A-

UNIVERSITY OF CALIFORNIA AT IRVINE
The Henry Samueli School of Engineering
Department of EECS

Instructor: *Ray Klefstad*
Team Member:
Wong, Albert Tin King
Chan, Leona

EECS 129B
Winter 2006

Automated Board Game System

Fully customizable, our automated board game system does it all!
Your board game can now jump out of your monitor and be played in front of your eyes.
Our system is capable of moving pieces from any location on the board to another location.

Table of Contents

1: Project Description

The main objective for this research is to build a 2-dimension unit board with the abilities to automate pieces movement with computer controls. The purpose of such automation ability is for accuracy and precision since we controls the movement and steps precisely with computer programs.

The hardware consists of a board with a moving unit built under the board that is able to move in longitude and latitude direction. The latitude and longitude direction will have 4 wooden sticks installed to enable movements under the board. There will be 2 stepper motors installed on the wooden sticks in the latitude and longitude direction which control the steps of the electromagnetic part. Attached to the latitude direction is an electromagnetic part, that has the ability to "catch and release" another magnet on top of the board. When the electromagnetic part is raised up, it "catches" a magnet that is on the surface of the board, drag it to specify location and release it. Thus we can simulate movements on the board with such a unit.

The stepper motor is connected to the microcontroller, Arduino, which is placed under the board and connected to a computer through a USB cable. The serial communication will allow the PC to communicate with the board, and to control the steps and directions of the electromagnetic unit. The software that will simulate the movement on the board will run entirely on a computer and it stores all the information of the pieces that is on top of the board. The software also calculates the movement process from one location of the board to another location on the board and sends the function through serial communications to the microcontroller. The microcontroller will read the input from the software program and implement the movements on the board. Our sample game will be a human versus human tic tac toe game simulation that can test the connection between PC and board and the correctness of the moving unit.

2: System Level Block Diagram

The system of hardware consists of 1 "moving unit", 2 set of perpendicular wheel tracks, and an outer box structure.

1. Moving Unit

The moving unit is built with some form board as the case (4 4" by 2.5", 1 4" by 4"), a normal motor with 2-input (V+, GND), a special gear box (with a moving arm part) connected to the motor, a magnet (for Catch/Release use), 2 stepper motors (attached 1 washer around the motor axis), and 1 stepper motor driver. This set of motor and driver (A) is used for forward/downward direction movement of moving unit.

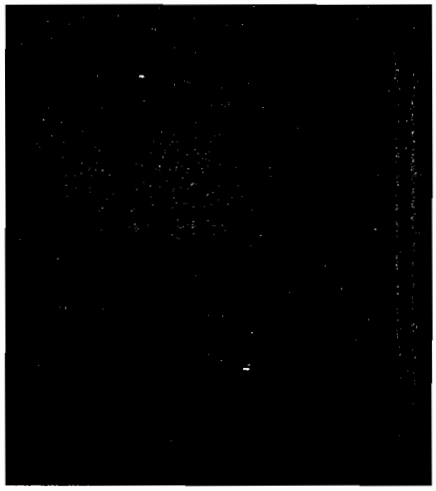
The moving unit is the major unit in hardware, and it determined the minimum size of the whole system structure. The current version is a moving block with 4" by 2.5" by 5".

2. 2 Sets of Wheel Tracks

One set of wheel tracks is made of 2 wood sticks (11" 3/8), 2 stepper motors, 2 form blocks. The form blocks are used to install the steppers as one on each, and connected 2 form blocks with 2 wood sticks parallel so it fits the moving unit to sit on top on the wood sticks as a track.

The other set of wheel tracks is only made of 2 wood sticks (11" 3/8).

