

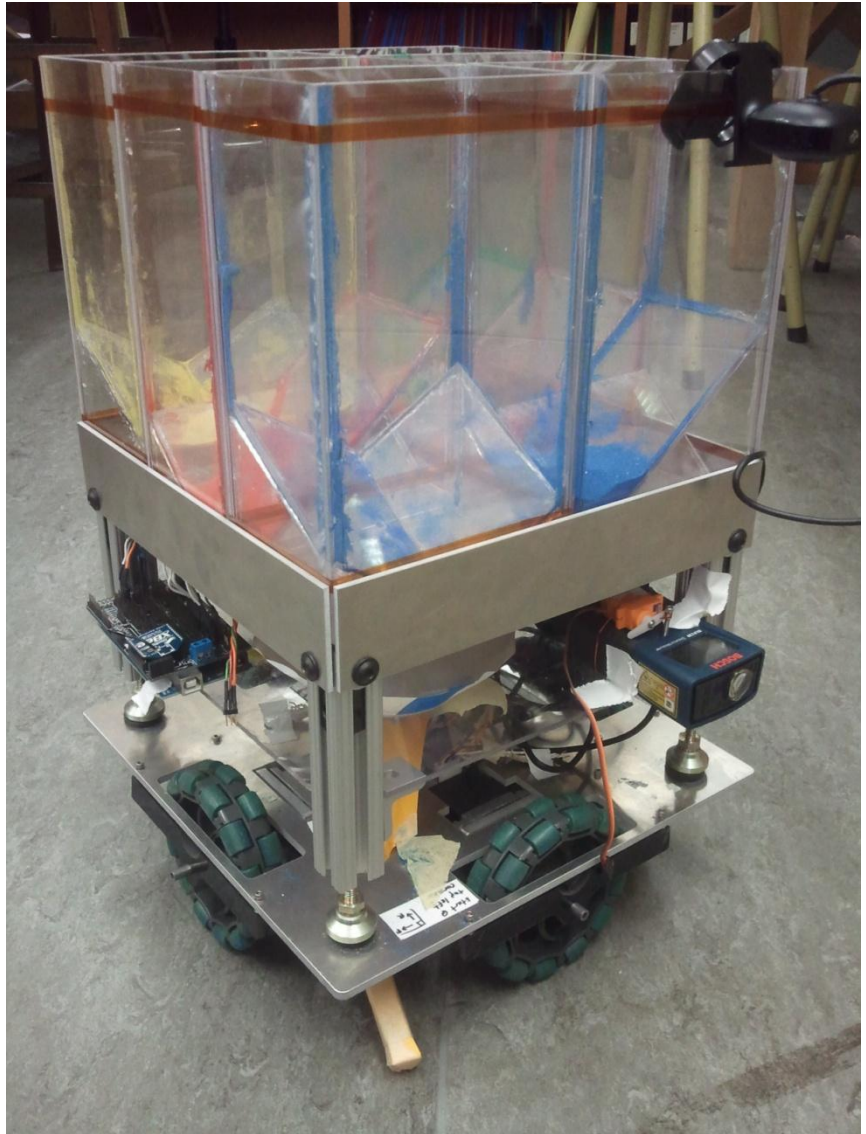
# Sand-painting Autonomous Nomadic Depositor (S.A.N.D.)

Tristan Miller  
Pranav Saxena  
Rosanna Yee

Project Sponsors:  
David Long  
Antoine Morris

Applied Science 479  
Engineering Physics  
UBC

January 9, 2012



## Executive Summary

The purpose of this project is to design and develop a low-cost, portable, and autonomous sand painting robot. The robot uses 6 different colours in its sand painting and covers an area up to 6m x 6m fully with 1cm x 1cm sand pixels. It can be assumed that the surface the robot works on is dry, even, and level, and that there is no wind present.

A simple computer interface to the robot is provided which allows the user to input the physical dimensions of the room the robot is allotted, the image the robot is to paint, and other information to direct the robot. The robot operates in an x-y plane in a printer-like fashion, with stepper motors driving omni-wheels. The sand deposition system is gravity-fed, with servo motors controlling the sand flow out of 6 sand canisters. Two laser distance meters are used to track the position and alignment of the robot. The system would consist of wireless communication via Xbee between the robot and a laptop. The laptop serves as a control centre for directing the robot's movement, sand dispensing, refilling, etc.

After testing, the robot has met requirements for consistency of amount of sand delivered in sand shots, movement precision between pixels, and the time required for the robot to distribute sand per pixel. However, at the current stage of development, not all components of the robot are fully finished. Sand level measurement using IR sensors have not been implemented and the integration of position tracking via laser distance meter in the operation of the robot has yet to be completed. While the GeckoDrive boards currently used to control the stepper motors are functional, due to their expense, an alternative driver board, A4983 Pololu stepper motor driver carrier, would later replace the GeckoDrive board instead. The team is committed to further work past the January, with the goal of completing all required aspects of the robot by March 8, 2012.

# Table of Contents

Executive Summary	pg ii
Table of Contents	pg iii
List of Figures and Tables	pg iv
1. Introduction	pg 1
1.1 Background and Motivation	pg 1
1.2 Existing Technologies	pg 1
1.3 Project Objectives	pg 2
1.4 Scope and Limitations	pg 3
1.5 Overview of Report	pg 3
2. Discussion	pg 4
2.1 Final Design	pg 4
2.2 Alternative Methods	pg 11
2.3 Testing and Reviewing the System	pg 15
2.4 Results	pg 17
2.5 Discussion of Results	pg 17
2.6 Key Issues	pg 20
3. Conclusions	pg 21
4. Project Deliverables	pg 22
4.1 List of Deliverables	pg 22
4.2 Ongoing Commitments	pg 22
4.3 Financial Summary	pg 23
5. Recommendations	pg 24
6. Operation Manual	pg 25
7. Appendices	pg 30
Appendix A - Software Code	pg 30
Appendix B - Technical Specifications	pg 31
Appendix C - Meeting Logs	pg 32
8. References	pg 44

# List of Figures and Tables

## Figures:

Figure 1 – Sand Mandala	pg 1
Figure 2 – CAD design showing sand canisters	pg 4
Figure 3 – Servo and tube for flow control	pg 5
Figure 4 – CAD design showing collecting funnel and exit tube	pg 5
Figure 5 – Temporary nozzle attached to exit tube	pg 5
Figure 6 – Vex Robotics Omni-wheel	pg 6
Figure 7 – Overview of laser tracking system	pg 7
Figure 8 – XBee module communication setup	pg 8
Figure 9 – Images of the software GUI	pg 9
Figure 10 – Alternative sand dispensing	pg 11
Figure 11 – Alternative sand dispensing from below	pg 12
Figure 12 – Sample mat marker pattern	pg 14
Figure 13 – Example of sand pixels deposited by robot	pg 18
Figure 14 – Bosch DLR130K display	pg 18
Figure 15 – Example output of template recognition test	pg 19
Figure 16 – Power board wiring diagram	pg 27
Figure 17 – Arduion MEGA 2560 IO Expansion Shield	pg 28

## Tables:

Table 1 – Results gathered	pg 17
Table 2 – Results to be gathered	pg 17
Table 3 – Bill of Materials	pg 23
Table 4 – List of Screws	pg 25

# 1. Introduction

## 1.1 Background and Motivation

This project is sponsored by David Long and Antoine Morris from The Practice of Everyday Design, a company specializing in installation art, building design, and product design. The inspiration for the sand painting robot comes from Tibetan sand mandalas, which are highly detailed “paintings” with coloured sand. The sand painting robot was proposed as a project which combines engineering, art, and design in creative and innovative ways. The intention of this project is that with a simple computer interface, any person could create aesthetically striking works of art. The sand paintings would be large scale (up to 6m x 6m), highly detailed, and visually stunning. The system should be autonomous, portable, and be versatile enough to be used for a variety of venues for art exhibitions.

While the purpose of the robot is to create sand mandalas and paint other images in sand for art exhibitions, the application of the robot and the sand deposition system could be relevant for different fields. One example would be agriculture, where the robot could be modified to drill a small hole and deposit seeds instead of sand.



Figure 1. Sand Mandala

Source: <http://vlone2010.wordpress.com/2010/10/18/sand-mandala-creation-tenzin-yiggyen/>

## 1.2 Existing Technologies

Similar concepts of integrating art and engineering has been implemented before, where the robot is programmed to create a piece of art. However, a sand painting robot has not been attempted before, especially to the level of sophistication and accuracy this project strives to achieve. The closest project

that was found was an autonomous robot that dropped piles of sand along a spiral path to track visually the energy of the sun as it moved across the sky. Most robots working with sand had defined areas where the robot would interact with the sand (i.e. a sandbox), where the sand is manipulated by pushing or drawing in the sand. In these cases, the way the robot interacts with the sand and the physics behind how the sand moves is completely different from sand painting.

More relevant to designing our project are autonomous robots that do not have fixed areas where they work in, can align and position themselves fairly accurately, and are light and portable. An example of this is the Hecker motion system, which is an autonomous robot that produces graffiti art and text. Compared to graffiti art robots that use spray cans, using sand as an art medium presents unique technical difficulties such as weight, leakage, controlling how sand becomes deposited, and potential wear due to abrasion.

### **1.3 Project Objectives**

The goals of this project are to deliver a robot which can fulfill the following criteria:

- Autonomous
  - Robot should not require supervision after initial set-up and alignment
- Portable
  - Less than 50lb or 23kg
  - Maximum linear dimensions (height + length + width) of 158cm
- Produce a sand painting using 6 different coloured sands, covering an area ranging from 3m x 3m to 6m x 6m fully without sparse areas
- Perform in a variety of environments, both indoors and outdoors, and be resistant to corrosion.
  - It can be assumed that the surface the robot works on is dry, even, and level, and that there is no wind.
- A simple computer interface to the robot is to be provided which allows the user to input the physical dimensions of the room the robot is allotted, the image the robot is to paint, and other information to direct the robot.
- Deliver this robotic system, fully implemented with GUI in January 2012

### **1.4 Scope and Limitations**

The nature of the project has many limitations on the design of the system. The robot's design must be light, durable, portable, accurate, and robust. Aside from the technical requirements of the robot, there are also logistics of how the system works that can impose limits and potential challenges (for example, the path of the robot must never intersect where the sand has been deposited, how the sand is refilled, held, and dispensed, etc).

The main challenges in designing the autonomous sand painting robot were to devise a cheap, reliable, effective method with the following constraints:

- Reliability and repeatability:
  - The sand painting at its maximum size (6m x 6m) would have 360,000 “sand pixels” the robot needs to deposit uniformly and consistently.
  - Moving over a range of 6m repeatedly, there could be slight drift in the trajectory of the robot which needs to be rectified or else the error accumulates.
- Portability:
  - Total weight of the robot system needs to be  $\leq 23\text{kg}$ .
  - Size of the system needs to fit within a box of linear dimension (height + length + width)  $\leq 158\text{cm}$ .
  - Due to its portability, the robot system needs to be simple for the user to disassemble, reassemble, and set-up.
- Position control:
  - The robot must have accurate positioning on the order of millimetres (smallest sand pixel is 1cm x 1cm).
  - The robot needs to realize if there is change in its orientation and be able to correct it.
- Sand
  - The type of sand used is very fine coloured sand. There are challenges in designing the robot such that there is sufficient sealing, controlling sand flow, and ensuring there is no leak or clogs in the system.
  - Depositing sand onto the ground surface could prove to be difficult as the sand tends to scatter over a large area. The sand used has very small, rounded grains and is deposited in small amounts in short, staccato bursts.
- Refilling:
  - To cover an area of 6m x 6m requires more sand than the robot would likely be able to carry. This would mean the sand receptacles would need to be refilled. While project sponsors would prefer to have the robot autonomously refill its sand containers, this would require a supplementary system.

## 1.5 Overview of Report

In the following sections, the project will be described in further detail. In “Section 2. Discussion”, the implementation and alternative methods are given. In addition, the tests carried out to characterize the system and the results are discussed. “Section 3. Conclusion” summarizes the findings of the project. “Section 4. Project Deliverables” gives a list of the deliverables and the financial breakdown of the project. “Section 5. Recommendations” gives a series of suggestions surrounding the issues encountered.

## 2. Discussion

### 2.1 Final Design

In this section the various methods employed in the designing, building, and testing of the system are detailed. The methods could be divided into 4 main sections: sand dispensing, mobility, position tracking, and electrical/software components.

