



**INSTITUTO POLITECNICO NACIONAL**  
**Unidad Profesional Interdisciplinaria**  
**en Ingeniería y Tecnologías Avanzadas**



## **Practica subrutina**

Asignatura: Integración de un sistema robótico

Grupo: 4MM14

Profesor: Rivas Bonilla Elizabeth

No. Equipo: 3

### **Integrantes:**

Correa Escobar José Luis

Fernando Baena Soriano

García Jiménez Luis Josué

Mendoza Meza Manuel Everardo

## Contenido

<b>INSTRUCCIONES .....</b>	<b>3</b>
<b>DESARROLLO .....</b>	<b>3</b>
<b>CONCLUSIONES .....</b>	<b>8</b>
<b>Correa Escobar José Luis .....</b>	<b>8</b>
<b>Fernando Baena Soriano.....</b>	<b>9</b>
<b>Mendoza Meza Manuel Everardo.....</b>	<b>9</b>

## INSTRUCCIONES

Crear la imagen que se adjunta por medio de dos programas: un principal y una subrutina. Revisar PDF adjunto.



*Figura 1: Trazos a realizar en la practica*

## DESARROLLO

Se creó un New Job, dentro de la modalidad Job del teach pendant, el cual será el programa padre para esta práctica. El nombre asignado a este programa es ESE\_3\_P.



*Figura 2: Nombre del programa padre*

El paso siguiente es colocar una hoja de papel en el área de trabajo del robot Yaskawa, a la par de un plumón en la punta de la herramienta con el fin de realizar los trazos.

Cabe resaltar el uso de lo visto a lo largo del curso, ya que, para poder operar el robot Yaskawa es necesario observar que no tenga activado los botones de paro de emergencia y que no se encuentre ninguna persona dentro del espacio de trabajo, salvo el operador. Una vez considerado eso, procedemos con la realización del código.

Dentro del programa padre, se optaron por poner los puntos correspondientes al trazo del círculo interno del engrane, como se muestra en la imagen

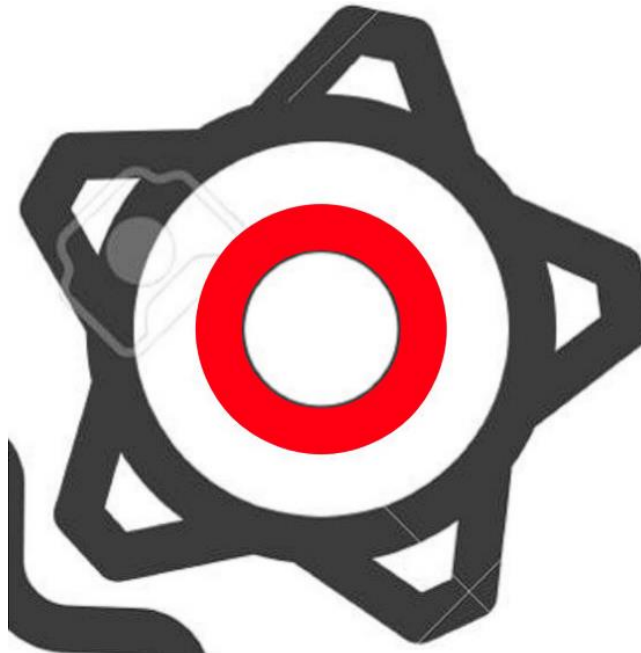


Figura 3: Trazo a realizar por el programa padre (marcado de color rojo)

Para realizar este círculo, se hizo uso del siguiente código:

