

Manual de operación para ROBOT EPSON SCARA T3

Practicante: Javier Alejandro Amador Sánchez.

Dr.: Roberto Salas.

Dra.: Eloina Rodríguez.

Fecha revisión: 01/08/2022

Contenido

Introducción	3
Recomendaciones	4
Advertencias	4
Especificaciones generales	5
Panorama del robot y su entorno	6
Sistema de coordenadas del robot	7
Secuencia de encendido:.....	8
Lenguaje de programación.....	10
Comandos más utilizados.....	11
Programación	12
Puertos digitales de entradas y salidas (I/O INPUTS).....	15
Implementación de Python Como Lenguaje de Programación Eficaz.	18

Introducción

El presente documento tiene la finalidad de dar a conocer los aspectos generales para una correcta manipulación del equipo por parte del operador, por lo cual debe ser leído a detalle ya que puntos importantes son mencionados y son la base para el buen funcionamiento de robot.

El robot tiene aplicaciones variadas y su uso es intuitivo, existen 2 formas de programación:

- Mediante la colocación manual del robot en distintos puntos y grabarlos, dicha acción toma tiempo para el usuario.
- Conocer las coordenadas (x , y) e introducirlas directamente con el teclado, sin necesidad de mover los brazos del robot.

La implementación para un desarrollo de programas se realizó con el objetivo de evitar localizar puntos en el área de trabajo del robot. Dicha mejora es **eficaz** en cuanto a la generación de un programa, pues solo se necesita conocer el radio de la pieza y el número de puntos de adhesivo a colocar.




Recomendaciones

Al utilizar por primera vez el robot, el operador debe realizar una inspección visual y de reconocimiento. Se deben identificar los controles principales, área de trabajo, paro de emergencia y además que esté libre de objetos que interfieran en el funcionamiento.

Cada usuario tiene la responsabilidad de entregar la estación del robot limpia, ordenada y apagada.

Advertencias

- **No** hacer fluir acetona por la válvula, primero consultar manual de limpieza y mantenimiento.
- Por ningún motivo se deben colocar partes del cuerpo y/o objetos dentro del área de trabajo del robot, mientras se encuentre en movimiento.
- **No** ejercer fuerza sobre los brazos y husillo.
- **No** colocar recipientes con líquidos sobre la estación (a excepción del contenedor de adhesivo debidamente cerrado).
- Evitar mover los brazos del robot en los límites de desplazamiento. (consultar Coordenadas y límites)

 WARNING	Este símbolo indica posibles daños físicos, incluso la muerte si no se siguen las indicaciones de operación.
 WARNING	A diferencia del anterior, este símbolo indica la posibilidad de sufrir una descarga eléctrica. No tocar las partes donde este símbolo este presente.
 CAUTION	Indica que: Si las indicaciones no se siguen correctamente se puede causar daño físico al operador y equipo.

Especificaciones generales

Ejes:

- **X, Y, Z:** con movimiento en base al sistema de coordenadas rectangulares (cartesianas)
- **U:** rotación del eje z.

Largo del brazo / Alcance:

- Eje 1 + Eje 2: 400 mm (225+175 mm)
- Eje 3 (Z): 150 mm

Repetibilidad:

- Eje 1 + Eje 2: +/-0,020 mm
- Eje 3: +/-0,020 mm
- Eje 4: +/-0,020 grados

Carga útil (kg):

- Máx.: 3 kg
- Nominal: 1 kg

Ambiental:

- Temperatura: 5 °C - 40 °C
- Humedad: 10 %- 80% (sin condensación)

Requisitos de energía:

- 100-240 V de CA

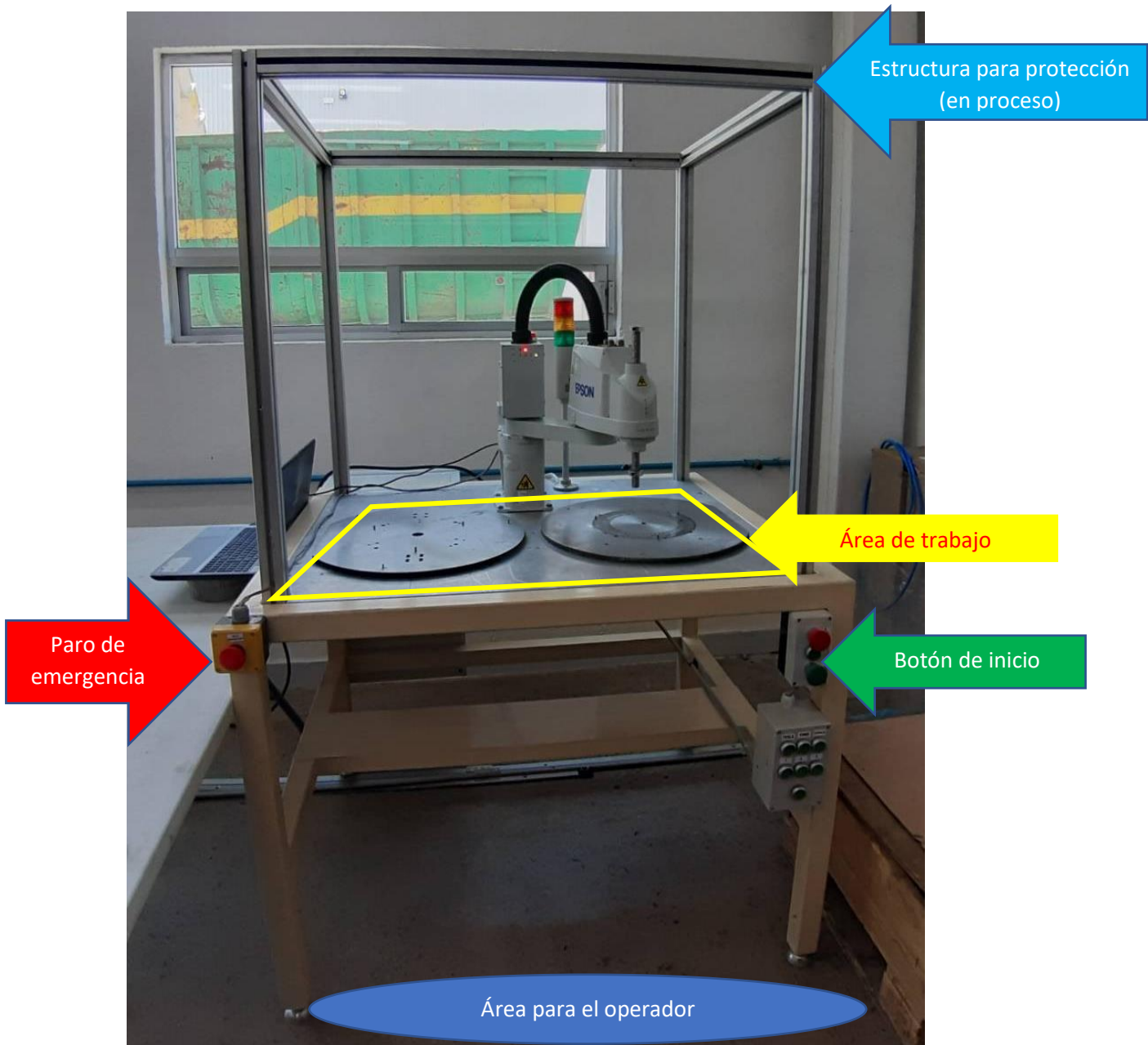
Software para manipulación:

- EPSON RC+ 7.0

Lenguaje de programación:

