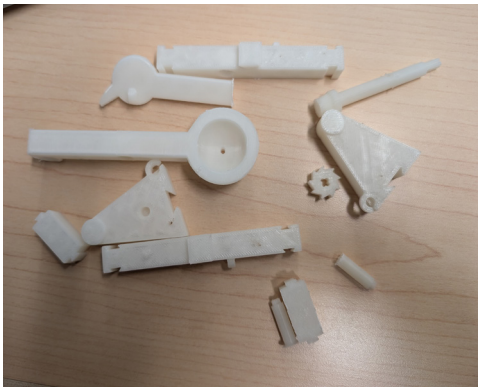
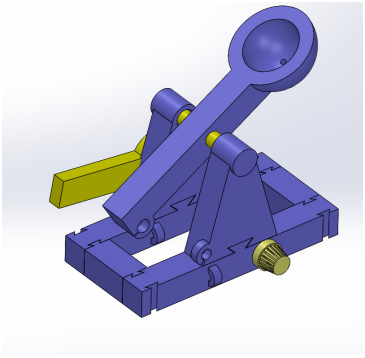
Maksim Stroikin

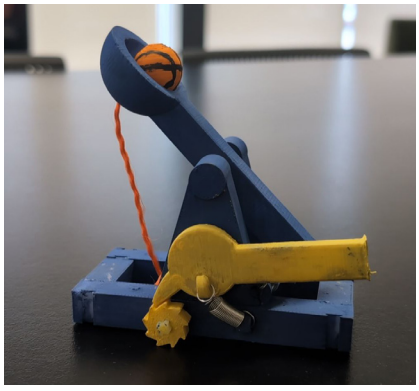
Engineering Systems and Computing (Co-op) Student | University of Guelph

Richmond Hill, ON | mstroikin11@gmail.com | 437-557-6980 | linkedin.com/in/maksim-stroikin

Kinder-Maxi Basketball Launcher







What:

Designed a spring-powered basketball launcher toy through a series of iteratively improved and tested prototypes

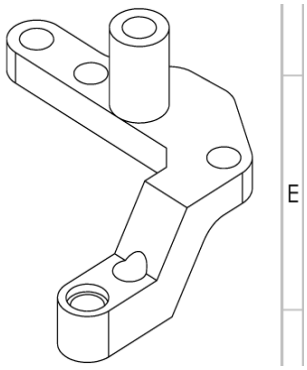
How:

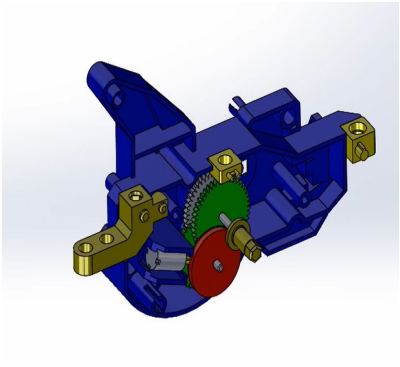
Created hand-drawn technical sketches and modeled all components in **SolidWorks**. Sliced and 3D printed parts using a **Prusa 3D printer**. Integrated a ratchet and pawl mechanism to store and release kinetic energy, allowing for controlled tension and launch.

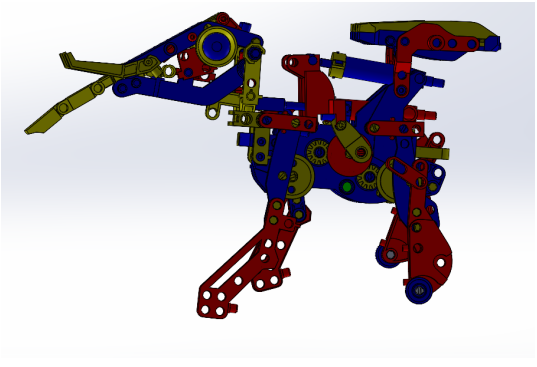
Result:

The resulting toy could launch projectiles at an adjustable distance of up to 3 meters away while fitting inside a 10cm Kinder Maxi Egg when disassembled.

Toy Elephant 3D Assembly







What:

Worked in a 4-person team to reverse engineer and digitally replicate an elephant model from a 12-in-1 solar hydraulic kit.

How:

Individually measured each component using a caliper, 3D modeled and created 2D technical drawings in **Solidworks**, and followed the kit's manual to build a complete digital assembly. Designed and animated an exploded view to illustrate the model's internal structure and assembly process.