#### Sand dispensing

The main components in the sand dispensing system from top to bottom are the sand canisters, the canister plate, the six servo motors controlling sand flow, the collecting funnel, exit tube, and the nozzle. The sand is held in 6 containers on the robot, with each container holding different pre-coloured sand (i.e. blue, red, yellow, black, green, and white). These receptacles are placed in a 2x3 formation, and are placed above the canister deck (Figure 2). The canister deck has 6 holes where the containers are fitted through. Tubes are attached to each sand canister, and sand flow through these tubes is controlled by servo motors that change the position of the tubes.

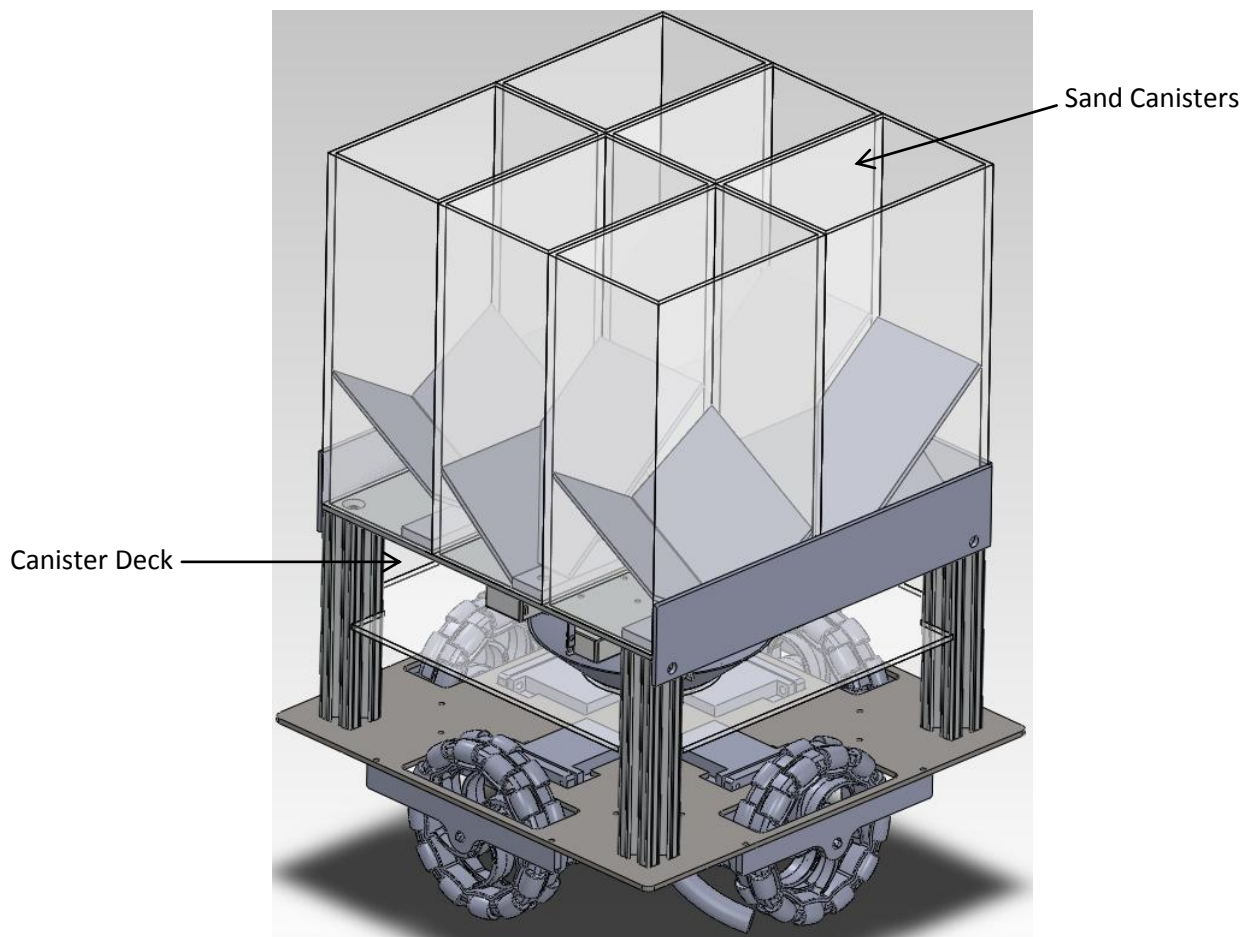


Figure 2. CAD design showing sand canisters



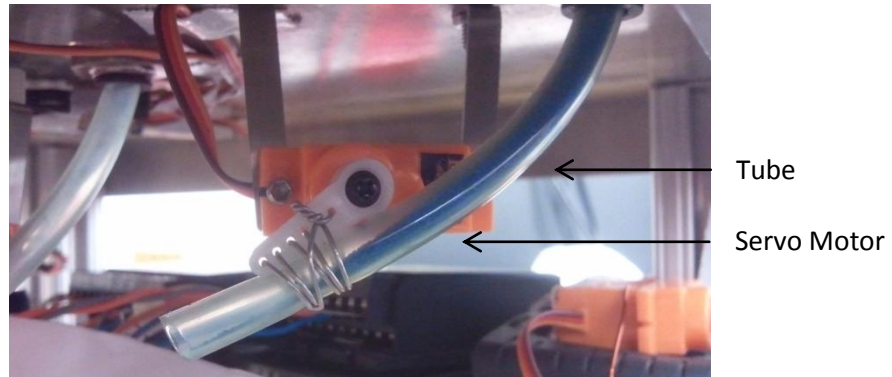


Figure 3. Servo motor and tube for sand flow control

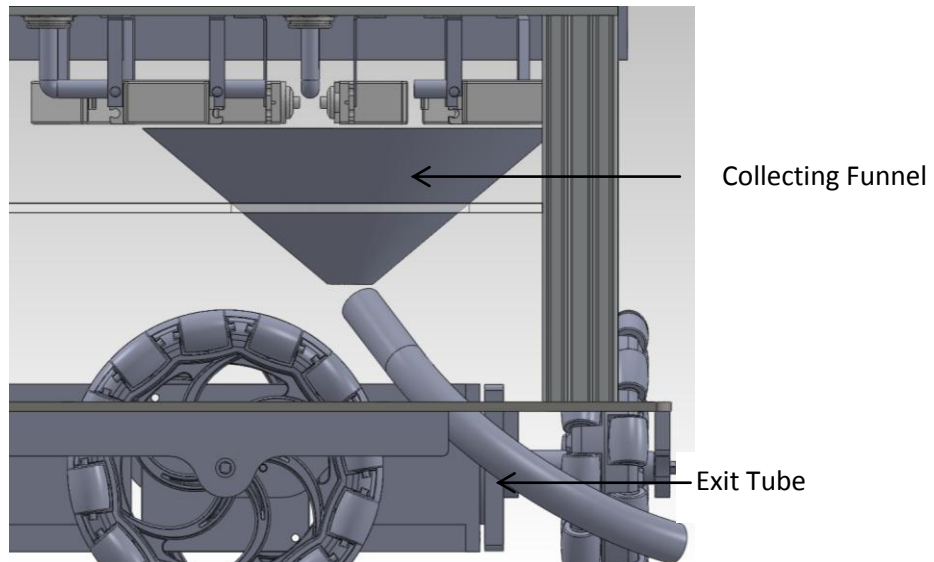


Figure 4. CAD design showing collecting funnel and exit tube

Sand that falls through the tubes is collected in a common funnel. The exit tube attached to the base of the funnel delivers the sand to the nozzle. The current design of the nozzle is an overhanging hood that focuses the incoming sand into a square pixel (Figure 5).



Figure 5. Temporary nozzle attached to exit tube

## Mobility

The movement of the robot is constrained along the x and y directions with four bi-directional wheels. The wheels are driven in one direction, but passive in the other. By placing each pair of wheels at right angles and controlling the motors to spin at different times, movement along one direction at a time can be achieved.



Figure 6. Vex Robotics Omni-wheel

Source: <http://www.vexrobotics.com/products/accessories/motion/276-2185.html>

Stepper motors are used to drive the wheels. These motors can precisely rotate at steps of fractions of a degree and provide sufficient torque without the need of gearboxes.

The robot traverses the image in an x-axis-dominant fashion, like a printer. The entire x length is traversed for each row with the y-axis traversed a centimetre at a time, after each row is completed. The robot is likely to veer off-centre during travel and result in a curved line of sand. To correct this, the x-directional wheels are each driven by separate stepper motors to allow for orientation correction. Based on the detected position tracking feedback from laser distance meters (described below), the two motors use a PID control loop to compensate for error and maintain a straight line.

The stepper motors alone should be sufficient for open-loop position control, but since precise positioning is crucial to this system an error or slip in traction with the floor cannot be risked. In conjunction with the stepper motors, two lasers are used to accurately read and calculate the robots position in the x and y directions.

## Position Tracking

The chosen method to implement closed loop position & orientation feedback depends on the use of two laser distance meters. Figure 3 shows a simple schematic of the setup. The painting will be surrounded by two perpendicular walls, which will be a maximum height of 15 cm. The lasers are attached to the robot and detect the distance between the robot and the walls with a resolution of 1.5 mm. The two laser distance meters are in parallel, facing the y-direction. To provide alignment

feedback, readings from the two Y-facing meters are utilized in a PID control loop to minimize the difference between readings. This strategy ensures the robot maintains a straight path. To take a distance measurement, a servo motor on the laser meter is actuated to prompt a measurement. To check the x-direction, a mirror attached to a servo arm would be lowered into the laser's path to reflect the light at a 90° angle to measure distance in the x-direction. By first ensuring the robot's alignment, the robot's position can be accurately checked and fixed by simply probing each laser in either the x or y-direction.

From this, a simple calculation of  $\# \text{ steps} = \frac{\# \text{ steps}}{\text{revolution}} \times d \times \frac{\text{revolution}}{2\pi r}$  gives the required translation to correct the position. Note that  $d$  is the distance to be travelled in either the x or y direction, given a meter reading and the required location according to software. The  $\# \text{ steps/revolution}$  is a characteristic of the stepper motor (for this project, 200 steps/revolution). Lastly, the  $2\pi r$  term gives us the wheel circumference, the distance travelled per revolution.

The laser distance meters are a commercial item, and as such, do not have a useful serial data output. We attempted to access the circuitry of the meters and read the output data being sent to the LCD screen. After much efforts, the output information was observed to be a constantly changing assortment of 1's and 0's, even while the screen should be displaying a constant distance value. This proved difficult to decipher, and combined with project time constraints, was abandoned to save time. We instead fell back to the previously explored method of template recognition. By saving template images of each digit on the distance meter readout (0-9), input images from a webcam can be used to read the distance readout. For each digit, each template is used to scan the input image for similarities. The most similar digit is chosen and a final eventual readout is deciphered.

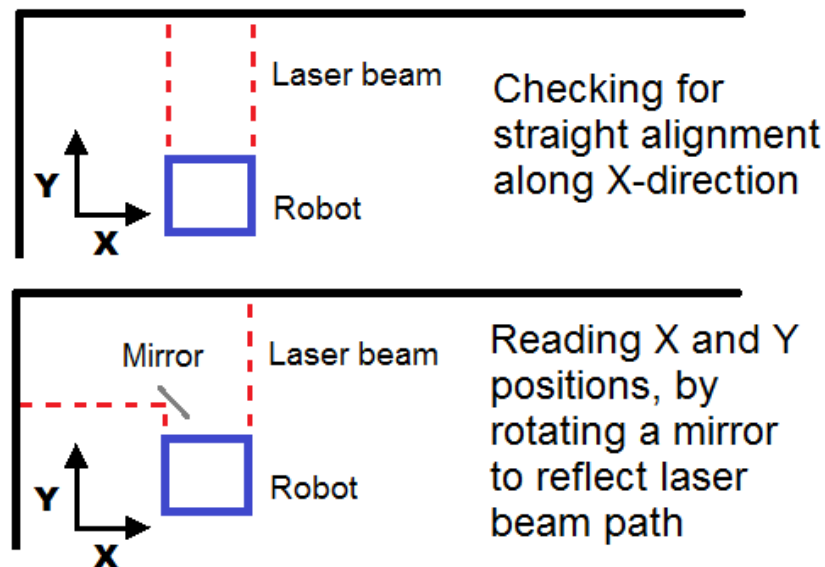


Figure 7. Overview of laser tracking system

## Electrical/Software Component

With the many actuators and sensors involved, organization is required to properly handle all wiring, boards, and processing software. An Arduino Mega 2560 Microcontroller is used to handle all digital and analog input/output. The Arduino acts as a slave device to the host C++ application running on a nearby computer, relaying all sensor information and setting motor output via wireless communication. This is done via two XBee modules using serial communication (one connected via USB to the computer and one wired directly to the Arduino).