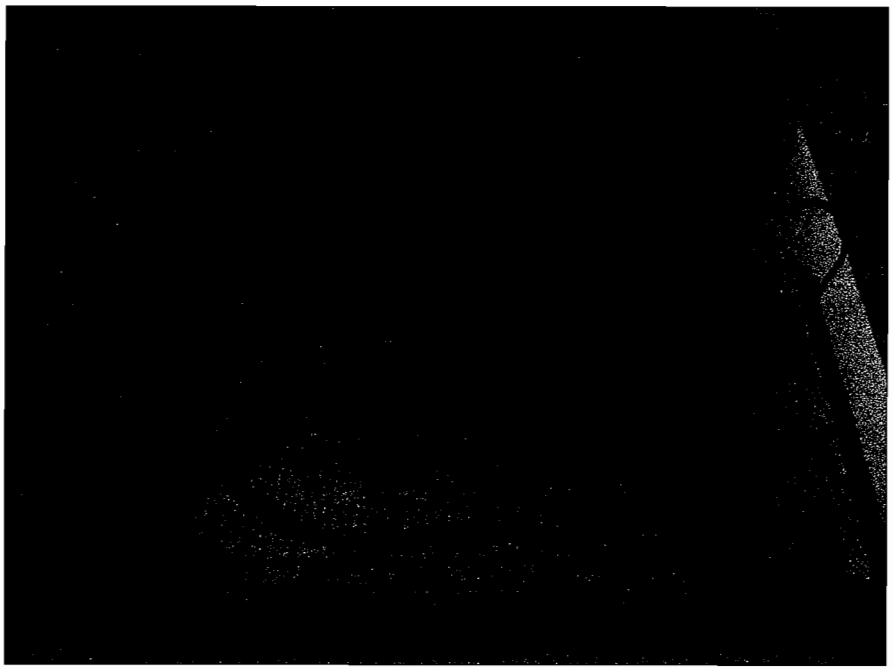
The wood sticks determined the maximum size of the whole system structure and the actual movable range of moving unit. And it is installed on the side of the inner structure of the case positioned just the height under the washer of the motor.



3. The outer structure

The outer structure made of a set of foam boards (4 7" by 13", 1 11" by 13"), a breadboard with half of the wires and circuits, an Arduino USB borad, a stepper motor driver, and a 12-Volt adapter.

The outer structure is sized as 11" by 13" by 7". We fixed the Arduino USB board and driver beside the breadboard at the bottom of the structure as shown in the picture below. The breadboard is connected with the 12-Volt adapter as power source. And this motor driver (B) is connected to the 2 motors in Wheel Tracks for left/right direction movement.



