

SOLDADURA ROBOTIZADA 4.0

Ponente: José Calvo Fernández



Robot ABB modelo IRB1410_5kg_1.44m

TRABAJO FIN DE CURSO

EDUARDO DE JUAN MOLINERO

ÍNDICE

1. JUSTIFICACIÓN	3
2. SOFTWARE	4
3. PRÁCTICA DIDÁCTICA	7
3.1. Ejes del robot y movimientos	7
3.2. Programar con la consola de programación.....	9
3.3. Definir los distintos sistemas de ejes	10
3.4. Programar una trayectoria de la herramienta.....	13
3.5. Programar una soldadura por arco	14
4. BIBLIOGRAFÍA Y WEBGRAFÍA.....	15

1. JUSTIFICACIÓN

La realización de este trabajo tiene dos fines principales. Uno es recopilar información fundamental relacionada con los contenidos vistos a lo largo del curso “soldadura robotizada 4.0”, organizado por el CFIE de Miranda e impartido por José Calvo Fernández en las dependencias del IES Fray Pedro de Urbina de Miranda de Ebro, en noviembre del 2018 y el segundo fin, es aplicarlo a través de la realización de una práctica didáctica sobre soldadura robotizada.

El curso ha sido enfocado hacia la programación del robot que aparece en la portada de este trabajo, para la soldadura por arco, pero podría ser utilizado para otras muchas aplicaciones como la manipulación de piezas, paletizado, pintado de vehículos, etc. sustituyendo la pistola de soldadura amarrada a la brida final del robot, por una pistola aerográfica o el útil correspondiente.

Por este motivo y por disponer la robótica de todo un mundo de opciones y posibilidades, primero voy a dar unas pinceladas sobre los programas de simulación y de aplicación que cuenta la marca ABB y seguidamente iré con la práctica didáctica, la cual, por aspirar a ser completa, va a necesitar varias sesiones que va a depender del interés y ritmo de aprendizaje que imponga el grupo de alumnado en su momento.

En cualquier caso, creo que debemos pensar con una mente abierta, ya que en un futuro cercano, es posible que tengamos que adaptar lo visto en este curso, a otras aplicaciones y a otros modelos o marcas de robots, con entornos de programación diferentes pero manteniendo la misma base operativa.

2. SOFTWARE

La marca de robots ABB ofrece la posibilidad de descargar desde su página Web <https://new.abb.com/products/robotics/es/> los siguientes programas para el diseño, programación y configuración de las estaciones de trabajo:

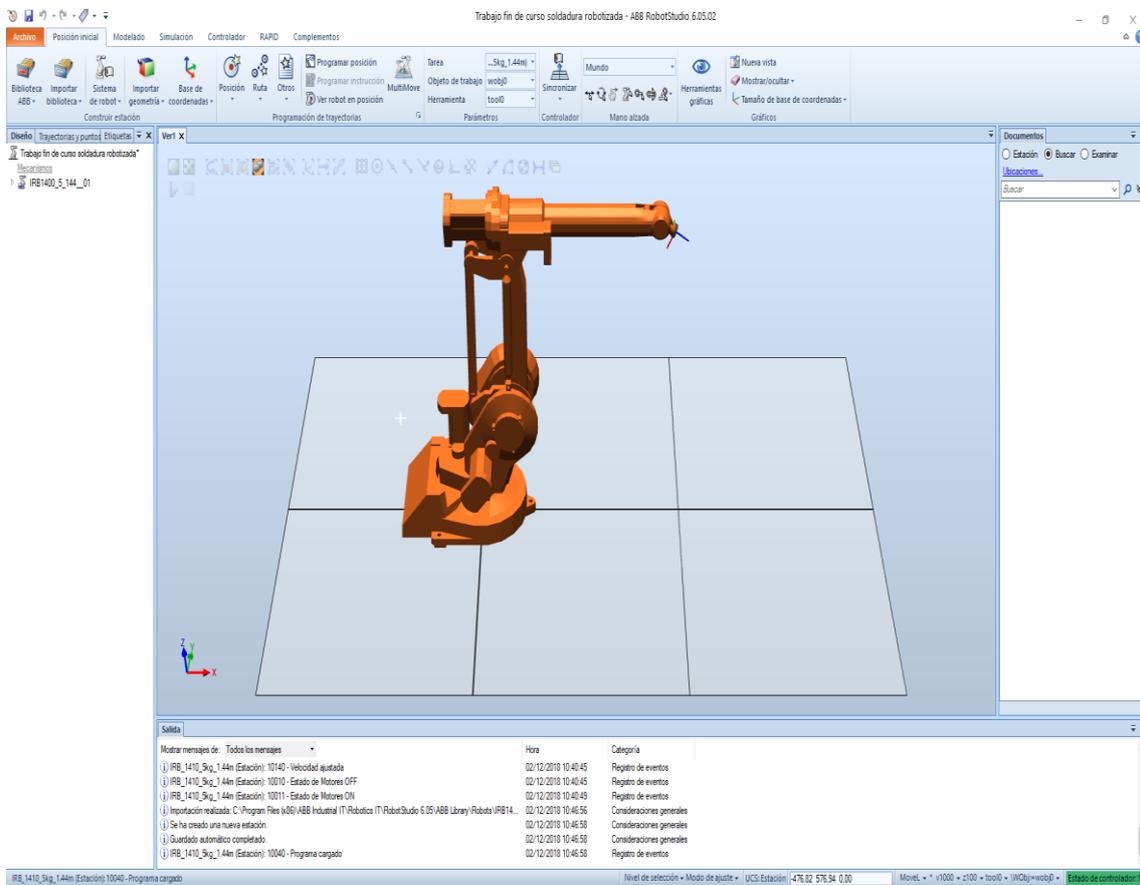
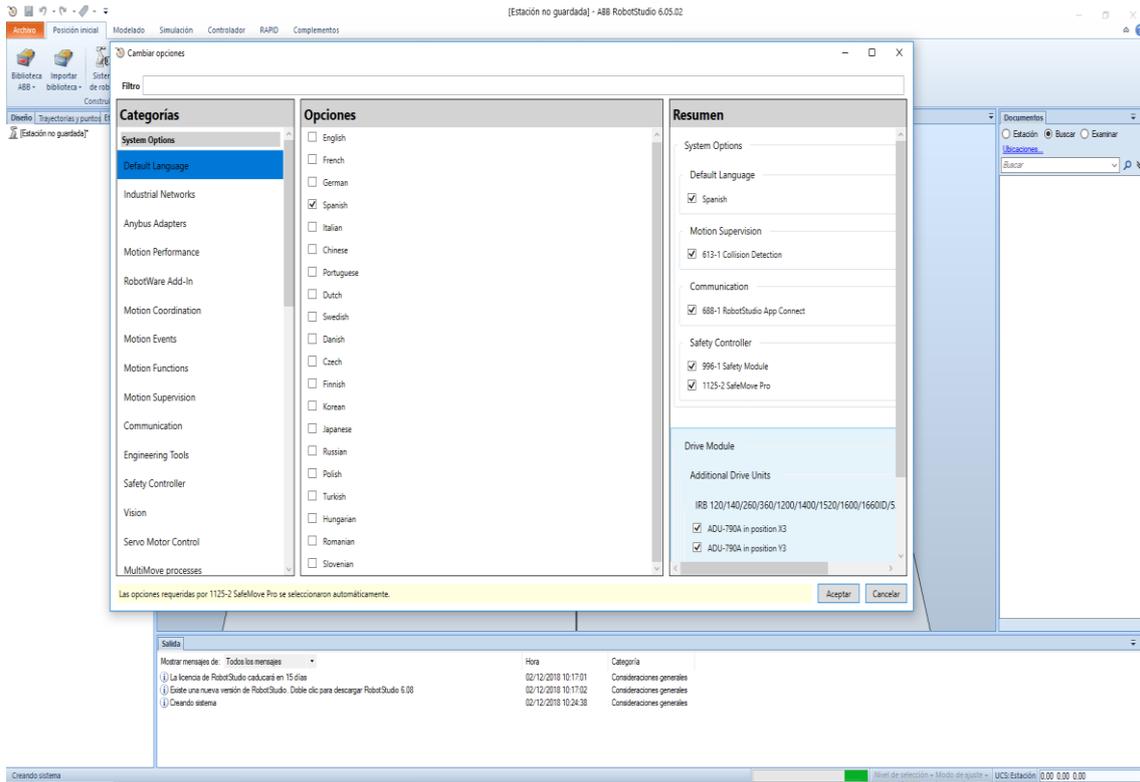
- **QuickTeach:** Este software permite simular la consola de programación del robot en un PC. Aparece en las webs como obsoleto y requiere licencia. Para instalarlo en el ordenador pide un archivo .licreqx que hay que solicitar a la marca.
- **RobotStudio:** es una copia exacta del software real que hace funcionar el robot en producción. Ello permite simulaciones muy realistas, con archivos de configuración y programas de robot reales idénticos a los utilizados en las instalaciones.

