

AN-NAJAH NATIONAL UNIVERSITY



Faculty of Engineering and Information Technology
Computer Engineering Department

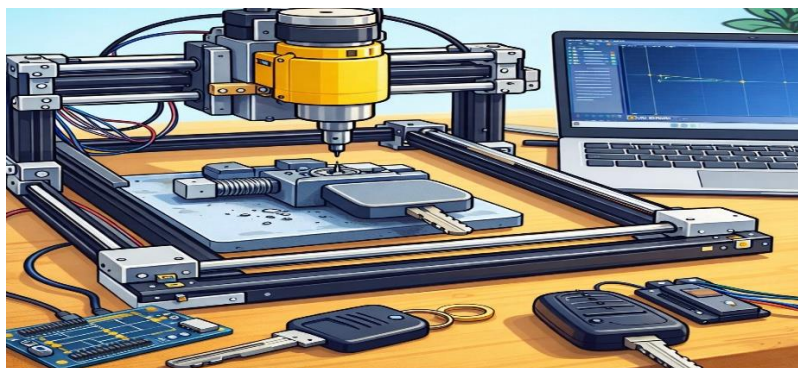
Hardware Graduation Project

The Cars Keys Cutter Project

CNC-Based Car Key Cutting Machine

Students : Mohammed Alhasoun , Obada Hussien

Supervisor Samer Mayala



A report submitted in partial fulfillment of the requirements for the
Bachelor's Degree in Computer Engineering

Academic Year: **2025 – 2026**

Acknowledgements

We would like to express our sincere gratitude to our supervisor, **Dr. Samer Mayala**, for his continuous guidance, technical support, and valuable feedback throughout the development of this project. His expertise and encouragement played a crucial role in shaping the final outcome of our work.

We would also like to thank our families and colleagues for their moral support and motivation during the development and testing phases of this project. Their encouragement helped us overcome many technical and practical challenges.

Disclaimer

The authors of this report are Mohammed Alhasoun and Obada Hussien from the Department of Computer Engineering at An-Najah National University. This report has not been altered beyond academic assessment requirements and may contain unintentional technical or linguistic inaccuracies. The opinions, results, and conclusions expressed are solely those of the authors. The university assumes no responsibility for the use of this report for purposes other than academic evaluation.

Table of Contents

1. Introduction	7
2. Constraints and Earlier Coursework	8
2.1 Constraints	8
2.2 Relevant Coursework	8
3. Methodology	9
3.1 Standards and Specifications	9
3.2 Design Overview	9
3.3 Technical Specifications	9
3.4 Hardware Design	10
3.5 Control and Automation	10
3.6 Programming and Circuit Integration	10
3.7 LaserGRBL Software and CNC Motion Control	11
3.7.1 LaserGRBL User Interface and Functions	11
3.7.2 Toolpath Visualization of the Car Key Profile	11
3.7.3 CNC Motion Control and Axis Movement	11
3.7.4 G-code Execution and System Operation	12
3.7.5 Accuracy, Safety, and System Reliability	12
3.8 Components List	14
3.9 Custom Designed Components	19
4. Results and Discussion	20
4.1 Mechanical Performance	20
4.2 Cutting Accuracy	20
4.3 System Evaluation	20
5. Conclusions and Recommendations	21
5.1 Conclusions	21
5.2 Future Work	21
6. References	22
7. Appendix A – G-code Sample	23

List of Figures

Figure 1: Complete System Overview of the CNC Car Key Cutting Machine

Figure 2: LaserGRBL interface showing the toolpath visualization and CNC motion control of the car key cutting process

Figure 3: Arduino Uno microcontroller board

Figure 4: Stepper motors used for axis motion control

Figure 5: DC spindle motor (12V, 120W)

Figure 6: Lead screws (8 mm diameter, 2 mm pitch)

Figure 7: V-slot aluminum profiles used for machine frame

Figure 8: Linear bearings (12 mm inner diameter)

Figure 9: Rubber wheels and belts for smooth motion

Figure 10: DC step-down converter module

Figure 11: Cutting tools (2 mm diameter end mills)

Figure 12: Mechanical key clamping mechanism

Figure 13: A4988 driver Stepper Motor Driver- Normal Quality

Figure 14: linear motion guide rods

Figure 15: CAD model of custom-designed mechanical mounting parts

Figure 16: CAD prototype of key geometry

Figure 17: 3D-printed prototype key used for toolpath validation

Abstract

The Cars Keys Cutter Project aims to develop an automated CNC key cutting machine using Arduino, providing an efficient and accurate solution for cutting and duplicating car keys from digital designs. The importance of this project lies in making precision key manufacturing accessible to students, technicians, and small locksmith workshops that require fast, reliable, and low-cost key production.