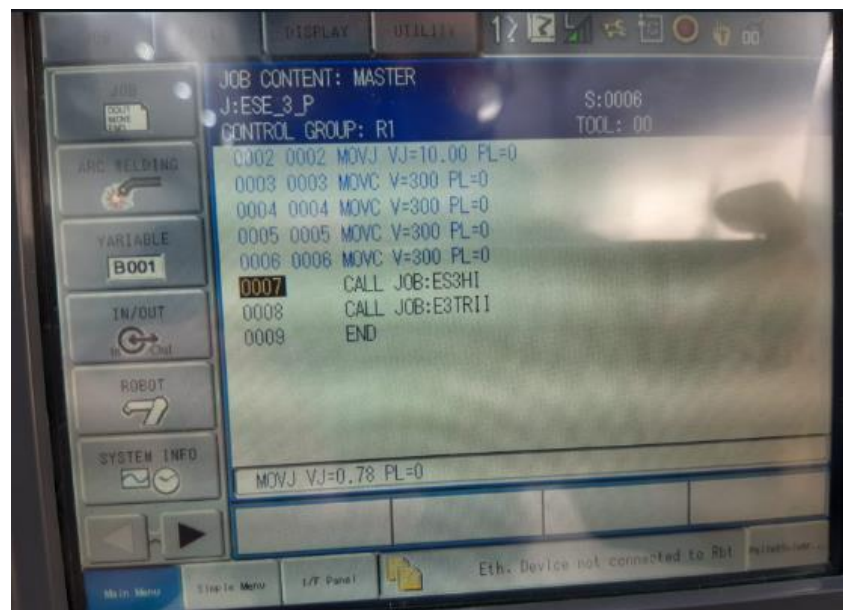


Figura 4: Código del programa padre

Como se puede observar en la imagen, los primeros dos puntos corresponden al desplazamiento del robot desde un punto alto hasta la superficie del papel, en donde se coincide con el contorno del círculo. Posteriormente, se traza la circunferencia colocando 4 puntos como lo marca la documentación del robot Yaskawa®.

Después, se colocaron las líneas correspondientes a los programas hijos mediante CALL.

En primer programa hijo realiza la función de trazar el círculo exterior, como se muestra en la imagen, marcado de color rojo

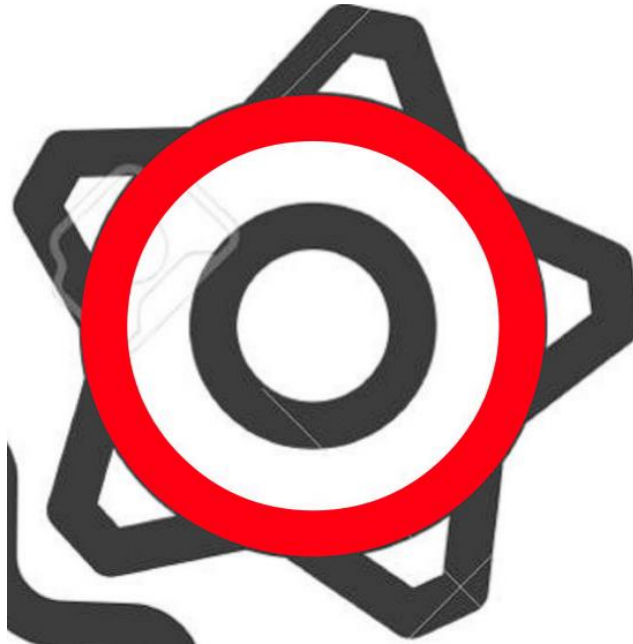


Figura 5: Trazo a realizar por el primer programa hijo (marcado de color rojo)

El código usado para realizar este trazo se muestra a continuación

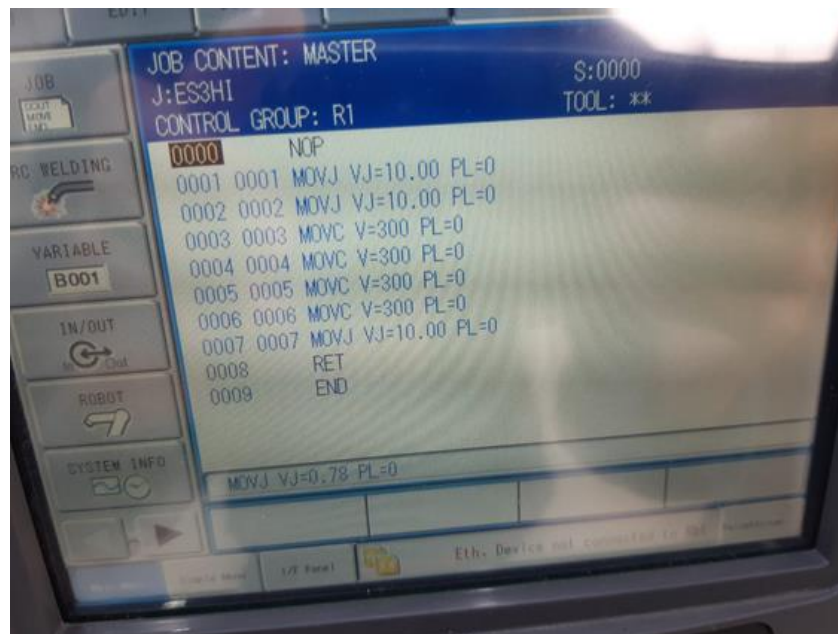


Figura 6: Código del primer programa hijo

De la imagen se puede observar cómo se realizan los mismos movimientos de las primeras 6 líneas del programa padre, siendo la única diferencia las coordenadas de los puntos

correspondientes a la circunferencia. También se puede observar que después de trazar el círculo, el robot tiende a regresar a su posición inicial por arriba de la hoja y sin tocarla.

