

[Seleccionar fecha]

IES JUAN RUBIO
ORTIZ

MANUAL ROBÓTICA. KUKA

INDICE

A.- ROBOT	pág. 3
1. Generalidades.....	pág. 3
2. Mecánica del robot.....	pág. 3
B.- SEGURIDADES-ROBOT	pág. 5
C.- MANEJO DEL ROBOT	pág. 8
1. Panel de control (KCP).....	pág. 8
1. Barra de estado	pág. 9
2. Accionamientos arriba-derecha	pág. 10
3. Teclas izquierda.....	pág. 11
4. Parte inferior	pág. 11
5. Teclas alrededor de la pantalla.....	pág. 12
6. Space Mouse- Ratón.....	pág. 13
7. Parte trasera del KCP.....	pág. 13
2. Ejecución manual del programa.....	pág. 14
1. Inicialmente	pág. 14
2. Seleccionar tipo de movimiento (teclado/ratón/programa).....	pág. 15
3. Seleccionar modo de ejecución del programa (paso a paso/continuo).pág.	15
4. Modificar la velocidad de desplazamiento	pág. 16
5. Ejecutar líneas de programa de forma individual.....	pág. 16
3. Ejecución automática de un programa.....	pág. 18
1. Definir la velocidad	pág. 18
2. Ejecutar un programa	pág. 18
3. Detener la ejecución del programa.....	pág. 18
4. Continuar con la ejecución del programa	pág. 19
5. Resetear un programa.....	pág. 19
6. Salir de un programa	pág. 19
4. Los sistemas de coordenadas.....	pág. 20
5. Visualización y modificación de Entradas/salidas (Output/Input)	pág. 21
1. Visualización.....	pág. 21
2. Modificación.....	pág. 22
6. Visualización y modificación de variables de programa	pág. 23
1. Visualización.....	pág. 23
2. Modificación.....	pág. 24
3. Visualizacion Resumen de Variables.....	pág. 25
7. Visualización y pequeñas modificaciones de programa.....	pág. 26
1. Modo usuario/experto	pág. 26
2. Visualización de una Fold en programa.....	pág. 27
3. Algunas modificaciones en el programa.....	pág. 28
8. Apunte final	pág. 29
9. Ejercicios prácticos planteados	pág. 30

A.- ROBOT

1.- GENERALIDADES

El robot del fabricante KUKA es un robot industrial con seis ejes de articulación.

El cuerpo base está fabricado con fundición de metal liviano, lo que resulta una alta frecuencia natural del robot, presentando unas buenas propiedades dinámicas con alta resistencia a las vibraciones.

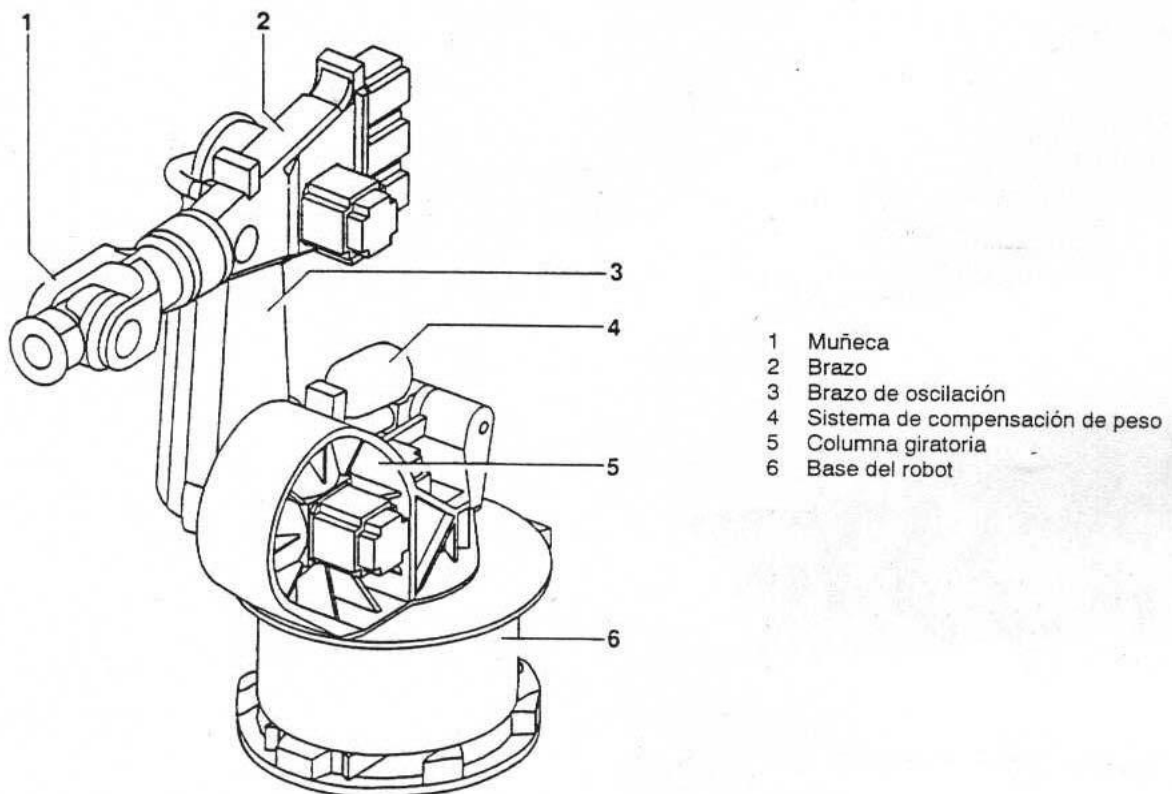
Las articulaciones y los reductores se mueven prácticamente libres de juegos, y todas las piezas en movimiento están cubiertas. Todos los motores de accionamiento son servomotores de CA sin escobillas, en técnica enchufable y libres de mantenimiento, asegurados contra sobrecargas.

Los ejes principales son del tipo de lubricación permanente. Es necesario un cambio de aceite solo después de 20.000 horas de servicio. Es rápido y brinda seguridad de servicio, requiriendo un mantenimiento mínimo y sencillo. Tienen una vida útil promedio entre 10 y 15 años, según el uso.

Está equipado con una unidad de control, cuya electrónica de mando y de potencia se encuentran juntas e integradas en un armario de control.

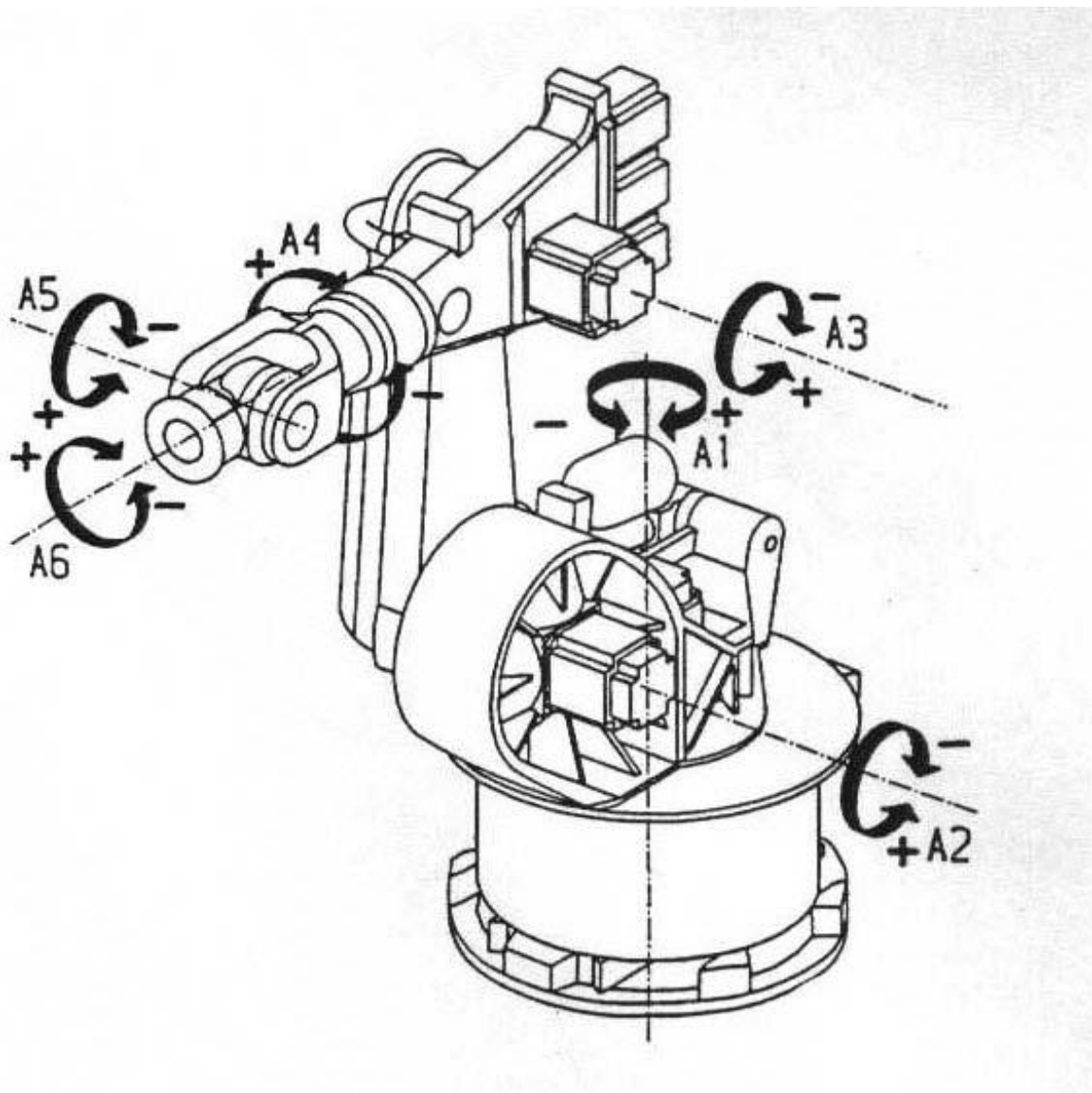
2.- MECÁNICA DEL ROBOT

El robot consta de una base fija (6), sobre la cual gira, alrededor de un eje vertical, la



“c

Las posibilidades de los movimientos del robot pueden verse en la siguiente figura:



La carga y el peso propio de los componentes de la articulación se compensan estáticamente, casi completamente por medio de un sistema hidroneumático de compensación de peso de circuito cerrado. Este sistema soporta el eje 2, y puede variarse su efecto de acuerdo con la carga y posición del robot.

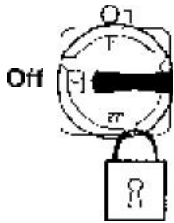
La medición de los trayectos para los ejes se realiza a través de un sistema de medición absoluto con un resolver para cada eje.

El accionamiento se efectúa por medio de servomotores de CA de baja inercia, de mando transistorizado. En las unidades motrices van integrados el resolver y el freno.

El campo de trabajo del robot es limitado en todos los ejes por medio de fin de carrera software, además limitándose los ejes 1,2,3,5 por medio de topes finales mecánicos de absorción de energía.



B.- SEGURIDADES – ROBOT



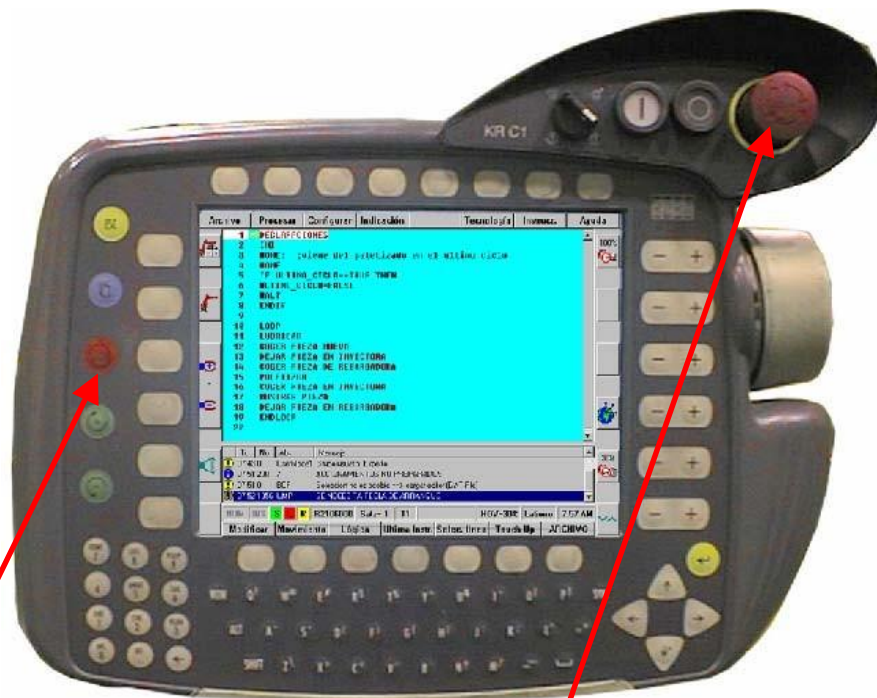
- Para realizar trabajos de reemplazo, ajuste, mantenimiento o reparación, debe desconectarse el sistema de robot y asegurarlo contra una puesta en marcha indebida mediante uno o varios candados.

- Después de una colisión a velocidad mayor a la de desplazamiento manual contra los topes de uno o varios ejes, deben reemplazarse inmediatamente los topes afectados por nuevos.
- El operario deberá informar inmediatamente a su responsable de cualquier anomalía que observe en el robot, y utilizarlo solo cuando este esté en condiciones idóneas.
- Cualquiera (responsables o personal de mantenimiento) que haga una modificación en el robot que pudiera incidir sobre las seguridades, deberá avisar inmediatamente a todos y cada uno de los operarios, debiéndole ser aprobada previamente tal modificación por el Coordinador de Seguridad e Higiene.
- Cuando se tenga que entrar dentro del recinto cerrado, la puerta deberá permanecer abierta durante todo el tiempo que haya una persona dentro, por lo tanto cuando alguien vaya a cerrar la puerta deberá asegurarse de que no hay nadie dentro, y siempre, uno de los que está dentro deberá tener la llave que rearma el micro de la puerta.
- Cuando haya que mover el robot para trabajos de ajuste o programación que requieran entrar dentro de la célula, solo podrá haber una persona dentro, la que está en posesión de la consola de programación, estará permanentemente atento a los movimientos del robot, tendrá la puerta abierta y trabajará en modo de funcionamiento manual-lento, o sea, en T1. **Nunca en T2** (manual rápido).
- Para trabajos en el sistema de compensación de peso, debe asegurarse mecánicamente el eje en el cual apoya el mismo, para evitar movimientos independientes.

- En caso de fallos, puede moverse mecánicamente el robot a través de los accionamientos de los ejes principales. Debe ser utilizado exclusivamente en caso de emergencia (p.e. liberar a una persona). Esto ha de hacerse con el interruptor principal del armario desconectado y candado. Una vez movido el robot mecánicamente, deben ajustarse todos los ejes del robot.
- Los motores, durante el servicio alcanzan temperaturas que pueden causar quemaduras en la piel.
- Si por cualquier motivo se ha de sacar un servomotor (por avería o cambio de este), el eje del servomotor que se va a quitar deberá estar firmemente sujeto, ya que dejarán de actuar los frenos y ese eje se caerá con el peligro de colisión que puede originar. Además, si este eje se mueve, habrá que volver a ajustar todos los ejes del robot.
- Nunca desconectar una manguera estando conectado el robot.



Tipos de parada:




Stop (botón rojo). Esta es la que habrá que realizar preferentemente, el robot hace una parada suave, no se sale de su trayectoria.

Parada de emergencia (seta de la botonera, seta general de la célula,). Esta es una parada que habrá que hacer siempre que sea necesario, pero no de forma habitual. El robot hace una parada brusca, en seco, saliéndose de su trayectoria.

Para volver a la ejecución de programa

Para volver a la ejecución de programa

1. Pulsar al botón de marcha (botón verde). 

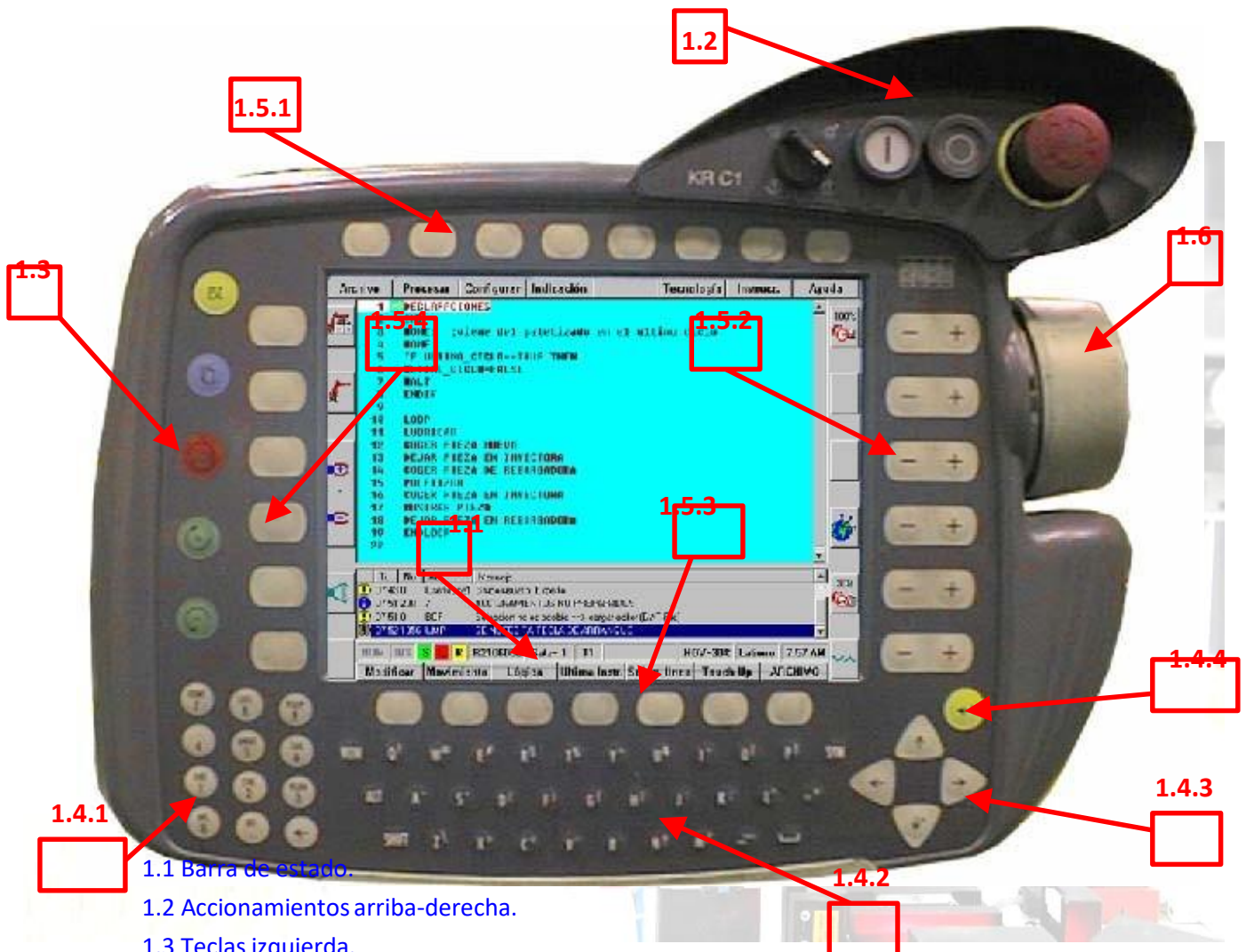
1. Levantar la seta de emergencia en la consola.
2. Confirmar esa parada en el menú de la consola.
3. Pulsar al botón de conectar accionamientos (botón blanco).
4. Pulsar al botón de marcha (botón verde).



C.- MANEJO DEL ROBOT

1. EL PANEL DE CONTROL (KCP)

En este están todos los elementos para la programación y operación del sistema robot, con excepción del interruptor principal.



1.1 Barra de estado.

1.2 Accionamientos arriba-derecha.

1.3 Teclas izquierda.

1.4 Parte inferior.

1.4.1 Campo numérico

1.4.2 Teclado

1.4.3 Teclas de cursor

1.4.4 Tecla de entrada

1.5 Teclas alrededor de la pantalla.

1.5.1 Teclas de grupos de menús

1.5.2 Teclas de funciones de estado derecha

1.5.3 Softkeys

1.5.4 Teclas de funciones de estado izquierda

1.6 Space mouse-ratón.

1.1 Barra de estado.

Esta suministra información sobre los estados importantes de servicio. A continuación se muestran algunos ejemplos. De izquierda a derecha, esta muestra:

NUM	INS	S	I	R	R2106000	Satz= 11	T1		HOV=30%	Latierro	8:48 AM
NUM	INS	S	⊙	R	R2106000	Satz= 1	T1		HOV=30%	Latierro	7:52 AM
NUM	INS	S	I	R	R2106000	Satz= 1	AUT		POV=100%	Latierro	8:06 AM
NUM	INS	S	I	R	R2106000	Satz= 11	T1		POV=1%	Latierro	8:09 AM
NUM	INS	S	⊙	R	R2106000	Satz= 11	T1		POV=1%	Latierro	8:11 AM

NUM.- Están activos los números del teclado, no las funciones.

INS.- Está activo el modo de inserción de texto.

S.- (verde) El interpretador submit está en ejecución, siempre en verde, si estuviera en rojo, es porque se ha detenido voluntariamente la ejecución de este, y puede conllevar problemas graves de seguridad.

I.- (verde) Accionamientos conectados.

R.- (verde) Ejecutándose un programa. (gris)

No hay programa seleccionado.

(rojo) El programa seleccionado y arrancado ha sido detenido. (Amarillo/negro) El puntero está sobre la primera/ultima línea de programa.

PROG_SEL. Se visualiza el nombre de programa en ejecución.

Satz=5. Se visualiza el número de paso de la línea de programa en ejecución.

MODO SERVICIO:

T1.- Manual lento. **T2**

.- Manual rápido.

AUT.- Automático.

EXT.- Automático externo, cuando un ordenador o plc asume el control del robot.

?.- No se ha seleccionado ningún modo.

PROG_EDIT. Se visualiza el nombre de programa en edición. **POV/HOV.-** el

% de la velocidad respecto a la de programa/manual. **NOMBRE_ROBOT.-**

Se visualiza el nombre del programa.