- SPELL+

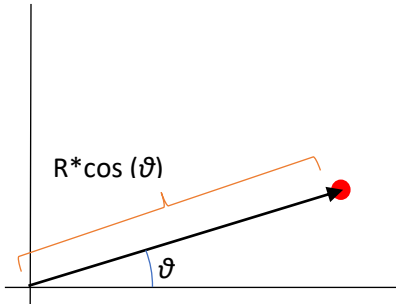
Panorama del robot y su entorno



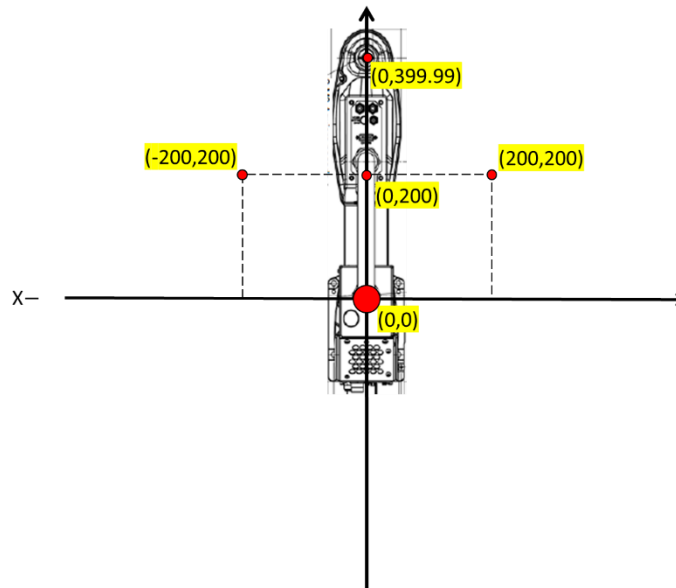
Sistema de coordenadas del robot

Las coordenadas rectangulares (cartesianas) que son las que conocemos como un sistema de ejes donde existen 4 cuadrantes. También se le conocen como coordenadas generalizadas y su interpretación es sencilla.

También existen las coordenadas espaciales que a diferencia su principal factor para ser ubicadas es un ángulo θ que se toma en cuenta a partir del eje de las abscisas (x) 0° y la distancia respecto al centro.



En la siguiente imagen se muestran las coordenadas del robot y su interpretación:



*Silueta del robot tomada del manual de EPSON

El robot se visualiza superiormente y el gráfico muestra las direcciones de los ejes, debe tomarse en cuenta hacia donde están los signos para evitar accidentes. Se recomienda realizar un ejercicio de ubicación de los puntos descritos en la imagen.

Secuencia de encendido:

Antes de encender el equipo, Asegúrese que no existan objetos dentro del área de trabajo y elimine cualquier condición insegura y evite un riesgo de trabajo, siga las instrucciones.

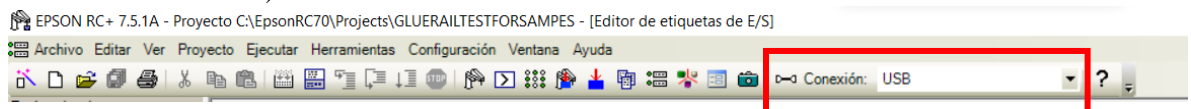
1. Conecte el cable principal a la red eléctrica de 110 v AC.
2. Gire el interruptor que se encuentra en la parte posterior del robot.



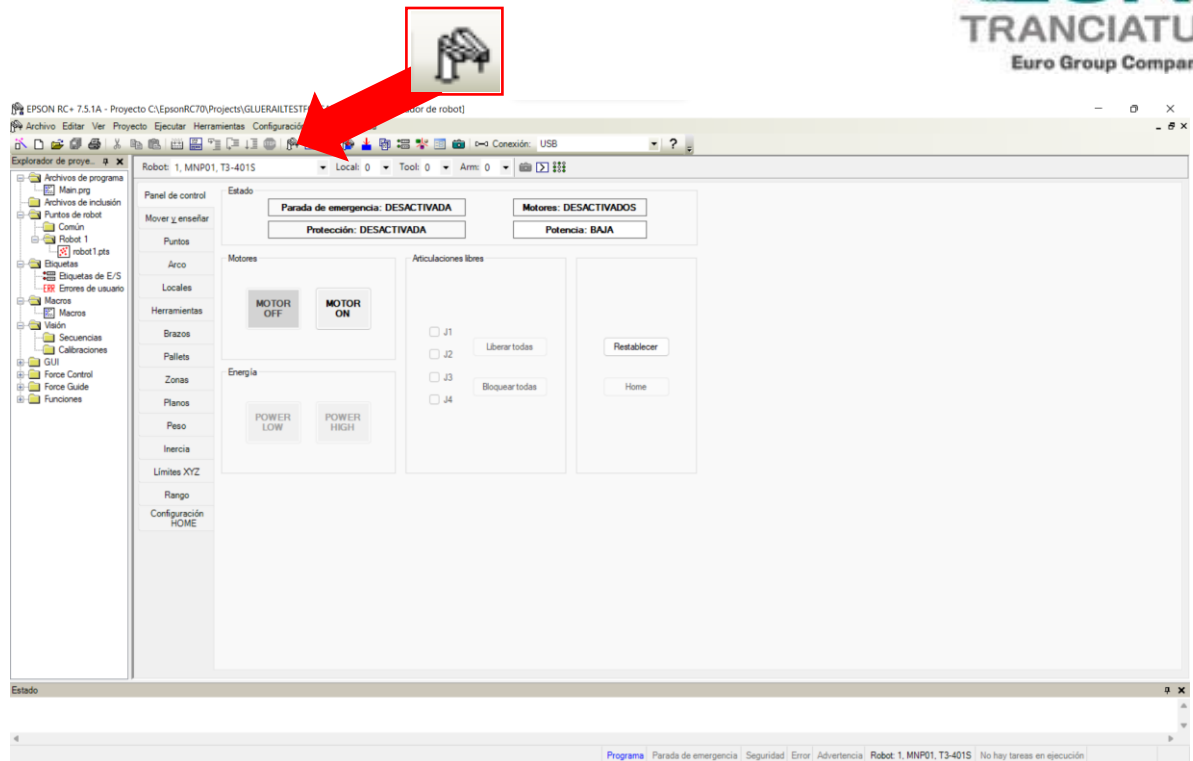
3. Los leds principales comenzaran a parpadear, espere a que se apaguen y solo se encienda el led "AUTO".



4. **Inicio de manejo de robot.** Abrir el software EPSON RC+ 7.0
5. En conexión, escoja USB> Modo manual (Conecte el cable USB a su pc y ejecute la conexión mencionada).