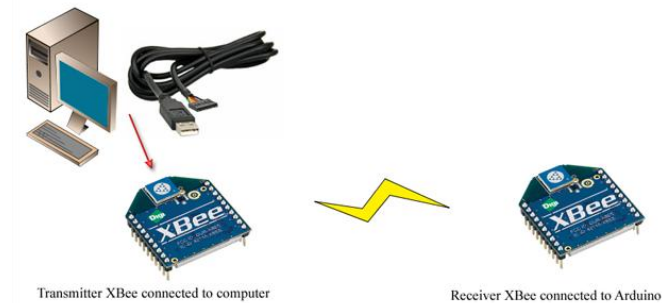


Figure 8: XBee module communication setup

[\[http://www.ladyada.net/make/xbee/arduino.html\]](http://www.ladyada.net/make/xbee/arduino.html)

The host C++ application handles all data processing requirements and also uses image processing libraries to continuously scan the input sand pattern image as the robot operates. This allows the use of a more powerful processor to manage robot operation, analyze input images and determine output colours, and to easily read image data from webcams to read the laser distance meters. Each of the four stepper motors utilize their own controller board for power and control management– i.e. number of steps, speed, and direction (GeckoDrives). These boards, along with the servo motors, are interfaced with the controlling Arduino via an intermediary board (see Figure 5). This board allows easy power connections for the stepper and servo motors, and also allows Arduino control signals to be properly routed to each actuator. Note that the stepper motors are numbered in a clockwise fashion, 1 to 4. This means that motors 1&4 and 2&3 can share a direction signal, as they will either move in the same direction (rotational movement), or not move while the other is moving (straight movement). Please see the operations manual for a description of this power/control board. Male and female header pins have been utilized as much as possible to allow connections to be neat and reliable. Also, the chassis has been assembled to allow space for all wiring and boards (on the Lexan plate holding the funnel). A removable wall will also be designed for the chassis.

The completed software component for both the host C++ and slave Arduino applications can be found as an attached zip file.

The host C++ application has been implemented as a C++/CLR Windows Form project using the Microsoft Visual Studio 2010 IDE. CLR stands for common language runtime and allows the use of managed (automatic memory garbage removal) C++ code with the Windows .NET framework. The .NET framework is a software framework which allows programming languages access to a vast

assortment of libraries, and in our case, allows access to the USB ports of the PC. This in turn grants access to the XBee module and allows very simply communication between the host and slave. The host application also utilizes a GUI to simplify user input and output up-to-date robot information as it operates. This GUI has two sections, the first being a simple interface for end-user operation, and the second being a set of testing and diagnostic operations to ensure good operation, and possibly highlight any failing components.

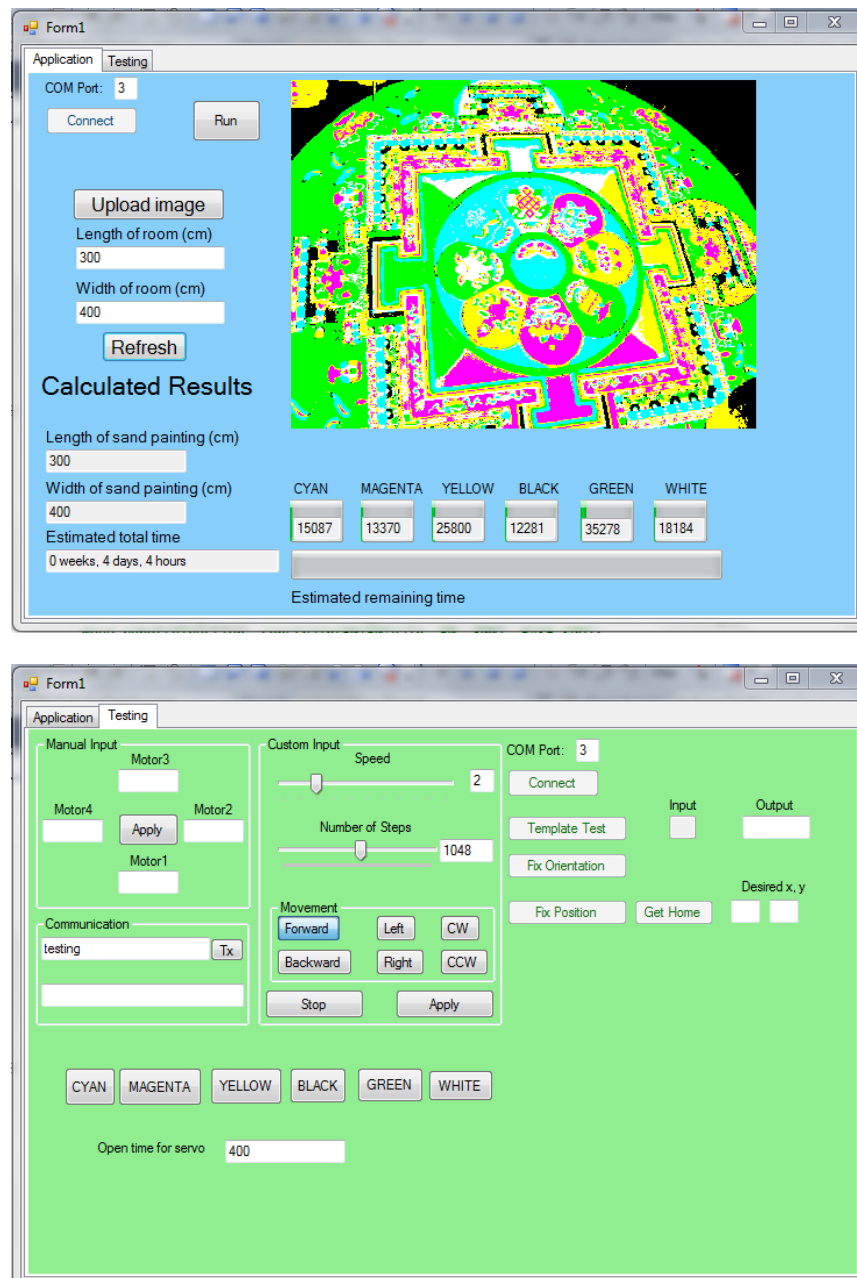


Figure 9: Images of the software GUI, both operation (showing preview output image) and diagnostic components

The first operation component of the GUI allows the user to input an image file of the desired sand painting pattern. Along with the maximum available floor space for the pattern, the image is analyzed and each pixel colour is transformed into a colour from the available set of cyan, magenta, yellow, green, black, and white. This is done by specifying a range of values in the RGB colour model for each available colour and simply assigning each pixel. Also, if the floor space is an integer number of times greater than the number of pixels in the input image, the output image is scaled appropriately. This is done by copying each pixel the same number of times in the x and y directions of the image matrix. Also, pseudo code has currently been written to properly scale the output image if the available floor space is less than the input image resolution. This involves utilizing a written image processing algorithm to simply resize the image (by reducing resolution) before assigning colours. This will be implemented before the upcoming Project fair.

The “Run” button depicted above frees the end-user of control and begins the autonomous operation of the robot, according to the specified input. This operation simply involves the robot moving in a printer-like fashion across the given floor space. Initially, the robot aligns according to the 90° angle set by the boundary walls and sets the current location as the home location (all later distance readings are set relative to this home position). The robot then moves to each pixel and drops the necessary coloured sand, according to the input image. The robot also routinely checks and fixes its alignment and position (utilizing feedback).

As mentioned, the robot utilizes a PID control loop to properly realign itself with the boundary walls. The difference between each y-facing laser’s reading is used to generate a current error (either positive or negative). Over time, the current error (used for the proportional component), the change in error with time (used for the derivative component), and the accumulation of error over time (used for the integral component) is used to decide an appropriate actuation of the stepper motors (according to number of steps) to accurately and quickly realign the robot.

Once the robot is realigned, the robot switches one laser meter to read in the x-direction, and each meter is probed to decipher how many steps are needed in each x and y direction to reposition the robot. This is done according to the equation described in the Position Tracking section above.

All separate components of the “Run” operation are embedded in the diagnostic component of the GUI. Here, the user can test the:

- stepper motors, by setting the desired speed, number of steps, and type of movement
- XBee serial communication channel, by sending test messages and receiving a reply
- servo motors, by manually actuating any servo
- reading of the laser distance meters via template recognition via camera input
- robot’s ability to realign, according to the nearby boundary walls
- robots ability to reposition to a new position, given a home position

## 2.2 Alternative Methods

Many alternative designs were considered for this project. This section discusses the methods that were not chosen and the reasoning behind the decision to use a different method.

### Alternative Designs - Sand dispensing

The main components in the sand dispensing system from top to bottom are the sand receptacles, the sensor disk, the select disk, and the gate.

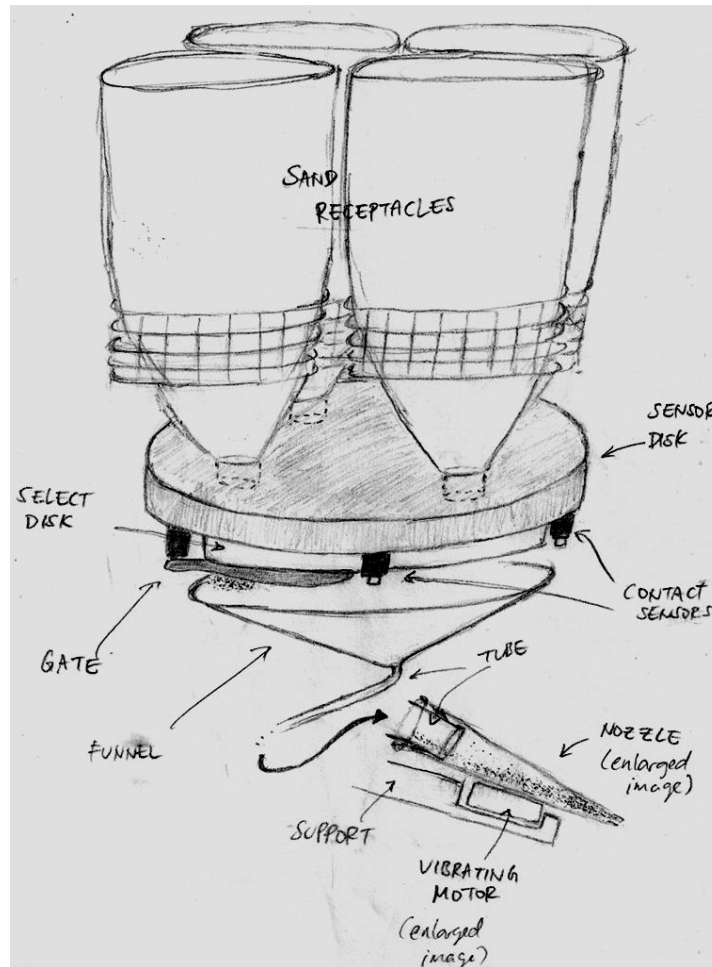


Figure 10. Alternative sand dispensing method

The sand would be held in the same formation as the proposed method, and is placed above the sensor disk. The sensor disk has 4 holes where the containers are fitted through, as well as 12 contact sensors. The 4 outermost contact sensors detect when the gate is in the “open” position, and the 4 innermost contact sensors detect when the gate is in the “closed” position. These contact sensors are placed along 0° and 90° lines in a cross formation. The other set of 4 sensors on the sensor disk are placed on diagonal lines at 45° angles.



Below the sensor disk is the select disk which rotates around to select which colour sand is to be dispensed. A gap in the select disk allows one of the 45° angle contact sensors to be uncovered, while the 3 other contact sensors are covered. Depending on which contact sensor is uncovered, the position of the select disk can be determined.

Attached to the select disk is the gate at a 45° angle to the gap in the select disk. When the select disk is oriented appropriately, the gate should be directly under a sand receptacle. The gate rotates to either open or closes the opening to the sand receptacle.