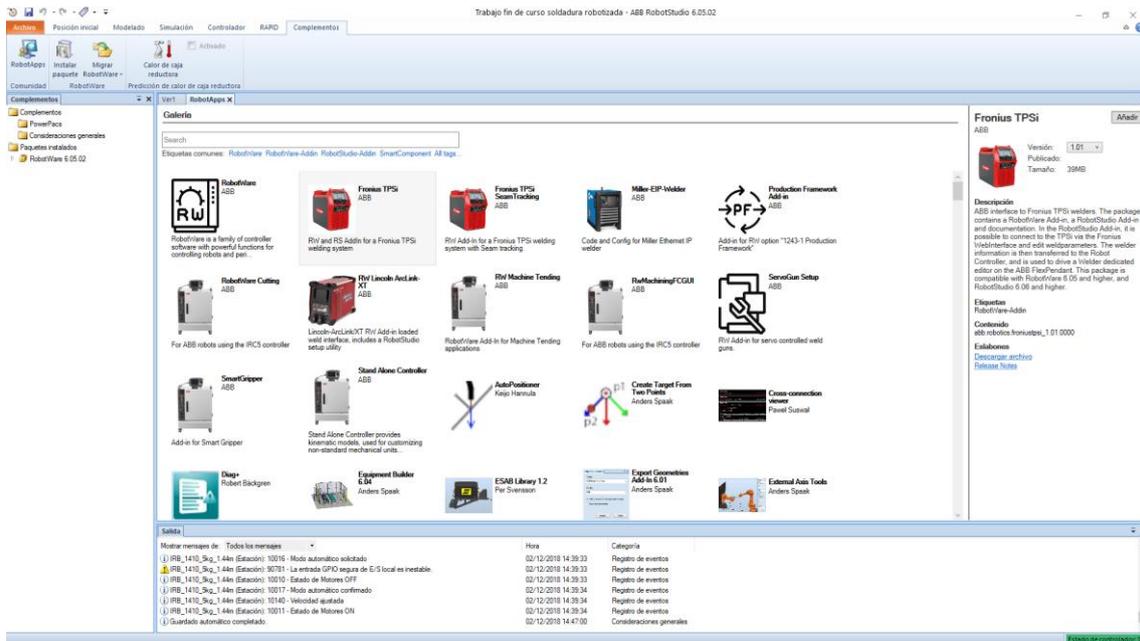
Este software podemos encontrarlo en la página <https://new.abb.com/products/robotics/es/robotstudio> y se puede descargar e instalar en el PC sin problemas. Lo malo es que la licencia de prueba expira a los 30 días y hay que renovarla. En dicho enlace encontramos también tutoriales y explicaciones de cómo utilizarlo. También se pueden encontrar videos tutoriales en Youtube.

Al abrir el programa debemos configurar el idioma (inglés por defecto), y las *System Options* en función de la aplicación, comunicación y seguridad necesaria.

Seguidamente creamos la *Estación* introduciendo el modelo de robot y a partir de aquí podemos definir la posición inicial del robot con los sistemas de ejes, modelar las piezas de trabajo, realizar simulaciones de movimientos del robot, añadir controladores y pantallas de programación virtuales como el Flex Pendant, programar con lenguaje *Rapid* y otros paquetes complementarios en función de la aplicación del robot.

