Burr Puzzle Maker Manual

By:
Kelvin Lim
Brian MacPhee
Nikolas Campanello
Daniel Hasibuan

Submitted to:
Ralph Stirling
As partial fulfillment
To the manufacturing class
Final project

Table of Contents **Page**

Introduction

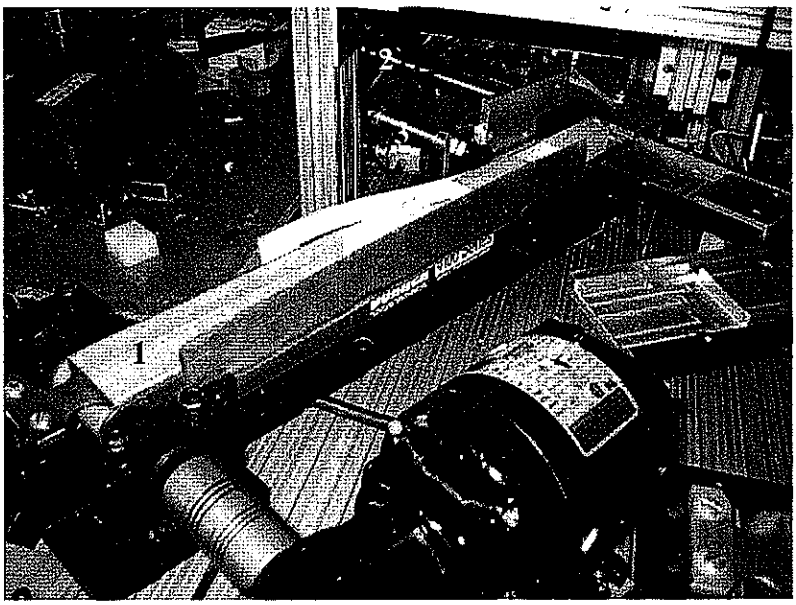
Our burr puzzle piece machine will cut notches on the faces of each block depending on the state logic and ladder diagram. The machine will need to meet criteria for loading and unloading of blocks, operation sequence, sound level, and safety. The following documentation will allow user to properly operate machine.