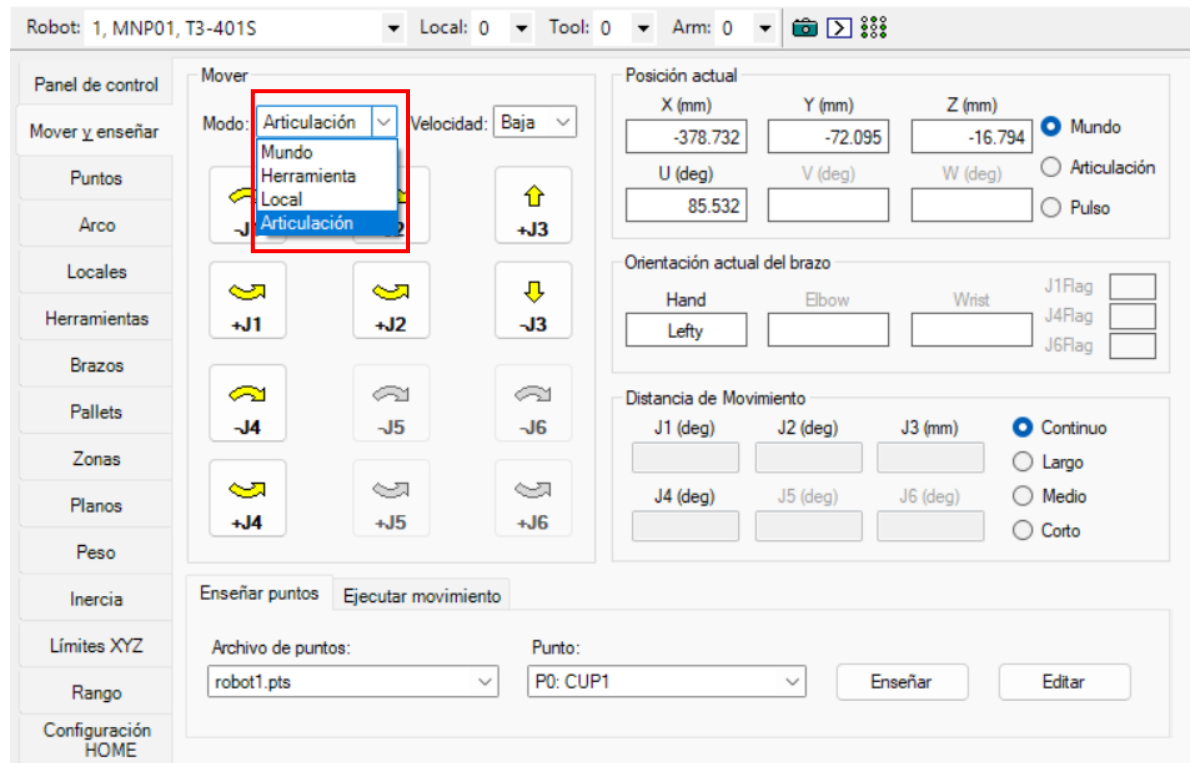


6. Una vez en comunicación, Abra el Administrador de Robot.



7. Active el motor y establezca la energía en alto (POWER HIGH).
8. En la pestaña “Mover y Enseñar” es donde podrá mover cada eje del robot, hasta el punto deseado.

Existen varios modos, pero el más comprensible es el modo articulación, puede probar los demás si lo desea.



Esta pestaña tiene información como:

- Velocidad para mover los ejes

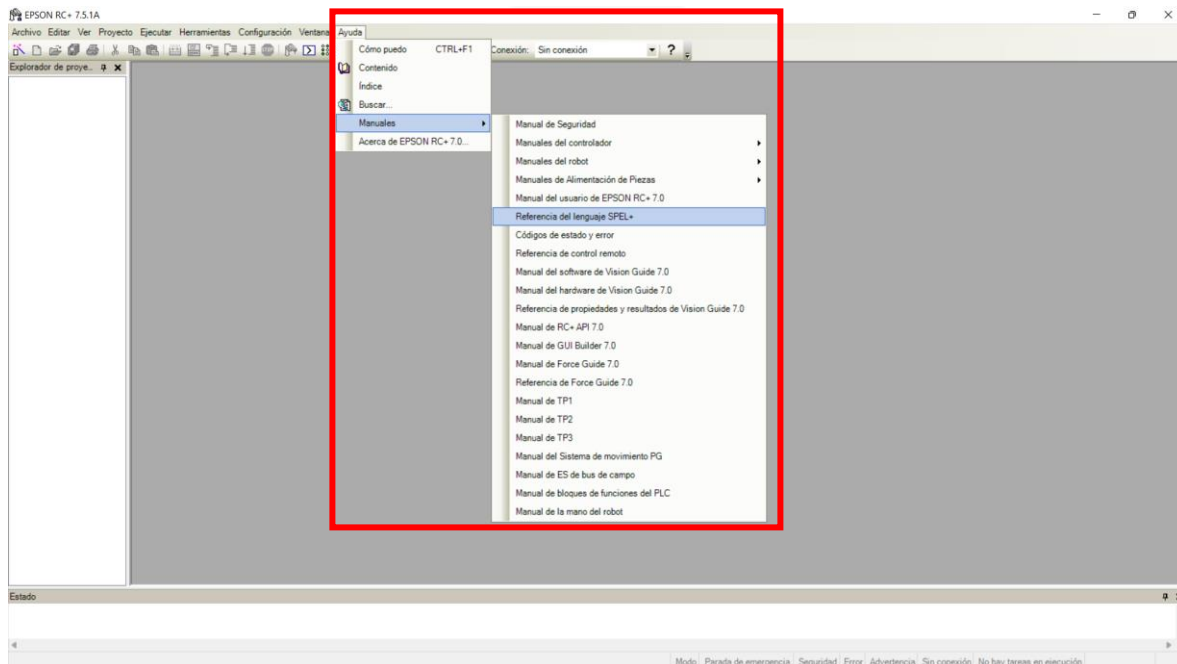
- Posición en coordenada
 - Modo de enseñar punto
 - Modo ejecutar punto enseñado
9. Para enseñar un punto basta con mover el robot a la posición deseada, una vez realizado esto, seleccione el punto que desea grabar (hay muchos puntos disponibles para enseñarle al robot)

Clic en enseñar y después inserte una etiqueta alusiva al nombre del punto.

10. Para ejecutar dicho punto es necesario moverse a la pestaña “Ejecutar movimiento”.

Lenguaje de programación

El lenguaje SPELL+ es propio de los robots EPSON, es intuitivo sin embargo el uso constante de sus comandos es una forma de aprenderlo rápidamente, así como grabarse cada código para la ejecución de un movimiento o tarea. El manual de referencia se encuentra disponible en el menú del software Ayuda>manuales>referencia de lenguaje SPELL+.

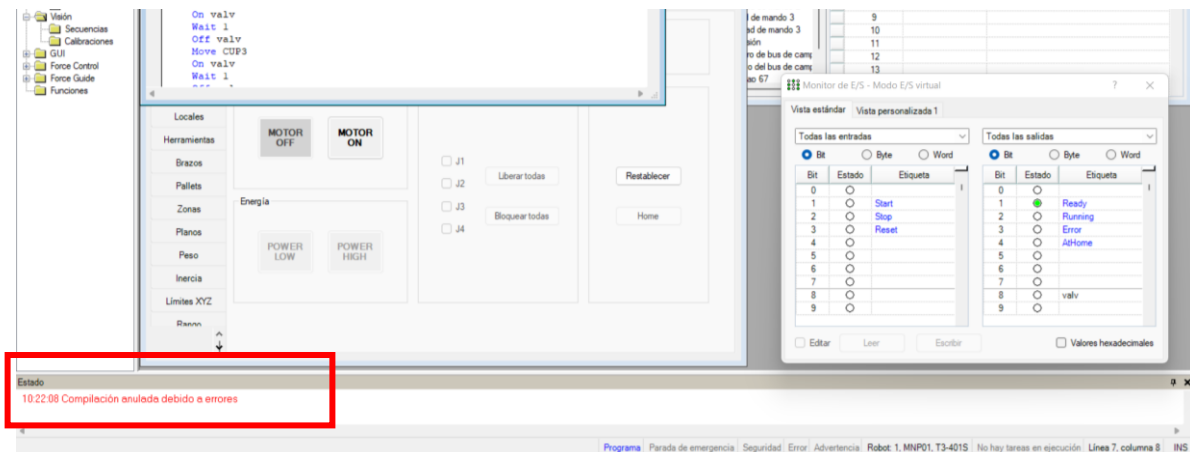


Comandos más utilizados

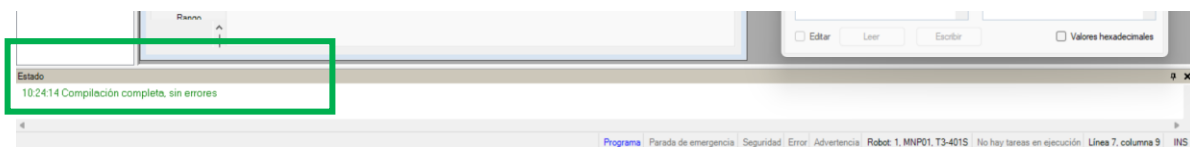
Los códigos más utilizados para la ejecución de un programa sencillo son los siguientes:

- **Jump:** Salta a un punto con un movimiento de punto a punto.
- **Move:** Mueve el robot con interpolación lineal.
- **Go:** Mueve el robot a un punto con un movimiento de punto a punto.
- **Home:** Mueve el robot a la posición de reposo definida por el usuario.
- **Speed:** Define/arroja la velocidad para los comandos de movimiento de punto a punto.
- **Accel:** Define/arroja la aceleración y desaceleración para el movimiento de punto a punto.
- **ON:** Enciende una salida configurada.
- **OFF:** Apaga la salida configurada.

