#/##/##

EML4551-2

Team 301: SoutheastCon Competition

Author1 Name: First M. Last;Author2 Name: First M. Last;Author3 Name: First M. Last;Author4 Name: First M. Last;Author5 Name: First M. Last

FAMU-FSU College of Engineering 2525 Pottsdamer St. Tallahassee, FL. 32310

Abstract

The IEEE SoutheastCon Student Hardware Competition is a yearly robotic design contest. This year's game is to design a robot, that drives itself, to do 1 of 2 things or both, within a 3-minute time limit. The 1st game is to use 10 push buttons that are along one of the walls of the field to type out as many digits (0-9) of the number PI (3.14159...) as possible. The 2nd game is to stack Lego Duplo blocks of different colors in order, within the scoring area. There are 10 different block colors and each color represents a different digit (0-9). The goal is to stack as many blocks as possible in the order of PI.

Our robot follows a planned route to gather the Duplo blocks and stacks them as it goes around the field. Toward the end of the 3-minute time limit, the robot moves over to the goal to place the stack of Duplo blocks. The robot carries the Duplo blocks by having two claws, one on the bottom and one on the top. The claw at the bottom lines up the Duplo blocks so the top claw can pick it up with no problems. Then the top claw moves up and holds the Duplo block in place so it can stack with the new incoming Duplo block.

Keywords: list 3 to 5 keywords that describe your project.

Disclaimer

Your sponsor may require a disclaimer on the report. Especially if it is a government sponsored project or confidential project. If a disclaimer is not required delete this section.

Acknowledgement

These remarks thanks those that helped you complete your senior design project. Especially those who have sponsored the project, provided mentorship advice, and materials. 4

- Paragraph 1 thank sponsor!
- Paragraph 2 thank advisors.
- Paragraph 3 thank those that provided you materials and resources.
- Paragraph 4 thank anyone else who helped you.

Table of Contents

List of Tables

List of Figures

Notation

Chapter One: EML 4551C

1.1 Project Scope

Project Brief

We will be competing in the 2020 Southeast Con hardware competition. The competition will consist of 2 sub-competitions. The goal of the first challenge in the competition is to accurately stack Lego Duplo blocks representing the digits of pi. Each digit from 0 – 9 is given a color representation (e.g. 3 is represented by an orange block). The goal of the second challenge is to push buttons in an order that represents the digits of pi. Both challenges must be completed by a completely autonomous device that fits into a 1 ft^3 space. Other robots, such as the robot showcased in the video by TKJ electronics, can recognize the color of and stack Duplo blocks, but it is not mobile, nor does it fit into the required volume (Electronics, 2017).

Description

The product will be an autonomous robot with the capabilities of completing at least one of the two challenges set for the 2020 SoutheastCon hardware competition.

Key Goals

The main goal of this project is to build an autonomous robot, or robots, that meet the specifications and can score as many points as possible. Another key goal is that the robot can score by pushing the buttons in the correct order, or stack 2x2 Duplo colored bricks in the correct sequence. Lastly the robot(s) can traverse the playing field.

Primary and Secondary Markets

The primary market will be the judges at the competition, where our product will compete. The secondary market will include potential future employers, and other schools who will see our robot competing.

Stakeholders

The primary investor for this project is our sponsor, the FAMU-FSU College of Engineering. Our professors, Dr. Hooker and Dr. McConomy are also stakeholders in the project, as they will be providing our final grade. Dr. Harvey, our academic adviser, will provide us with valuable knowledge from past experiences with the project.

Assumptions

We must assume that the measurements as well as the rules given to us in the rules will be accurate so we can build a robot that works for those specifications. We are assuming that the Duplo blocks that we purchase (for practice) will be the same as those used in the competition. We will also assume a certain brightness of lighting so that cameras to be able to accurately see the buttons or Legos.

1.2 Customer Needs

Because of the nature of the project, it was difficult to define who the customers were. The conclusion was that our stakeholders, those invested in our project, and the rules given to us would be our primary source of customer needs. We have also planned to communicate with those in charge of the competition for clarification of rules and guidelines. The judges in the competition may also be a potential customer, but given the time constraints, their responses could not be currently included.

Rules Breakdown

The goal of the IEEE SoutheastCon 2020 Hardware competition is to design a robot that stacks a series of Lego Duplo blocks, and/or inputs as many digits on a set of 10 pushbuttons in a three-minute competition. These challenges will take place in an arena like the one depicted on fig 1.

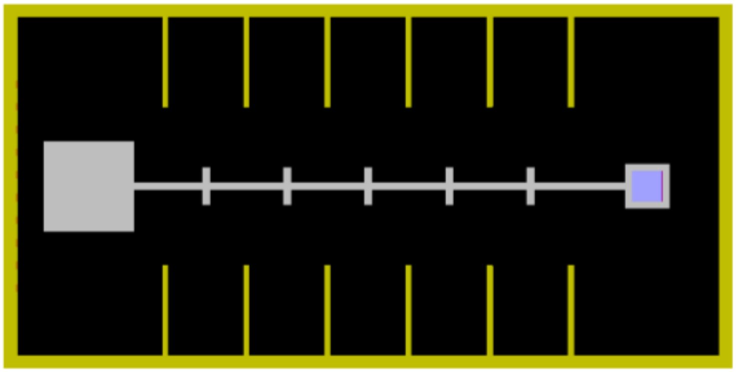

Figure 1. IEEE SoutheastCon 2020 Hardware Competition Arena (Radford, 2019)

Robot: Is completely autonomous and fits entirely within a 12”x12”x12” cube at start of match. The robot is required to stay within the arena. The robot may expand beyond initial size and may be split in several robots. Independent robots ought to communicate with each other over a wired link. A clear and visible labeled start switch is required for all robots. The robot cannot contain any explosives, pyrotechnic, toxic or corrosive materials, or flammable liquid gases. The robot cannot pose any danger to the arena, judges or spectators. Flying robots are not allowed.

Challenges:

Stacking Pi Digits: Stack as many digits of Pi within the competition match time. Points are awarded by total height of stack and how many digits of pi are in the correct order, starting from bottom to top. A different Lego Duplo block will represent a different digit using the following protocol:

Digit	Resistor color	Official Lego color
0	Black	Black
1	Brown	Reddish Brown
2	Red	Red
3	Orange	Orange
4	Yellow	Yellow
5	Green	Bright Green
6	Blue	Blue
7	Violet	Lavender
8	Gray	Dark Gray
9	White	White

Figure 2. Standard Resistor-Lego Block (Radford, 2019)

The Lego blocks must be stacked within a 5”x5” hole at the opposite the robot’s starting position. Each team may provide a base. If base is provided in advance, it may not contain motors, LEDs, processor, beacons or switches. Teams are responsible for stocking the ten bins with 10 blocks. Lego blocks may be arranged in any fashion inside the bins, however only on Lego block color is allowed per bin. A maximum of 15 blocks of each color is available per team.

The score will be based on the height of the stack after the end of the match. The robot needs to release the stack once the time of the match finishes. If more than one stack exists, the highest one will be used.

Pressing Buttons: Enter as many digits of Pi on the 10 pushbuttons within competition time. Points are awarded by total number of buttons presses as well as how many digits of pi are entered in the correct order. At start of competition, pushbutton 3 will illuminate, and then number 1, and so on in the order of digits of Pi. Each button must be pressed for at least 50 ms, and between each button press at least 100 ms should pass. Teams will receive smaller points for any button presses after the last valid button press.

Competition: The competition will consist of two preliminary runs. The top four highest scoring teams in preliminary runs will then compete in final run. The final round will only consider points scored in that final round. Each match will last for a maximum of 3 minutes.

Scoring**:** The points awarded by category are depicted in the following figure:

Description	Number of points
Total stack sequenced correctly	20 * N * N
Additional stack not sequenced correctly	N * N
Total button presses sequenced correctly	10 * N
Additional button presses not sequenced correctly	N (max of 100 counted)

Figure 3. Points Awarded Per Category (Radford, 2019)

Survey Responses

In order to further define our customer needs, a survey was sent to our professors and adviser. Dr. Bruce Harvey emphasized the avoidance of 3D printing as a source of manufactured parts, as another team did in the past. Dr. McConomy's response was to be perfect, as the success of the product heavily relies on the robot(s)' accuracy. One of the questions asked in the survey was to rank possible functions of the robot. Dr. Harvey's and Dr. Hooker's responses to the question can be found in Table 1.

Customer Rank Order below.