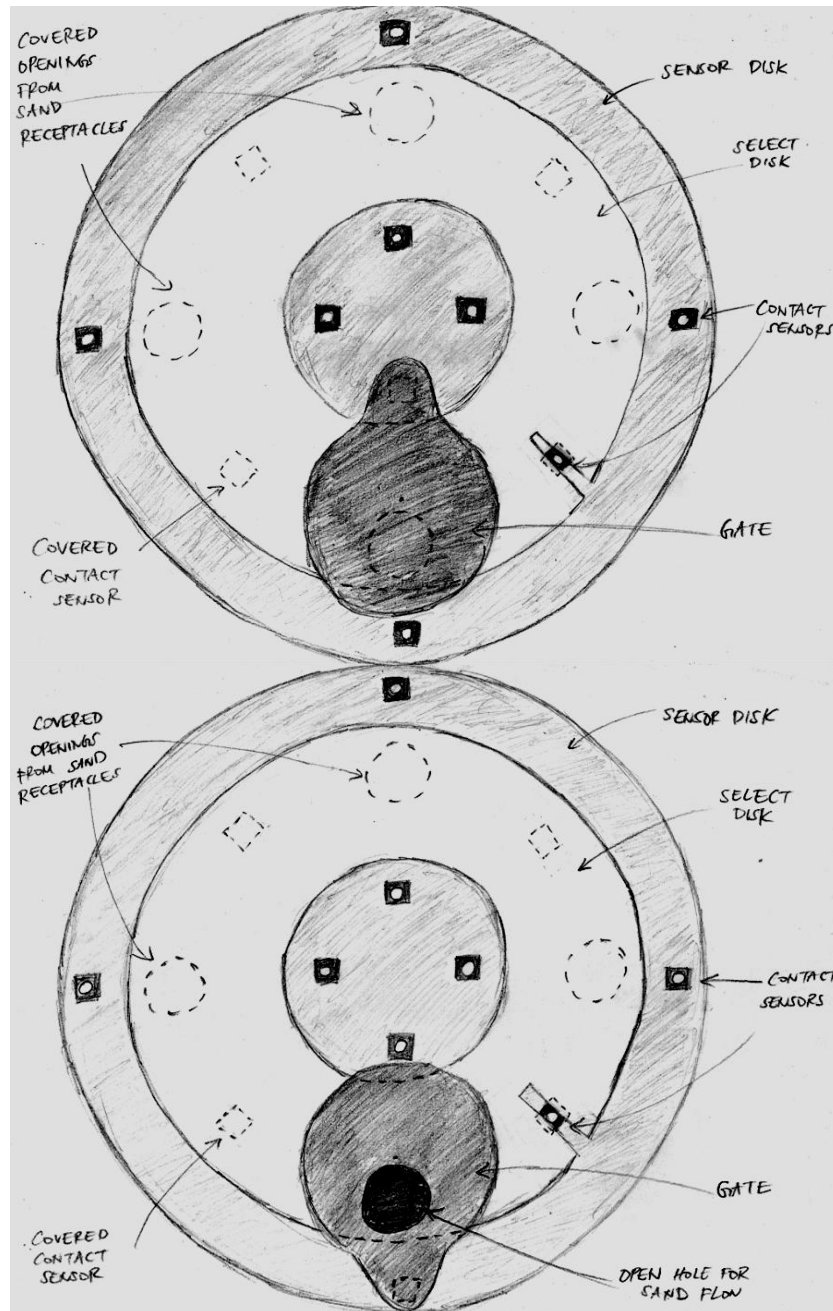


Figure 11. Alternative sand dispensing from below – closed/open position



This method was not chosen as the rotating disks imply that there would be a large amount of surface rubbing against each other where there is flow of sand. The potential issues with this are dealing with sealing issues, friction, and abrasion, which would wear down the system. Using solenoid valves are slightly more expensive, but are a much simpler, cleaner, and effective strategy.

### **Rail System vs. Independent Robot**

In brainstorming, how the robot could be designed was broken into two main classes: using a rail system or having an independent robot. A rail system would offer accuracy, the freedom for the robot to take any path in sand painting, and could be applied towards a Cartesian system or a radial system. However, the bulk, weight, and cost of manufacturing a rail system that could be disassembled into shorter segments and reassembled made using rails not a feasible choice given the project's portability constraint. An independent robot would be much lighter and could hold more sand at any time, but would have issues with accurate position tracking.

### **Alternative Designs - Position Tracking**

In order to develop a robust position and orientation tracking system, a few ideas have been proposed. The first utilizes two string encoders and an accurate earth magnetic module to accurately track. The string encoder is a device capable of detecting and measuring linear position using a flexible cable and a spring-loaded spool. With two encoders located at two adjacent corners, the position and orientation of the robot can be very accurately tracked using triangulation and slope calculation. The magnetic module would be used to aid in orientation tracking, in the case where a wheel loses traction and the slippage causes some rotation in the robot's orientation.

Although using string encoders would be the easiest and most accurate method to track the robot's position, the price of a string encoder measuring over the area where the sand painting takes place would cost in the range of \$2000CAD. This far exceeds the projects budget and is thus not a viable option.

A second idea proposed was the use of a y-directional laser and sonar system attached to a timing belt. As the robot moves in the x direction, a detector attached to the robot senses laser position and shape to reorient the robot along the straight laser path. A sonar module attached to the robot is used to track the x position of the robot over the 6m range. A metal sheet can be attached underneath the laser to reflect all ultrasonic pulses. Also, since the timing belt rail is assembled exactly straight, a simple encoder is sufficient to ensure accurate y movement.

There are many issues with this concept. The use of a rail for the laser system assumes that the room the robot is working in can hold a 6m long rail. Using a sonar module for tracking the position in the x direction may not provide a good enough resolution since for long range measurements, the best resolution commercial sonar sensors have is approximately 1cm.

Another option developed to track the position of the robot was a mat marker system. This entails placing a mat underneath the robot with graphics (markers) in each pixel of the painting and then using a camera on the robot to track the position through image processing. A sample of these markers is shown in figure 4. Each of the markers is unique and is placed in a unique pixel. Therefore, once the camera on the robot determines which marker it is hovering over, it is able to calculate precisely where it is. Also, using the edges of the graphic we could find the orientation of the robot by comparing the square of the image with the reference square of the camera.

This method is not the primary choice of position tracking because of the visually obtrusive nature of the mat. If there are any blank areas of the painting then the mat would be visible between the sand. Also, it is preferred that the final painting have no remnant of the robotic creation.

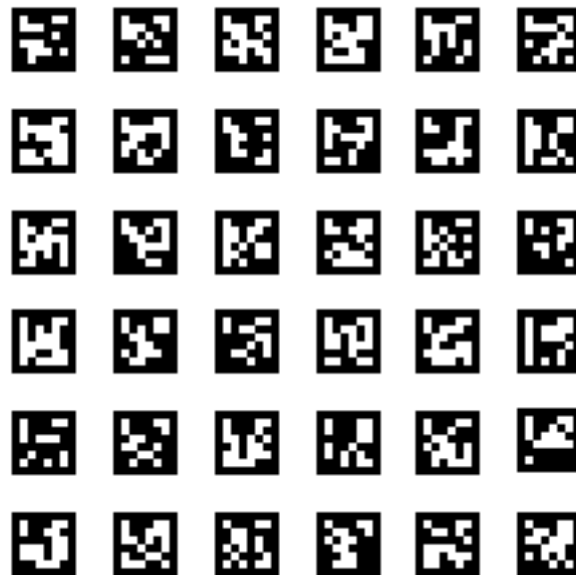


Figure 12: Sample mat marker pattern

## 2.3 Testing and Reviewing the System

Throughout development we tested and evaluated the robot's performance in various areas. These subsections include Sand Dispersal, Mobility, Position Tracking, and Communications. Some tests were not performed since we are continuing to develop and fine-tune the robot.

### Testing and Reviewing - Sand dispensing

The possible problems with sand dispersal include:

- sparse sand deposition inside 1cmx1cm pixel
- sand deposition outside pixels
- clumping sand deposits
- varying volumes of sand dropped throughout sand container levels

Two testing methods were devised and performed to evaluate the robot's performance in overcoming the above issues.

- 1) Once the settings of the robot were configured (i.e. sand nozzle angle, time intervals of sand release, etc.), we set the robot to paint out 1m line segments with only one colour. This tested:
  - clumping or sparse sand placement
  - variance in discrete sand volumes deposited (as entire container is depleted)
  - the even flow of sand out of the nozzle (possible clumping inside the robot)
- 2) Once configured we set the robot to paint out a 1m line segment with all four colours alternating. This tested mixing of colours inside the nozzle and on the floor.

The sand is quite fine and so is not expected to clump at all. The challenge is to not let sand spread too easily on the floor. We found that with the current setup the sand will mix and spread on the floor so this aspect will need to be refined in future revisions.

### Testing and Reviewing - Mobility

The possible problems with mobility include:

- an inability to consistently move exactly 1cm at a time
- constantly having to reorient the robot (x-directional wheels not in sync)
- wheel slipping (weight imbalance)

In order to test these issues we performed the Sand Dispersal method with additional conditions & observations. These included:

- testing 1cm movement by marking consecutive wheel positions on floor and measuring in between
- counting the number of necessary reorientations in code
- observing slipping

We need to be able to consistently move exactly 1cm at a time. Stepper motors tend to move fractions of a degree per step, giving us the needed precision in movement. We found the error in movement was less than 1mm as seen in the table in the next section. This provides us with the needed accuracy. The second part of the test has not been performed yet since it requires the closed loop feedback of the lasers which is not yet implemented. The slipping observed appeared to be minimal but will need to be tested further on different surfaces.

### **Testing and Reviewing - Position Tracking**

The possible problems with position tracking include:

- an inability to consistently and accurately calculate the current location
- an inability to travel to a desired location
- unreliable or inaccurate laser sensor readings

Since the closed loop feedback with lasers has not yet been implemented, a test method was devised but not performed. This method is to place the robot onto a grid and allow it to reorient and travel to a randomly generated location on the grid. This test will:

- test the robot's ability to calculate its current location and orientation
- test the robot's ability to get to a desired location
- this will also concurrently test the value of the laser sensor readings

With well-developed flat walls, it is expected that the robot will be able to consistently calculate its location and move to any desired location on the grid.

### **Testing and Reviewing -Communications**

The possible problems with communications include:

- an inability to parse host & slave commands (back and forth between PC/Arduino)
- corruption of message data

Since this aspect of the design is used to perform every other test (mobility, sand deposition, position tracking) a test method was not developed. Rather, we observed the reliability of the communications during use of the Arduino and no issues were discovered at any time.

## 2.4 Results

**Table 1. Results gathered**

Test Parameter	Target	Average Result	Average Error	Variance
Mass of each sand shot	+/- 0.05 g	0.3 g	0.044 g	0.00134 g <sup>2</sup>
Open loop straightness	+/- 10mm over 1m	N/A	+/- 4.2 mm	4.7 mm <sup>2</sup>
Accuracy of moving 1cm interval	10mm +/- 1mm	9.51mm	0.78 mm	1.042 mm <sup>2</sup>
Time to deposit a pixel and move to next pixel	<= 3 seconds	1.168 s	N/A	0.087 s <sup>2</sup>
Accuracy of template recognition in reading laser distance meter	>=95% accuracy	83.75% accuracy	3.33% for large digits 55% for small digit	N/A

**Table 2. Results to be gathered:**

Test Parameter	Target
Accuracy of template recognition in reading laser distance meter (with new webcam/lighting)	>=95% accuracy
Number of adjustments in position over 1m	<= 3 adjustments over 1m
Time taken to realign position	<= 2 seconds
Straightness tolerance with position tracking	+/- 5mm over 6m
Random walk test	+/- 2mm in x and y position

## 2.5 Discussion of Results

### Results - Sand dispensing

The size of sand pixels were fairly consistent, with the servo motors controlling the amount of sand delivered in each sand shot. From testing, it was found that on average the amount of sand per pixel was 0.3 grams. The results show that the average error falls within our targeted range of +/- 0.05g. As we did not have access to a scale with appropriate sensitivity, (scale in the lab has minimum resolution of 0.5g), the mass was calculated based on volume measurement with a small syringe and an assumption that all the sand colours have a density of 2kg/L.

In another test, the robot printed a 6x6 pixel sand painting, alternating colours blue, green, and yellow. There was a little mixing between sand colours as the nozzle on the exiting tube shaped the sand into small squares. Each pixel was relatively distinct with boundaries between pixels.

Visually, the distribution of sand in each pixel has good coverage, with few gaps in the painting, and no clumping was observed. It is assumed that for a larger sand painting the visual impact of small imperfections would be greatly reduced. The speed at which the robot completed a sand painting was much faster than targeted. As depicted above, we tested the time taken to deposit a sand pixel and move to the next. This was found to be 1.168 seconds, which translates to 4.87 days for painting a 6m x 6m sand painting.