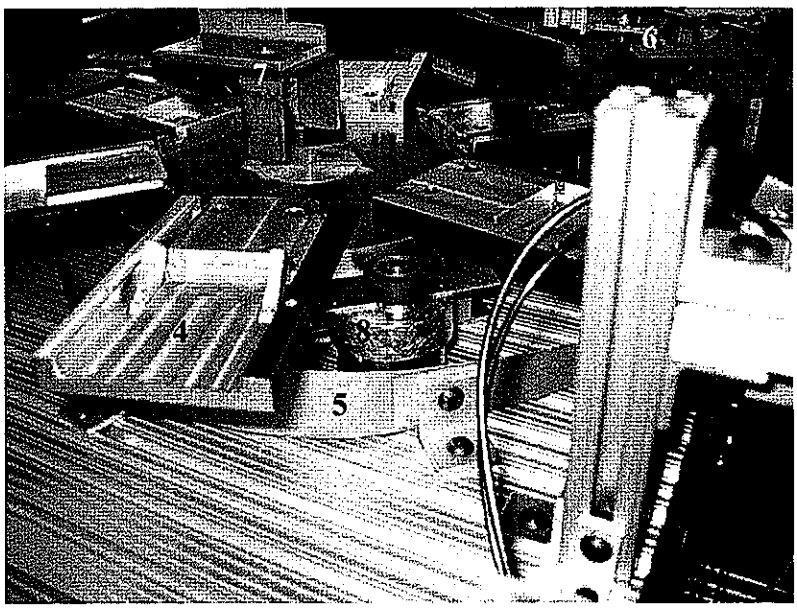
Design Criterion

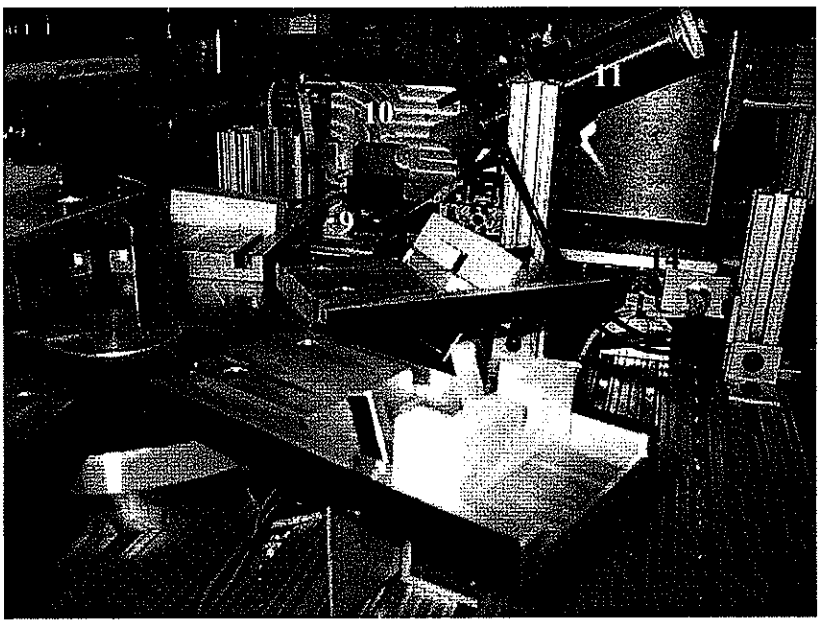
- Hardwood blocks with dimensions 0.75 x 0.75 x 2.25 inches, ± 0.003 (95%).
- Machine must be able to cut from 1-4 0.375 square notches on each block using the router.
- Blocks must enter and exit machine, via conveyor belt, and rotated if needed to be cut on more than one face.
- Stationary dust collection device will be incorporated into machine and not hinder manufacturing operation.
- A clearly visible shutoff switch is necessary to stop all production in the case of an emergency.
- Noise reduction measures be implemented in the design

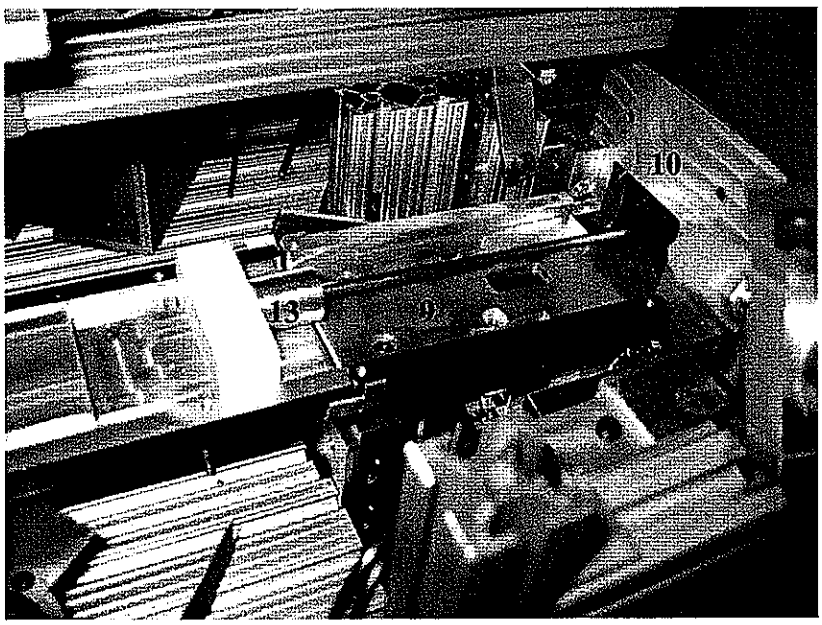
List of parts

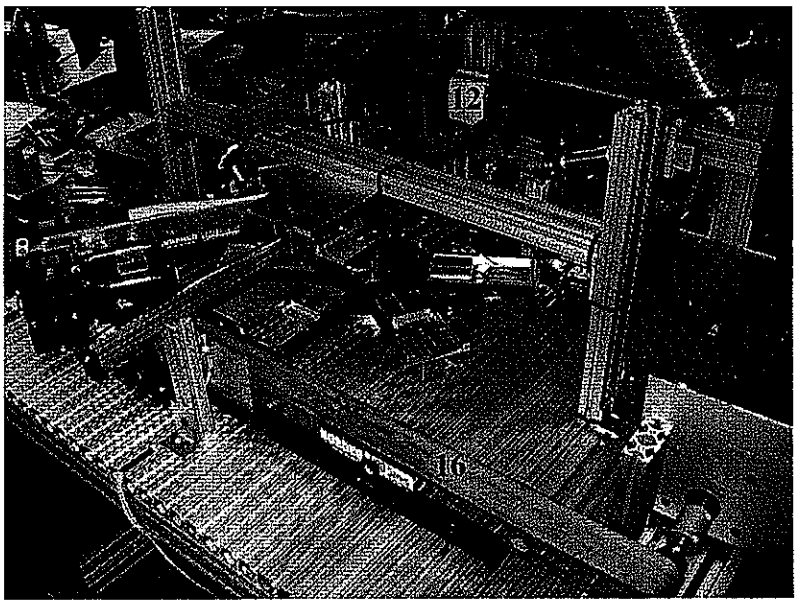
A. Mechanical parts
   1. Input conveyor
   2. Block loader
   3. Block stopper
   4. Turntable with flippers
   5. Flipper cam
   6. Turntable stop
   7. Turntable block extractor
   8. Turntable motor
   9. Bridge-to-carriage
   10. Carriage
   11. Block clamp
   12. Linear actuator
   13. Carriage ejection cylinder
   14. Router
   15. Sacrificial block
   16. Exit conveyor