Dr. Hooker, prior to completing the survey, advised that we perform to the best of our ability. He emphasized the importance of planning to avoid redesigns in the future.

Table 1.
Customer Rank Order

	Dr. Harvey	Dr. Hooker
Mobility	1	6
Accurate Stacking	4	1
Accurate Button Pushing	5	2
Stacking	2	3
Button Pushing	3	4
Color Recognition	8	5
Speed	7	7
Automatic Shutdown	6	8

From the responses in Table 1, one of the more important customer needs is that the product can stack the Duplo blocks. Something important to note is that the rules were updated and the way the points are scored has drastically changed. Even though in the survey responses button pushing buttons tops the priority list along with mobility, the rule change has drastically reduced the points we get from pushing buttons so we will have to put a lot less weight into it. Regardless of this, seeing the responses from our advisors has helped us get a sense of what is expected of us so we can have a more guided and informed approach as to what to prioritize in the robot.

1.3 Functional Decomposition

Functional Decomposition is used to break down a system into smaller parts. Often, the system can be complex, and in order to approach the solution with ease, it is necessary to pull the system apart. In the cross-reference table below, the Robot is broken down into its main functions. Through the flow chart depicted in Figure 15, the major functions were then broken down into minor functions.

After collecting data from the survey and speaking with our advisors, the customer needs gathered were converted into functions and placed into both a function hierarchy chart and a cross reference table.

Connection to Systems

The most important system in our product, according to the cross-reference table is the navigation system. The navigation system plays a role in all the functions required for a successful product. The recognition system follows the navigation system in order of importance. The least important system was the push system, because it provides little reward and only applies to one function. All of the different components that will make up the systems, mechanical, software, and electrical, will be intertwined, so no component is more important than the other. The mechanical system will play the largest role in the stacking mechanism.

For the power source for the robot we will be using a battery with enough power to supply every component in the robot and a factor of safety to prevent overloading.

We will choose what sensors to use depending on what task we choose to have our robot do:

Lego Stacking:

- Color sensor: senses the color of the Lego blocks
- Sonar sensor: to be able to tell where we are in the field and where the obstacles/Legos/stacking place is

Button Pushing:

- Light sensor: being able to tell what buttons are lit so we can press them in order
- Would also require a sonar sensor to move and see where the buttons are

The software will interrelate all the before mentioned points, as it will govern the following modules:

- Pushing Buttons: This module will control lighting sensors and mechanical equipment which will push the buttons.
- Stack: This module will control sonar sensors, to verify closeness to Legos, and mechanical component used to stack Legos.
- Navigation: This module will control sonar sensors, motor, and steering.

The "push" system is the least important system, because it is only involved in the button pushing mechanism. In the design of the product, the system may be removed altogether, if there are conflicts in the design. The "navigation" system involves the driving of the product around the field, as well as finding the blocks in the bins and avoiding barriers or other obstacles.

Action and Outcome

The goal of the IEEE SoutheastCon 2020 Hardware competition is to design a robot that stacks a series of Lego Duplo blocks, and/or inputs as many digits on a set of 10 pushbuttons in a three-minute competition. This product will be a robot that has all the characteristics necessary to do the challenges of the competition.

1.4 Target Summary

When assessing what our robot could do it is important to consider the best-case scenario as well as the worst-case scenario to have a ballpark estimate of where we could be placing. Considering the bare minimum, we would want our robot to do we can divide it into two categories, a Lego stacking robot and a button pressing robot. For the robot pressing button the bare minimum we would want is a robot that can push one button once every second even if the button is not in the order of pi. However, the rules limit incorrect button presses to 100 so the robot should be able to score 100 points in the competition. If that is what is expected of our robot then it should be able to move to from the starting position to the buttons in less than 20 seconds, it should then align itself with one of the buttons in less than 10 seconds, and after that it should press the same button once every second to leave breathing room for the 150 ms required between button pushes(50ms length of button being pushed and 100ms between pushes). Now considering the minimum required for the Lego stacking robot we would want it to score at least as many points as the push button robot so we would set the minimum at 100 points. The best way to achieve this would be by stacking 15 blocks out of order. We could do this by having 15 Legos of the same color already stacked before the competition began and just having the robot pick it up and take it to the designated drop off area. To do this we would need a robot tall enough to be able to carry the stack of 15 Legos. The robot would also be able to drop it off carefully as to not have the stack topple over. Since it would have 3 minutes to complete this task the speed it does it at would not be super important, the most important thing would be to have it do it effectively as in that it can pick up the whole stack and move with it without disturbing it. Now considering the maximum attainable goals lets again start with push buttons. For the push buttons we would need to press the buttons of pi in the correct order as quickly as possible. Assuming that the average time it would take the robot to move from one button to another would be 1 second and if the robot has to move, we can neglect the 100 milliseconds between button pushes and just account for the 50 milliseconds required to push the button. This would give us a time of 1.05 seconds between button pushes and given that we have 180 seconds and assuming it takes us 5 seconds to go from the starting point to the buttons and subtracting 10 percent since this ideal case will never happen we get 150 button presses in correct order. With this we would get 1500 points from the button pressing. Now considering the best-case scenario for the Lego stacking robot we would have the robot stack as many Legos as it could in the correct order. Assuming we have the base placed before time began and also assuming that it takes the robot an average time of 5 seconds to go from the Lego bins to the drop off station and it takes it an average of 2 seconds to stack the Legos as well as an average time of 5 seconds to go from the drop off station to the correct Lego bin we are looking at an average time of 12 seconds between stacking blocks. Considering we have 180 seconds this means that we would be able to stack 15 blocks in correct order in this time. However once again taking away 10 percent from it we would get roughly 13 blocks in correct order which would give us 3380 points. This robot would need to be mobile since it is very time sensitive and it should be able to correctly identify where the correct Legos are either with color recognition or hard coding the locations of the Legos of each color. It should also be able to stack Legos at a height of around 13 blocks. Considering this information, we can get a rough estimate of how many points we can get and set realistic goals for ourselves.

Method of Validation

The mission critical functions include to navigate the playfield, recognize where the Duplo blocks are at, pick the Duplo blocks and stack them correctly at the base. For this project a replica arena or playfield will be built at the FAMU-FSU college of engineering, and a set of identical Duplo blocks will be purchased. Using these before mentioned elements, several tests will be made in order to assess the successfulness of the design. Once the robot is built, it will be tested on the arena for:

Navigation: The robot will need to pass successfully different checks such as navigating correctly to the base, and to each of the bins. These will be done by selecting different path combinations to the bins and base. For example, one of the possible combinations could be, the robot starts at initial position and then will first go to bin numbered 1, to number 3, and finally to the base. To deem the navigation effective, the robot must do at least 10 consecutive different path combinations without a problem.

Recognition: A set of tests will be made to check that the robot effectively recognizes where the Duplo blocks are. For this the robot will be assigned to go to a bin having Duplo blocks and will need to stop once it encounters blocks at a "pick up" distance. To ensure the recognition works correctly the robot will need to pass 10 consecutive tests without a problem, these tests will be done on random bins and the placement of the Duplo blocks will be aleatory.

Pick up: A set of tests will be made to check that the robot effectively picks up the Duplo Blocks. The robot will have several blocks laying at a "pick up" distance and it will need to pick them up successfully, this implies that the robot must pick up the blocks and not let them go unless specified. For these tests the blocks will be in two combinations, alone or as stacked towers. To deem the *pickup* efficacious the robot will need to pass 10 consecutive tests without a problem.

Stacking: A set of tests will be made to check that the robot effectively stacks the Duplo Blocks. The robot will already have block or blocks which have already been picked up and will need to stack these effectively at the base. To deem the stacking efficacious the robot will need to pass 10 consecutive tests without a problem.

Discussion of Measurements

Most of the measurements will be measured using sensors, so the received voltage will need to be converted to the metric used for each function. The “navigate the play field” function is broad and may require some additional metrics to fully define it. Our team may also use the painted lines in the arena to assist in the navigation from bin to bin, which would require additional sensors. The navigation can be measured with ultrasonic, photoelectric, or infrared sensors as well as predefined paths in the software. The “avoid barriers” function will probably use the same sensors as the “navigate the play field” function. The main objective of this function is to prevent the robot’s collision with the borders of the playing field. This function will be measured by the distance from the closest barrier. Both the “Differentiate between buttons” and the “Differentiate between blocks” functions are measured by the time to reach the appropriate location. They will still need additional measurements to verify the location. The correct button, for instance, will illuminate so the robot may need an additional sensor to identify the correct button. Lifting and lowering the block will be measured by the height difference between the current block and the previous block placed. This will potentially be measured through calculating the movements internally, rather than an outside reference.

