Professor: Mr. Joseph Chretien Jr. **Student:** Matthew Terrance Carlos Boyea

Introduction

This project will expand upon the procedures which were developed last semester with a new mock industrial project. It will follow the same development procedure as the last project, but it will broaden the repertoire of FANUC features which are interacted with. This project will use macros and/or sub-programs to modularize the code. This project will also use position registers and/or data registers to universalize movement and perform simple math calculations.

The premise of this project is to mimic CNC programming in a nondestructive way. Oftentimes in CNC programming, robots will have detachable and re-attachable tools which they'll swap out when developing a product (similar to how drill bits are attached & detached from a drill). A CNC device will then use these tools to machine a part out of wood, metal, or another material. Instead of machining a product in 3D space, our robot will be drawing on paper in 2D space; instead of switching and using drilling tools, our robot will be switching and using drawing tools (markers, etc.).

The goal is to create a set of programs which individually draw a shape via position registers, to create a set of programs which individually switch out drawing tools, and then write a program (DRAW_SHAPES) which uses these tools (and maybe a bit of math) to create an image.

Stipulations

Any safeguards which are written for a file or macro to prevent collisions with objects outlined in the Precondition must be designed to never run into these objects, even if the machine is midway through executing its last command. These safeguards will likely be reduced to one or two macros which are placed at the beginning of a file that safely jog the machine away from potential collisions.

DRAW_SHAPES Algorithm

Precondition: Draw surface S exists at PR[6] with safe X+30 and Y+40. Tool holder T's slot S1 exists at PR[7].

Postcondition:

Procedure:

The precondition is that a drawing surface has been placed on the ground in front of the base of the robot and three drawing utensils have been placed at the given locations in front of the robot.

Approach

Aa

PR[4] | Snapshot 1 | Dedicated to store a Snapshot of a position for math operations
PR[5] | Snapshot 2 | Dedicated to store a Snapshot of a position for math operations
PR[6] | Sharpie Draw Origin | Stores the 0,0 location of the sharpie tool at pressure 0
PR[11] | Utensil Holder 1 | Saves the storage location of the first tool
PR[12] | Utensil Holder 2 | Saves the storage location of the second tool relative to the first tool
R[1] | a | Variable a for calculations
R[2] | b | Variable b for calculations
R[3] | c | Variable c for calculations

R[4] | i | Variable i for calculations
R[5] | j | Variable j for calculations
R[6] | k | Variable k for calculations
R[7] | n | Variable n for calculations
R[8] | p | Variable p for calculations
R[9] | q | Variable q for calculations
R[12] | Current Draw Tool | Stores the current draw tool #
R[13] | Target Draw Tool | Stores the target draw tool #
R[14] | Target X Position | Stores the first drawing x pos
R[15] | Target Y Position | Stores the first drawing y pos
R[16] | Next X Position | Stores the second drawing x pos
R[17] | Next Y Position | Stores the second drawing y pos
R[18] | Target Pressure | Stores the target pressure of the drawing utensil into the drawing surface
R[19] | Next Pressure | Stores the next pressure of the drawing utensil into the drawing surface
M[1] | Open Grippers |
M[2] | Close Grippers |
M[3] | Draw Target Position | Go to target pos (R[14}, R[15])
M[4] | Draw Target Pressure | Go to target pressure (R[18])
M[5] | Draw Target Position & Pressure | Go to target position at target pressure

DRAW_TRGT_POS

DRAW_TRGT_PRESS

DRAW_TRGT_POS_PRESS

DRAW_POINT

```
GOTO pressure 0
GOTO target x, y
GOTO target pressure
```

DRAW_LINE

```
GOTO pressure 0
GOTO target x, y
GOTO target pressure
GOTO next x, y + next pressure
```