11
10
9


10
13



B. Sensors/inputs

PLC inputs	Sensor
X_0	Bridge-to-clamp down
X_1	Load cylinder retract
X_2	Load cylinder extended
X_3	Block in conveyor
X_4	Turntable input position ready
X_5	Turntable block extractor forward
X_6	Turntable block extractor middle
X_7	Turntable block extractor backward
X_{10}	Router retracted in
X_{11}	Router extracted out
X_{12}	Carriage ejection cylinder extended
X_{13}	Carriage ejection cylinder retracted
X_{14}	Block in carriage
X_{15}	Counter sensor
X_{16}	Linear actuator input

C. Outputs

PLC output	Cylinder/Ouput
Y_0	Turntable stop
Y_1	Input conveyor block stopper
Y_2	Input conveyor block slider
Y_3	Turntable block extractor forward
Y_4	Turntable motor
Y_5	Turntable block extractor backward
Y_6	Bridge-to-carriage up
Y_7	Block clamp on
Y_{10}	Sacrificial block engaged
Y_{11}	Router cutter forward
Y_{12}	Carriage ejetion cylinder

Operating the machine

A. Starting the machine

! Before starting the machine, make sure the program is loaded up into the PLC and that the machine is free from debris or blocks or any person and their body parts.

1. Plug the air hose into the main valve.
2. Flip the PLC switch to "Stop".
3. Turn the large red button clockwise to pressurize the cylinders and to power up the machine and router.
4. Wait until the linear actuator finished initializing. The process is done when the carriage stops in front of the bridge-to-carriage. You can load blocks onto the input conveyor while the linear actuator is initializing. To load, simply place blocks on top of the conveyor as it moves until there is no more space for another one.
5. Make sure that the turntable block extractor is in the middle of the turntable. A good way to find this out is to make sure the sensor on top of the middle router that senses the turntable block extractor is on.
6. Turn on the vacuum system. Make sure the dust collection gate above the machine is open. To turn the vacuum system on, press on the large green button on the vacuum system motor.
7. Flip the PLC switch to "Run".

B. Clearing Jams

! In case of any emergency, PRESS THE LARGE RED BUTTON used to pressurize the cylinder system. This button cuts out all the air and electricity to the machine and to the router.

Jams falls under this category. Once the large red button is pressed, the system will need to be restarted the same way you restart the first time.

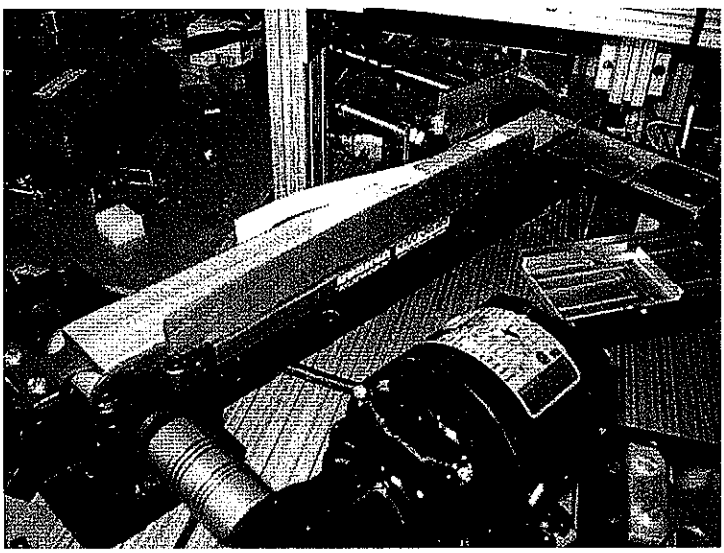
Clear any jams and find out the cause of the jam before restarting the machine again.

