

**MÓDULO DE PRÁCTICAS DE LABORATORIO PARA LA CELDA DE
MANUFACTURA DEL LABORATORIO DE PROCESOS INDUSTRIALES DE LA
UNIVERSIDAD DEL MAGDALENA**

MÓDULO DE PRÁCTICAS DE LABORATORIO

**ELKIN MANUEL RONDÓN PERALTA
CARLOS MARIO CALDERÓN MEDINA**

**UNIVERSIDAD DEL MAGDALENA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
2010**

TABLA DE CONTENIDO DEL MÓDULO DE PRÁCTICAS DE LABORATORIO

CAPÍTULO 1.....	6
1 CONCEPTOS TEÓRICOS	6
1.1 INTRODUCCIÓN DEL CAPÍTULO 1	6
1.2 LA AUTOMATIZACIÓN INDUSTRIAL	6
1.2.1 Objetivos de la automatización industrial	7
1.2.2 Aplicación de la automatización industrial	7
1.3 ROBÓTICA INDUSTRIAL	8
1.3.1 Componentes de los robots industriales	9
1.3.1.1 Manipulador	9
1.3.1.2 Efecto final.....	9
1.3.1.3 Fuente de poder	9
1.3.1.4 Sistema de control	10
1.4 INTRODUCCIÓN Y PARTES DE LA CELDA DE MANUFACTURA	10
1.4.1 Estación de trabajo 1	11
1.4.2 Brazo robótico.....	12
1.4.3 Gripper.....	12
1.4.4 Controlador NXC100.....	13
1.4.5 Controlador lógico programable	14
1.4.6 Base de desplazamiento lineal	14
1.4.7 Prensa Neumática	15
1.4.8 Sistema de almacenaje.....	15
1.4.9 Plantillas	16
1.4.10 RACKs Y BUFFER	17
1.5 COMO INICIAR DE MANERA ELÉCTRICA LA CELDA DE MANUFACTURA	18
1.6 LABORATORIO TEÓRICO	20
1.7 REVISIÓN DE CONCEPTOS TEÓRICOS DE LA CELDA DE MANUFACTURA	20
CAPÍTULO 2.....	22
2. CONTROL DE MANDO NXC100.....	22
2.1 INTRODUCCIÓN DEL CAPÍTULO	22
2.2 TEACH PENDANT	23

2.1.1 Miscelánea de teclas del Teach Pendant.....	24
2.1.2. Pantalla del Control de Mando.....	30
2.2 PRÁCTICA DE LABORATORIO.....	31
2.3 RECONOCIMIENTO DEL CONTROL DE MANDO.....	31
2.4 PRÁCTICA DE LABORATORIO.....	36
2.5 MOVIMIENTO EN COORDENADAS JOINT.....	36
2.6 PRÁCTICA DE LABORATORIO.....	43
2.7 MOVIMIENTO EN COORDENADAS CARTESIANAS.....	43
2.8 PRÁCTICA DE LABORATORIO.....	49
2.9 MOVIMIENTO EN COORDENADAS DE HERRAMIENTA.....	50
2.10 PRÁCTICA DE LABORATORIO.....	53
2.11 ABRIR Y EJECUTAR UN TRABAJO.....	53
2.12 PRÁCTICA DE LABORATORIO.....	56
2.13 ENSEÑAR POSICIONES.....	56
2.14 PRÁCTICA DE LABORATORIO.....	61
2.15 CREAR UN JOB.....	61
2.16 PRÁCTICA DE LABORATORIO.....	68
2.17 CREAR UN JOB UTILIZANDO VARIABLES.....	69
2.18 PRÁCTICA DE LABORATORIO.....	75
2.19 SALTO Y ETIQUETA.....	75
2.20 PRÁCTICA DE LABORATORIO.....	79
2.21 SUBROUTINAS.....	79
2.22 PRÁCTICA DE LABORATORIO.....	83
2.23 TEMPORIZACIÓN.....	83
2.24 PRÁCTICA DE LABORATORIO.....	85
2.25 USO DE LOS COMANDOS CALL, JUMP Y TIMER.....	86
2.26 PRÁCTICA DE LABORATORIO.....	91
2.27 PERIFÉRICOS DEL NXC100.....	91
CAPÍTULO 3.....	99
3. SCORBASE.....	99
3.1. INTRODUCCIÓN DEL CAPÍTULO.....	99
3.2 ENTRAR A SCORBASE.....	100

3.3 PRÁCTICA DE LABORATORIO.....	102
3.4 MOVIMIENTO DE LA BANDA DESLIZANTE.....	102
3.5 PRÁCTICA DE LABORATORIO.....	106
3.6 PROGRAMACIÓN DE POSICIONES.....	106
3.7 PRÁCTICA DE LABORATORIO.....	111
3.8 SALTO Y ETIQUETA.....	112
3.9 PRÁCTICA DE LABORATORIO.....	117
3.10 TEMPORIZACIÓN.....	117
3.11 PRÁCTICA DE LABORATORIO.....	124
3.12 DEFINIR VARIABLES.....	124
3.13 PRÁCTICA DE LABORATORIO.....	129
3.14 SALTO CONDICIONAL.....	129
3.15 PRÁCTICA DE LABORATORIO.....	133
3.16 VISUALIZACIÓN DE VALORES DE VARIABLE.....	133
3.17 PRÁCTICA DE LABORATORIO.....	137
3.18 COMANDOS DEL CONTROLADOR NXC100.....	137
3.19 PRÁCTICA DE LABORATORIO.....	142
3.20 SUBRUTINAS.....	142
CAPÍTULO 4.....	152
4. CONTROLADOR LÓGICO PROGRAMABLE.....	152
4.1 INTRODUCCIÓN DEL CAPÍTULO.....	152
4.2. ENTRADAS Y SALIDAS.....	154
4.2.1. Salida de Relé.....	155
4.2.2. Salida por Colector Abierto.....	156
4.2.3. Salidas Análogas.....	157
4.2.4. Terminales de entradas y Leds.....	157
4.2.5. Entradas Análogas.....	159
4.3. PRÁCTICA DE LABORATORIO.....	160
4.4. SALIDAS Y ENTRADAS DEL PLC.....	161
4.5. PRÁCTICA DE LABORATORIO.....	166
4.6. USO DE CONDICIONALES.....	166
4.7. PRÁCTICA DE LABORATORIO.....	175

4.8. INTERACCIÓN CON LOS PUERTOS ANÁLOGOS DEL CONTROLADOR LÓGICO PROGRAMABLE	175
CAPÍTULO 5.....	180
5 APLICACIONES	180
5.1 INTRODUCCIÓN DEL CAPÍTULO.....	180
5.2 PRÁCTICA DE LABORATORIO.....	181
5.3 APLICACIÓN MULTIPROPÓSITO	181
5.4 APLICACIONES DE ENSAMBLAJE.....	188
5.5 PRÁCTICA DE LABORATORIO.....	189
5.6 APLICACIÓN DE ENSAMBLAJE 1	189
5.7 PRÁCTICA DE LABORATORIO.....	192
5.8 APLICACIÓN DE ENSAMBLAJE 2.....	193
5.9 PRÁCTICA DE LABORATORIO.....	195
5.10 APLICACIÓN DE ENSAMBLAJE 3.....	196
5.11 PRÁCTICA DE LABORATORIO.....	199
5.12 APLICACIÓN DE ENSAMBLAJE 4.....	199

CAPÍTULO 1

1 CONCEPTOS TEÓRICOS

1.1 INTRODUCCIÓN DEL CAPÍTULO 1

En este capítulo se dará una breve introducción a todo el sistema automatizado de manufactura, se hablará de los conceptos básicos de los componentes que lo conforman y al final del capítulo se hará un laboratorio teórico con el fin de aclarar todo la teoría y partes de la Celda de Manufactura y así proseguir con los demás capítulos.

1.2 LA AUTOMATIZACIÓN INDUSTRIAL

Se define la automatización, por lo general, como el proceso de hacer que las máquinas sigan un orden predeterminado de operaciones con poca o ninguna mano de obra, usando equipo y dispositivos especializados que ejecutan y controlan los procesos de manufactura. La automatización en todo su potencial, se logra usando diversos dispositivos, sensores, actuadores, técnicas y equipo capaces de observar y controlar todos los aspectos del proceso de manufactura, de tomar decisiones acerca de los cambios que se deben hacer en la operación y de controlar todos los aspectos de ésta.

1.2.1 Objetivos de la automatización industrial

La automatización tiene los siguientes objetivos principales:

- a) Integrar diversos aspectos de las operaciones de manufactura para mejorar la calidad y la uniformidad del producto, minimizar los tiempos de ciclo y esfuerzos y con ello, reducir los costos de mano de obra.
- b) Mejorar la productividad, reduciendo los costos de manufactura a través del mejor control de la producción. Las piezas se cargan, alimentan y descargan de las máquinas con más eficiencia; las máquinas se usan con mayor eficiencia y la producción se organiza con más eficiencia.
- c) Mejorando la calidad empleando procesos más repetibles.
- d) Reducir la intervención humana, el aburrimiento y la posibilidad del error humano.
- e) Reducir daños a las piezas, causados por el manejo manual de las partes.
- f) Aumentar el nivel de seguridad para el personal, en especial bajo condiciones de trabajo peligrosas.

1.2.2 Aplicación de la automatización industrial

Se puede aplicar la automatización a la manufactura de todo tipo de bienes, desde materias primas hasta productos terminados y en todo tipo de producción, desde talleres hasta grandes instalaciones manufactureras. La decisión de automatizar

una instalación nueva o existente de producción requiere tener en cuenta los siguientes puntos:

- a) El tipo de producto manufacturado.
- b) La cantidad y la velocidad de producción requerida.
- c) La fase particular de la operación de manufactura que se va a automatizar
- d) El nivel de capacitación de la mano de obra disponible.
- e) Todo problema de confiabilidad y mantenimiento que se pueda relacionar con los sistemas automatizados.

1.3 ROBÓTICA INDUSTRIAL

Un robot o autómatas industrial se define como un manipulador multifuncional reprogramable, diseñado para mover materiales, piezas, herramientas u otros dispositivos mediante movimientos programados y ejecutar tareas diversas. La organización internacional de normalización o ISO describe al robot como una máquina formada por un mecanismo que incluye varios grados de libertad, que frecuentemente tiene el aspecto de uno o varios brazos terminados en una muñeca, capaces de sujetar una herramienta, pieza o dispositivo de inspección. En particular, la unidad de control de un robot debe usar un dispositivo de memoria. Estas máquinas de propósito general se suelen diseñar para ejecutar una función respectiva, pero con la alternativa de adaptarla a otras operaciones.

1.3.1 Componentes de los robots industriales

1.3.1.1 Manipulador

Se le llama brazo y muñeca. El manipulador es una unidad mecánica que permite movimientos parecidos a los de un brazo y una mano humana. El extremo de la muñeca puede llegar a un punto en el espacio que tenga un conjunto específico de coordenadas, en una orientación específica. La mayor parte de los robots tiene seis articulaciones rotacionales. También hay robots con cuatro grados y cinco grados de libertad pero por definición esta clase no es muy diestra, porque para serlo se necesitan seis grados de libertad.

1.3.1.2 Efector final

El extremo de la muñeca de un robot tiene un efector final, llamado también herramienta de extremo de brazo. Según el tipo de operación, los efectores finales pueden estar equipados con lo siguiente:

- a) Sujetadores, ganchos, Palas, electroimanes, campanas de vacío y dedos adhesivos, para manejar materiales.
- b) Pistolas de aspersión para pintar.
- c) Accesorios para soldar por puntos y con arco, y para corte con arco.
- d) Herramientas motorizadas, como taladros, llaves de tuerca y desbarbadores.
- e) Los efectores finales se suelen fabricar a la medida de necesidades especiales de manejo.

1.3.1.3 Fuente de poder

Cada movimiento del manipulador, en los ejes lineales y de rotación, se controla y regula mediante actuadores independientes que usan una fuente de poder eléctrica, neumática o hidráulica. Cada fuente de energía y cada tipo de motor tiene sus propias características, ventajas y limitaciones.

1.3.1.4 Sistema de control

Conocido también como controlador, el sistema de control es el sistema de comunicaciones y de procesamiento de información que emite los comandos de movimiento del robot. Es el cerebro del robot; almacena datos para iniciar y finalizar los movimientos del manipulador. También son los nervios del robot; se interconecta con computadoras y otros equipos, como celdas de manufacturas o sistemas de ensamble.

1.4 INTRODUCCIÓN Y PARTES DE LA CELDA DE MANUFACTURA

Para comenzar es importante tener claro el concepto de sistema de manufactura flexible cuyas siglas son FMS. Es un nuevo enfoque de la producción que con la aplicación de la tecnología ha creado sistemas altamente automatizados, También es una filosofía de la producción que se basa en el control efectivo del flujo de materiales a través de una red de estaciones de trabajo muy versátiles y compatible con diferentes grados de automatización.

Los FMS disponen de un sistema de manejo de materiales automatizado que transporta las piezas de una máquina a otra hacia dentro y fuera del sistema. El manejo de los FMS permite flexibilidad productiva, gestión en tiempo real y

acelerado nivel de automatización, así que una celda en línea es en resumen, aceptar el ingreso de materia prima y sacar productos listos para ser ensamblados.

Figura 1.1. Sistema de manufactura flexible.¹



En esta imagen se visualiza la Celda FMS del laboratorio de la Universidad del Magdalena, se hará una descripción de cada parte que la compone así como su funcionamiento.

Este sistema se compone de dos estaciones de trabajo, se le llamará a estas estaciones WS1 y WS2 (Work Station). Este módulo comprende única y exclusivamente todo lo que está relacionado con la estación de trabajo WS1 y el cual contiene un software denominado SCORBASE, que es principalmente el software que se usará a lo largo de los capítulos de este módulo.

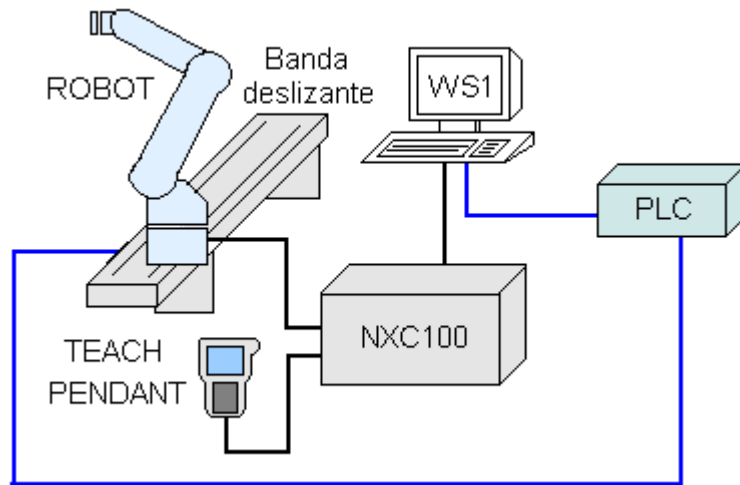
1.4.1 Estación de trabajo 1

También llamado WS1, esta estación de trabajo no es más que un PC, que tiene conexión con un PLC (controlador lógico programable) y un ACL (control de

¹ Fuente: fotografía de sistema de manufactura flexible de la Universidad del Magdalena tomada desde la página web de www.intelitek.com.

lenguaje avanzado) más adelante se detallarán estos dispositivos. La estación de trabajo 1 permite la coordinación de la celda de manufactura.

Figura 1.2. Esquema de conexión de PC².



1.4.2 Brazo robótico

El robot que hace parte del FMS, es de la línea MOTOMAN HP3 de la compañía Yaskawa electric corporation. Es un brazo robótico con 6 grados de libertad controlado por un ACL (control de lenguaje avanzado).

1.4.3 Gripper

² Fuente: Elaboración propia.

Es un dispositivo periférico diseñado para permitirle a un robot realizar operaciones de manipulación de material así como operaciones de ensamblaje. Consiste en un par de tenazas que se cierran sobre la pieza a transportar, o se abren para ubicar la pieza en un lugar determinado.

Figura 1.3. Gripper sosteniendo una pieza de materia prima³



El sistema utilizado para abrir y cerrar las piezas puede ser servo (mecánico) o neumático. El sistema del cual se dispone en el laboratorio de procesos industriales es neumático y se debe trabajar con una presión superior a los 80 psi.

1.4.4 Controlador NXC100

Es un ACL (control de lenguaje avanzado) es uno de los más pequeños en los controladores de su clase. Su compacta y robusta construcción minimiza el espacio y permite un fácil montaje en los transportadores, en armarios de control, o en otros pequeños lugares. Es ligero y fácil de instalar. Su función es la de controlar el brazo robótico, y lo hace mediante un Teach pendant o control de mando que incorpora una interface de usuario.

Figura 1.4. Controlador NXC100 y control de mando⁴.

³ Fuente: Fotografía tomada del sistema de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena.



1.4.5 Controlador lógico programable

Para esta celda de manufactura el PLC cumple una función muy importante, y es hacer desplazar el brazo robótico por una banda transportadora de nombre LSB (Linear Slidebase Belt-drive) o Base de desplazamiento lineal. En el capítulo 4 se describirá con detalle las especificaciones técnicas del PLC.

Figura 1.5. Controlador lógico programable⁴.



1.4.6 Base de desplazamiento lineal

La base de desplazamiento lineal, es una banda deslizante que permite desplazar el robot a lo largo de una distancia definida, acercando de esta manera al robot a las diferentes estaciones de trabajo de la celda de manufactura; en otras palabras puede servir como enlace a varios puntos dentro de una misma estación o como enlace entre varias estaciones.

⁴ Fuente: Fotografía tomada del sistema de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena.

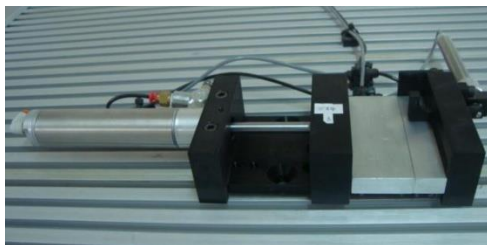
Figura 1.6. Base de desplazamiento lineal⁵.



1.4.7 Prensa Neumática

Al igual que el gripper, es un dispositivo periférico o actuador neumático diseñado para mantener fijo el material de trabajo. Por lo general se usa para ensamblaje, soldadura o cualquier operación de trabajo que requiera que el material de trabajo este fijo. Consiste en un par de cilindros neumáticos que se cierran sobre la pieza que se piensa fijar, o se abren para permitirle al brazo robótico remover el material de la prensa, al igual que el gripper requiere de una presión de aire de 80psi para su funcionamiento, su control es realizado desde la interfaz de mando por medio del Controlador NXC100.