Esto podemos observarlo en las siguientes imágenes:

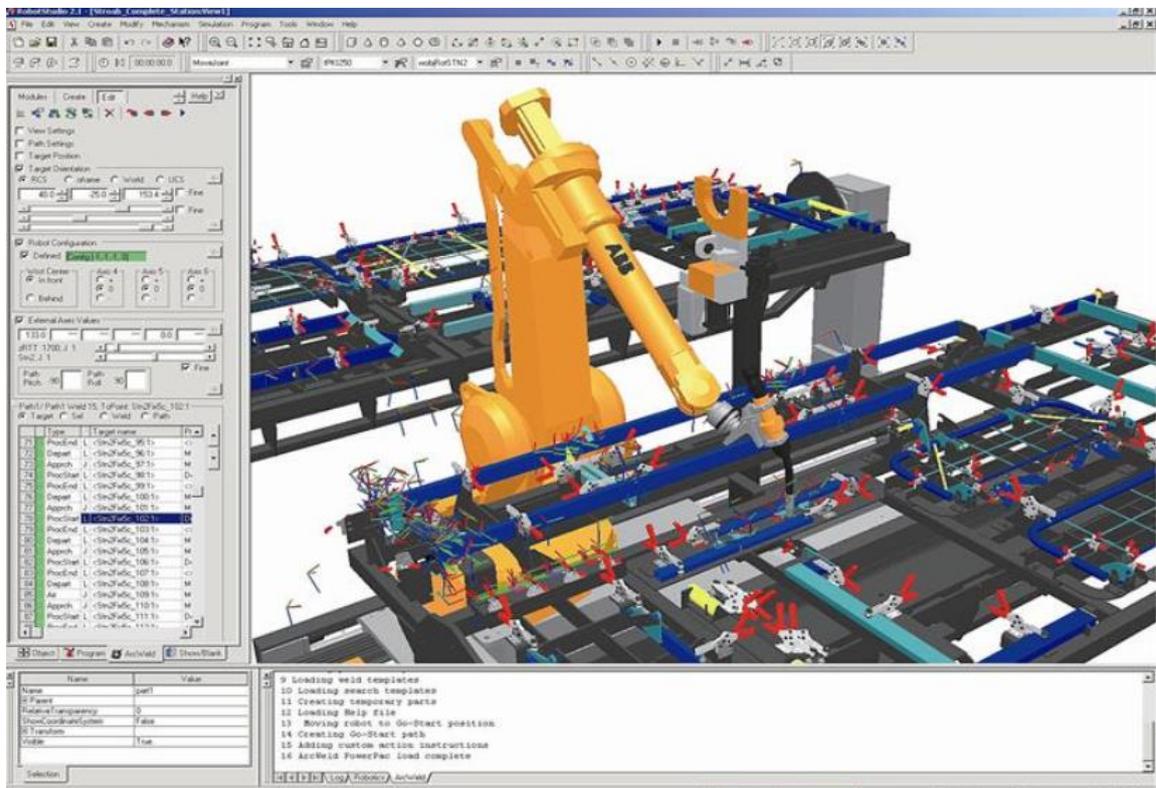




- **Arc Welding:** Es un software de aplicación para soldadura por arco muy interesante tanto para el programador del robot como para el soldador. Es de fácil uso y ayuda a mejorar el proceso de soldadura por maximizando el rendimiento de sus sistemas robóticos.

El software para ésta y otras aplicaciones podemos encontrarlo en la pág.:

<https://new.abb.com/products/robotics/es/software-para-aplicaciones>



3. PRÁCTICA DIDÁCTICA

Objetivos:

1. Conocer los ejes del robot y movimientos.
2. Programar con la consola de programación.
3. Definir los distintos sistemas de ejes.
4. Programar una trayectoria de la herramienta.
5. Programar una soldadura por arco.

3.1. Ejes del robot y movimientos.

El robot ABB IRB1410 sito en el IES Fray Pedro, cuenta con 6 ejes que pueden ser girados uno a uno desde la consola de programación deshabilitando el movimiento lineal y seleccionando el grupo 1 (ejes 1,2,3) o el grupo 2 (ejes 4,5,6) pulsando el botón correspondiente, siendo el eje 1 el de la base y el 6 el de la muñeca. El origen o punto 0 es el que está situado en el centro del eje que une la base con el tronco del robot.

Los giros pueden ser negativos o positivos y se giran mediante el joystick de la consola, que ha de ser movido con suavidad para no chocar con el robot alguna protección, pues tiene un pequeño tiempo de retardo y una posterior aceleración. Para que se mueva el robot es necesario mantener pulsado el botón de desbloqueo que habilita el movimiento del robot.

En la pantalla de la consola podemos visualizar las coordenadas del punto central de la brida donde se montará la herramienta o útil.