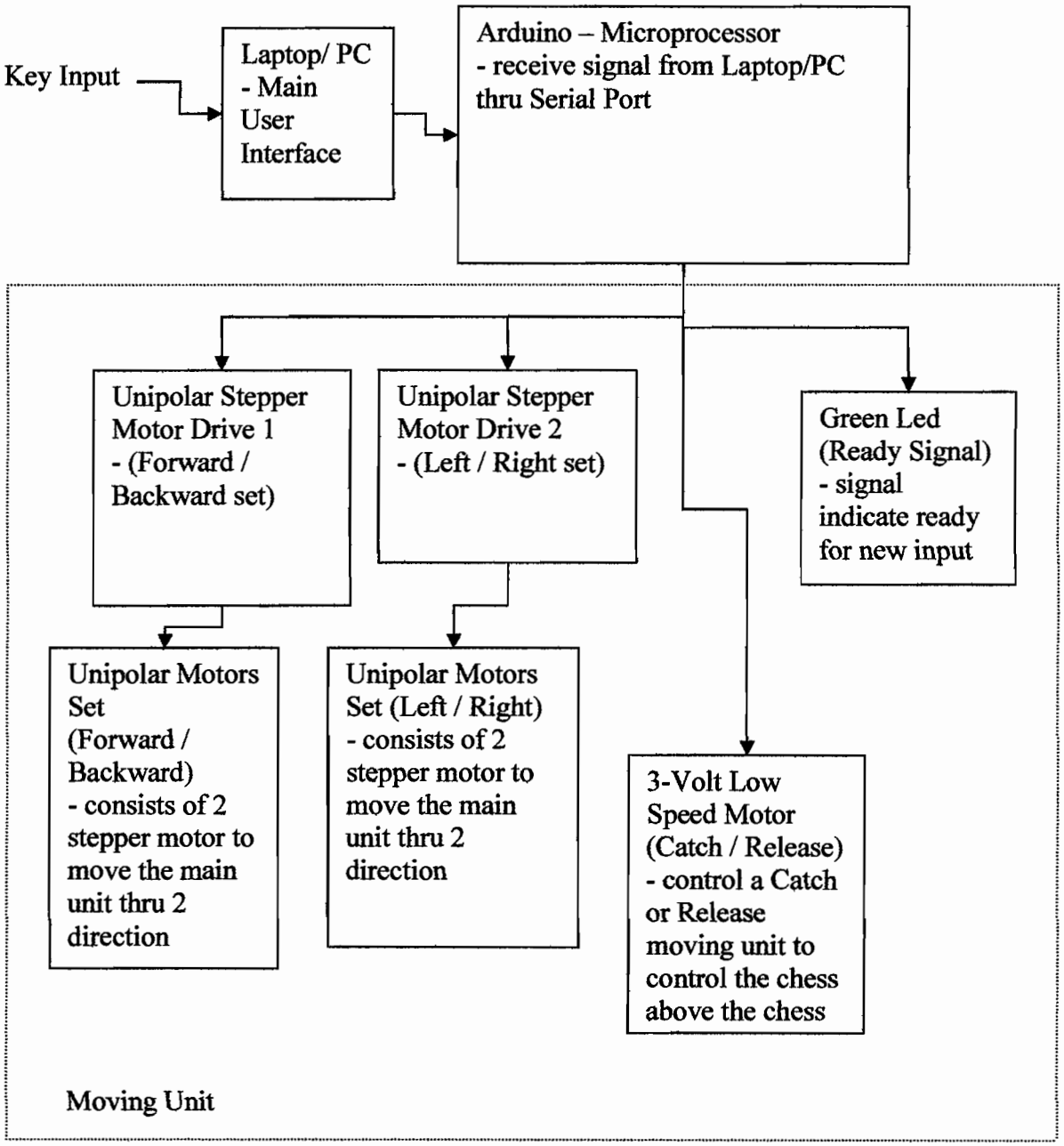


Figure 1. Moving Unit System Level Block Diagram

3: Circuit Level Block Diagram

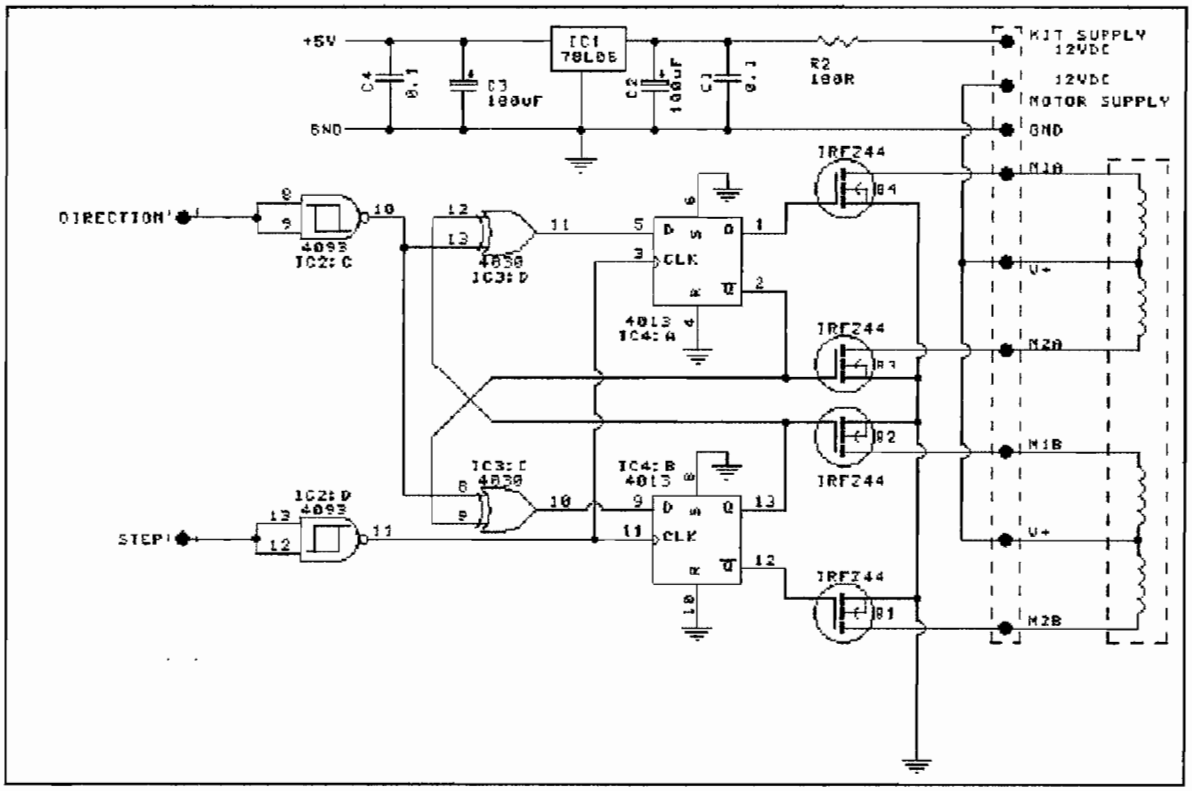


Figure 2. Unipolar Stepper Motor Circuit Schematic

Unipolar Stepper Motor Circuit Schematic

- Direction, Step connected to Arduino USB board for input signal
- M1A, M2A connected to Stepper Motor Light/Dark color wires (2 wires as a set), and M1B, M2B connected to the remaining color wires set.