Figura 1.7. Prensa neumática⁵.



1.4.8 Sistema de almacenaje

⁵ Fuente: Fotografía tomada del sistema de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena.

El Sistema de almacenaje y recuperación automatizada o ASRS (siglas en inglés de Automated Storage and Retrieval System) funciona como la principal fuente de materia prima en la Celda de Manufactura. Se utiliza normalmente un puesto ASRS. Los estantes ASRS vienen indexados y pueden contener plantillas (*templates*) vacíos o cargados de piezas. La celda de manufactura dispone de 2 M6 ASRS, 6 index o capacidad de 6 plantillas cada uno.

Un estante ASRS también puede utilizarse como una estación de productos en proceso o como una estación de productos terminados.

Figura 1.8. Sistema de almacenaje o ASRS⁶.

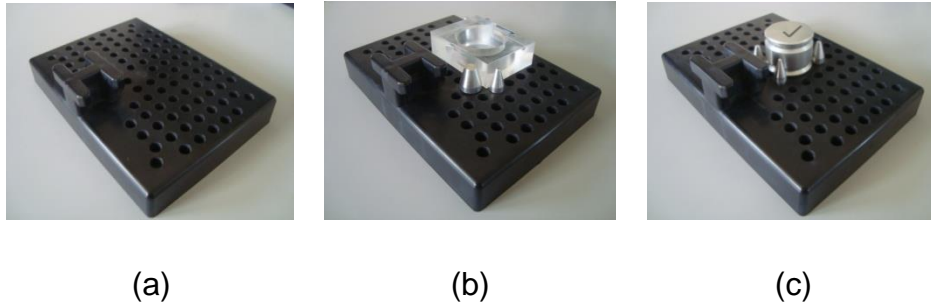


1.4.9 Plantillas

Las plantillas son bandejas de plástico que pueden contener varios tipos de piezas, el asa superior o delantera de la plantilla permite su fácil sujeción por el gripper del robot.

Figura 1.9. Plantillas: (a) Plantilla vacía (b) Plantilla con materia prima (c) Plantilla con otro tipo de materia prima⁶.

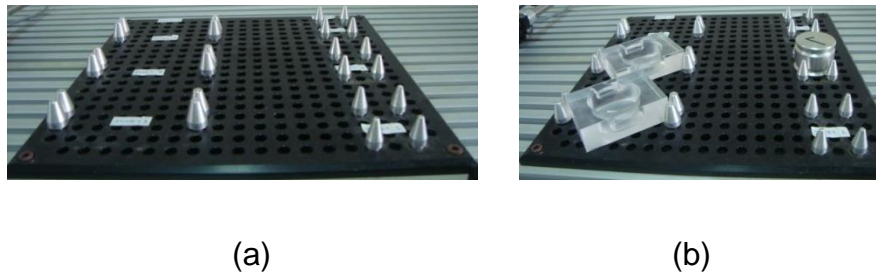
⁶ Fuente: Fotografía tomada del sistema de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena.



1.4.10 RACKs Y BUFFER

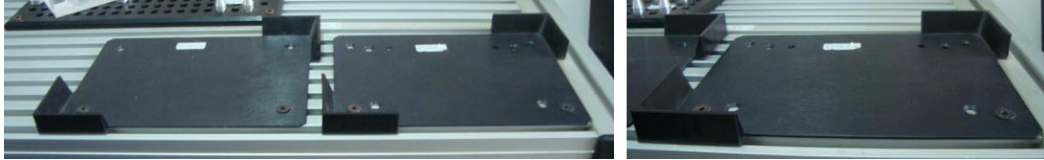
Al igual que los ASRS los racks y los buffers son dispositivos de almacenamiento indexados, la diferencia radica en que estos sirven de almacenamiento temporal. Un buffer funciona como un sistema de almacenamiento previo para la materia en proceso, para el laboratorio éste representa una placa metálica donde se colocan las plantillas que se extraen del ASRS. Un rack es una placa metálica o plástica cuadrada con una matriz de agujeros en los cuales se ajustan pasadores (similar a las plantillas) de acuerdo a la geometría del material de trabajo.

Figura 1.10. (a) Rack sin materia prima (b) Rack con materia prima⁷.



⁷ Fuente: Fotografía tomada del sistema de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena.

Figura 1.11. Buffer indexado.



1.5 COMO INICIAR DE MANERA ELÉCTRICA LA CELDA DE MANUFACTURA

Los pasos para iniciar la celda de manufactura son los siguientes:

1. Asegurarse que los multitomas estén debidamente conectados y encendidos, tenga en cuenta la señalización de energía para hacer la conexión.
2. Encender el PC1 o estación de trabajo 1.
3. Encender el PLC, el switch de encendido se encuentra en la parte de atrás del mismo.
4. Halar el botón rojo de emergencia del PLC, en caso de riesgo o emergencia se debe presionar éste botón para detener el cualquier proceso que tenga que ver con el PLC.
5. Encender el controlador NXC100, el switch de encendido se encuentra en la parte frontal del mismo.
6. Halar el botón rojo de emergencia del controlador NXC100 y del Teach pendant o control de mando, en caso de alguna emergencia o riesgo con el brazo robótico se debe presionar cualquiera de los dos botones de emergencia.

Nota: Es importante asegurarse que todas las conexiones de los dispositivos estén correctas. En el momento de apagar toda la celda, se recomienda apagar cada dispositivo del paso 6 al 1 como lo indica la secuencia de encendido.

1.6 LABORATORIO TEÓRICO

1.7 REVISIÓN DE CONCEPTOS TEÓRICOS DE LA CELDA DE MANUFACTURA

1.6.1 Objetivos generales de aprendizaje

Manejar los conceptos importantes de la automatización en la industria y en el sistema de manufactura flexible del laboratorio de procesos industriales.

1.7.2 Objetivos específicos de aprendizaje

- Manejar el lenguaje técnico usado en la celda de manufactura.
- Identificar los dispositivos hardware que componen la celda de manufactura.
- Identificar las aplicaciones de la robótica industrial.

1. ¿Cuál es el significado de las siglas FMS?

2. ¿Cuál es el significado de las siglas PLC?

3. ¿Cuál es la referencia del brazo robótico?

4. ¿Cuáles son los componentes de un robot industrial?

5. ¿Cuántos grados de libertad tiene el brazo robótico HP3 MOTOMAN?

6. ¿Qué quiere decir ACL?

7. ¿Cuál es primer paso para iniciar el funcionamiento eléctrico del FMS?

8. ¿Cómo se llama el software que utiliza WS1 para la coordinación de la celda de manufactura?

9. ¿Qué tipo de mecanismo utiliza el GRIPPER para cerrar y abrir sus pinzas?

10. ¿Cuántas plantillas se pueden almacenar en el ASRS?

11. ¿Qué almacena una Rack?

12. ¿Qué almacena un Buffer?

1.6.2 Conclusiones

CAPÍTULO 2

2. CONTROL DE MANDO NXC100

2.1 INTRODUCCIÓN DEL CAPÍTULO

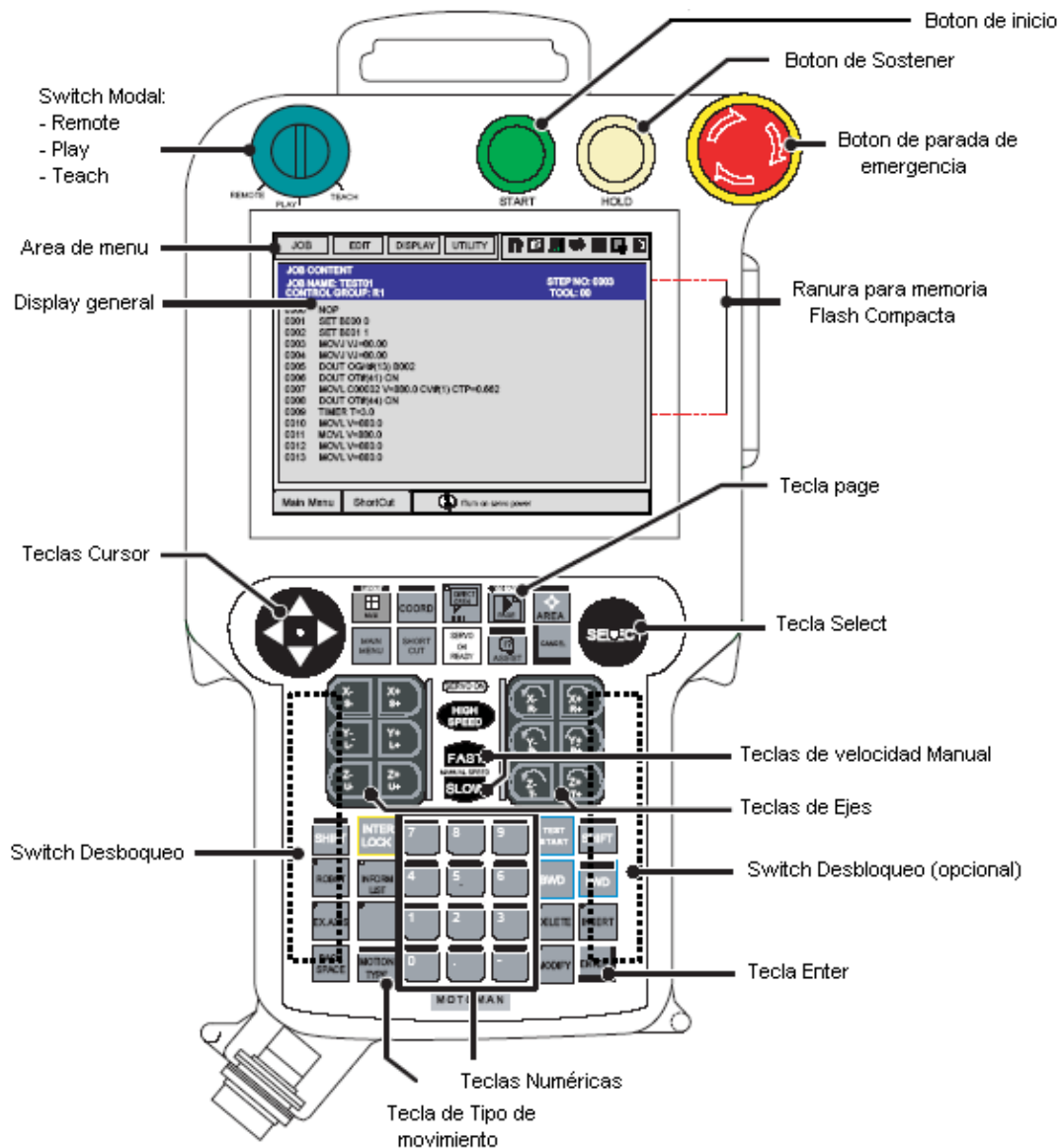
En este capítulo aprenderá el manejo del **Teach Pendant** o control de mando del controlador NXC100, aprenderá a mover los 6 grados de libertad del brazo robótico y programará sobre el **Teach Pendant**.

2.2 TEACH PENDANT

Antes de comenzar a manipular el brazo robótico, es importante conocer la estructura física y funciones de algunas teclas importantes del control de mando o **Teach pendant**.




Figura 2.1. Esquema del control de mando o Teach pendant⁸.





⁸ Fuente: Imagen tomada del manual de operador NXC100 para propósito general. Manual numero RE-CSO-A032.


















2.1.1 Miscelánea de teclas del Teach Pendant







Aquí se muestra en gran parte la función de los botones del control de mando del NXC100.

TECLA	DESCRIPCIÓN
	<p>Botón de parada de emergencia: Como su nombre lo indica, sirve para bloquear los movimientos del robot en caso de emergencia.</p>
	<p>Switch de desbloqueo: En modo TEACH, si se presiona levemente activa los ejes del robot, con lo cual se debe escuchar un leve pitido del servo. Si no se presiona o se presiona fuertemente los ejes del robot quedaran bloqueados.</p>
	<p>Switch modal: Permite seleccionar el modo de operación TEACH, PLAY o REMOTE.</p> <p>En modo TEACH se pueden operar los ejes manualmente, enseñar posiciones y editar trabajos (JOBS).</p> <p>En modo PLAY se ejecutan los trabajos enseñados.</p> <p>En modo REMOTE el robot es controlado por una señal externa.</p>
	<p>Cursor: Permite desplazarse a través de las áreas e ítems de trabajo.</p>

	Select: Permite seleccionar los ítems que son mostrados en la pantalla.
	Main Menu: Muestra u oculta la pantalla de menú del display general.
	Área: Permite movilizarse entre las área de la pantalla.
	Display: Activa el comando para buscar páginas.
	Direct Open: Muestra el contenido relacionado a una línea (p.e. muestra el JOB contenido en un CALL)
	Coord: Permite seleccionar el sistema de coordenadas (Joint, Cartesiano, Herramienta y Usuario) en modo TEACH.
	Velocidad manual: Permite controlar la velocidad (INCH, SLOW, MED, FAST) del robot en modo TEACH.
	High Speed: Al presionar esta tecla y mover un eje se alcanza la máxima velocidad.

	Motion Type: Permite seleccionar el tipo de interpolación (joint, lineal, circular, y helicoidal) entre dos puntos
	Robot: Activa los ejes de operación del robot (siempre activo cuando no hay ejes externos).
	Axis: Activa los ejes externos del robot (cuando hay ejes externos).
	Teclas de Ejes: Mueven los ejes del robot. Se debe tener en cuenta el sistema coordinado y la velocidad al operarlos.
	Test Start: Se puede probar un JOB sin necesidad de pasar a modo PLAY. INTERLOCK+TEST START prueba los JOBS de forma controlada.
	Forward: Permite moverse a través de los bloques creados en un programa
	Backward: Permite moverse a través de los bloques creados en un programa en sentido contrario.
	Inform List: Muestra la lista de comandos disponibles para la edición de un trabajo.

	Cancel: Permite cancelar la introducción de datos y quita informes de error.
	Delete: Borra instrucciones y datos introducidos. El borrado concluye con la tecla ENTER y cuando el LED este encendido.
	Insert: Inserta nuevas instrucciones y datos. La inserción concluye con la tecla ENTER y cuando el LED este encendido.
	Modify: Modifica datos de posición, instrucciones y datos enseñados. La modificación concluye con la tecla ENTER y cuando el LED este encendido.
	Enter: Registra instrucciones, datos y datos de posición.
	Shift: Cambia la función de otra tecla mientras se presiona esta.
	Interlock: Cambia la función de otra tecla mientras se presiona esta.

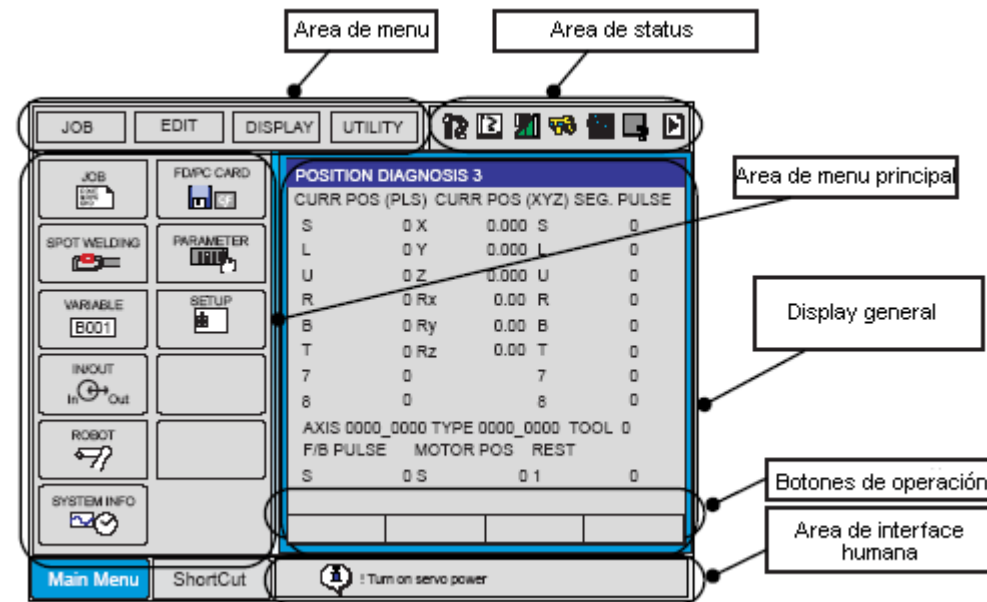
	<p>Teclado Numérico: Ingresa números cuando el símbolo ">" aparece. "." Y "-" representan punto decimal y símbolo menos respectivamente.</p>
 START	<p>Start: En modo PLAY, ejecuta un JOB enseñado.</p>
 HOLD	<p>Hold: Detiene un JOB ejecutado en modo PLAY.</p>
	<p>Servo on ready: En modo PLAY y REMOTE activan los ejes del robot (cumple la misma función que el SWITCH DE DESBLOQUEO)</p>
	<p>No aplican para el presente sistema.</p>
	<p>Back Space: Borra el último carácter mientras se está introduciendo caracteres.</p>

De ahora en adelante cada vez que se refiera al uso de una tecla, se hará la equivalencia expresando el nombre de la tecla entre llaves:



2.1.2. Pantalla del Control de Mando

Figura 2.2. Pantalla del control de mando o Teach Pendant⁹.



Esta pantalla tiene la posibilidad de funcionar de manera táctil, es decir con el tacto es posible presionar sobre algunas opciones que muestra la pantalla. Por ejemplo, si se tiene el **Teach Pendant** encendido, y se toca con un dedo la opción **Main menu**, a continuación se desplegará una serie de opciones como lo muestra la figura 2.2.

⁹ Fuente: Imagen tomada del manual de operador NXC100 para propósito general. Manual numero RE-CSO-A032.

2.2 PRÁCTICA DE LABORATORIO

2.3 RECONOCIMIENTO DEL CONTROL DE MANDO

2.3.1 Objetivo general de aprendizaje

Iniciar de manera eléctrica la celda de manufactura y hacer un reconocimiento del control de mando del NXC100 en modo de enseñanza.