Para más información se recomienda consultar el manual de lenguaje en donde se detalla cada comando y también se incluye un ejemplo de sintaxis para aprender a redactar y los elementos necesarios que acompañan al código. Para que un programa se ejecute con normalidad debe ser compilado, esto significa que se analizara en busca de posibles errores de redacción, si es que existen se despliega un mensaje de error en la ventana de estado:



Si el código fue compilado, procederá a cargarse en el controlador del robot.



Para iniciar el ciclo del programa se requiere de una acción START, esta se activa mediante el arreglo de entradas digitales, donde se hace llegar un voltaje (24 v DC) y se lee como bit en estado HIGH, lo cual inicia el ciclo. Lo mismo ocurre para STOP y RESET, dichas entradas están establecidas por defecto siendo las principales en cualquier programación.

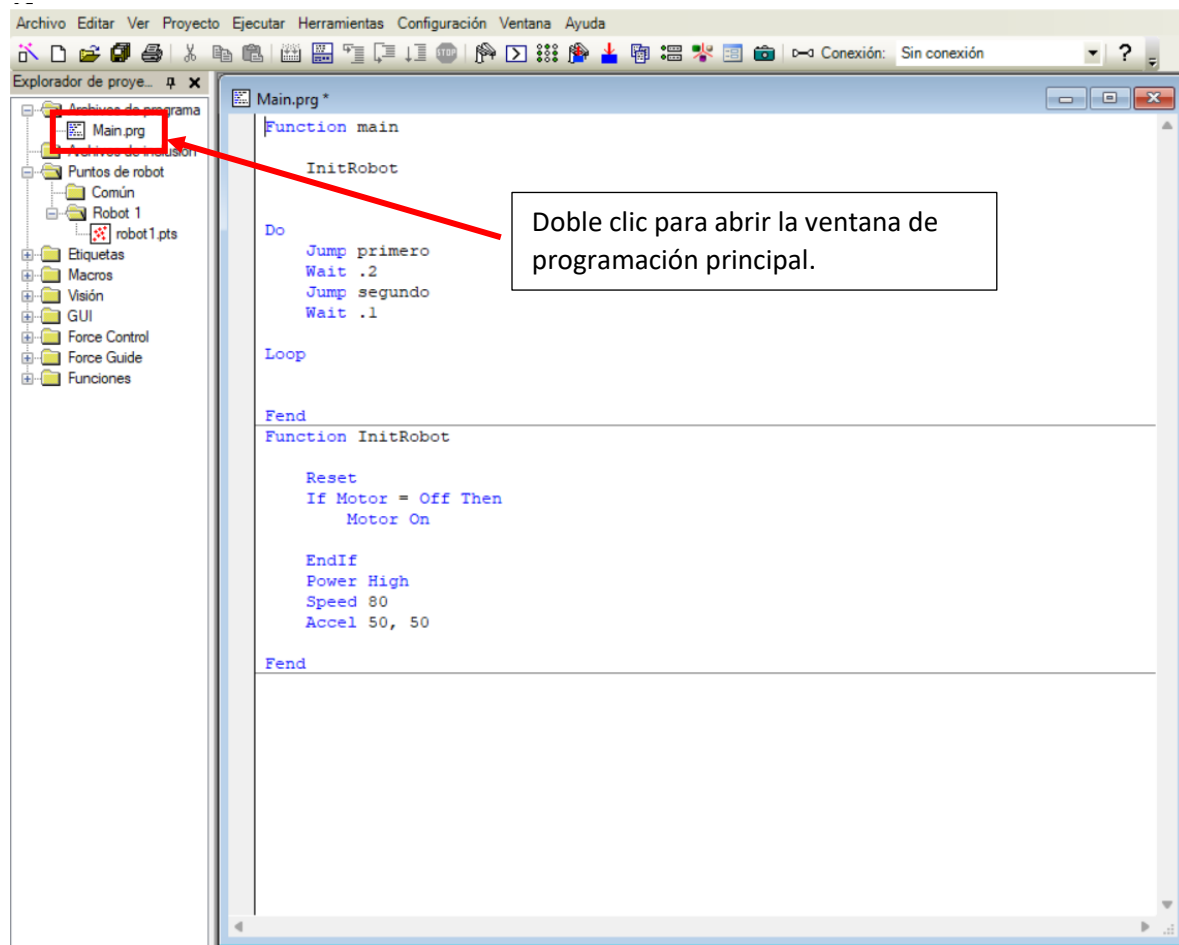
Se dispone de entradas para ser configuradas con la implementación de sensores y/o instrumentos extras que se requieran para realizar algún proceso. Así mismo se dispone de salidas para configurar

e igualmente se incorporan 3 salidas por defecto que son READY, RUNNING, ATHOME las cuales indican el estado en que se encuentra el robot.

Hasta este punto el robot puede ejecutar cualquier punto grabado.

Programación

EPSON posee un manual de referencia del lenguaje SPELL+, en el cual se encuentran todos los códigos para usar en el robot, además de una completa sintaxis de cada uno de ellos y su explicación a fondo para entender cómo usarlos.



En la ventana **Main.prg** se encuentra la programación o en su defecto el lienzo en blanco para comenzar a programar. El código mostrado comienza con un Function main que es un encabezado por default del programa.

Fend: es una línea que divide el programa en particiones a manera de separa programa principal e instrucciones.

InitRobot: es una función que se mandara llamar, como se puede ver esta separada porque contiene solo los parámetros para iniciar el robot: comienza con una condición **if** que sirve para