C. Changing puzzle types

Since we have a turntable that flips blocks as it turns, the turntable would have to go through two cycles to complete a puzzle. In a sense, the computer sees 12 blocks instead of 6 blocks. To change the table that contains all the cutting positions, turn on the PLC in term mode. Open the DirectSOFT32 program; go to tools, and the memory editor. Change the values in location V1200 to V1213. There are sixteen places for ones and zeros. The last 4 places specify which slots are to be cut. There are 6 slots to the width of each block. The middle four will be the ones being cut. For example, 0000000000001111 would mean that all four middle slots of the block will be cut.

Major sections operations

A. Loading blocks

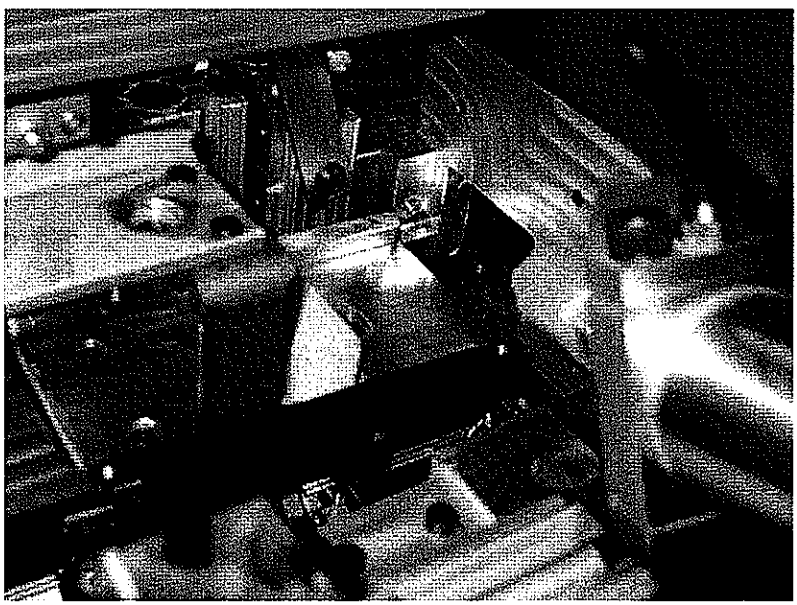


Blocks get unto the turntable from the conveyor one at a time, six blocks per cycle. It will input six block to the turn table, one for each slot, and waits as the turntable goes another full rotation before it inputs the next six blocks in to be cut.

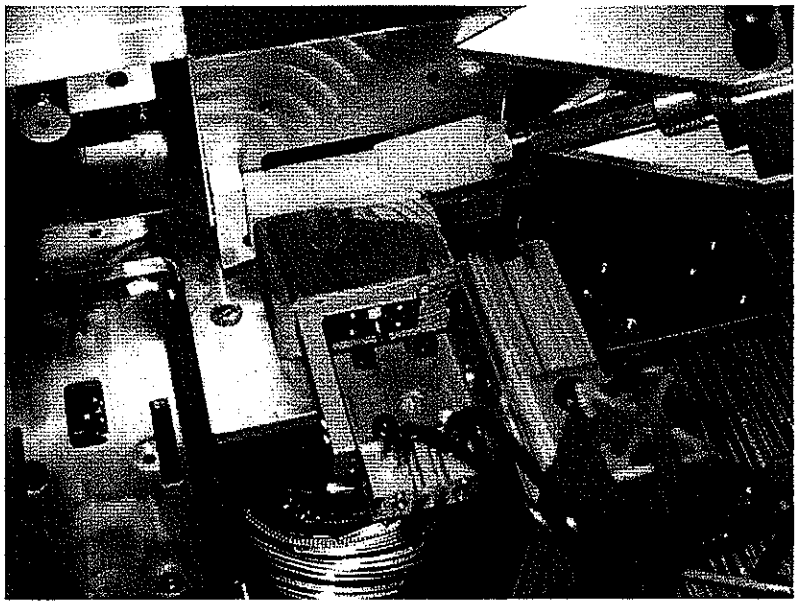
B. Turntable

The turntable holds six blocks. Each slot has a flipper mechanism so that every single block gets flipped. The use of this type of turntable can decrease production time if a second router is used.