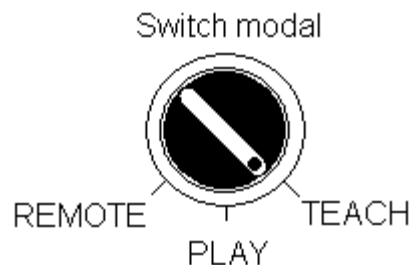
2.3.2 Objetivos específicos de aprendizaje

- Utilizar el modo de enseñanza del control de mando.
- Iniciar la celda de manufactura.
- Cambiar la velocidad de los servos del robot.
- Cambiar el tipo de coordenada del robot.
- Utilizar el switch de desbloqueo para activar los servos del brazo.

2.3.3 Concepto importante: Modo Teach

El Teach Pendant funciona en tres modos diferentes que son: REMOTE, PLAY y TEACH.

Figura 2.3.1. Switch modal en posición de enseñanza¹⁰.



Con una pequeña perilla podrá elegir el modo de funcionamiento, dependiendo de lo que se quiera hacer, el modo TEACH es el más usado y fundamental ya que permite mover de forma manual el robot y enseñarle posiciones.

¹⁰ Fuente: Elaboración propia.

2.3.4 Procedimiento

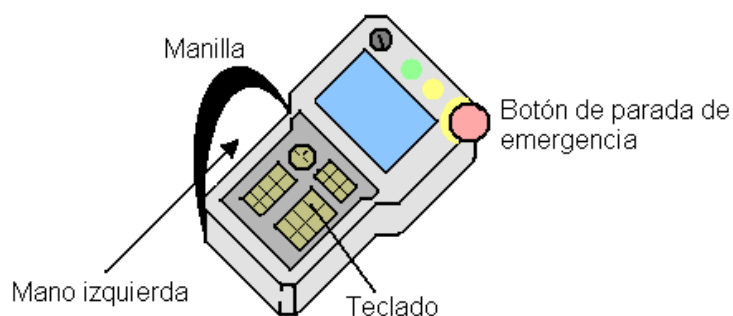
Como ya se había mencionado antes, el modo de enseñanza es el más esencial por que permite manipular todas las opciones y funciones del Teach Pendant. Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos para hacer el reconocimiento del control de mando.

1. Encienda el controlador NXC100 con los pasos antes mencionados en el capítulo 1.
2. Sujete el Teach pendant e introduzca la mano izquierda en la manilla, dejando el dedo pulgar sobre el teclado.

Figura 2.3.2. Forma de coger el Teach pendant con la mano izquierda¹¹.



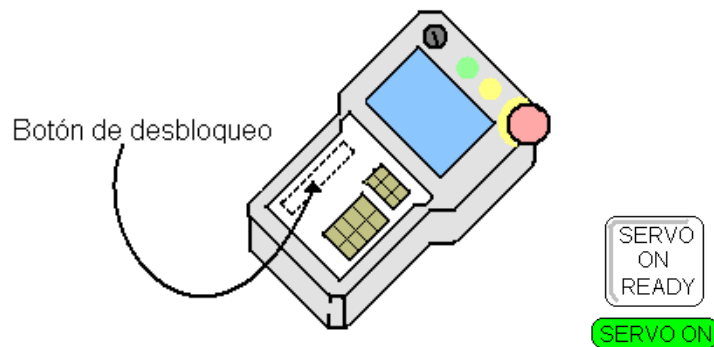
3. Asegúrese de desbloquear los ejes del robot. Para ello se debe desactivar el **{BOTÓN DE PARADA DE EMERGENCIA}** del Teach Pendant girándolo

¹¹ Fuente: Elaboración propia.

levemente hacia la derecha, haga lo mismo con el **{BOTÓN DE EMERGENCIA}** del controlador NXC100.

4. Convoque el modo de enseñanza girando el **{SWITCH MODAL}**, parte superior izquierda del Teach pendant, gírelo hasta la marca TEACH.
5. Presione levemente el **{Switch de Desbloqueo}**, escuchará un leve pitido que indica que los ejes están listos para moverse, además un indicador luminoso de SERVO ON deberá iluminar fijamente. Se presionará **{Switch de Desbloqueo}** únicamente para mover el robot de forma manual.

Figura 2.3.3. Teach pendant mostrando el switch de desbloqueo¹².

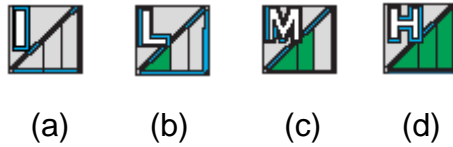


6. Observe el área de status de la pantalla que se encuentra en la parte superior derecha. En ella debe aparecer uno de los siguientes símbolos:

Figura 2.3.4. Velocidades de movimiento del robot: (a) Incremental (b) Baja (c) Media (d) Alta¹³.

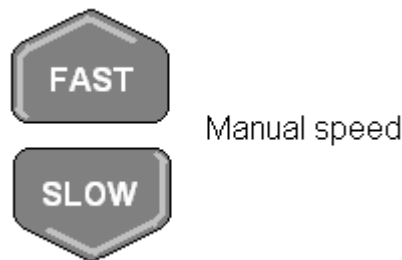
¹² Fuente: Elaboración propia.

¹³ Fuente: Manual de operador de NXC100 numero RE-CSO-A032.



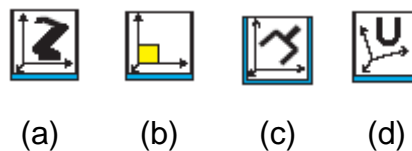
7. Presione las teclas de velocidad manual y observe los cambios en el área de status de la pantalla del Teach pendant.

Figura 2.3.5. Teclas de velocidad manual para los servos del brazo robótico¹⁴.



8. Observe en el área de status de la pantalla, debe aparecer alguno de los siguientes símbolos.

Figura 2.3.6. Tipo de coordenadas del robot: (a) Joint (b) Cartesiana (c) Herramienta (d) Usuario¹⁵.



9. Presione la tecla **{COORD}** y observe el cambio de sistema de coordenadas.

¹⁴ Fuente: Elaboración propia.

¹⁵ Fuente: Manual de operador de NXC100 numero RE-CSO-A032.

Figura 2.3.7. Ubicación de la tecla COORD en el Teach pendant¹⁶.



2.3.5 Conclusiones

2.4 PRÁCTICA DE LABORATORIO

2.5 MOVIMIENTO EN COORDENADAS JOINT

¹⁶ Fuente: Elaboración propia.

2.5.1 Objetivo general de aprendizaje

Utilizar el tipo de coordenadas Joint para mover los 6 grados de libertad o ejes del brazo robótico.

2.5.2 Objetivos específicos de aprendizaje

- Identificar cada eje del brazo robótico.
- Utilizar las teclas del Teach Pendant que mueven cada eje del brazo robótico.
- Llevar el brazo robótico a la posición inicial.

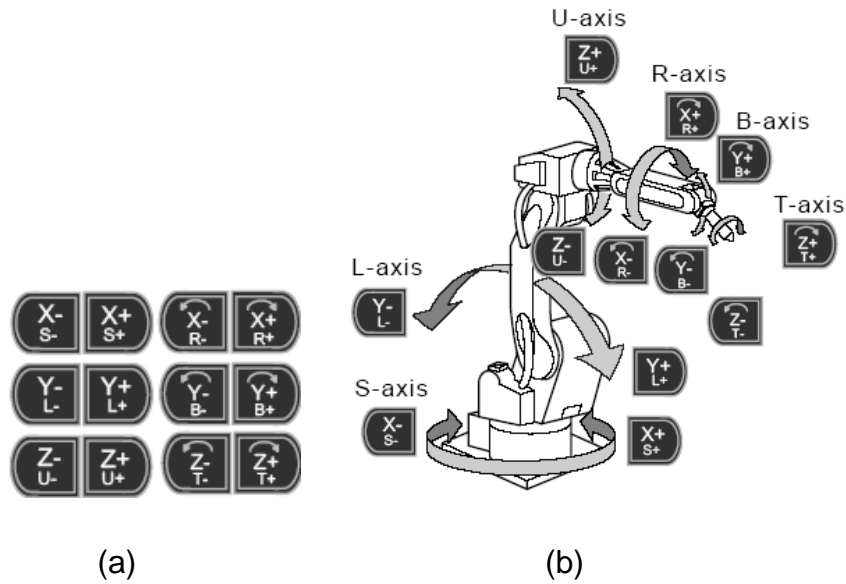
2.5.3 Concepto importante: Ejes del brazo robótico

Observe la figura 2.5.1 la cual muestra la distribución de los ejes y la relación de cada botón de **{Teclas de Ejes}** con el robot.

Los ejes S, L y U son los ejes mayores los cuales representan los movimiento principales y de gran desplazamiento. Los ejes R, B y T son los ejes que brindan movimientos de precisión.

Figura 2.5.1. (a) Teclas de ejes del Teach Pendant (b) Ejes del robot¹⁷.

¹⁷ Fuente: Manual de operador de NXC100 número RE-CSO-A032.






2.5.4 Procedimiento

Los diferentes movimientos que se van a trabajar en este laboratorio son en coordenadas joint. Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos para mover el brazo en este tipo de coordenadas.

1. Observe el área de status de la pantalla del Teach Pendant y verifique si el robot se encuentra activado en coordenadas Joint  , si es diferente cambie el tipo de coordenada con el botón **{COORD}**  .
2. Cambie la velocidad de los servos, colóquela en media  para evitar cualquier tipo de accidente.

3. Para comenzar a mover los ejes del robot es necesario llevarlo a una posición inicial, esto se hace de la siguiente manera:

a) Dirijase al Teach pendant y presione el botón **{Main menu}** para desplegar varias opciones.

b) Elija la opción **{ROBOT}**, se desplegará otras opciones, en esa nueva lista elija **{WORK HOME POS}**.

A continuación en el espacio de Display general del Teach Pendant aparecerá en la parte superior el título WORK HOME POSITION, observe que aparecen los 6 ejes del robot S,L,U,R,B,T con origen 0,0,0,0,0,0 y valores actuales o current que dependen de la posición del robot. Home position o posición inicial tiene valores 0 en todos los ejes del robot.

Figura 2.5.2. Display general mostrando la posición inicial del brazo robótico¹⁸.

WORK HOME POSITION		
	ORIGIN	CURRENT
R1: S	0	234
L	0	212
U	0	129
R	0	-7
B	0	-12234
T	0	0

Display general

4. Seleccionada la opción **{WORK HOME POS}**, presione el botón de **{Switch de Desbloqueo}** que se encuentra en la parte de atrás del Teach Pendant,

¹⁸ Fuente: Elaboración propia.

manténgalo presionado y asegúrese que el indicador luminoso de SERVO ON este encendido y un leve pitido se mantenga.





5. Presione el botón **{FWD}** de la botonera del Teach Pendant, este botón se encuentra en la parte inferior derecha. Tenga en cuenta que para llevar el robot a la posición inicial se debe tener presionado **{Switch de Desbloqueo}** + **{FWD}**. Si el robot ya se encontraba en la posición inicial suelte los botones.
6. Mueva el eje **S**, presione **{Switch de Desbloqueo}** manténgalo presionado y luego pulse el botón  de **{Teclas de Ejes}** puede elegir en presionar **S-** o **S+**, el brazo debe moverse sin problemas haciendo un movimiento de giro.
7. Mueva el eje **L** , el movimiento del brazo deberá ser adelante y atrás.
8. Mueva el eje **U**,  y hágalo en dirección **U-**, el eje U se moverá hacia abajo, asegúrese que no haya ningún obstáculo y siga moviéndolo hasta que el brazo se detenga por sí solo, escuchara un fuerte pitido, esto se debe a que existe un límite de movimiento para cada eje del brazo. Observe el área de interface humana, parte inferior derecha de la pantalla del Teach Pendant, deberá decir; “PULSE LIMIT (MIN.)” para evitar este mensaje simplemente mueva el eje U en dirección **U+**.
9. Diríjase a la figura 2.5.1, donde aparece la relación de las **{Teclas de Ejes}** con el brazo robótico y demuestre los movimientos de los demás ejes.
10. En la pantalla táctil del Teach Pendant seleccione **{Main menu}**, luego **{ROBOT}**, en la lista de opciones seleccione **{CURRENT POSITION}**, en el espacio de Display general aparecerá algo como la siguiente figura:

Figura 2.5.3. Posición actual de cada eje del Robot¹⁹.

CURRENT POSITION		TOOL: 00
COORDINATE	PULSE	
R1: S	234	Display general
L	212	
U	129	
R	-7	
B	-12234	
T	0	

Asegúrese que *COORDINATE* sea de tipo *PULSE*, si no es así, presione el

botón **{SELECT}**  del teclado del Teach Pendant, se desplegarán las opciones, *PULSE*, *BASE*, *ROBOT*, *USER*, esto es simplemente el sistemas de coordenadas que se quiere observar para cada eje del robot. Escoja *PULSE* y presione **{SELECT}**.

11. Lleve los ejes del brazo robótico a las siguientes posiciones.

S = entre 61000 y 62000

L = entre -60000 y -61000

U = entre -45000 y -46000

R = entre -86000 y -87000

¹⁹ Fuente: Elaboración propia.

B = entre -83000 y -84000

T = entre 10000 y 10200

12. Preguntas.

a) ¿Qué tipo de movimiento observó cuando movió los diferentes ejes del robot en coordenadas Joint?

b) ¿Qué ocurre cuando los ejes llegan al límite de movimiento?

c) ¿Es posible mover el robot con las **{Teclas de Ejes}** en el modo remoto?

2.5.5 Conclusiones

2.6 PRÁCTICA DE LABORATORIO

2.7 MOVIMIENTO EN COORDENADAS CARTESIANAS

2.7.1 Objetivo general de aprendizaje

Realizar movimientos del brazo robótico con referencia a un sistema de coordenadas cartesianas.

2.7.2 Objetivos específicos de aprendizaje

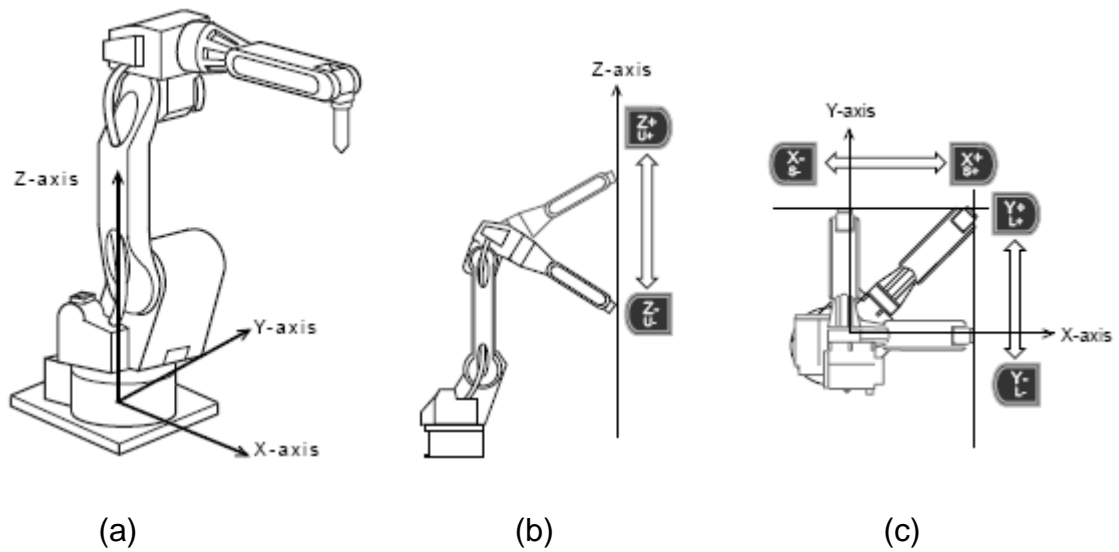
- Efectuar movimientos de cada eje en coordenadas cartesianas.
- Identificar las trayectorias de los diferentes ejes del manipulador en el sistema de coordenadas cartesianas.

2.7.3 Concepto importante: Sistema de ejes en tres dimensiones

En coordenadas Joint los ejes del brazo robótico se podían mover de forma independiente, en coordenadas cartesianas la base del brazo forma un sistema de ejes en tres dimensiones X, Y, Z. por lo que en consecuencia se va a trabajar con los ejes **X**, **Y** y **Z** y no con S, L, U, R, B, T como en coordenadas Joint.

Figura 2.7.1. (a) Sistema cartesiano **X**, **Y** y **Z** (b) Movimiento paralelo al eje **Z** (c) Movimiento paralelo al eje **X** o **Y**²⁰.

²⁰ Fuente: Manual de operador de NXC100 numero RE-CSO-A032.



2.7.4 Procedimiento


Para esta práctica de laboratorio se hará un análisis de los movimientos en coordenadas cartesianas. Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga a continuación los siguientes pasos.

1. Con el Teach Pendant a la mano y en modo **{TEACH}**, lleve el brazo robótico a home position.

2. Presione el botón **{COORD}** y escoja coordenadas cartesianas .

3. Cambie la velocidad de servo a media  para comenzar los primeros movimientos, luego puede cambiar a velocidad alta.

4. Compruebe la trayectoria **Z** presionando **{Switch de Desbloqueo}** y **Z-** o **Z+** de **{Teclas de Ejes}**. Observe que la herramienta del robot traza una línea vertical paralela al eje Z, en caso de exceder el área de alcance del robot suelte **{Switch de Desbloqueo}** y presione **{RESET}** de la pantalla táctil, esta opción aparecerá en la parte inferior derecha del Display general, para llevar al robot nuevamente a una posición no limitada debe retomar el mismo eje pero en sentido contrario. Este error ocurre cuando el robot sale de su envolvente, muy diferente cuando su movimiento se ve limitado por algún giro o movimiento de un eje particular, entonces para el error de envolvente el Teach pendant siempre va pedir reiniciar la operación el cual se hace con **{RESET}**.

5. Pruebe la trayectoria con el eje **X** utilizando las **{Teclas de Ejes}**, hágalo en ambas direcciones y observe que la herramienta traza una línea paralela al eje X.

6. Pruebe la trayectoria con el eje **Y**, aquí se demuestra que el origen del sistema cartesiano es la base del robot.

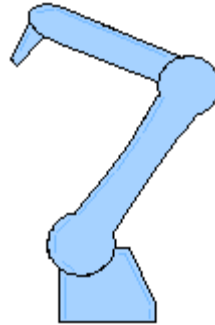
7. Con el botón **{COORD}** cambie el sistema de coordenadas a joint. Recuerde que en **{CURRENT POSITION}** debe tener la opción COORDINATE en *PULSE*, también puede ver los pulsos de los ejes en el Display general de WORK HOME POSITION.

