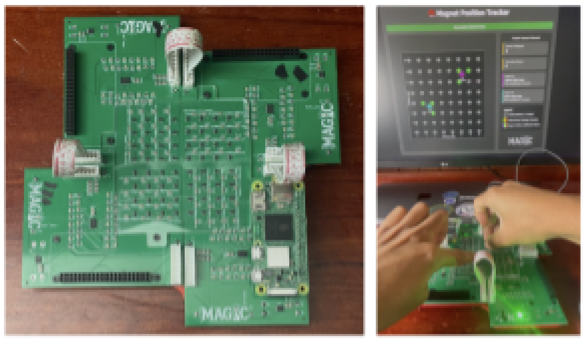


Figure 10a: The magnetic field heatmap for a magnet positioned at (-4,4). Figure 10b: The corresponding Hall effect sensor outputs for that magnet position.

The position estimation process began with voltage thresholding to detect active sensors. Connected-component labelling was then applied to group the neighbouring active sensors into clusters. Within each cluster, the magnet's position was calculated using a weighted centre-of-mass method, where sensor voltage values acted as weighting factors to determine precise coordinates in millimetres. To minimise false detections caused by overlapping magnetic fields, a minimum separation of 12mm was enforced. Clusters closer than this threshold were merged, and their combined centre-of-mass recalculated. The resulting dimension-agnostic clustering method is inherently scalable, allowing the 8×8 sensor grid to serve as a proof of concept for larger arrays that could be used in a full-size chess board. A visual representation of the magnet-clustering interface is shown in Figure 11b.



(a) (b)

Figure 11a: Modular PCB. Figure 11b: Clustering UI.

Additional exploration was done into a machine learning approach, to further improve the sensing resolution. Drawing inspiration from (Vizel et al., 2025), who demonstrated successful magnetic field mapping using sparse sensor arrays enhanced with fully connected neural networks, an investigation was undertaken to develop a similar machine learning-based approach for this system. However, preliminary investigations revealed significant practical limitations. The trained model exhibited prohibitively long inference times unsuitable for real-time tracking, and generating high-quality training data proved challenging due to inadequate access to *COMSOL Multiphysics* for accurate magnetic field simulation. Consequently, this approach was not pursued further.

3.4 PCB Design and Implementation

The minimum PCB order quantity was five boards. To take advantage of this, the team designed a compact 4×4 cm sensor array that could be replicated across multiple boards. By combining four of the boards in a 2×2 configuration, a sensing area of 8×8 cm would be created (see Figure 11a). This modular approach reduced manufacturing costs while allowing the design to be easily scaled up in future iterations.

Each module shared signal lines for power - SDA, SCL, and multiplexer select - connected by pin headers for quick assembly. The multiplexer-enabled pins were hardwired to the main controller board with the Raspberry Pi and assigned their own ADC port. Software logic selected the active multiplexer as needed.

This modular approach performed better than expected in early testing, demonstrating that a scalable, cost-effective continuous sensing solution could be achieved. Although a full-size continuous board remains outside the financial scope of the project, the modular prototype successfully validated the concept and provided a foundation for future development.

4 Methodology - Software

As the chess systems' gantry is constrained to piece manipulation in the X-Y plane, movements are restricted to two dimensions. Consequently, chess moves that would conventionally utilise vertical displacement - such as moving a knight - must instead be executed through the manipulation of adjacent pieces to establish a viable path to target. This is termed as the "crowd control problem."

4.1 Naive Continuous Algorithm

The "crowd control problem" can be solved when piece diameters are ≤ 30mm. In this case all chess movements (straight, diagonal and L-shape) remain possible in the absence of vertical displacement,, even when the entire board or path to target is full. This is done by creating space for the piece to move via a fixed set of minute movements on the neighbouring pieces within a smaller grid on the board. Figure 12 illustrates the implementation of this approach for an L movement, demonstrating how pieces can be strategically repositioned within the smaller 2✕3 grid to allow for the movement.

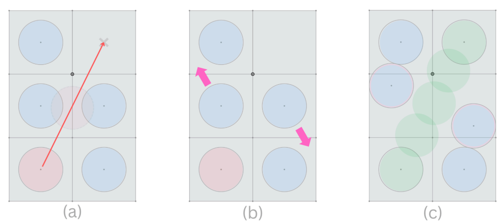


Figure 12: Naive L movement sequence with 25mm pieces.