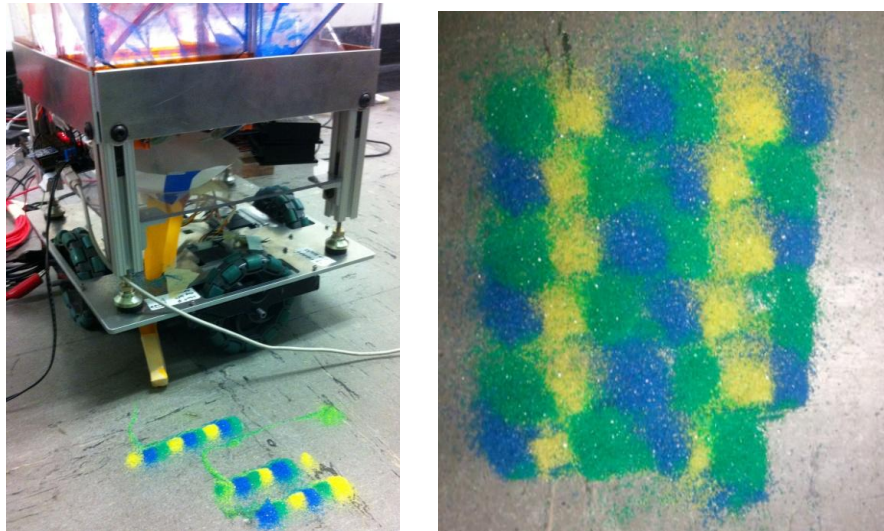


Figure 13. Example of sand pixels deposited by robot

### Results - Mobility

Since position tracking was not yet integrated with the system, the mobility tests were done with open-loop control. This means there was no feedback from the laser distance meters. As such, we consider a short term distance for the stepper motors to hand with accuracy. We expect a maximum deflection from the path of 1cm over 1m. Our target was met, as we observed a maximum deflection of  $\pm 4.2\text{mm}$ . The accuracy of movement in 1cm intervals was tested using stepper motors. Earlier attempts at a mobility system using DC motors showed that DC motors would not be able to offer a reliable step size for moving between pixels, with an error margin of roughly 50%. With large wheels, the accuracy of movement becomes more difficult as an error in angle of rotation is amplified in translation to distance travelled. With ungeared stepper motors, there was less backlash in the wheels and better control of the step size and speed of the wheels. The results show that the error in movement falls within the targeted range of moving 10mm per step with an error of  $\pm 1\text{mm}$ .

### Results - Position tracking

Template recognition tests show performance below the targeted accuracy, with 16.25% overall error in readings. The display of the laser distance meter displays the distance in units of meters, where the millimetre unit is shown in a smaller font.



Figure 14. Bosch DLR130K display

The webcam currently used cannot be adjusted for focal length, with a defined range of 0.4m – 2m, which is not ideal for this application. Because there is a limit on the camera's resolution and the display lies outside of the focal length, the digits on the display appeared blurry, affecting the template recognition. For the millimetre measurement, the smaller font was more difficult to recognize and thus more prone to error. This resulted in a 45% accuracy rate for detected the millimetre measurement, compared to a 96.7% accuracy rate for the larger font measurements.

Also, since the laser distance meters do not have backlit displays, a source of lighting needs to be provided for conditions with poor lighting. In the tests run, the webcam was placed above the laser meters so that adequate lighting was present. In the final design, the laser distance meter and the webcam would face each other horizontally, with LEDs added to act as an external light source. Placing the laser distance meter under the motor chassis would reduce unwanted glare and provide a more controlled environment for template recognition. With a more suitable webcam (higher resolution with adjustable focal length), and IR LED lighting, the template recognition would be vastly improved and become more robust.

Note that the template recognition is very sensitive to the orientation of the webcam with reference to the laser distance meter, and will need to be fixed.

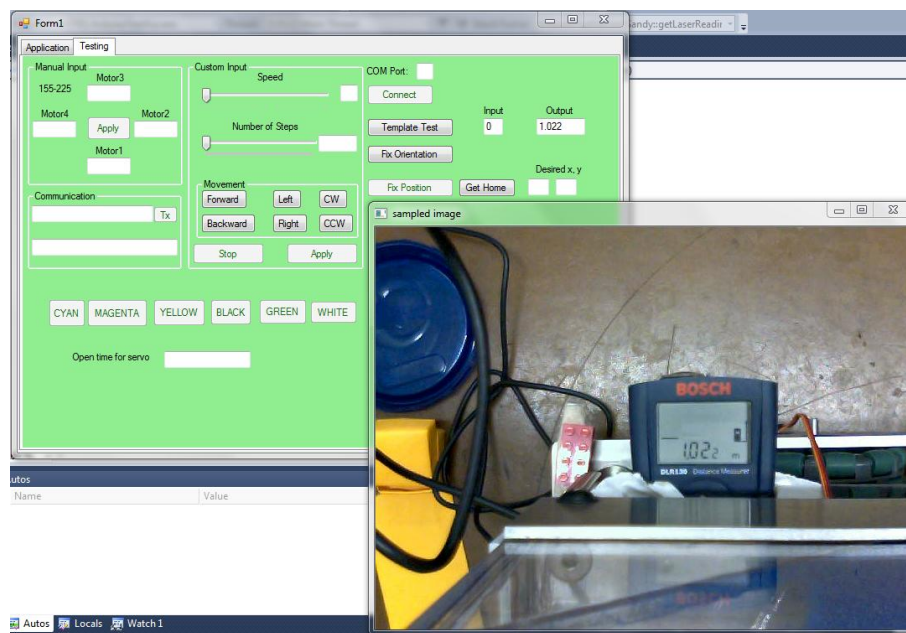


Figure 15: Example output of template recognition test

## Results - Communication

Throughout testing, communication through Xbee was very reliable. For better reliability, a delay of 2 ms was added after every byte as to not overwhelm the buffer.

## 2.6 Key Issues

The robot is not fully completed yet. The following components should be fixed before the project fair on March 8, 2012.

- Integration of closed loop feedback position tracking via the laser distance meters
- Investigate into webcams with a shorter focus distance (~10cm)
- Design and build laser and webcam supports to fix components onto chassis
- Implement sand level measurement using IR sensors with sand canister lids
- Replace GeckoDrive boards with A4983 Pololu stepper motor driver carriers to reduce cost
- Replace current funnel/exit tube system with more reliable and permanent structure
- Increase height of chassis to enable more reliable positioning of servo/tube deposition
- Organize and condense cables



### 3. Conclusions

This report describes the design, development, and testing of the Sand-painting Autonomous Nomadic Depositor robot, an autonomous sand-painting robot. There are three main components to this project: the mobility, sand-dispersal, and position tracking capabilities of the robot. The mobility component is completed, as the stepper motors can be reliably and precisely controlled via the GUI host application. The Arduino microcontroller successfully handles all commands and relays digital signals to autonomously control robot movement. The sand-dispersal component of the robot is implemented and currently works at half-capacity (3 out of 6 servos work). This component will be redesigned and fixed by simply designing the robot with more height, allowing the servo tubes, sand funnel, and sand tube/nozzle to utilize a steeper angle – guaranteeing reliable and constant sand flow. The position tracking capability of the robot has been designed and implemented in software. The capability in this aspect has been proven by using webcams and template recognition to successfully read the laser distance meters (albeit currently with some error). In further work, the position/orientation feedback system will be integrated into the robot by designing and assembling mounts for the webcams, laser distance meters, and associated servo motors to actuate the mirror and button inputs.

The capabilities of the robot thus far have been tested and analyzed. The mass of each sand shot has been found to be  $0.3\text{g} \pm 0.044\text{g}$ , which reaches the target difference of  $\pm 0.05\text{ g}$ . The open loop straightness (robot movement without position feedback) was tested and analyzed to have a maximum average deflection of  $\pm 4.2\text{mm}$ , which reaches the target of  $\pm 1\text{cm}$ . The accuracy of moving 1cm intervals via the stepper motors was found to be an average of  $9.51\text{mm} \pm 0.78\text{mm}$ , which reaches our target of  $10\text{mm} \pm 1\text{mm}$ . The target average error of 1mm was met, and the average movement of 9.51 will either be adjusted or fixed during runtime using position feedback. The time for the robot to deposit one pixel and move to the next was found to be a minimum of 1.128s, which reaches our target of 3 seconds. Lastly, the accuracy of the template recognition component to read the laser distance meters was found to be 83.75%, lower than the target accuracy of 95%. As discussed, once a webcam with better focus is used, and the webcams/laser distance meters are relatively fixed with mounts, this accuracy is expected to improve to within our target range.

Overall, we have successfully designed, developed, and tested an autonomous robot which is capable of autonomously generating a physical sand painting via three colours and without position/orientation feedback. Further work will be completed to fix current issues and deliver a robot with full position feedback and a reliable sand dispensing system.

## 4. Project Deliverables

### 4.1. List of Deliverables

#### Original Deliverables:

- A copy of the final report documenting the results of the project and recommendations
- A prototype version of the sand painting autonomous robot with the following criteria:
  - Autonomous
    - Robot should not require supervision after initial set-up and alignment
  - Portable
    - 50 lb or 23kg
    - Maximum linear dimensions (height + length + width) of 158cm
  - Produce a sand painting using 6 different coloured sands, covering an area up to 6m x 6m fully without sparse areas
- Perform in a variety of environments, both indoors and outdoors, and be resistant to corrosion.
  - It can be assumed that the surface the robot works on is dry, even, and level, and that there is no wind.
- A simple computer interface to the robot is to be provided which allows the user to input the physical dimensions of the room the robot is allotted, the image the robot is to paint, and other information to direct the robot.
- Software package with documentation for the robotic control and GUI
- A simple manual describing how to use the system and its various features, including a description of every component in the system
- Complete set of design drawings and CAD files

While the robot in its current state does not satisfy the original deliverables we will continue working on the robot until the project fair on March 8<sup>th</sup>. Currently the robot is able to deposit different colors of sand without using the lasers as position feedback. The mobility and sand deposition is fully implemented but not with the desired accuracy and reliability. The laser position system is designed but has not been tested. The software is almost entirely developed except for some minor bugs that need to be exterminated and a resizing function in the GUI (explained in section 2.1).

### 4.2. Ongoing commitments by team members

We have agreed with the sponsor to continue working on the robot until the project fair on March 8<sup>th</sup>. We will aim to complete the original deliverables detailed in Section 4.1 and solve the key issues in Section 2.

### 4.3. Financial Summary

**Table 3. Bill of Materials**

Item #	Description	Quantity	Vendor	Cost
1	Stepper motors (23KM-K066-00V)	4	Digikey	\$ 294.52
2	IR sensors (QRD1114)	6	Digikey	\$ 9.96
3	4" Omni-Wheels, 2 pack (P/N 276-2185)	2	Vex Motors	\$ 53.98
4	Arduino Mega 2560	1	RobotShop	\$ 59.95
5	Arduino Mega IO Expansion Shield	1	DFRobot	\$ 18.70
6	Pololu Motor Shield <u>OR</u> GeckoDrive G201	4	RobotShop	\$ 79.80 <u>OR</u> \$ 500.00
7	Xbee Module (MaxStream, 4214A-XBEE)	2	Digikey	\$ 42.66
8	USB - Xbee Adapter	1	RobotShop	\$ 24.28
9	Micro-servo motors (HK15178)	9	Hobby King	\$ 17.01
10	Sandtastik Coloured Sand	6 x 1lb	DeSerres	\$ 60.00
11	Laser Distance Meters	2	Amazon	\$ 164.60
12	Lexan (24"x48"x1/8", 8574K21)	1	McMaster-Carr	\$ 34.31
13	Aluminum (12"x12"x0.25", Alloy 6061)	1	McMaster-Carr	\$ 64.79
14	Motor shaft adapters	4	Project Lab	\$ 1.20
15	Nuts and bolts, screws	x	Project Lab	x
16	Retractable power cord	1	Rona	\$ 20.96
17	Delrin stepper motor mounts (12"x12"x3/16")	1	McMaster-Carr	\$ 23.13
18	Servo motor mounts (stainless steel)	6	McMaster-Carr	\$ 12.14
19	Tubing	0.5 ft	Project Lab	\$ 2.00
20	Vibrating motor (VPM2 disc motor)	1	RobotShop	\$ 3.36
21	Shipping / Waterjet costs	x		x
	Total:			\$ 987.35 <u>OR</u> \$ 1407.55

## 5. Recommendations

This robot could be enhanced through the following projects:

1. Automatic Sand Refilling – Due to the required mobility of the robot, it cannot contain all the sand for a 6m x 6m painting all at once. The current design needs a person to refill the containers whenever the robot runs out of a particular color. A valuable addition would be to allow the robot to refill the sand automatically. This could possibly be done by using a fountain drink design on the side of the canvas. The robot “plugs” into the refilling station and pushes a button to allow the sand to fill up the containers.
2. Wireless – Currently everything except for the Arduino communication is wired. This makes for bulk and reduced portability of the robot. Any of the components could be switched over to wireless except for possibly the servo and stepper motor power, as this would require a battery that would need to be recharged during painting.
3. Color Mixing – The robot uses six colors to paint with. This provides a rough representation of the original image but can be drastically closer if more colors are used. Since adding additional colors to the robot increases the size and weight, a more viable option could be to mix new colors on-the-fly. Some possible ideas are to mix the different colors of sand as they exit the robot or to use white sand and paint it the desired color before depositing it.

