



UNIVERSIDAD NACIONAL DE COLOMBIA

FACULTAD DE INGENIERÍA

ROBÓTICA

LABORATORIO 1: ROBÓTICA INDUSTRIAL

Autores:

Fernando Cárdenas Acosta

Raúl Ignacio Marín Medina

Docente:

Pedro Fabián Cárdenas Herrera

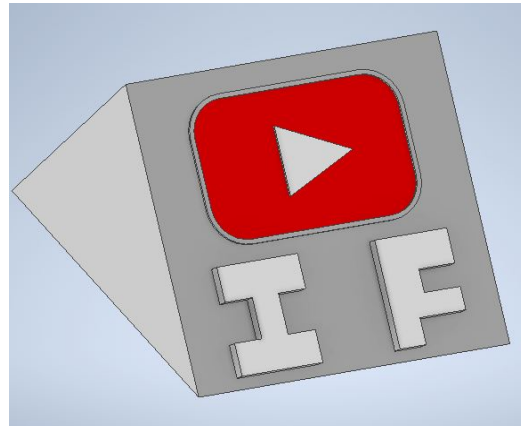
22 de Marzo del 2024

1 Planteamiento del problema

Para responder a la pregunta de si sería posible llevar a cabo el marketing y engagement de un público objetivo hacia una marca por medio del uso de sistemas automatizados como robots industriales, se plantea el uso del robot IRB 140 de la marca ABB con el fin de calibrar la herramienta necesaria para realizar el logotipo de una marca en tablero y configurar estas trayectorias mediante simulación en el software RobotStudio para finalmente aplicar el código obtenido en RAPID para su ejecución en el Robot IRB 140 y trazar un logo real mediante movimientos tipo MoveJ y MoveL. Para este caso, se selecciona el logotipo de la empresa Youtube, el cual consiste de trayectorias circulares y lineales en el espacio de trabajo, de igual manera, este logotipo se trazará sobre una superficie horizontal e inclinada junto con las iniciales del nombre para cada integrante del grupo.



(a)



(b)

Figura 1: a) Robot IRB 140 y b) Logo de Youtube basado en Skeeth modelado en Autodesk Inventor con Iniciales I (Ignacio) y F (Fernando).

1.1 Requerimientos Generales

A continuación, se mencionan las condiciones que debe cumplir el trazo del logotipo al momento de desarrollar la solución en el software RobotStudio para su posterior ejecución en el LABSIR.

- La zona de dibujo debe ser lo más amplia posible (tomando en cuenta la envolvente del trabajo y manipulador final). Para cumplir este parámetro se comparará la envolvente del efector final respecto a la posición del workobject dentro de la estación en RobotStudio y verificar si la posición del tablero es adecuada.
- Las trayectorias a desarrollar deberán realizarse en un rango de velocidades entre v_{100} y v_{1000} .
- La zona tolerable de errores máxima debe ser de ± 10 .
- El movimiento debe partir de una posición HOME especificada (puede ser el HOME del robot) y realizar la trayectoria de cada palabra y decoración con un trazo continuo. El movimiento debe finalizar en la misma posición de HOME en la que se inició. Para cumplir este parámetro se considerará la posición de Home según calibración de [1] y la función

TeachTarget para especificar el punto de Home dentro de la rutina principal en RAPID para cada entrada digital.

- La zona de dibujo debe ser establecida sobre un pliego de papel. El papel puede ser fijado sobre una superficie horizontal o sobre una plano inclinado.
- Los nombres deben estar separados.

2 Implementación en RobotStudio

2.1 Diseño de Herramienta

Para el diseño de la herramienta se tomaron las medidas respectivas del flanche del robot, esta debía tener 4 agujeros para sujetar la pieza con tornillos M6, de igual manera, debía conaiderar que el TCP (Tool Central Point) con su respectivo sistema de coordenadas estuviera en la punta de la herramienta, donde se introduce el marcador, esto con el fin de que todas las trayectorias fuesen trazadas por el TCP. Adicionalmente, el sistema de referencia en el TCP de la herramienta portamarcadores debía tener un ángulo diferente a 0 relativo al flanche del robot, ya que si existen dos sistemas de coordendas orientados en la misma dirección habrá indeterminación en la pose del robot.



Figura 2: Herramienta portamarcador con agujeros para acople en flanche de Robot (Modelo en Autodesk Inventor).

2.2 Calibración de la Herramienta

Después de realizar el modelado 3D en Autodesk Inventor, se debe exportar en archivo con extensión tipo .SAT, de igual manera, se usa la función "Set Position" en la pestaña "Home" para definir la posición relativa respecto a un sistema de referencia específico. Para el caso de ajustar la herramienta se realizan los giros respecto al sistema local de la herramienta el cual viene por defecto del modelo 3D. Para crear la herramienta, se debe entrar a la pestaña "Modelo" y dar clic en la opción "Crear herramienta", donde se tendrá que seleccionar el TCP en la geometría, después, se arrastra la herramienta hacia el robot y se actualiza la posición, finalmente, se obtiene el resultado a continuación.