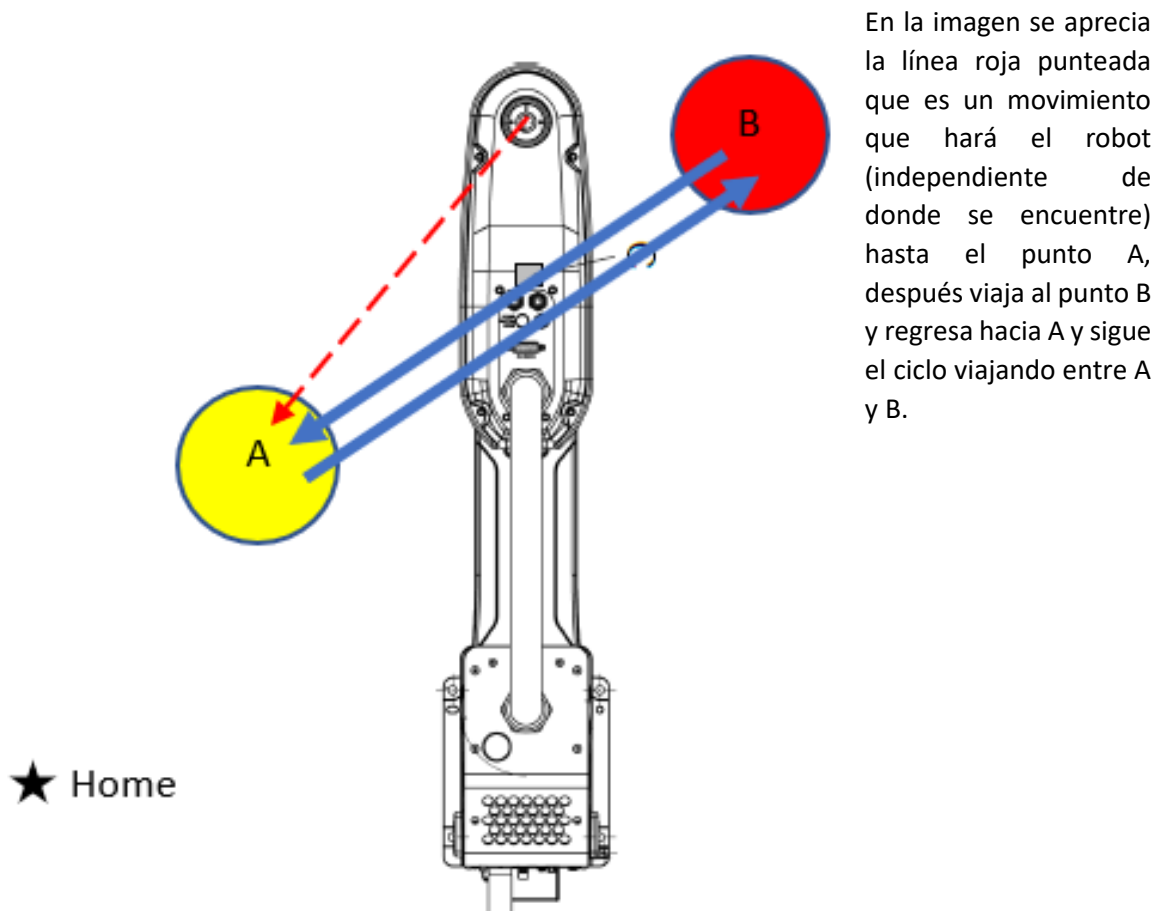
reinicio del robot, si el motor esta apagado se reinicia el robot y enciende el motor, una vez que el motor este encendido la condición termina con las instrucciones de velocidad y aceleración a la que se moverán los brazos mientras se ejecutan las instrucciones.

Do & Loop: para ciclar un programa comúnmente se usan estas instrucciones y entre ellas se coloca lo que se debe hacer. El robot lo hará en un ciclo infinito hasta que un factor externo lo detenga (sensores, desconexión de red eléctrica, fallo general).

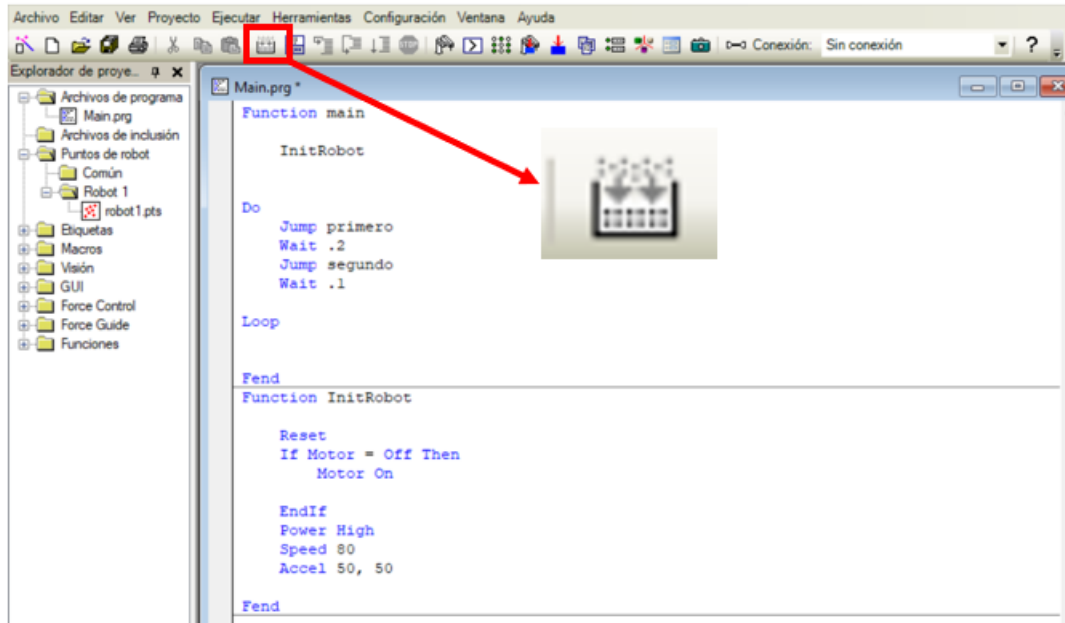
La serie de pasos que seguirá el robot es la siguiente:

El comando **jump** ejecuta un punto grabado en un movimiento lineal, en este caso se le esté ordenando que valla al punto denominado “*primero*” (A).

Seguido tenemos un comando **wait**, que determina el tiempo de espera en la orden anterior, esto quiere decir que cuando el robot este en el punto “*primero*” se esperará 0.2 segundos, después ira al punto “*segundo*” (B) y espera 0.1 segundos, y este ciclo lo hará infinitamente debido al **loop**.