HORA .- Hora actual del sistema

1.2 Accionamientos (arriba-derecha)

De derecha a izquierda:



1.- PULSADOR DE PARADA DE EMERGENCIA.

Representa el dispositivo de seguridad más importante, se acciona en los casos de peligro. Para que los accionamientos puedan volver a ser conectados es necesario desenclavar la tecla de parada de emergencia y confirmar el correspondiente mensaje en la ventana de mensajes.

2.- ACCIONAMIENTOS DESCONECTADOS.

Se utiliza cuando trabajamos en automático. En T1 o T2 los accionamientos se desconectan al soltar el hombre muerto.

3.- ACCIONAMIENTOS CONECTADOS.

Se utiliza cuando trabajamos en automático. En T1 o T2 los accionamientos se conectan al pulsar el hombre muerto.

4.-SELECTOR MODO DE SERVICIO. Hay cuatro modos: T1.-

Manual lento.

T2 .- Manual rápido. Peligroso, no lo emplearemos nunca.


AUT.- Automático.


EXT.- Automático externo, cuando un ordenador o plc. asume el control del robot.

1.3 Teclas izquierda.





De arriba abajo:

 (amarillo) para interrumpir la programación de una función o para cerrar menús abiertos accidentalmente.

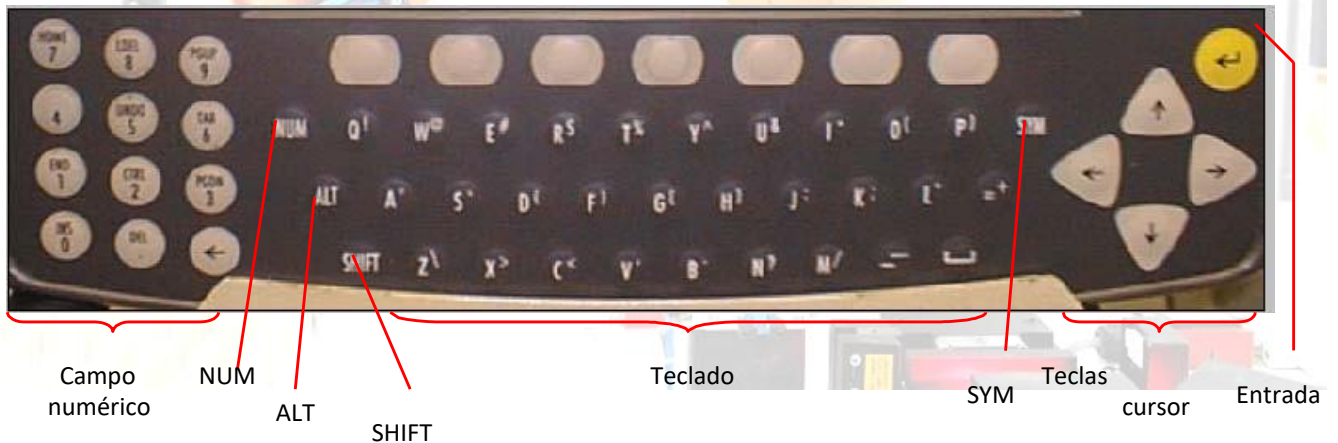
 (azul) Tecla de selección de ventana, para conmutar entre ventanas abiertas.

 (rojo) Parada de programa.

 Arranque adelante.- (verde) Pulsando esta tecla se arranca un programa seleccionado. Para ejecutar un programa en modo T1 o T2 debe estar pulsado previamente el pulsador de hombre muerto.

 Arranque atrás.- (Verde) No emplear nunca., es para moverse hacia atrás en el programa, es útil solo programando. Este solo está disponible en T1 y T2, y en modo paso a paso.


1.4 Parte inferior.



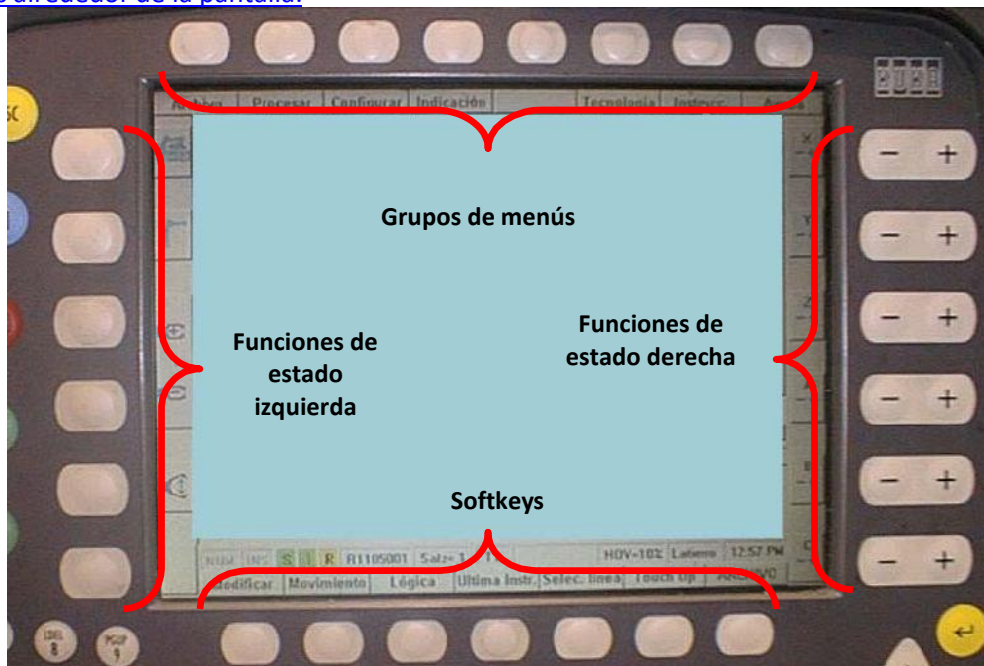
De izquierda a derecha:

- **Campo numérico.**- A través de este se da entrada a cifras, y en un segundo nivel, conmutando mediante la tecla **NUM** se tiene acceso a unas funciones de mando del cursor que no emplearemos por no ser necesario (home, ldel, pgup, undo, tab, end, ctrl, pgdn, ins, del). Para evitar confusiones podemos optar por tener siempre activado el modo numérico (NUM en negro)
- **Teclado.**- Para insertar caracteres. Para insertar un carácter en mayúscula se efectúa con la tecla **SHIFT**, y para varios teniéndola pulsada. Se dispone de caracteres especiales en un segundo nivel, al que se accede al pulsarse la tecla **SYM**. La función de la tecla **ALT** es similar a

la de un ordenador, nosotros apenas la emplearemos, quizá para abrir un menú desplegable pulsando ALT- flecha abajo.

- **Teclas de cursor.**- Sirven para modificar la posición del cursor de edición y para cambiar la posición entre los campos de los formularios inline y listas de parámetros.
-  **Tecla de entrada.**- (amarillo) Es similar a la tecla "Enter" de un PC.

1.5 Teclas alrededor de la pantalla.



En sentido horario, empezando desde arriba:

- **Teclas de grupos de menús.**- Mediante estas teclas se abre un menú, por el que nos podemos mover con las teclas de cursor y accionar una función mediante la tecla enter.
- **Teclas de funciones de estado derecha.**- Sirven para la selección de opciones de servicio, para conmutar funciones individuales y para definir valores. Estas son:
 - Teclas de movimiento manual con teclas (por ejes, por coordenadas) o por ratón.
 - Teclas de velocidad e movimiento, manual y automático.
 - Teclas de brillo y contraste de la pantalla.
- **Softkeys.**- Para seleccionar las funciones representadas y que se adaptan dinámicamente.
- **Teclas de funciones de estado izquierda.**- Sirven para la selección de opciones de servicio, para conmutar funciones individuales y para definir valores. Están:
 - Tecla de selección de tipo de movimiento (teclas, ratón o no manual).
 - Tecla de modo de ejecución de programa (paso a paso, continuo o instrucción a instrucción).
 - Tecla de accionamiento de la pinza.

1.6 Space mouse-ratón.



Space mouse-ratón. Sirve para el movimiento en modo manual del robot, se desaconseja su uso por llevar mucho tiempo el acostumbrarse a su uso.

1.7 Parte trasera del KCP.



1.7.2 Botón blanco.- pulsador de hombre muerto.

1.7.1 Dos pulsadores alargados blancos.- son pulsadores de hombre muerto.

1.7.3 Botón verde.- pulsador arranque de programa hacia delante.

Estos pulsadores sirven para mover al robot en modo manual (T1 o T2), para lo cual será necesario pulsar primeramente el hombre muerto para conectar los accionamientos, estos pulsadores de hombre muerto tienen tres posiciones.

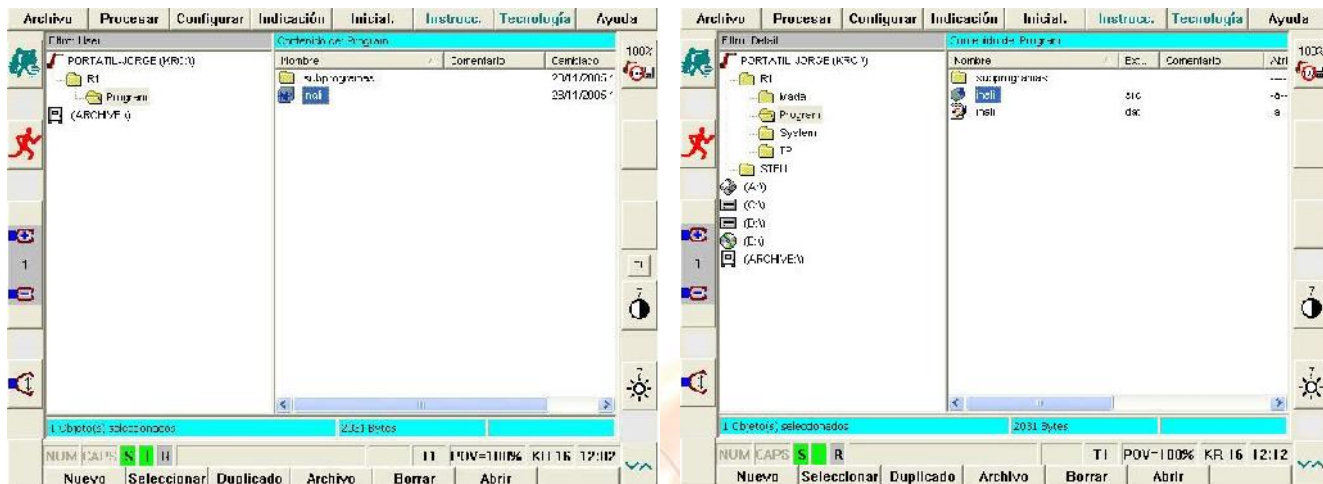
- 1- Sin presionar Motores OFF
- 2- Presionando primera posición Motores ON
- 3- Presionando hasta tope Motores OFF.

Seguidamente moverlo bien por coordenadas o por ejes pulsando la coordenada correspondiente o por programa pulsando la tecla de avance de programa.

2. EJECUCIÓN MANUAL DEL PROGRAMA

2.1 Inicialmente

Para estar en este punto el selector de modo de funcionamiento ha de estar en T1/T2.



Modo Usuario

Modo Experto

Para abrir un programa bastará con desplazarnos con las teclas de cursor y pulsar la tecla enter sobre el nombre de programa que queremos abrir (en modo experto este deberá tener extensión SRC).



Si hay ya un programa abierto y queremos cerrar el programa actual, dar a la tecla para abrir el menú PROCESAR y seleccionar CANCELAR PROGRAMA y apareceremos de nuevo en la pantalla donde se ven todos los programas.

Con el programa seleccionado, el robot ejecutara el programa cuando nosotros estamos pulsando el hombre muerto (blanco) y marcha (verde), caso contrario se interrumpe. El robot ejecuta el programa deteniéndose en las instrucciones de entrada de señal (IN) si no las recibe, permitiéndonos simularlas. Las salidas (OUT) las activa.



Este modo de ejecución de programa en manual se utilizará solo para ajustes, cuando se está realizando el programa o no se está del todo seguro de que esté bien y **nunca en T2**. No es peligroso en cuanto a movimientos si se ejecuta en T1, pero es fácil que sino se conoce bien el programa simulemos una entrada en un momento no correcto, se caiga la pieza que estamos manipulando o que el robot colisione.

2.2 Seleccionar



Movimiento con el ratón.



Movimiento con las teclas + / -



Movimiento manual desactivado.

La selección del tipo de movimiento se efectúa pulsando la función de estado de la izquierda, la que está arriba del todo, y al pulsarla vamos alternando entre un tipo de movimiento y otro.

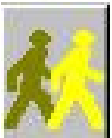
El tipo de movimiento estará preferiblemente en manual desactivado, lo que solo nos permite ejecutar el programa. Cuando queramos movernos en manual fuera de los puntos del programa bien porque queremos modificar la posición de un punto, porque queremos sacar al robot de una posición después de una colisión o por alguna situación atípica, en tal caso activaremos el movimiento manual con teclas, lo que nos permitirá ejecutar el programa y también desplazarnos fuera de los puntos del programa.

[2.3 Seleccionar modo de ejecución del programa. \(paso a paso/continuo\)](#)

Seleccionar el modo de ejecución del programa, accionando la tecla de función de la izquierda la segunda empezando desde arriba. Hay tres modos:



Continuo (hombre rojo corriendo), en el que la ejecución de programa no se detiene hasta que no se suelte la tecla de marcha.



Paso a paso (dos hombres andando), en el que la ejecución de programa se detiene en cada paso.



Instrucción por instrucción, no emplear, cuando vemos una instrucción, en realidad para el pueden ser varias.

Hay que tener mucho cuidado ya que para el robot las entradas y salidas no son entendidas como pasos, en las entradas espera a que simulemos, pero las salidas son activadas y continua con la siguiente. Para el robot de paso a paso es de punto a punto, realizando todo lo que esté entre estos, parándose en las entradas que habrá que simular.

2.4 Modificar la velocidad de desplazamiento.

Hay dos velocidades de movimiento del robot:



Velocidad de ejecución de programa (tecla de función de estado derecha arriba). A esta se le denomina POV=100% y esta aparecerá en la barra de estado.



Velocidad en manual fuera de programa (tecla de función de estado derecha abajo, en la que aparece una mano). A esta se le denomina HOV=100% y esta aparecerá en la barra de estado.

Estos valores pueden ir modificándose según nos movemos. Para pequeños ajustes será necesario reducirla a un 3% o incluso un 1%, mientras que fuera de todo peligro de colisión se podrá llevar al 100%.



Tener cuidado, no confundirlas, por ejemplo, puede pasar que queramos movernos fuera del programa, se baja la velocidad de ejecución de programa y al moverlo vemos que este se mueve muy deprisa, pudiendo haber hecho que colisione porque era la velocidad fuera de programa la que teníamos que haber modificado.

2.5 Ejecutar líneas de programa de forma individual.

Hay que distinguir entre el cursor de edición “ | ” y el puntero de programa (línea blanca con una flecha amarilla), que nos indica cuál es la línea de programa que se está ejecutando. Cuando se ejecutan ordenes (abrir folder, borrar línea,..) siempre se harán sobre el cursor de edición, excepto la ejecución del programa que siempre será sobre el puntero de programa.

```
1  DECLARACIONES
2  INI
3  HOME: ;viene del paletizado
4  HOME
5  IF ULTIMO_CICLO==TRUE THEN
6  ULTIMO_CICLO=FALSE
7  HALT
8  ENDIF
9  |
10 LOOP
11 → LUBRICAR
12 COGER PIEZA NUEVA
13 DEJAR PIEZA EN INYECTORA
14 COGER PIEZA DE REBARBADORA
15 PALETIZAR
16 COGER PIEZA EN INYECTORA
17 MOSTRAR PIEZA
18 DEJAR PIEZA EN REBARBADORA
19 ENDLOOP
```

Cursor de edición
(parpadeante)

Línea de ejecución de
programa

Para moverse a otra línea de ejecución de programa, mover el cursor de edición, y pulsar la tecla de función de abajo donde pone “**Selec. linea**”.



Tener mucho cuidado al realizar esto, ya que hay que conocer el programa para saber qué es lo que estamos haciendo, y tener muy en cuenta que al saltar a otra posición, el primer desplazamiento lo hará fuera de trayectoria conocida, con el consiguiente peligro de colisión.



Es necesario siempre ejecutar primeramente la línea **INI ()** del programa que contiene datos necesarios para la ejecución del programa. Si por ejemplo, seleccionamos un programa y sin ejecutar INI () vamos a otro punto del programa nos dará continuos errores.



3. EJECUCIÓN AUTOMÁTICA DE UN PROGRAMA

Primeramente se ha de seleccionar un programa, y el selector de modo de servicio en automático.

3.1 Definir la velocidad de ejecución



Modificando la tecla de función de estado derecha arriba

3.2 Ejecutar un programa.

Conectar los accionamientos del robot, y a continuación dar al pulsador de marcha. El primer paso lo hará en lento, hasta alcanzar COI (punto de coincidencia) y a partir del siguiente a la velocidad programada y al % puesto por el usuario en la KCP.

El desplazamiento puede ser paso a paso o en continuo.

3.3 Detener la ejecución del programa.

Pulsando la tecla stop.



[3.4 Continuar con la ejecución del programa.](#)

Si solo se dio a la tecla de stop bastará con dar a marcha (tecla verde), si se abrió la puerta, se accionó una seta... es necesario rearmar esa emergencia, confirmar esa emergencia en el teclado, conectar los accionamientos y dar al pulsador de marcha.

Para confirmar la emergencia en el teclado tendremos que tener activa la pantalla de mensajes (la de abajo), si esta no está activa, mediante la tecla azul para la selección de ventana, iremos pulsando hasta que esté activa esta, entonces aparecerá un softkey donde pone "Confirmar" y otro "Confirmar todos", pulsando a cualquiera de estos siempre que la emergencia haya sido quitada conseguiremos confirmarla, y poder seguir con la ejecución de programa.

[3.5 Resetear un programa.](#)

[3.6 Salir de un programa.](#)



4. LOS SISTEMAS DE COORDENADAS

Para el desplazamiento manual del robot, se debe seleccionar un sistema de coordenadas respecto al cual se puedan hacer referencia los movimientos del robot. Se disponen de cuatro sistemas:



Sistema de coordenadas específicas de ejes, en el cual cada eje puede ser desplazado individualmente en dirección positiva o negativa.



Sistema de coordenadas de la herramienta. El origen se encuentra en la herramienta.



Sistema de coordenadas de la pieza. El origen se puede tomar en una pieza sobre la que se trabaja, en nuestro caso este no es de mucha utilidad, y se puede decir que para el movimiento en manual, este coincide con el universal.

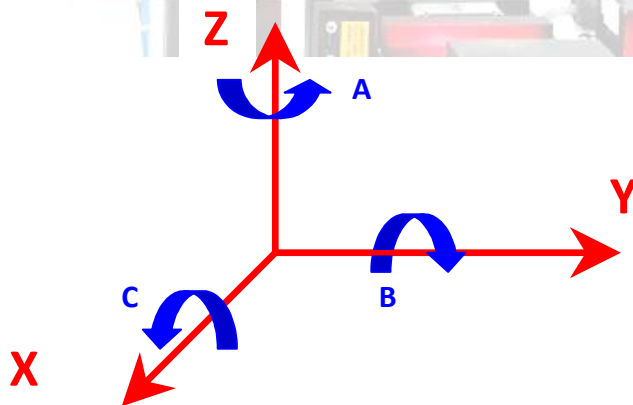


Sistema de coordenadas universales. Su origen se encuentra en la base del robot.

Se recomienda utilizar el de coordenadas universales, aunque para tener un mayor control del robot, y sacarlo de alineamientos de ejes puede resultar más efectivo el de coordenadas específicas de ejes.

Recordar que las rotaciones son:

- A giro en torno al eje Z.
- B giro en torno al eje Y.
- C giro en torno al eje X.



5. VISUALIZACIÓN Y MODIFICACIÓN DE ENTRADAS/SALIDAS.

5.1 Visualización.

Indicación	Inicial.	Instrucc.	Tecn
0 Entradas/Salidas		0 Entrada Digital	
1 Posición actual		1 Salida Digital	
2 PANTALLA INALI		2 Salidas Analógicas	
3 Variable		3 Garra	
4 Diagnóstico		4 Automático externo	
5 Windows			
6 Editor de Iconos			

Indicación	Inicial.	Instrucc.	Tecn
0 Entradas/Salidas		0 Entrada Digital	
1 Posición actual		1 Salida Digital	
2 PANTALLA INALI		2 Salidas Analógicas	
3 Variable		3 Garra	
4 Diagnóstico		4 Automático externo	
5 Windows			
6 Editor de Iconos			

Estas entradas/salidas pueden visualizarse en cualquier momento del proceso, para ello habrá que seleccionar la opción de menú "INDICACIÓN", se abre un menú de selección, seleccionamos la "0 Entradas/Salidas", para ver las entradas seleccionaremos la opción "0 Entrada Digital" y para ver las salidas, la opción "1 Salida Digital". Se nos abrirá una nueva ventana en la que podremos ver el estado de estas.



5.3 Modificación.

	Tecnología	Instrucc.	Ayud
<input type="radio"/>	1	LUBRICADOR	Producto 1 Plato Fij
<input type="radio"/>	2	LUBRICADOR	Producto 1 Plato M
<input type="radio"/>	3	LUBRICADOR	Aire Plato Fijo
<input type="radio"/>	4	LUBRICADOR	Aire Plato Movil
<input type="radio"/>	5	LUBRICADOR	Avance
<input checked="" type="radio"/>	6	LUBRICADOR	Retorno
<input type="radio"/>	7	PINZA	Abrir
<input type="radio"/>	8	PINZA	Cerrar
<input type="radio"/>	9	PALET LLENO	
<input type="radio"/>	10	LUBRICADOR	Producto 2 Plato Fij
<input type="radio"/>	11	LUBRICADOR	Producto 2 Plato M
<input type="radio"/>	12	-	-
<input type="radio"/>	13	-	-
<input type="radio"/>	14	REBARBADORA	Avanzar Centrac
<input type="radio"/>	15	REBARBADORA	Retornar Centrac
<input type="radio"/>	16	REBARBADORA	Iniciar Ciclo
<input type="radio"/>	17	INYECTORA	Avanzar Expulsion
<input type="radio"/>	18	INYECTORA	Retornar Expulsion
<input type="radio"/>	19	INYECTORA	Inicio Ciclo

HOV=30% Latierro 8:48 AM

Modificar Entradas Cerrar



La garra puede ser accionada también de forma diferente. En modo manual (T1/T2), al pulsar hombre muerto, en las teclas de funciones de estado izq., abajo aparece dibujada una garra, al accionarla activamos la salida de la garra. Hay que pulsar esta tecla con el hombre muerto pulsado, y a continuación comprobar si ha hecho lo que queríamos



Al abrir/cerrar la pinza de esta forma comprobar siempre si ha hecho finalmente lo que queríamos, porque el robot no sabe si tiene la garra cerrada o abierta, y al dar a esa tecla va activando una vez abrir y otra cerrar alternativamente. Si no ha hecho lo que queríamos (suele pasar en el 90% de los casos) habrá que pulsar de nuevo. Tener cuidado con esto, si le mandamos ir a coger una pieza con la garra cerrada habrá una colisión.