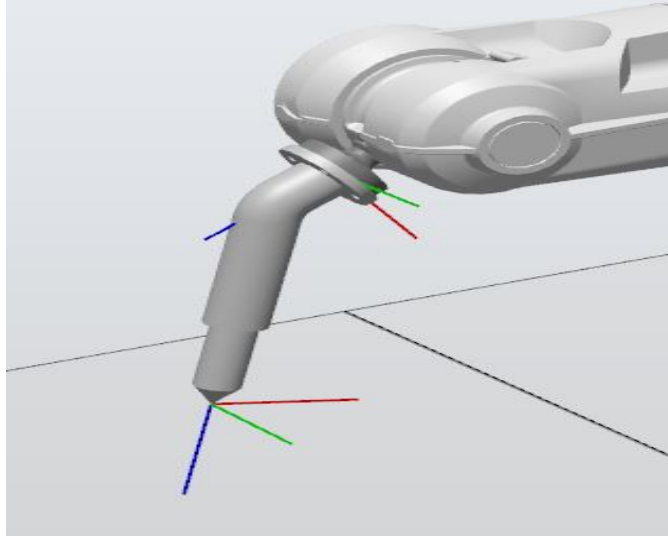


Figura 3: Herramienta portamarcador acoplada al flanche del robot en RobotStudio después de crear Tooldata.

Es importante resaltar que la dirección del sistema de coordenadas para el Tooldata deberá coincidir con el marco creado posteriormente en el workobject, etso evitará poses no deseadas en el robot y la facilidad de un trazo continuo sin tener que esforzar la estrcutura mecánica del robot (tanto alcance como puntos de singularidad).

2.3 Workobject, Targets y Path

A partir del modelo 3D del tablero se pueden trazar los puntos asociados a cada trayectoria mediante el uso del Robtarget, el cual es un tipo de dato propio de RAPID para indicar la posición de un punto respecto a un marco de referencia, la herramienta asociada al punto, el tipo de interpolación para lograr el movimiento en el espacio de trabajo (movimiento lineal, en articulaciones o circular) al igual que la velocidad (asociada al efector final del robot) y la zona de tolerancia, la cual debe ser máximo Z10, para este caso, se simuló con valores de Z0 y se observó que el robot no presentaba atascamiento en cambios bruscos de trayectorias ni error de trazo en bordes agudos, por lo que se decidió trabajar con Z0.

Para trazar estos puntos, se debe defniur en primer lugar, el objeto importado como workobject, el cual será un objeto definido en el espacio de trabajo con el cual el robot podrá interactuar, para esto, se accede en la pestaña "Home" y se selecciona la opción "Otros", donde aparecerá la opción crear un workobject. Al dar clic se requiere definir un plano asociado al workobject para definir

el sistema de coordenadas locales al objeto, este se obtuvo mediante el plano generado por tres puntos (tomar en cuenta la regla de la mano derecha para generar los planos), a partir de esto, todos los puntos creados con la opción "Target" estarán definidos respecto al workobject tal como se muestra en la figura a continuación.

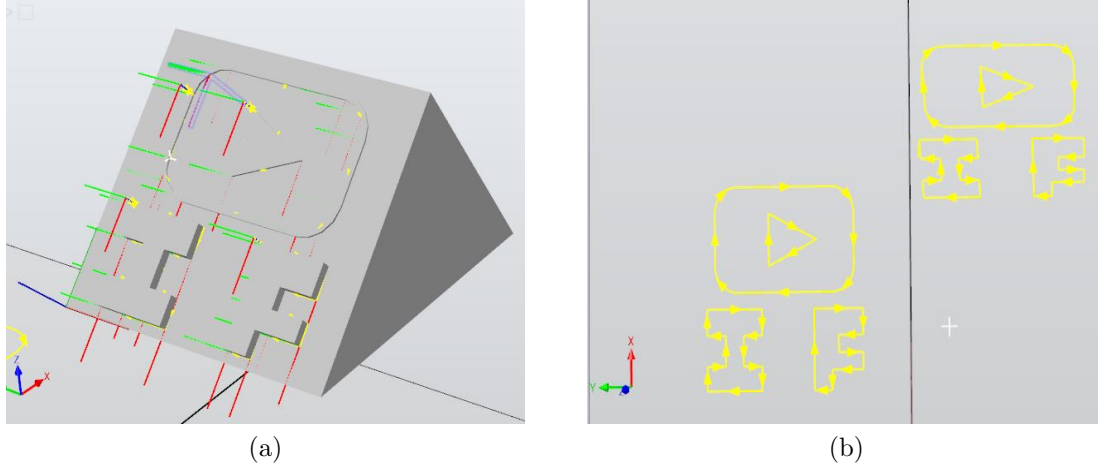


Figura 4: a) Definición del modelado 3D del tablero como Workobject en RobotStudio y b) Trayectorias definidas para el robot en plano horizontal e inclinado.