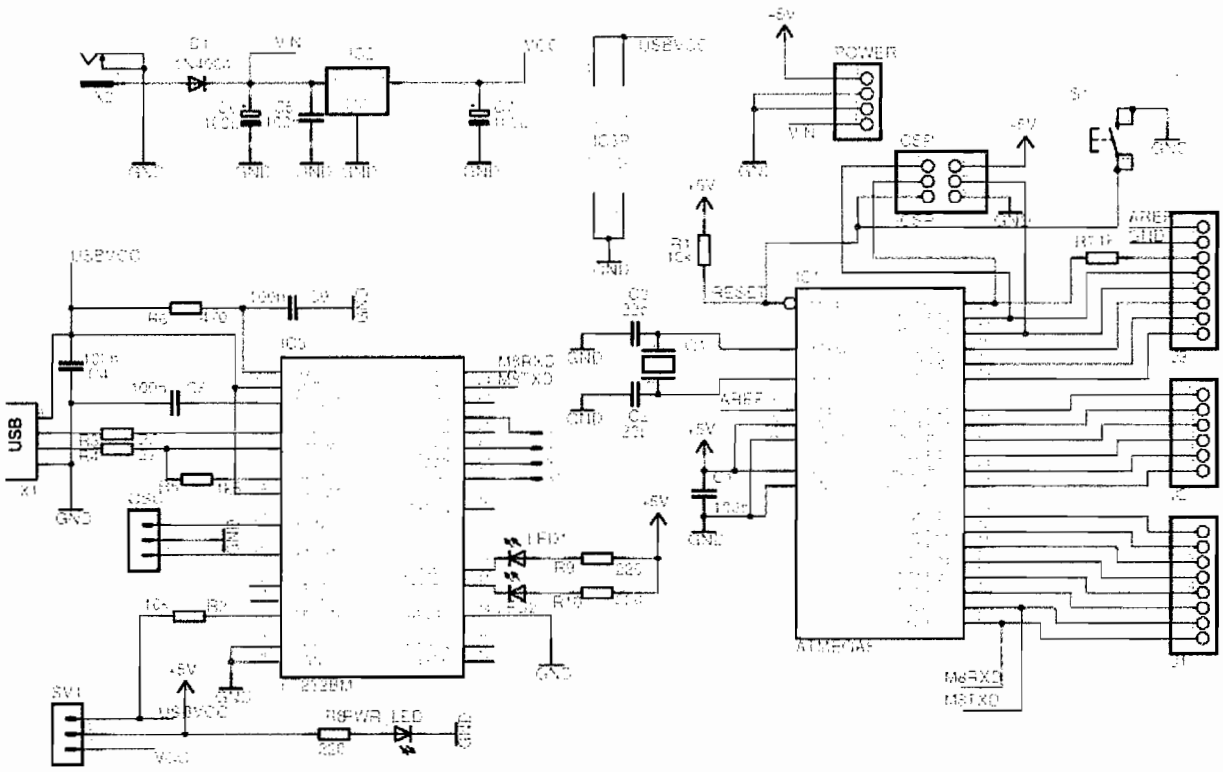
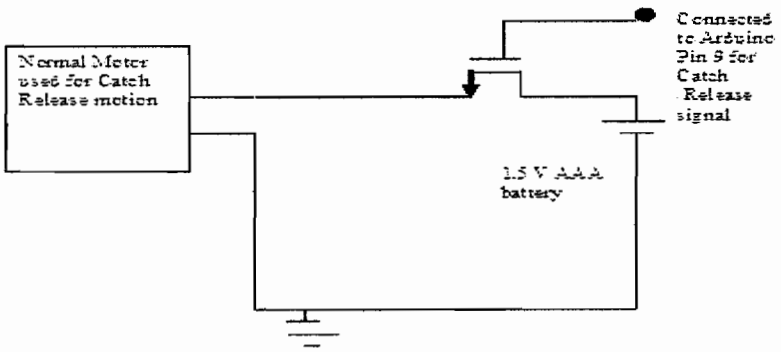