Por último, se tiene la parte correspondiente a los dientes del engrane, como se muestra en la imagen

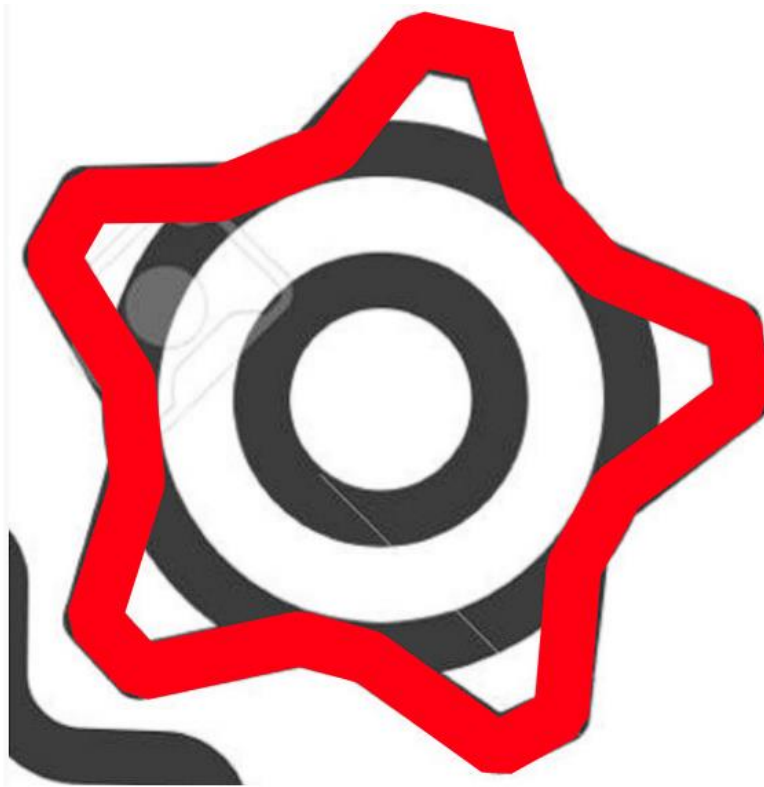
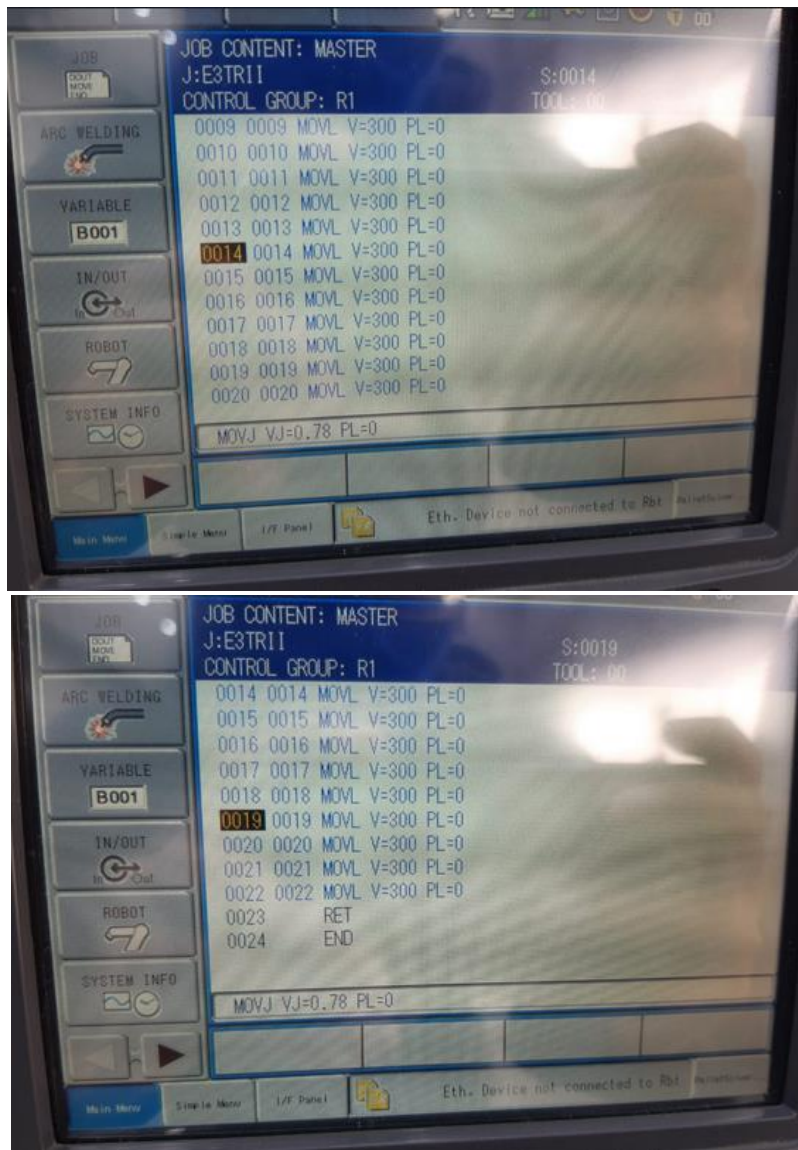