Tener mucho cuidado al activar estas salidas sino se sabe lo que se está haciendo. Podemos causar alteraciones en el ciclo de trabajo. Por lo tanto se recomienda no modificar estas a no ser que sea para cosas especiales, como sacar el robot de una posición

...

6. VISUALIZACIÓN Y MODIFICACIÓN DE VARIABLES DE PROGRAMA.

6.1 Visualización.



Las variables del programa pueden visualizarse y modificarse en cualquier momento del proceso, para ello habrá que seleccionar la opción de menú "INDICACIÓN", se abre un menú de selección, seleccionamos la "2 Variable", y seleccionaremos la opción "0 Unicos"

Se nos abrirá una nueva ventana, en la que podremos ver cuatro campos, dos de ellos a blanco (los que podemos modificar) que son Nombre y Valor nuevo, y otros dos a gris (los que no podremos modificar) que son Valor actual y módulo.

The screenshot shows the software interface with the 'Indicación' menu open. The main window displays a list of instructions (1-19) with line 11 'LUBRICAR' highlighted. A variable modification window is open on the right, showing the following fields:

- Nombre: posicion (dropdown menu)
- Valor actual: 12 (text field)
- Valor nuevo: (empty text field)
- Modul: "/R1/R2106000.SRC" (text field)

At the bottom of the interface, there is a status bar with the following information:

NUM	INS	S	C	R	R2106000	Satz= 11	T1	HOV=30%	Latierro	8:50 AM
					VarKorDataPa	Nombre	Nuevo valor	Cerrar		

Below the status bar, there is a message log with the following entries:

Ti...	No	Abs.	Mensaje
08:09	1350		COI ALCANZADA
08:49	200	/	ACCIONAMIENTOS NO PREPARADOS

Para movernos por estos campos, la manera más cómoda es pulsar la tecla del campo donde queremos situarnos. Por ejemplo, si queremos situarnos en el campo Nombre, pulsar la tecla softkey de abajo donde pone “Nombre”.

Para seleccionar la variable a modificar tendremos que situarnos en el campo pulsando la tecla NOMBRE. A continuación tenemos varias posibilidades:

Escribir el nombre de la variable y pulsar enter.

Movernos con las teclas arriba y abajo hasta encontrar la variable.

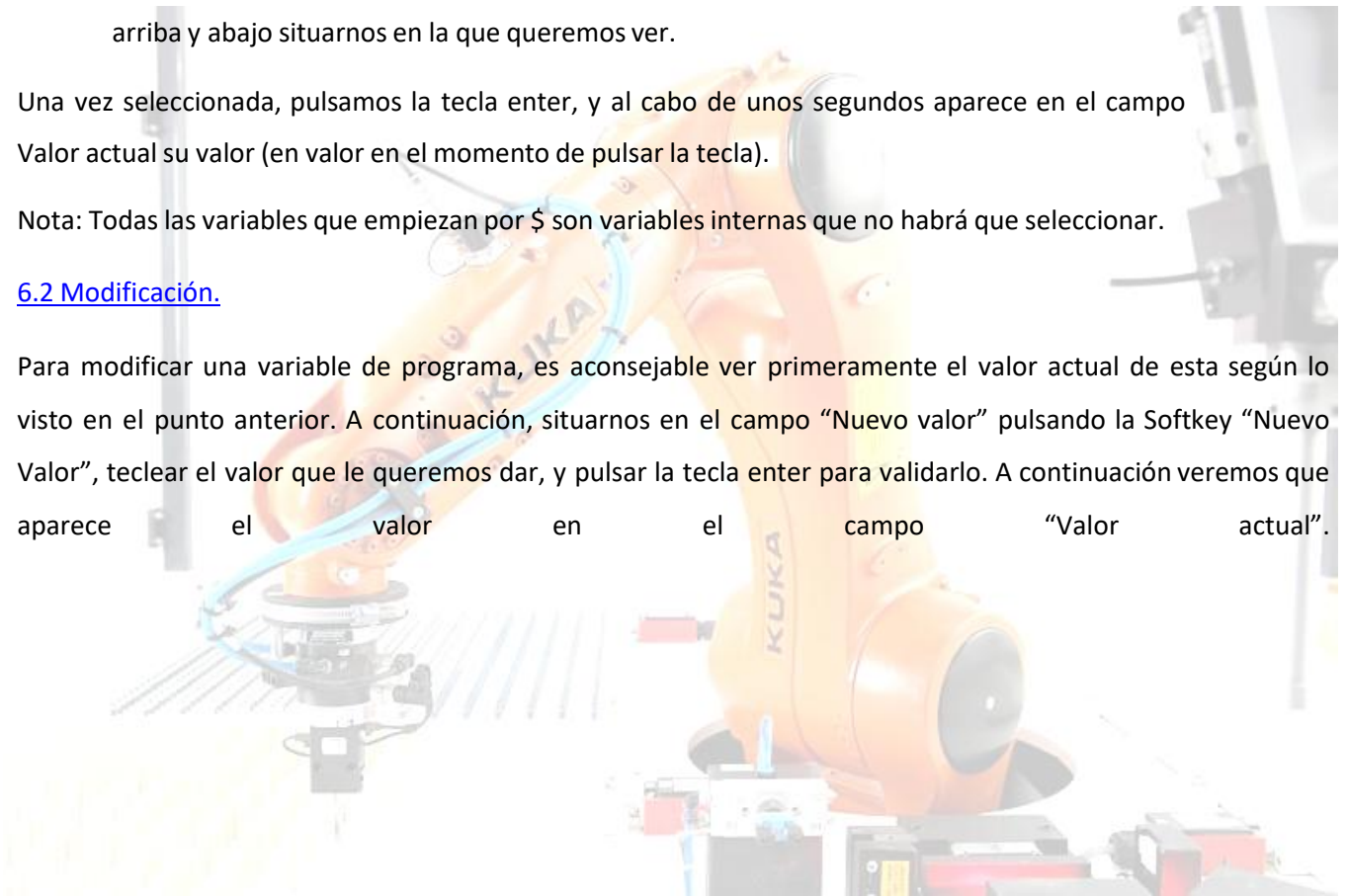
Presionar ALT y flecha abajo para abrir el desplegable y ver todos los nombres, entonces con las teclas arriba y abajo situarnos en la que queremos ver.

Una vez seleccionada, pulsamos la tecla enter, y al cabo de unos segundos aparece en el campo Valor actual su valor (en valor en el momento de pulsar la tecla).

Nota: Todas las variables que empiezan por \$ son variables internas que no habrá que seleccionar.

6.2 Modificación.

Para modificar una variable de programa, es aconsejable ver primeramente el valor actual de esta según lo visto en el punto anterior. A continuación, situarnos en el campo “Nuevo valor” pulsando la Softkey “Nuevo Valor”, teclear el valor que le queremos dar, y pulsar la tecla enter para validarlo. A continuación veremos que aparece el valor en el campo “Valor actual”.



6.3 Visualización Resumen de variables.



Para visualizar un conjunto de variables. Ir a “Indicaciones 2-variables 1-resumen 0-visualización “se nos abrirá otra pantalla donde podemos comprobar el estado y modificarlo si es necesario.



No modificar nunca variables que comiencen con \$.

Tener mucho cuidado al modificar estas variables, ya que podemos causar desperfectos.



7. VISUALIZACIÓN Y PEQUEÑAS MODIFICACIÓN DE PROGRAMA.

7.1. Modo de usuario/experto.

Existen dos grupos de usuario llamados “Usuario” y “Experto”. Siempre se trabajará en modo usuario, modo que es mucho más sencillo y que se caracteriza por su mayor facilidad para la selección de programas y porque no permite realizar tantas modificaciones en el programa como el modo experto. El modo experto permite realizar muchas más modificaciones, nosotros utilizaremos este modo fundamentalmente y porque es imprescindible para abrir las folder (agrupamiento de instrucciones en una sola línea, nosotros vemos solo “lubricar”, esto es una folder, que en realidad, si la abrimos veremos que tiene instrucciones).



Siempre trabajaremos en modo “usuario”, a no ser que tengamos que abrir una folder para modificar el programa.

The screenshot shows a control panel interface with a menu on the left and a configuration window on the right. The menu has the following items:

Archivo	Procesar	Configurar	Indicación
10	LUBRICAR	0 Entradas/Salidas	
11	COGER PII	1 Driver de E/S	
12	DEJAR PII	2 Interpretador SUBMIT	
13	COGER PII	3 Teclas de estado	
14	PALETIZAR	4 ConfigOVERRIDE	
15	COGER PII	5 Grupo de usuario	
16	MOSTRAR I	6 Ajustes	
17	DEJAR PII	7 Reinicializar USERTech	
18	ENDLOOP		
19			
20	HOME		

The configuration window on the right is titled "Grupo usuario" and contains the following text:

Grupo usuario
Seleccione un grupo de usuario

Grupo usuario actual:
Experto

At the bottom of the interface, there is a status bar with the following information:

Ti...	No	Abs.	Mensaje
08:09	1350		COI ALCANZADA
08:49	200	/	ACCIONAMIENTOS NO PREPARADOS

Below the status bar, there are several indicators and buttons:

NUM	INS	S	C	R	R2106000	Satz= 10	T1	HOV=30%	Latierro	8:55 AM
Usuario	Experto									Cerrar

Para modificar el grupo de usuario, habrá que pulsar la tecla de la barra de menú “Configuración” y a continuación desplazarse y seleccionar la opción “5 Grupo de usuario”. Entonces nos aparecen en las softkey de abajo a la izquierda dos teclas donde seleccionaremos el grupo de usuario deseado, “Usuario” o “Experto”. Para pasar a modo experto, se pide la contraseña.



¡Atención! Para modificaciones en el programa es preferible que el robot esté parado, fundamentalmente porque al modificar una línea mediante el cursor de edición, a continuación el cursor de puntero de programa salta a esta línea modificada, con lo que podría producirse un salto de una parte del programa a otra con el consiguiente peligro de colisión. Una vez modificada una línea, hay que llevarla a la línea de programa donde debe continuar el programa.



Nota: si realizamos cambios en el programa estos cambios no se habrán almacenado. Hasta cancelar el programa si los cambios fueron hechos con un programa seleccionado, si estamos en modo edición al cerrar nos preguntara que hacemos con los cambios, **|| Cancelar || || Si || No ||**

Para Realizar copia de seguridad. “**Archivo 3 Archivo 0 Todo** “ se guarda una copia en disquete.

7.2 Visualización de una FOLD en programa.

Procesar	Configurar	Indicación
0 FOLD		0 FOLD actual abr/cer
1 EditLineDelete		1 Abrir todas FOLD
2 Resetear programa		2 Cerrar todas FOLD
3 Cancelar programa		
4 Optimización Online		

Para abrir una folder, llevamos el puntero de edición a esa línea, y a continuación pulsamos la tecla para abrir el menú “Procesar”, donde nos aparecen tres opciones:

- **0 FOLD actual abrir/cerrar.** Con esta opción abrimos solo la folder en la que tenemos el punto de edición, esta es la opción más aconsejable, ya que si abrimos todas es fácil perderse en el programa. Si la folder está abierta, al pulsar esta opción lo que hacemos es cerrarla.
- **1 Abrir todas las FOLD.**
- **2 Cerrar todas las FOLD.**

Dentro del programa, vemos por ejemplo el siguiente tipo de instrucciones:

- **PTP.-** Ir a un punto.
- **LIN.-** Ir a un punto en línea recta.
- **CIR.-** Describir una circunferencia.

Cuidado, el robot solo tiene puntos de destino, y va a ellos desde donde esté, por el camino más corto, si desplazamos el robot y luego le hacemos ir a un punto irá por el camino más corto para él, el que lleva menos tiempo pudiendo funcionar todos los servomotores a la vez.

- **WAIT FOR IN.**-Espera a que le llegue señal de..
- **OUT.**- Activa (TRUE) o desactiva (FALSE) una salida.
- **PULSE.**- Activa (TRUE) una salida por un tiempo.
- **;mensaje.**- mensaje informativo.
-

7.3 Algunas modificaciones en el programa.

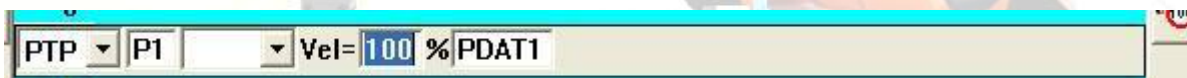
7.3.1 Modificar la posición (coordenadas) de un punto.

Desplazamos el robot a la nueva posición a grabar, situamos el cursor de edición sobre la línea/instrucción de ese punto, pulsamos la tecla softkey “Modificar”, y luego a la tecla softkey “Touch Up”, nos pide confirmar si de verdad queremos modificar ese punto, pulsamos a la tecla softkey “Si” y ya se ha almacenado. (Recordar hacer copia de seguridad después de cada modificación).

7.3.1 Modificar la velocidad de un punto.

Lo que vamos a hacer es modificar la velocidad de desplazamiento.

Situamos el puntero de edición sobre el punto a modificar, pulsamos el softkey “Modificar”, entonces se nos abre un menú con todos los detalles de este punto, como son su nombre, velocidad y datos de aceleración y aproximación.

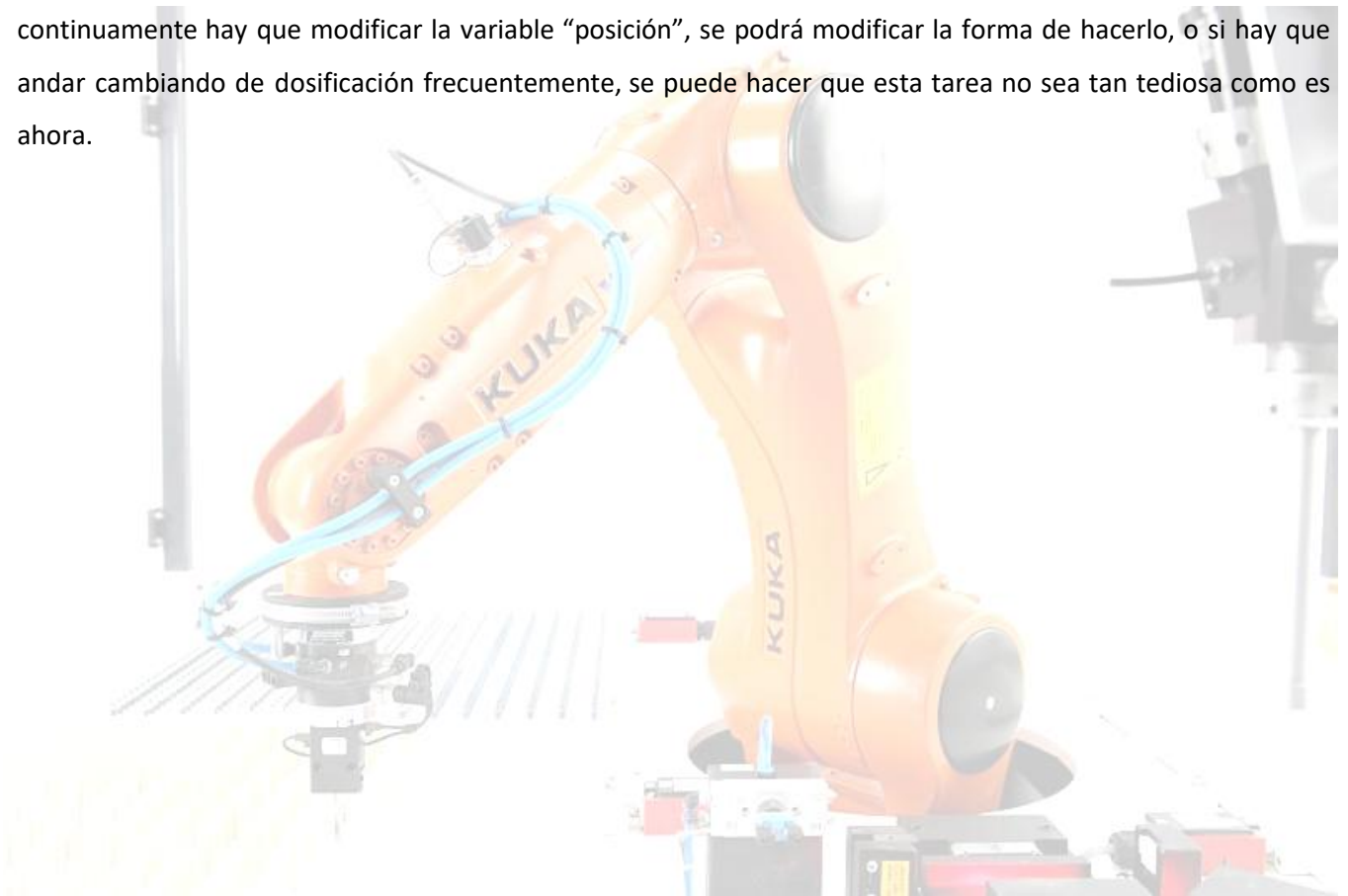


Con las flechas derecha e izquierda nos desplazamos a la casilla donde pone “VEL=”, y ahí le introducimos por teclado la velocidad, teniendo en cuenta que si es un movimiento LIN o CIR, el dato se lo daremos en m/s, (la máxima es 2m/s), si es un PTP el valor se lo daremos en % (el máximo es 100, es el más rápido). Para validar la instrucción una vez introducido el nuevo valor de velocidad, pulsamos a la tecla Softkey “Instrucción OK”.

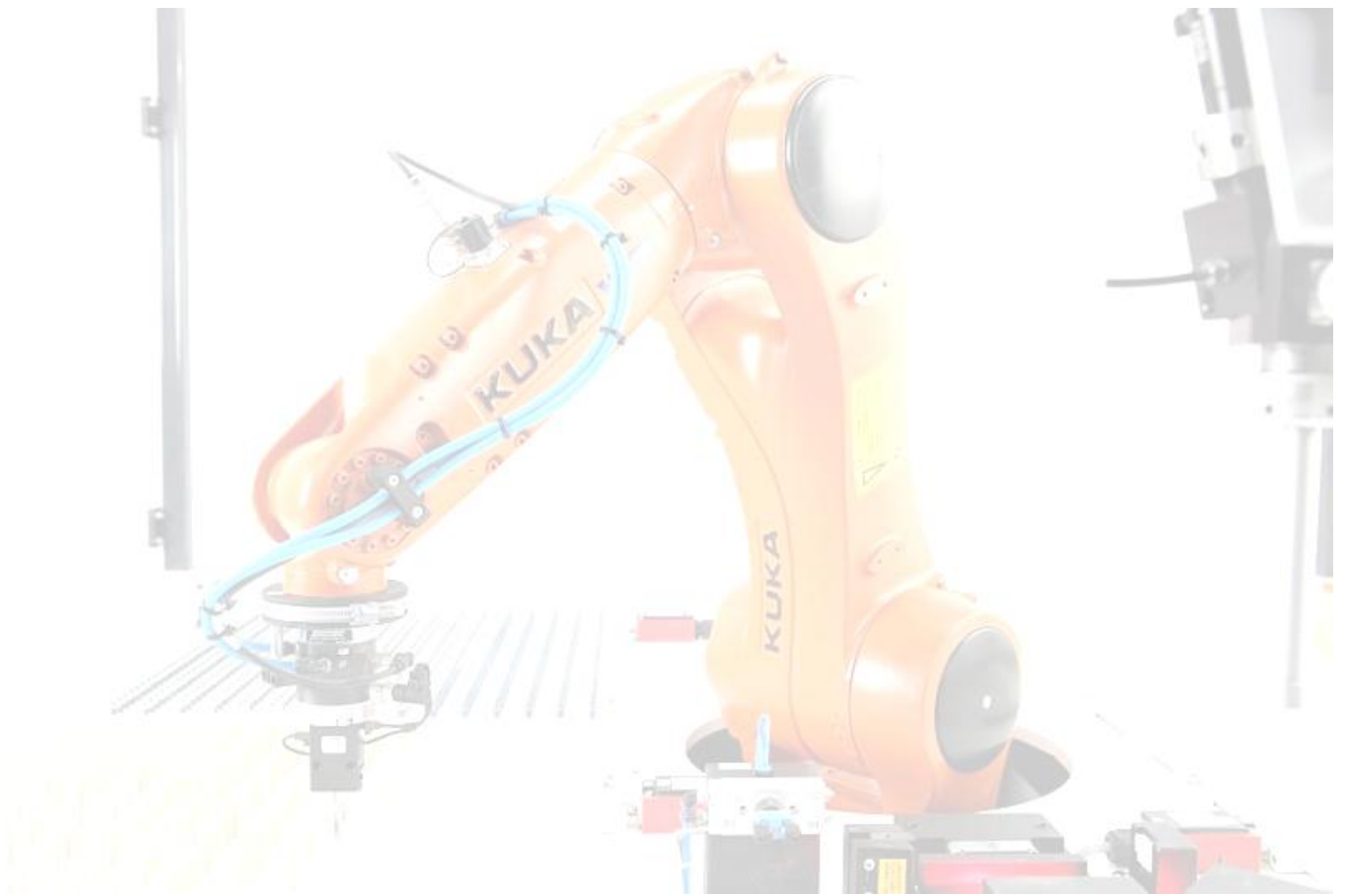
8. APUNTE FINAL.

Se ha intentado que de ninguna forma haya peligro para el operario, si alguien observa alguna inseguridad o tiene alguna idea sobre cómo mejorar las seguridades, por favor que lo diga.