Figure 3. Arduino Circuit Schematic

Arduino Circuit Schematic

- Pin 11 connected to Left/Right set motor driver as STEP output
- Pin 10 connected to Left/Right set motor driver as DIRECTION output
- Pin 3 connected to Forward/Backward set motor driver as STEP output
- Pin 2 connected to Forward/Backward set motor driver as DIRECTION output
- Pin 9 connected to Catch/Release motor as signal input
- GND connected to breadboard as common ground
- USB connected to Laptop/PC for keystroke input
- Pin 13 connected to green LED to indicate signal as ready for input

Figure 4. Catch/Release Motor Circuit Schematic



4: Software Description

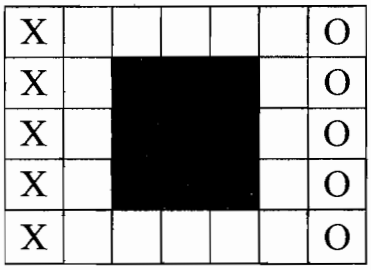


Figure 5. Tic Tac Toe Board Layout

The software part of our project is mainly written in Java Applet with the Java Serial Communications API. The software serves as controller to the hardware part while keeping track of where all the pieces are located. The Java applet has the ability to write commands into the serial port and those commands will be read by the program in our micro-controller, Arduino.

The program that resides in the Arduino microcontroller is written in C. It produces high and low pulses with timing delays for 5 digital output ports that control the direction and the power to the stepper motors and our catch and release unit. Basically, the program in Arduino will read the byte from the serial port when the serial byte is available and it will run one of the 16 functions that are set in the program. The set of functions are as follows:

Function	# of Steps	Code
Move Left	1	A
Move Left	2	B
Move Left	3	C
Move Left	4	D
Move Right	1	E
Move Right	2	F
Move Right	3	G
Move Right	4	H
Move Top	1	I
Move Top	2	J
Move Top	3	K
Move Bottom	1	L
Move Bottom	2	M
Move Bottom	3	N
Catch	N/A	O
Release	N/A	P

Table 1. Arduino Functions Code

The other part of the software is a tic tac toe Java applet program with two players (Human Vs Human). The program is written with Java Applet so it can run in a web browser environment and it will play a game of tic tac toe with a board layout that is similar to the one shown in Figure 5. There will be 5 pieces for each side, X and O respectively, and the board is drawn with width 7 and height 5. When the game is in progress, the pieces from each side are being brought into the Tic Tac Toe Board area according to the players' movements in order from top to bottom. The screen shot of the applet is shown in Figure 6. The area of the tic tac toe playing area will be clickable and the java applet will have action listeners that will listen to an action performed on the board. The program will keep track of the movements of all the pieces and calculate the movements of the pieces then send the respective function code to the serial port so that Arduino can read from it and simulate the movements. I have used the native libraries of RXTX, which provides serial communications in the Java development environment.