Summary

The final design selected must achieve all the targets listed in Table 9 below. The metric and target for the navigate the play field function was deemed critical because the “navigate” function is the most important function for our product. The other functions are still necessary for complete success, but without navigation, no points can be scored. If another function, such as finding the blocks, were to fail points could still be attained through pushing buttons. The motion of the robots is a function not explicitly listed in the targets and metrics, but falls under the navigate system.

1.5 Concept Generation

The morphological chart was used to generate fifty of the one hundred possible solutions to the problem Other methods, such as biomimicry and the "anti-problem" were used to generate the other fifty concepts. The primary methods used were individual brainstorming, group brainstorming, and biomimicry.

Concept 1.

The robot has a sorter on its body with blocks that slide in. The robot will first grab the block using an elevator and claw. A gate opens releasing the next color necessary in the stacking sequence. The robot will sit on 4-wheels and will be rear wheel drive. The two wheels in the front will be used for steering, guided by a servo motor. Once the gate opens, the block will slide to a point at which the arm can lift and place it in the correct stack.

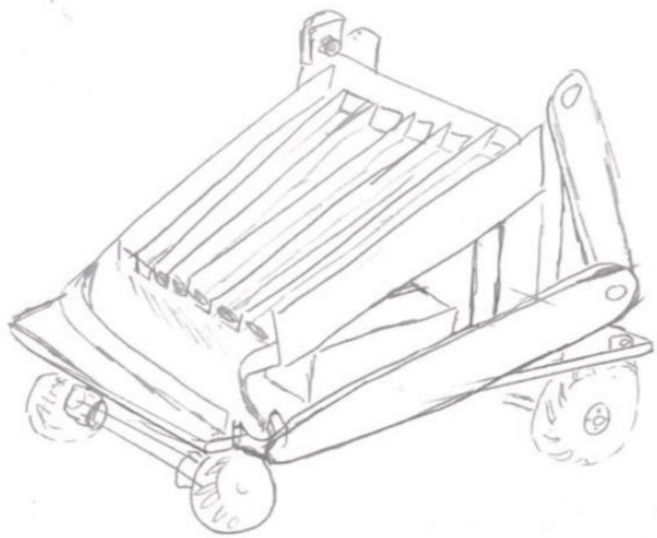

Figure 4. The color sorting robot

Concept 2.

The following concept, termed “the scorpion,” used biomimicry to develop the general idea. The scorpion’s tail was mimicked in its design. Fully imitating the scorpion’s tail would require a large number of motors, so the design was limited to 4 motors in the “tail.” The robot will still sit on 4 wheels, with the rear wheels driving the robot and the front wheels steering. Each rear wheel will have an individual motor for increased mobility.

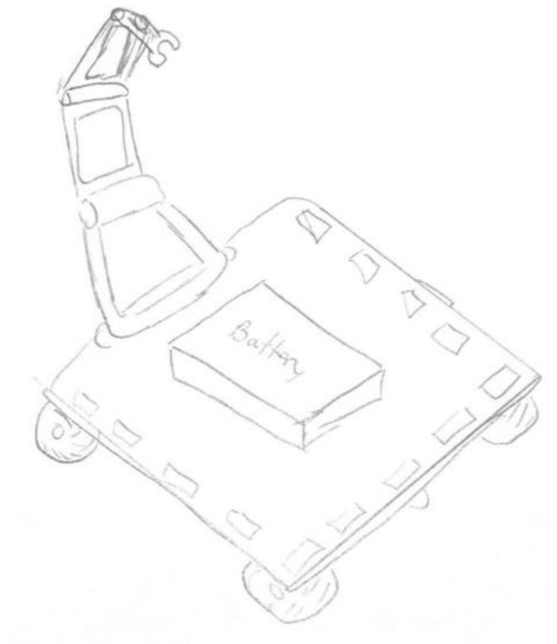

Figure 5. The scorpion

Concept 3.

Robot to stack Legos that has 2 arms to hold the Lego and stack it and has a hammerlike appendage on top to make sure the Legos stick together.

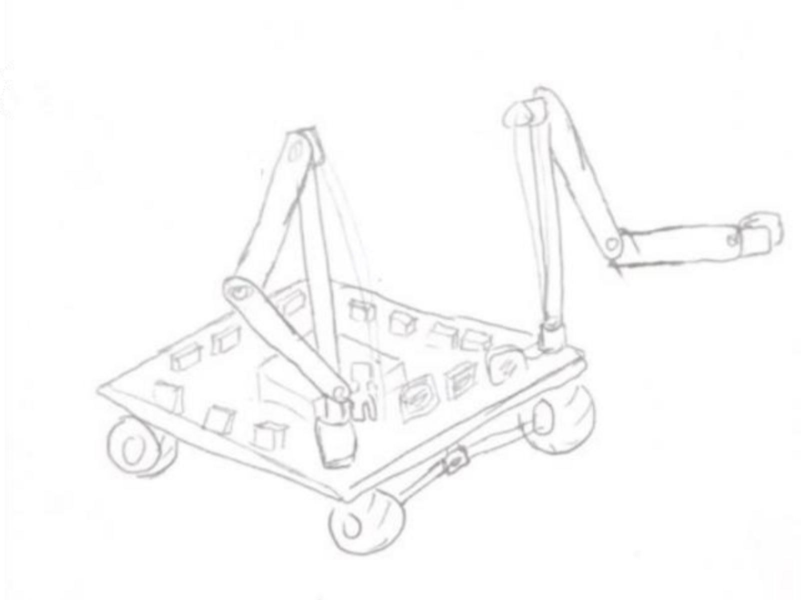

Figure 6. Robot with two arms.

Concept 4.

The robot uses an elevator and two claws to raise and lower blocks. It will use infrared sensors to navigate the playfield and avoid the barriers. The navigation will be supplemented using line following. It will find blocks using sonar and will differentiate between the blocks' colors and the buttons using a camera. It will have a 4-wheeled base and will be programmed using C++. The robot has a lift kit, to maximize its height.

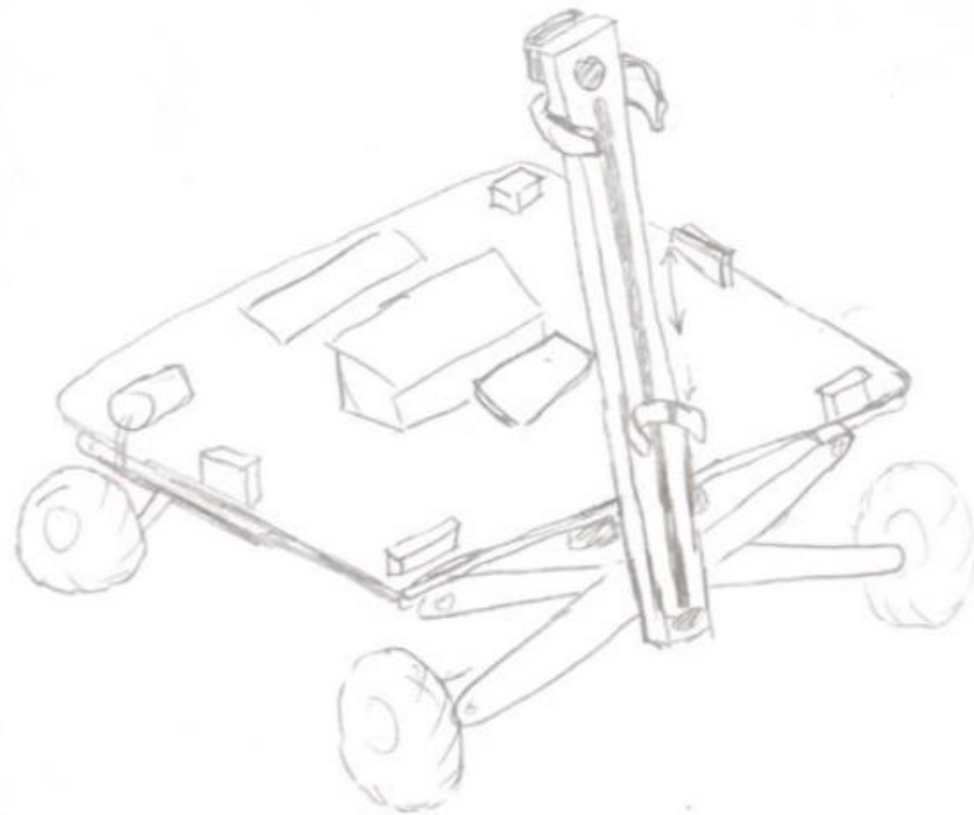

Figure 7. Robot with two claws and lift kit.