The project focuses on the integration of mechanical design, electronic control, and software-based automation. The system uses stepper motors for precise motion along the X, Y, and Z axes, while a high-speed DC spindle motor performs the cutting operation. An Arduino microcontroller running GRBL firmware interprets uploaded G-code files and translates them into synchronized motor movements, ensuring accurate reproduction of key geometries.

The methodology includes mechanical assembly using V-slot aluminum profiles, lead screw motion systems, and linear bearings to ensure rigidity and stability. The electronic subsystem integrates motor drivers, voltage regulation modules, and safety components. The software workflow relies on generating a G-code file that represents the key shape, which is then uploaded and executed using the Laser GRBL control software.

Experimental results demonstrate that the system is capable of producing car keys geometries with high dimensional accuracy and repeatability. Compared to commercial CNC key cutting machines, this project presents a simplified and educational alternative suitable for academic environments while maintaining practical relevance. The project provides valuable hands-on experience in CNC technology, automation, and computer-aided manufacturing.

The G-code executed by the controller follows the ISO 6983 standard format for CNC motion commands. Experimental measurements using a vernier caliper showed an average dimensional error of less than 0.15 mm.

Initial validation was performed using a 3D-printed prototype key to verify toolpath accuracy and groove geometry.

1. Introduction

Car key duplication and manufacturing require high precision due to the strict mechanical tolerances needed for proper lock engagement. Traditional manual key cutting machines rely heavily on operator skill and are prone to human error, leading to inconsistent results. Commercial CNC key cutting machines offer excellent accuracy but are often expensive and inaccessible for educational institutions and small workshops.

The Cars Keys Cutter Project addresses this gap by introducing a low-cost CNC-based solution that automates the key cutting process using digital designs. By converting a digital key model into a G-code file, the system can accurately reproduce car keys with minimal user intervention.

This project demonstrates how embedded systems, motion control, and computer-aided manufacturing can be combined into a functional CNC machine. The proposed system emphasizes accuracy, repeatability, safety, and educational value, making it suitable for both academic learning and small-scale practical applications.

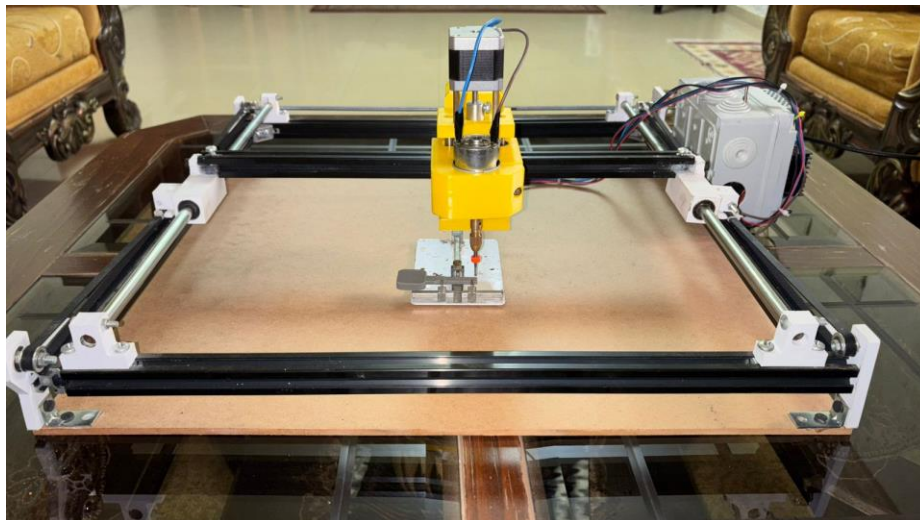


Figure 1: Complete System Overview of the CNC Car Key Cutting Machine

2. Constraints and Earlier Coursework

2.1 Constraints

Several challenges were encountered during the development of this project:

- **Mechanical Alignment:** Ensuring accurate alignment of lead screws, bearings, and linear motion components was critical to achieving precise cuts.
 - **Vibration and Stability:** Cutting operations introduce vibrations that require a rigid frame and proper mounting of components to maintain machining accuracy.
 - **Tool Wear:** Selecting appropriate cutting parameters was necessary to avoid excessive tool wear or breakage.
 - **Power Regulation:** Stable voltage supply for the spindle motor was essential to maintain consistent cutting performance.
-

2.2 Relevant Coursework

The development of this project was strongly supported by knowledge gained from previous courses:

- **Microcontrollers and Microcontrollers Lab:** Provided experience with Arduino programming, motor control, and firmware integration.
 - **Electronics and Circuits:** Enabled proper design of power regulation and motor driving circuits.
 - **Digital Systems:** Contributed to understanding control logic and system integration.
 - **Technology Project Management:** Assisted in planning, documentation, and feasibility analysis.
-

3. Methodology

3.1 Standards and Specifications

The system uses standard CNC principles and follows ISO-based G-code conventions. USB serial communication is used between the computer and the Arduino controller. In this project, the CNC job is expressed using ISO 6983-1 G-code conventions. The GRBL controller communicates over USB serial at 115200 baud using the 8-N-1 configuration.

The machine operates on a 12V DC power system, ensuring safety and compatibility with commonly available components.

3.2 Design Overview

The project consists of three main subsystems:

1. **Mechanical System:** Provides precise motion along three axes using lead screws, bearings, and V-slot aluminum profiles.
2. **Electronic Control System:** Uses an Arduino microcontroller with GRBL firmware to control motors and the spindle.
3. **Software System:** Generates and executes G-code files using Laser GRBL software.

3.3 Technical Specifications

Value	Parameter
< 0.15 mm	Positioning Accuracy
-1 mm	Cutting Depth
DC motor (12V, 120W)	Spindle Speed
30–500 mm/min	Feed Rate Range
Defined by X–Y axis travel limits	Working Area

3.4 Hardware Design

- **Frame:** V-slot aluminum profiles with metal mounting brackets.
 - **Motion System:**
 - Lead screws (8mm diameter, 2mm pitch)
 - Bearings with 12mm inner diameter
 - Rubber wheels and belts for smooth motion
 - **Motors:**
 - Stepper motors for X, Y, and Z axes
 - DC spindle motor (12V, 120W)
 - **Cutting Tool:**
 - Collet size: 3.17mm
 - Cutting bit diameter: 2mm
 - **Voltage Regulation:** DC step-down module for spindle power stabilization
 - **3D Printed Parts:** Used for bearing housings, structural supports, and protective covers
-

3.5 Control and Automation

- The Arduino runs **GRBL firmware**, which interprets G-code commands.
 - The spindle motor is activated and controlled during cutting operations.
 - Axis movements are synchronized based on G-code instructions.
 - Limit switches ensure safe homing and prevent mechanical overtravel.
-

3.6 Programming and Circuit Integration

The control software **LaserGRBL** is used to upload and execute G-code files. The workflow includes:

1. Designing the key geometry.
2. Generating the corresponding G-code file.
3. Uploading the file to LaserGRBL.
4. Arduino interprets commands and drives motors accordingly.

The provided G-code file defines tool paths, cutting depth, feed rate, and movement sequence.

3.7 LaserGRBL Software and CNC Motion Control

LaserGRBL is used as the main software interface for controlling and monitoring the CNC-based car key cutting machine. The software enables loading, visualizing, and executing G-code files generated from the digital key design. It serves as a communication link between the computer and the Arduino controller running GRBL firmware.

LaserGRBL communicates with the controller through a USB serial connection operating at a baud rate of 115200 with an 8-N-1 configuration. The software streams G-code commands to the controller in real time, ensuring accurate and synchronized motion of the CNC machine.

3.7.1 LaserGRBL User Interface and Functions

The LaserGRBL user interface consists of several functional sections that allow efficient control of the CNC machine. The connection panel is used to select the correct COM port and establish communication with the controller. Once connected, the software displays the machine status and current axis positions.