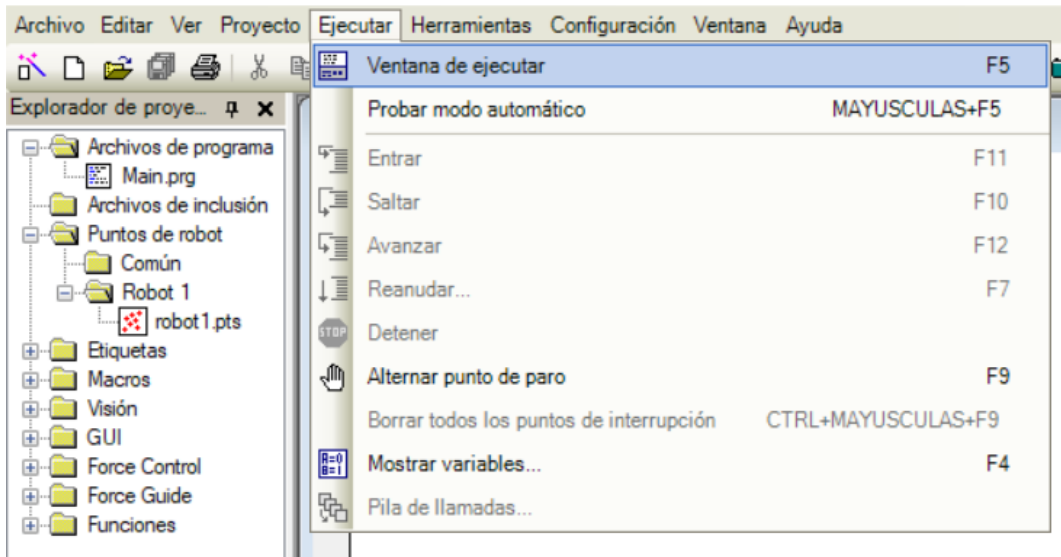
Para



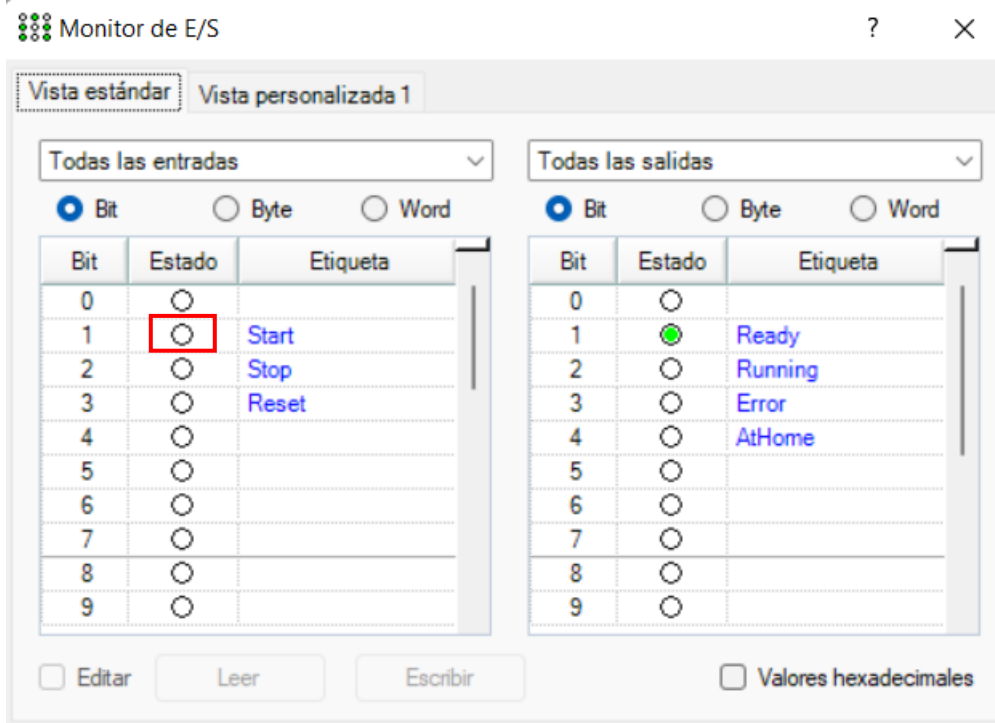
cargar el programa en el controlador del robot es necesario compilar, de esta manera el software analiza la sintaxis de la programación, si existe algún error no se podrá compilar hasta que sea corregido. El icono se para compilar es el siguiente:

S

Si el programa fue compilado con éxito, procederá a subirlo al controlador, una vez terminado este paso el robot está listo para ser ejecutado:



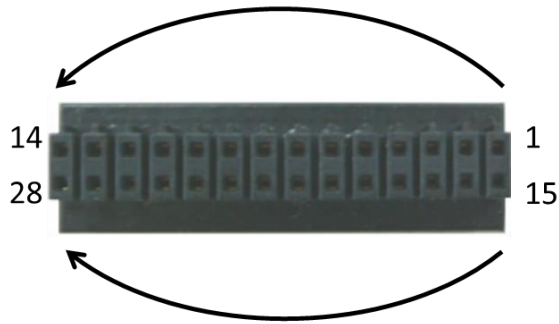
Una vez dentro de la “ventana de ejecutar” se puede probar el programa forzando las etiquetas entrada dando doble clic. Para iniciar el ciclo se debe forzar la etiqueta “start”.



Puertos digitales de entradas y salidas (I/O INPUTS).

En la parte posterior del robot se dispone de 2 arneses para controlar equipo. La alimentación es de 12 a 24 V DC y la corriente que se consume son 10mA.

ENTRADAS



El conector dispone de 28 entradas, pero por defecto de fabrica quedan disponibles 12 entradas para conectar sensores y/o equipo.

Se debe alimentar el común de las entradas con 0 V. El equipo o sensores debe mandar una señal de 24 V. Así se cierra el circuito y generalmente esta es la aplicación más común.

La siguiente tabla muestra el número de pin y su bit correspondiente:

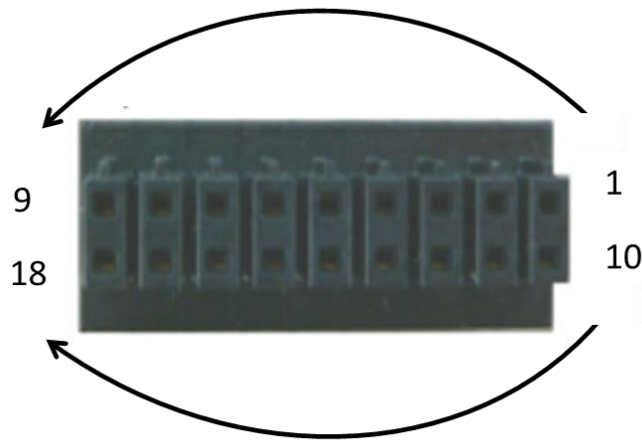
Pin No.	Signal Name	Pin No.	Signal Name
1	Input No. 0 (Start)	15	Input No. 1 (SelProg1)
2	Input No. 2 (SelProg2)	16	Input No. 3 (SelProg4)
3	Input No. 4 (Stop)	17	Input No. 5 (Pause)
4	Input No. 6 (Continue)	18	Input No. 7 (Reset)
5	Input common No. 0 to 7	19	Input common No. 8 to 15
6	Input No. 8	20	Input No. 9
7	Input No. 10	21	Input No. 11
8	Input No. 12	22	Input No. 13
9	Input No. 14	23	Input No. 15
10	Input No. 16	24	Input No. 17
11	Not Used	25	Not Used
12	Not Used	26	Not Used
13	Not Used	27	Not Used
14	Input common No. 16 to 17	28	Not Used

*nota: el número de pin es la posición correspondiente en el arnés, el bit es la asignación de entrada, NO LOS CONFUNDA.

- Los bits de 0 a 7 están destinados a los botones de uso principal y se alimentan con 0 V con el pin #5.
- Los bits 8 al 15 se disponen para adición de sensores y/o equipo y se alimentan con 0 V por el pin #19.

SALIDAS

En las salidas se alimenta el común con 24 V, debido a que se destinan para actuadores, generalmente.



Pin No.	Signal Name	Pin No.	Signal Name
1	Output No. 0 (Ready)	10	Output No.1 (Running)
2	Output No.2 (Paused)	11	Output No.3 (Error)
3	Output No.4	12	Output No.5 (SafeguardOn)
4	Output No.6 (SError)	13	Output No.7 (Warning)
5	Output common No. 0 to 7	14	Output common No. 8 to 11
6	Output No. 8 (EstopOff)	15	Output No. 9
7	Output No.10	16	Output No.11
8	Not Used	17	Not Used
9	Not Used	18	Not Used

Los bits 0 a 7 se alimentan con 24 V por el pin #5.

Los bits 8 a 11 se alimentan con 24 V por el pin #14.

En este modelo de robot solo disponemos de 4 salidas libres, por atributos siempre están designadas las salidas pausa, corriendo, error y listo. Dichas salidas no pueden ser modificadas por seguridad.

Para fines del proyecto se están utilizando las entradas con una botonera que contiene los 3 mandos principales:

- Inicio.
- Paro.
- Reinicio del robot.

Y en las salidas:

- Estado de la torreta.
- Válvula electroneumática.

Implementación de Python Como Lenguaje de Programación Eficaz.

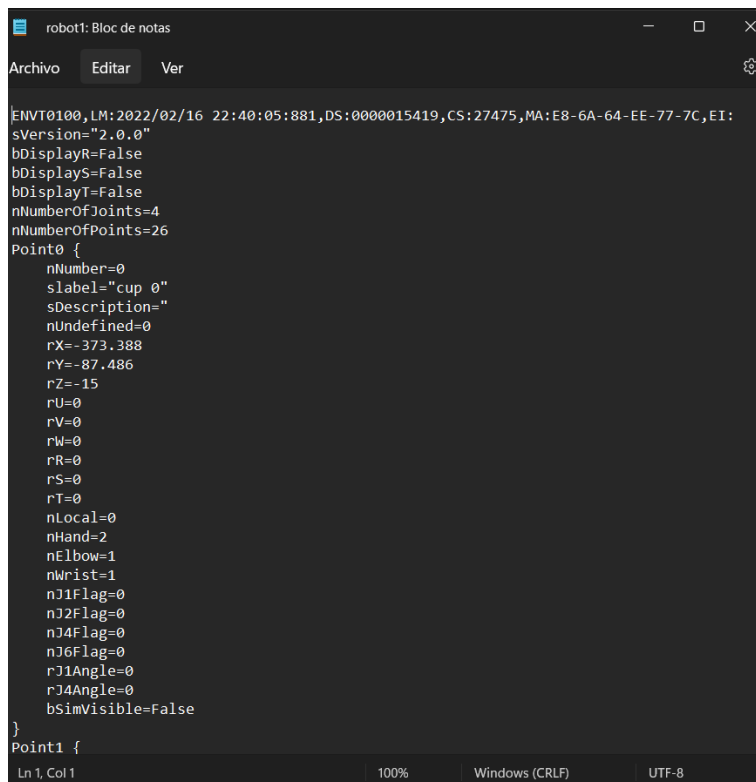
La forma de programar los robots EPSON T3 es de manera repetitiva y puede ser automatizada, debido a que los pasos para grabar un punto siempre tendrán la misma mecánica se opta por dividir el espacio de trabajo del robot en:

- Coordenadas globales.
Cuentan desde el área de home hasta cualquier punto dentro del espacio de trabajo.
- Coordenadas locales
Solo caracterizan los puntos de trabajo de la pieza y recordemos que la pieza está dentro del área de trabajo global.