Concept 5.

The robot a slide system on either side to collect the blocks we need and drag them along with the robot back to the base of the stack. It will use a claw on a track to move around grab the next block in the sequence and stack it. It will use infrared sensors to navigate the playfield and avoid the barriers. The navigation will be supplemented using line following. It will find blocks using sonar and infrared sensors. It will have a 4-wheeled base and will be programmed using C++.

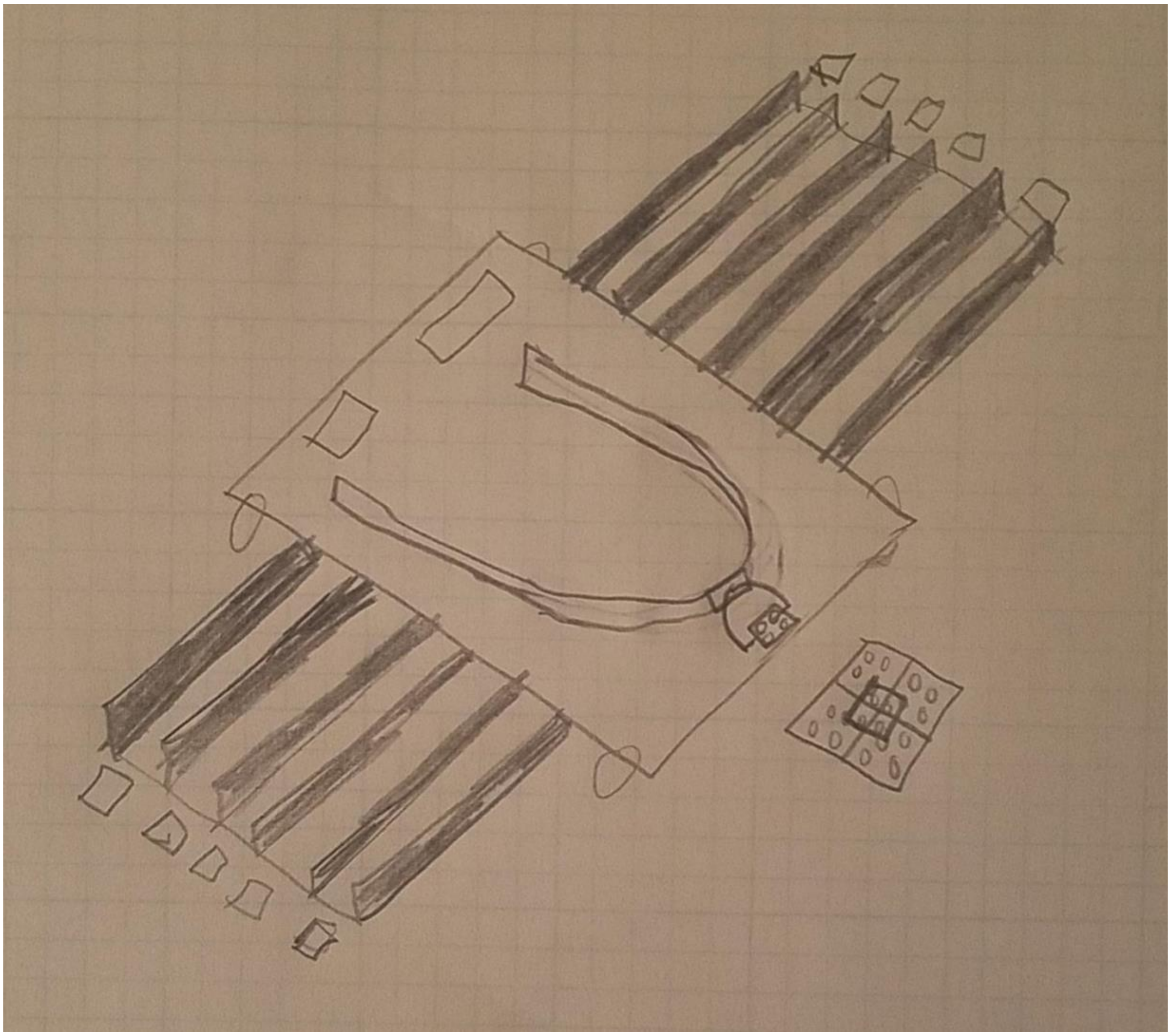

Figure 8. Robot with slide system

Concept 6.

The robot uses a spring-based elevator in 10 individual hoppers on the robot to store and supply bricks. It will have a claw track to grab the individual blocks and bring it over to the base for stacking. It will use infrared sensors to navigate the playfield and avoid the barriers. The navigation will be supplemented using line following. It will find blocks using sonar and infrared sensors. It will have a 4-wheeled base and will be programmed using C++.

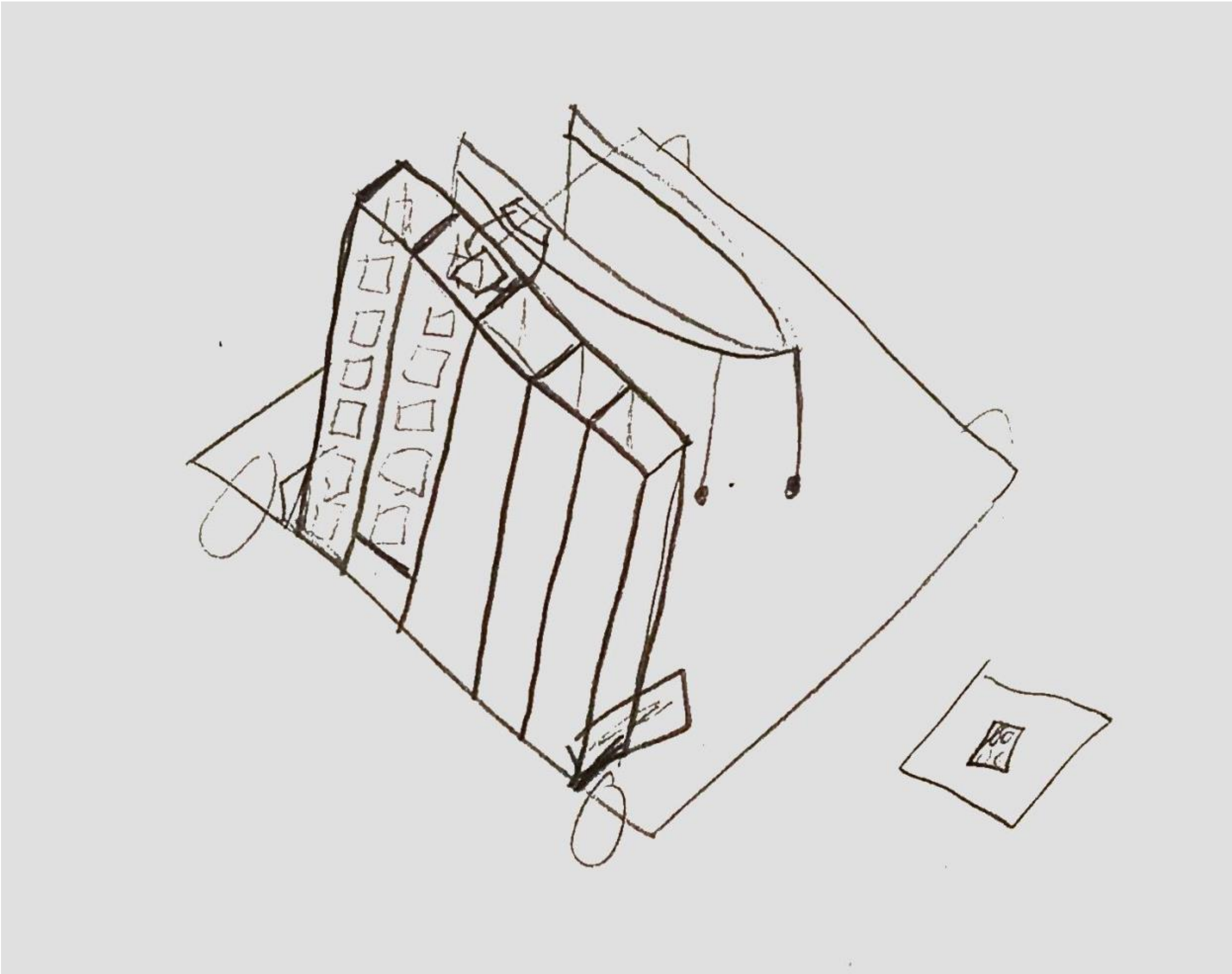

Figure 9. Spring elevator robot

Concept 7.

