



PRÁCTICA #2

TEMA: PROGRAMACIÓN CON TEACH PENDANT, LENGUAJE AS & ONLINE-KRTERM

1. Objetivos

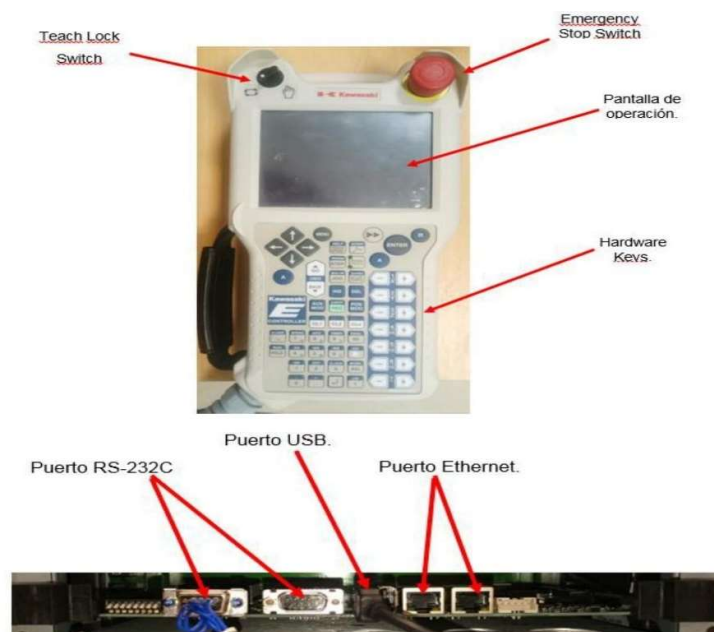
- Programar rutinas a partir del Teach Pendant.
- Introducción a lenguaje AS de robot Kawasaki.
- Programar una tarea, con programación de lenguaje AS.
- Definir variables con respecto al área de trabajo del robot.
- Conocer las características y la configuración de la programación offline.
- Establecer una red de comunicación ethernet entre el robot y una computadora.

2. Marco teórico

Unidad manual de servicio Teach Pendant (TP).

El teach pendant es un tipo de interfaz HMI diseñada para la programación y verificación de los programas a ejecutar por parte del robot industrial.

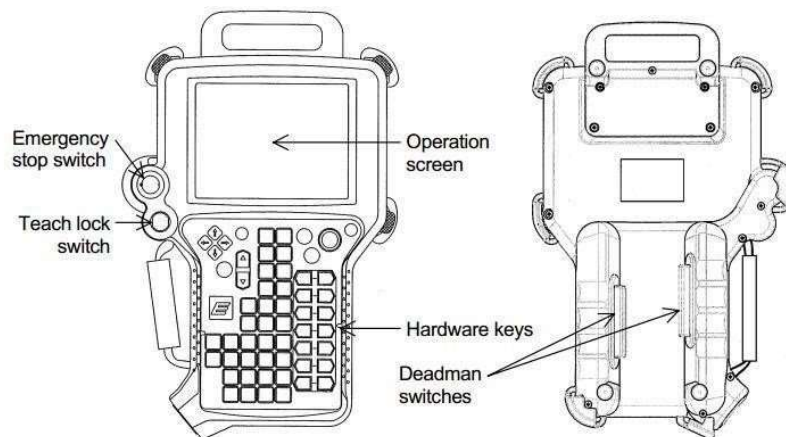
El TP está conformado por diversos interruptores y un conjunto de teclas que son necesarios para la operación manual del robot, además la pantalla de operación nos permite editar y visualizar todo tipo de datos.





Partes del Teach Pendant.

El TP contiene una diversa gama de botones donde el usuario puede configurar al robot industrial para que ejecute una acción ya sea de manera manual o repetitiva.



Configuración del teclado del TP.



Nota: Revisar el Manual de operación, capítulo 2, sección 3 y 4 (página 21-27)

Interfaz de trabajo del Teach Pendant.

La pantalla del TP se divide en tres áreas (A, B, C).