C. Cutting blocks



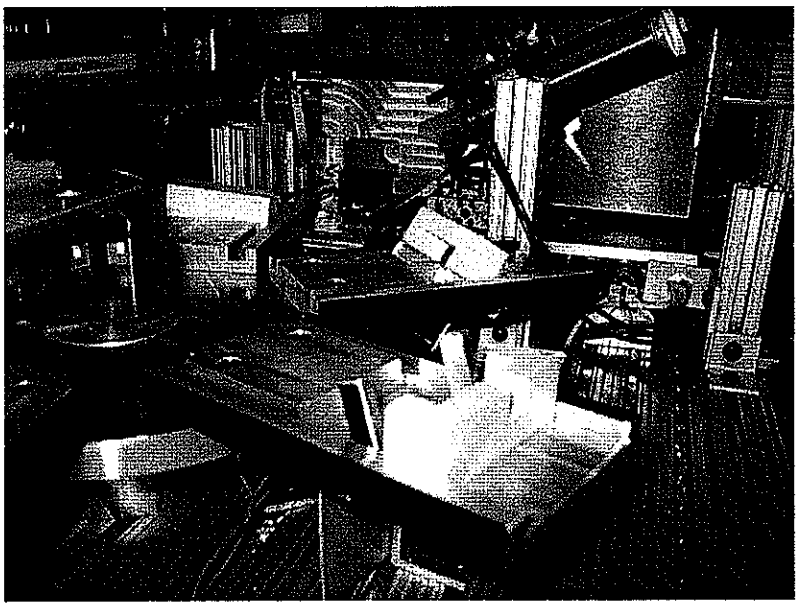
Blocks are fed up into the carriage using the turntable block extractor when it moves forward (towards the carriage) and the bridge-to-carriage is up.



The block gets clamped with block clamp and the bridge-to-carriage goes down to allow the block being moved to its cutting position. When it gets there, the sacrificial block slides up against it and the block is cut with the router moving forward.

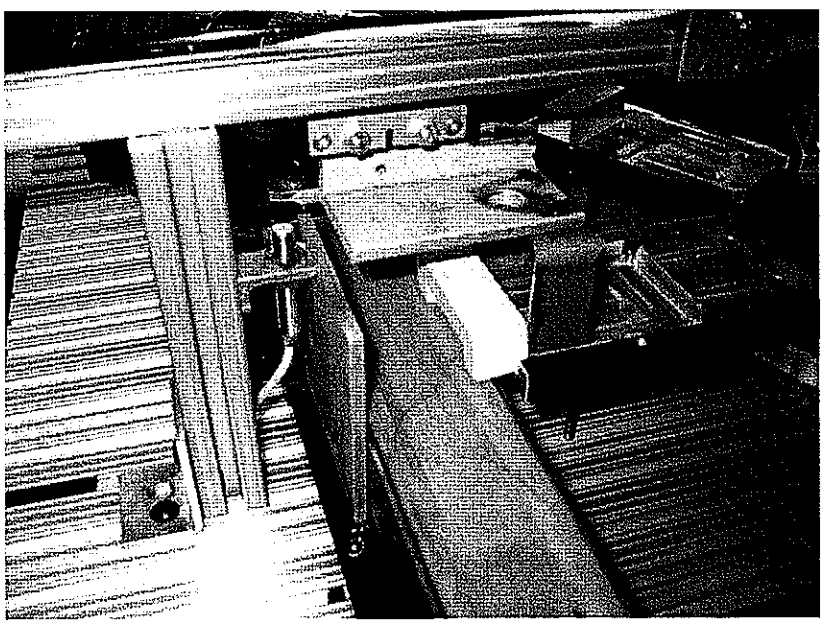
Once the block is cut, the carriage returns to home position, which is right in line with the bridge-to-carriage and the bridge-to-carriage lifts up. Once the big clamp releases, carriage ejection cylinder extends and pushes the cut block back onto the slots on the turntable.

D. Flipping blocks



As the turntable rotates, the flipper mechanism on each slot catches the flipper cam, slowly rotating the blocks as the turntable rotates. After a 120 degree rotation of the turntable, the blocks would be 90 degrees rotated

E. Extracting blocks



After the blocks are cut (twice if applicable), on the second round of the cycle, the block will be ejected by the turntable block ejector onto the exit conveyor.