The concept of the stingray originated from the need to get the most amount of blocks to the base in an organized fashion in the most efficient way. The stingray is composed of a net-like arms which are used to drag the Legos from the bins in an organized fashion all the way to the base, as depicted on figure 1 and 2.

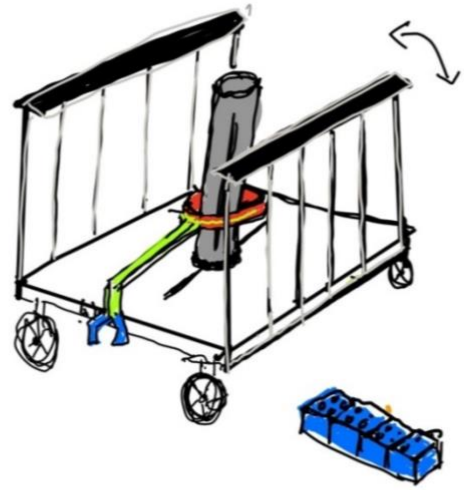

Figure 10. Stingray design (sketch) - arms folded

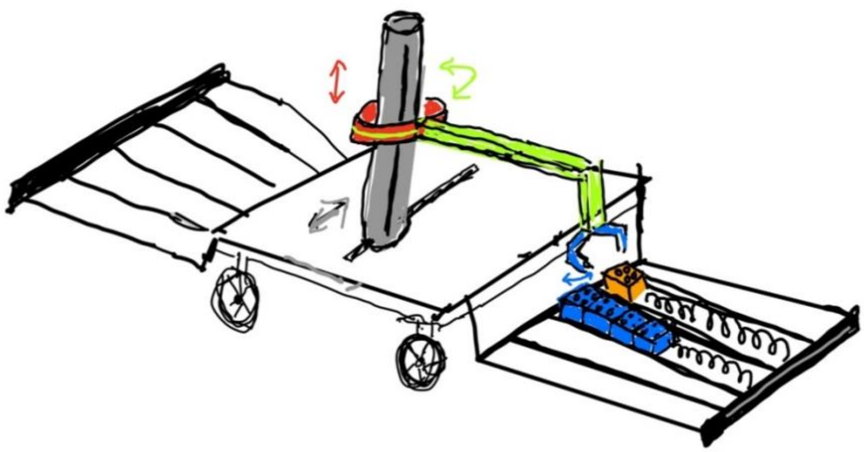

Figure 11. Stingray design (sketch) – arms unfolded

1.6 Concept Selection

Function and customer needs were used to evaluate the ideas generated in the concept generation phase. From the 100 generated ideas, the eight most relevant concepts were considered. The evaluation process was done by using several concept selection tools such as house of quality table, Pugh matrices, and the analytical hierarchy process. By using all of these it was possible to make a concept selection.

House of Quality

The house of quality is a design tool of function arrangement. It identifies and classifies customer needs and identifies their relevance, it also identifies the engineering characteristics required for those needs and correlates the two. This tool was used to evaluate the six customer needs present in this project, acceleration, distance from barrier, block height, time to reach correct bin, end time behavior, and time to locate block within bin. By studying these it was possible to assign a numerical relevance of each of these needs to the project.

The results obtained after doing the house of quality are shown on table 1. From the obtained results it can be observed that the block height customer need will play the most relevant role in the project. This need has a relative weight of 21%, it was followed close behind by the needs, distance from barrier, time to reach correct bin, and end time behavior all with a 17% relative weight. The needs considered less relevant according to the house of quality are time to locate block within bin and acceleration with 15% and 13% respectively. A binary pairwise comparison table was used to evaluate the customer needs, Stack duplo blocks correctly, mobility, robot volumes, color recognition, speed, automatic shutdown, and button pushing.

Table 2. Binary Pairwise Comparison

Calculate Weight Factor of Customer Requirements								
Customer Requirements	1	2	3	4	5	6	7	Total
Stack Duplo Blocks Correctly	-	1	0	1	1	1	1	5

Mobility	0	-	0	1	1	1	1	4
Robot Volume	1	1	-	1	1	1	1	6
Color Recognition	0	0	0	-	0	1	1	2
Speed	0	0	0	1	-	1	1	3
Automatic Shutdown	0	0	0	0	0	-	1	1
Button Pushing	0	0	0	0	0	0	-	0

It can be observed that the customer most relevant according to the binary pairwise comparison is the robot volume, with a total of 6 points. Robot volume plays an important role as more volume will allow more functionality to be added to the robot. The maximum value is fixed by the competition rules, if the volume of the robot is more than that specified by the competition, the team will automatically be disqualified. The second most important need is to stack the duplo blocks correctly, with a total of 5 points. Stacking blocks correctly is the method that will allow for the largest amount of points to be scored in the least amount of time. The goal of the project is to score the most amount of points within the given time, so it is logical for this need to play an important role in the project. This need is followed by mobility, speed, automatic shutdown, and button pushing, with total points 4,3,2,1, and 0 respectively.

Pugh Matrix

We used a Pugh matrix as another form of concept selection in which we took all of the engineering characteristics and add weights to the ones we consider more important than others to be able to make a ranking to see which concept would be the best to pick given their total score in the criteria. This can be seen in table 3 as well as table 4.

Table 2. Pugh Matrix 1

Engineering Characteristics	Datum	Weights	Scorpion	Robot w/ two arms	Claws and lift kit	Slide System	Spring based elevator	Stingray
Acceleration	0	1	1	1	1	1	0	-1
Distance from Barrier	0	1	1	1	1	0	0	1
Block Height	0	2	0	1	0	0	0	1
Time to reach correct bin	0	1	-1	-1	0	1	-1	0
End time behavior	0	1	1	1	1	1	0	1
Time to locate block	0	1	0	-1	0	1	0	1
Total Score	-	-	2	3	3	4	-1	4
Rank	-	-	5	3	3	1	6	1

Table 3. Pugh Matrix 2

Engineering Characteristics	Weights	Datum (Scorpion)	Robot with two arms	Slide System	Claws and lift kit	Stingray
Acceleration	1	0	-1	-1	-1	-1
Distance from barrier	1	0	0	-1	0	0
Block height	2	0	1	0	1	1
Time to reach correct bin	1	0	0	1	1	1
End time behavior	1	0	0	1	0	1
Time to locate blocks	1	0	1	1	1	1
Total	-	-	1	1	3	4
Rank	-	-	3	3	2	1

We set the baseline to be zero and if it is better than the baseline we give it a one and if it is worse than the baseline we give it a negative one however if it is relatively similar to the baseline we will give it a zero. From our engineering characteristics we gave the most weight to block height since this determines the maximum size of our Lego stack and having the most amount of Legos stacked will be important considering our robot is limited in size and since we will most likely be stacking the Lego in pi order every additional block will give us a considerable amount more points, for example 10 blocks is 2000 points while 11 blocks is 2420 points giving us a difference of 420 points showing the importance one block makes. From the first Pugh matrix we got that the stingray and the slide system design were tied for the best design. To remedy this, we made a second Pugh matrix having the scorpion design as our baseline and comparing the other designs to it to have another point of reference. Applying the same rules as the first Pugh matrix we came with the results that the stingray was better than the slide system finally declaring the stingray design as the best option according to the Pugh matrices we made.

Analytical Hierarchy Process

For the last analysis method, we used an AHP (analytical hierarchy process) to determine which idea to select. These can be seen from Table 4, the weights we calculated can be seen in Table 5 and the final values for each design in Table 6. For the AHP we put the concepts in the first column as well as a datum to have as a reference point and we put the criteria on the first row. After this we arbitrarily rated how good we thought a given design would be at given criteria. Using the rules given to us to make an AHP we took the reciprocal and made weights for each category. After that we added the sum of the results with the weights to get the final results. This method has its advantages in that it is a more mathematically rigorous method using principles of linear algebra and has more weights than the ones used in the Pugh matrix.

Table 4.