Las áreas B y C tienen estados activados y desactivados, las funciones que están activadas sobre las áreas son operables.

Nota: Revisar el Manual de operación, capítulo 2, sección 5, 6, 7 (página 27-36)

Programación

El controlador E70 del robot Kawasaki RS03N tiene la opción de crear programas por varios métodos que se mencionan a continuación:

- a) Robot de empleo:
 - Robot modo On-Line.
 - Robot Virtual o Robot modo Off-Line.
 - Combinación de ambos modos.
- b) Dispositivos de empleo:
 - Teach Pendant (TP).
 - Computadora de escritorio (PC).
 - Combinación de ambos instrumentos.
- c) Instrucciones de empleo:
 - Instrucciones compuestas.
 - Instrucciones tipo Mono-funciones (AS instrucciones).

Instrucciones y parámetros de la programación en bloque.

Esta programación se basa en instrucciones compuestas, cada una de las cuales está dividida por parámetros tales como:

- Interpolación.
- Velocidad.
- Precisión.
- Temporizador.
- Señales E/S.

A continuación, se muestra la pantalla principal, donde podemos observar las diferentes áreas:

- Área del programa.
- Área de instrucciones.

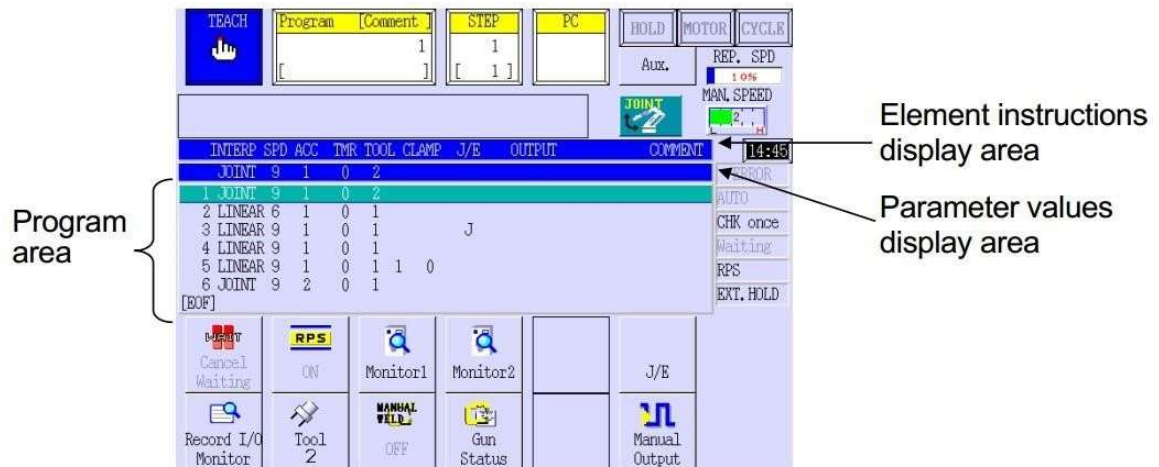


Tabla de valores de los parámetros.

Handling specification

Element intruc-tion	Interpolation	Speed	Accu-racy	Timer	Tool	Clamp	WK (Work)*	J/E (jump /End)	Output	Input
Parameter	JOINT/LINEAR/LIN2/ CIR1/CIR2/FLinear/ FCIR1/FCIR2/XLIN	0-9	0-4	0-9	1-9	No disp., 1-2	No disp., C	J, E	1-64 or 1-96	1-64 or 1-96
Keys	$\boxed{A+} \boxed{INTERP}$	$\boxed{A+} \boxed{SPD}$	$\boxed{A+} \boxed{ACC}$	$\boxed{A+} \boxed{TMR}$	$\boxed{A+} \boxed{TOOL}$ or $\boxed{<Tool>}$	$\boxed{CL\ 1/}$ $\boxed{CL\ 2}$	$\boxed{A+} \boxed{<Work>}$	$\boxed{A+} \boxed{J/E}$ or $\boxed{<J/E>}$	$\boxed{A+} \boxed{OX}$	$\boxed{A+} \boxed{WX}$

Descripción general del sistema AS

El lenguaje AS es usado para la comunicación con robots o para la programación.