El manejo del robot hasta donde se ha explicado es sencillo, pero hace falta un poco de práctica para coger confianza y entender todos los pegos o problemas que puedan surgir. Todo lo que se ha explicado es cómo funciona el sistema robot y cómo este interfiere con el resto de elementos actualmente, y para un rotor genérico. Según vayamos conociendo nuestra aplicación, se podrá ir viendo qué cosas conviene modificar para evitar y hacer más sencillas todo tipo de operaciones peligrosas o repetitivas. Por ejemplo, si vemos que continuamente hay que modificar la variable "posición", se podrá modificar la forma de hacerlo, o si hay que andar cambiando de dosificación frecuentemente, se puede hacer que esta tarea no sea tan tediosa como es ahora.



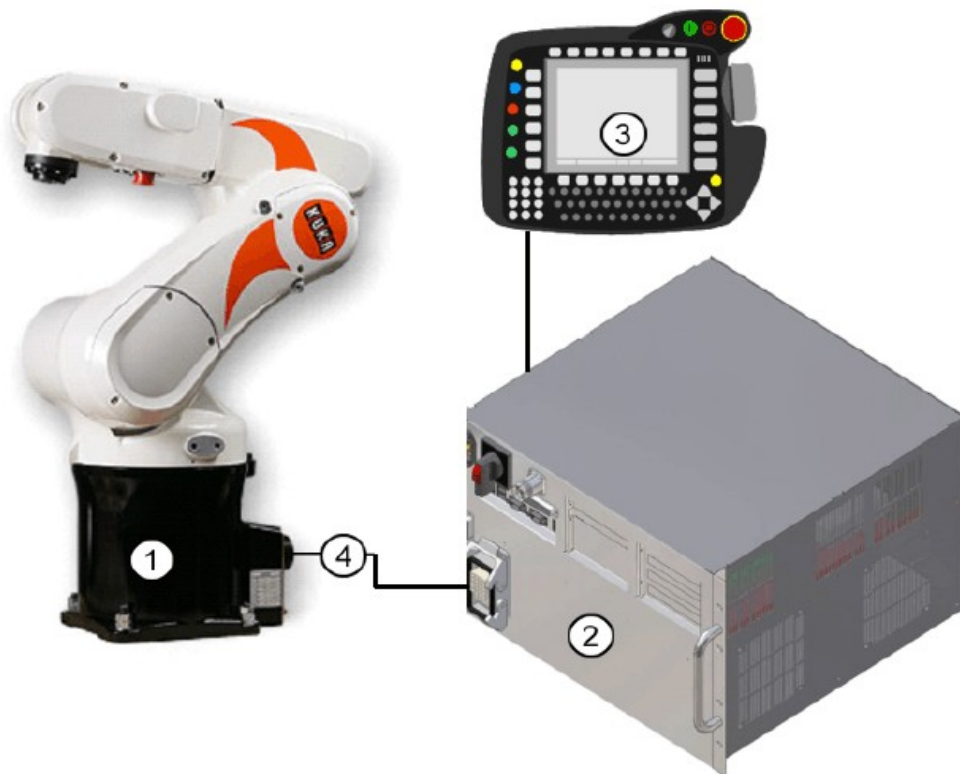
9. EJERCICIOS PRÁCTICOS PLANTEADOS.



1.- PRESENTACIÓN SISTEMA DE ROBOT KUKA

1.- COMPONENTES

Los componentes del sistema de robot instalado en el aula son:



- | | | | |
|---|-----------------------------|---|-------------------------------------|
| 1 | Robot | 3 | KCP (Unidad manual de programación) |
| 2 | Unidad de control del robot | 4 | Cables de unión |

2.- CAMPO DE TRABAJO

El robot está limitado en sus movimientos eje por eje, es decir, existen zonas a las que no puede acceder porque mecánicamente es imposible (tope mecánico).

A su vez se define por software, eje por eje, el ángulo máximo que puede alcanzar el robot en un sentido y en el otro, de modo que se evita en todo momento que el mismo alcance el tope mecánico (finales de carrera software).

Ejemplo para el eje 1:

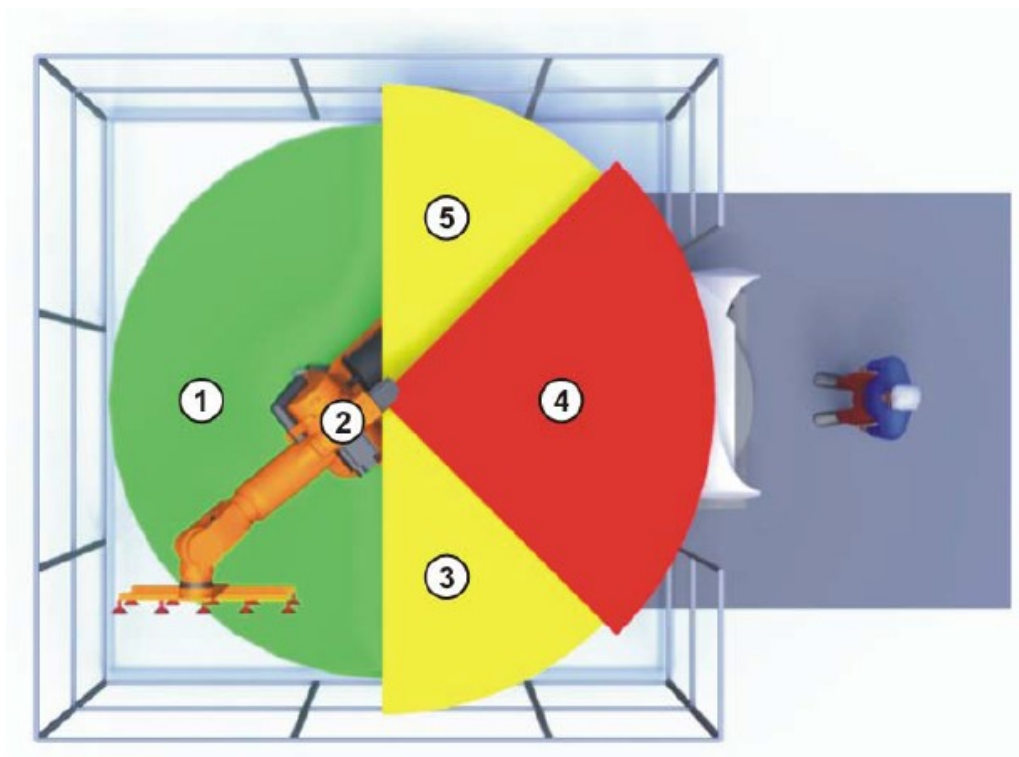


Fig. 3-1: Ejemplo zonas del eje 1

- | | | | |
|---|--------------------|---|--------------------|
| 1 | Campo de trabajo | 4 | Zona de protección |
| 2 | Robot | 5 | Carrera de frenado |
| 3 | Carrera de frenado | | |

3.- SEGURIDADES

El sistema dispone, entre otros elementos de seguridad, de un pulsador de parada de emergencia y de pulsador de “hombre muerto”.

Se pueden añadir más elementos de seguridad a través de la interfaz X11.

3.1.- Pulsador Parada de Emergencia



Fig. 3-2: Pulsador de PARADA DE EMERGENCIA en el KCP

1 Pulsador de PARADA DE EMERGENCIA

Una vez que se activa ésta queda enclavada y se emite un mensaje por pantalla indicando tal acción.

Para volver a poner el robot en funcionamiento habrá que,

- 1º) girar y desenclavar
- 2º) confirmar el mensaje

3.2.- Pulsador de hombre muerto

El robot dispone en su parte trasera de tres pulsadores de este tipo.

Este pulsador tiene tres posiciones:

- a) hacia fuera o no pulsado
- b) posición central o medio pulsado
- c) posición de pánico o pulsado a fondo

Para que funcione correctamente habrá que mantener al menos uno de ellos en posición central.

El que existan tres pulsadores de hombre muerto es para ofrecer al operario que maneja la KCP una posición más cómoda.



Fig. 3-3: Pulsadores de hombre muerto en el KCP

1 - 3 Pulsador de hombre muerto

4.- MODOS DE FUNCIONAMIENTO

El robot posee 4 modos de funcionamiento:

- (1) T1
- (2) T2
- (3) Aut.
- (4) Aut. Ext.

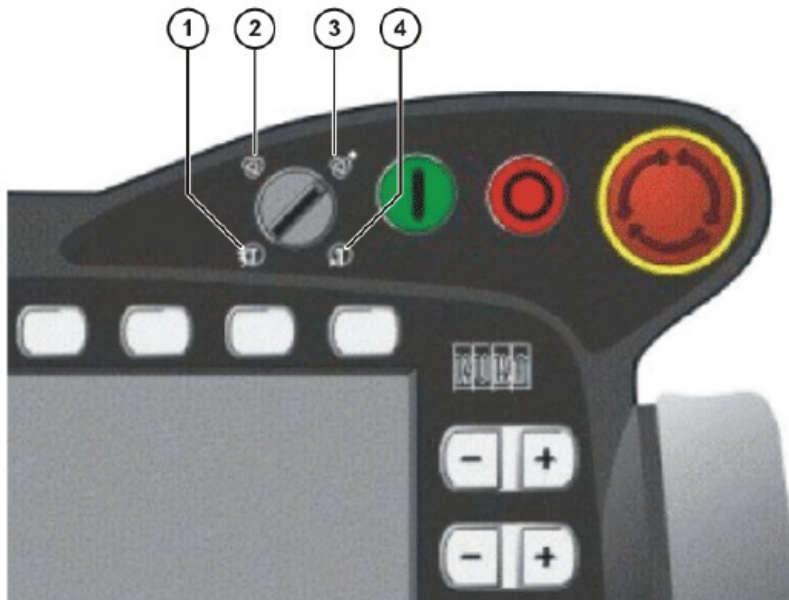


Fig. 3-4: Selector de modos de servicio

- | | | | |
|---|------------------|---|------------------------------|
| 1 | T2 (Test 2) | 3 | AUT EXT (Automático Externo) |
| 2 | AUT (Automático) | 4 | T2 (Test 1) |

El cambio de un modo a otro se hace por medio de una llave. Si se cambia el modo de funcionamiento durante la ejecución del programa el robot se detiene automáticamente.

4.1.- T1 o Test 1

Se pueden realizar movimientos manuales a una velocidad máxima de 250mm/s.

Se puede ejecutar un programa con una velocidad máxima de 250mm/s.

Se debe mantener el pulsador de hombre muerto en posición central.

4.2.- T2 o Test 2

Se pueden realizar movimientos manuales a una velocidad máxima de 250mm/s.

Se puede ejecutar un programa a la velocidad programada.

Se debe mantener el pulsador de hombre muerto en posición central.

4.3.- Aut. o Automático

No se pueden realizar movimientos manuales.

Se puede ejecutar un programa con el circuito de seguridad cerrado.

Se utiliza cuando no existe una unidad de control superior.

El pulsador de hombre muerto carece de función.

4.4.- Aut. Ext. o Automático Externo

No se pueden realizar movimientos manuales.

Se puede ejecutar un programa comandado por una unidad de control superior como un PLC por ejemplo con el circuito de seguridad cerrado.

El pulsador de hombre muerto carece de función.

PRÁCTICA: 1**PRESENTACIÓN DEL ROBOT KUKA****OBJETIVOS:**

Identificar los distintos componentes del robot.
Distinguir los dispositivos de seguridad instalados en la célula.
Conocer los modos de funcionamiento disponibles.

DESCRIPCIÓN:

Se trata de presentar el robot.

INSTRUCCIONES:

1. Identificar los cuatro componentes esenciales del robot (ver figura).
2. Comprobar los dispositivos de seguridad (activación/desactivación).
3. Cambiar los modos de funcionamiento y comprobar los avisos.

DIBUJO:**EJERCICIOS:**

Realizar las acciones indicadas para un primer contacto con el robot.

2.- UNIDAD MANUAL DE PROGRAMACIÓN KCP

La KCP contiene todas las funciones necesarias para el manejo y programación del robot.

1.- LADO FRONTAL



Fig. 4-1: Parte delantera del KCP

1	Selector de modos de servicio	10	Bloque numérico
2	Accionamientos CONECTADOS	11	Softkeys
3	Accionamientos DESCONECTADOS / SSB-GUI	12	Tecla de arranque hacia atrás
4	Pulsador de PARADA DE EMERGENCIA	13	Tecla de arranque
5	Space Mouse	14	Tecla de STOP
6	Teclas de estado derecha	15	Tecla de selección de ventana
7	Tecla de entrada	16	Tecla ESC
8	Teclas del cursor	17	Teclas de estado izquierda
9	Teclado	18	Teclas de menú

2.- TECLADO

- **NUM:** Se conmuta entre la función Números y las funciones de mando del bloque numérico. La barra de estados muestra cual de las dos funciones se encuentra activa.
- **ALT:** Se utiliza en combinación con otra tecla. La tecla tiene función de autoretención pulsándola una vez. No es necesario, por tanto, mantenerla pulsada.
- **SHIFT:** Se conmuta entre la escritura mayúscula y la minúscula. La tecla tiene función de autoretención pulsándola una vez. No es necesario entonces mantenerla pulsada para escribir 1 caracter en mayúscula. Para poder escribir varios caracteres en mayúscula, debe mantenerse pulsada la tecla de SHIFT. Con SIM+SHIFT se conmuta a escritura mayúscula continua.
- **SYM:** Debe ser pulsada para poder utilizar la segunda asignación a las teclas de letras por ej. el caracter "#" sobre la tecla "A". La tecla tiene función de autoretención pulsándola una vez. No es necesario, por tanto, mantenerla pulsada.



Fig. 4-2: Teclado KCP

3.- BLOQUE NUMÉRICO

Para conmutar entre la función numérica y la función de mando debe pulsarse la tecla NUM.



Fig. 4-3: Bloque numérico del KCP

Tecla	Función de mando
INS (0)	Conmuta entre el modo de inserción y el de sobreescritura.
DEL (.)	Borra el caracter colocado a la derecha del cursor.
<-	Borra el caracter colocado a la izquierda del cursor.
END (1)	Coloca el cursor al final de la línea en la cual se encuentra.
CTRL (2)	Se utiliza en combinaciones con otras teclas.
PG DN (3)	Pasa hojas de la pantalla en dirección del fin del archivo.
(4)	----
UNDO (5)	Anula la última entrada realizada. (Esta función no es soportada por el momento.)

Tecla	Función de mando
TAB (6)	Coloca el foco o el curso sobre el próximo elemento de la superficie. Observación: Si un elemento no puede ser alcanzado con la tecla TAB, en lugar de ella, utilizar las teclas del cursor.
HOME (7)	Coloca el cursor al comienzo de la línea en la cual se encuentra.
LDEL (8)	Borra la línea en la que se encuentra el cursor.
PG DN (9)	Pasa hojas de la pantalla en dirección del comienzo del archivo.

4.- PARTE TRASERA



Fig. 4-4: Parte trasera del KCP

- | | | | |
|---|---------------------------|---|---------------------------|
| 1 | Placa característica | 4 | Pulsador de hombre muerto |
| 2 | Tecla de arranque | 5 | Pulsador de hombre muerto |
| 3 | Pulsador de hombre muerto | | |

PRÁCTICA: 2

KCP (Kuka Control Panel)

OBJETIVOS:

Conocer e identificar los distintos grupos de teclas de la KCP.

DIBUJO:



DESCRIPCIÓN:

Se trata de presentar la KCP.

EJERCICIOS:

Realizar las acciones indicadas para un primer contacto con la KCP.

INSTRUCCIONES:

1. Identificar teclas de accionamiento.
2. Identificar teclas de estado derecha/izquierda.
3. Identificar teclas de cursor.
4. Identificar teclado.
5. Identificar bloque numérico.
6. Identificar softkeys.
7. identificar teclas con función fija (arranque y parada de programa, selección de ventana, etc).

3.- SUPERFICIE DE OPERACIÓN HMI

El KCP (KUKA Control Panel) es la unidad manual de programación del sistema del robot. Contiene todas las funciones necesarias para el manejo y programación del robot.



Fig. 5-1: Superficie de operación para teclas de estado, teclas de menús, softkeys

- | | |
|--------------------------------------------|------------------------------------------|
| 1 Teclas de estado a la izquierda | 5 Teclas de estado a la derecha |
| 2 Teclas de estado a la izquierda (Iconos) | 6 Teclas de estado a la derecha (Iconos) |
| 3 Teclas de menú | 7 Softkeys |
| 4 Teclas de menú (Iconos) | 8 Softkeys (Iconos) |

1.- VENTANA EN LA SUPERFICIE DE OPERACIÓN

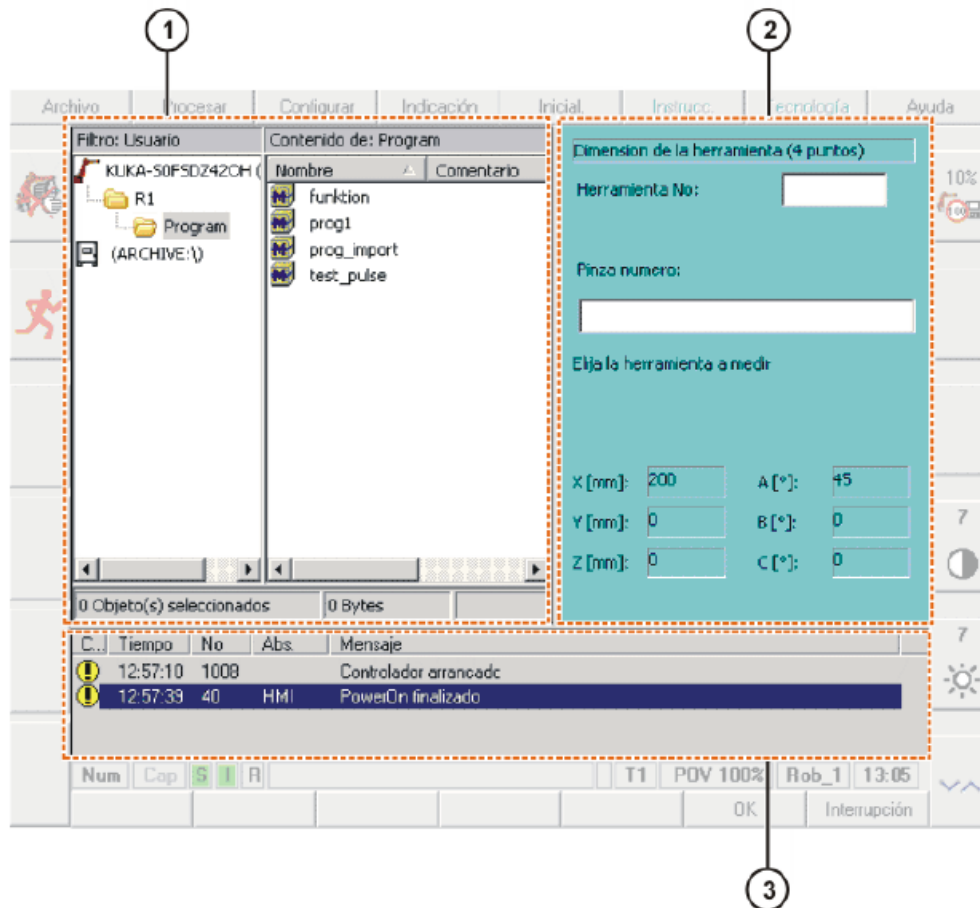


Fig. 5-2: Superficie de operación ventana

- 1 Ventana principal
- 2 Ventana opcional
- 3 Ventana de mensajes

- **VENTANA PRINCIPAL:** muestra el navegador o el programa seleccionado o abierto.
- **VENTANA OPCIONAL:** se asignan a funciones individuales o ejecución de trabajos. No se visualizan permanentemente en la superficie de operación.
- **VENTANA DE MENSAJES:** La ventana de mensajes muestra mensajes de fallo, del sistema y de diálogo.

Sólo se presentan en pantalla 3 ventanas como máximo. Con la tecla de selección de ventanas se conmuta entre ventanas. La ventana seleccionada se marca con color de fondo azul.

2.- BARRA DE ESTADOS

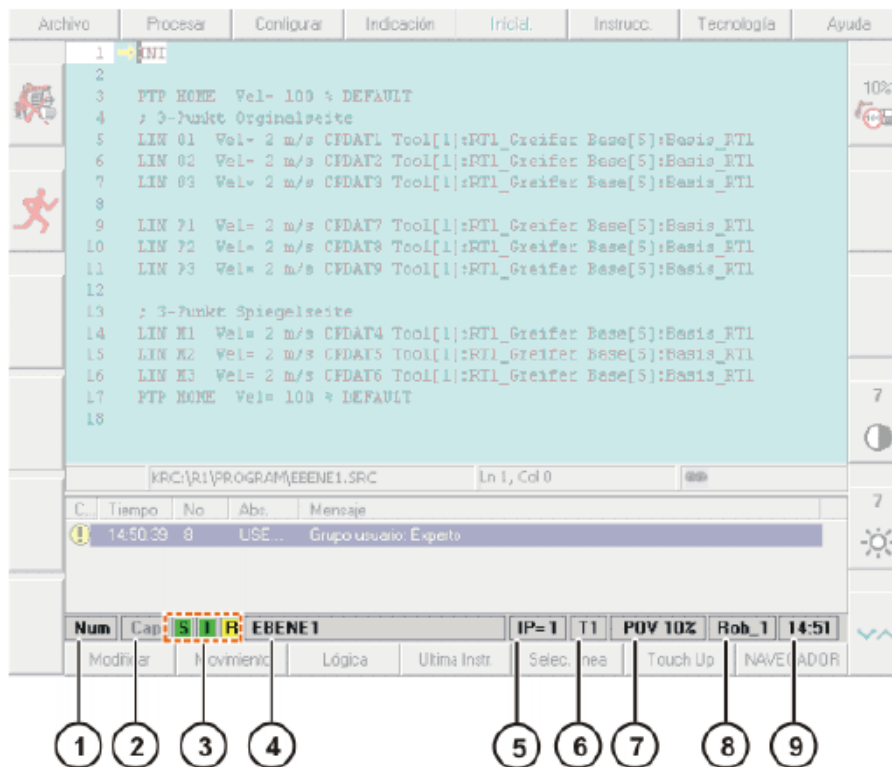












Fig. 5-9: Superficie de operación de la barra de estados

- 1 Estado del bloque numérico
- 2 Estado de escritura minúsculas/mayúsculas
- 3 **S**: Estado del interpretador Submit
- E/S**: Estado de los accionamientos
- R**: Estado del programa
- 4 Nombre del programa seleccionado
- 5 Número del paso actual
- 6 Modo de servicio actual
- 7 Override actual
- 8 Nombre del robot
- 9 Horario del sistema

Símbolo	Descripción
	Está activada la función de números en el bloque numérico
	Está activada la función de mandos en el bloque numérico

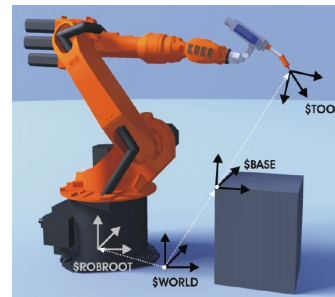
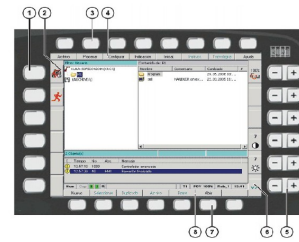
Símbolo	Descripción
	Las mayúsculas están activadas.
	Las minúsculas están activadas.