De esta forma cada punto que este dentro de la pieza, esta referenciado respecto a las coordenadas globales. Un arreglo que se desarrollo fue crear una relación entre ambas coordenadas y así generar un archivo de puntos más dinámico. Si se requiere colocar un disco y marcar su centro como coordenada (0,0); es posible, aunque un solo disco para el área de trabajo seria desaprovechar al máximo el alcance del robot.

El objetivo es colocar la mayor cantidad de discos posible en el área de movimiento alcanzada por el brazo, existe el problema de establecer como (0,0) el centro de cada uno de los discos, la única forma de posicionar el robot seria manualmente y grabar cada uno de los puntos

EPSON RC+ 7.0 maneja un archivo de puntos con extensión “. pts” en donde guarda cada posición que se le señale. He aquí un ejemplo del archivo:



```
robot1: Bloc de notas
Archivo  Editar  Ver

ENV0100,LM:2022/02/16 22:40:05:881,DS:0000015419,CS:27475,MA:E8-6A-64-EE-77-7C,EI:
sVersion="2.0.0"
bDisplayR=False
bDisplayS=False
bDisplayT=False
nNumberOfJoints=4
nNumberOfPoints=26
Point0 {
  nNumber=0
  sLabel="cup 0"
  sDescription=""
  nUndefined=0
  rX=-373.388
  rY=-87.486
  rZ=-15
  rU=0
  rV=0
  rW=0
  rR=0
  rS=0
  rT=0
  nLocal=0
  nHand=2
  nElbow=1
  nWrist=1
  nJ1Flag=0
  nJ2Flag=0
  nJ4Flag=0
  nJ6Flag=0
  rJ1Angle=0
  rJ4Angle=0
  bSimVisible=False
}
Point1 {
Ln 1, Col 1      100%  Windows (CRLF)  UTF-8
```

Es un archivo en el bloc de notas que contiene los puntos en un arreglo como se muestra en la imagen superior. Tiene un encabezado principal que es exclusivo de cada programa y le asigna un código de autenticación, a continuación, vienen los puntos en orden, comenzando desde 0 como el primer punto y finaliza en 25, se observa que el número de puntos total es 26.

Para insertar el archivo .pts obtenido y que el robot lo acepte en el proyecto creado se debe hacer lo siguiente:

1. Entrar al software EPSON RC+ y crear un proyecto nuevo o abrir uno existente.
2. Se debe guardar un punto en una posición aleatoria con una etiqueta, posteriormente cierre el administrador de robot.
3. En el explorador de archivos de su PC valla a donde se encuentre instalado el software.
Para Windows generalmente se sigue esta ruta:

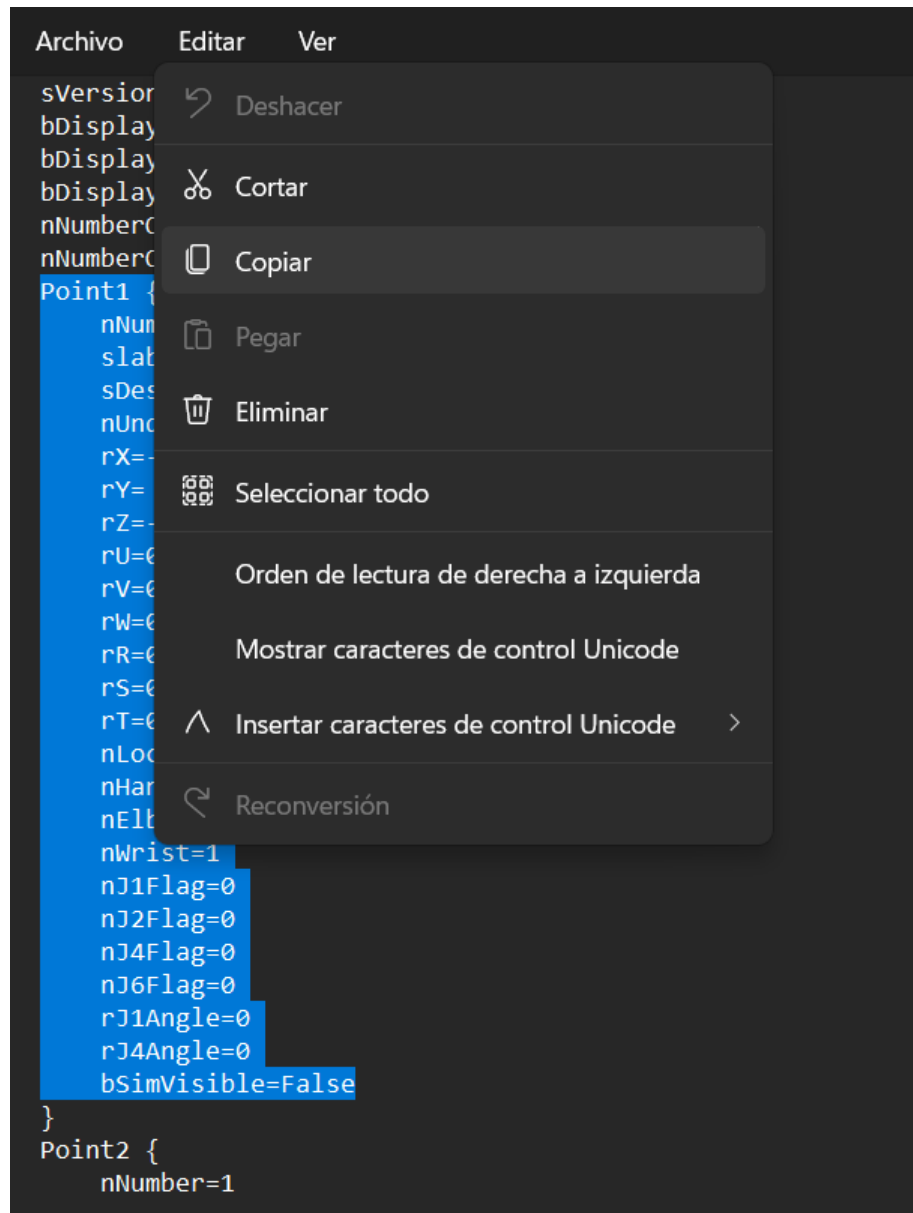
Este equipo > Windows (C:) > EpsonRC70 > projects >