8. Lleve el brazo robótico a la siguiente posición:

S = entre 0 y -2
L = entre -38000 y -39000

U = entre -4700 y -5800
R = entre 0 y -2
B = entre -79000 y -80000
T = entre 0 y -2

Figura 2.7.2. Posición que debe resultar del paso 8²¹.






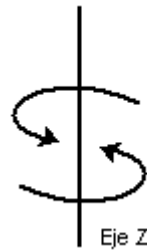
9. Ahora cambie el tipo de coordenadas a cartesianas  , después presione **{Switch de Desbloqueo}** y  de **{Teclas de Ejes}** puede hacerlo con una velocidad alta  . Observe que la herramienta comienza a girar alrededor del eje Z.

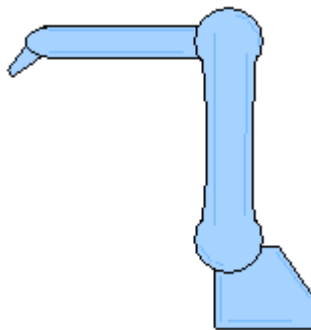
Figura 2.7.3 Trayectoria de la herramienta en el eje Z²².

²¹ Fuente: Elaboración propia.

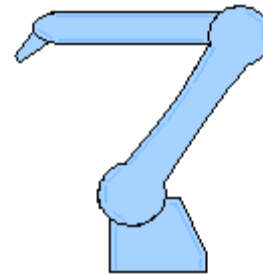


10. Repita el paso 9 ahora con y , la trayectoria de la herramienta es un giro alrededor del eje X y Y respectivamente. trate de llevar el robot a las siguientes posiciones para que pueda apreciar bien este ejercicio.

Figura 2.7.4. (a) Posición para hacer prueba en X (b) Posición para hacer prueba en Y²².



(a)



(b)

11. Preguntas

a) ¿Por qué es más fácil entrar en el error de límite de movimiento y envolvente en coordenadas cartesianas que en coordenadas Joint?

²² Fuente: Elaboración propia.

b) ¿Cuál es la ventaja de usar coordenadas cartesianas?

2.7.5 Conclusiones

2.8 PRÁCTICA DE LABORATORIO

2.9 MOVIMIENTO EN COORDENADAS DE HERRAMIENTA

2.9.1 Objetivo general de aprendizaje

Mover el brazo robótico con referencia a un sistema de coordenadas de herramienta.

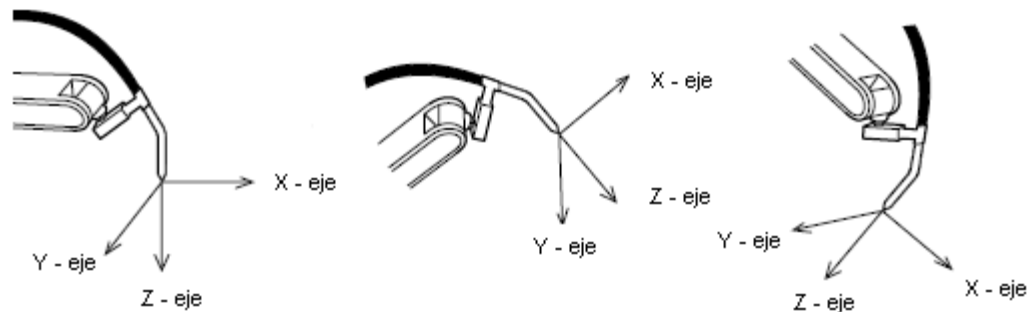
2.9.2 Objetivos específicos de aprendizaje

- Verificar cada eje en coordenadas de herramienta.
- Proponer aplicaciones del brazo robótico en coordenadas de herramienta.

2.9.3 Concepto importante: Sistema coordinado de herramienta

Este tipo de coordenada a veces resulta ser muy útil, para saber cómo funciona piense un poco en la regla de la mano derecha que es la que se aplica en la herramienta o Gripper del brazo robótico. El eje Z siempre será paralelo a la punta de la herramienta, la dirección de los otros ejes depende de cómo esté ubicada la herramienta en este caso el Gripper.

Figura 2.9.1. Sistema de coordenadas en la herramienta del brazo robótico²³.




2.9.4 Procedimiento

²³ Fuente: Manual de operador de NXC100 numero RE-CSO-A032

La nomenclatura de ejes que trabaja este sistema de coordenadas es **X**, **Y** y **Z**. Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos para la elaboración de este laboratorio.

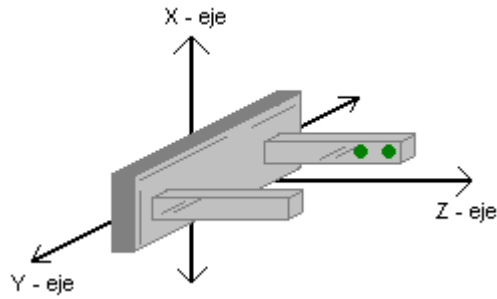
1. Lleve el robot a home position y después cambien el tipo de coordenadas con el botón **{COORD}** del teclado del Teach pendant, elija coordenadas de herramienta  .
2. Elija la velocidad de movimiento que desee, se recomienda la velocidad alta, pero trate siempre de hacer desplazamientos cortos.
3. Presione **{Switch de Desbloqueo}** y **{Teclas de ejes}**, hágalo con



, y observe que el brazo se moverá de acuerdo con el sistema coordenado de la herramienta.

Figura 2.9.2. Sistema cordenado del gripper²⁴.

²⁴ Fuente: Elaboración propia.



4. Pruebe con en coordenadas de herramienta, ¿Qué ocurre?, haga una descripción de la trayectoria de la herramienta con respecto a los ejes X, Y, Z.

5. Preguntas.

- a) ¿Qué ventajas presenta mover el robot en coordenadas de herramienta?

- b) Imagine que la herramienta del robot puede ser cualquier instrumento tipo industrial, ¿Qué aplicaciones podría encontrar para coordenadas de herramienta?

2.9.5. Conclusiones

2.10 PRÁCTICA DE LABORATORIO

2.11 ABRIR Y EJECUTAR UN TRABAJO

2.11.1 Objetivo general de aprendizaje

Abrir y ejecutar un trabajo desde el control de mando del brazo robótico.

2.11.2 Objetivos específicos de aprendizaje

- Abrir un trabajo o JOB.
- Ejecutar un trabajo o JOB.
- Analizar las líneas de código de un trabajo.
- Utilizar el modo PLAY del Teach Pendant.

2.11.3 Concepto importante: Programación del brazo robótico

El brazo robótico controlado por el ACL (controlador de lenguaje avanzado) tiene la opción de ser programado desde la interface del Teach Pendant, es decir que se pueden crear trabajos o JOBS con un lenguaje de programación propio de la línea de robótica industrial MOTOMAN. Este lenguaje funciona con líneas de código que se ejecutan de manera secuencial. El Teach Pendant puede guardar hasta 1280 posiciones, si se utiliza una memoria externa podrá efectivamente tener más capacidad.

2.11.4 Procedimiento

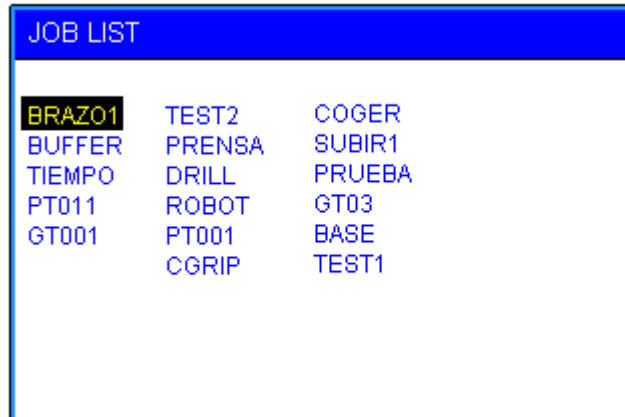
Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos para abrir y ejecutar un trabajo o JOB previamente desarrollado.

1. Para abrir un trabajo presione **{Main menu}** del teclado del Teach Pendant o de la pantalla táctil y presione la opción **{JOB}**, se desplegará una lista de opciones, elija **{SELECT JOB}**.

Figura 2.11.1. Lista de trabajos contenidos en el Teach Pendant²⁵.



2. Utilice el cursor del teclado del Teach Pendant para buscar en el *JOB LIST* el trabajo o JOB de nombre **BRAZO1**, selecciónelo con la tecla **{SELECT}**, a continuación se mostrará un conjunto de líneas de código como el siguiente:

```
000 NOP
001 MOVJ P0700 VJ=30.00
002 MOVJ P0701 VJ=30.00
003 MOVJ P0702 VJ=10.00
004 MOVJ P0703 VJ=10.00
005 MOVJ P0704 VJ=40.00
006 MOVJ P0705 VJ=50.00
007 END
```

²⁵ Fuente: Elaboración propia.

El espacio donde se visualizan las líneas de código se llama *JOB CONTENT*, este espacio muestra la secuencia en la que se ejecuta el trabajo.

3. Estando en el *JOB CONTENT* con el trabajo **BRAZO1**, cambie de modo con el **{Switch modal}** a **PLAY**.
4. Luego presione el botón **{SERVO ON READY}** de la botonera del Teach Pendant, este debe encender de manera constante un indicador luminoso de color verde acompañado de un pitido agudo.
5. Aléjese por lo menos 1 metro del brazo robótico y presione el botón verde **{START}** que se encuentra en la parte superior del Teach Pendant. Atención, el brazo comenzará a moverse.
6. Presione nuevamente **{START}** y observe el Display general, cada línea de código se ejecuta de manera secuencial hasta llegar a *END*.

2.11.5 Conclusiones

2.12 PRÁCTICA DE LABORATORIO

2.13 ENSEÑAR POSICIONES

2.13.1 Objetivo general de aprendizaje

Enseñar posiciones del brazo robótico mediante el manejo del Teach Pendant.

2.13.2 Objetivos específicos de aprendizaje

- Conocer los niveles de seguridad del Teach Pendant.
- Ejecutar una posición previamente guardada.
- Guardar una nueva posición en el Teach Pendant.

2.13.3 Concepto importante

Una línea de código puede representar el arribo a un punto específico o una posición del robot, por ejemplo: **MOVJ P0700**, es una instrucción que hace que el robot se mueva a la posición número **0700** el cual contiene los pulsos programados de cada eje del brazo robótico. Sin importar donde se encuentre actualmente el brazo él tratará de llegar a dicho punto, el Teach Pendant tiene una memoria interna para guardar varios PAGE o posiciones de robot, con un rango que va desde 0000 hasta 1279.

2.13.4 Procedimiento




Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos.

1. Utilice el Teach Pendant para llevar el robot a Home Position.

2. El Teach Pendant maneja niveles de seguridad para proteger trabajos y posiciones, esos niveles de seguridad son:

- Modo de operación (OPERATING MODE). 
- Modo para editar (EDITING MODE). 
- Modo privilegiado (MANAGEMENT MODE). 

Las llaves que aparecen en el área de estatus del Display general representan los niveles de seguridad.


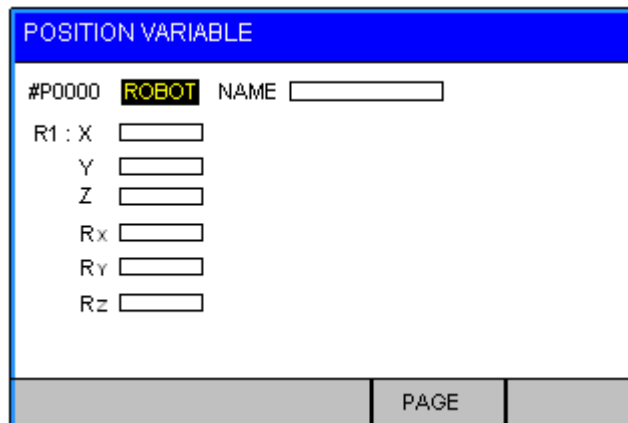

3. Presione **{Main menu}** luego elija la primera opción del menú, **{SYSTEM INFO}** se desplegarán otras opciones, escoja **{SECURITY}**. En el Display general aparece un pequeño cuadro que muestra el tipo de seguridad actual, presione **{SELECT}**, se mostrará los tres niveles de seguridad, con el cursor y **{SELECT}** escoja **EDITING MODE**, este modo se activa con una contraseña el cual es 00000000, la puede escribir con las teclas numéricas, posteriormente presione **{ENTER}** parte inferior derecha . Deberá aparecer las dos llaves en el área de estatus, si no es así repita el proceso. Este modo de seguridad permite crear posiciones y editar un JOB o trabajo.
4. Presione **{Main menu}**, en las opciones de menú elija **{VARIABLE}** y en la nueva lista de opciones, escoja **{POSITION(ROBOT)}**. Deberá mostrarse el siguiente cuadro:

Figura 2.13.1. Cuadro de posiciones del Teach Pendant²⁶.

²⁶ Fuente: Elaboración propia.



5. Presione **{PAGE}**, parte inferior derecha de la pantalla, esta le mostrará un pequeño cuadro, digite 703 con las teclas numéricas y presione **{ENTER}**, en el Display general aparecerán los pulsos de cada eje de la posición 703 previamente creada para este laboratorio.
6. Para ejecutar la posición 703 en el brazo robótico, mantenga presionado el **{Switch de desbloqueo}** y al mismo tiempo presione **{FWD}**  (parte derecha del teclado) sin soltar los botones el robot deberá desplazarse hasta ubicar los pulsos que muestra cada eje del brazo robótico en dicha posición.
7. Para enseñar una nueva posición presione **{PAGE}** y digite 706 luego presione **{ENTER}**. Aparecerá la posición 706 con sus ejes, en caso de no haber ningún valor guardado en los ejes ya sea en coordenadas joint o cartesianas estos estarán precedidos por unos asteriscos.
8. Usando **{Teclas de ejes}** del brazo robótico lleve el robot a una posición arbitraria.
9. Una vez que el robot se encuentre en la posición que desea guardar presione **{SELECT}** aparecerán las opciones, *PULSE*, *BASE*, *ROBOT*, *USER* y *TOOL* elija *PULSE* y presione **{SELECT}**.

10. De manera secuencial pulse **{Switch de desbloqueo}** y sin soltarlo presione la tecla **{MODIFY}** luego oprima **{ENTER}**, ambos botones se encuentran en la botonera del Teach Pendant en la parte inferior derecha. Observe los ejes, cada uno ya tiene un valor que indica un pulso y todos los valores constituyen una posición del brazo robótico.

11. Preguntas.

a) ¿Las posiciones como 700, 701, 702 o cualquiera, se consideran variables?

b) ¿Es posible utilizar un mismo PAGE para varios JOBS?

c) Explique el funcionamiento de la tecla **{FWD}**.

d) Explique el funcionamiento de la tecla **{MODIFY}**.

e) ¿Se puede grabar una posición sobre una posición existente? Inténtelo con la posición 706.

12. Como ejercicio utilice los PAGE 707, 708, 709 y 710 para guardar posiciones y ejecútelas después una por una. Si en el Teach Pendant encuentra que estos PAGE ya están ocupadas con coordenadas bórrelas.

2.13.5 Conclusiones

2.14 PRÁCTICA DE LABORATORIO

2.15 CREAR UN JOB

2.15.1 Objetivo general de aprendizaje

Crear un trabajo de manera básica con una secuencia de posiciones del brazo robótico utilizando el Teach Pendant.

2.15.2 Objetivos específicos de aprendizaje

- Enseñar posiciones con el método directo.
- Editar posiciones.
- Introducir líneas de código en un trabajo.
- Ejecutar paso a paso un trabajo.
- Utilizar la opción INTER LOCK para simulación.
- Diseñar un diagrama para describir un proceso.

2.15.3 Concepto importante: Método de enseñanza directa

Con el Teach Pendant es posible utilizar dos formas distintas para crear un trabajo o JOB, una de ellas es grabando directamente las posiciones en el programa el cual resulta muy fácil y la otra es implementando el PAGE como variable. A la primera forma de grabar posiciones se le llamará método de enseñanza directa y al otro, método de enseñanza por variables.

2.15.4 Procedimiento

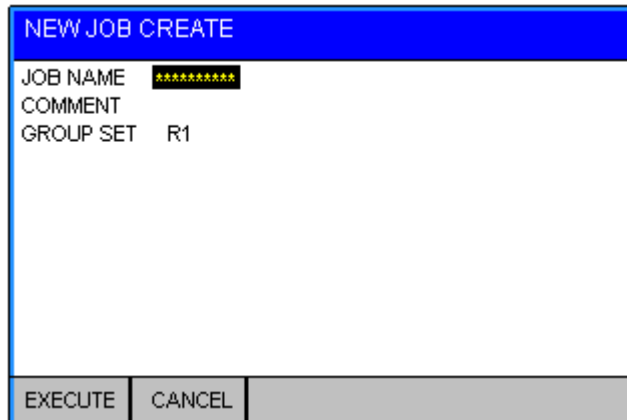
Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos.

1. Con el Teach Pendant en mano y un nivel de seguridad de EDITING MODE diríjase a **{Main menu}**, luego entre a **{JOB}**, en las opciones que se despliegan elija **{CREATE NEW JOB}** observará algo como:

Figura 2.15.1. Crear un nuevo trabajo²⁷.



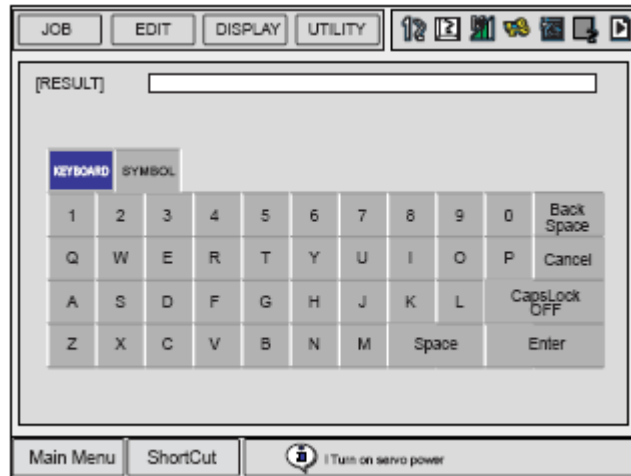
NEW JOB CREATE	
JOB NAME	*****
COMMENT	
GROUP SET	R1
EXECUTE	CANCEL

2. Presione **{SELECT}**, enseguida aparecerá un cuadro con apariencia de teclado en el Display general donde podrá escribir el nombre del trabajo o JOB. Para este laboratorio escriba “*estudi1*”.