Para trazar las trayectorias, se usó la opción "Path" en la pestaña "Home", la cual permite crear una trayectoria vacía e ingresar los targets tipo Robtarget(). En función de la trayectoria, se selecciona el tipo de movimiento que el robot realizará, por ejemplo, para el contorno del logo de Youtube, se requirió de MoveC, el cual permite movimiento circular al juntar dos puntos que formen el arco en el espacio, mientras que el interior del logo y las iniciales consisten de trazos lineales en el espacio, por lo que se usó MoveL para el trazado de trayectorias, de igual manera, se usó el MoveJ para moverse en ciertos puntos donde no se quería trazar una trayectoria, sin embargo, se necesitaba mover el efector final, por ejemplo, para pasar del final de una trayectoria al inicio de otra, para ir a la posición de mantenimiento o volver a la posición de Home.

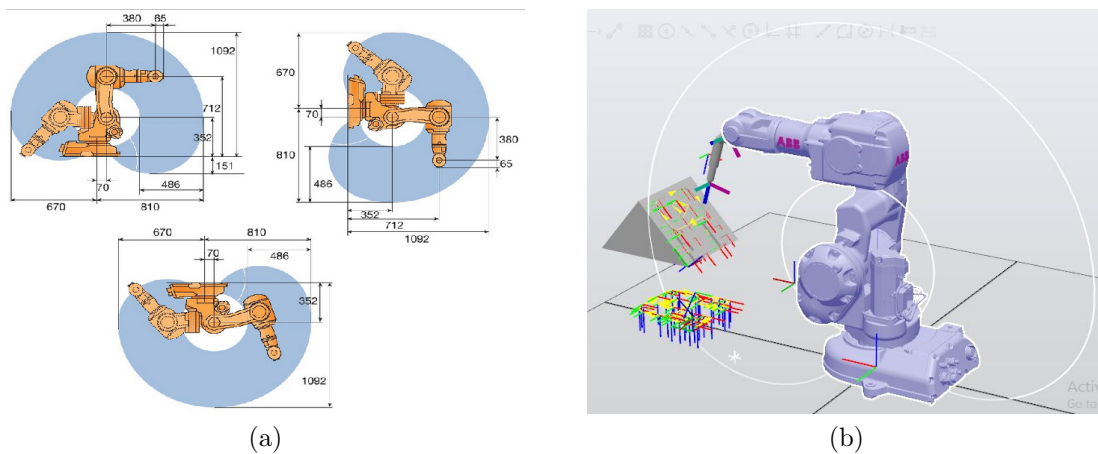


Figura 5: a) Envolvente de trabajo para IRB 140 y b) envolvente de trabajo considerando la dimensión de la herramienta en RobotStudio

2.4 Entradas y Salidas Digitales

Las entradas y salidas digitales se configuran en la pestaña "Simulación" con la opción "I/O Simulator" con el fin de interactuar de manera física con el controlador IRC5 al presionar ciertos botones que cambian el estado lógico de las señales definidas como entrada, para este caso, se define como entrada $DI - 01$ y $DI - 02$, estas activarán dos acciones diferentes en el `main()`, la primera será realizar la trayectoria para el trazo del logotipo mientras que la segunda llevará el robot a una posición de mantenimiento para realizar el cambio de la herramienta. Las señales de salida $DO - 01$ y $DO - 02$ cambiarán con la función `SETDO` la cual cambia el estado lógico de la señal de salida para ciertas condiciones de entrada digital. A continuación, se presenta el código de entradas y salidas digitales.

```
PROC main()
  SetDO DO_01,0;
  WHILE TRUE DO
    IF DI_01 = 1 THEN
      SetDO DO_01,1;
      !Path_50;
      !Path_60;
      !Path_70;
      !Path_80;
      Path_10;
      Path_100;
      Path_110;
      Path_120;
      MoveJ Target_440,v400,z0,MyNewTool\WObj:=Workobject_4_2;
    ENDIF
    IF DI_02 = 1 THEN
      SetDO DO_01,0;
      Path_130;
    ENDIF
  ENDWHILE
ENDPROC
```

Figura 6: Código RAPID para entradas y salidas digitales.

2.5 Simulación y Código RAPID

En función del código RAPID, se establece una función `main()`, la cual será la acción del robot durante la ejecución del ciclo al activar las entradas digitales, a continuación, se presentan los pasos que se deben tomar en cuenta para implementar el código RAPID con el controlador IRC5, además, el paso a paso de acción del robot planteada como diagrama de flujo para las entradas digitales.