Machine Specification

A. Input block dimension
Hardwood blocks with dimensions 0.75 x 0.75 x 2.25 inches, ± 0.003 (95%).

B. Output block dimension
One to four 0.375 inches square notches on either one or two sides of the blocks.

CAD pictures

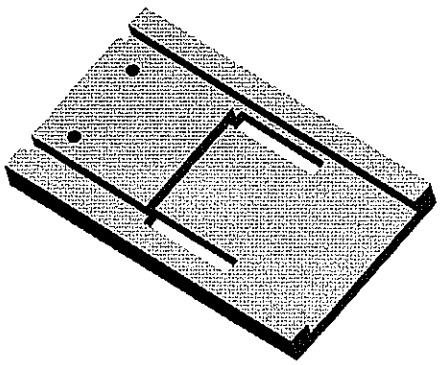


Figure 1 Turntable slots

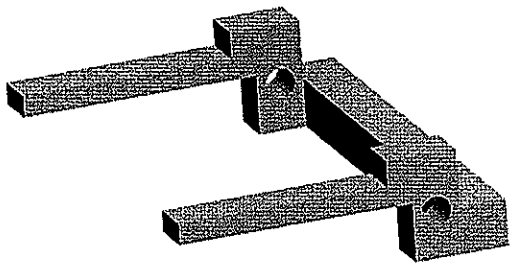


Figure 2 Flipper mechanism

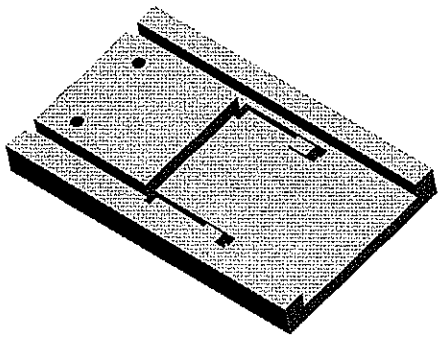


Figure 3 Turntable slot with flipper

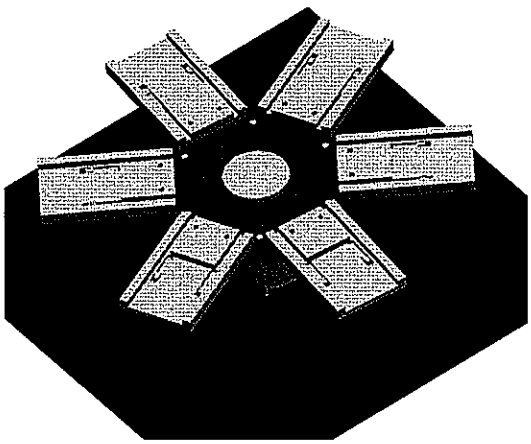


Figure 4 Turntable assembly

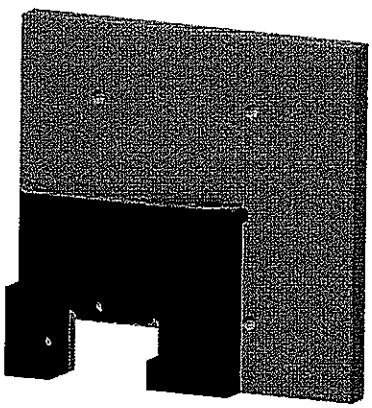


Figure 5 Carriage assembly

Suggestions for future improvement

Clearing all the bugs from the program would be the first step. Our machine, at its current state will not cut a whole puzzle pieces. It will cut blocks but sometimes in the middle of the program, the program stops.

Out machine have a lot of parts that were manufactured with the CNC machine. We designed our tolerance to be quite small, small enough to reduce our margin of error during installation. Because of this, if there were a little deviation our machine will not work as smoothly. In the future we would suggest having adequate margin of error in the tolerances during installation and machining to allow flexibility.

Our project was designed to have a mass production system in mind. As we see in Nelson Irrigation manufacturing line, they use mainly turntables. Although we decided to use turntables before we went to Nelson Irrigation, it seems that turntables are the way to go for multiple processes for a part. Our machine is designed to have a continuous flow of blocks to keep the program running.

It is a team project, but it would be more efficient if the different tasks are given to specific people. Specialization would work better in the long run rather than have everyone working on the same thing. One example is programming. It would be easier for one person to debug a program rather than have three different people working on it at different times.