Final AHP Matrix

Final Matrix Transposed						
	Acceleration	Distance from Barrier	Block height (raising and lowering)	Time to Reach Correct Bin (Color Determination)	End Time behavior	Time to locate block within bin
Datum - Color Sorting	0.047	0.091	0.102	0.249	0.106	0.069
Scorpion	0.304	0.284	0.117	0.059	0.199	0.072
Robot with two arms	0.161	0.146	0.091	0.044	0.206	0.238
Claws and lift kit	0.259	0.213	0.318	0.209	0.118	0.074
Slide system	0.061	0.041	0.058	0.116	0.181	0.117
Spring-based elevator	0.116	0.103	0.151	0.104	0.038	0.076
Stingray	0.052	0.123	0.163	0.219	0.153	0.355

Table 5.

Criteria Weights

Criteria Weights
0.157
0.054
0.176
0.211
0.065
0.337

Table 6.

Concept and Alternative Value

Concept	Alternative Value
Datum - Color Sorting	0.113
Scorpion	0.133
Robot with two arms	0.152
Claws and lift kit	0.185
Slide system	0.098
Spring-based elevator	0.100
Stingray	0.219

However, we did choose arbitrary numbers since we are only speculating how the designs will perform in a given task compared to the other designs. After completing all these steps for each design we ended up with the values on table 7 which indicate that given the values we chose the best design is the stingray. Following the completion of all the steps for each design we ended up with the values on table 7 which indicate that given the values we chose the best design is the stingray.

Final Concept

After evaluating the results obtained from the analytical hierarchy process, the stingray design is the concept that will be selected for the project. The results obtained for the stingray on the AHP final matrix, alongside the criteria comparison weight, and the concept and alternative value tables (Table 4, Table 5, andTable 6 respectively), are the highest obtained results within each of the tables. The concept of the stingray originated from the need to get the most amount of blocks to the base in an organized fashion in the most efficient way. The stingray is composed of a net-like arms which are used to drag the Legos from the bins in an organized fashion all the way to the base, as depicted in Figure 12 and Figure 13.

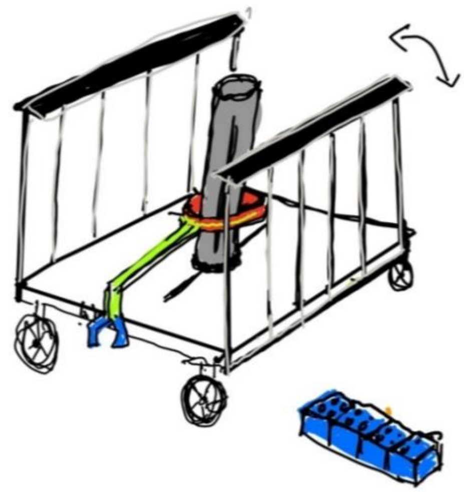

Figure 12. Stingray design (sketch) - arms folded

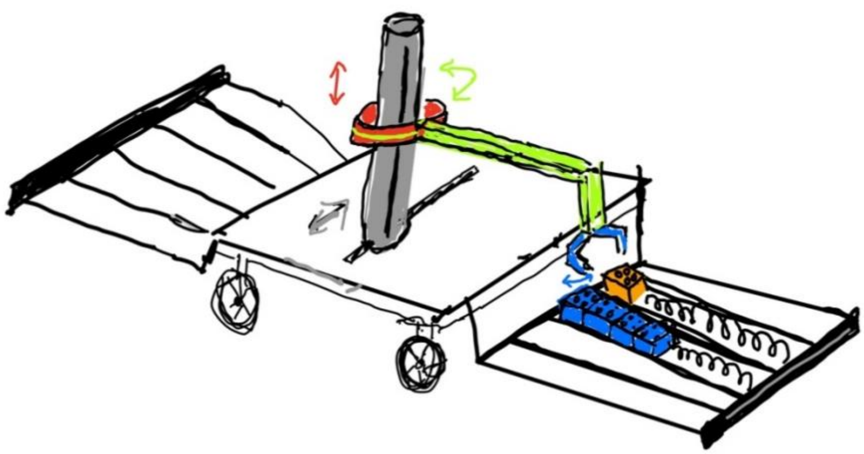

Figure 13. Stingray design (sketch) - arms unfolded

These arms will eliminate the need for a camera-based recognition system that will determine the color of the blocks, as the design will know the color of blocks due to their position within the arm. The design of the arms will also eliminate the need for the robot to travel back and forth from the base to the bins to get more Lego blocks of a single color. The robot will only need to go once to each bin, and from there it will collect all the necessary blocks for the stack within that bin. From there an arm, composed of a lift and a claw that moves 360 degrees pick the Lego Blocks and stacks them correctly on the base.

1.8 Spring Project Plan

The following Gantt chart shows our team's plan for the spring semester. By the end of this fall semester, we aim to have a functioning line following robot so we can have the frame and basic path programming by the end of the semester. Considering we have the replica of the arena that will be used in the competition, we will have an accurate representation of how our line following will roughly be by the end of the fall semester.

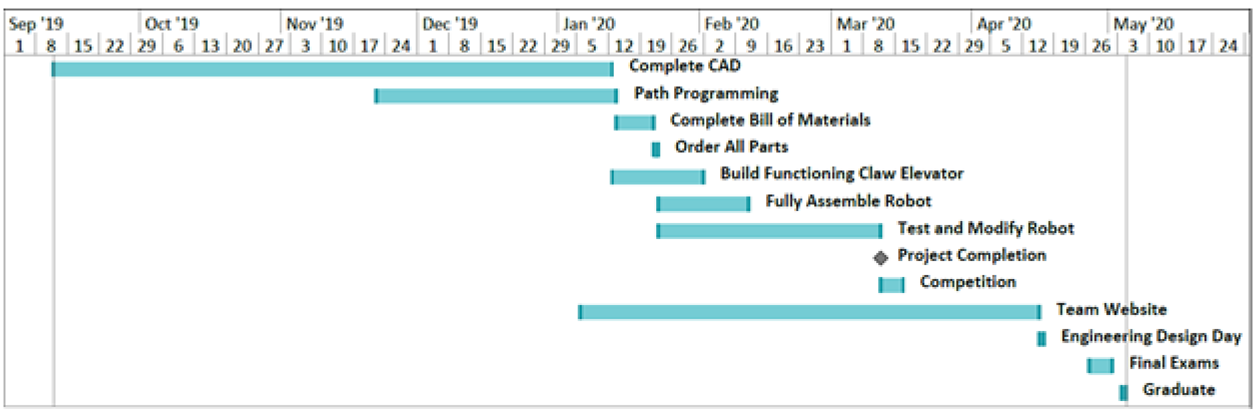

Figure 14. Gantt chart.

By the start of the spring semester we want to have a complete CAD of the entire robot so we can have an idea of how everything will fit and how we can assemble it. Also, near the beginning of the spring semester we want to have the path programming down so we can focus on the rest of the robot. Even though it might seem like having the line following is most of the path programming that is not the case since we need to move the robot out of the line to pick up the Legos and from there back to the line and follow the sequence of Legos we want so we will have to work on all that to make a successful robot that can follow the path it needs to. By the beginning of February, we want to have a functioning claw elevator since that is the most important part of the robot along with the path following. By February 19 we want to put all this together and have a fully assembled robot since we want to have some time to be able to work on it and make any fixes that will be necessary since it is almost guaranteed that something will go wrong or we will make some changes to optimize the robot. By the end of February, we want to have the project complete meaning that we want the robot to be fully functioning and complete the goals we set for it. This will give us two weeks to make any adjustments we need for the competition. March 14th is the competition so by that point we will have to hope that our efforts of the previous months will give us a good result. After the competition the bulk of our work is done, and we only need to concern ourselves with finishing the assignments for senior design. After that we concern ourselves with graduating in May 2nd.

Table 7.
Spring Milestones

Activity	Expected Completion Date
Completed CAD	January 12, 2020
Path Programming	January 13, 2020
Completed Bill of Materials	January 21, 2020

All parts ordered	January 22, 2020
Functioning Claw Elevator	February 1, 2020
Fully Assembled Robot	February 11, 2020
Testing	March 11, 2020
Project Completed	March 11, 2020
Competition	March 16, 2020
Team Website Completion	April 15, 2020
Engineering Design Day	April 16, 2020
Final Exams	May 1, 2020
Graduation	May 2, 2020

Chapter Two: EML 4552C

2.1 Spring Plan

Project Plan.

Build Plan.

Appendices

FAMU/FSU College of Engineering

Department of Electrical and Computer Engineering

Code of Conduct

Team 301 (IEEE SoutheastCon Student Hardware Competition)

Names:
Isabel Barnola
David Bowen
Diego Campos
Alex Ndekeng
Abiel Souverain

Date 9/20/2019

Mission Statement

Team 301 is committed to ensuring a positive work environment that supports professionalism, integrity, respect, and trust. Every member of this team will be contributing full effort to the creation and maintenance of such an environment in order to bring out the best in all of us as well as this project.