The file management section allows loading the G-code file that defines the car key profile. Execution progress and estimated machining time are displayed during operation. Manual jogging controls are also available, enabling precise movement of the X, Y, and Z axes for homing, alignment, and calibration purposes.

3.7.2 Toolpath Visualization of the Car Key Profile

LaserGRBL provides a graphical preview of the toolpath before executing the cutting process. This preview displays the exact trajectory that the cutting tool will follow during machining, based on the loaded G-code file.

The displayed toolpath represents the geometric profile of the car key, including the teeth and grooves. The horizontal direction corresponds to movement along the X-axis, while the vertical variations represent changes along the Y-axis. This visualization allows verification of the key shape, cutting sequence, and dimensional accuracy before actual material removal.

3.7.3 CNC Motion Control and Axis Movement

The CNC machine operates using a three-axis motion system controlled by GRBL firmware. LaserGRBL sends sequential G-code commands to the controller, which interprets them and generates step signals for the stepper motors.

- The **X-axis and Y-axis** are responsible for forming the two-dimensional shape of the key profile using linear interpolation commands (G1).

- The **Z-axis** controls the cutting depth by lowering the tool to a predefined depth during cutting and retracting it to a safe height after completion.
- **Rapid positioning commands (G0)** are used for non-cutting movements such as tool positioning and homing.

Feed rates specified in the G-code control the speed of tool movement. Lower feed rates are used during cutting operations to improve accuracy and surface finish, while higher feed rates are used for rapid movements.

3.7.4 G-code Execution and System Operation

After verifying the toolpath through the preview, the cutting process is initiated from LaserGRBL. The software streams G-code commands line by line to the Arduino controller. GRBL processes each command in real time, ensuring coordinated motion between all axes.

During execution, the current position of the tool is displayed on the interface, allowing continuous monitoring of the machining process. The software also provides pause and emergency stop functions to enhance operational safety.

3.7.5 Accuracy, Safety, and System Reliability

The integration of LaserGRBL with GRBL firmware provides a stable and reliable CNC control system. The ability to preview the toolpath, manually control axis movement, and monitor execution significantly reduces the risk of errors and tool collisions.

This control approach ensures accurate and repeatable cutting of car keys, making the system suitable for both educational applications and small-scale practical key duplication.

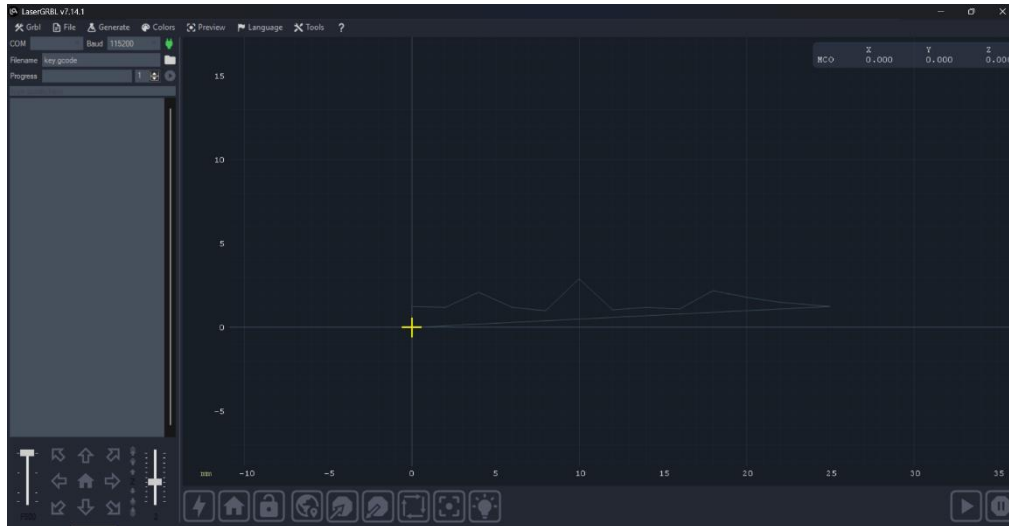


Figure 2: LaserGRBL interface showing the toolpath visualization and CNC motion control of the car key cutting process