In Figure 12a, it is observed how a 25mm piece is unable to perform the L move, as such adjacent pieces are displaced into the corners, indicated by the directional arrows in Figure 12b, thereby creating sufficient clearance for the target piece to traverse diagonally shown in Figure 12c.

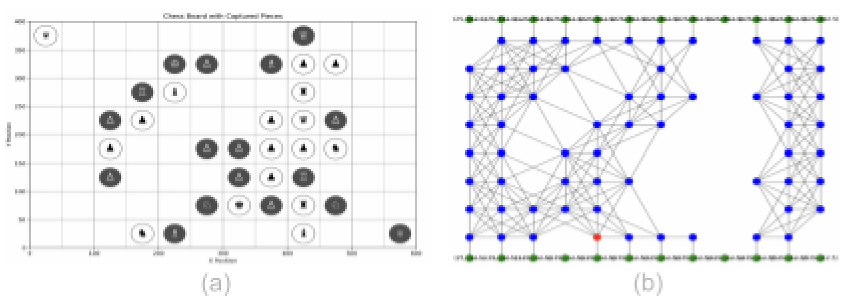


Figure 13a: Board view. Figure 13b: graph view from e1 bishop.

Knowing that complex movements - such as straight, diagonal and L-movements - are possible if isolated to a smaller grid on the board, a simple graph can be created that directly incorporates these edge types and assigns higher weights to complex L and diagonal edges. Later, when a path is found to have taken an L-edge, the mechanical gantry performs a sequence of actions (i.e. Figure 12b) before moving to the next edge. As this is a weighted graph, running Dijkstra's algorithm (Dijkstra, 1959) yields the most efficient path, ensuring the system avoids complex manoeuvres (such as the L-movement sequence) when simpler sequential movements (e.g., orthogonal combinations) suffice. Figure 13b illustrates the graph created for a given board (Figure 13a).

To evaluate the performance of the algorithm, a test script was developed to simulate *n* chess games. This script was employed throughout the development process to check that every move in a game was executable on the movement graph to be used by the physical chess board. Upon completion of the algorithm, it successfully simulated 50,000 game moves without failure - meaning all moves would be executable on the 2D plane by the MAGIC gantry. Furthermore, the diameters of the game pieces were reduced from 30 to 22 mm to enable diagonal movement, thereby facilitating more efficient pathfinding and enhancing the overall playing experience.

4.2 GUI

Before having a physical chess board and gantry system to test on, a working GUI was essential for debugging and writing code. The GUI shows the entire chess board as well as the graveyard and all the pieces in it. It illustrates the legal moves when a piece is selected and the path the crowd control algorithm uses to move pieces. The GUI also has several buttons:, one button toggles to show the mouse coordinates corresponding to the real world coordinates on the full size physical board. Other buttons turn off path showing, reset the board and print the board state for later tests.

4.3 API Integrations

The Lichess API integration enables remote gameplay between other physical MAGIC boards and online opponents, addressing the project's core goal of fostering meaningful long-distance interaction. Through the Berserk Python library and authenticated API access, users can engage in live online matches against any player on the Lichess platform, with moves automatically transmitted and physically executed on the board. When playing with a MAGIC chess board, this can also be extended to play one MAGIC chess board against another.

4.4 Game Modes & Features

The chess robot includes a range of game modes made accessible through an onboard LCD interface. Users can easily navigate the menus to select between several modes:

-Player vs Robot: Users can play against the Stockfish chess engine, with options to choose side (black or white) and difficulty.

- Robot vs Robot: The Stockfish engine plays against itself, allowing users to observe AI gameplay or experiment by setting different difficulty levels for each “robot.”
- Online Play: Through Lichess integration, users can play live games against online opponents. If two users had their own MAGIC board they could play remotely using the physical board in real time.
- Archived Game Playback: Users can upload and replay any historical or personal chess game using PGN files. The gantry autonomously makes each move on the board, enabling analysis or educational demonstrations.

5 Discussion

5.1 Final implementation

The results demonstrate that the magnetic gantry system achieved reliable, repeatable piece manipulation and detection through the integration of precise mechanics, Hall effect sensing, and intelligent control. The discrete Hall effect sensor array consistently identified magnet positions across the board, maintaining detection up to 7mm above the 3mm thick acrylic playing surface. This confirmed the effectiveness of low-cost magnetic sensing for discrete position tracking and validated simulation findings regarding voltage-detection thresholding and behaviour. Research into continuous sensing provided further insight into magnetic fields and signal overlap, establishing the practical limits of sensor spacing and confirming the discrete approach as the most viable within the project’s scope. Challenges such as racking, electrical noise, and electromagnet alignment were progressively mitigated through design iterations, resulting in a stable and smooth final system.