## 6. Operation Manual

### Warnings and Things to Note:

- Power supplies are rated to output a lot of current and may take a while for the power to dissipate fully.
- Always wait until the green LED light on the power supply is off before changing connections.
- Do not connect circuits when the power supply is active. Always connect circuits first, and then plug in power supplies.
- Be careful which wire goes to ground and which goes to power, as reversing polarity would cause damage.

**Table 4. List of screws**

Location on Robot	Screw Type	Diameter of screw	Pitch	Length of screw	Washer	Nut	# of screws
Sand canister deck to top end of 80-20 posts	Flat, counter-sunk head, hex screw drive	1/4 inch	1/20 inch	1/2 inch	Yes, Fender washer between canister deck and 80-20	No	4
Sand canister wall support to sides of 80-20 posts	Flanged-button head, hex screw drive	1/4 inch	1/20 inch	1/2 inch	Yes, washer between screw and outside edge of canister wall	80-20 inset	8
L-brackets (supports Lexan deck) to 80-20 posts	Flanged-button head, hex screw drive	1/4 inch	1/20 inch	1/2 inch	No	80-20 inset	4
Chassis deck to bottom of 80-20 posts	Socket head, hex screw drive	1/4 inch	1/20 inch	1/2 inch	No	No	4
Delrin wheel/motor mounts to chassis deck	Socket head, 4-40, hex screw drive	0.112 inch (4-40)	1/40 inch	1/2 inch	Yes; washer between spring and delrin mounts	No	16
Stepper motor to its adapter	Socket head, 4-40, hex screw drive	0.112 inch (4-40)	1/32 inch	1/2 inch	No	No	2
Stepper motor adapter to delrin motor mount	Socket head, hex screw drive	0.138 inch (4-40)	1/32 inch	1/2 inch	No	No	4

## Assembly

### *Pre-assembled components*

Pre-assembled components do not require assembly, and should not be adjusted unless required. In general usage, it should be sufficient for the user to just add/remove the sand canisters to the robot for assembly/disassembly and to leave the lower half of the robot intact after disassembly.

- Stepper motors / Motor mounts / Omni-wheels
  - Do not try to remove the stepper motors, delrin motor mounts, or the omni-wheels. The entire bottom deck should remain intact and can be packed away as a single unit.
- Tube fitting / Sand canister
  - The tube fitting at the bottom of the sand canister and the walls and ramps of the sand canister should not be separated.
  - Each individual sand canister should be removed from the robot, but the canisters themselves should not be dismantled.
- Sand shot servo motors / Canister deck / Tubing
  - Do not try to remove the servo motors attached to the canister deck, or the tubing the servos are attached to. Simply unplug the tubing from the sand canisters by pushing on the tube fitting's black ring to release the tubing and pull gently.
- 80-20 posts / L-brackets
  - The height the L-brackets are set at is determined by the angle of approach the exit tube would have with the ground and should not be readjusted.

*Full procedure to assemble robot (from beginning; generally would just add sand canisters):*

1. Check that all components of the robot are available.
  - a. Arduino Mega 2560 board with Xbee module
  - b. USB-Xbee module (to be connected to laptop, not robot)
  - c. 24V 100W power supply and extension cords (connected to wall)
  - d. 5V 50W power supply and extension cords (connected to wall)
  - e. 6 sand canisters with tube fitting
  - f. 6 sand canister lids with IR sensors
  - g. Canister deck with 6 attached servo motors and tubes
  - h. 4 aluminum 80-20 posts with attached L-brackets
  - i. Lexan deck (supports funnel)
  - j. Funnel / exit tube
  - k. Chassis deck with 4 attached stepper motors, motor and wheel mounts, and omni-wheels
  - l. 2 laser distance meters with servos (1 per meter to press button to prompt reading, and 1 with mirror attached to read distance in x-direction)
  - m. 2 webcams with USB extension cables
  - n. Screws, nuts, and washers necessary. (Refer to List of Screws)
2. Attach the 80-20 posts to the chassis deck using 1/4-20 socket head screws with the L-brackets upright with each pair facing each other.

3. Place the Lexan deck on the L-brackets of the 80-20 posts. The hole in the Lexan deck should be in the same corner as the oval hole in the chassis deck.
4. Fit the funnel/exit tube through the Lexan deck and the chassis deck, so that the funnel sits in the hole in the Lexan deck and the exit tube protrudes through the oval hole in the chassis deck, facing outward.
5. Attach the nozzle on the exit tube, so the opening of the nozzle is facing down.
6. Slide 80-20 insets into the slots on the outside edges of the 80-20 posts (should have 8 insets). These insets would be used to hold up the sand canister support walls.
7. Place fender washers on top of the 80-20 posts so the holes match.
8. Place the canister deck on top of the fender washers, matching the holes in the 80-20 posts. The canister deck should be placed with the countersunk holes facing up and the attached servos below. Make sure that the holes in the canister deck are in the same corner as the funnel/exit tube.
9. Screw together the canister deck, fender washers, and the 80-20 posts using flat-head, 1/4-20 countersunk screws.
10. Place the sand canisters on top of the canister deck, such that the tube fitting fits into the holes on the canister deck. This should form a 2 x 3 grid of sand canisters.
11. Take each tube connected to the servos and push it into the corresponding tube fitting on the bottom of the sand canisters.
12. On the outside edge of the 80-20 posts, attach sand canister wall supports using flanged-button head 1/4-20 screws, washers, and 80-20 insets. Position the wall supports as high on the 80-20 as possible to maximize support for the sand canisters.
13. [Laser distance meters and webcams placement to be added for final design in March. These should be attached to the motor chassis and should not require assembly.]
14. Connect power, Arduino, servos, steppers, Xbee module, etc as shown in figures below.

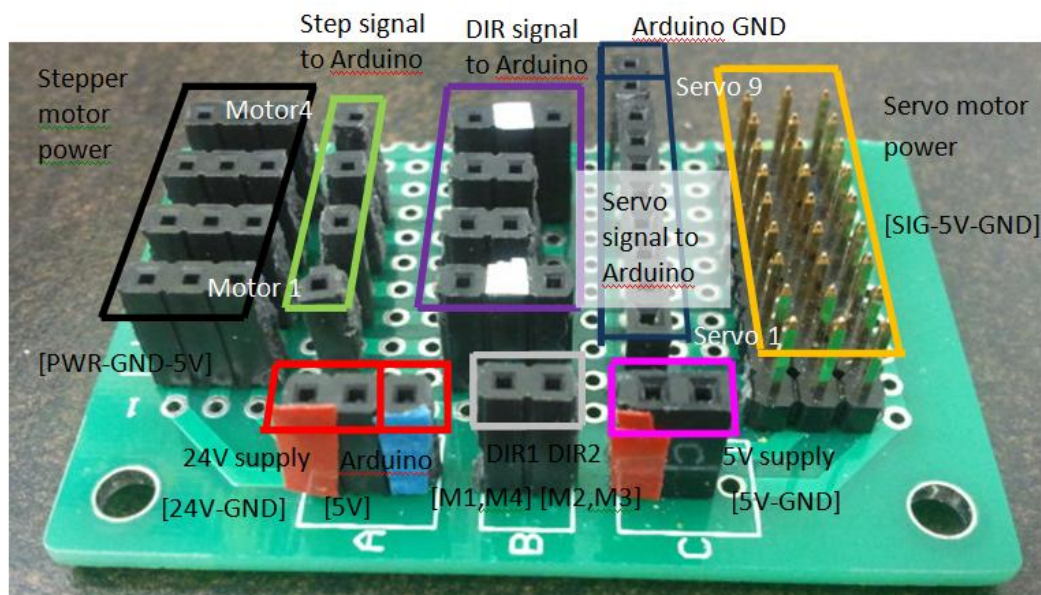


Figure 16. Power board wiring diagram

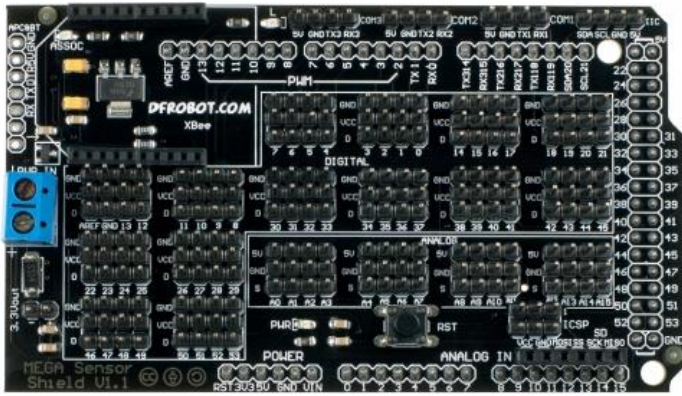


Figure 17. Arduino MEGA 2560 IO Expansion Shield

Source: [http://www.dfrobot.com/index.php?route=product/product&product\\_id=339](http://www.dfrobot.com/index.php?route=product/product&product_id=339)

15. Fill the sand canisters with the appropriate colour of sand, indicated on the canister deck. There should be a “map” of which canister goes to which servo, what colour sand to use, etc, glued on the canister deck. To ensure the colouring of the sand painting is correct, use the map.
16. Place gradient stick (with paddle) in the canisters after filling. Then place canister lids on top of sand canisters, with the gradient stick passing through the slot in the lid.
17. Plug in the power lines from the power supply. Make sure that the power supplies are switched off and not powered. Be careful which wire goes to ground and which goes to power, as reversing polarity would cause damage.
18. With the power supplies still switched off, plug in the power supplies to the wall outlet.
19. Switch on power supplies.
20. Turn on laser distance meters.
21. Connect webcams to laptop.

## Set up

1. After assembly, open the GUI software in Visual Studios.
2. Click on the green run button. The GUI should appear after compiling.
3. Under the Applications tab, upload the image to be painted. A preview of the sand painting should appear.
4. Input the room dimensions and press “Refresh”. The size of the sand painting, estimated time until completion, and the estimated amount of sand per colour should appear. If not, then a warning message should appear informing the user of the error.
5. Plug in the USB-Xbee module. The laptop will communicate to and receive information from the Arduino via wireless communication.
6. From Device Manager, find what the COM port the USB-Xbee module is assigned to. Input this number into the COM Port text box and click “Connect”.



7. Under the Testing tab, check if the stepper motors and servo motors are operational by selecting different directions and speeds to move the stepper motors, and by opening the servos. (Opening the sand servos should cause sand to come out of the nozzle.) When testing the button servos, make sure the laser distance meter is on and active before moving on.
8. After placing the robot in the initial position to start sand painting, press “Start” when ready.

## **Operation**

1. After setting up, the sand painting robot should be able to navigate the room and sand-paint by itself. However, there are instances where the robot may require assistance.
  - a. If a colour of sand is detected to be empty during the sand painting process, the robot will stop painting and send an email notification to the user to refill the sand colour
  - b. If position measurements are consecutively inconclusive or inconsistent, then the robot would assume there is something blocking the laser. The robot will stop painting and email the user to remove the obstacle.

## **Disassembly**

1. Switch off power supplies. Wait until the green LED on the power supplies is dim to unplug the power supplies.
2. Empty out the sand canisters.
3. Unplug the sand tubing by pressing on the black ring of the tube fitting and gently pulling. Remove the sand canisters. Keep rest of robot intact.
4. Unplug USB cables. Pack robot, power supplies, etc away. Close GUI and shut down laptop.