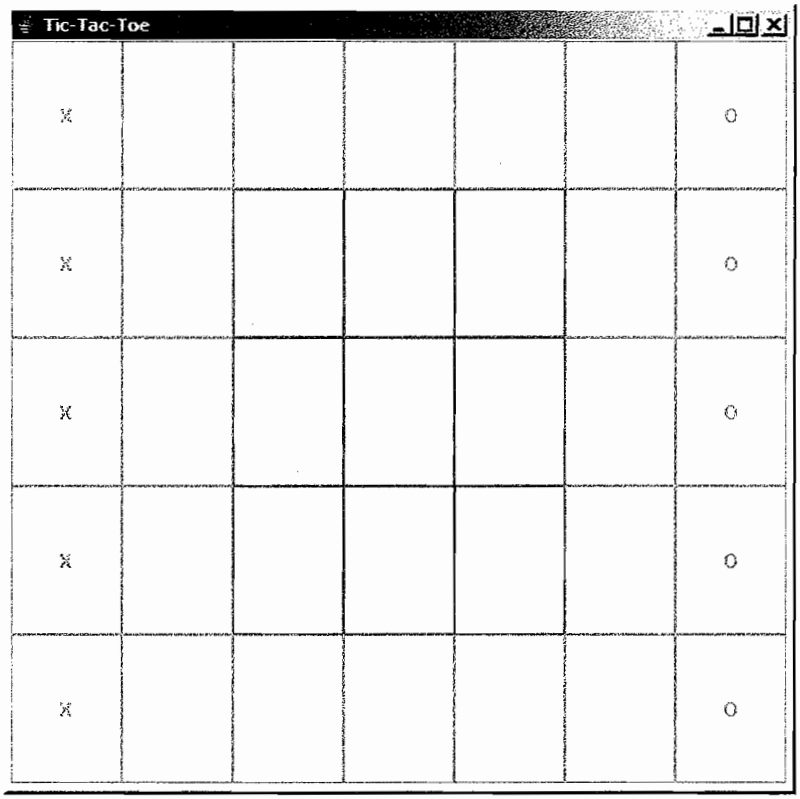


Figure 6. Java Applet Tic Tac Toe Program with clickable buttons

5: System Test Plan

1. Stepper Motor and drivers

The stepper motor and its drivers are tested by connecting the stepper motor to a 12V power source to see if the gear for the stepper motor actually turns left or right.

2. Moving Ability with Stepper Motor Internal Controls

With the selection of internal stepper motor control, we can test if the drivers and the hardware are working properly without any software interfaces. The system can be tested for its moving ability after all the hardware is built with the stepper motor internal control switch.

3. Moving Ability with software interfaces

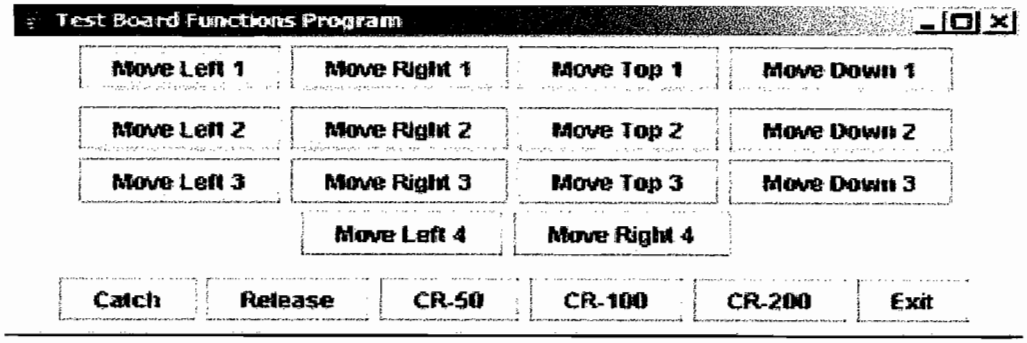


Figure 7. Sample GUI to test on board functions

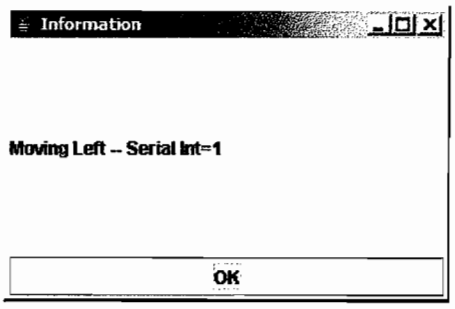


Figure 8. Sample on-click on the GUI

The software interfaces include simple functions with GUI such as the one in Figure 7. The software has an action listener which will listen to the action performed on the GUI. When any of the GUI button is clicked, a pop up box will pop up like the one in Figure 8 informing the user of such actions, the software will sent the appropriate serial code to the serial communication port. The microcontroller will read from the serial port and perform the movement.

4. **Ability to move from one location on the game board to another location on the board**

This is done with the assistance of the software program shown in Figure 6. The software program will calculate the location and the movements needed to move from one location to another and send the appropriate function codes to the serial port to simulate the movement of the pieces. The function codes will be read by the microcontroller and the movement will be simulated.

6: Parts list

Total Cost = $207.18

FROM SPARKFUN (http://www.sparkfun.com)

quantity	description	Price
1	Arduino USB board	$31.95

FROM JAMECO (http://www.jameco.com)

quantity	part numbers	description	Price
1	36792	100' reel 22awg, solid black wire	$5.49
1	36856	100' reel 22awg, solid red wire	$5.49
1	36920	100' reel 22awg, solid yellow wire	$5.49
1	211086	green led	$0.50
1	38236CM	TRANSISTOR,2N2222A,NPN,GP	$0.39
1	NA	7" USB Cable	$3.00

FROM HOBBY ENGINEERING (http://www.hobbyengineering.com/)