Despite the successful implementation of the chess board and chess functions, the final board missed the initial expectations of having a broader use across many different applications mainly due to the lack of full scale integration of continuous piece detection. Since robust examples of each aspect were made separately it shows the initial expectations are possible and require only a matter of time and money to integrate.

Overall, the findings show that an accessible, modular gantry architecture can achieve dependable performance without complex or expensive sensing methods, demonstrating a strong foundation for future refinement toward continuous tracking and broader applications in automation.

5.2 Retrospect

In retrospect, task allocation within the team was effective; however, the team's lack of experience in certain fields meant that time could have been distributed more strategically. A greater focus on the electrical subsystem in preliminary stages would have allowed earlier research into PCB design, reducing costs and integration delays later on.

A minimum viable product approach could have also accelerated the design and development of the board, prioritising functionality over refinement early in the project. For instance, the mechanical team could have developed a basic working gantry sooner, freeing resources for PCB and software development. Similarly, the electrical team should have transitioned earlier to discrete piece detection to establish a functioning MAGIC board, enabling parallel integration while continuing research on continuous detection.

5.3 Documentation and Future Plans

Throughout development, the software team utilised MkDocs to detail functions, dependencies and code structure to support maintainability and future collaboration. With this documentation, the team aims to release the project as an open-source repository, allowing other developers and hobbyists to replicate, expand or adapt the system for their own chess automation projects.

5.4 Future Improvements

If there was more time to work on the project, several improvements could be made to increase the system's reliability, functionality, and ease of use. One major enhancement would be fully integrating continuous movement across the entire board to achieve smoother and more precise piece handling. The gantry would also benefit from a more robust cable management system for the electromagnet, as the current floating cable can become tangled with the belt or snagged on other components. Expanding the system to support a wider variety of board and strategy games beyond chess would further increase its versatility. Other potential improvements include refining the path-planning algorithms for faster and more efficient movement, enhancing the GUI for clearer feedback and easier control, and improving hardware performance through better sensor calibration and noise reduction. Finally, adding wireless connectivity or a web-based interface would make the system far more user-friendly and accessible for remote operation.

6 Conclusion

This project aimed to build a fully automated chess board as a proof of concept, demonstrating how objects on a 2D plane can be moved automatically under electronic control. It integrated electrical, software and mechanical components, creating a cohesive system to play chess. The final design has a variety of different applications, such as enabling two artificial intelligences to compete, or allowing a human player to face either an AI or another person remotely via the internet. By extending this concept, the system could be adapted for other board games or any 2D environment involving object manipulation. By intertwining online interaction and physical movement, MAGIC allows deeper connections between physical reality and the virtual world.

References

1. Blackler, E. (2024). *RFID vs. NFC: What are the 5 key differences?* Nortech.
2. Cao, Z., Chen, P., Ma, Z., Li, S., Gao, X., Wu, R., Pan, L., & Shi, Y. (2019). Near-Field Communication Sensors. *Sensors (Basel, Switzerland)*, *19*(18). https://doi.org/10.3390/s19183947
3. Cook, J. (2023). *Hall effect vs. reed switch: Magnetic sensor comparison.* Arrow Electronics.
4. Collins, D. (n.d.). *What is racking in gantry systems, and how can it be avoided (Part 1).* Linear Motion Tips. https://www.linearmotiontips.com/what-is-racking-in-gantry-systems-and-how-can-it-be-avoided-part-1/
5. Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik, 1*(1), 269–271.
6. Paun, M.-A., Sallese, J.-M., & Kayal, M. (2013). Hall effect sensors design, integration and behavior analysis. *Journal of Sensor and Actuator Networks, 2*(1), 85–97. https://doi.org/10.3390/jsan2010085
7. Vizel, M., Alimi, R., Lahav, D., Schultz, M., Grosz, A. and Klein, L. (2025) 'Magnetic source detection using an array of planar Hall effect sensors and machine learning algorithms', *Applied Sciences*, 15(2), p. 964. doi: 10.3390/app15020964.