Figura. 2.15.2. Interfaz de teclado del Teach Pendant²⁸.

²⁷ Fuente: Elaboración propia.

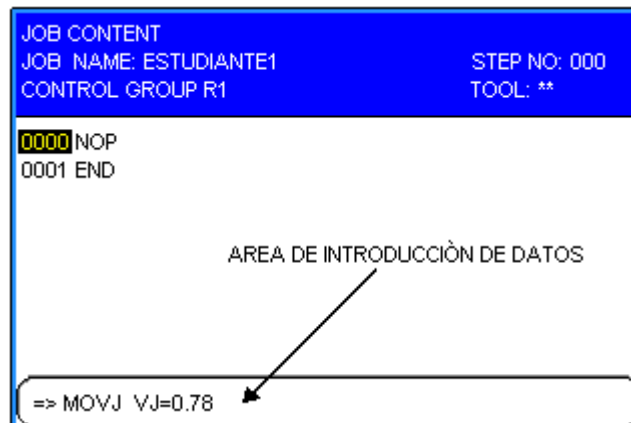
²⁸ Fuente: Manual de operador de NXC100 numero RE-CSO-A032



Una vez escrito el nombre digite **{ENTER}**, puede hacerlo desde el teclado o desde el teclado del Teach Pendant.

3. Presione la opción **{EXECUTE}** que se encuentra en la parte inferior izquierda del Display general.

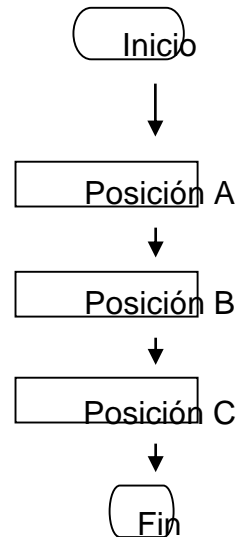
Figura. 2.15.3. Interfaz de programación²⁹.



Antes de comenzar a programar es necesario saber y tener claridad de lo que se quiere hacer.

²⁹ Fuente: Elaboración propia.

4. Se desea realizar la siguiente secuencia de posiciones del robot.



Mueva con **{Teclas de ejes}** el robot a una posición inicial cualquiera el cual está representada por Posición A. Ya definido el punto asegúrese que el cursor se encuentra señalando la primera línea de código 0000 NOP y presione sin soltar **{Switch de desbloqueo}** luego **{INSERT}** (parte inferior derecha de la botonera) y después **{ENTER}**. Enseguida se inserta o agrega un línea de código **0001 MOVJ VJ=0.78** esta línea es entonces la posición A, MOVJ representa el tipo de movimiento, en este caso Joint y VJ es la velocidad que por defecto es de 0.78 y el cual es un poco lenta, pero se puede cambiar colocando el cursor sobre la línea que se acabó de insertar es decir, **MOVJ VJ=0.78**, presione **{SELECT}** enseguida el cursor se posiciona en el área de introducción de datos, lleve el cursor donde dice 0.78 y presione **{SELECT}**, digite el número 40 y presione **{ENTER}** dos veces. Observe que la velocidad de la línea de código se modificó.

5. Escoja otros dos puntos para el robot e inserte la posición B y C repitiendo el paso 4.

6. Ejecute la primera línea de código ubicando el cursor en 0001, presione sin soltar **{Switch de desbloqueo}** y después **{FRD}**. Haga lo mismos con la línea 0002 y 0003.
7. Con el cursor vallase a la línea 0000 NOP, esta instrucción hace el papel de inicio en un programa cualquiera. Presione sin soltar **{Switch de desbloqueo}** luego sin soltar también oprima **{INTER LOCK}** (parte inferior izquierda de la botonera) y presione **{FWD}**, mantenga estos tres botones presionados mientras observa el robot moviéndose, esto es simplemente una simulación antes de ejecutar todo el programa en modo **PLAY**.
8. Cambie al modo **PLAY** y presione **{SERVO ON READY}** después **{START}**.
9. Preguntas.

a) ¿Cuál es la diferencia de utilizar 5 posiciones en lugar de 2 para llegar a un punto específico?

b) Explique el funcionamiento de **{TEST START}**.

c) ¿Cuál es la diferencia en utilizar el modo **PLAY** y la forma **{INTER LOCK}** mas **{TEST START}**?

10. Ejercicio.

a) Diseñe un programa o JOB con 5 posiciones, haga el diagrama de flujo en el cuadro de abajo. Las condiciones son:

- La primera posición debe ser home position.
- La herramienta del brazo robótico debe llegar al RACK 2 en el INDEX 1.
- La última posición debe tener una velocidad de 3.

Ejecute y observe el trabajo de ejemplo BRAZO2 en el Teach pendant el cual solo tiene dos posiciones para llegar al RACK2 INDEX 2.

b) Dibuje el diagrama para las cinco posiciones como en el paso 4.



2.15.5 Conclusión

2.16 PRÁCTICA DE LABORATORIO

2.17 CREAR UN JOB UTILIZANDO VARIABLES

2.17.1 Objetivo general de aprendizaje

Crear un trabajo utilizando posiciones de brazo robótico como variables mediante el Teach Pendant.

2.17.2 Objetivos específicos de aprendizaje

- Utilizar PAGE en un trabajo.
- Ingresar al repertorio de instrucciones del Teach Pendant.
- Utilizar movimiento lineal en un trabajo.
- Utilizar movimiento circular en un trabajo.
- Editar la velocidad de movimiento de una línea de código.
- Diseñar diagrama de procesos.

2.17.3 Concepto importante: Variable o PAGE

En el laboratorio 2.14 se hizo la forma directa de guardar posiciones en el mismo espacio de programación, el único inconveniente es que esas posiciones no se pueden usar en otros trabajos o JOB, es aquí donde cobra importancia trabajar con variables o PAGEs, debido a que permiten ser utilizadas en varios trabajos.

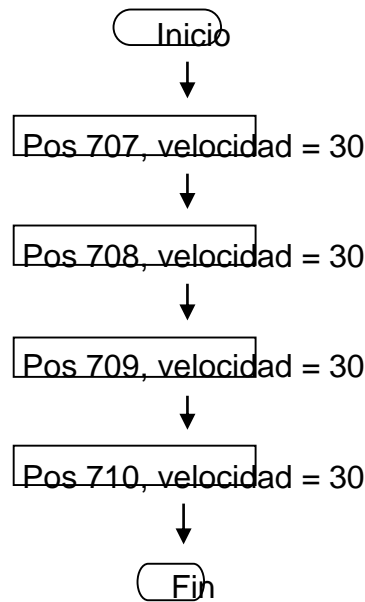
2.17.4 Procedimiento

En este laboratorio se indicará paso por paso la realización de un JOB con variables o PAGE. Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos.

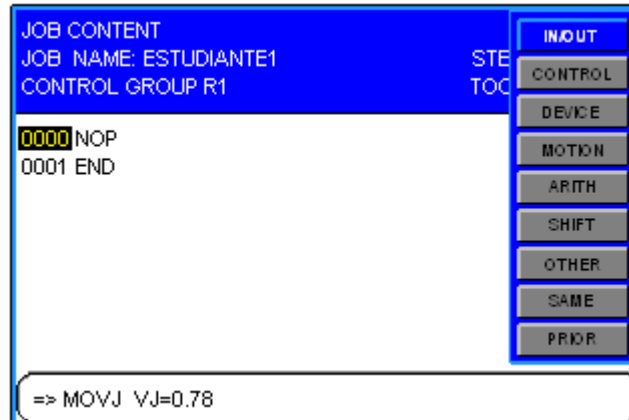
1. Entre a **{Main menu}**, **{VARIABLE}**, **{POSITION(ROBOT)}** y utilice los PAGE 707 al 710 para guardar posiciones de manera arbitraria pero respetando los parámetros de seguridad del robot.
2. El algoritmo a realizar será:



Asegúrese de estar en un nivel de seguridad de EDITING MODE contraseña 00000000.

3. Diríjase a menú y luego a **{JOB}**, **{CREATE NEW JOB}**. Como nombre para el JOB utilice “*estudi#*” y oprima **{EXECUTE}**.
4. Estando en el JOB CONTENT presione la tecla **{INFORM LIST}** (parte inferior izquierda del teclado), esta se activará con un indicador luminoso color verde y se mostrará un repertorio de instrucciones o comandos en la parte derecha del display general.

Figura. 2.17.1. Contenido del trabajo mostrando la lista de comandos de programación³⁰.



5. Del repertorio de instrucciones ingrese a **{MOTION}**, en seguida se muestra las instrucciones que tienen que ver con movimiento:

MOVJ = Movimiento en Joint

MOVL = Movimiento lineal

MOVC = Movimiento circular

MOVS = Movimiento espiral

6. Utilice movimiento Joint, presione **{MOVJ}**, en la zona de edición debe aparecer **MOVJ P0000 VJ**, el valor de la velocidad en Joint es de 0 a 100. Con el cursor seleccione *P0000* y oprima **{SELECT}**, escriba 707, el PAGE para la primera posición y presione **{ENTER}**. Luego seleccione *VJ = #* y presione **{SELECT}**, digite un valor moderado puede ser 30, oprima **{ENTER}**, nuevamente oprima **{ENTER}**, la línea de código para la posición 707 se tuvo que haber insertado en el contenido de trabajo.

³⁰ Fuente: Elaboración propia.

7. Vuelva a seleccionar **{MOVJ}** de la lista de comandos, en la zona de edición aparecerán los datos de la línea que acabó de insertar, modifíquela para 708 y repita el mismo proceso para 709 y 710. El programa final deberá ser como el siguiente:

```
0000 NOP
0001 MOVJ      P0707      VJ = 30.00
0002 MOVJ      P0708      VJ = 30.00
0003 MOVJ      P0709      VJ = 30.00
0004 MOVJ      P0710      VJ = 30.00
0005 END
```

Para desactivar la lista de instrucciones oprima **{INFORM LIST}**.

8. Haga la simulación con **{INTER LOCK}** y **{TEST START}**.
9. Ejecute el trabajo en el modo PLAY.
10. Preguntas.
- a) Compare los dos métodos para hacer una secuencia de posiciones.
- _____
- _____
- _____
- b) Abra el trabajo de nombre CIRCULAR en el JOB LIST ejecute y analice los movimientos, saque sus conclusiones.
- _____
- _____
- _____

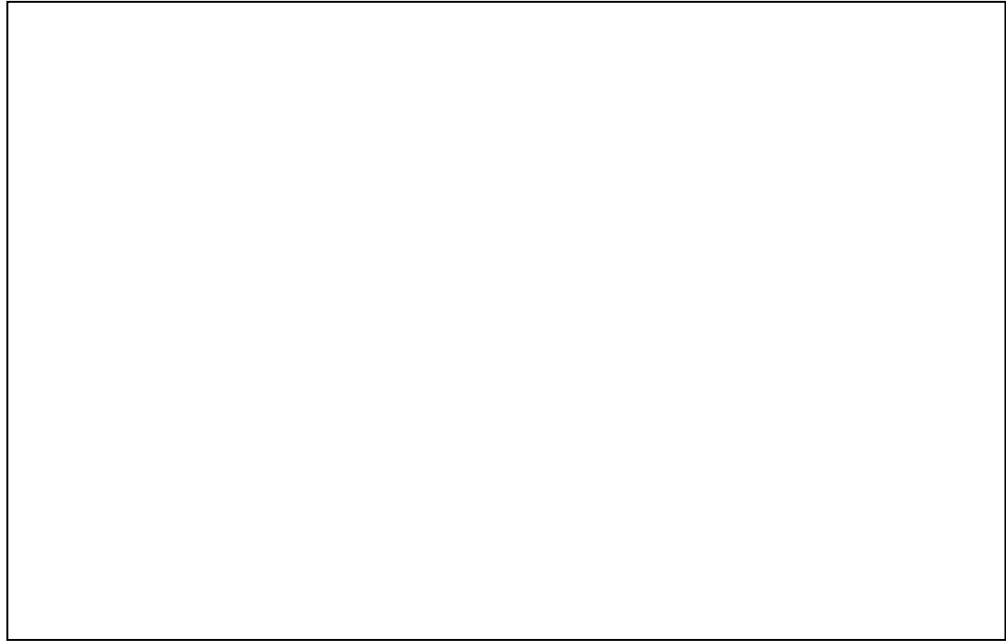
- c) Abra el trabajo de nombre LINEAL en el JOB LIST ejecute y analice los movimientos, saque sus conclusiones.

- d) Abra el trabajo de nombre ESPIRAL en el JOB LIST ejecute y analice los movimientos, saque sus conclusiones.

11. Ejercicio.

- a) Utilice 5 PAGE de 711 al 715 para hacer un JOB, las condiciones son:
- El primer PAGE deber ser la posición de home position.
 - 712 y 713 con velocidades de 40 con movimiento en joint.
 - 714 velocidad de 60 con movimiento circular.
 - la herramienta del brazo robótico debe llegar cerca de la prensa con la última posición, 715 lineal y velocidad 15.

b) Dibuje el diagrama de flujo del ejercicio 11.



2.17.5 Conclusión

2.18 PRÁCTICA DE LABORATORIO

2.19 SALTO Y ETIQUETA

2.19.1 Objetivo general de aprendizaje

Manipular las instrucciones JUMP y LABEL en un trabajo.

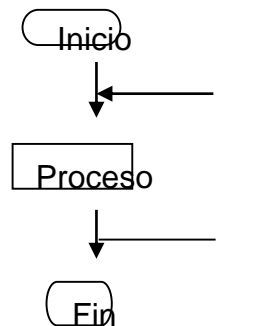
2.19.2 Objetivos específicos de aprendizaje

- Comprender la función del comando JUMP.
- Entender la importancia del comando LABEL en la secuencia de un programa.

2.19.3 Concepto importante: Comandos de control JUMP y LABEL

JUMP y LABEL, son instrucciones o códigos muy usados en programación que permiten hacer saltos en líneas de códigos etiquetadas. La forma de usar JUMP es estableciendo una etiqueta o LABEL en la línea de código donde se quiere que el salto.

Ejemplo:



Este ejemplo muestra la aplicación de un JUMP, cuando el proceso va a finalizar ocurre un salto hasta una etiqueta después de inicio lo que provoca que el proceso

se vuelve a repetir. Suponga que el proceso son tres posiciones de robot, entonces el salto se representaría en el código del Teach Pendant como:

```
0000 NOP
0001 SALTO
0002 MOVJ P0700      VJ=30
0003 MOVJ P0701      VJ=30
0004 MOVJ P0702      VJ=30
0005 JUMP SALTO
0006 END
```

2.19.4 Procedimiento

Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos de este laboratorio.

1. Abra el trabajo SALTO en el JOB LIST del Teach Pendant, ejecútelo y observe la función del JUMP.
2. Detenga la ejecución del trabajo SALTO con el botón **{HOLD}**, parte superior derecha del Teach Pendant.
3. Haga un JOB, sin utilizar PAGE, es decir cree las posiciones de forma directa, por lo menos 4 posiciones, utilice como nombre “*estudi#*”.

4. En el JOB CONTENT de “*estudi#*”, ubique el cursor en la primera línea es decir, 0000 y a continuación presione el botón **{INFORM LIST}** de la botonera, en la lista de comandos elija la opción **{CONTROL}** y en ella **{LABEL}**, en el área de introducción de datos aparecerá =>*LABEL.
5. seleccione la palabra LABEL con el cursor y oprima **{SELECT}** enseguida se activará el teclado, escriba INICIO y presione **{ENTER}**, note que en el área de introducción de datos el nombre cambia a INICIO, para insertar esta nueva instrucción oprima el botón **{INSERT}** y luego **{ENTER}**, la etiqueta se tuvo que haber agregado con éxito con el nombre de INICIO.
6. Ubique el cursor en la penúltima línea de código de “*estudi#*”, abra la lista de comandos y entre a **{CONTROL}**, elija **{JUMP}**, en el área de introducción de datos debe haber algo como, JUMP *LABEL, ubique el cursor donde dice LABEL y oprima **{SELECT}**, por medio de la interface de teclado debe digitar la palabra INICIO y luego oprimir **{ENTER}**. Para insertar la instrucción oprima el botón **{INSERT}** y luego **{ENTER}**.
7. Haga la simulación oprimiendo sin soltar **{Switch de desbloqueo}** luego **{INTER LOCK}** y después **{TEST START}**.
8. Ejecute el trabajo en modo **PLAY**.
9. Preguntas.
 - a) ¿Qué ocurre si el nombre de la etiqueta o LABEL no coincide con el nombre del salto o JUMP?

b) ¿Cuál es el mensaje de error arrojado por el Teach Pendant?

2.19.5 Conclusiones.

2.20 PRÁCTICA DE LABORATORIO

2.21 SUBRUTINAS

2.21.1 Objetivo general de aprendizaje

Utilizar la instrucción que permite generar subrutinas en un trabajo o JOB.

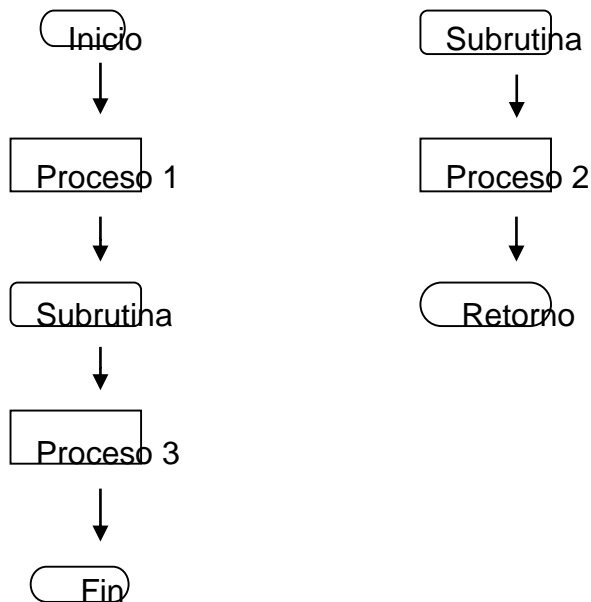
2.21.2 Objetivos específicos de aprendizaje

- Entender el significado de subrutina.
- Comprender el funcionamiento de la instrucción CALL.