Result:

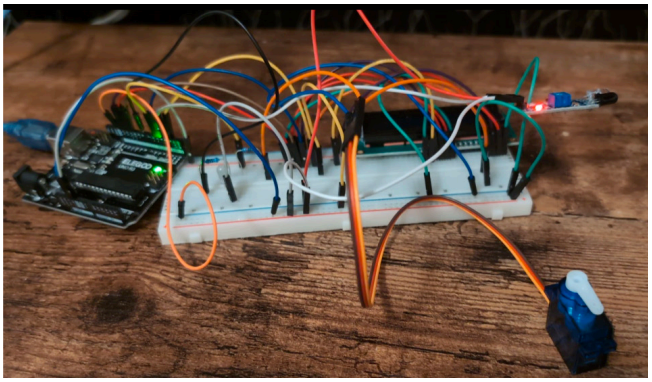
Produced a detailed assembly consisting of 68 unique components and 139 total parts, all modeled and animated within **SolidWorks.**

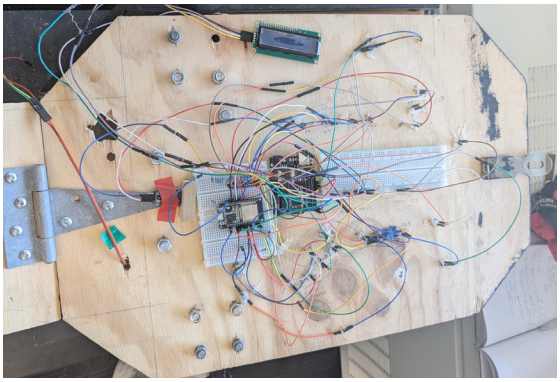
Maksim Stroikin

Engineering Systems and Computing (Co-op) Student | University of Guelph

Richmond Hill, ON | mstroikin11@gmail.com | 437-557-6980 | linkedin.com/in/maksim-stroikin

Pinball Machine





What:

Created an **Arduino**-powered pinball machine automating the collection of aluminum by allowing users to play a game of pinball in exchange for a pop tab.

How:

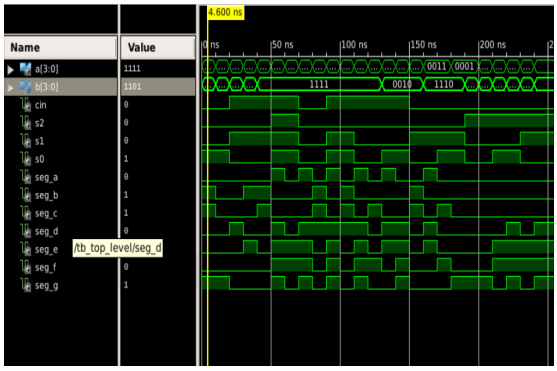
Designed and assembled a circuit utilizing six sensors, an LCD display, 4W speakers, and 30 LEDs. Programmed an **Arduino** and **ESP32** module using **Arduino**'s **C++** based IDE to detect user interactions, activate sound effects and lights, track user scores, and upload the results to a dedicated website.

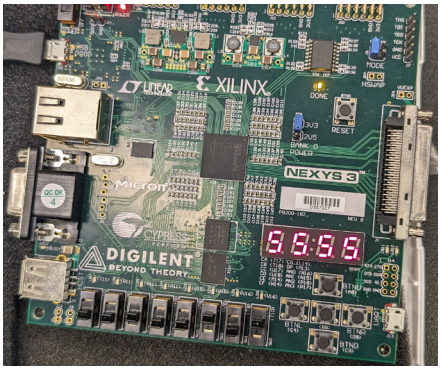
Result:

The design was showcased at a campus carnival event and contributed to the collection of large quantities of aluminum.

4-bit Arithmetic Logic Unit

S_2	S_1	S_0	C_{in}	Operation	Function
0	0	0	0	G = A	Transfer A
0	0	0	1	G = A+1	Increment A
0	0	1	0	G = A+B	Addition
0	0	1	1	G = A+B+1	Add with carry input of 1
0	1	0	0	$G = A + \bar{B}$	A plus 1's complement of B
0	1	0	1	$G = A + \bar{B} + 1$	Subtraction
0	1	1	0	G = A - 1	Decrement A
0	1	1	1	G = A	Transfer A
1	0	0	X	$G = A \wedge B$	AND
1	0	1	X	$G = A \vee B$	OR
1	1	0	X	$G = A \oplus B$	XOR
1	1	1	X	$G = \bar{A}$	NOT (1's complement)





What:

Designed and implemented a 4-bit Arithmetic Logic Unit (ALU) capable of performing fundamental arithmetic and logic operations.

How:

Used **VHDL** to incorporate a half adder, full adder, logic circuit, B-logic input, and arithmetic circuit modules. Created a test bench and simulated the design using **Xilinx ISE (Foundation tools)**. Programmed and mapped the 4-bit ALU onto the **NEXYS 3 FPGA** board.