De igual manera, se mencionan las funciones aplicadas en RAPID y su descripción según [2]:

- `wobjdata`: Este tipo de dato se usa para describir el objeto que el robot soldará, procesará, moverá o tendrá cualquier tipo de interacción. En casos donde la herramienta sea estática, es importante definir el `wobj` ya que la posición, velocidad y trayectorias dependen directamente del movimiento de este. De igual manera, el robot podrá hacer jogging en las direcciones establecidas por el `wobj`.

Tendrá el componente "robot hold" el cual será un valor booleano que indica si el efector final agarra el `wobj`, el componente "ufprog" indica si el sistema coordinado del usuario que

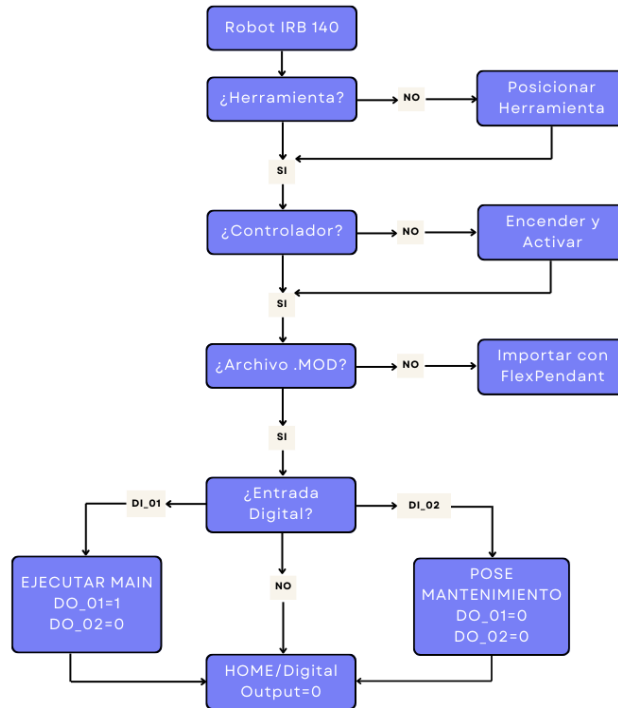


Figura 7: Diagrama de Flujo de Acciones del Robot en Función de las Entradas y Salidas Digitales.

se define será fijo o será variable dependiendo de la aplicación, lo cual afectará directamente la pose del robot. Por último, tendrá el componente "oframe", el cual será el marco de referencia asociado al wobj directamente, este se puede expresar como una pose en función de un target o como una orientación en cuaterniones.

- Tooldata: Se usa este tipo de dato para describir las características de una herramienta, estos parámetros serán la posición y orientación del TCP (Tool Center Point) y las características geométricas e inerciales de la herramienta, este TCP se define en función del sistema de coordenadas del flanche mientras que para herramienta estacionaria, se define el TCP en función de las coordenadas globales, tendrá componentes "robhold" el cual es un valor booleano para determinar si el robot está tomando la herramienta y tendrá "tframe" el cual especifica el marco de referencia respecto al cual se encuentra definido el sistema de coordenadas de la herramienta.
- SETDO: Se usa para cambiar el valor de una salida digital en función de una señal de entrada, esta función tendrá tres componentes importantes, los cuales son "sdelay", el cual indica el tiempo de retraso que habrá en el cambio (mantiene ejecución del main() hasta que se cumple el delay específico y activa cierta salida digital), también se encuentra "Sync" el cual espera a que la señal de entrada se active físicamente para dar la salida lógica, por último, esta "signaldo" la cual será la señal a la que se le cambiará el estado lógico.

3 Robot IRB 140

3.1 Implementación de Código RAPID

Después de terminar la simulación de trayectorias y ajustar parámetros de error de posición, velocidad y offset de puntos entre trayectorias mediante la función MoveJ, se sincroniza el workspace con la estación y el controlador. En la pestaña RAPID aparecerá la sección módulo, la cual contiene el código con la información de Robtarget, Trajectories, Tooldata y Wobjdata. Se guarda esta información como archivo con extensión .MOD, para realizar la ejecución, se agrega este archivo a la memoria del controlador mediante entrada USB al FlexPendant entrando en las ventanas "Program Editor", "Edit", "Load", "Select and Ok", para finalizar con la opción "PP to Main" la cual reconoce el main() del archivo .MOD. Por último, se da play en el FlexPendant, en este punto, el controlador está a la espera de una entrada digital para ejecutar trayectoria (D0-01) o llevar el robot a pose de mantenimiento (D0-02).