Símbolo	Color	Descripción
	gris	El interpretador Submit ha sido desactivado.
	rojo	El interpretador Submit ha sido detenido.
	verde	Interpretador SUBMIT está en ejecución.
	verde	Accionamientos preparados
	rojo	Accionamientos no preparados
	gris	No se encuentra seleccionado ningún programa.
	amarillo	El puntero de paso está sobre la primera línea del programa seleccionado.
	verde	El programa ha sido seleccionado y se encuentra en ejecución.
	rojo	El programa seleccionado y arrancado ha sido detenido.
	negro	El puntero de paso está sobre la última línea del programa seleccionado.

PRÁCTICAS: 3**SUPERFICIE DE OPERACIÓN HMI****OBJETIVOS:**

Profundizar en el manejo de la KCP.

Mover manualmente el robot.

DIBUJO:**DESCRIPCIÓN:**

Se trata de manejar la KCP interactuando con los distintos botones, menús asociados, etc.

EJERCICIOS:

Realizar las operaciones descritas en la ventana instrucciones.

INSTRUCCIONES:

1. Seleccionar las ventanas abiertas (principal, opcional y de mensajes).
2. Leer la información de la barra de estado.
3. Identificar los sistemas de coordenadas world (mundo), robroot (local), base (plano de trabajo) y tool (herramienta).

4.- SISTEMAS DE COORDENADAS

En el sistema de robot se definen cuatro sistemas de coordenadas:

- WORLD (mundo)
- ROBROOT (base del robot)
- BASE (plano de trabajo)
- TOOL (herramienta)

WORLD : El sistema de coordenadas WORLD (sistema de coordenadas universales) es un sistema de coordenadas cartesianas de definición fija. Es el sistema genérico de coordenadas para los sistemas de coordenadas BASE y ROBROOT. Por defecto, el sistema de coordenadas WORLD se encuentra en el pie del robot.

ROBROOT: El sistema de coordenadas ROBROOT es un sistema de coordenadas cartesianas que siempre se encuentra en el pie del robot. Describe la posición del robot en relación al sistema de coordenadas WORLD. Por defecto, el sistema de coordenadas ROBROOT se cubre con el sistema de coordenadas WORLD. Con \$ROBROOT puede definirse un corrimiento del robot respecto al sistema de coordenadas WORLD.

BASE: El sistema de coordenadas BASE es un sistema de coordenadas cartesianas que describe la posición de la herramienta. Hace referencia al sistema de coordenadas WORLD. Por defecto, el sistema de coordenadas BASE se cubre con el sistema de coordenadas WORLD. Es desplazado por el usuario hacia la herramienta.

TOOL: El sistema de coordenadas TOOL es un sistema de coordenadas cartesianas cuyo punto de trabajo se encuentra en la herramienta. Hace referencia al sistema de coordenadas BASE. Por defecto, el origen del sistema de coordenadas TOOL se encuentra en el centro de la brida. (Se denomina entonces sistema de coordenadas FLANGE.) El sistema de coordenadas TOOL es desplazado por el usuario en el punto de trabajo de la herramienta.

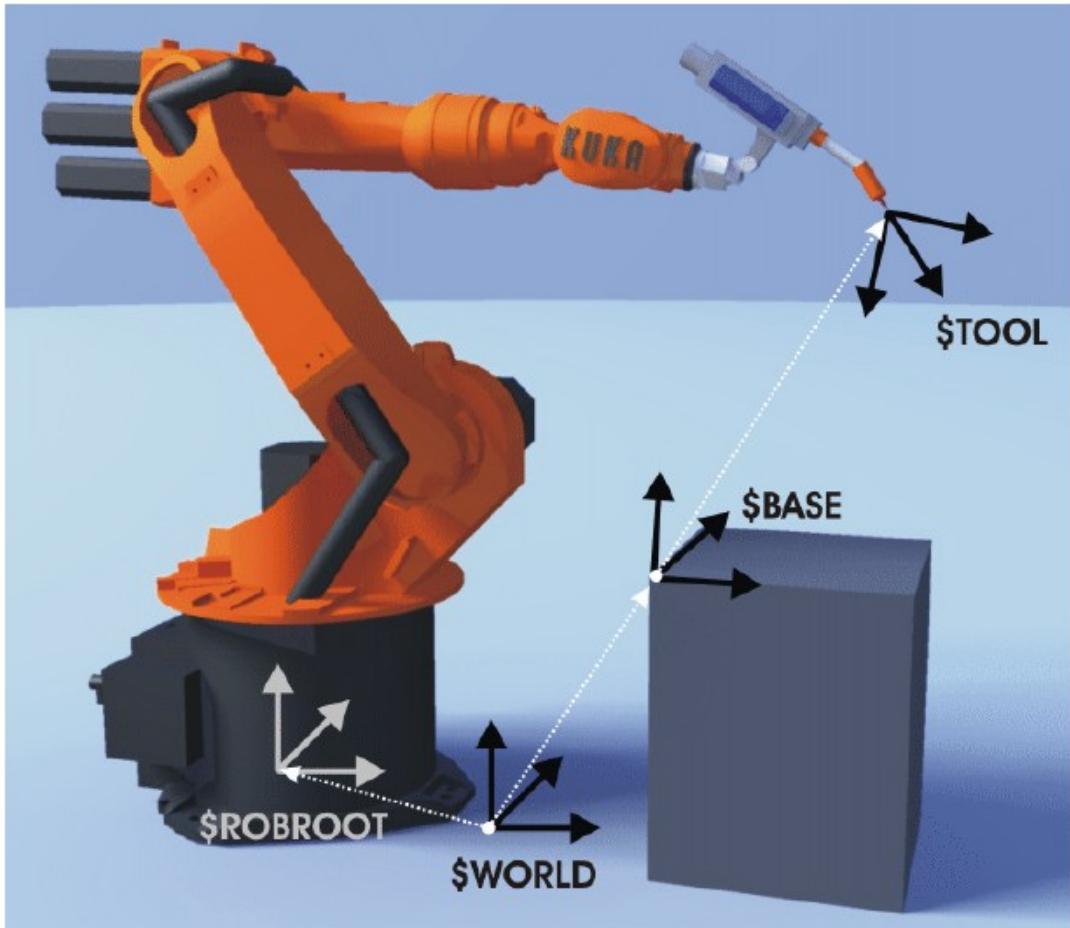


Fig. 6-2: Vista - Resumen de los sistemas de coordenadas

1.- MEDICIÓN DE LA HERRAMIENTA

En la medición de la herramienta el usuario asigna a una herramienta montada en la brida de acople del robot, un sistema de coordenadas cartesianas (sistema de coordenadas TOOL).

El sistema de coordenadas TOOL tiene su origen en un punto definido por el usuario. Este se denomina TCP (Tool Center Point). Por regla general, el TCP es colocado en el punto de trabajo de la herramienta.

Se memorizan los siguientes datos:

- X, Y, Z Origen del sistema de coordenadas TOOL, referido al sistema de coordenadas FLANGE
- A, B, C Orientación del sistema de coordenadas TOOL, referido al sistema de coordenadas FLANGE

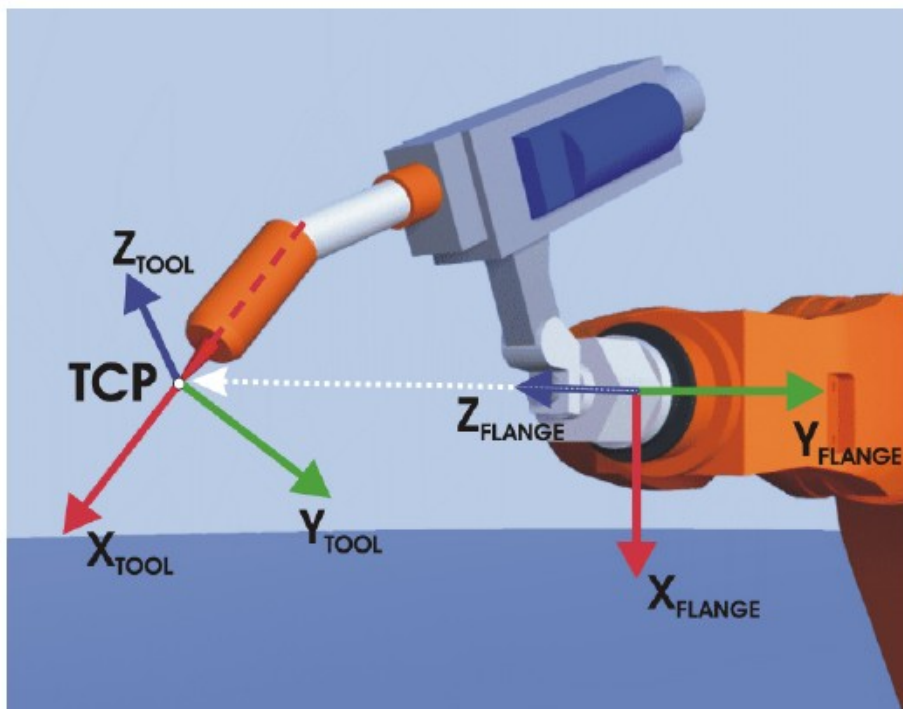


Fig. 7-1: Principio de la medición TCP

1.1.- Medición por el método XYZ-4puntos

Con el TCP de la herramienta a medir debe desplazarse el robot a un punto de referencia desde 4 direcciones diferentes. El punto de referencia puede ser cualquiera. La unidad de control del robot calcula el TCP a partir de las distintas posiciones de la brida.

Para ejecutar este método debe estar montada la herramienta en el robot y además estar seleccionados los modos de operación T1 y T2.

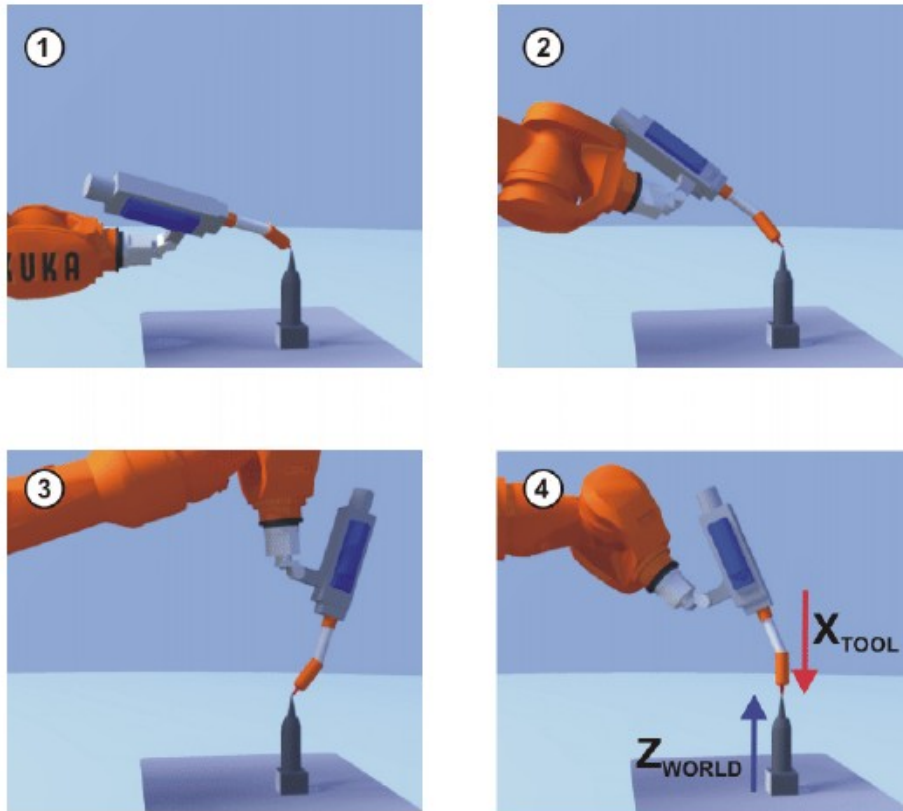


Fig. 7-2: Método XYZ 4-Puntos

1. Seleccionar secuencia de menú **Inicial.** > **Medición** > **Herramienta** > **XYZ 4-Puntos.**
2. Declarar un número y un nombre para la herramienta a medir. Confirmar con **OK.**
3. Con el TCP desplazarse a un punto de referencia. Confirmar con **OK.**
4. Con el TCP desplazarse al punto de referencia desde otra dirección. Confirmar con **OK.**
5. Repetir el paso 4 dos veces.
6. Pulsar **Guardar.**

1.2.- Medición por el método XYZ Referencia

Se efectúa la medición de una nueva herramienta con referencia a una herramienta ya conocida. La unidad de control del robot compara las posiciones de la brida y calcula el TCP de la nueva herramienta.

Teniendo una herramienta ya medida montada sobre la brida y seleccionados los modos de servicio T1 o T2:

Determinar los datos del TCP de la herramienta medida:

1. Seleccionar secuencia de menú **Inicial.** > **Medición** > **Herramienta** > **XYZ Referencia.**
 2. Declarar el número de la herramienta medida.
 3. Anotar los valores de X, Y y Z.
 4. Cerrar la ventana con **Cancelar.**
-
1. Seleccionar secuencia de menú **Inicial.** > **Medición** > **Herramienta** > **XYZ Referencia.**
 2. Declarar un número y un nombre para la nueva herramienta a medir. Confirmar con **OK.**
 3. Declarar los datos del TCP de la herramienta ya medida. Confirmar con **OK.**
 4. Con el TCP desplazarse a un punto de referencia. Confirmar con **OK.**
 5. Liberar la herramienta de esa posición y desmontarla. Montar la nueva herramienta.
 6. Desplazar el TCP de la nueva herramienta al punto de referencia. Confirmar con **OK.**
 7. Pulsar **Guardar.**



Fig. 7-3: Método XYZ - Referencia

1.3.- Entrada numérica

Los valores X, Y, Z y A, B, C referidos al FLANGE se declaran manualmente bien porque se conocen si previamente la herramienta ha sido

diseñada por un software en 3D o bien porque se mide directamente sobre dicha herramienta.

1. Seleccionar secuencia de menú **Inicial**. > **Medición** > **Herramienta** > **Entrada numérica**.
2. Declarar un número y un nombre para la herramienta a medir. Confirmar con **OK**.
3. Dar entrada a los datos. Confirmar con **OK**.
4. Pulsar **Guardar**.

2.- MEDICIÓN DE LA BASE

En la medición de la Base el operario asigna una superficie de trabajo o una herramienta a un sistema de coordenadas cartesianas (sistema de coordenadas Base).

Ventajas de la medición BASE:

- El TCP puede moverse de forma manual a lo largo de los cantos de la superficie de trabajo o de la pieza.
- Puntos pueden ser programados por aprendizaje en relación a la Base. Si la Base debe ser desplazada, por ej. si la superficie de trabajo fue desplazada, se desplazan también los puntos y no tienen que ser programados nuevamente.

2.1.- Medición de los 3 puntos

Con el robot en modo T1 o T2 y con una herramienta ya medida acoplada en la brida de montaje:

1. Seleccionar secuencia de menú **Inicial**. > **Medición** > **Base** > **3-Puntos**.
2. Declarar un número y un nombre para la nueva Base a medir. Confirmar con **OK**.
3. Declarar el número de herramienta montada. Confirmar con **OK**.
4. Con el TCP mover el robot a la nueva Base. Confirmar con **OK**.
5. Con el TCP desplazar el robot a un punto del eje X positivo de la nueva Base. Confirmar con **OK**.
6. Con el TCP desplazar el robot a un punto del plano XY con valor Y positivo Confirmar con **OK**.
7. Pulsar **Guardar**.

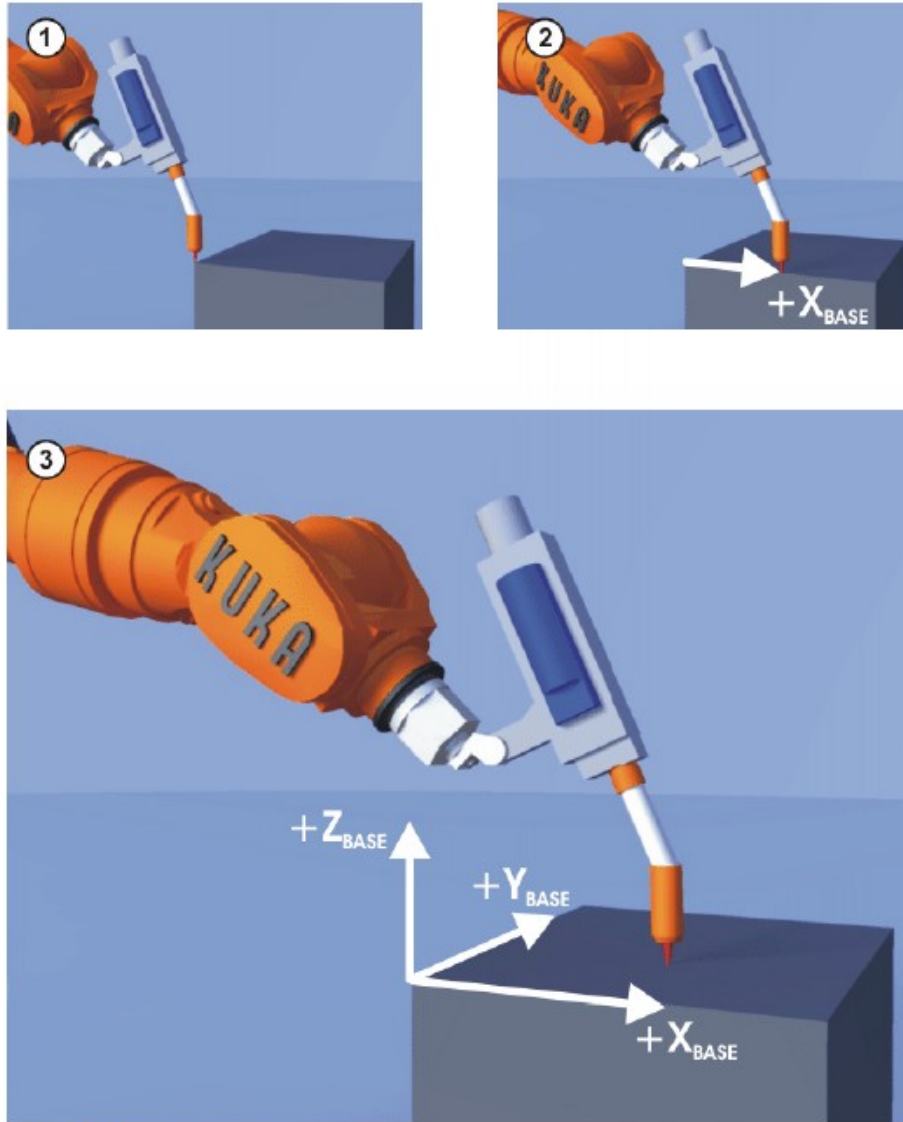


Fig. 7-6: Método de los 3 puntos

2.2.- Método indirecto

El método indirecto es utilizado cuando no es posible llegar con el robot al origen de la base, por ej. porque se encuentra en el interior de una pieza o fuera del campo de trabajo del robot.

Debe efectuarse el desplazamiento a 4 puntos de la Base, cuyas coordenadas sean conocidas. La unidad de control del robot calcula la Base utilizando estos puntos.

Con una herramienta medida montada sobre la brida de acople, los modos T1 o T2 seleccionados y conociendo las coordenadas de 4 puntos accesibles con el TCP de la nueva Base, por ej. por CAD.

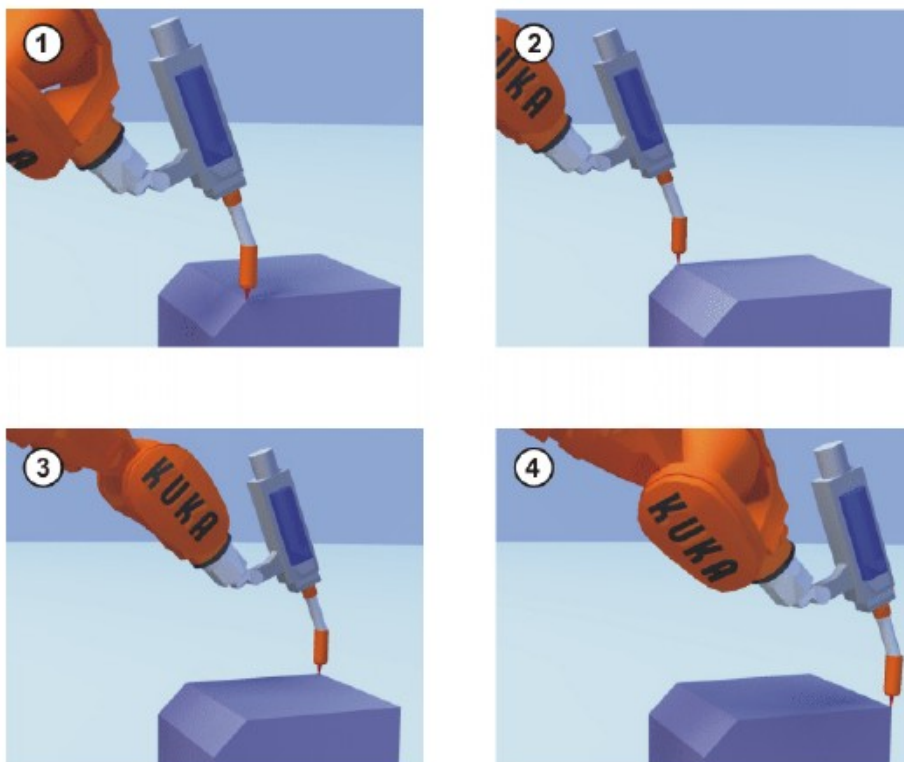


Fig. 7-7: Método indirecto

2.3.- Entrada numérica

Se conocen los siguientes valores numéricos, por ej. por CAD: distancia del origen de la Base al origen del sistema de coordenadas WORLD y giro de los ejes de Base, referido al sistema de coordenadas WORLD.