Result:

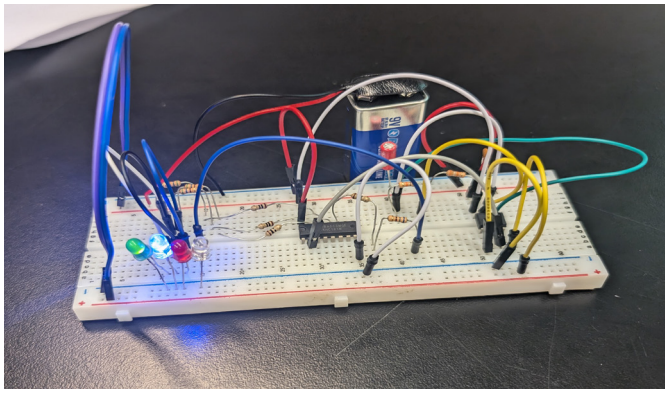
The design successfully executed all assigned operations with 4-bit numbers and displayed the result on a 7-segment display.

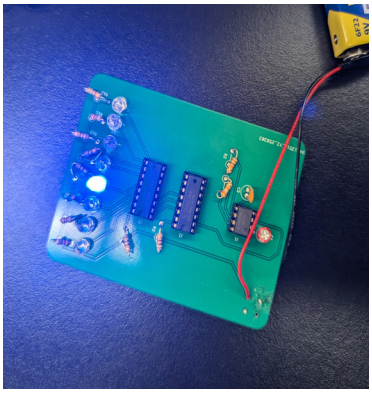
Maksim Stroikin

Engineering Systems and Computing (Co-op) Student | University of Guelph

Richmond Hill, ON | mstroikin11@gmail.com | 437-557-6980 | linkedin.com/in/maksim-stroikin

8-LED Sequencer with Timer and Multiplexer





What:

Designed and built an 8-LED sequencer circuit that lights up LEDs one at a time in a timed pattern, using a timer, multiplexer, and binary counter

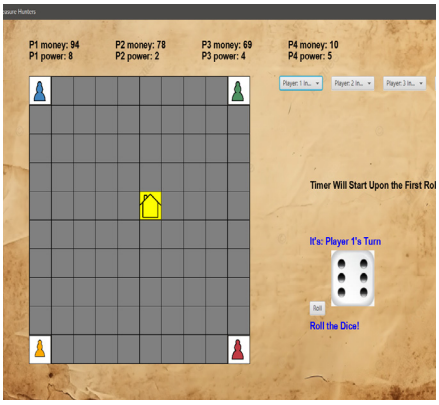
How:

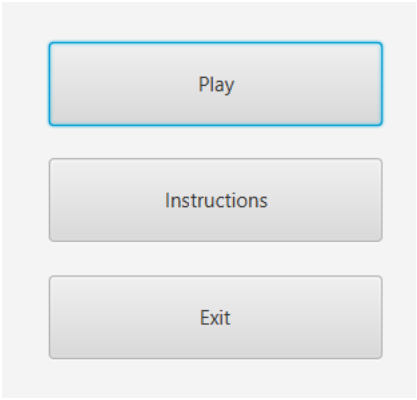
Created a prototype of the circuit on a breadboard, then created a custom **Printed Circuit Board (PCB)** using **KiCad**. Soldered all components onto the board, including a **SN74HC138N** multiplexer, **SN74HC161N** binary counter, and **NE555P**, 12 resistors, 2 capacitors, and 8 LEDs.

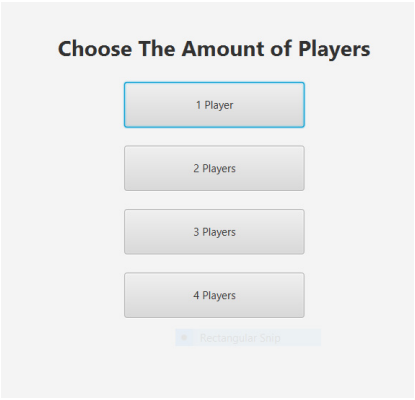
Result:

Successfully produced a compact, functional circuit that lights up 8 LEDs in a repeating, sequential pattern.

Java FX 2-4 Player-game







What:

Developed a 2–4 player turn-based strategy game using **JavaFX** as a final project for an Object-Oriented Programming course.

How:

Led a four-member team, utilizing **GitHub** for progress tracking and code collaboration. Using **Java**, I coded and implemented features including a “fog of war” system, random item generation, a shop for buying and collecting items, and player-versus-player battles.

Result:

Successfully created a multiplayer game that met and exceeded course requirements, earning a final grade of 100%.