2.21.3 Concepto importante: Instrucción CALL

Esta instrucción sirve como en la mayoría de los lenguajes de programación para hacer un llamado a una subrutina dentro de un programa, en el Teach Pendant se hace exactamente lo mismo. El objetivo fundamental de esta instrucción es llamar a otros trabajos del JOB LIST.

Ejemplo:



En el ejemplo del diagrama de flujo se observa el proceso 2 dentro de una subrutina, cuando esta termina hace el retorno al programa principal.

2.21.4 Procedimiento

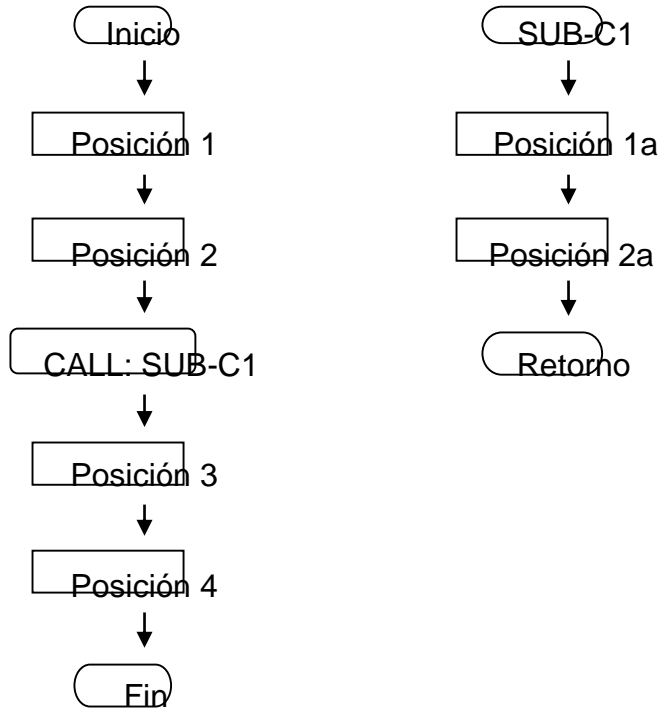
Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos del laboratorio.

1. Ejecute el trabajo de nombre CALL que se encuentra en el JOB LIST y observe el comportamiento del JOB CONTENT y JOB NAME.
2. Haga un JOB sin utilizar PAGE, es decir cree las posiciones de forma directa, por lo menos 4 posiciones, utilice como nombre “*estudi#*”.
3. Haga otro JOB de 2 posiciones, utilice como nombre SUB-estudi#.
4. Abra el primer JOB que hizo y ubique el cursor entre la posición 2 y 3, luego oprima **{INFORM LIST}** para abrir el repertorio de instrucciones, abra **{CONTROL}** y seleccione **{CALL}**, en el área de introducción de datos debe aparecer CALL JOB: JOB, ubicando el cursor en JOB y presionando **{SELECT}** entrará al JOB NAME, busque el segundo trabajo que hizo y presione **{SELECT}**.

5. Inserte la instrucción, el trabajo deberá tener el siguiente esquema:



6. Haga la simulación oprimiendo sin soltar **{Switch de desbloqueo}** luego **{INTER LOCK}** y **{TEST START}**.

7. Preguntas.

a) ¿Cuáles son las ventajas de utilizar subrutinas?

b) Explique qué beneficios brinda la instrucción CALL dentro de la secuencia de un programa.

- c) Las instrucciones JUMP y CALL son comandos que alteran notablemente la secuencia de un programa. ¿Cuál es la diferencia más significativa entre estos dos comandos?

2.21.5 Conclusiones

2.22 PRÁCTICA DE LABORATORIO

2.23 TEMPORIZACIÓN

2.23.1 Objetivo general de aprendizaje

Utilizar la instrucción de temporización en un trabajo.

2.23.2 Objetivos específicos de aprendizaje

- Comprender el funcionamiento de comando TIMER.
- Editar valores de temporización.

2.23.3 Concepto importante: Instrucción TIMER

Este comando es bastante útil gracias a que permite hacer temporizaciones a un programa o cualquier proceso. Para el brazo robótico esta instrucción tiene relevancia debido a que las pausas de tiempo entre posiciones son importantes para las aplicaciones de este sistema.

Ejemplo:

```
0000 NOP
0001 MOVJ      VJ=10.00
0002 TIMER    T=3.00
0003 MOVJ      VJ=10.00
0004 END
```

Entre la línea de código 0001 y 0003 hay un tiempo de espera de 3 segundos.

2.23.4 Procedimiento

Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

Siga los siguientes pasos.

1. Abra el trabajo en el LIST JOB de nombre TIMER y observe el comportamiento del programa.
2. Haga un JOB sin utilizar PAGE, es decir cree las posiciones de forma directa, por lo menos 4 posiciones, utilice como nombre TIMER1.
3. Ubique el cursor en la posición 0001, presione **{INFORM LIST}** en el repertorio de instrucciones entre a **{CONTROL}** y seleccione **{TIMER}**. En el área de introducción de datos deberá aparecer TIMER T=1.00, el valor viene por defecto, ubique el cursor sobre T=1.00 y oprima el botón **{SELECT}**, digite 3 como valor, esto representará 3 segundos, oprima **{ENTER}**. Para insertar esta nueva línea de código oprima **{INSERT}** y **{ENTER}**.
4. Ubique el cursor en la línea de código 0003, y repita el proceso del punto 3, de igual manera con una temporización de 3 segundos. El programa debe tener una temporización entre la línea 0001 y 0003 y otra entre la 0003 y 0005.
5. Haga la simulación oprimiendo sin soltar **{Switch de desbloqueo}** luego **{INTER LOCK}** y **{TEST START}**.
6. Preguntas

a) ¿Por qué es importante la utilización de temporizaciones?

b) ¿En qué tipo de aplicaciones resulta conveniente usar el comando TIMER?

2.23.5 Conclusión

2.24 PRÁCTICA DE LABORATORIO

2.25 USO DE LOS COMANDOS CALL, JUMP Y TIMER

2.25.1 Objetivo general de aprendizaje

Utilizar los diferentes tipos de instrucciones de control para realizar un trabajo mediante el Teach Pendant.

2.25.2 Objetivos específicos de aprendizaje

- Identificar algunas partes de la celda de manufactura.
- Utilizar la instrucción CALL.
- Utilizar la instrucción TIMER.
- Utilizar la instrucción JUMP.
- Hacer la descripción de un proceso.

2.25.3 Concepto importante: Utilización de instrucciones de control

Para los trabajos de las aplicaciones del brazo robótico se utilizan las instrucciones de control debido a que la programación debe ser más elaborada y robusta, esto permite tener un manejo adecuado de los procesos que se realizan.

2.25.4 Procedimiento

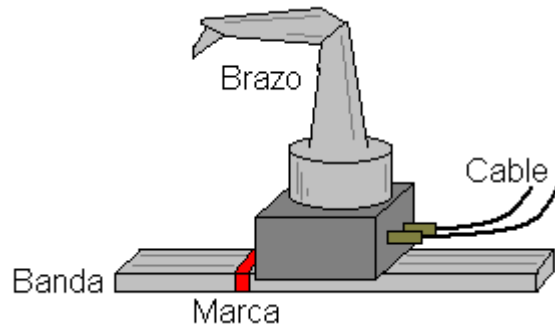
Para este laboratorio debe contar con:

- Controlador NXC100.
- Control de mando del NXC100.

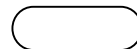
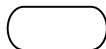
Siga los siguientes pasos.

1. Asegúrese que el brazo robótico se encuentre posicionado en la marca de banda deslizante, si no es así consulte al tutor encargado.

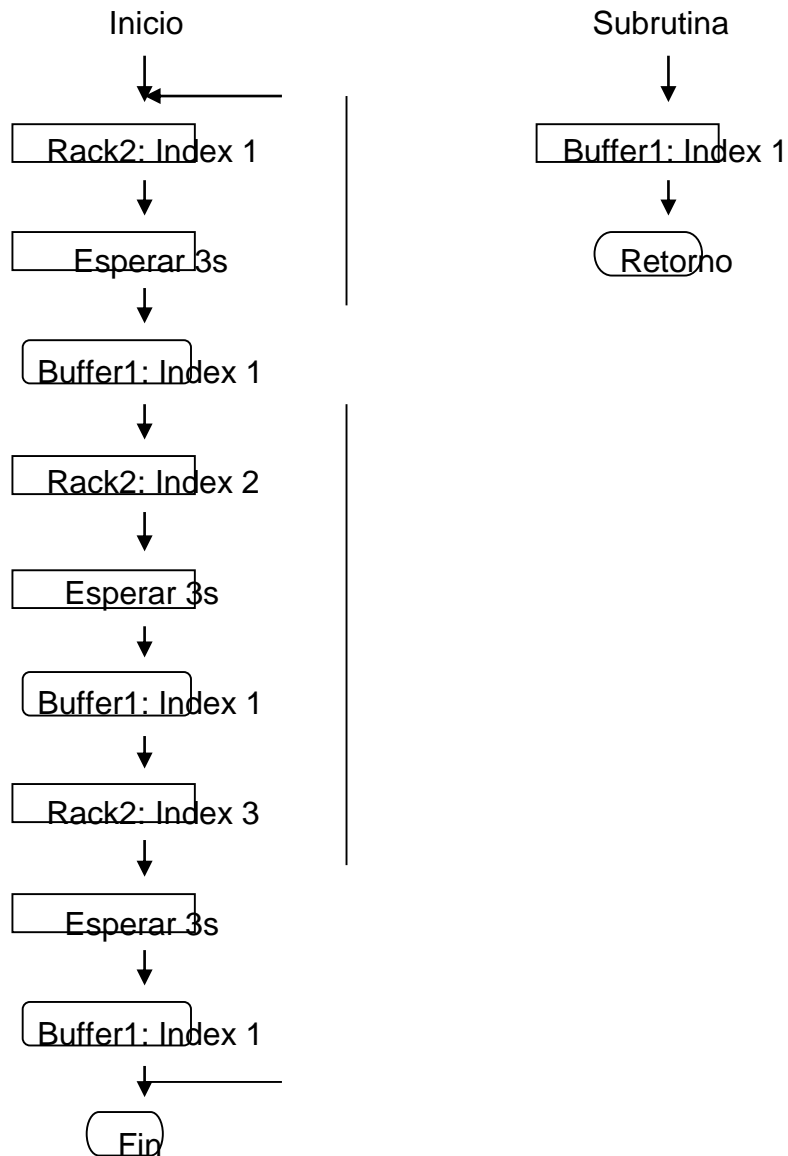
Figura. 2.25.1 Marca de banda deslizante³¹.



2. Abra el JOB de nombre BRAZO3 y antes de ejecutarlo observe y comprenda el diagrama de flujo:



³¹ Fuente: Elaboración propia.



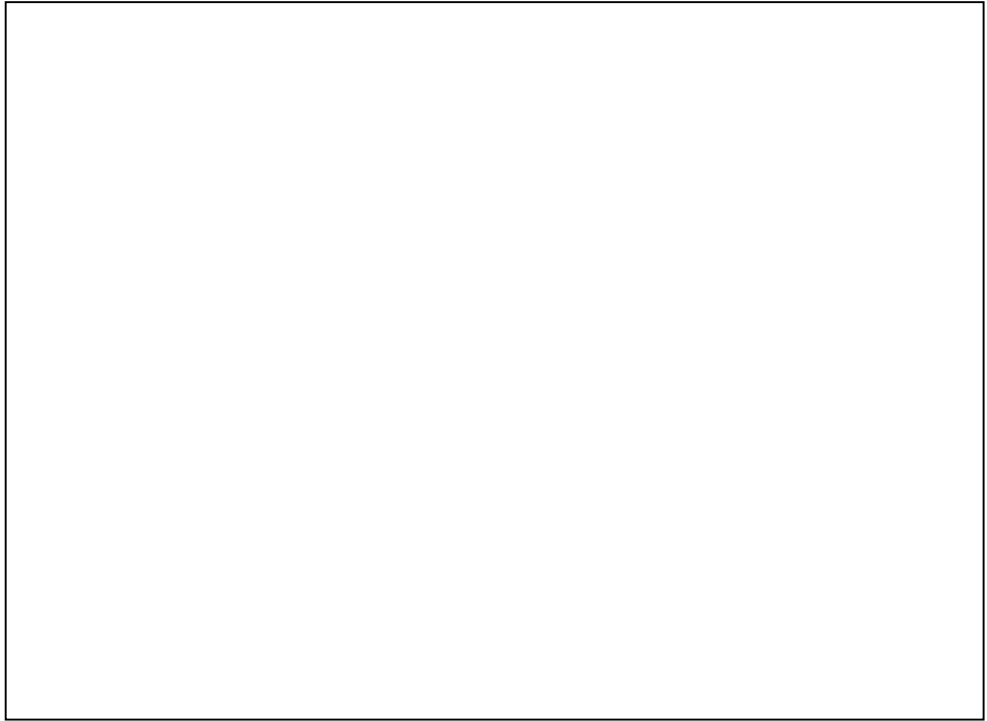
3. Ejecute el JOB paso por paso con **{Switch de Desbloqueo}** y **{FWD}**, observe bien cada posición.
4. Ejecute el JOB con el modo **{PLAY}**, presione **{SERVO ON READY}** y luego **{START}**. Para detener el proceso oprima el botón **{HOLD}** que se encuentra al lado del botón **{START}**.
5. Ejercicios.

a) Haga una descripción del JOB BRAZO3.

b) Se pretende diseñar un proceso donde la herramienta va a llegar al INDEX1, INDEX2 e INDEX3 del RACK1, pero después de llegar a un INDEX del RACK1 debe hacer un arribo al INDEX2 del BUFFER1 para después ir al INDEX siguiente del RACK1. Las consideraciones son:

- Utilice el comando TIMER para hacer temporizaciones de 3 a 5 segundos cada vez que la herramienta del brazo llegue a un INDEX.
- Utilice el comando CALL para evitar repeticiones de código cada vez que la herramienta llegue a INDEX2 del BUFFER1.
- Utilice un JUMP para que el proceso sea repetitivo.
- Utilice velocidades no tan altas y haga combinaciones con los tipos de movimientos.

c) Dibuje el diagrama de flujo.



2.25.5 Conclusiones

2.26 PRÁCTICA DE LABORATORIO

2.27 PERIFÉRICOS DEL NXC100

2.27.1 Objetivo general de aprendizaje

Manejar los periféricos de la Celda de Manufactura mediante el Teach Pendant.

2.27.2 Objetivo específico de aprendizaje

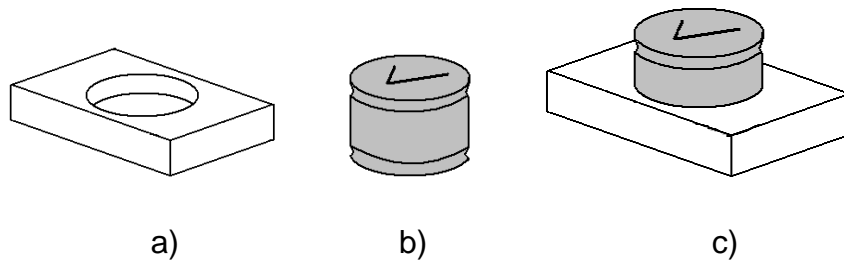
- Identificar los tipos de materia prima de la celda de manufactura.
- Activar el Gripper de forma manual con el Teach Pendant.
- Activar la prensa de forma manual con el Teach Pendant.
- Utilizar JOB para abrir o cerrar la prensa y el Gripper.
- Utilizar el RACK de la celda de manufactura.
- Diseñar una aplicación con el RACK, materia prima y periféricos.

2.27.3 Concepto importante: Gripper y Prensa

El Gripper y la prensa son periféricos controlados por el NXC100, la función principal del Gripper es capturar plantillas y materia prima, más adelante se mostrará otras funciones de esta herramienta. La prensa cumple la función de sostener parte de la materia prima para ser ensamblada, en las grandes industrias de ensamblaje y producción es importante fijar siempre las partes de ensamblado para garantizar la precisión. Estos periféricos funcionan con un sistema neumático, por lo que es necesario tener activado el compresor de aire con una presión de 80 PSI.

Figura 2.27.1. a) Materia prima b) Materia prima c) Producto terminado³².

³² Fuente: Elaboración propia.



En la celda de manufactura del laboratorio de procesos industriales se tienen estas piezas de materia prima, cuya utilización es para fines pedagógicos y el cual representan cualquier tipo de material listo para pasar por un proceso de ensamblado.

2.27.4 Procedimiento

Para este laboratorio debe contar con:

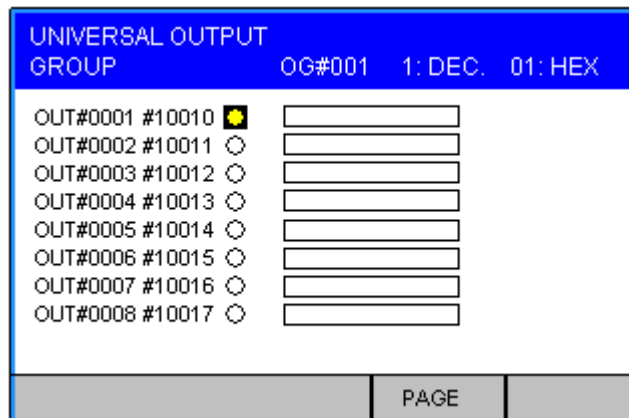
- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Prensa.
- RACK.
- Materia prima.

Siga los pasos de este laboratorio para utilizar los periféricos que tiene el NXC100 usando el Teach Pendant.

1. Asegúrese que el sistema neumático este funcionando correctamente y que el manómetro marque 80 psi.
2. Asegúrese que el brazo robótico se encuentre posicionado en la marca de la banda deslizante, si no es así consulte al tutor encargado.

3. Con el Teach Pendant en mano presione **{Main menu}**, luego **{IN/OUT}** y seleccione **{UNIVERSAL OUTPUT}**.

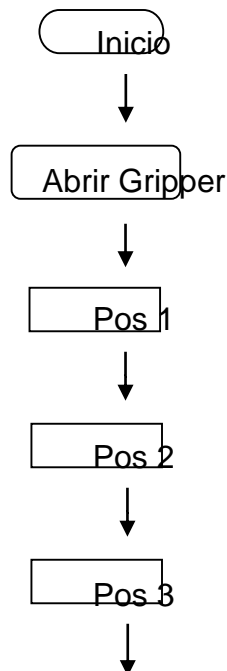
Figura 2.27.2. Interfaz de dispositivos controlados por el NXC100³³.