3.2 Calibración del Workobject Real

A pesar de que el controlador puede ejecutar el código RAPID, es necesario considerar que el espacio de trabajo real será diferente con el espacio de trabajo virtual, puesto que los bordes físicos pueden interrumpir la correcta ejecución de las trayectorias o en casos no favorable, se podría romper la herramienta o causar un daño en el conjunto mecánico del robot, por lo tanto, se propone calibrar la herramienta en función de un nuevo sistema de referencia para el Workobject real, para este caso, el modelo en el Workspace de RobotStudio coincide en dimensiones y geometría con el tablero real, sin embargo, para indicarle al robot donde debe escribir se realiza la calibración del WOBJ real como:

- Ingresar a Program Data en el FlexPendant, escoger el Workobject deseado según la información de Wobjdata importada.
- Modificar posición del workobject con la función "Modify" y crear nuevo sistema de coordenadas con "Three Points", acá se debe seguir el sentido de la mano derecha, ya que el nuevo sistema de coordenadas para el workobject debe coincidir con el sistema de coordenadas en el TCP de la herramienta para evitar errores de trayectoria y puntos de singularidad.
- Con la función "Jogging" posicionar los puntos tocando el workobject real, estos puntos deben ser un origen, un punto en X y otro punto en la dirección Y para generar el nuevo plano.
- Guardar la información "Save" y actualizar los puntos respecto al nuevo workobject. Esto permite que al activar la entrada digital, el robot escriba sobre el nuevo plano el cual coincide con la superficie física del workobject.

3.3 Resultados

Tras la ejecución del código RAPID en el controlador ICR5, en primer lugar, se realizó la trayectoria del logo en el aire con el fin de detectar fallas asociadas a puntos de singularidad y alcance real del efector final en el espacio de trabajo. Después, se realizó la calibración del Workspace para que el robot escribiera en el plano donde se ubicó el pliego, sin embargo, para el primer intento no

se había considerado offset entre los inicios de trayectoria lo cual rayaba el pliego al cambiar de trayectorias, esto se corrigió al subir los puntos de inicio en RobotStudio y sincronizar nuevamente con la estación.

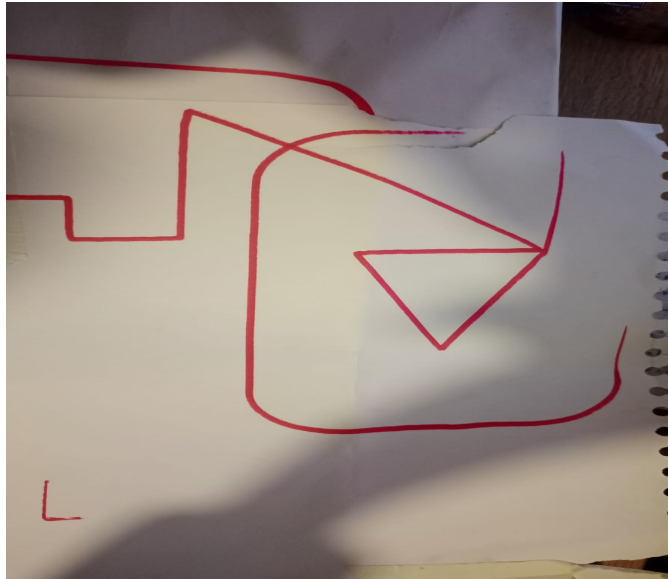


Figura 8: Intento 1 (Área de trabajo incompleta y trazos erróneos en MoveJ).

Para el segundo intento, se observa que el logotipo abarca todo el área de trabajo y el logotipo se encuentra completo.

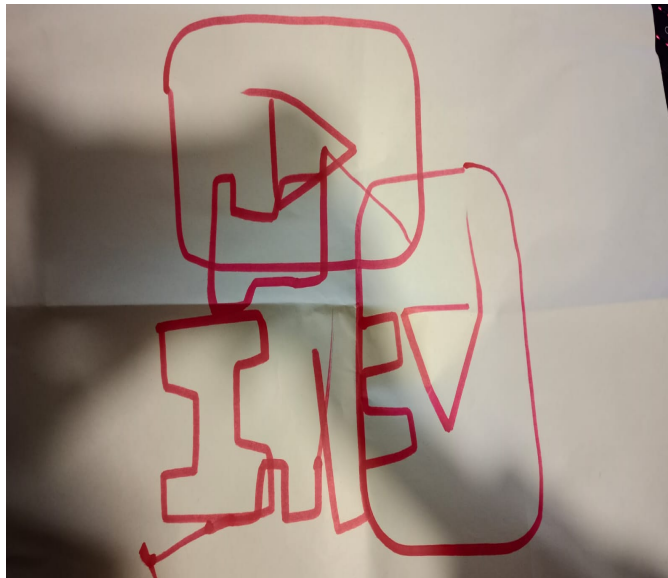


Figura 9: Intento 2 (Área de trabajo completa, prueba de calibración del Workobject real respecto al TCP de herramienta).

Se realizó la prueba final haciendo uso de las entradas digitales, para observar la prueba, se encuentra el video en la sección "Links Importantes".