## **7. Appendices**

### **Appendix A – Software Code**

**See attached files.**

## **Appendix B – Technical Specifications**

All relevant datasheets for components used in the sand painting robot can be found in the shared P017 dropbox, under folder “Specs”. These files include:

- Arduino MEGA2560 Schematic
- Arduino MEGA2560 IO Expansion Board Schematic
- IR Reflective object sensor QRD1114
- 5V power supply VGS-50-XX
- 24V power supply VGS-100-XX
- A4988 Pololu Microstepping driver
- Stepper motors 23KM-KSTD
- Bosch DLR130K Laser distance meter manual
- Hobby servo Fundamentals
- Vex Robotics Omni-wheel

## Appendix C – Meeting Logs

### Pre-meeting Brainstorm:

- **Requirements:**

- Autonomous robot
- Highly detailed sand painting of 3m x 3m to 6m x 6m in area
- Sand covers floor
- Fully portable
- Versatile; usable in a variety of installations
- Work conducted on campus at UBC
- Simple computer interface

- **Questions:**

- Portability:
  - In terms of portability, is the sand included? How much sand is required? (Weight concerns)
  - Size of system? Weight?
- Setting/Environment:
  - What types of locations do "a variety of installations" entail? Indoor/outdoor? Flooring? Walls? Ceiling?
- Power: can it be assumed that robot can use power from an outlet provided? Or would it require a battery pack/external generator?
- Computer interface:
  - What should the program be able to do? Read in an image file to base painting on? Generate a painting based on generic patterns?
  - What much detail is required? What is the resolution? Smallest physical pixel size (around 1cm x 1cm)?
  - Software to plan and stock the sand?
- Sand quality:
  - Grain size? Shape? Coarse/fine? Uniformity? Colour? Grading?
  - Who would provide the sand? Standard type of sand? Specs on friction coefficient, etc
- Colour:
  - Pre-coloured sand? Sand to be coloured as it is deposited? If so, colour with paint/powder/ink pad/chalk block?
  - Custom colouring or "round" to the nearest generic colour? Colours may vary based on the different images used.
- Time limit:
  - How quickly does the painting need to be completed?
- Timeline
  - Sept 25 Project proposal due
  - Jan 9 Final report due
  - Mar 8 Project Fair
- Budget – how much?
- Deliverables
  - Downscaled prototype? Proof of concept?
- Possible to obtain sample images and some sand?

- **Thoughts/Comments:**

- How is the robot getting the sand? Does it hold a receptacle which needs refilling every so often? Holding sand could be heavy. Else, it needs to be able to retrieve the sand - a project within itself.
- Colouring sand could be challenging to get even colouring, strong pigmentation, etc.
- Nozzle:
  - Would metal tube be provided or similar nozzle to be produced in shop? Or a new design for nozzle?
  - Could have simple funnel/cone, with vibrator motor to shake gently
  - Need to control flow, width, blocking/releasing, etc.
- Camera feedback system:
  - Is it necessary? (For RC systems, yes.) Would require robot to be able to identify error and fix it
- Motion control:
  - Position is crucial - need to know how far the robot has moved
  - Range finder? Sensor? Stepper motor? Encoder?
- Communication/Control :
  - Arduino
  - C++
  - Bluetooth/Xbee

- **Design ideas:**

- Radial:
  - One end of track held up with wire to ceiling, fixed position, other end of track has wheel which drives in circular path. Robot moves on track, changing position as radius, outer wheel changes angle theta.
- Cartesian:
  - X-Y tracks with belts on a fixed structure, snap together, "IKEA" skeleton.
  - No fixed skeleton, X-Y tracks with wheels on each end of each track. Each track moves independently.
- Free roam:
  - "RC car" robot which drives in a spiral path, starting in the centre, depositing sand as it goes along
  - "RC car" robot which drives over sand with specialized tires to prevent sand transfer/disruption
  - Long track with two wheels and each end, moving in Y-direction. Robot moves along track in X-direction. Could also have fixed "anchor" in the centre to allow for spinning, radial motion (no longer free roam, but could be alternative radial design).

- **To do:**

- Design of robot/chassis
- Design of sand acquisition/holding/releasing system
- Design of movement of sand-deposit head
- Design of nozzle and sand deposition method
- Figure out how to control range of movement (finite increments and position)
- Software GUI design
- Coding for robot control
- Power/sensor/motor circuits

# Meeting Topic: Team AWESOME Launch Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:00 PM / Thursday, September 15, 2011 / JJ Bean, 146 W. Cordova St., Vancouver

## Agenda

- State what are the requirements of the project
- Address questions from pre-meeting brainstorm
- Outline budget, milestones, and deliverables
- Discuss design ideas
- Develop action plan

## Notes

- David is based in Vancouver. He will be out of town for 4 weeks at the end of September (Sept 23 - Nov 1), but will be available via Skype and email.
- Antoine is located in Toronto. He will be equally involved in the project and will be available via Skype and email.
- Robot will use 3-5 different pre-coloured sands.
- The maximum physical pixel size for the sand is roughly 1cm x 1cm.
- There is no time constraint on how long the robot takes to finish a sand painting. However, there should be a time estimate provided.
- There could be multiple robots operating simultaneously on the same sand painting.
- Could have a software program to provide a user interface to generate geometric shapes and patterns. For now, robot can read in an image/vector file (e.g. jpeg, bitmap, png, gif, tif)
- Assume for now the environment the robot is deployed in is indoor, with no wind, on a flat and level ground surface. Ideally would be able to implement outdoors as well (May need to set up a free-standing structure for outdoor use).
- Ideally, the robot fills the entire room with sand, but likely will have a border of white space where no sand is placed. Need to figure out the offset of white space in the room.
- Ideally, the robot would not require manual refilling. The robot should not require constant manual refilling, but acceptable if refilling occurs every 6 hours or so.
- Robot should be able to remember where it last stopped and the necessary information to continue, should there be an interruption in operations.
- If something happens during the robot's run-time which disrupts the set-up (e.g. a peg gets knocked over, power runs out, etc), can the system correct itself?
- The robot would have access to power from an outlet, but it is preferable to have a battery pack.
- Robot and required components (excluding sand) needs to be able to fit into a standard check-in luggage with size and weight restrictions, or be able to fit inside of a crate for easy shipping.

- It is not possible to alter the environment the robot is deployed in permanently (e.g. drill holes into the ground to anchor supports).
- Sand physics could be an issue
  - Need to be able to handle a range of variable conditions which may affect sand flow such as humidity, salt content, temperature, etc.
  - If robot "prints" sand by pixel, could create lots of small mounds of sand, which would have a different texture than desired.
  - Likely the sand would fall into conical mounds. There could also be overlapping issues for neighbouring pixels.
  - How can one smooth out these issues without complexity?
- Budget is estimated to be \$500CAD from the sponsors for now, but has room for growth depending on the project's needs.
  - Items less than \$50CAD can be purchased without pre-authorization with sponsors reimbursing later.
  - Larger ticket items (prices above \$50CAD) will need approval from sponsors.
- A full size, operational robot should be provided as a part of the deliverables.
- Recording images, time lapse videos, and robust documentation are strongly recommended
- Robot developed could have alternative uses and applications:
  - Planting, farming, fine control depositing
- Material of robot needs to be light and durable - could use aluminum
- Could have different nozzle sizes for applying different widths
- Think about how the sand is obtained, held, delivered, and potentially recycled/removed. Could have rotating hoppers and gates for control.

## Action Items

- Next meeting: Thursday September 22 @ 1:15pm, JJ Bean again
- Submit draft of proposal to David sometime early next week
- Look into "snap together" parts for belts, tracks, rails
- Buy pre-coloured sand, test and understand "sand physics", and select the right type to use
- Research whether sand painting robots have been implemented, existing robots that deal with sand, and other relevant robotic systems that have been done, and how do they do it.
- Test out nozzles for depositing the sand and select the right type to use for even distribution and good resolution size
- Work out logistics for refuelling the sand (Chambers loaded with sand with barrel rotating? Switching the sand being held?)
- Power - look into battery packs
- Look into reading and working with "vector" file types in C/C++ (e.g. Autocad)
- All the design "To do"s listed in Pre-meeting Brainstorm
- Make a Bill of Materials. Check with Jon to see if components can be provided by Project Lab.

# Meeting Topic: Team AWESOME 2nd Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:30 PM / Thursday, September 22, 2011 / Woodward's Building, 149 W. Cordova St., Vancouver

## Agenda

- Discuss position tracking and its issues
- Discuss budget increase
- Feedback on draft of project proposal

## Notes

- David leaves for London tonight (RE: out of town Sept 23 - Nov 1), but will be available via Skype and email. We will have our next meeting same time Thursday via Skype and use Teamviewer.
- Position Tracking
  - Rail system vs. independent robot
    - Size, weight, budget -> prefer independent since rails are bulky, heavy, and costly to machine and mill
    - Would need to have position tracking for independent robot
  - String encoder
    - Pros: easy to install, small, light, little time consumption, accurate
    - Cons: cost (~\$2000), would need to have multiple ones
  - Laser system
    - Laser mounted on rail (off robot)
      - For alignment, can't really tell distance
      - Need timing belt, gears, stepper motor, power, etc
    - Sonar doesn't have the resolution needed (~1cm)
    - Laser distance meter
      - Relatively cheap, would use 2 laser distance meters, 2 walls (x,y)
      - Need to hack the meters, or use a camera to read the display
      - Would need to have walls set up straight and at right angle
- Mat
  - Robot navigates x-y on mat, which has pixel images printed on it encoding position
  - Clean, easy to set up, needs to be durable, non-reflective, etc
  - Encoding system used can be a combination of library images, or a binary code
  - Need to keep mat still and stable (e.g. yoga mat, place mat, etc)
  - Need to print very high resolution
  - Would need a high resolution camera and a light. May need rubber grommets.
  - How is the camera communicating? Wireless range?



- Potential issues:
  - Matching seams would be difficult if they can't print full size =(
  - Would the robot's weight distort the mat?
  - Would there be crinkles in the mat from wear and tear?
  - What happens if the image rubs off?
- Settled on using 3 stepper motors, gear boxes, omni wheels, using solenoid valves, and mat system
- Use magnetic sensor to determine robot orientation
- Comment on pixel by pixel printing and sand mound issue - sand is very fine, tends to scatter. Depending on sand velocity, nozzle shape and size, should be able to control sand deposition.
- Refilling:
  - Too much sand in one application, so will need to refill
  - Likely will need to refill manually. If automated, then need a refilling station, and may not have time to do this.
  - Robot should have sensor which detects that it's out of sand in a colour and then email to notify that it's out.
  - Would be nice to have a lid on top of sand containers
- Better to power robot through power cord from an outlet. Should look into ways for the cord to be held up and not touch the sand at any point.
- Buying parts - email project lab and sponsors with bill of materials, have the lab order and sponsors reimburse
- Budget is increased to \$600CAD, with some room to expand

## Action Items

- Next meeting: Thursday September 29 @ 1:30pm, via Skype
- Complete project proposal (due September 25) and send copy to sponsors
- Install Teamviewer
- Make a BOM and send to Jon and sponsors for approval
  - Magnetic sensor
  - Sand level sensor
  - Webcam
  - Decide on stepper motors, wheels, and gearboxes (though Jon says they're available in the project lab)
  - Battery pack / Wind up power cord
- Look into banners and signage company and see what types of materials they use, whether they can print 6m x 6m, if there is visual skew over large area, and what the charge is
- Make a 6x6 pixel example image for the mat system
- Look into wind up power cord and decide how to deal with cord

### Action items from last meeting:

- Buy pre-coloured sand, test and understand "sand physics", and select the right type to use
- Test out nozzles for depositing the sand and select the right type to use for even distribution and good resolution size
- Power - look into battery packs
- Look into reading and working with "vector" file types in C/C++ (e.g. Autocad)
- All the design "To do"s listed in Pre-meeting Brainstorm
- Make a Bill of Materials. Check with Jon to see if components can be provided by Project Lab

# Meeting Topic: Team AWESOME 3rd Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:30 PM / Thursday, September 29, 2011 / via Skype

## Agenda

- Discuss position tracking and its issues
- Discuss Rehman's advice
- Discuss budget increase