3.8 Components List

- Arduino Uno



Figure 3

- Stepper Motors



Figure 4

- DC Spindle Motor (12V, 120W)



Figure 5

- Lead Screws (8mm, 2mm pitch)



Figure 6

- V-slot Aluminum Profiles



Figure 7

- Bearings (12mm inner diameter)



Figure 8

- Rubber Wheels and Belts



Figure 9

- DC Step-Down Converter



Figure 10

- Cutting Tools (2mm)



Figure 11

- Mechanical Key Clamp

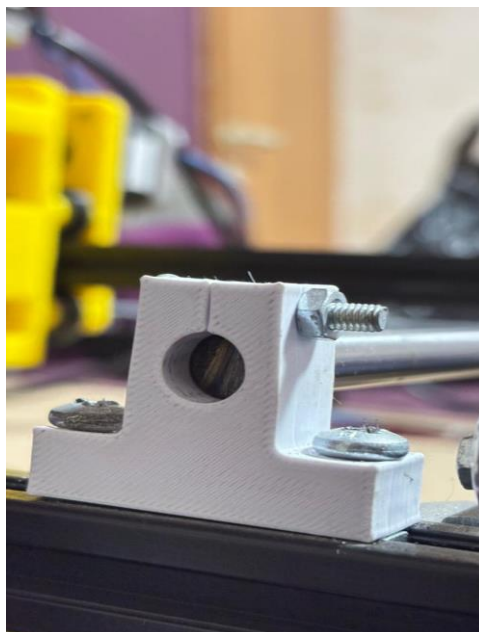


Figure 12

- **A4988 driver Stepper Motor Driver- Normal Quality**



Figure 13

- **linear motion guide rods**



Figure 14

3.9 Custom Designed Components

Figure X: CAD model of mechanical mounting parts

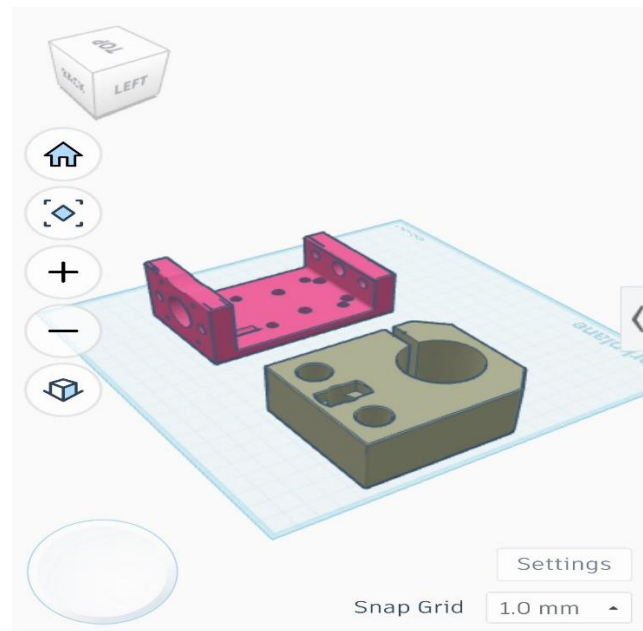


Figure 15

- **Figure Y:** Key geometry prototype (CAD).

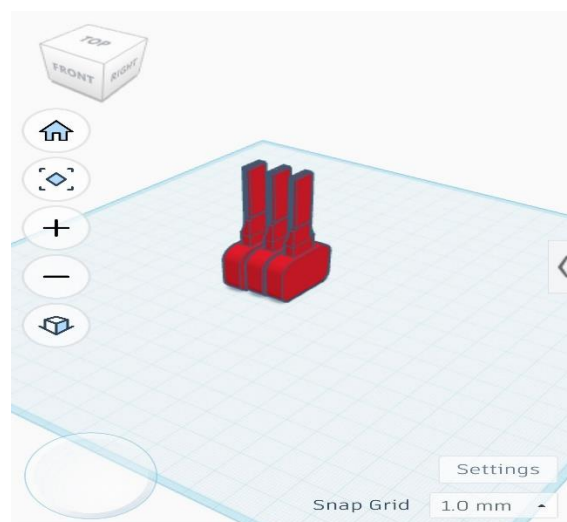


Figure 16

4. Results and Discussion

4.1 Mechanical Performance

The machine demonstrated stable and smooth motion across all axes. The rigid aluminum frame minimized vibration during metal cutting operations.