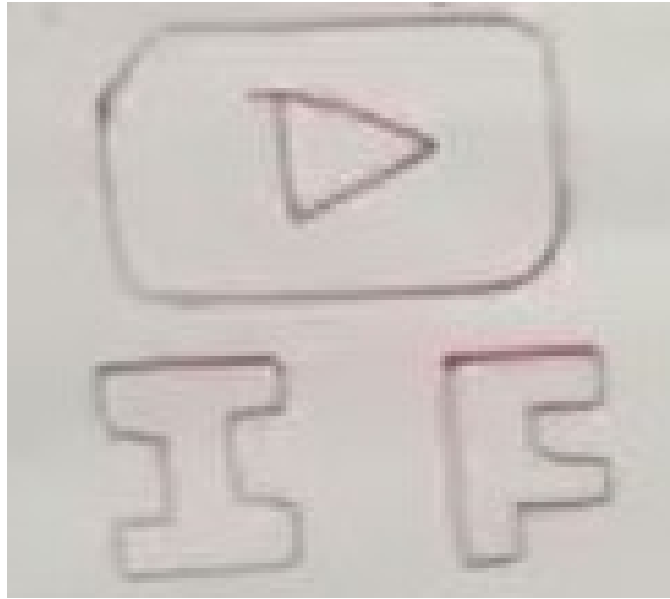


Figura 10: Resultado Final.

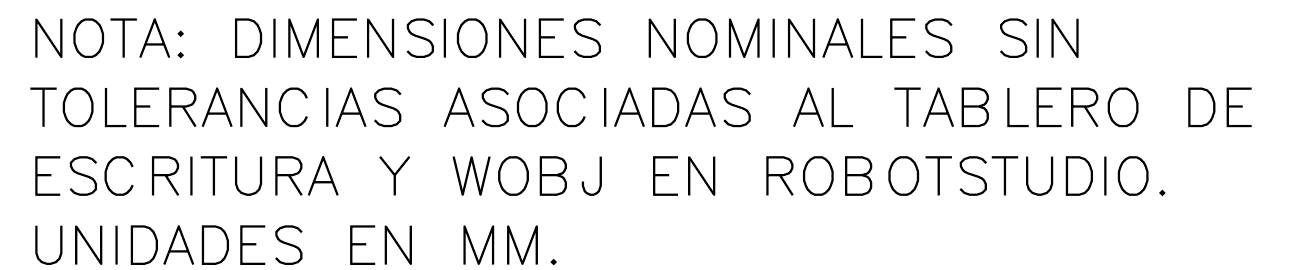
4 Links Importantes

A continuación, se encuentran los links para visualizar la simulación de trayectorias en RobotStudio para plano horizontal y plano inclinado, de igual manera, se muestra el video realizado en el LABSIR con trayectoria real del robot IRB 140 con uso de entradas y salidas digitales. Por último, se adjunta el repositorio GitHub con el código RAPID en archivo .mod.