Si queremos realizar movimientos lineales de la herramienta, sólo tenemos que seleccionar el sistema de ejes Mundo que equivale al Base en nuestro caso por no haber más robots en el entorno de trabajo y habilitar el modo lineal con la tecla de movimiento (se deshabilita el modo reorientación).

Hay que tener claro que el eje X es positivo hacia nosotros, el eje Y es positivo hacia la derecha y el eje Z es positivo hacia arriba, viendo el robot de frente y por delante.



BASE

Para evitar movimientos indeseados del robot debemos pulsar la seta de emergencia de la consola y la del armario y soltarlas cuando trabajemos con él.

3.2. Programar con la consola de programación.

La consola consta de una pantalla de visualización, una botonera, un joystick y una seta de emergencia. Alrededor de la pantalla tiene varias teclas de selección y función.



Debemos pulsar la tecla de programación y crear el path adecuado para guardar los programas en el módulo deseado.

Para crear un módulo de programa tenemos que pulsar la tecla View – Instr.(main) – módulos - Program Module – pulsar la tecla New... situar el cursor sobre Name y con el teclado numérico de la consola escribir el nombre que queremos darle al módulo (por ej. PRACTICA). Tiene que aparecer en Type: Program Module – pulsar Enter. Después pulsar OK para volver a la pantalla anterior y observar que en Program Modules aparece el path: MAIN/PRACTICA.

Con el cursor sobre el módulo PRACTICA pulsar Enter para ver la lista de rutinas. Si aparece No Routines, pulsar la tecla New...y escribir el nombre que queremos (por ej. PRÁCTICA 1) con el teclado numérico como antes, pulsar OK 2 veces y observar que el path es MAIN/PRACTICA/PRACTICA1.

Para programar instrucciones, en la pantalla debe Program Instr y al lado el path anterior. Debajo del salto de pantalla (2 líneas discontinuas

horizontales) ha de aparecer <SMT> a la izquierda y las instrucciones de programación a la derecha en la pantalla de visualización.

3.3. Definir los distintos sistemas de ejes.

El robot puede trabajar sincronizadamente con 4 sistemas de ejes distintos que son World, Base, Tool y Wobj.

Como solo tenemos un robot, el sistema de ejes World y Base coinciden y sitúan su origen en el punto central 0 de la base del robot. Sin embargo, nos interesa mucho más que el robot se mueva en función de un sistema de coordenadas con origen en la punta de la boquilla de soldadura en nuestro caso, por lo que se hace necesario “declarar la herramienta”, es decir, definir las coordenadas de la herramienta y su orientación respecto del sistema de coordenadas Base.

Para ello, en la ventana de programación pulsar View – Data Types - Tooldata y pulsar Enter. Es importante que los datos de la herramienta se guarden en el path Program Data - MAIN/USER con el fin de tenerlos disponibles para cualquier programa del robot.

Con el cursor sobre Tool0 pulsar New...y poner el nombre a la herramienta con el teclado numérico (PISTOLA por ej.). Después pulsar la tecla Special – define Coord – Method: 4 points TCP... Pulsando la tecla de saltar la doble línea, seleccionar el primer punto de aproximación.

Para definir el TCP (Tool Center Point), sobre la mesa de soldadura colocamos un objeto acabado en punta hacia la cual, moveremos el robot primero con movimientos rápidos y después con movimientos incrementales large, medium y small hasta tocarlo con la punta de la herramienta. En ese momento pulsamos en ModPos para grabar el punto.

Este proceso lo repetimos otras 3 veces más poniendo el robot en distintas posiciones e inclinaciones para que su controladora triangule y calcule las

coordenadas del TCP. Al pulsar OK obtendremos el resultado con el error medio y los Cuaterniones.

Dentro de la TOOL Data Declaration – name: PISTOLA debemos buscar la instrucción robhold y comprobar que está en TRUE - USER para poder trabajar con el nuevo sistema de coordenadas de la PISTOLA.

Si empleamos el método de 3 puntos y una dirección, tendremos que proceder de idéntica manera pero además, tendremos que calcular nosotros e introducir el valor de los 4 Cuaterniones para que sepa la dirección de la herramienta.

Es muy importante conocer la inclinación de la boquilla de soldadura para que el robot se acerque o se aleje de las piezas a soldar en la misma dirección que el hilo de soldar evitando así colisiones de la boquilla con las piezas.