4.2 Cutting Accuracy

The produced 3D-printed prototype keys closely matched the digital design. Repeated cutting operations showed consistent depth and shape, confirming system reliability.

4.2.1 Quantitative Accuracy Analysis

To evaluate the cutting accuracy of the CNC key cutting machine, several critical dimensions of the produced keys were measured and compared with the original digital design. Measurements were performed using a vernier caliper. The average dimensional error was found to be less than **0.15 mm**, which indicates good precision and repeatability suitable for practical key cutting applications.



Figure 17: 3D-printed PLA prototype key used for toolpath and groove geometry validation

4.3 System Evaluation

The integration of Arduino, GRBL, and LaserGRBL resulted in a reliable CNC control system suitable for educational and small-scale production use.

5. Conclusions and Recommendations

5.1 Conclusions

This project successfully designed and implemented a CNC-based car key cutting machine. The system achieved accurate, repeatable results while maintaining a low-cost and educational design. It demonstrates the practical application of embedded systems, automation, and CNC manufacturing.

5.2 Future Work

- Support for additional key types and full metal key cutting validation
 - Improved vibration damping.
 - Integration of automatic key alignment.
 - Enclosure for enhanced safety.
 - Image processing: Using Raspberry Pi AI to capture the shape of the key and cut it.
-

6. References

1. Arduino, "Arduino® UNO R3 Datasheet (A000066)," Arduino Documentation. <https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf?>
2. GRBL Firmware Documentation. https://lasergrbl.com/usage/?utm_source
3. CNC Machining Fundamentals. https://academy.titansofcnc.com/files/Fundamentals_of_CNC_Machining.pdf

7. Appendix A – G-code Sample

The following G-code program defines a basic CNC setup and controls the linear cutting motion of the tool along the **XY plane** with a fixed cutting depth. The program begins with unit selection, positioning mode, plane selection, and feed-rate mode, then performs safe positioning (G0), followed by controlled cutting moves (G1), and finally ends the program.

```
G21
G90
G17
G94

G0 Z5
G0 X0 Y1.25

G1 Z-1.5 F120
G1 X2 Y1.20 F30
G1 X4 Y2.10
G1 X6 Y1.20
G1 X8 Y1.00
G1 X10 Y2.90
G1 X12 Y1.05
G1 X14 Y1.20
G1 X16 Y1.10
G1 X18 Y2.20
G1 X20 Y1.80
G1 X22 Y1.50
G1 X25 Y1.25

G0 Z5
G0 X0 Y0
M30
```

7.1 Initialization and Mode Selection

G21 – Set Units to Millimeters

Sets all dimensional units in the program to **mm**, ensuring correct interpretation of coordinates and cutting depth.

G90 – Absolute Positioning Mode

All tool movements are referenced to a fixed origin point (0,0,0). This improves predictability and repeatability during machining.

G17 – XY Plane Selection

Selects the **XY plane** (primarily affects arc commands like G2/G3 and certain cycles). It is commonly included as a standard setup command at program start.

G94 – Feed Rate per Minute

Sets feed mode to **units per minute (mm/min)**, which is the standard feed interpretation for most milling operations.

7.2 Rapid Positioning (Non-Cutting Moves)

G0 – Rapid Positioning

Moves the tool quickly without cutting, used for safe travel and positioning.

- G0 Z5 retracts the tool to a safe height before lateral movement.
 - G0 X0 Y1.25 positions the tool above the starting point.
-

7.3 Cutting Phase (Linear Interpolation)

G1 – Linear Cutting Move

Performs a straight-line move at the specified feed rate while cutting.

- G1 Z-1.5 F120 plunges to a cutting depth of **-1.5 mm** at **120 mm/min** (controlled vertical feed).
 - G1 X2 Y1.20 F30 starts the cutting path at **30 mm/min** for better precision.
 - Subsequent G1 X... Y... lines continue the toolpath using the same feed rate until changed.
-

7.4 End and Safe Return

- G0 Z5 retracts the tool after cutting to avoid collisions.
- G0 X0 Y0 returns the tool to the origin.
- **M30 – Program End and Reset** ends the program.