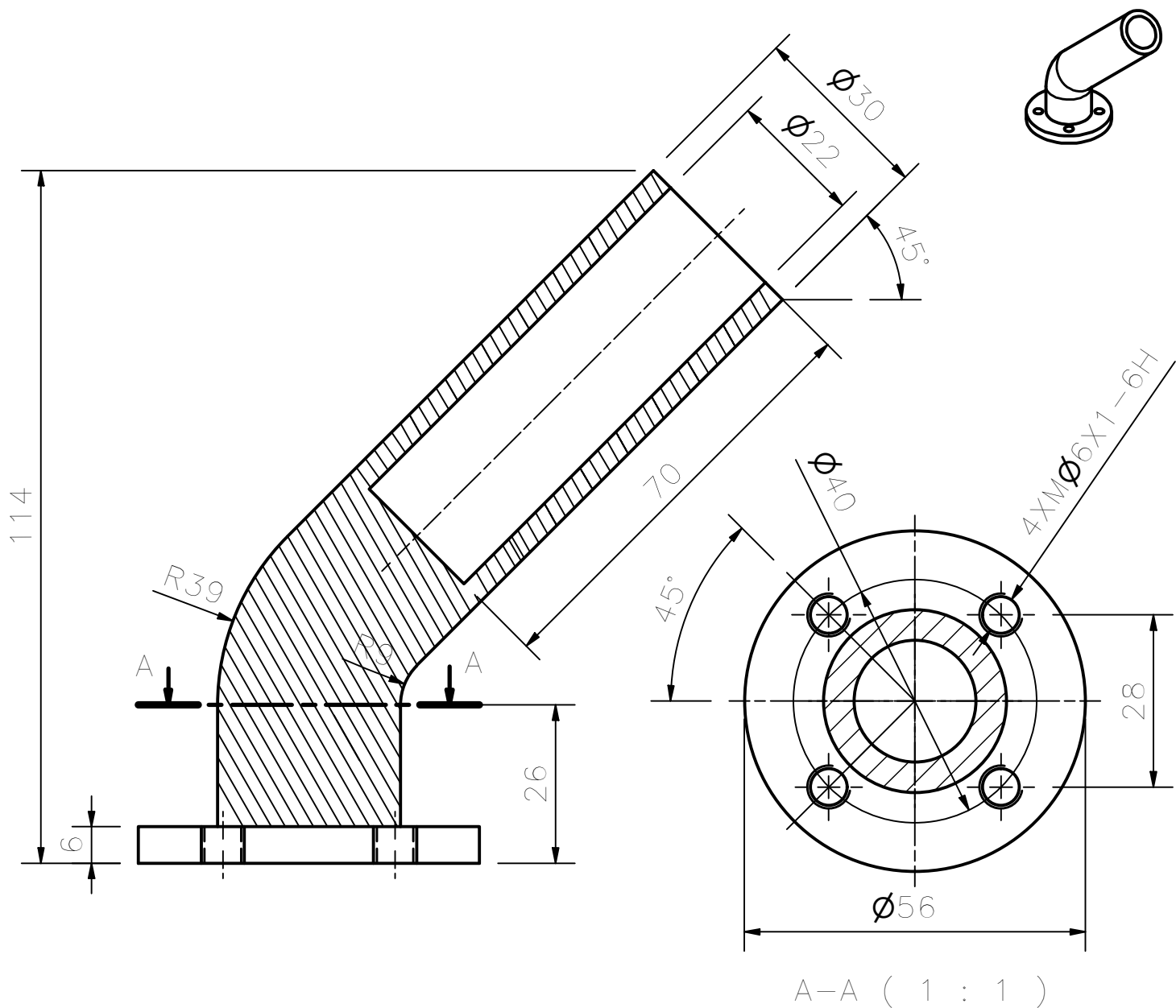
- Simulación en Plano Horizontal: <https://youtu.be/VAoQ1diFxxI>
- Simulación en Plano Inclinado: <https://youtu.be/0ls0QD2tfM0>
- Uso del Robot IRB 140 en LABSIR: https://youtu.be/sZnGTWtp_ts
- Repositorio GitHub: <https://github.com/ramarinm/Robotica-Laboratorio>

5 Referencias


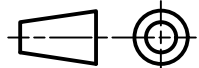
- [1] “Manual del producto IRB 140.” Available: <https://library.e.abb.com/public/a56b0629dd9f404db7e384375a3c4003/3HAC027400%20PM%20IRB%20140-es.pdf?x-sign=mwijcy/OLqd77lbsq/WyG/d9Y0h4e0GsLbYeuqE28+kn8T57IaV9ZzMnusFzMW2E>
- [2] “Technical reference manual RAPID Instructions, Functions and Data types.” Available: https://library.e.abb.com/public/b227fcd260204c4dbeb8a58f8002fe64/Rapid_instructions.pdf