## Notes

- David still out of town, but will be available via email (RE: out of town Sept 23 - Nov 1). Since David is in London, UK for the month, we will postpone meetings until he comes back to the same/similar time zone.
- Position Tracking
  - Rail system vs. independent robot
    - From last meeting:
      - Size, weight, budget -> prefer independent since rails are bulky, heavy, and costly to machine and mill
      - Would need to have position tracking for independent robot
    - Before meeting with David, had meeting with Jon
      - Alternative: x- direction rail system with 2 wheels that would roll in y-direction; robot rides along rail to dispense in x-direction. Weight distributed with free rolling wheels off of rail.
      - Would need to lock wheels on rail so it doesn't shift while the robot moves about
      - Less compound error if move in small increments - drift over large distance amplifies small errors
    - At meeting with David
      - Discussed Rehman's advice
        - Good advice overall
        - If laser sensors can be hacked properly, should be very accurate
        - Talked about 3-4 camera idea but not convinced about the accuracy since the cameras will be returning multiple estimates, bound to get errors in some areas
          - Calibration is easy but can reveal errors in the camera's construction (i.e. principal point)
          - Cameras need to be in exactly known positions and orientations

- Stereo camera would also work, but expensive
      - Minoru stereo camera is cheap, but has bad resolution
      - Could use two hi-res cameras, but need to make own enclosure and make sure everything is aligned perfectly (very difficult)
    - Overhead camera was mentioned - hard to make 6m x 6m scaffold perfectly; Pranav and Rosanna were on the Thunderbots team and are aware of this possibility
  - Laser system
    - Laser distance meter
      - Relatively cheap, would use 2 laser distance meters, 2 walls (x,y)
      - Need to hack the meters, or use a camera to read the display
      - Would need to have walls set up straight and at right angle
  - Decided not to use mat idea since the the mat would show through and doesn't leave a "seamless" feeling to the sand painting
  - Decided will not use magnetic sensor to detect orientation; instead use change in distance as a derivative to figure out orientation
  - Would likely not need gearboxes but need to do torque calculations to figure it out
  - Budget is increased to \$1000CAD, with some room to expand

## Action Items

- Next meeting: Thursday November 3 @ 1:30pm, via Skype
- Add to BOM
  - Sand level sensor
  - (Webcam?)
  - Battery pack / Wind up power cord
- Test out stepper motors and solenoid valves in Project lab to get a feel for what is needed
- Perform torque calculations to figure out how much is needed to move robot
- Start on SolidWorks design for robot

### Action items from last meeting:

- Buy pre-coloured sand, test and understand "sand physics", and select the right type to use
- Test out nozzles for depositing the sand and select the right type to use for even distribution and good resolution size
- Look into reading and working with "vector" file types in C/C++ (e.g. Autocad)
- Look into wind up power cord and decide how to deal with cord
- All the design "To do"s listed in Pre-meeting Brainstorm

# Meeting Topic: Team AWESOME 4th Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:30 PM / Thursday, November 3, 2011 / Eng Phys Project Lab, UBC

## Agenda

- Update project sponsors on progress
- Discuss work on mobility implementation

## Notes

- David is back from his trip. He will be staying in Vancouver for the rest of the project.
- Sponsor (David Long) visited project lab; discussed the robot, updated Sponsor on progress and showed him the chassis and wheels
  - Discussed sand colour again; the robot will be using 6 colours instead of 4 (CMYKGW: cyan, magenta, yellow, black, green and white)
- Implemented mobility system
  - Milled, drilled and threaded motor mounts. Mounted motors and wheels (2 regular wheels in x-direction, 2 omniwheels in y direction).
  - Added suspension to wheels to make robot more robust to uneven surfaces
  - Completed GUI interface that allows user to quickly test wheels at different speeds, directions (i.e. directional movement & rotation)
  - Set up Arduino to control motors and communicate to computer via Xbee.
- Working on GUI to take in user parameters and upload image to robot to print.
- Received laser distance meters; began to open
- Challenges:
  - Reducing backlash; The amount of play in the motor gearboxes is fairly large, given the resolution that we need to achieve (1cm x 1cm)
    - Backlash should be reduced when the weight of the sand is added to the robot
  - Dealing with uneven floors and slipping
  - Getting wheels to rotate at the same time without too much delay. (Currently, sending commands via serial communications, so it takes a while for the robot to get the message. Will have Arduino wait until all 4 motor signals are sent, then drive the wheels to reduce lag.)

## Action Items

- Next meeting: Thursday November 10 @ 1:30pm, via Skype
- Work on GUI interface
- Update Solidworks design to have 6 containers
- Work on getting information from laser distance meters
- Finish off mobility implementation

# Meeting Topic: Team AWESOME 5th Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:30 PM / Thursday, November 10, 2011 / via Skype

## Agenda

- Discuss GUI for image processing
- Discuss issues with backlash
- Discuss method on measuring sand level
- Discuss ordering materials for funnel and redoing mobility system

## Notes

- Waterjet machine is down for maintenance, so we will work on finishing designing and building the chassis
- Updated Solidworks design to have 6 containers
  - Model also redone to have reduced height, smaller funnel, output tubing and nozzle
- Software GUI implemented to do colour processing.
  - Some tweaking still required to improve and fix bugs. Results posted in P017 dropbox folder.
- Reducing backlash and dealing with uneven weight distribution
  - Solution: using stepper motors to reduce backlash (for 200 rev stepper motor, backlash calculated to be ~1.6mm if use current wheels)
  - Reduce size of wheels (2.75" wheels instead of 4" wheels).
  - David has approved this purchase
- Sand level measuring
  - Suggested method:
    - QRD reflective sensor mounted in the lid of sand canister, facing a hole in the centre of the lid;
    - A rectangular paddle is attached to a stick with a gradient colour sits on top of the sand and the stick goes through the hold in the centre of the sand canister lid. The QRD sensor reads the intensity of the gradient to determine the sand level. As the sand level decreases, the stick moves down.
    - Sponsor dislike wires, cumbersome
    - Pressure sensors are expensive
    - Need to know only when close to empty, but with continuous can allow for finer tuning and use info about weight change to control how much sand flows
- Pipe fitting - Will use a fitting to screw into the bottom of sand canisters; fitting will fit tightly over the tubing to seal joint
- Order Teflon sheet for funnel has been approved by sponsor
- Laser distance meter
  - Need to order ribbon cable+ connector
  - Email Bosch to get more help/ information about the LCD display/signal

## **Action Items**

- Next meeting: Thursday November 17 @ 1:30pm, via Skype
- Revise and update BOM
- Work on getting information from laser distance meters
- Revisit mobility implementation

# Meeting Topic: Team AWESOME 6th Meeting

## Attendees:

- David Long
- Tristan Miller
- Pranav Saxena
- Rosanna Yee

## Time/Date/Location:

1:30 PM / Friday, December 2, 2011 / via Skype

## Agenda

- Didn't meet last week as not much was done on the project.
- Missed David on Thursday, so rescheduled for Friday
- Update on project and discuss current hot topics

## Notes

- Update on project
  - Laser distance meters
    - Can power laser distance meters using lab power supply (5V)
    - Difficult to decode the signal to LCD display
    - Cut losses on hacking laser distance meters and use web cams to read display.
    - Need to press buttons on laser distance meters to prompt measurement
  - Stepper motors
    - Tested with full motion and sand deposition. (Video in P017 dropbox folder.)
    - Robot travels fairly straight with DC motors and steps in increments of 1 +/- 0.5cm. Error is too high for 1cm resolution (50% error)
    - Need to switch to stepper motors for better accuracy
  - Need to adjust the height of the funnel to allow for a steeper angle to ground so sand shots flow better and more consistently
  - Servo motors with sand canister design is working great
- Direction of project
  - Need to have complete design and a final report by January 9th deadline
  - May not be able to finish all components by deadline, but willing to work until complete
  - Would David prefer if we worked on making 1 or 2 aspects of the robot solid and robust, or have a generally functioning robot?
  - David would prefer a generally functioning robot by January
- Commitment to project
  - It may not be possible to finish everything on time by January. How much to add to project until we're "done" with it?
- We should revise the BOM and figure out how to make shaft couplers for stepper motors
- Exams start next week, so no meetings for next 2 weeks

## Action Items

- Next meeting: ??
- Revise and update BOM
- Switch to stepper motors

## 8. References

### Components research:

**Arduino Clone:** <http://www.digilentinc.com/Products/Detail.cfm?Prod=CHIPKIT-MAX32>

**Arduino Mega 1280:** [http://www.dfrobot.com/index.php?route=product/product&product\\_id=50](http://www.dfrobot.com/index.php?route=product/product&product_id=50)

**Arduino Mega I/O Expansion Shield:**

[http://www.dfrobot.com/index.php?route=product/product&product\\_id=339](http://www.dfrobot.com/index.php?route=product/product&product_id=339)

**Arduino Motor Shield:** <http://www.robotshop.com/ca/adafruit-motor-shield-kit-arduino-8.html>

**Xbee Module:** <http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail&name=XB24-AWI-001-ND>

**Xbee Arduino Shield:** <http://www.adafruit.com/products/126>

**USB-Xbee Adapter:** <http://www.robotshop.com/parallax-usb-xbee-adapter-board-2.html>

**Xbee tutorial:** <http://www.ladyada.net/make/xbee/modules.html>

**Arduino Mega 2560:** <http://www.robotshop.com/arduino-mega-2560-microcontroller-3.html>

**Sand:** <http://www.sandastik.com/sandastik-sand/sandastik-colored-sand>

**Stretch sensor:** <http://www.imagesco.com/sensors/stretch-sensor.html>

**Rotary encoder:** <http://parts.digikey.com/1/parts/957567-rotary-encoder-4-bit-gray-code-288xaa5k162a2.html>

**Stepper motor:** <http://www.robotshop.com/ca/rbsoy03-soyo-unipolar-stepper-motor.html>

**Stepper motor tutorial/ information:** <http://www.cs.uiowa.edu/~jones/step/>

**Omni-direction wheels:** <http://www.kornylak.com/wheels/rotacaster.html>

**Omni-wheel tutorial:** [http://www.societyofrobots.com/robot\\_omni\\_wheel.shtml](http://www.societyofrobots.com/robot_omni_wheel.shtml)

**Encoder:** <http://www.robotshop.com/ca/solutions-cubed-enc300cpr-easy-roller.html>

**Magnetic sensor:** <http://www.robotshop.com/ca/devantech-magnetic-compass-module-cmps03.html>

**Sonar sensor:** <http://www.robotshop.com/ca/maxbotix-xl-maxsonar-ae1-high-performance-sonar-module-1.html>

**String encoder:** <http://www.celesco.com/stringpots/index.htm>

**Accelerometer tutorial:** <http://www.dimensionengineering.com/accelerometers.htm>

**Linear solenoid actuators:** <http://www.mcmaster.com/#linear-solenoids/=e7izrt>

**Solenoid valve (966-1055-ND):** [http://www.crouzet.com/documents/pdf\\_pne/en/293.pdf](http://www.crouzet.com/documents/pdf_pne/en/293.pdf)

**Solenoid valve controller:** <http://www.robotshop.com/ca/content/PDF/datasheet-VC-2.pdf>

**Magnetic field sensor (XEN 1210):** <http://www.xensor.nl/pdf/files/sheets/xen1210.pdf>

**Laser distance sensor (FLUKE 411D):** <http://www.fluke.com/fluke/usen/Laser-Distance-Meters/411D.htm?PID=69331>

**Laser distance sensor (RIGID micro LM100):** <http://www.ridgid.com/Tools/micro-LM100>



## **Existing technologies:**

### **Tibetan Sand Mandala in progress:**

[https://www.youtube.com/watch?v=q6b7iro-qZ4&feature=player\\_embedded](https://www.youtube.com/watch?v=q6b7iro-qZ4&feature=player_embedded)

### **Hektor Motion System**

<http://hektor.ch/About/>

### **Automatic Graffiti Robot (y-axis rail)**

<https://www.youtube.com/watch?v=5uFsZEqCBV4>

### **Automatic Graffiti Robot (2 string system)**

<http://www.youtube.com/watch?v=qK0rrWFQIKQ>

### **Spiral Sunrise (sand dropping robot)**

<http://www.youtube.com/watch?v=FEEUWMxjHsk>

### **Sand drawing robot (finger in sand)**

<http://www.youtube.com/watch?v=ENp6viFrjlk>

### **Sand plotter robot (ball in sand)**

<http://www.youtube.com/watch?v=SLEfP6Ytgew>

### **Sand plotter (radial arms)**

<http://www.youtube.com/watch?v=s6FLZnb6Qy0>

### **Art plotter (XY, pen)**

<http://www.youtube.com/watch?v=Ji65HalG0D4&NR=1>

### **3D printer (binder ink printing)**

<http://www.youtube.com/watch?v=u7h09dTVkdw>

### **Sand casting (pre-treatment, wet sand)**

<http://www.youtube.com/watch?v=NNzaf6ilGtY>