1. Seleccionar secuencia de menú **Inicial.** > **Medición** > **Base** > **Entrada numérica.**
2. Declarar un número y un nombre para la nueva Base a medir. Confirmar con **OK.**
3. Dar entrada a los datos. Confirmar con **OK.**
4. Pulsar **Guardar.**

PRÁCTICAS: 4A**MEDICIÓN DE LA HERRAMIENTA****OBJETIVOS:**

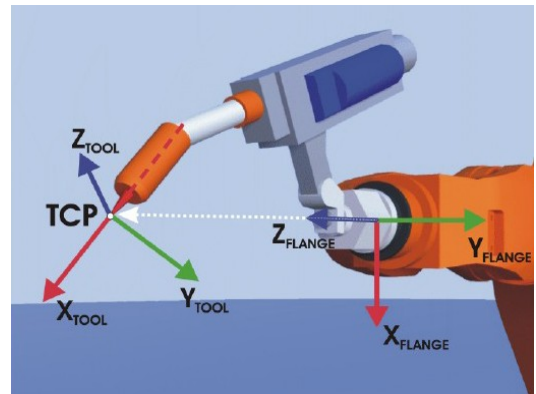
Definir una herramienta.

DESCRIPCIÓN:

Se trata de definir la herramienta de trabajo del robot con el fin de poder programar posteriormente las trayectorias de la tarea.

INSTRUCCIONES:

1. Seleccionar el menú Inicial> Medición> Herramienta> Entrada numérica
2. Definir un nº y nombre de herramienta.
3. Introducir las coordenadas X, Y, Z, A, B, C manualmente.
4. Confirmar y guardar.

DIBUJO:**EJERCICIOS:**

Colocar un bolígrafo o rotulador en el accesorio para montar la herramienta.

Medir las distancias y las orientaciones respecto al Tool Center Point del sistema de coordenadas flange.

Aplicar el método de medición de la herramienta por entrada numérica.

PRÁCTICAS: 4B**MEDICIÓN DE LA BASE****OBJETIVOS:**

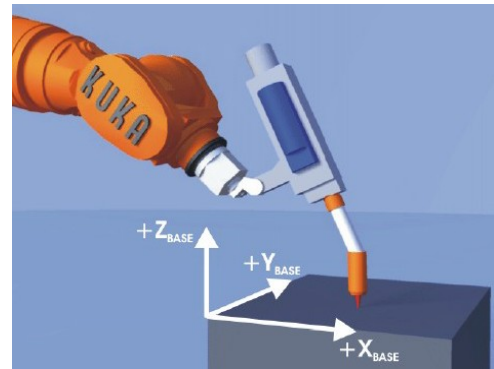
Definir una base.

DESCRIPCIÓN:

Se trata de definir la superficie de trabajo del robot con el fin de poder programar posteriormente las trayectorias de la tarea y los puntos de paso referenciados a ella.

INSTRUCCIONES:

1. Seleccionar el menú Inicial> Medición> Base> 3-puntos.
2. Definir un nº y nombre de base.
3. Introducir el nº de herramienta montada.
4. Llevar el robot a los tres puntos de definición (origen, eje X y plano XY).
5. Confirmar y guardar

DIBUJO:**EJERCICIOS:**

Colocar una mesa frente al robot.

Realizar los movimientos manuales necesarios para definir la base según el método de los 3 puntos.

5.- DESPLAZAMIENTO MANUAL DEL ROBOT

Existen dos formas de desplazar el robot de forma manual:

- **Desplazamiento cartesiano:** El TCP es desplazado en dirección positiva o negativa a lo largo de los ejes de un sistema de coordenadas.
- **Desplazamiento específico de los ejes:** Cada eje puede ser movido en forma individual, en dirección positiva y negativa.

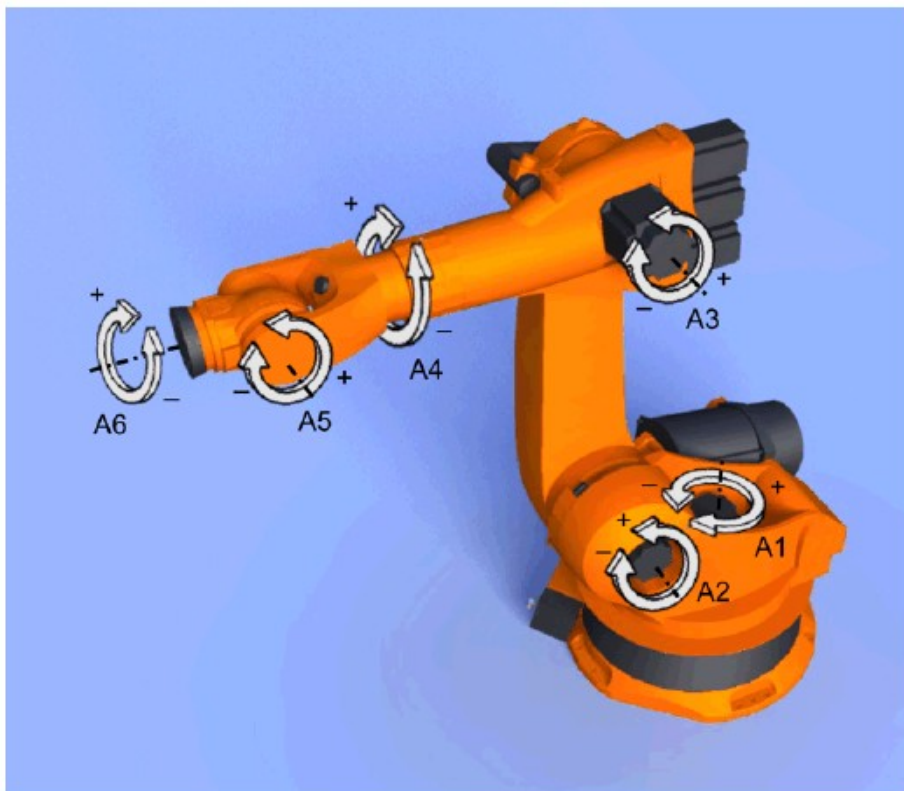


Fig. 6-3: Desplazamiento específico del eje

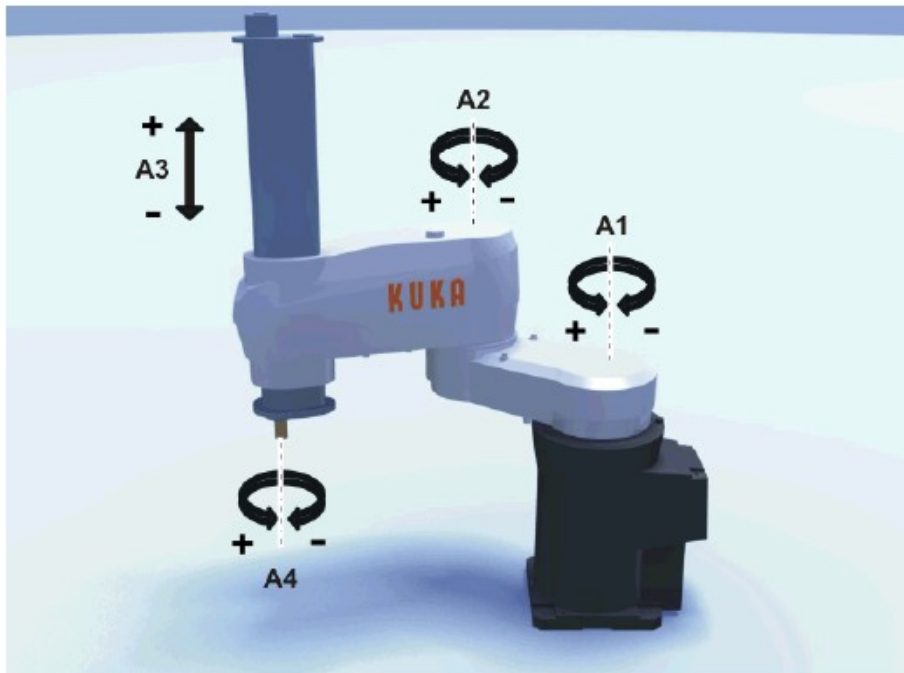


Fig. 4-1: Ejes del robot

1.- PROCEDIMIENTO

Existen dos elementos para mover el robot: el “Space Mouse” en la KCP y las teclas de desplazamiento de la superficie de operación HMI.

1º) En HMI seleccionar el tipo de movimiento:



2º) Seleccionar el porcentaje de override:



El override manual es la velocidad del robot en movimientos manuales. Se indica en porcentaje teniendo en cuenta que la velocidad máxima en este tipo de movimiento es 250mm/s.

Seleccionar la secuencia de menús **Configurar > Movimiento manual > Pasos de override en manual on/off.**

Activo	Significado
No	El override puede ser modificado en pasos de 1%.
Sí	Escalonamiento 100%, 75%, 50%, 30%, 10%, 3%, 1%

3º) Hay que seleccionar un sistema de coordenadas BASE y un sistema de coordenadas TOOL

1. Seleccionar la secuencia de menús **Configuración > Actual herramienta/base.**
2. En la barra de softkeys seleccionar si se ha de trabajar con una herramienta fija:
 - **Herr. extern.:** La herramienta es una herramienta fija.
 - **Herram.:** La herramienta se encuentra montada sobre la brida de acople.
3. En el campo **Herram. num.** declarar el número de herramienta deseado.
4. En el campo **Base num.** declarar el número de la base deseada.
5. Pulsar **OK.**

4º) Para un desplazamiento manual por ejes seleccionar el modo de servicio T1 o T2:

1. En la barra de estado izquierda seleccionar el tipo de movimiento "Teclas de desplazamiento":



2. En la barra de estado derecha seleccionar el desplazamiento específico del eje:



3. Ajustar el override manual.
4. Mantener apretado el pulsador de hombre muerto.
5. En la barra de estado derecha se indican los ejes 1 hasta 6.
Pulsar la tecla de estado positiva o negativa para mover un eje en dirección positiva o negativa, respectivamente.

Para un desplazamiento manual cartesiano seleccionar modo de servicio T1 y T2:

1. En la barra de estado izquierda, seleccionar el tipo de movimiento "Teclas de desplazamiento":



2. En la barra de estado derecha, seleccionar el sistema de coordenadas.
3. Ajustar el override manual.
4. Mantener apretado el pulsador de hombre muerto.
5. En la barra de estado derecha se visualizan las siguientes teclas de estado:

X, Y, Z: Para los movimientos lineales a lo largo de los ejes del sistema de coordenadas seleccionado

A, B, C: Para los movimientos de rotación alrededor de los ejes del sistema de coordenadas seleccionado

Pulsar la tecla de estado positiva o negativa para mover un eje en dirección positiva o negativa, respectivamente.






Es posible visualizar la posición del robot durante el desplazamiento del mismo. Seleccionar la secuencia de menú **Indicación > Posición actual**.

2.- MOVIMIENTO MANUAL INCREMENTAL

Se puede conseguir desplazamientos fijos del robot definiendo un intervalo en milímetros o grados.

El desplazamiento en milímetros se utiliza para desplazamiento por eje cartesiano; y el desplazamiento en grados se utiliza para la rotación sobre los ejes cartesianos (A, B, C) o para el desplazamiento por ejes.

Se puede aplicar cuando se quiera posicionar puntos a distancias iguales o cuando se quiera definir una separación de una posición por ejemplo en caso de fallo.

Tecla de estado	Descripción
	Movimiento manual incremental desconectado
	Incremento = 100 mm o 10°
	Incremento = 10 mm o 3°
	Incremento = 1 mm o 1°
	Incremento = 0,1 mm o 0,005°

3.- MOSTRAR POSICIÓN ACTUAL

Seleccionar secuencia de menús **Indicación > Posición Actual > Específico de ejes o Cartesiano.**

Nombre	Valor	Unidad
Herramienta/Base		
Tool_1 (1)	#BASE	Tool
Base_1 (1)	#BASE	Base
Posición		
X	1809,98	mm
Y	-10,00	mm
Z	1900,11	mm
Orientación		
A	0,00	gra
B	90,00	gra
C	0,00	gra
Posición de robot		
S	010	bin
T	000010	bin

Eje	Pos. [grad, mm]	Motor [deg]
A1	0,00	0,00
A2	-90,00	17136,18
A3	90,00	17135,36
A4	0,00	0,00
A5	0,00	0,00
A6	0,00	0,00

Fig. 6-8: Posición actual cartesiana y específica del eje

PRÁCTICAS: 5**MOVIMIENTOS MANUALES****OBJETIVOS:**

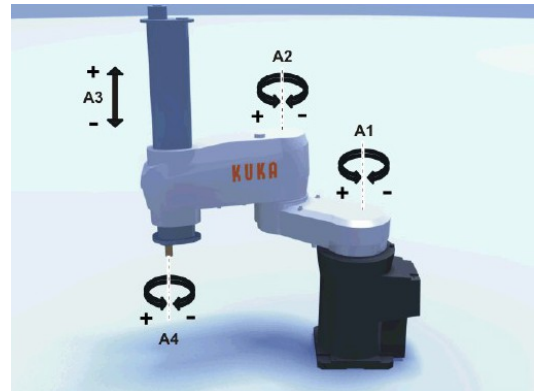
Conocer las distintas formas de movimiento del robot.

DESCRIPCIÓN:

Se trata de mover el robot linealmente (plano XYZ) y por ejes independientes.

INSTRUCCIONES:

1. Seleccionar movimiento manual por teclas de desplazamiento.
2. Seleccionar el porcentaje de "Override" manual (limitado a 250mm/s). Configurar los pasos 1%, 3%, 5%,...
3. Seleccionar la base y la herramienta.
4. Seleccionar el tipo de movimiento: cartesiano o por ejes.
5. Seleccionar el movimiento manual incremental (mm para cartesiano y ° para ejes). Probarlo.
6. Consultar la posición actual (Indicación> Posición actual> Cartesiano o Específico de ejes).

DIBUJO:**EJERCICIOS:**

Mover el robot manualmente con las teclas de desplazamiento.

Probar a variar la velocidad.

Mover el robot a incrementos.

Consultar la posición del robot.

6.- PROGRAMACIÓN DE MOVIMIENTOS

1.- NAVEGADOR

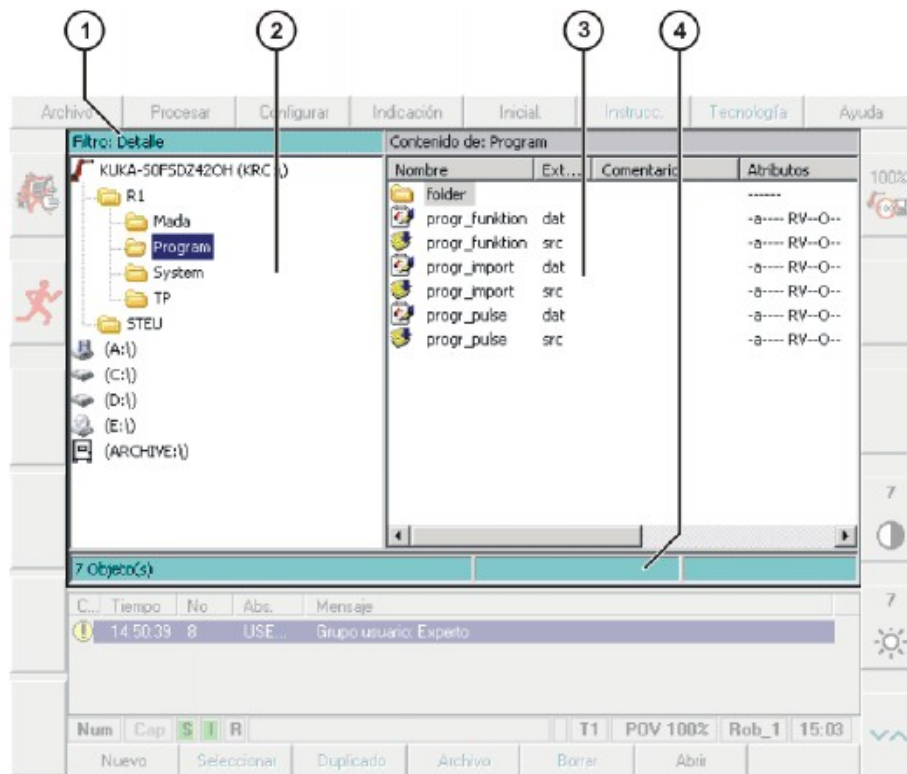


Fig. 8-1: Navigator (Navegador)

- | | |
|-----------------------------|---------------------|
| 1 Encabezamiento | 3 Lista de ficheros |
| 2 Estructura de directorios | 4 Línea de estados |

En el Navegador, el usuario administra programas y todos los ficheros específicos del sistema.

1.1.- Tipos de archivos

Un programa se puede ver como un módulo:



o como un conjunto de dos archivos .src y .dat.



Para verlos de una forma o de otra habrá que seleccionar “Archivo” => “Filtro” y escoger el filtro deseado (como condición previa la selección del grupo de usuario experto).

1.2.- Selección de un programa

1. Marcar el programa en el Navegador.
Si el programa es mostrado como fichero SRC o DAT, puede marcarse el fichero correspondiente.
2. Pulsar el softkey **Seleccionar**.
3. Ejecutar o modificar el programa.
4. Seleccionar secuencia de menú **Procesar > Cancelar programa**.

2.- ESTRUCTURA DE UN PROGRAMA

Un programa está escrito en lenguaje KRL (Kuka Robot Language).

Está estructurado de la siguiente forma:

```
1 DEF my_program( )
2 INI
3
4 PTP HOME Vel= 100 % DEFAULT
...
8 LIN point_5 CONT Vel= 2 m/s CPDAT1 Tool[3] Base[4]
...
14 PTP point_1 CONT Vel= 100 % PDAT1 Tool[3] Base[4]
...
20 PTP HOME Vel= 100 % DEFAULT
21
22 END
```

La línea DEF muestra el nombre del programa.

Si el programa es una función, la línea DEF comienza con "DEFFCT" y contiene otras indicaciones.

La línea INI contiene inicializaciones para variables internas y parámetros. ¡La línea INI no debe ser borrada!

La línea END es la última línea en cada programa. Si el programa es una función, la línea END se escribe "ENDFCT". ¡La línea END no debe ser borrada!

2.1.- Home Position

La HOME POSITION es una posición válida por encima de los límites del programa. Por regla general, se la utiliza como primera y última posición en el programa, por su definición unívoca y no crítica.

2.2.- Programar movimiento PTP

El robot desplaza el TCP al punto de destino a lo largo de la trayectoria más rápida. La trayectoria más rápida no es, en regla general, la trayectoria más corta y por ello no es una recta. Dado que los ejes del robot se mueven de forma rotacional, trayectorias curvas pueden ser ejecutadas de forma más rápida que las rectas.

No puede predecirse la trayectoria exacta.

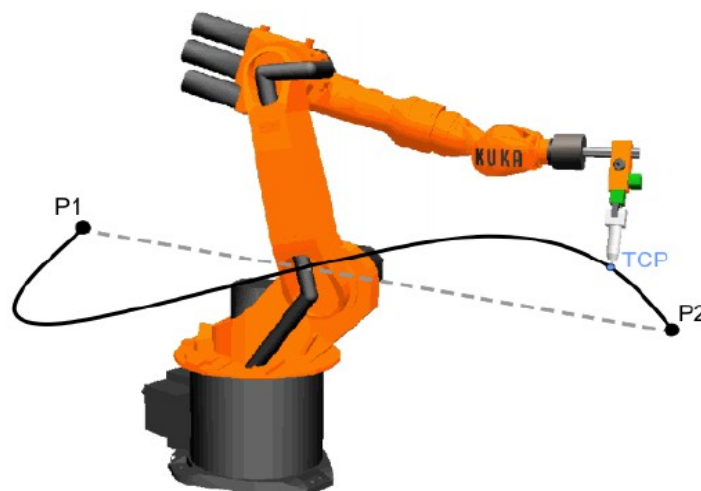


Fig. 8-3: Movimiento PTP

La instrucción PTP comprende la definición de parámetros como el punto de destino ,la velocidad de desplazamiento y si debe alcanzar el punto de forma exacta o aproximada (CONT).



El punto inicial es siempre el punto de destino de la instrucción anterior o cuando el robot no se mueve es la posición actual.

Para programar “por aprendizaje” realizar los siguientes pasos:

1. Mover el TCP a la posición la cual se programará como punto de destino.
2. Colocar el cursor en la línea **detrás** de la cual se insertará la instrucción de movimiento.
3. Seleccionar la secuencia de menú **Instrucc. > Movimiento > PTP**.
4. Declarar los parámetros en el formulario Inline.
5. Guardar la instrucción con el softkey **Instrucción OK**.

2.3.- Programar movimiento LIN

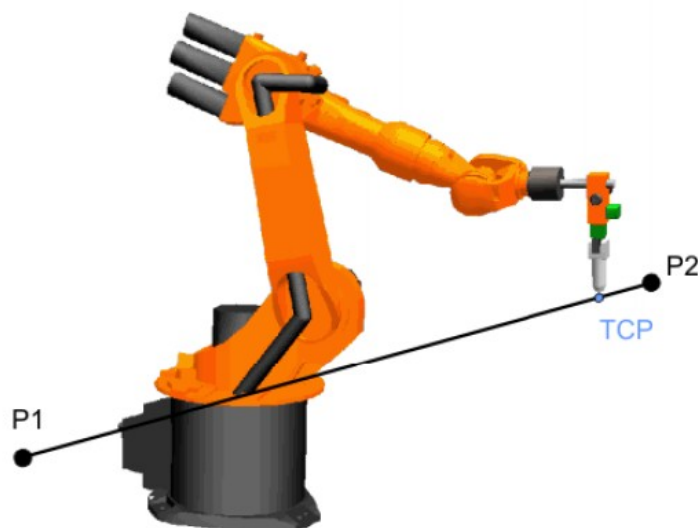


Fig. 8-4: Movimiento LIN

El robot conduce el TCP con una velocidad definida al punto de destino a lo largo de la trayectoria más corta. El trayecto más corto es siempre una recta.