El sistema AS se escribe en la memoria no volátil en la unidad de control del robot y cuando la alimentación del controlador está activada, el sistema AS se inicia y espera un comando que debe ejecutarse mediante el computador.

Características del sistema AS.

El lenguaje AS puede ser dividido en dos tipos:

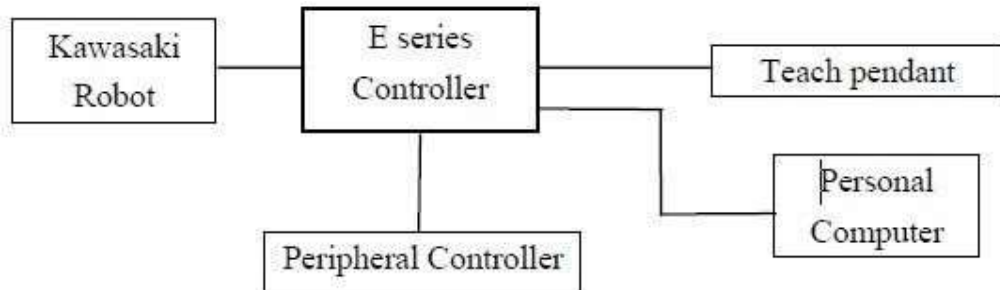
- Comandos de monitor.
- Instrucciones de programa.

Los comandos son usados para escribir, editar y ejecutar programas. Estos comandos son ingresados después del símbolo (>) que se muestra en la pantalla y se ejecutan inmediatamente. Las instrucciones son utilizadas para el movimiento del robot, del monitor o de otro control de señales externas, etc. en programas.



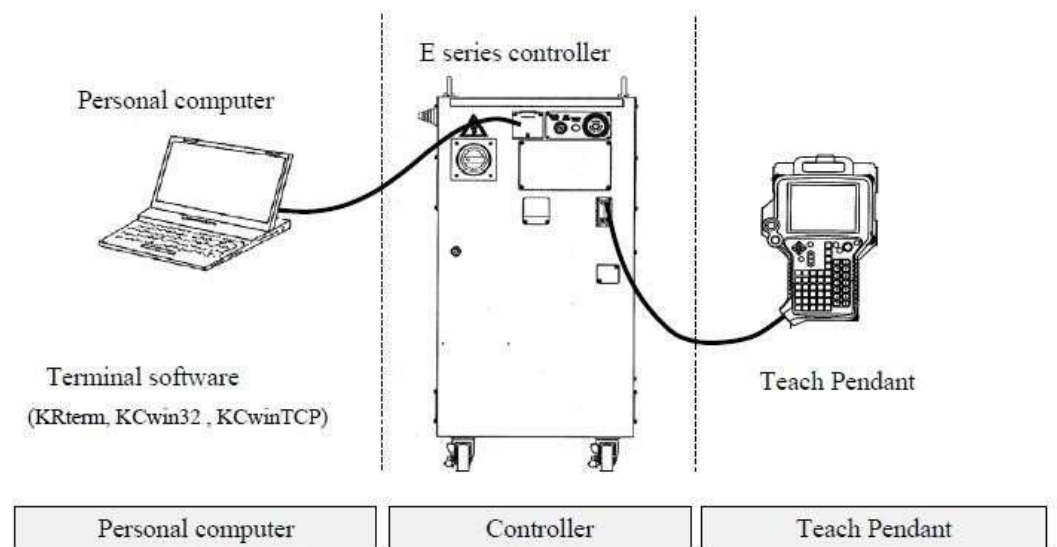
Configuración del sistema AS.

El controlador E del robot Kawasaki está distribuido por los siguientes componentes:



Conectando una computadora personal con el software terminal (KRterm) al controlador serie E se puede realizar las siguientes operaciones:

- Escribir comandos e instrucciones en lenguaje AS
- Guardar y cargar en el computador personal.