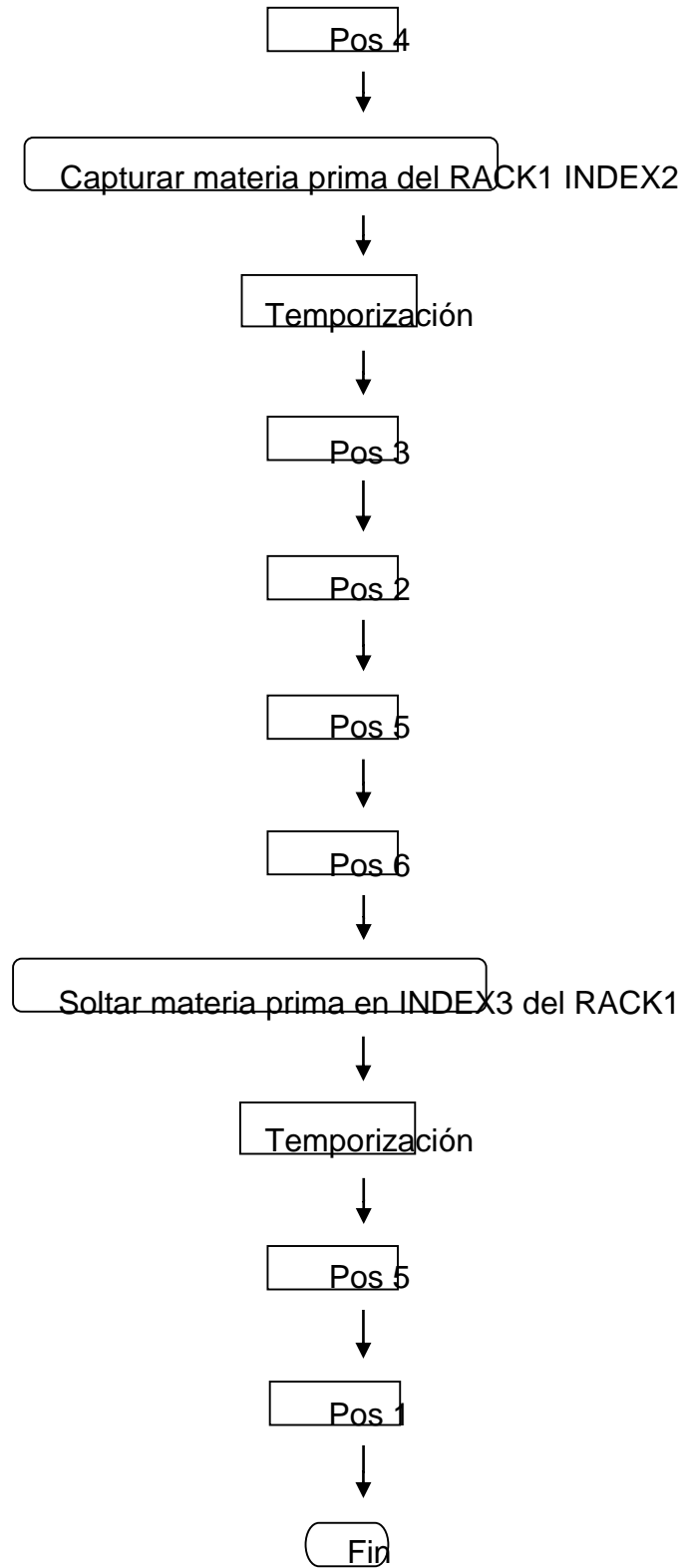


4. Con el cursor váyase al primer círculo de la salida OUT#0001 como lo muestra la figura 2.27.2 y mantenga hundido **{Switch de Desbloqueo}** sin soltarlo presione **{INTER LOCK}** de la botonera y después **{SELECT}**, el gripper se abrirá o se cerrará dependiendo del estado actual en que se encuentre.
5. Repita el paso 3 ahora en la salida OUT#0002, esta salida controla la prensa, el cual se abrirá o cerrará.
6. Abra el trabajo del JOB LIST de nombre CGRIP y ejecútelo con **{Switch de Desbloqueo}** mas **{INTER LOCK}** y **{TEST START}**. El Gripper deberá cerrarse.

³³ Fuente: Elaboración propia.

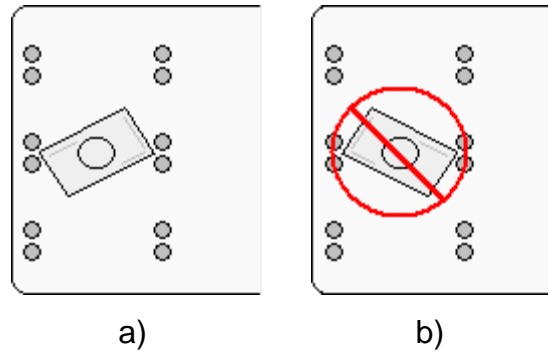
7. Abra el trabajo del JOB LIST de nombre OGRIP y ejecútelo con **{Switch de Desbloqueo}** mas **{INTER LOCK}** y **{TEST START}**. El Gripper deberá abrirse.
8. Abra el trabajo del JOB LIST de nombre CVICE ejecútelo con **{Switch de Desbloqueo}** mas **{INTER LOCK}** y **{TEST START}**. La prensa deberá cerrarse.
9. Abra el trabajo del JOB LIST de nombre OVICE ejecútelo con **{Switch de Desbloqueo}** mas **{INTER LOCK}** y **{TEST START}**. La prensa deberá abrirse.
10. Abra el trabajo del JOB LIST de nombre GRIPER1 y ejecútelo en el modo PLAY con **{SERVO ON READY}** y **{START}**. El Gripper deberá abrirse y cerrarse continuamente, observe las líneas de código. Para detener el proceso oprima el botón **{HOLD}** y vuelva al modo TEACH.
11. Analice el siguiente proceso:



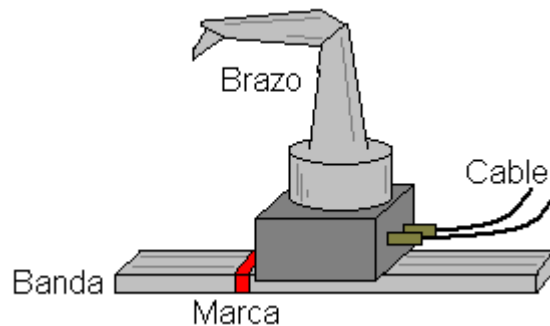


12. Coloque una materia prima como la que se muestra en la figura 2.27.1 a) en el Index2 del Rack1. Tenga mucho cuidado con la ubicación.

Figura 2.27.3. a) Vista desde arriba del Rack1 mostrando la correcta ubicación de la materia prima b) Forma incorrecta de ubicar la materia prima³⁴.



13. Asegure el robot en la marca de banda deslizante.



14. Abra el trabajo del JOB LIST de nombre GRIPER2, luego ejecútelo en el modo PLAY con **{SERVO ON READY}** y **{START}** y compare las líneas de código con el diagrama de flujo del paso 11. Note que para abrir y cerrar el Gripper se hace un llamado con el comando CALL a CGRIP para cerrar u OGRIP para abrir.

³⁴ Fuente: Elaboración propia.

15. Preguntas.

a) ¿Cuáles fueron los comandos que se utilizaron para el JOB GRIPER2?

b) ¿Cuál es el comando que se utiliza para activar un periférico?

16. Ejercicios.

c) Coloque materia prima en el INDEX1 del RACK2 (cilindro de aluminio).

d) Haga un JOB con el Teach pendant para capturar el cilindro en INDEX1 y llevarlo hasta el INDEX2 del RACK2, utilice el comando CALL para llamar al gripper. Grabe las posiciones de manera directa.

e) Abra el JOB de nombre GRIPER3 ejecútelo y analícelo, haga una descripción completa de él.

2.27.5 Conclusiones

CAPÍTULO 3

3. SCORBASE

3.1. INTRODUCCIÓN DEL CAPÍTULO

Este capítulo se enfocará principalmente al manejo del software SCORBASE, el cual permite interactuar con el controlador lógico programable que controla la banda que transporta el brazo robótico. Éste software presenta algunas posibilidades las cuales son:

- Control y supervisión en tiempo real del estado de las 8 entradas y salidas que dispone el controlador lógico programable. Se puede asignar nombres a cada entrada y salida. SCORBASE ofrece 4 entradas analógicas y 2 salidas analógicas.
- Control de la base deslizante lineal para el brazo.
- Por defecto guarda hasta 100 posiciones y puede manejar 1000 líneas de programa activas.
- Programación de interrupciones para el cambio del estado de las entradas.
- Comunicación con el controlador NXC100 para el control del brazo robótico MOTOMAN HP3.

Este software no programa directamente las posiciones del brazo robótico MOTOMAN HP3, esto se hace únicamente con el Teach Pendant, con el

SCORBASE simplemente se hace un llamado a los JOBS que se crean en el control de mando del robot.

3.2 ENTRAR A SCORBASE

Para entrar a SCORBASE FOR NXC100 debe asegurarse que el PLC y el controlador NXC100 estén encendidos, luego asegúrese que la llave de seguridad de hardware se encuentre conectada en el PC o estación de trabajo SW1, siga los siguientes pasos:


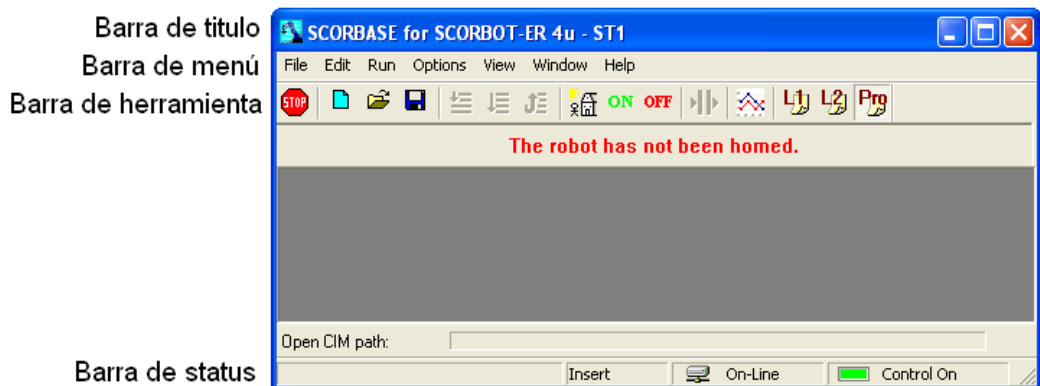
- En el escritorio abra el icono SCORBASE FOR NXC100  , si aparece un mensaje que diga; ¿desea pasar a control on? Presione **aceptar**, si aparecen otros mensajes diferentes oprima **aceptar**.

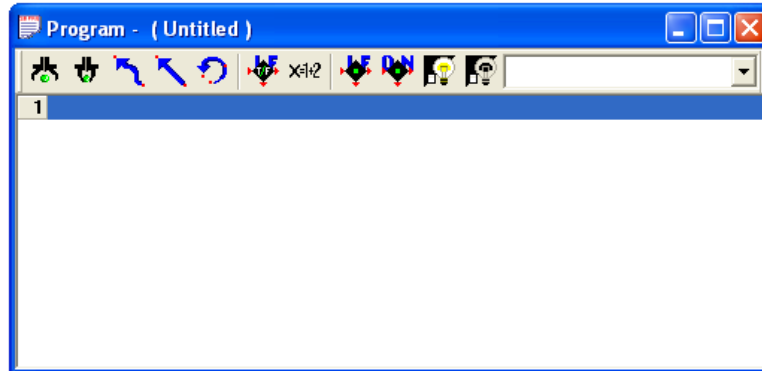
Figura 3.1. Apariencia del software SCORBASE³⁵.



- Luego haga clic en **File** y luego en **New**.

³⁵ Fuente: Imagen capturada del software SCORBASE for NXC100.

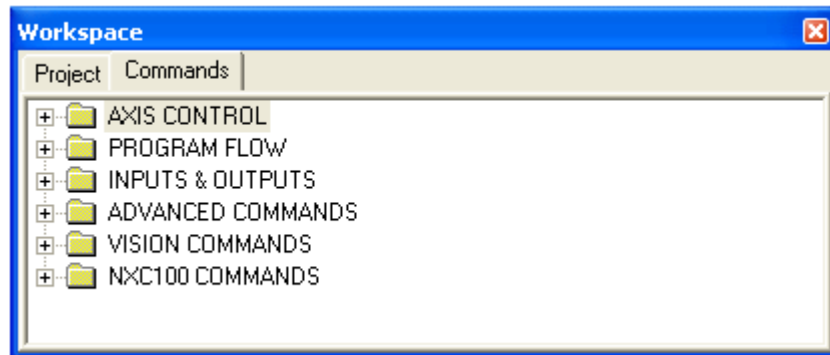
Figura 3.2. Apariencia de un nuevo programa de SCORBASE³⁶.






Este cuadro muestra el espacio donde se va a llevar a cabo la introducción de líneas de código para la elaboración de un programa.

- Observe la ventana de nombre Workspace, si no aparece haga un clic en **View** en la barra de menú y luego clic en **Workspace**.

Figura 3.3. Galería de comandos de SCORBASE³⁶.



En la barra de herramientas del Scorbace hay tres opciones    que corresponden a los diferentes niveles de dificultad, *Level1*, *Level2* y *Pro*, para el resto de este capítulo se va a trabajar en el nivel *Pro* debido a que este muestra los comandos relacionados con el NXC100.

³⁶ Fuente: Imagen capturada del software SCORBASE for NXC100.

3.3 PRÁCTICA DE LABORATORIO

3.4 MOVIMIENTO DE LA BANDA DESLIZANTE

3.4.1 Objetivo general de aprendizaje

Realizar desplazamientos lineales con la banda deslizante mediante el software Scorbace.

3.4.2 Objetivos específicos de aprendizaje

- Abrir Scorbace para NXC100.
- Llevar la banda deslizante a la posición inicial.
- Mover la banda deslizante de forma manual.
- Guardar posiciones de la banda deslizante.

3.4.3 Concepto importante: Banda deslizante

Como ya se había mencionado con anterioridad, la banda deslizante permite mover el robot a diferentes posiciones, estas posiciones se enseñan y guardan en Scorbace para posteriormente ser utilizadas en un programa, la banda deslizante está controlada por el PLC de la compañía Intelitek.

3.4.4 Procedimiento

Para este laboratorio debe contar con:

- Work Station 1 (Computador 1).
- Controlador lógico programable.
- Banda deslizante.

Siga los siguientes pasos de este laboratorio para la manipulación de la banda deslizante.



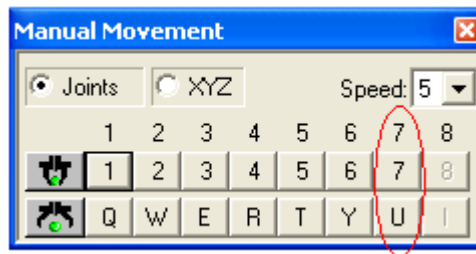
1. Asegúrese que la llave este conectada en el PC o SW1 y que por supuesto el PLC este encendido (Consulte al tutor).
2. Abra Scorbace y observe que en la barra de estatus el control debe activarse en ON  Control On, de lo contrario haga clic en **Options** en la barra de menú y luego en **Control-ON**.
3. La banda deslizante tiene una posición inicial, cuando aparece el mensaje **The robot has not been homed** debajo de la barra de herramientas, quiere decir que es necesario llevar la banda a Homme, esto se hace haciendo un clic en la opción  de la barra de herramientas o ingresando a **Run** y haciendo clic en **Search Hommen**, inmediatamente la banda se posiciona en el punto inicial.
4. Abra un nuevo proyecto de Scorbace.
5. Diríjase hacia una pequeña ventana de nombre **Manual Movement**. Si no la puede ubicar búsquela en **View** en la barra de menú.

Figura 3.4.1. Movimiento manual de ejes de Scorbace³⁷.



³⁷ Fuente: Imagen capturada del software SCORBASE for NXC100.

Esta ventana muestra los diferentes ejes que se pueden mover con el Scorbace, en este caso solo es posible mover el eje 7 donde está conectada la banda. Los demás ejes son para mover el brazo robótico que se conecta directamente al PLC de Intelitek el cual para esta celda de manufactura no está disponible.



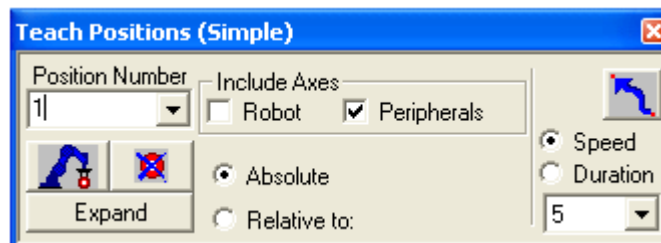

6. En la ventana de **Manual Movement** mantenga presionado el botón izquierdo del mouse en , la banda deberá deslizar al brazo robótico y si presiona  lo hará en sentido contrario.
7. Cambie la velocidad de desplazamiento, el rango va de 1 a 10 y repita el paso 6 con diferentes valores.
8. Con el **Manual Movement** lleve el robot a una posición cerca de Homme. Luego observe la ventana de nombre **Teach Positions** la cual permite guardar las posiciones de la banda deslizando. Si no la puede encontrar búsquela en **View** en la barra de menú.

Figura 3.4.2. Ventana para enseñar posiciones de la banda deslizando³⁸.





Debe escoger la opción **Peripherals** debido a que la banda es un periférico y en la opción **Position Number** escriba un número cualquiera de 1 a 1000, luego presione  para guardar dicha posición.

³⁸ Fuente: Imagen capturada del software SCORBASE for NXC100.

9. Con el **Manual Movement** lleve el robot a una posición central, luego guarde la posición como en el paso 8 con un número diferente.

10. Lleve el robot a una posición casi al final de la banda, luego guarde la posición como en el paso 8 con un número diferente.

11. Estando en la ventana **Manual Movement** busque la primera posición  guardada en el paso 8, y haga clic en  (ir a la posición seleccionada) esta opción se encarga de llevar mediante la base deslizante el robot a la primera posición grabada con anterioridad.

12. Repita el paso 10 para desplazar el robot a la segunda y tercera posición guardadas.

13. Preguntas.

a) ¿Qué ocurre cuando la banda deslizante llega a su límite?

b) ¿Qué periférico es controlado por el PLC?