La instrucción LIN comprende los mismos parámetros que la instrucción PTP excepto la velocidad que se da entre 0,001-2m/s.



2.4.- Programar movimiento CIRC

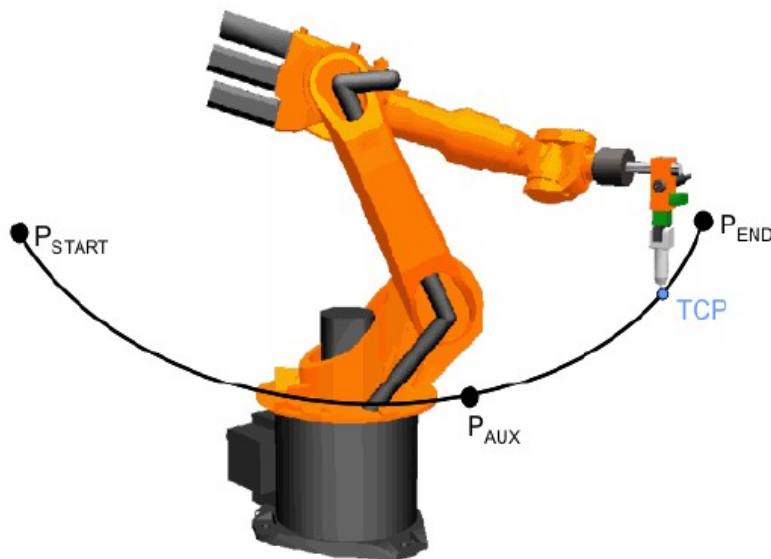
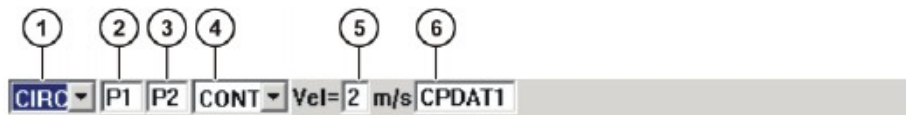


Fig. 8-5: Movimiento CIRC

El robot conduce el TCP con una velocidad definida al punto de destino a lo largo de la trayectoria circular. La trayectoria circular queda definida por el punto de arranque, un punto intermedio y el punto de destino.

En esta instrucción se “deben aprender” dos puntos: el punto de paso o punto intermedio y el punto de destino o punto final.

1. Mover el TCP a la posición la cual se programará como punto intermedio.
2. Colocar el cursor en la línea **detrás** de la cual se insertará la instrucción de movimiento.
3. Seleccionar la secuencia de menú **Instrucc. > Movimiento > CIRC.**
4. Declarar los parámetros en el formulario Inline.
5. Pulsar el softkey **TouchUP PI.**
6. Mover el TCP a la posición la cual se programará como punto de destino.
7. Guardar la instrucción con el softkey **Instrucción OK.**



2.5.- Ventanas de opciones

Es la ventana emergente para configurar las opciones de desplazamiento en cuanto a herramienta, base, velocidad, etc.

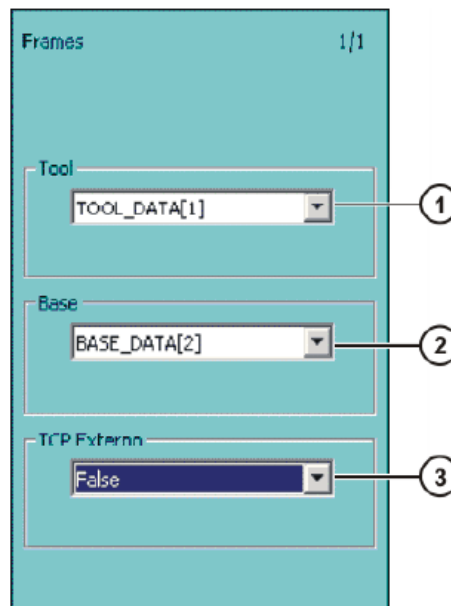


Fig. 8-9: Ventana de opciones Frames

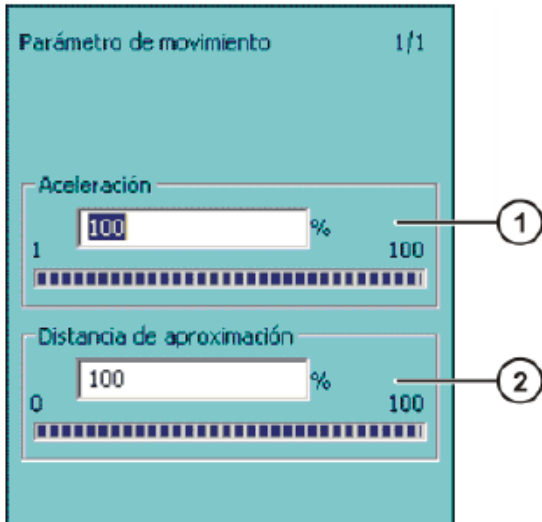


Fig. 8-10: Ventana de opciones Parámetros de movimiento (movimiento PTP)

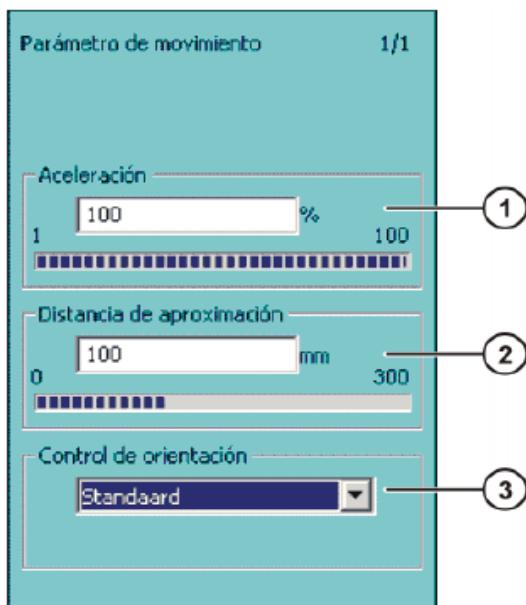


Fig. 8-11: Ventana de opciones Parámetros de movimiento (movimiento CP)

Con las aproximaciones a los puntos dejamos definida un área de acercamiento al punto destino pero no llegamos hasta él.

De esa forma nos ahorramos las aceleraciones y deceleraciones y por tanto tiempo de ciclo.

Es importante tener claro la aplicación. Cuando es necesario llegar exactamente al punto y cuando podemos simplemente bordearlo.

En los movimiento PTP, la aproximación se define como un porcentaje siendo el 100% el punto central de la trayectoria entre el punto origen y el punto destino.

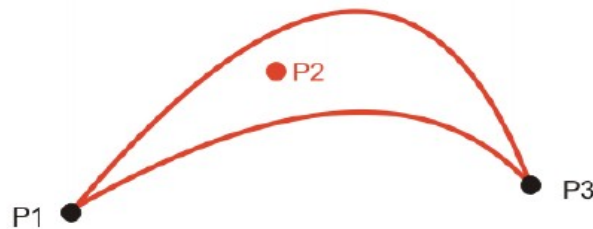


Fig. 8-12: Movimiento PTP, P2 tuvo un posicionamiento aproximado

En los movimientos LIN y CIRC el máximo es de 300mm para trayectorias con puntos origen y destino más largas; y la mitad de la distancia para trayectorias más cortas.

Es decir, si la distancia entre P1 (origen) y P2 (destino) es de 400mm, podemos definir un contorno de aproximación de hasta 300mm. Si la distancia entre dichos puntos es por ejemplo de 150mm, el contorno máximo que podemos definir es de 75mm.

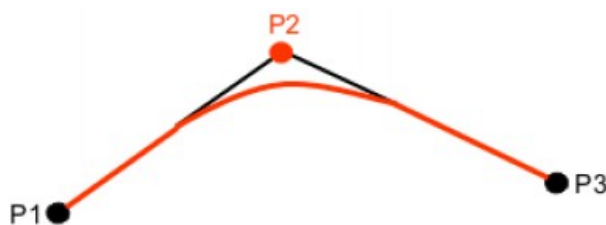


Fig. 8-13: Movimiento LIN, P2 tuvo un posicionamiento aproximado

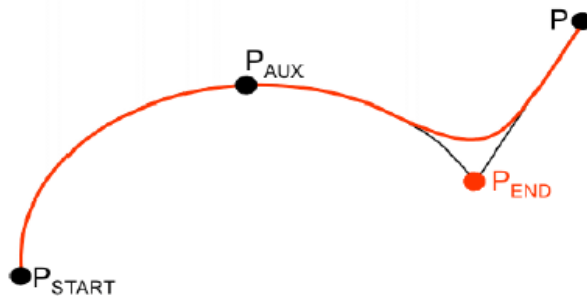


Fig. 8-14: Movimiento CIRC, P_{END} tuvo posicionamiento aproximado

En el caso de movimiento circular, el punto intermedio es siempre exacto y la distancia de contorno se refiere siempre al punto final.

En todos los casos, el arco que describe el robot al hacer en contorno no es circular.

Si no se selecciona CONT, es decir, se deja el campo vacío, el robot alcanza el punto programado.

2.6.- Modificar punto programado por aprendizaje

El procedimiento para reprogramar una posición ya programada en una instrucción de movimiento:

1. Llevar el robot con el TCP a la posición deseada.
2. Colocar el cursor en la línea de la instrucción la cual se ha de modificar.
3. Pulsar el softkey **Modificar**. Se abre el formulario inline para la instrucción.
4. Para movimientos PTP y LIN Pulsar el softkey **Touch Up** para asumir la posición actual del TCP como nuevo punto de destino.
Para movimientos CIRC:
 - Pulsar el softkey **Touch PI** para asumir la posición actual del TCP como nuevo punto intermedio.
 - Pulsar el softkey **Touch PF** para asumir la posición actual del TCP como nuevo punto de destino.
5. Confirmar la pregunta de seguridad con **Si**.
6. Guardar la modificación con el softkey **Instrucción OK**.

2.7.- Control de la orientación

Cuando se definen los puntos de una trayectoria no sólo se coge información de la posición con respecto a los ejes X, Y y Z. El punto programado almacena la orientación de la herramienta con respecto a dichos ejes, es decir, cómo queremos que llegue a dicho punto.

Al desplazarse el robot de un punto a otro ya sea en movimiento LIN, PTP o CIRC podemos elegir mantener la orientación de la herramienta constante o que la calcule el propio robot.

Con control de orientación “Standard”, el robot se dirige al punto final con la orientación que él mismo calcula pero llega con la programada en dicho punto; con el control “Constante”, el robot mantiene desde el inicio la orientación de la herramienta y no tiene en cuenta la orientación grabada en el punto final.

En los dos casos es posible que surjan singularidades o movimientos no permitidos para el robot. Para intentar salvar este problema podemos configurar PTP Manual como orientación aunque el robot se puede apartar levemente de la trayectoria (hay aplicaciones en las que esto no se puede permitir).

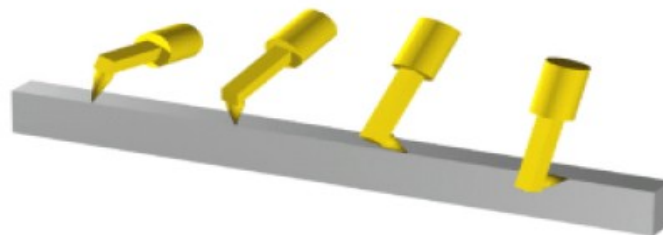


Fig. 8-15: Estándar o PTP manual

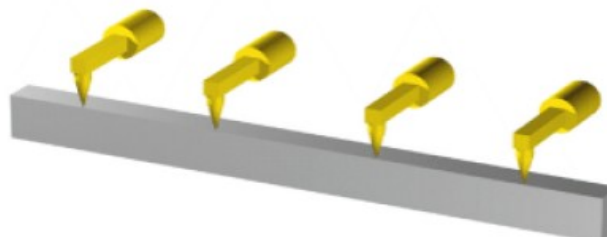






Fig. 8-16: Control orientación constante

3.- ARRANCAR UN PROGRAMA

3.1.- Modos de ejecución

Tecla de estado	Modo de ejecución del programa	Descripción
	GO	El programa se ejecuta sin detención hasta el final del mismo.
	MSTEP (Motion Step)	El programa se ejecuta con una parada después de cada paso de movimiento. La tecla de arranque debe ser pulsada nuevamente después de cada paso de movimiento.
	ISTEP (Incremental Step)	El programa se ejecuta con una parada después de cada línea del programa. También líneas del programa que no sean visibles, y líneas vacías, deben ser consideradas. La tecla de arranque debe ser pulsada nuevamente para cada línea. ISTEP sólo se encuentra disponible para el grupo del experto.
	Movimiento hacia atrás	Este tipo de ejecución del programa es seleccionado automáticamente si se pulsa la tecla Arranque del programa hacia atrás.

3.2.- Ajuste del override del programa (POV)

Es la velocidad del robot durante la ejecución del programa.

Se indica en porcentaje.

En el modo T1 la velocidad es máxima 250mm/s.

Utilizar la tecla siguiente para ajustar la velocidad:



3.3.- Arranque del programa hacia delante (manual)

Con el modo de servicio T1 o T2 y habiendo seleccionado un programa:

- Seleccionar el modo de ejecución del programa
- Mantener apretado el pulsador de hombre muerto hasta que la barra de estado indique



es decir, accionamientos preparados.

- Mantener pulsada la tecla de arranque hasta que en la ventana de mensajes aparezca la leyenda "COI alcanzado" El robot se queda detenido.
- Pulsar y mantener apretada la tecla de arranque.

NOTA: Para detener el programa en cualquier punto sólo habrá que soltar la tecla de arranque.

3.4.- Arranque del programa hacia delante (automático)

Con el modo de servicio Automático y habiendo seleccionado un programa:

- En la barra de estado izquierda seleccionar el tipo de movimiento GO.
- Pulsar la tecla Accionamientos conectados.
- Mantener pulsada la tecla de arranque hasta que en la ventana de mensajes aparezca la leyenda "COI alcanzado". El robot se queda detenido.
- Apretar la tecla de Start. El programa es ejecutado.

NOTA: Para detener un programa arrancado en modo automático, pulsar la tecla STOP.

3.5.- Resetear un programa

Para poder arrancar nuevamente desde el comienzo un programa interrumpido, debe ser reseteado. Elegir desde el menú Procesar => Resetear programa.

PRÁCTICAS: 6**PROGRAMACIÓN DE MOVIMIENTOS****OBJETIVOS:**

Conocer las instrucciones básicas de movimiento del robot.

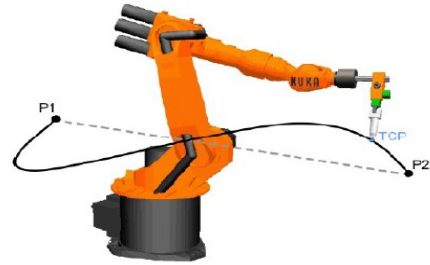
Definir trayectorias por aprendizaje de los puntos de paso.

DESCRIPCIÓN:

Se trata de crear un programa cuyo resultado sea el desplazamiento del robot sobre una trayectoria dada.

INSTRUCCIONES:

1. Crear un nuevo programa llamado "Cuadrado".
2. Seleccionar el modo en el que se quiere ver en el navegador (módulo o archivo .dat y .src).
3. Seleccionar el programa creado y modificar.
4. Identificar la estructura base del programa.
5. Llevar el robot a los puntos HOME y esquinas del cuadrado e introducir las instrucciones predefinidas (PTP, LIN, CIRC).
6. Ejecutar el programa.

DIBUJO:

```

1 DEF my_program( )
2   INI
3   ...
4   PTP HOME Vel= 100 % DEFAULT
5   ...
8   LIN point_5 CONT Vel= 2 m/s CPDAT1 Tool[3] Base[4]
9   ...
14  PTP point_1 CONT Vel= 100 % PDAT1 Tool[3] Base[4]
15  ...
20  PTP HOME Vel= 100 % DEFAULT
21  ...
22  END

```

EJERCICIOS:

Realizar un programa en el que el robot describa un cuadrado.

Realizar un programa en el que el robot escriba una palabra de al menos tres letras enmarcada entre dos líneas (entre letra y letra no debe haber trazo).

7.- INSTRUCCIONES DE ENTRADA, SALIDA Y TEMPORIZADOR

1.- TARJETAS DE E/S

La unidad de control del robot puede controlar hasta 4096 entradas/salidas digitales y hasta 32 analógicas. Esto quiere decir que se pueden añadir tarjetas de E/S a dicha unidad y configurarlas para ello.

Nuestro robot cuenta con 7E y 8S incluidas en su hardware.

	Entradas	Salidas
Digital	\$IN[1] ... \$IN[4096]	\$OUT[1] ... \$OUT[4096]
Analógico	\$ANIN[1] ... \$ANIN[32]	\$ANOUT[1] ... \$ANOUT[32]

2.- INSTRUCCIONES DE ENTRADA Y SALIDA

2.1.- OUT (activar salida)

Con el programa seleccionado y el cursor en la línea que se quiere insertar el programa, ejecutar **Instruc < Lógica < OUT < OUT**, rellenar los parámetros y guardar.



Fig. 8-17: Formulario inline OUT

Pos.	Descripción	Rango de valores
1	Número de la salida	1 ... 4096
2	Si para la salida existe ya un nombre, éste es visualizado. Solo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.	Libremente seleccionable

Pos.	Descripción	Rango de valores
3	Estado, al cual se ha de activar o colocar la salida.	TRUE, FALSE
4	<ul style="list-style-type: none"> ■ CONT: Procesamiento en avance ■ [vacío]: Ejecución con stop de avance 	CONT, [vacío]

2.2.- PULSE (activar salida de pulso)

Con el programa seleccionado y el cursor en la línea que se quiere insertar el programa, ejecutar **Instruc < Lógica < OUT < PULSE**, rellenar los parámetros y guardar.



Fig. 8-18: Formulario inline PULSE

Pos.	Descripción	Rango de valores
1	Número de la salida	1 ... 4096
2	<p>Si para la salida existe ya un nombre, éste es visualizado.</p> <p>Sólo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.</p>	Libremente seleccionable
3	<p>Estado, al cual se ha de activar o colocar la salida.</p> <ul style="list-style-type: none"> ■ TRUE: Nivel "High" ■ FALSE: Nivel "Low" 	TRUE, FALSE
4	<ul style="list-style-type: none"> ■ CONT: Procesamiento en avance ■ [vacío]: Ejecución con stop de avance 	CONT, [vacío]
5	Longitud del impulso	0.1 ... 3

2.3.- WAIT (tiempo de espera)

El movimiento del robot se detiene durante el tiempo programado.

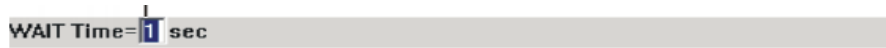


Fig. 8-21: Formulario inline WAIT

Pos.	Descripción	Rango de valores
1	Tiempo de espera	≥ 0 sec

2.4.- WAITFOR (tiempo de espera condicionado)

Con el programa seleccionado y el cursor en la línea que se quiere insertar el programa, ejecutar **Instruc < Lógica < WAITFOR**, rellenar los parámetros y guardar.

Se pueden encadenar varias condiciones utilizando combinaciones lógicas con operaciones AND, OR...