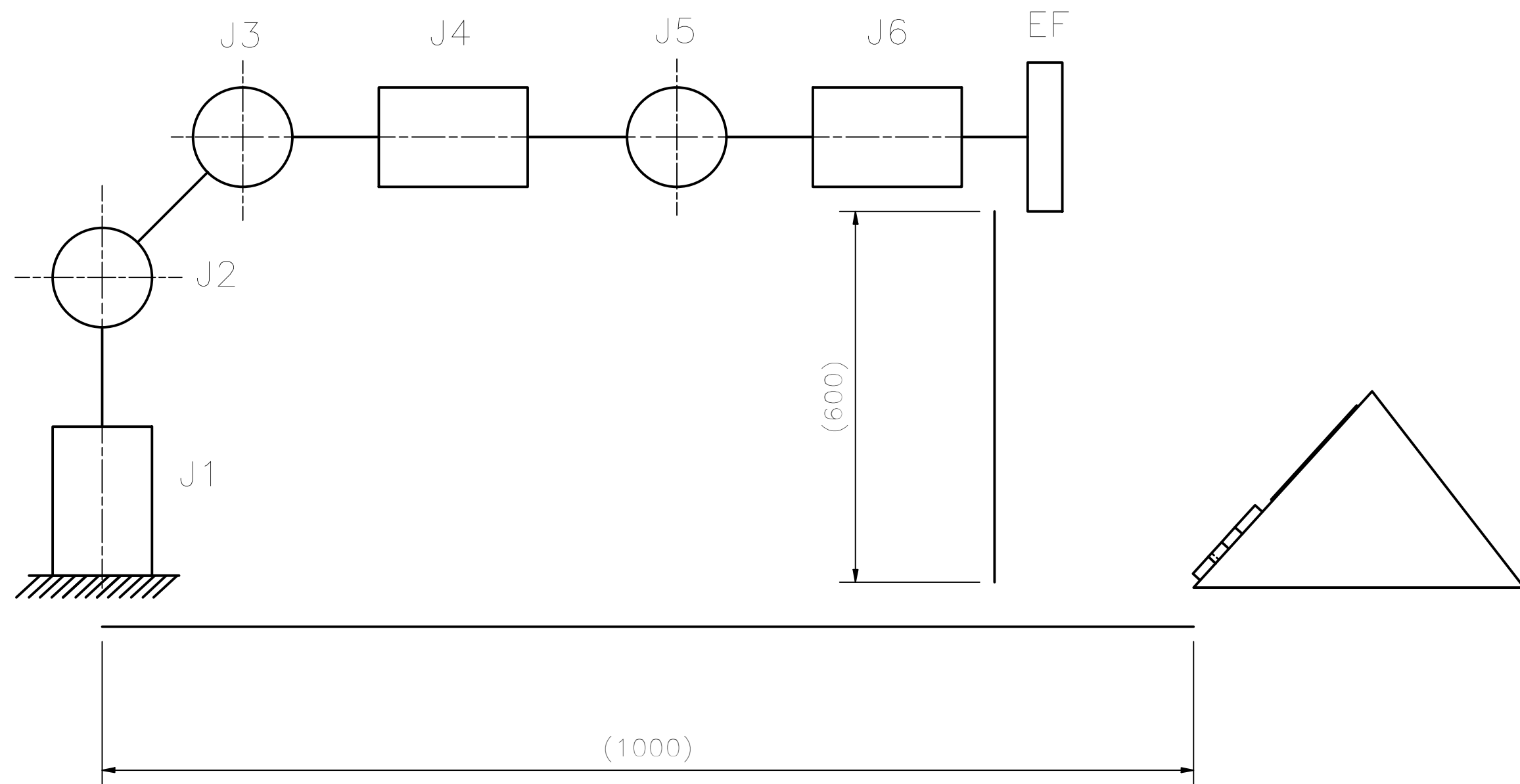


 <p>UNIVERSIDAD NACIONAL DE COLOMBIA FACULTAD DE INGENIERÍA DEPTO ING, MECÁNICA Y MECATRÓNICA</p>		
CONJUNTO:		
No Aplica		
SUBCONJUNTO:		
No Aplica		
SUB-SUBCONJUNTO:	TEM:	
No Aplica	No Aplica	
PIEZA		
Workobject Logo Marca	R/ 1	1 / 3


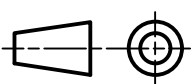


NOTA: SE ASUME TOLERANCIAS LINEALES DE 6% MENOS RESPECTO A MEDIDA NOMINAL DEBIDO A PROCESO DE FABRICACIÓN (IMPRESIÓN 3D EN ABS). UNIDADES EN MM.

DIBUJO: Raúl Ignacio Marín Medina	FECHA: 22/03/24	FIRMA: Raúl Ignacio Marín	 <p>UNIVERSIDAD NACIONAL DE COLOMBIA FACULTAD DE INGENIERÍA DEPTO ING, MECÁNICA Y MECATRÓNICA</p>		
REVISÓ: Fernando Cárdenas	FECHA: 22/03/24	FIRMA: Fernando Cárdenas			
APROBO: Fernando Cárdenas	FECHA: 22/03/24	FIRMA: Fernando Cárdenas			
MATERIAL: Impresión 3D - ABS	CANTIDAD: 1	CONJUNTO:	No Aplica		
REFERENCIA: No Aplica			SUBCONJUNTO: No Aplica		
SISTEMA: 	FORMATO: A4	ESCALA: 1 : 1	SUB-SUBCONJUNTO: No Aplica	TEM: No Aplica	
			PIEZA: Herramienta Portamarcador	R/ 1	2 /3



NOTA: ESQUEMÁTICO DE ROBOT IRB 140
6 GDL EN ESPACIO DE TRABAJO PARA
DESCRIBIR POSICIÓN DEL WORKOBJECT.
UNIDADES EN MM. ROBOT EMPIEZA EN
HOME.

DIBUJÓ: Raúl Ignacio Marín Medina	FECHA: 22/03/24	FIRMA: Raúl Marín	UNIVERSIDAD NACIONAL DE COLOMBIA FACULTAD DE INGENIERÍA DEPTO ING, MECÁNICA Y MECATRÓNICA 		
REVISÓ: Fernando Cárdenas	FECHA: 22/03/24	FIRMA: Fernando Cárdenas			
APROBÓ: Fernando Cárdenas	FECHA: 22/03/24	FIRMA: Fernando Cárdenas			
MATERIAL: No Aplica	CANTIDAD: 1		CONJUNTO: No Aplica		
REFERENCIA: No Aplica			SUBCONJUNTO: No Aplica		
SISTEMA: 	FORMATO: A3	ESCALA: N:5A	SUB-SUBCONJUNTO: No Aplica		TEM: No Aplica
			PIEZA Planta Posición WOBJ		R/ 1 3 / 3