2	K179	Unipolar Stepper Motor Driver Board Electronic Kit	$39.98 (19.99 each)
4	27964	Phase 12-Volt 90Ohm Unipolar Stepper Motor	$48.00 (12.00 each)

FROM FRY'S ELECTRONICS (http://www.frys.com)

1	71104	Mechanical Ostrich	$6.95
1	PST-1800MF	Syntech: AC/DC Adapter, 1800mA	$14.99

FROM HOME DEPOT (http://www.HomeDepot.com)

1	7-GE XL10	Lexan Polycarbonate 18"*24"	$12.95
1	148451	¾" Disc Iman Magnets	$3.00
4	B-548	Sliding Screen Door Roller Assembly	$20.00(5.00 each)
1	NA	1" Plastic Washer	$3.00
1	102566	Stanley Hot Melt Glue Sticks	$3.00

FROM CVS Pharmacy

2	NA	36*24 Foarm Board	$11.90(5.95 each)
1	NA	12" Wooden Stick	$3.00

7: Summary

This research project is actually interesting and educating. The whole system we built consists 60% of hardware and 40% of software. However, it does not mean that it is easier on software part. Indeed, the software part has the development flexibility in this project but it also carries 80% to 90% of selling points. This system can be connected with many different software and games to make it fun and attracting. It has to rely on the programmer's creativity and knowledge. The project is satisfactory but incomplete. It sound complicated, but we are going to explain it in very details.

We did not learn any hardware assembly before. In fact, we encounter many problems with parts collection and when we are assembling the hardware. The original idea of the project is to use an electro-magnet as a Catch/Release system since it save space, and it is easier to control thru digital signal instead of the use of gear box and a normal motor. However, a major problem that we have encountered is that the electro-magnet heats up so fast that, it will melt off any attachment around it. It is not worth to burn the system down just for making the system work perfectly at low price for 3 minutes. We have also considered using a normal 2-input motor but it still does not solve the problem of moving part perfectly. The side effects of using a 2-input motor is that we can hardly control the turn of motor as steps. It has the problem of over-turn or under-turn. Also the size has been increased to certain level that would limit the movable area of the moving unit. Stepper motor is an ideal solution of this problem, but yet it would need a special customized gear box and it would cost a lot more than the system we are using now.

Parts collection is the second problem we encountered. The wood stick is standard sized and thus limits the movable area of the moving unit. In addition, gear box is hard to get since most gear box and cogwheel is customized for special models, and the stores have a very limited selection. Therefore, we decided to use foam board as the main gear box material for building the system although it is not desirable. Nevertheless it is easier to cut to a box that fits our desirable size and easier to assemble the boards together. However, foam board also comes with the disadvantages that it is easy to break, bent, or burn. It keeps the inside temperature of the system relatively high since the foam board absorb heat and pose the risk of melting the hot glue we have used to glue some of the parts together. Metal or wood board would be more desirable, but it definitely would be much harder to cut and put the system together. We knew some of these problems in advance and some later progress, but we still decided to use the foam board because

it is a cheaper solution and it will also be easier to assemble. If we are going to put this design into the market, the hardware design would actually be sent to industrial professional to customize and produce. Therefore, the cost of total production would be cut to about one-eighth to one-tenth as of the cost now. It also will have a larger movable range for the inner structure since the parts will be customized and cut down to its ideal size.

To make use of stepper motor give us a small problem since we have never learned how to control a stepper motor. The motor we use has 5-input, with colored wires, and with no instruction or description when it is sold to us. In order to figure out how to control it, we need to do another small research on this topic. As we mentioned above, the process is fun and educational. Now, it works perfect in our system.

Choosing the perfect software to operate the system is important. We have chosen Java as the software programming language because of its flexibility to operate in different computer platforms. It does mean that it will work also with another programming language, but we choose to let our future customer to enjoy the capability of making a program in Java themselves.

This is a fun project, and it is also a potential gaming product in the market. With sufficient fund and hardware support, we can add more functions and features into the project to make it more enjoyable. One possible add-on is a special system that allows LED game board and the game pieces on top to interact and exchange information with the software. Game piece with movable arms and legs would become another lovely feature. Software development allows the user to enjoy the game with others or AI system. Indeed, multi-players gaming is possible. It also has the benefit that the system is not limited to the game type. It can also allows taking commands thru internet upon future development. With these unlimited possibilities, our system becomes more attractive and interesting.