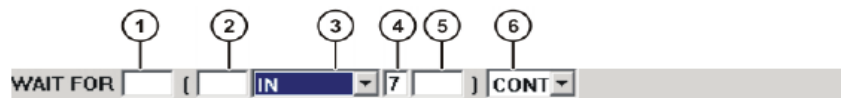


Fig. 8-22: Formulario inline WAITFOR

Pos.	Descripción	Rango de valores
1	<ul style="list-style-type: none"> ■ Agregar la combinación lógica externa. El operador se ubica entre las expresiones colocadas entre paréntesis. ■ Agregar NOT. Agregar el operador deseado o NOT con el Softkey.	AND, OR, EXOR, [vacío] NOT, [vacío]
2	<ul style="list-style-type: none"> ■ Agregar la combinación lógica interna. El operador se ubica dentro de una expresión colocada entre paréntesis. ■ Agregar NOT. Agregar el operador deseado o NOT con el Softkey.	AND, OR, EXOR, [vacío] NOT, [vacío]
3	Señal la cual se está esperando	IN, OUT, CYCFLAG, TIMER, FLAG
4	Número de la señal	1 ... 4096
5	Si para la señal existe ya un nombre, éste es visualizado. Sólo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.	Libremente seleccionable
6	<ul style="list-style-type: none"> ■ CONT: Procesamiento en avance ■ [vacío]: Ejecución con stop de avance 	CONT, [vacío]

2.5.- SYN OUT (conmutar sobre la trayectoria)

Con el programa seleccionado y el cursor en la línea que se quiere insertar el programa, ejecutar **Instruc < Lógica < OUT < SYN OUT**, rellenar los parámetros y guardar.

a) Opción START/END

La conmutación se puede disparar referenciando el punto de arranque o el de destino dentro de un movimiento dado LIN, PTP, CIRC

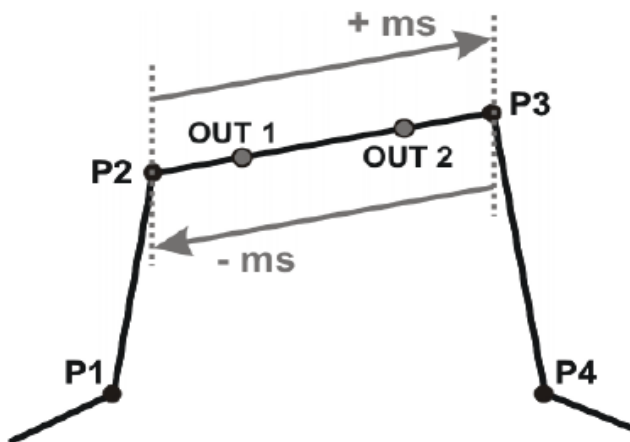


Fig. 8-23: Formulario Inline SYN OUT, opción START/END

Pos.	Descripción	Rango de valores
1	Número de la salida	1 ... 4096
2	Si para la salida existe ya un nombre, éste es visualizado. Sólo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.	Libremente seleccionable
3	Estado, al cual se ha de activar o colocar la salida.	TRUE, FALSE
4	Punto en el lugar de disparo de la función de conmutación <ul style="list-style-type: none"> ■ START: Se conmuta en el punto de arranque del paso de movimiento. ■ END: Se conmuta en el punto de arranque del paso de movimiento. 	START, END Opción PATH: (>>> 8.7.15 "Formulario Inline SYN OUT, opción PATH" página 103)
5	Corrimiento en tiempo del punto de conmutación Observación: La indicación del tiempo es absoluta. El punto de disparo se modifica en función de la velocidad del robot.	-1 000 ... +1 000 ms

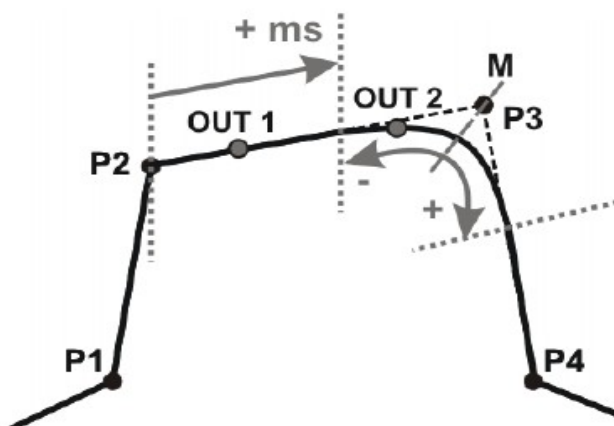
Ejemplo 1: el punto de arranque y el destino son de parada exacta.

```
LIN P1 VEL=0.3m/s CPDAT1  
LIN P2 VEL=0.3m/s CPDAT2  
SYN OUT 1 '' State= TRUE at START Delay=20ms  
SYN OUT 2 '' State= TRUE at END Delay=-20ms  
LIN P3 VEL=0.3m/s CPDAT3  
LIN P4 VEL=0.3m/s CPDAT4
```



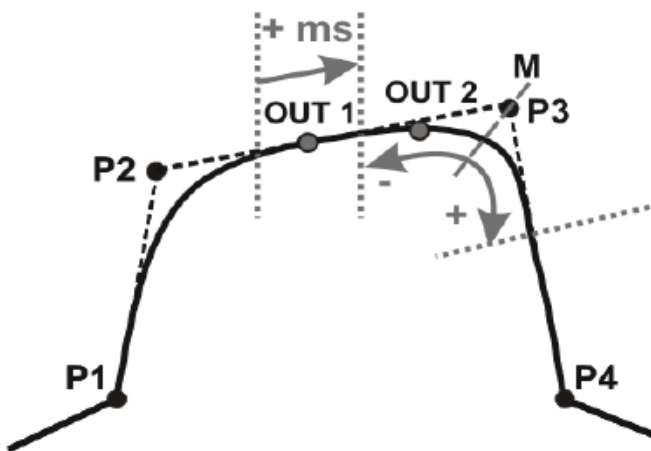
Ejemplo 2: el punto de arranque es de parada exacta y el destino es de posicionamiento aproximado.

```
LIN P1 VEL=0.3m/s CPDAT1  
LIN P2 VEL=0.3m/s CPDAT2  
SYN OUT 1 '' State= TRUE at START Delay=20ms  
SYN OUT 2 '' State= TRUE at END Delay=-20ms  
LIN P3 CONT VEL=0.3m/s CPDAT3  
LIN P4 VEL=0.3m/s CPDAT4
```



Ejemplo 3: los puntos de arranque y destino son de posicionamiento aproximado.

```
LIN P1 VEL=0.3m/s CPDAT1  
LIN P2 CONT VEL=0.3m/s CPDAT2  
SYN OUT 1 '' State= TRUE at START Delay=20ms  
SYN OUT 2 '' State= TRUE at END Delay=-20ms  
LIN P3 CONT VEL=0.3m/s CPDAT3  
LIN P4 VEL=0.3m/s CPDAT4
```



Con las instrucciones sincronizadas SYN OUT podemos jugar con el tiempo y los pasos por puntos aproximados. Nos ahorramos tiempo de ciclo al evitar la desaceleración y aceleración del robot. Los tiempos definidos son absolutos; la velocidad del robot provoca que la distancia entre la activación de la salida y el punto en cuestión sea más grande o más pequeña. El máximo de distancia lo marca las líneas punteadas grises (analizar las gráficas).

b) Opción PATH

La función de conmutación se dispara por distancia al punto de destino y con un retardo o adelanto en tiempo incluido.

Esta opción sólo se puede utilizar para movimientos LIN y CIRC.

Los puntos de conmutación se pueden atrasar hasta un máximo de un punto de parada exacta; y adelantar hasta el punto de origen si es de parada exacta o hasta el comienzo del entorno si es de posicionamiento aproximado.



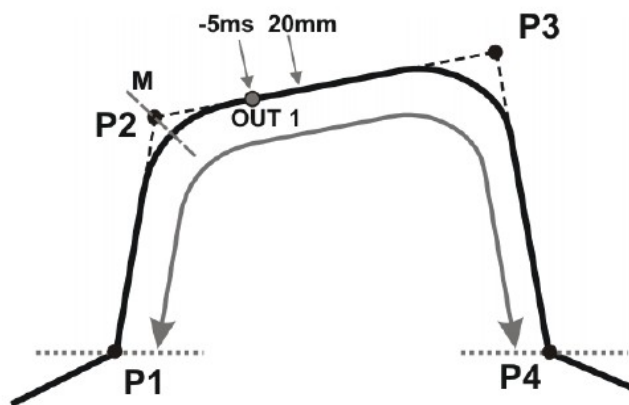
SYN OUT 2 State= TRUE PATH = 0 mm Delay= 5 ms

Fig. 8-24: Formulario Inline SYN OUT, opción PATH

Pos.	Descripción	Rango de valores
1	Número de la salida	1 ... 4096
2	Si para la salida existe ya un nombre, éste es visualizado. Sólo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.	Libremente seleccionable
3	Estado, al cual se ha de activar o colocar la salida.	TRUE, FALSE
4	---	PATH Opcion START o END: (>>> 8.7.14 "Formulario Inline SYN OUT, opción START/END" página 101)
5	Distancia del punto de conmutación desde el punto de destino Este campo sólo es mostrado cuando en Pos. 4 se ha seleccionado PATH .	-2 000 ... +2 000 mm
6	Corrimiento en tiempo del punto de conmutación Observación: La indicación del tiempo es absoluta. El punto de disparo se modifica en función de la velocidad del robot.	-1 000 ... +1 000 ms

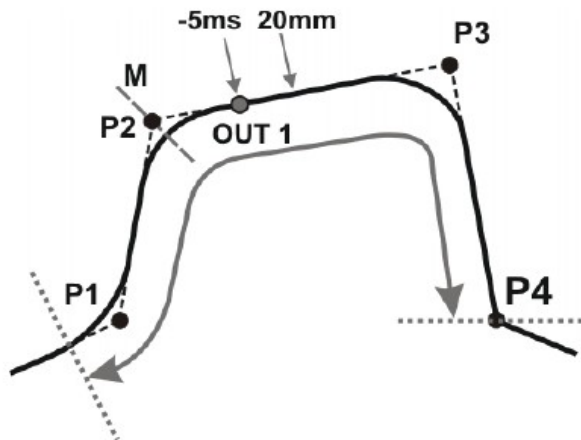
Ejemplo 1: el punto de partida es de parada exacta y el de destino de pso aproximado.

```
LIN P1 VEL=0.3m/s CPDAT1  
SYN OUT 1 '' State= TRUE at START PATH=20mm Delay=-5ms  
LIN P2 CONT VEL=0.3m/s CPDAT2  
LIN P3 CONT VEL=0.3m/s CPDAT3  
LIN P4 VEL=0.3m/s CPDAT4
```



Ejemplo 2: los puntos de partida y destino son de posicionamiento aproximado.

```
LIN P1 CONT VEL=0.3m/s CPDAT1  
SYN OUT 1 '' State= TRUE at START PATH=20mm Delay=-5ms  
LIN P2 CONT VEL=0.3m/s CPDAT2  
LIN P3 CONT VEL=0.3m/s CPDAT3  
LIN P4 VEL=0.3m/s CPDAT4
```



2.6.- SYN PULSE (activar un pulso sobre la trayectoria)

Con el programa seleccionado y el cursor en la línea que se quiere insertar el programa, ejecutar **Instruc < Lógica < OUT < SYN PULSE**, rellenar los parámetros y guardar.

Se puede activar el pulso con referencia a un punto origen o un punto destino. Se puede hacer por distancia y por tiempo.



Fig. 8-25: Formulario inline SYN PULSE

Pos.	Descripción	Rango de valores
1	Número de la salida	1 ... 4096
2	Si para la salida existe ya un nombre, éste es visualizado. Sólo para el grupo del experto: Pulsando la tecla del softkey Textos largos puede declararse un nombre.	Libremente seleccionable
3	Estado, al cual se ha de activar o colocar la salida.	TRUE, FALSE
4	Duración del impulso	0,1 ... 3 s

Pos.	Descripción	Rango de valores
5	<ul style="list-style-type: none"> ■ START: El impulso es disparado en el punto de arranque del paso de movimiento. ■ END: El impulso es disparado en el punto de destino del paso de movimiento. <p>Ejemplos y límites de conmutación ver SYN OUT. (>>> 8.7.14 "Formulario Inline SYN OUT, opción START/END" página 101)</p> <ul style="list-style-type: none"> ■ PATH: El impulso es disparado en el punto de destino del paso de movimiento. <p>Ejemplos y límites de conmutación ver SYN OUT. (>>> 8.7.15 "Formulario Inline SYN OUT, opción PATH" página 103)</p>	START, END, PATH
5	<p>Distancia del punto de conmutación desde el punto de destino</p> <p>Este campo sólo es mostrado cuando en Pos. 4 se ha seleccionado PATH.</p>	-2 000 ... +2 000 mm
6	<p>Corrimiento en tiempo del impulso</p> <p>Observación: La indicación del tiempo es absoluta. El punto de disparo se modifica en función de la velocidad del robot.</p>	-1 000 ... +1 000 ms

3.- CONSULTAR E/S DIGITALES

Seleccionar la secuencia de menús **Indicación > Entradas/Salidas > Salidas digitales o Entradas digitales.**

Se puede simular la activación de la entradas siempre y cuando el robot esté en modo de operación T1 o T2, la entrada esté seleccionada y el pulsador de hombre muerto presionado. Para ello cambiar el valor a TRUE.

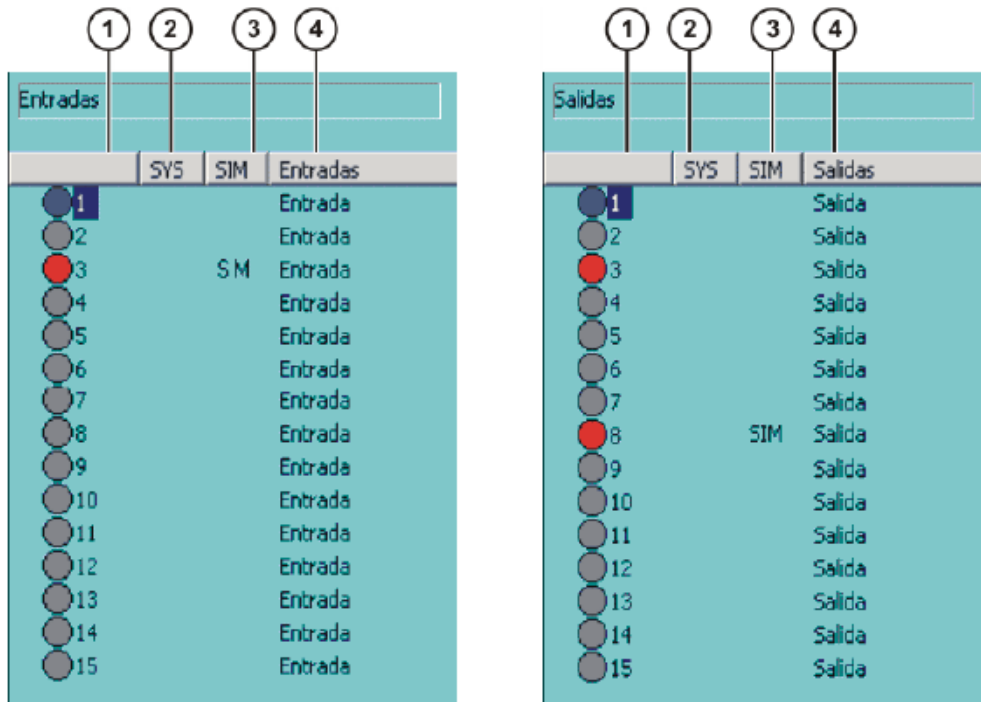


Fig. 6-9: Entradas / salidas digitales

NOTA: Las entradas SYS 1025 y 1026 son P_On y P_Off. No se pueden modificar.

PRÁCTICAS: 7**PROGRAMACIÓN ENTRADAS, SALIDAS Y TEMPORIZADORES****OBJETIVOS:**

Conocer las instrucciones lógicas de entrada, salida y temporizador del robot.

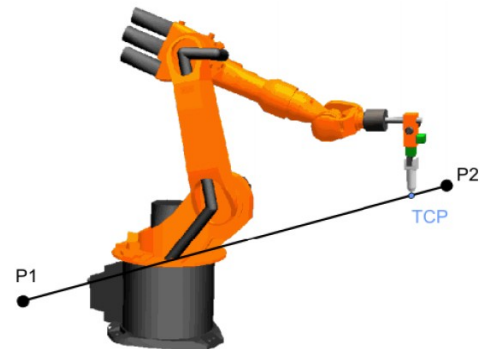
Definir trayectorias con lectura de entradas, activación de salidas y control de tiempos.

DESCRIPCIÓN:

Se trata de crear un programa cuyo resultado sea el desplazamiento del robot sobre una trayectoria dada.

INSTRUCCIONES:

1. Abrir el programa llamado "Cuadrado" de la práctica anterior.
2. Insertar las nuevas instrucciones WAITFOR, WAIT, OUT, OUT SYN, OUT PULSE en los puntos que marca el ejercicio.
3. Ejecutar el programa.
4. Consultar en la KCP el estado de las entradas y las salidas.

DIBUJO:

```

LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SYN OUT 1 '' State= TRUE at START Delay=20ms
SYN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
  
```

EJERCICIOS:

Modificar el programa "Cuadrado" jugando con los puntos de posicionamiento aproximado y de parada exacta.

Desplazar al punto P1, esperar 5s, desplazar al punto P2, esperar a la entrada 1, desplazar al punto P3, activar la salida 1, desplazar a P4, activar la salida 2 sincronizada por tiempo y por distancia.

8.- INSTRUCCIONES DE REPETICIÓN Y CONTROL

1.- SUBROUTINAS

Una rutina no es más que un programa que se puede llamar a ejecución desde otro.

Ejemplo:

- a) creamos el programa “primero” en el que el robot realiza una determinada tarea

```
DEF primero ()
```

```
INI
```

```
PTP HOME
```

```
LIN P1
```

```
LIN P2
```

```
LIN P3
```

```
PTP HOME
```

```
END
```

- b) creamos el programa “principal” en el que se hace una llamada a “primero”

```
DEF principal ()
```

```
INI
```

```
PTP HOME
```

```
primero()
```

```
PTP HOME
```

```
END
```

De esta forma puedo ejecutar el programa “primero” tantas veces como quiera sin tener que repetir las instrucciones incluidas en él.

Otra opción es crear dentro del programa primero una subrutina.

Ejemplo:

a) se crea el programa “primero” y dentro de él una subrutina llamada “segundo”

DEF primero()

INI

PTP HOME

segundo()

PTP HOME

END

DEF segundo()

.....

END

Puedo llamar tantas veces como quiera a la subrutina “segundo” dentro del programa “primero”.

Desde el programa principal no puedo llamar a “segundo” a no ser que sustituya “DEF segundo()” por “global DEF segundo()”.

En cualquier caso hay que ver en función de la aplicación concreta que se esté programando lo que más interesa: crear un archivo con distintas subrutinas o crear archivos diferentes que serán llamados desde un programa en cuestión.

2.- VARIABLES

Anteriormente hemos visto que cada programa creado se compone de un módulo con dos archivos: *.src (conjunto de instrucciones) y *.dat (conjunto de datos).

En la carpeta SYSTEM del navegador podemos ver el archivo CONFIG que no es más que el archivo general de datos del robot. En él se incluye información de los puntos, herramientas, bases, datos de carga, etc. Dicho archivo se compone de varias carpetas entre ellas **USER GLOBALS** o **USER DEFINED VARIABLES** en la que podemos definir variables propias e incluso inicializar a la vez (ejemplo: DECL INT vueltas = 1234). Para abrirlo seguir la secuencia de menú **Procesar < Fold < Abrir**.

En cada archivo, entre las líneas **DEF** e **INI** se pueden declarar las variables locales que pueden ser de varios tipos. Siempre deben inicializarse detrás de la línea INI:

DEF principal ()

```
DECL INT cajas
DECL REAL cota
DECL BOOL estado
DECL CHAR letra
```

INI

```
cajas = 12
cota = 123.35
estado = false
estado = $IN[1]
letra = "d"
```

NOTA: Para poder escribir entre las líneas DEF e INI: 'Configurar' → 'Extras' → 'Editor' → 'Def. Línea'.

Si se tiene una variable con el mismo nombre en el archivo '**config**' (global) y en un programa (local) siempre prevalece la local, es decir, no se modifica la variable global.

Sin embargo, si dentro de un programa llamo a una variable global sin declararla sí que se modifica en el archivo '**config**'.

Se puede vincular una entrada o una salida a un nombre. Es lo que en Kuka se llama **signal**. Se declaran en el archivo '**config**'.

USERS GLOBALS

User definers types

SIGNAL O_MOTOR \$OUT[5]

SIGNAL O_VALOR \$OUT[1] TO \$OUT[3] ; por ejemplo para seleccionar de 0 a 7 programas

3.- ESTRUCTURAS DE CONTROL Y REPETICIÓN

3.1.- IF

Es una estructura condicional. Ejemplo:

IF (cajas<10) AND \$IN[2] THEN

primero()

ELSE

segundo()

ENDIF

Si la variable 'cajas' está por debajo de 10 y la entrada 2 está activada entonces se ejecuta 'primero'; en otro caso se ejecuta 'segundo'.

3.2.- SWITCH

Estructura que distingue entre casos. Ejemplo:

SWITCH cajas

CASE 1

primero()

CASE 2

segundo()

DEFAULT

tercero()

ENDSWITCH

Si la variable 'cajas' vale 1, se ejecuta 'primero'; si la variable 'cajas' vale 2, se ejecuta segundo; en cualquier otro caso se ejecuta 'tercero'.

Ejemplo enumeraciones:

SWITCH \$mode_op

CASE #T1

primero()

CASE #T2

segundo()

ENDSWITCH

Se trata de un conjunto acotado, es decir, la variable puede tomar sólo unos determinados valores. En este caso es la variable de sistema de modo de operación. Si está seleccionado el modo Test 1 se ejecuta 'primero'; si está seleccionado el modo Test 2 se ejecuta 'segundo'. Puedo crear mi propia enumeración (conjunto acotado) en el archivo 'config'.

3.3.- LOOP

Estructura de ejecución continua (bucle). Ejemplo:

```
LOOP  
  
    primero()  
  
    IF $IN[1] THEN  
  
        EXIT  
  
    ENDIF  
  
ENDLOOP
```

El programa 'primero' se ejecuta de forma indefinida hasta que se active la entrada 1. No se sabe el número de veces que se repetirá la ejecución.

3.4.- REPEAT

Estructura repetitiva. El programa 'primero' se ejecuta al menos una vez ya que no consulta la condición hasta el final. Ejemplo:

```
REPEAT  
  
    primero()  
  
UNTIL $IN[1]
```

3.5.- WHILE

Estructura repetitiva. El programa 'primero' se ejecuta si la entrada 1 está activa. Si no se cumple nunca no se ejecuta.

```
WHILE $IN[1]  
  
    primero()  
  
ENDWHILE
```

3.6.- FOR

Estructura que se repite un número de veces conocido de antemano.
Ejemplo:

```
FOR vueltas=1 TO 10
```

```
    primero()
```

```
ENDFOR
```


PRÁCTICAS: 8**INSTRUCCIONES DE REPETICIÓN Y CONTROL****OBJETIVOS:**

Conocer las instrucciones condicionales y de repetición y definir variables.

Escribir el texto del programa línea por línea sin utilizar los formularios del menú.

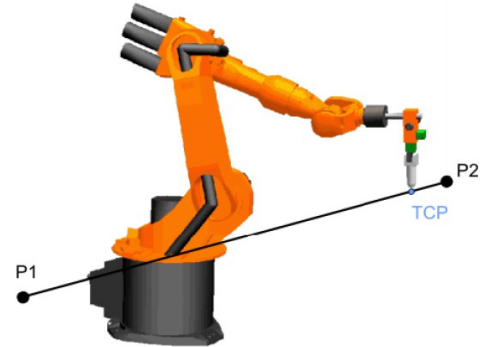
Realizar programas más complejos orientados a una tarea concreta.

DESCRIPCIÓN:

Se trata de programar la formación de un palet con el mínimo número de líneas de programa posibles.

INSTRUCCIONES:

1. Crear el programa "Palet".
2. Definir las variables necesarias para que todo quede configurable (varCajasBase, varAlturas, varDesplazamientoX, varDesplazamientoY, varDesplazamientoZ, etc)
3. Se definirá como punto inicial de dejada un punto concreto del sistema de coordenadas (este será el medido con la cuadratura donde se depositará el palet)
4. Cada paquete se supondrá que está posicionado en el punto de recogida a la activación de la entrada 1
5. Se deben definir puntos de aproximación en las cogidas y las dejadas para seguridad

DIBUJO:

```

LIN P1 VEL=0.3m/s CPDAT1
LIN P2 VEL=0.3m/s CPDAT2
SVN OUT 1 '' State= TRUE at START Delay=20ms
SVN OUT 2 '' State= TRUE at END Delay=-20ms
LIN P3 VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4
  
```

EJERCICIOS:

El robot debe formar dos tipos de palet:

- a) palet de base 8 y altura 3 (el formato del paquete es 300x400mm)
- b) palet de base 5 y altura 3 (el formato del paquete es 400x600mm)