Roles

Each team member is delegated a specific role based on their experience and skill sets and is responsible for all here-within:

Team members:

Team Leader: David Bowen

Manages the team as a whole; develops a plan and timeline for the project, delegates tasks among group members, and finalizes all documents and provides input on other positions where needed. The team leader is responsible for promoting synergy and increased teamwork. If a problem arises, the team leader will act in the best interest of the project. He is the point of contact between team members, advisors, and sponsors. The team leader takes the lead in organizing, planning, and setting up meetings. In addition, he is responsible for keeping a record of all correspondence between the group and 'minutes' for the meetings. Finally, he gives or facilitates presentations by individual team members and is responsible for overall project plans and progress.

Lead Power Electronics Engineer: Alex Ndekeng

The lead Power Electronics Engineer is responsible for putting together the various power electronic and controls of the robot; assembling the wiring for the parts provided by the other team members. This role will also involve testing the parts for the desired electrical outputs.

Lead Software Engineer: Isabel Barnola

The lead software engineer is responsible to design the underlaying architecture of the software that will control the robot, and to oversee all coding done by any other team member. Ensures that completed product controls the robot.

Lead Design Engineer: Abiel Souverain

The lead design Engineer is responsible for the 3D CAD models of the product and the drawings of the cad models. Ensures completed product can withstand necessary forces. They will possibly provide an analysis in ProE.

Lead Signal Process Engineer: Diego Campos

The lead signal engineer will be responsible for making sure that every interconnected device in the robot such as sensors and cameras will successfully communicate with one another in an efficient manner.

Lead Robotics Engineer: David Bowen

The lead robotics engineer is responsible for the assembly of the robot and ensuring that all the parts (electrical and mechanical) mesh and work together. They will also be accountable for leading the testing of the robot with the lead software engineer.

All Team Members:

- Work on certain tasks of the project
- Buys into the project goals and success
- Delivers on commitments
- Adopt team spirit
- Listen and contribute constructively (feedback)
- Be effective in trying to get message across
- Be open minded to other's ideas
- Respect other's roles and ideas
- Be ambassador to the outside world in own tasks

Communication

The main form of communication will be basecamp among the group, as well as through regular meetings of the whole team. Email will be a secondary form of communication for issues not being time sensitive. Email will also be used for communication with our advisors and professors. For the passing of information, i.e. files and presentations, basecamp will be the main form of file transfer and proliferation.

Each group member must have a working basecamp and email for the purposes of communication and file transference. Members must check their basecamp and emails at least twice a day to check for important information and updates from the group. Members will be initially informed via the Outlook calendar, so it is very important that each group member checks their email frequently. Completed files will be uploaded to basecamp but working documents will be uploaded to Office 365.

If a meeting must be canceled, an email must be sent to the group at least 6 hours in advance. Any team member that cannot attend a meeting must give advance notice of 6 hours informing the group of his absence. Reason for absence will be appreciated but does not need to be specific. Meetings will be given a priority level from 1 – 3, with 3 being the highest priority (cannot miss). Repeated absences in violation with this agreement will not be tolerated.

Team Dynamics

The students will work as a team while allowing one another to feel free to make any suggestions or constructive criticisms without fear of being ridiculed and/or embarrassed. If any member on this team finds a task to be too difficult it is expected that the member should ask for help from the other teammates. If any member of the team feels they are not being respected or taken seriously, that member must bring it to the attention of the team for the issue to be resolved. We shall NOT let emotions dictate our actions. Everything done is for the benefit of the project and together everyone achieves more.

Ethics

Team members are required to be familiar with the NSPE Engineering Code of ethics as they are responsible for their obligations to the public, the client, the employer, and the profession. There will be stringent following of the NSPE Engineering Code of Ethics.

Dress Code

Team meetings and Adviser meetings will be held in casual attire or Power Ranger costumes. Sponsor meetings and group presentations will be business formal to formal as decided by the team per the event.

Weekly and biweekly Tasks

Team members will participate in all meetings with the sponsor, adviser and instructor. During said times ideas, project progress, budget, conflicts, timelines and due dates will be discussed. In addition, tasks will be delegated to team members during these meetings. Repeat absences will not be tolerated.

Decision Making

It is conducted by consensus and majority of the team members. Should ethical/moral reasons be cited for dissenting reason, then the ethics/morals shall be evaluated as a group and the majority will decide on the plan of action. Individuals with conflicts of interest should not participate in decision-making processes but do not need to announce said conflict. It is up to each individual to act ethically and for the interests of the group and the goals of the project. Achieving the goal of the project will be the top priority for each group member. Crucial decisions will be made by the entire team. Smaller decisions will be made by individuals or a subset of the team. Below are the steps to be followed for each decision-making process:

- Problem Definition – Define the problem and understand it. Discuss among the group.

- Tentative Solutions – Brainstorms possible solutions. Discuss among group most plausible.
- Data/History Gathering and Analyses – Gather necessary data required for implementing Tentative Solution. Re-evaluate Tentative Solution for plausibility and effectiveness.
- Design – Design the Tentative Solution product and construct it. Re-evaluate for plausibility and effectiveness.
- Test and Simulation/Observation – Test design for Tentative Solution and gather data. Re-evaluate for plausibility and effectiveness.
- Final Evaluation – Evaluate the testing phase and determine its level of success. Decide if design can be improved and if time/budget allows for it.

Conflict Resolution

In the event of discord amongst team members the following steps shall be respectfully employed:

- Communication of points of interest from both parties which may include demonstration of active listening by both parties through paraphrasing or other tool acknowledging clear understanding.
- Administration of a vote, if needed, favoring majority rule.
- Chess Game
- Team Leader intervention.
- Instructor will facilitate the resolution of conflicts.