3. Comandos de lenguaje AS

Comando	Descripción
Home	Retorno a la posición de origen del robot.
Align	Alineación perpendicular del terminal del robot con el plano más cercano.
Jmove	Movimiento Joint del robot (cada articulación se mueve independientemente, de tal forma que busca la mejor trayectoria para llegar al punto deseado)
Lmove	Movimiento lineal del robot. Ej: Lmove p1 Donde p1 está previamente definido como: p1 -340 -10 -60 0 180 0
Jappro	Se aproxima tantos milímetros a un punto sobre el eje Z. Ej: Jappro p1,50
Hmove	Realiza movimiento híbrido, combina entre movimiento Joint y linear
Xmove	Realiza el movimiento de un punto a otro, pero se interrumpe si se activa una señal externa para dirigirse al próximo punto. Ej. Xmove P2 till 1001 Empieza a moverse al punto P2 y si la señal 1001 es activada ya no se dirige al punto P2, si no que se mueve hacia el siguiente punto que se encuentre en la próxima línea.
Where	Muestra los valores de cada articulación y las coordenadas x,y,z del terminal con respecto a la base activa.
Open1	Habilita la primera ventosa
Open2	Habilita la segunda ventosa
Ldepart	Se aleja tantos milímetros sobre el eje z del punto en el que se encuentra Ej: Ldepart 50
Speed	Determina la velocidad con la que se moverá el robot. Esta instrucción es válida solo para la siguiente línea de movimiento. Ej: Speed 5 (tomo el 5% de la velocidad máxima(6000mm/s))
Speed Always	A partir de esa instrucción el robot se moverá con esa velocidad hasta encontrar una nueva instrucción de cambio de velocidad. Ej: speed 65 mm/s always
Drive	Permite que el robot mueva cierta articulación tantos grados. Ej1: drive 4,-30 Ej2: Drive 4,30,0.75 Significa que se moverá la 4ta articulación, 30 grados positivos con velocidad de 0.75% de la máxima permitida por el robot



Draw	Realiza un movimiento cartesiano. Ej: Draw 25,-35,20 El robot se moverá con respecto a la base: 25 mm en X, -35 mm en Y y 20 mm en Z.
TDraw	Realiza un movimiento cartesiano. Ej TDraw 25,-35,20 El robot se moverá con respecto a la herramienta: 25 mm en X, -35 mm en Y y 20 mm en Z.
SETHOME	Varia la posición home predefinida
SET2HOME	Varia la posición home 2 predefinida
AVE_TRANS	Calcula el punto medio entre dos puntos. Ej: AVE_TRANS(P1,P2)

Nota: Revisar el Manual de lenguaje AS, capítulo 4, sección 2 (páginas 58 - 63) y manual de operación sección 9.0

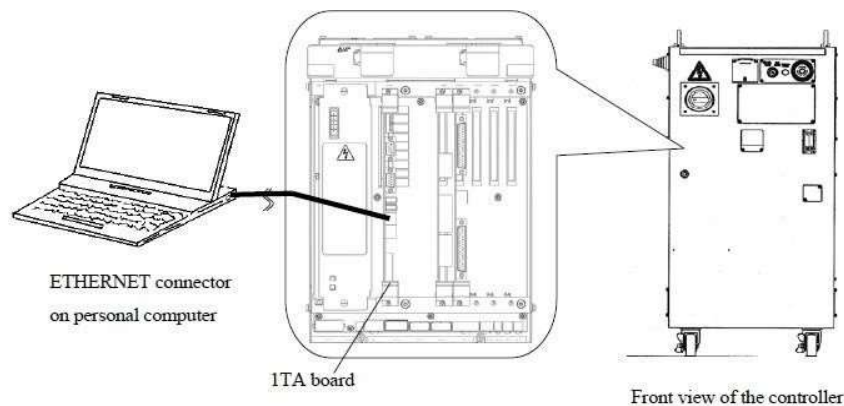
4. Descripción.

Comunicación ethernet con el robot

La conexión del controlador con la computadora personal usando Ethernet es posible mediante el software KRterm.