4. Una vez que este en la carpeta “projects” busque el nombre de su proyecto y despliegue el contenido.
5. Ubique el archivo con extensión .pts y de clic en abrir. El archivo tiene la siguiente estructura:

```
robot1: Bloc de notas
Archivo  Editar  Ver

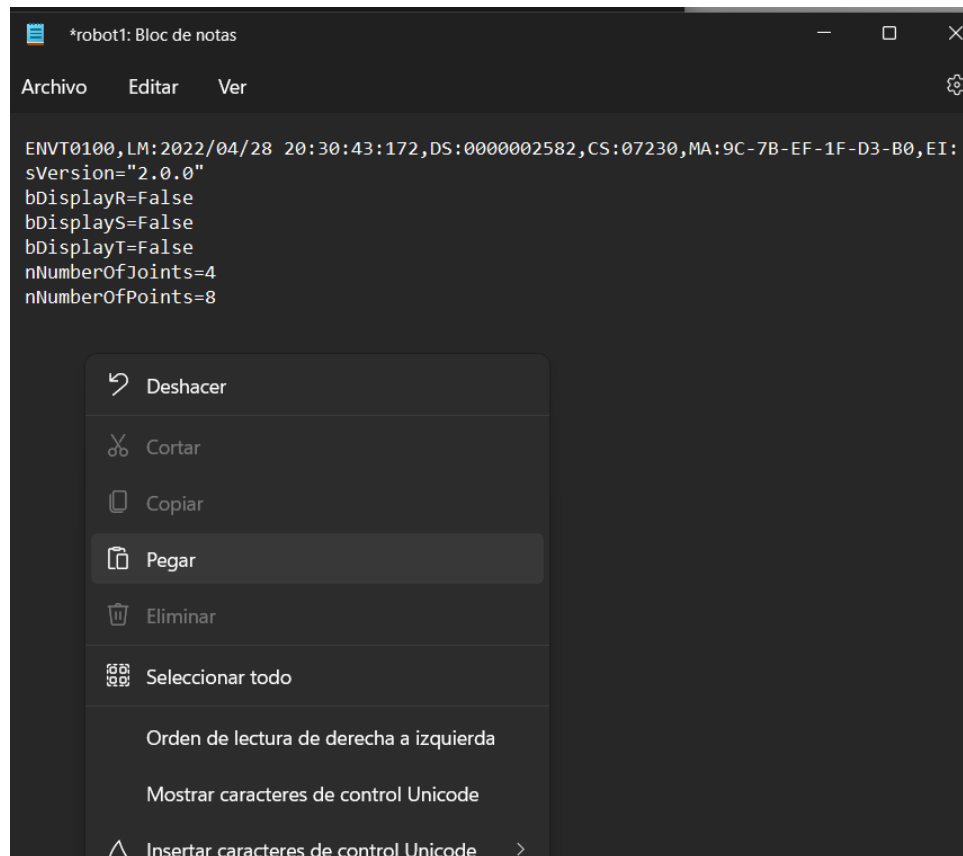
|ENV0100,LM:2022/04/28 20:30:43:172,DS:0000002582,CS:07230,MA:9C-7B-EF-1F-D3-B0,EI:
sVersion="2.0.0"
bDisplayR=False
bDisplayS=False
bDisplayT=False
nNumberOfJoints=4
nNumberOfPoints=8
Point1 {
  nNumber=0
  sLabel="primero"
  sDescription=""
  nUndefined=0
  rX=296.404
  rY=-69.365
  rZ=-1
  rU=-5
  rV=0
  rW=0
  rR=0
  rS=0
  rT=0
  nLocal=0
  nHand=2
  nElbow=1
  nWrist=1
  nJ1Flag=0
  nJ2Flag=0
  nJ4Flag=0
  nJ6Flag=0
  rJ1Angle=0
  rJ4Angle=0
  bSimVisible=False
```

6. Se necesita el encabezado del archivo legítimo que contiene el punto grabado así que se realiza un intercambio de información. Del archivo generado debemos copiar todos los puntos y sustituirlos en el archivo .pts original.



Se deben copiar todos los puntos existentes sin agregar espacios.

7. Insertamos en el archivo original sin modificar el encabezado.



En la línea "nNumberOfPoints=" se debe especificar el número de puntos totales que se tienen y una vez realizado esto damos clic en guardar.

8. Volvemos a abrir el proyecto > administrador de robot > puntos

EPSON RC+ 7.5.1A - Proyecto C:\EpsonRC70\Projects\GLUE_RAIL_RNLT - [Administrador de robot]

Archivo Editar Ver Proyecto Ejecutar Herramientas Configuración Ventana Ayuda

Conexión: N2 Sample

Robot: 1, robot1, N2-A450SR Local: 0 Tool: 0 ECP: 0

Panel de control Archivo de puntos: robot1.pts

Mover y enseñar

Número	Etiqueta	X	Y	Z	U	V	W	Local	Hand
0	cup0	-361.282	-103.024	-69.000	0.000	0.000	0.000	0	Righty
1	cup1	-369.606	-111.647	-69.000	0.000	0.000	0.000	0	Righty
2	cup2	-361.282	-120.270	-69.000	0.000	0.000	0.000	0	Righty
3	cup3	-255.229	-226.323	-69.000	0.000	0.000	0.000	0	Righty
4	cup4	-246.606	-234.647	-69.000	0.000	0.000	0.000	0	Righty
5	cup5	-237.983	-226.323	-69.000	0.000	0.000	0.000	0	Righty
6	cup6	-131.930	-120.270	-69.000	0.000	0.000	0.000	0	Righty
7	cup7	-123.606	-111.647	-69.000	0.000	0.000	0.000	0	Righty
8	cup8	-131.930	-103.024	-69.000	0.000	0.000	0.000	0	Righty
9	cup9	-237.983	3.029	-69.000	0.000	0.000	0.000	0	Righty
10	cup10	-246.606	11.353	-69.000	0.000	0.000	0.000	0	Righty
11	cup11	-255.229	3.029	-69.000	0.000	0.000	0.000	0	Righty
12	cup12	-229.502	-14.643	-69.000	0.000	0.000	0.000	0	Righty
13	cup13	-295.856	-26.343	-69.000	0.000	0.000	0.000	0	Righty
14	cup14	-339.166	-77.958	-69.000	0.000	0.000	0.000	0	Righty
15	cup15	-339.166	-145.336	-69.000	0.000	0.000	0.000	0	Righty
16	cup16	-295.856	-196.951	-69.000	0.000	0.000	0.000	0	Righty
17	cup17	-229.502	-208.651	-69.000	0.000	0.000	0.000	0	Righty
18	cup18	-171.151	-174.962	-69.000	0.000	0.000	0.000	0	Righty
19	cup19	-148.106	-111.647	-69.000	0.000	0.000	0.000	0	Righty
20	cup20	-148.106	-111.647	-69.000	0.000	0.000	0.000	0	Righty
21	cup21	-171.151	-48.332	-69.000	0.000	0.000	0.000	0	Righty
22	cup22	-326.313	-140.658	-69.000	0.000	0.000	0.000	0	Righty
23	cup23	-289.017	-185.105	-69.000	0.000	0.000	0.000	0	Righty
24	cup24	-231.877	-195.181	-69.000	0.000	0.000	0.000	0	Righty
25	cup25	-181.628	-166.170	-69.000	0.000	0.000	0.000	0	Righty
26	cup26	-161.794	-111.647	-69.000	0.000	0.000	0.000	0	Righty

Eliminar P0 Eliminar todo Guardar Restaurar

Para guardar, se puede hacer clic en cualquier parámetro a manera de darle una modificación insignificante. Un ejemplo seria seleccionar cualquier valor que este en 0 y volver a ponerlo en 0 para que no se afecte nada y se activa la opción de guardado.

9. Una vez guardado el sistema ya reconoce los puntos y pueden ser llamados en la ventana “main” para realizar el código de ejecución.

Complementos para aprovechar al máximo las capacidades del robot y software

Para que la experiencia de uso sea la mas eficiente y a manera de que el operador fomente el autoaprendizaje se enlistan las opciones mas comunes en tema de resolución de problemas y de exploración para el dominio de herramientas:

- Programación (leer manual del lenguaje referencia SPELL+ para conocer la amplia lista de comandos para realizar acciones).
- Controlador (opciones para la programación y opciones generales del controlador del robot se encuentran aquí).
- Modos de ejecución (para visualizar los movimientos antes de realizarlos en tiempo real con el robot).