```
// Arduino Nano Co-processor Code
// Controls all machine microservos
// Takes commands through I2C connection to PIC

#include <Wire.h>
#include <Servo.h>

int i, j;

int gatePosition = 0;

Servo dispenserServo[3]; // 0 -> round, 1 -> flat, 2 -> long
Servo gateServo;

int box_position[3] = {20, 25, 0};
int mid_position[3] = {90, 90, 75};
int dump_position[3] = {150, 180, 140};

void setup() {

  Serial.begin(9600);
  Serial.println("Setup");

  Wire.begin(8);
  Wire.onReceive(receiveEvent);

  dispenserServo[0].attach(9);
  dispenserServo[1].attach(10);
  dispenserServo[2].attach(11);

  gateServo.attach(3);
  gateServo.write(60);
  delay(500);

  dispenserServo[0].write(mid_position[0]);
  delay(500);

  dispenserServo[1].write(mid_position[1]);
  delay(500);

  dispenserServo[2].write(mid_position[2]);
  delay(500);
}

void loop() {

}

void receiveEvent(void) {
  uint8_t x = Wire.read();
  Serial.println(x);

  int action = (x & B11000000) >> 6;
  int servoNum = (x & B00110000) >> 4;
  int number = (x & B00001100) >> 2;

  switch (action) {
    case 0:
      dump(servoNum);
      Serial.println("dump\n");
      break;
    case 1:
      dispense(servoNum);
      Serial.println("dispense\n");
      break;
    case 2:
      flipGate();
      break;
    case 3:
      ret(servoNum);
      Serial.println("return\n");
      break;
  }
}

void dispense (int dispenser) {
    dispenserServo[dispenser].write(box_position[dispenser]);
    Serial.println(box_position[dispenser]);
}

void ret (int dispenser) {
  dispenserServo[dispenser].write(mid_position[dispenser]);
  Serial.println(mid_position[dispenser]);
}

void dump (int dispenser) {
    dispenserServo[dispenser].write(dump_position[dispenser]);
    Serial.println(dump_position[dispenser]);
}

void flipGate() {
  if (gatePosition == 0) {
    gateServo.write(120);
    gatePosition = 1;
    Serial.println("Gate 0");

    delay(1000);
  }
  else {
    gateServo.write(60);
    gatePosition = 0;
    Serial.println("Gate 1");

    delay(1000);
  }
}
```