Para establecer la conexión se seguirán los siguientes pasos:

1. Conexión entre la PC y el controlador.

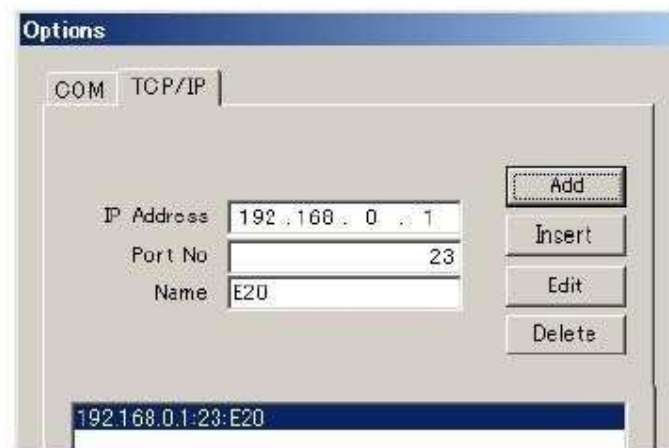


2. Ejecutamos el programa KRterm.
3. Registrar la dirección IP para conectar al robot.



Seleccionamos la pestaña TCP/IP e ingresamos los siguientes parámetros:

- Dirección IP. 192.168.30.13
- Puerto. 23
- Nombre (Opcional).



Estos parámetros sirven para conectar al robot con la red, finalmente le damos click en Add.

4. Conectamos al robot con la red.
5. Si la conexión fue un éxito, se mostrará la información del robot seguido de un mensaje "login:", ingresamos "as" después de este mensaje.

Nota: Para más información acerca de la comunicación ethernet mediante el programa KRterm revisar el manual de Lenguaje AS, capítulo 2, sección 6 (página 22-29)



5. Procedimiento 1

- a) Revisar que el interruptor del selector de modo del esté en modo “teach” para operar y configurar de forma manual al robot.
- b) Crear un nuevo programa con el Teach Pendant .
- c) Guardar diferentes puntos en el espacio de trabajo del robot, para después ejecutar en modo Repeat.
- d) Ejecutar el programa creado desde el Teach Pendant.

6. Procedimiento 2

- a) Abrir el editor de texto de su preferencia(bloc de notas o Notepad++)
- b) Para empezar se coloca:

.PROGRAM primerprograma ()

- c) Toda la programación a continuación será la que el robot ejecute en forma cíclica, infinitas veces o una sola vez dependiendo de lo que el usuario desee.
- d) Lo primero que se coloca en esta programación es:

BASE NULL ;lo cual elimina cualquier referencia previamente configurada

- e) Acontinuación:

SPEED 10 ;el robot se moverá en el 10% de su velocidad máxima

- f) Y colocaremos los JMOVE o LMOVE que se desee que el robot ejecute, para ello necesitamos definir los puntos, lo cual se realizará más adelante, por el momento solo colocaremos:

JAPPRO P1,25
LMOVE P1
LMOVE P2
C1MOVE P3
C2MOVE P4

- g) Y esa sección se termina con:



.END

h) A continuación colocamos en la siguiente línea:

.TRANS

i) En esta sección definiremos los puntos que se ocuparán:

P1 -350 0 -50 0 180 0

P2 -350 -100 -50 0 180 0

P3 -320 -100 0 0 180

P4 -320 0 0 0 180 0

j) Y finaliza esta sección con:

.END

k) A continuación se guarda el documento con el nombre que ustedes deseen pero con extensión .PG

l) El programa debería quedar así.

m) Ejecutar programa

```

1 .PROGRAM prueb1()
2 BASE NULL
3 SPEED 10
4 JAPPRO P1,25
5 LMOVE P1
6 LMOVE P2
7 CMOVE P3
8 CMOVE P4
9 .END
10 .TRANS
11 P1 -350 0 -50 0 180 0
12 P2 -350 -100 -50 0 180 0
13 P3 -320 -100 0 0 180 0
14 P4 -320 0 0 0 180 0
15 .END
16

```

NOTA: Es importante dar un *ENTER* al final, ya que si no lo hacen saldrá en su ejecución que no existe programa.