Figura 7: Trazo a realizar por el segundo programa hijo (marcado de color rojo)

El código usado se muestra a continuación







*Figura 8: Líneas de código del segundo programa hijo*

Como se puede observar en las imágenes, el programa se compone de movimientos lineales en su mayoría. Cada uno de estos puntos son colocados en la esquina de cada diente, lo que da como producto final el contorno correspondiente al engrane.

Una vez realizado todo el programa, se procede con la simulación paso a paso, es decir línea por línea, y de esta forma observar si el código funciona de la manera esperada.

Una vez observado la simulación, se procede con la ejecución del código completo. El resultado obtenido se muestra en la figura



*Figura 9: Resultado final de la practica*

El video de la ejecución se puede ver en el siguiente enlace:

[https://youtu.be/ZB\\_MYhbx72c](https://youtu.be/ZB_MYhbx72c)

## CONCLUSIONES

**Correa Escobar José Luis**

Con base en la práctica realizada se logró comprender la importancia de usar subrutinas para la realización de un programa con el objetivo de evitar instrucciones repetitivas, además de evitar tener largas líneas de código que puedan generar un problema al momento de reprogramar la rutina completa. De igual manera es de utilidad para crear tareas en específico, que puedan ser realizadas en otros códigos lo que significa una gran ventaja pues reciclar rutinas ya hechas ayuda a ahorrar tiempos y hacer mas eficiente una rutina completa con base en subrutinas para una acción en específico. Finalmente es importante destacar que a lo largo de práctica se observaron las ventajas de usar las subrutinas y así evitar la repetición de código y hacer mas eficiente la manera en de programar los robots.

**García Jiménez Luis Josué**

Existen tareas en las que el robot debe realizar trayectorias complejas o repetitivas, lo que hace que el código de programación del manipulador también se haga más extenso. Para facilitar esto es posible implementar subrutinas, es decir, códigos que tienen la programación de una tarea específica y que se va a realizar para diferentes tareas o más de una vez en una actividad. Las subrutinas requieren de un código fuente que es el que las llama a partir de una instrucción, en el ejemplo mostrado en el documento no se reutilizan



las mismas figuras para generar el engranaje, en su lugar, se emplean subrutinas para generar las diferentes partes que lo conforman: la circunferencia interna, la externa y los dientes. Por esta razón, no es posible visualizar el verdadero potencial de las subrutinas, ya que los tres códigos generados se pudieron haber realizado en uno sólo, ahorrando unas cuantas líneas encargadas de llamar las subrutinas.

#### **Fernando Baena Soriano**

Se realizó de manera correcta el trazado del dibujo con forma de engrane a través de subrutinas con el uso del robot Yaskawa®. Para la presente práctica se hizo uso de dos programas hijos (circunferencia exterior y dientes) y de uno padre (circunferencia interior) con ayuda de la función CALL.

Se pudo observar que esta característica es similar al llamar una función en un entorno de programación, y facilita la comprensión del programa a través de la reducción del código con el que se trabaja.

#### **Mendoza Meza Manuel Everardo**

Para concluir con esta actividad, la subrutina refleja el comportamiento de diferentes códigos (hijos) que dependen de un principal (padre) para realizar su acción, por lo que, crear figuras como la mostrada en el documento, refleja la implementación del nombre de la práctica subrutina. De esta forma, mientras más programas hijos podamos realizar, mayores procedimientos podemos realizar.

Es importante mencionar que, incluso con diferentes comandos como son MOVC, MOVL, etc. Que son escritos dentro de un código hijo, no varía el comportamiento del padre, así como su edición y su escrito pueden ser modificados incluso cuando la rutina se encuentra unida.