Appendix B: Functional Decomposition

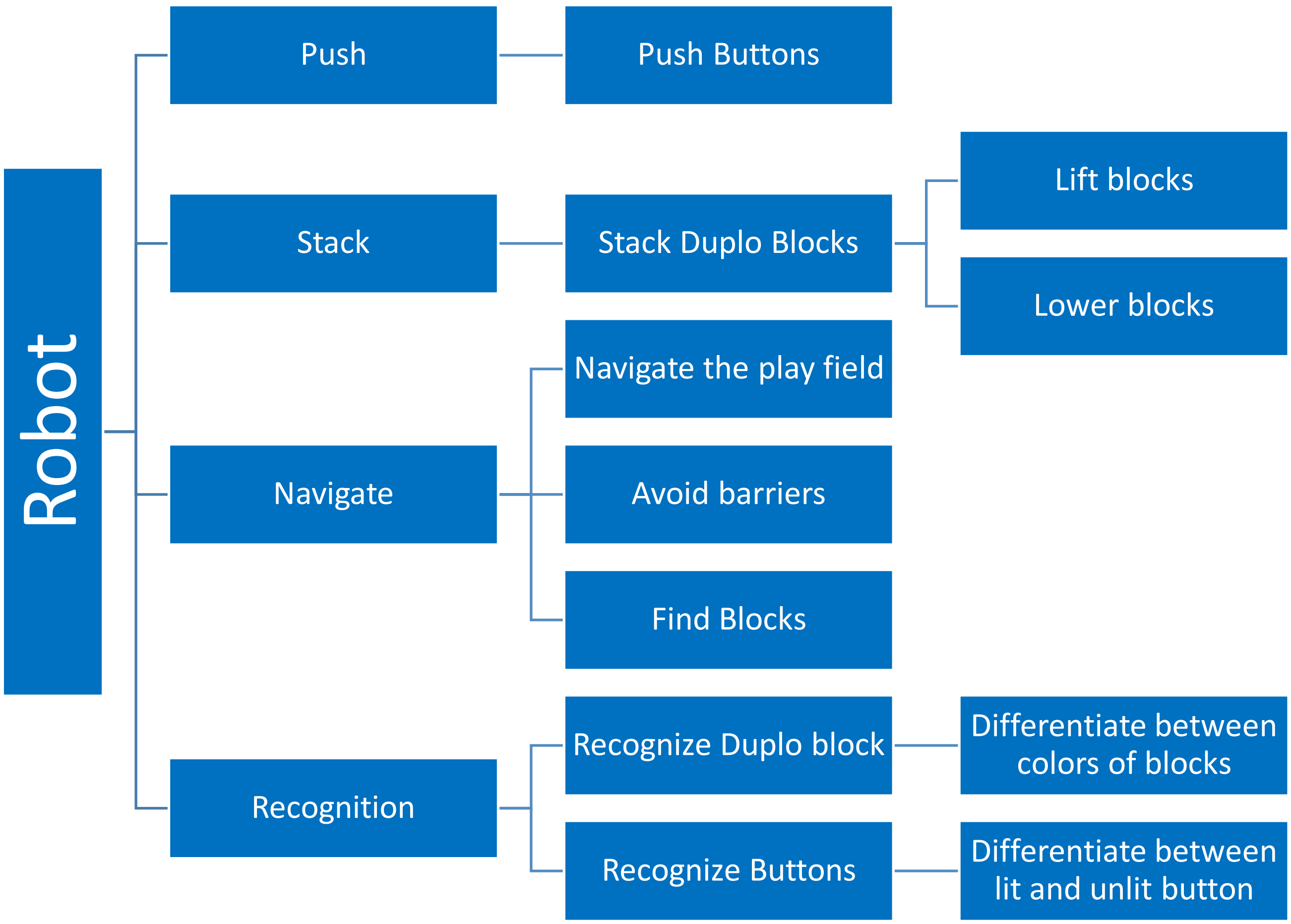

Figure 15. Functional Decomposition Hierarchy

Table 8.
Cross-Reference Table

	System			
Function	Push	Stack	Navigate	Recognition
Navigate the play field				
Avoid Barriers				
Push Buttons				
Find Blocks				
Lift Duplo Blocks				
Lower Duplo Blocks				
Differentiate between blocks				
Differentiate between buttons				

Appendix C: Target Catalog

Table 9.
Functions, Metrics, and Targets

Functions	Metrics	Target
Navigate the Play Field*	Number of path combinations completed	10
Avoid Barriers	Distance from barrier (inches)	>1 in
Push Buttons	Depth of button push (inches)	1/8 in
Find Block	Time to locate block (seconds)	<5 sec
Lift Duplo Block	Height reached (inches)	~1.5 block height above previous block
Lower Duplo Block	Height reached (inches)	1 block height above previous block
Differentiate between buttons	Time to locate proper button (seconds)	<10 sec
Differentiate between blocks	Time to reach correct bin (seconds)	<20 sec

* Indicates Critical Targets

Appendix D: Concepts

1. Basic, 4-wheeled robot, swiveling arm in center with 2 joints and sonar sensors for navigation.
2. 3 wheels, 2 in front, omnidirectional in rear, same arm as idea 1
3. Arm anchored at 12” height, 3 joints (can reach higher than 12”)
4. Design 1, infrared sensors for navigation
5. Use color recognition and sonar to find blocks in the correct order
6. Arm that lifts block and stacks upside down (like garbage truck) stack entire stack at the end on platform.

7. Robot stacks in pi order in increments of 10 and places stack near platform and puts the stacks together at the end.
8. Crane arm that can be extended when initialized (slider)
9. Robot stores a certain amount of each color needed at a time on body and places in correct order on platform.
10. The robot has a sorter on its body with blocks that slide in. A gate opens releasing the next color necessary in the stacking sequence.
11. Elevator arm used to stack blocks from below
12. Robot with leg like extremities used to increase height and to move through field.
13. Car like robot with lift kit, to maximize height of robot.
14. Two robots, one gathers Lego Blocks at base, another robot stacks them in correct order.
15. Robot with up to 10 arms, picks ten stacked towers, move ten towers to base and from there it takes each Blocks in correct Pi order and stacks them at base.
16. Same idea as before, though Robot only stacks in correct order until sometime within the three minutes, then it stacks the remaining towers on top of the stack that’s on the base.
17. Robot has a tray where it stores ten stacked towers to then later use to stack at base.
18. Scissor lift arm that stacks blocks.
19. Robot gathers all blocks and arranges the blocks in Pi order horizontally and then flips the stacked tower vertically before setting it down in base.
20. Car like robot with lift kit and a scissor lift to maximize its height.
21. 2 robots one stacks Legos like the one in idea 1 the other pushes buttons in pi order

22. Same as before except second robot pushes one button 100 times so it doesn't have to move much
23. The robot has a pincer like appendage at the back that picks up the Lego (like a scorpion)
24. Uses line following sensors to guide itself to the blocks general direction then uses sonar sensors same as 1
25. Same as 24 except it uses hard coded locations after the line sensor instead of sonar
26. Robot to stack Legos that has 2 arms to hold the Lego and stack it and has a hammerlike appendage on top to make sure the Legos stick together
27. Robot to push button that has 4 arms and extends them to push the correct ones in pi order
28. A robot that stacks Legos but detects if they fell over and if they do it starts from the last one standing
29. A robot that stacks the Legos from the bottom up and holds it at the end to keep it stable but at the opposite end from where judges will see
30. The robot stacks Lego blocks on a mobile platform
31. The legos are setup upside down, so that robot can flip them as they are picked up
32. The pickup motion is the same as using two hands to pick up an open box and putting the box on your head.
33. From the mobile platform, the robot will locate and hold the stack. (without pushing it over) the arms that pick up blocks can be used to hold the stack.
34. The robot will raise the blocks from the mobile platform and set it on top of the stack.
35. If order isn't a priority, and the arm is strong enough, the robot will raise from the bottom and push a new stack on the bottom.

36. Robot that has a tail like a scorpion and picks up blocks by "stinging" them.
37. Robot can use omni direction wheels and sensors to facilitate sideways movement and finely adjust the position of the robot with respect to the stack of legos so that the arms can be more precise in stacking blocks.
38. The robot can use sensors in front of the vehicle to determine when it has reached the white lines and markings (center of the playfield) and be able to determine whether it should turn left or right towards other bins or head to the Lego stacking platform.
39. The robot will track the white lines on the field to navigate. The robot will pick up the legos with a claw hand. It will stack 5 bricks at a time within the robot and stack each stack on the base.
40. The robot will use a "spatula" to pick up the blocks.
41. There will be 2 robots, one will collect the blocks and deliver it to the 2nd bot. The other will be stationed at the base and be stacking the blocks brought to it by the 1st robot.
42. There will be 2 robots, the 1st will push buttons on the wall. The 2nd will collect and stack blocks.
43. 3 robots, 1st will push the buttons, 2nd will collect and deliver the blocks over to the base, and 3rd will stack the blocks on the base.
44. The robot has an elevator with 2 claws that grabs a block and lifts up, then the lower claw will lift the next block up to the stack and force them together, and it will repeat the process until it has 10 together. Then it will place the stack on the base and repeat until the time is up.

45. Robot with 5 bins on each side of the robot to hold the blocks of each color. Arms will take from the bins as needed and stack the blocks.
46. Robot will collect blocks from each bin 1 at a time; drive it over to the base and stack them 1 at a time.
47. Robot will push the buttons using a solenoid.
48. Robot will push the buttons using a piston.
49. 10 robots that will each go collect 1 block at a time and bring them over to the base and stack them.
50. 3 robots, 1st will grab bricks from the left side of the fields, 2nd will grab bricks from the right side, and 3rd

Table 10.
Morphological Chart

Push Buttons	Lift/Lower blocks	Navigate the playfield	Avoid Barriers	Find Blocks	Differentiate between blocks	Differentiate between buttons	Motion	Coding Language
Piston	Arm and claw	Infrared	Infrared	Sonar	Camera	Camera	4 Wheels	C++
Navigation	Elevator and two claws	Sonar	Sonar	Infrared	Hard coding Locations	Hard coding locations	3 Wheels	C
	Elevator and one claw	Infrared and sonar	Infrared and sonar	Camera			Tracks	Python
		Hardcoding locations	Hard coding locations	Proximity sensor			Spherical Wheels	MATLAB
		Camera	Camera				Legs	Java
		Line following	Proximity sensor					

51. The robot uses an arm and claw to lift and lower the blocks. Infrared sensors will be used to navigate the playfield and avoid barriers. It will use sonar to find the blocks and a

camera to differentiate between blocks and buttons. The robot will sit on a 4-wheeled base and its motion will be coded using C++. The robot will push the buttons using a piston.

References

Electronics, T. (2017, March 25). Universal Robots vision-based LEGO DUPLO stacker. Retrieved from https://www.youtube.com/watch?v=fLyhVfCzkio

Radford, R. (2019, June 1). *southeastcon2020.* Retrieved from github: https://github.com/ncgadgetry/southeastcon2020/blob/master/Rules/PiDayRules_Jun1_2019.pdf