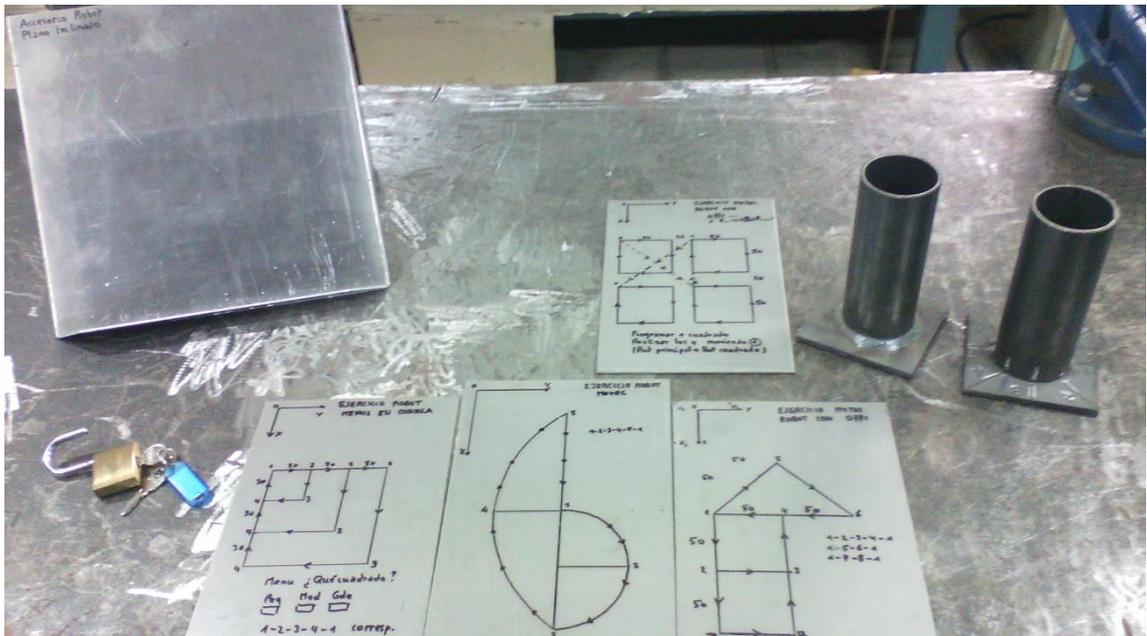
TCP

Una vez hayamos declarado la herramienta, también es necesario declarar el sistema de coordenadas sobre la mesa de trabajo (Wobj) o usuario. Para ello, al igual que antes comprobamos que estamos en Coord: Base, Tool: PISTOLA. Program Data - MAIN/USER. Pulsamos Special – Define coord. – Work Object Coordinates Definition - User method: 3 points.

Si la mesa de trabajo (Wobj) está nivelada y horizontal, con 3 puntos podemos definir el sistema de coordenadas del usuario. Pulsamos Enter, bajamos a Obj Method – pulsamos la tecla de saltar la línea doble y seleccionamos User X1 – acercamos el robot con precisión hasta el origen de coordenadas del sistema de coordenadas de la mesa de trabajo y pulsamos ModPos. para grabarlo.

Seleccionamos User X2 y teniendo bloqueado el desplazamiento en Z desplazamos la punta de la herramienta con un movimiento lineal una longitud cualquiera en dirección X y pulsamos ModPos.

Por último, seleccionamos User Y1 y repetimos la operación anterior sobre el eje Y pulsando ModPos. Pulsamos OK dos veces y ya tenemos declarado el usuario MESA (Wobj).



3.4. Programar una trayectoria de la herramienta.

Una vez definidos todos los sistemas de coordenadas, podemos programar trayectorias con comodidad seleccionando por ej. el sistema de coordenadas Wobj, moviendo el robot a los puntos de cambio de dirección y grabarlos con las órdenes MoveJ (movimiento lineal rápido sin precisión de 3 aproximación y retirada), MoveL (movimiento lineal preciso a una velocidad determinada), MoveC (movimiento circular preciso a una velocidad programada) y offs (Offset para ir a puntos relativos que se encuentran a distancias conocidas de un punto grabado anteriormente) y usando la instrucción Special - Align... podemos grabar puntos que se encuentran alineados con otros en una dirección X o Y). Antes de programar puntos, en la pantalla de visualización debemos tener: Program Routines: MAIN/PRACTICA/PRACTICA1. Coord: Wobj - Tool: PISTOLA y el procedimiento es el siguiente:

Llevamos la punta de la herramienta del robot al primer punto de la trayectoria deseada y lo grabamos pulsando sobre la instrucción MoveJ con velocidad máx. Volvemos a desplazar la herramienta al siguiente punto y lo grabamos con la instrucción MoveL indicando la velocidad deseada. Repetimos esta operación cuantas veces sea necesario. En el caso de que la trayectoria sea curva hay que definir el punto final de la curva y otro punto intermedio de dicha curva y grabarlos con la instrucción MoveC indicando la velocidad.

Para finalizar, desplazamos la herramienta a un punto de retirada teniendo la precaución de que no colisione con objetos que se interpongan en el camino, teniéndose que grabar otro punto intermedio si es necesario. Situamos el cursor sobre el primer punto MoveJ, lo copiamos y lo pegamos al final del programa o Proc. De esta forma, la herramienta ira al primer punto y se retirará a velocidad máxima y recorrerá el resto de la trayectoria a la velocidad adecuada.

Una vez tengamos el programa, es importante comprobar paso a paso que el robot hace los movimientos deseados sin colisiones y en caso necesario, habrá que modificar los puntos y las velocidades incorrectas. Posteriormente pulsando Start, el robot realizará un ciclo completo a las velocidades de trabajo. Si todo es correcto, lo pondremos en modo Producción a máximo rendimiento.

3.5. Programar una soldadura por arco.

Además de tener que programar la trayectoria del robot, a la hora de soldar por arco hay que tener en cuenta numerosos factores, como el peso de la pistola de soldadura, el tipo de materiales a soldar, grosor de las piezas, posición de soldadura, gas empleado, alimentación del hilo de soldeo, comienzo y fin de la soldadura evitando cráteres, inclinación de la boquilla de soldadura, velocidad de soldeo, forma del zigzag, tensión e intensidad de soldeo, tiempo previo de apertura de gas protector y tiempo posterior para cierre de aportación de gas, etc.

Todos esos parámetros hay que introducirlos, grabarlos y guardarlos siempre que sea posible en el módulo común USER, para recurrir a ellos cuando se vayan a soldar las mismas piezas o ejecutar uniones similares, pues resulta una ardua tarea. Por esto, para la realización de la práctica se va a partir de los datos de soldaduras probadas con anterioridad.

Los parámetros de la soldadura, velocidad de soldeo en m/s, tensión de soldeo en V y velocidad del hilo en m/min y otros, se almacenan en `welldata` (`wd`), en `weavedata` (`wv`) se encuentran los parámetros correspondientes al zigzag de la boquilla durante el soldeo y en `seamdata` (`sm`) los parámetros correspondientes al inicio y final de la soldadura.

Además de las instrucciones de movimiento vistas en el punto anterior, es necesario emplear las instrucciones `ArcL` para soldar linealmente, `ArcC` para soldar en arco, `ArcL\On` para inicio de soldadura y `ArcL\Off` para la finalización del cordón evitando la formación de cráteres al no moverse la pistola de soldadura ni el robot durante ese tiempo.

Es conveniente crear en la trayectoria puntos de soldeo ficticio previos al cordón, de forma que se abra la electroválvula del gas de protección y la pistola se mueva igual que cuando suelde, pero sin arco eléctrico. Hay que prestar también atención a los cambios de dirección de los cordones y a la forma de finalización de los mismos.

Pulsando `Start` se realizará un ciclo para modificar el valor de todos los parámetros que lo requieran antes de poner el robot en modo producción.

4. BIBLIOGRAFÍA Y WEBGRAFÍA

Calvo, José; *Manual práctico de programación de robot ABB*; curso 2017/18
Programación de sistemas automáticos de fabricación mecánica

ABB Robótica; Noviembre 2018; <https://new.abb.com/products/robotics/es/>

Software RobotStudio; Noviembre 2018;
<https://new.abb.com/products/robotics/es/robotstudio>

Software para aplicaciones; Noviembre 2018;
<https://new.abb.com/products/robotics/es/software-para-aplicaciones>

Software para soldadura por arco; Noviembre 2018;
<https://new.abb.com/products/robotics/es/software-para-aplicaciones/software-soldadura-por-arco>