3.4.5 Conclusiones

3.5 PRÁCTICA DE LABORATORIO

3.6 PROGRAMACIÓN DE POSICIONES

3.6.1 Objetivo general de aprendizaje

Crear un programa con tres posiciones diferentes de banda deslizante para ejecutarlo de forma secuencial.

3.6.2 Objetivos específicos de aprendizaje

- Introducir un comando de comentario.
- Programar posiciones de la banda.
- Cambiar parámetros de las posiciones de la banda.
- Ejecutar un programa de tres formas diferentes.
- Detener la ejecución de un programa.

3.6.3 Concepto importante: Workspace

Workspace es una ventana que permite tener acceso al repertorio de comandos del Scorbace en los tres niveles, novato, avanzado y profesional. Siendo el último nivel el más completo.

3.6.4 Procedimiento

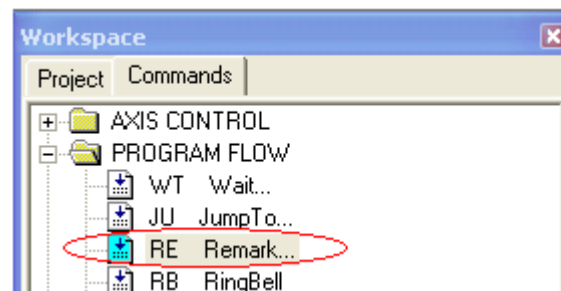
Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.

Siga los siguientes pasos.

1. Ingrese a Scorbace en modo profesional y abra un nuevo proyecto.
2. Grabe 3 posiciones de banda.
3. En la ventana del Workspace expanda la opción PROGRAM FLOW de los comandos.

Figura 3.6.1. Ventana del workspace mostrando el comando Remark³⁹.



4. Haga doble clic en la opción **Remark** y en el cuadro de texto que aparece escriba "Posiciones". Observe que en el programa se agrega una línea de comentario.

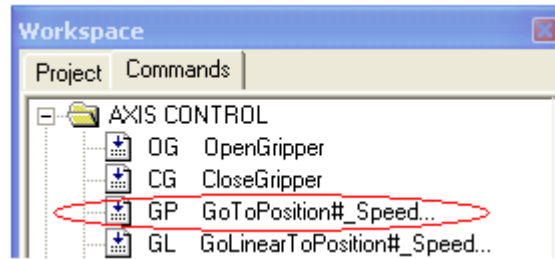
Figura 3.6.2. Programa mostrando una línea de comentario³⁹.



5. En la ventana del Workspace expanda la opción AXIS CONTROL.

³⁹ Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.6.3. Ventana del workspace mostrando el comando *ir a posición*⁴⁰.

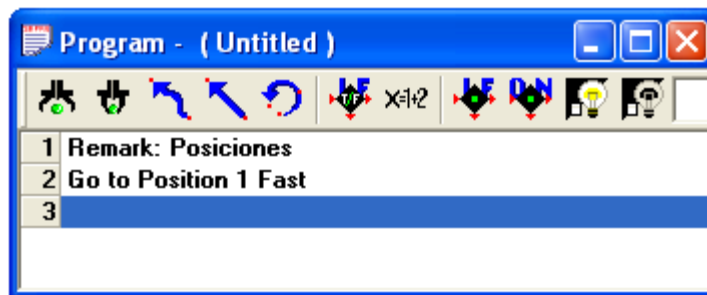


Haga doble clic en la opción **GoToPosition#_Speed**, aparecerá un cuadro donde deberá cargar la posición que desea.



, puede modificar parámetros como velocidad y tipo de movimiento, escoja movimiento joint.

Figura 3.6.4. Línea de código para ir a la posición programada⁴⁰.










6. Introduzca las otras posiciones con el comando **GoToPosition#_Speed** en el espacio de trabajo.

⁴⁰ Fuente: Imagen capturada del software SCORBASE for NXC100.


Figura 3.6.5. Líneas de código de las tres posiciones⁴¹.



Puede escoger cualquier velocidad, siendo **Fast** la más rápida y 1 la más lenta.

7. Ubique el cursor o haga clic en la primera línea de código, observe en la barra de herramientas las opciones ,  y  el cual permiten ejecutar las líneas de código de manera secuencial con tres formas diferentes, línea por línea, todas las líneas o todo el programa de forma continua. Presione la opción línea por línea , cuando la banda deslizante llegue a la primera posición vuelva a hacer clic en la opción línea por línea o **Run Single Line**, haga lo mismo con la siguiente posición.
8. Ubique el cursor o haga clic en la primera línea de código y haga clic en la opción  **Run Single Cycle** de la barra de herramientas, el programa se ejecuta hasta la última línea de código.
9. Haga clic en la opción  **Run Continuously** de la barra de herramientas, el programa se ejecutará de forma continua, para detenerlo presione **Stop** , el cual permite detener el proceso inmediatamente o puede presionar

⁴¹ Fuente: Imagen capturada del software SCORBASE for NXC100.

Pause  el cual detiene el proceso después de haber terminado de ejecutar cualquier línea de código.

10. Preguntas.

a) ¿Cuál es la diferencia de utilizar **Stop** o **Pause** para detener un proceso?

b) ¿Qué función y qué importancia tiene el comando **Remark**?

c) ¿Es posible utilizar los comandos **GoToLinerPosition#_Speed**, **GoToCircularPosition#_Speed** para un periférico? Demuéstrelo.

11. Ejercicio.

a) Grabe 9 posiciones, cada una con velocidad de 1, 2, 3.....9 respectivamente, numérelas del 10 al 19.

b) Ejecute el programa.

3.6.5 Conclusiones

3.7 PRÁCTICA DE LABORATORIO

3.8 SALTO Y ETIQUETA

3.8.1 Objetivo general de aprendizaje

Crear un programa con una secuencia continua de posiciones de banda.

3.8.2 Objetivos específicos de aprendizaje

- Utilizar el comando Label.
- Utilizar el comando Jump To.

3.8.3 Concepto importante: Comando Jump y Label

Jump es un comando de salto incondicional. El apuntador de programa salta hasta una etiqueta o Label específica creada por el programador, muchas veces se utiliza el salto y la etiqueta para optimizar la secuencia de un programa.

3.8.4. Procedimiento

Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.

Siga los siguientes pasos.

1. Asegúrese que la llave este conectada al PC y que el PLC este encendido (consultar al tutor).
2. Abra Scorbase for NXC100 desde el escritorio del PC o SW1, oprima aceptar a cualquier mensaje que aparezca.

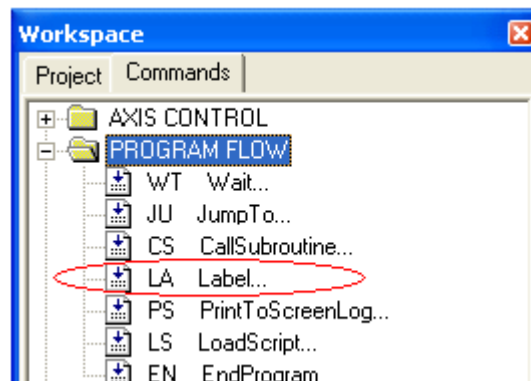
- Haga clic en **File** en la barra de menú y luego en **Open Project**, busque en escritorio la carpeta *LABORATORIOS ESTUDIANTES* y abra el proyecto **POSICIONES.WS** o cree un proyecto con 4 posiciones de banda.

Figura 3.8.1. Programa con 4 posiciones de banda⁴².



- Ubique el cursor en la línea 2, es decir en **Go to Position 1 Fast** luego en el **Workspace** y en los comandos de **PROGRAM FLOW** haga doble clic en **Label**.

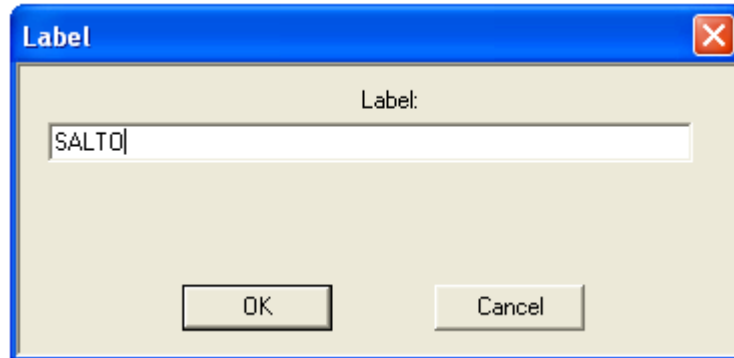
Figura 3.8.2. Comando label o etiqueta⁴².



Al hacer doble clic en este comando aparecerá la siguiente ventana:

⁴² Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.8.3. Ventana de introducción de nombre de la etiqueta⁴³.



Escriba la palabra SALTO y haga clic en Aceptar, se agregará una etiqueta entre la primera línea y la tercera línea de código.

5. Ubique el cursor en la última línea, en este caso será la número 7.

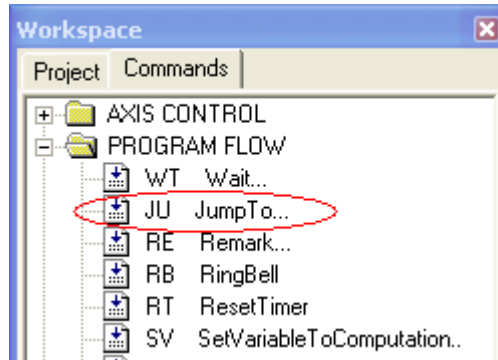
Figura 3.8.4. Programa con la etiqueta y las 4 posiciones⁴³.



6. En el **Workspace** y en los comandos de **PROGAM FLOW** haga doble clic en **Jump To**.

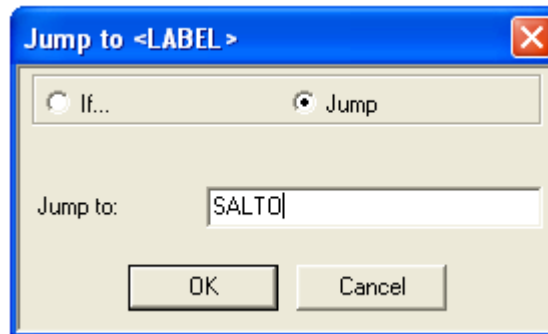
⁴³ Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.8.5. Comando Jump To⁴⁴.



Al hacer doble clic en este comando aparecerá la siguiente ventana:

Figura 3.8.6. Salto no condicional a la etiqueta SALTO⁴⁴.



Tendrá que escribir la palabra SALTO que corresponde al nombre de la etiqueta donde va llegar el salto, observe que hay dos opciones en la ventana, IF y Jump, la primera es para condicionar el salto, la segunda es el salto normal hacia la etiqueta. Haga clic en OK, enseguida se añade la línea de código al final.

⁴⁴ Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.8.7. Programa con etiqueta y código de salto⁴⁵.



7. Ubique el cursor en la primera línea de código en **Remark: Posiciones**, y luego ejecute todas las líneas de código con la opción **Run Single Cycle** en la barra de herramientas, el robot comenzará a desplazarse por medio de la banda de forma continua a causa del comando **Jump To**. Para parar el proceso haga clic en **Pause** en la barra de herramientas.

3.8.5 Conclusiones

⁴⁵ Fuente: Imagen capturada del software SCORBASE for NXC100.

3.9 PRÁCTICA DE LABORATORIO

3.10 TEMPORIZACIÓN

3.10.1 Objetivo general de aprendizaje

Crear temporizaciones en un programa.

3.10.2 Objetivos específicos de aprendizaje

- Utilizar el comando Wait de Scorbace.
- Editar el tiempo de temporización del comando Wait.
- Observar el comportamiento de la temporización en la barra de estatus.

3.10.3 Concepto importante: Comando Wait

Este comando permite hacer temporizaciones entre instrucciones o líneas de código el cual resulta muy útil para ciertas aplicaciones que se verán más adelante.

3.10.4 Procedimiento

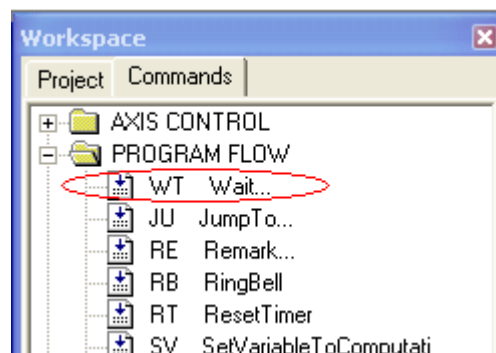
Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.

Retome el programa del laboratorio anterior del comando Jump y Label.

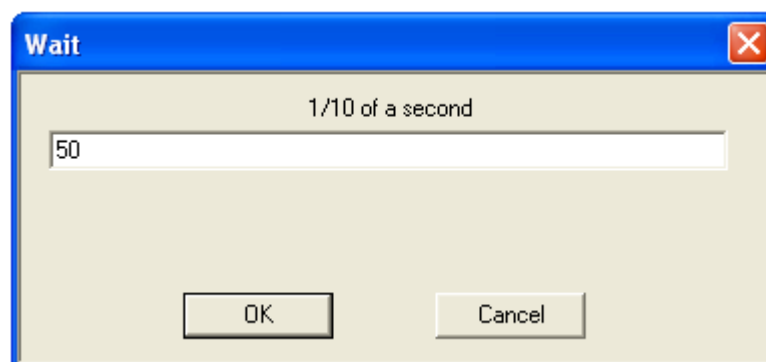
1. Se quiere colocar una temporización de 5 segundos entre la posición 1 y la posición 2, para comenzar ubique el cursor en la línea número 4 o en la instrucción **Go To Position 2 Fast**.
2. En el **Workspace** y en los comandos de **PROGRAM FLOW** haga doble clic en **Wait**.

Figura 3.10.1. Comando Wait⁴⁶.



Al hacer doble clic en este comando aparecerá la siguiente ventana:

Figura 3.10.2. Introducción de tiempo para el comando Wait⁴².



⁴⁶ Fuente: Imagen capturada del software SCORBASE for NXC100.

La relación de tiempo para este comando es decimas de segundo, en efecto se digita 50 que correspondería a 50 decimas de segundos o 5 segundos. Haga clic en OK y observe que la instrucción de temporización se agrega entre las posiciones 1 y 2.

Figura 3.10.3. Temporización de 5 segundos entre la posición 1 y 2⁴⁷.



3. Ejecute el programa con **Run Single Cycle** en la barra de herramientas y verifique que la temporización entre las posiciones 1 y 2 es 5 segundos.
4. Repita el mismo proceso para añadir una temporización de 4 segundos entre las posiciones 2 y 3, una temporización de 3 segundos entre las posiciones 3 y 4.

⁴⁷ Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.10.4. Temporizaciones entre las 4 posiciones⁴⁸.



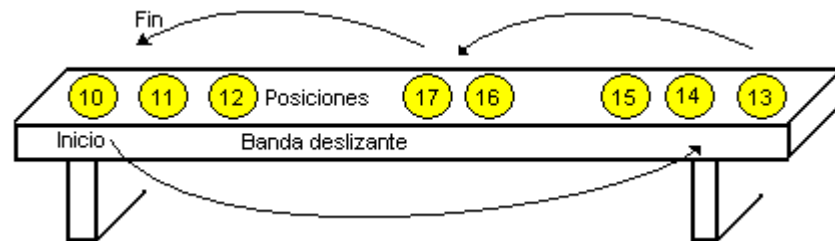
5. Ejecute el programa con **Run Single Cycle** en la barra de herramientas y verifique las temporizaciones de 5, 4 y 3 segundos.
6. Pregunta.
 - a) ¿Qué puede observar en la barra de estatus mientras ejecuta un programa con temporización? Explique.

7. Ejercicio

⁴⁸ Fuente: Elaboración propia.

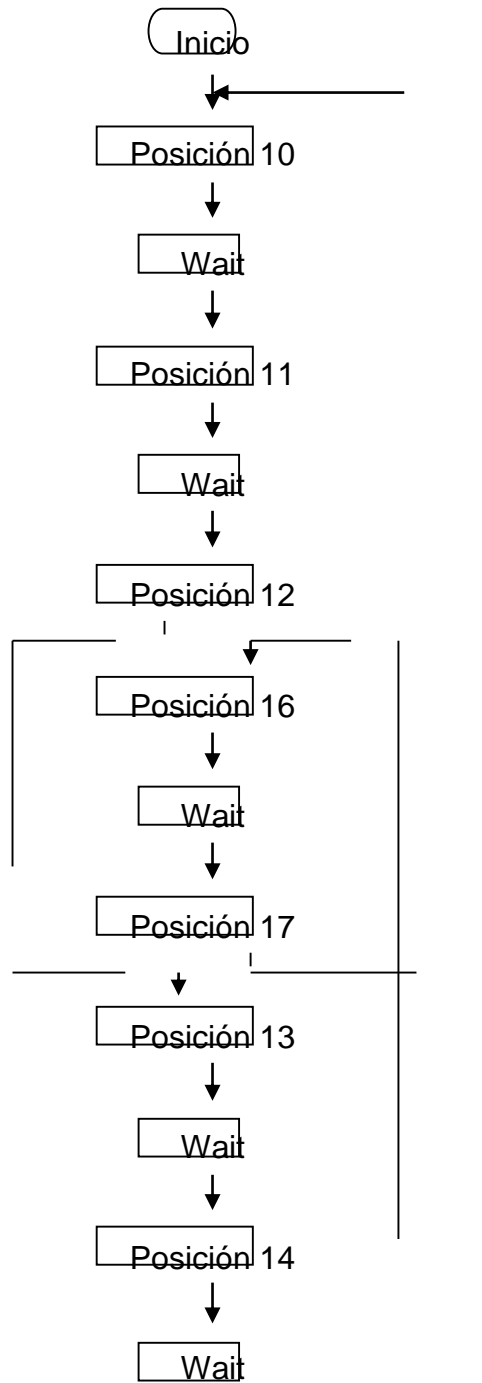
- a) Se quiere hacer uso de saltos y temporizaciones para este ejercicio, en el dibujo se muestra las posiciones numeradas en la banda y en ese orden la secuencia.

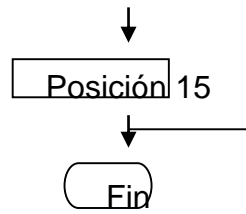
Figura 3.10.5. Secuencia de posiciones de la banda deslizante⁴⁹.



- b) Deberá hacer un programa en el Scorbace para estas posiciones, tenga en cuenta los nombres 10,11.....17 de esas posiciones. En el siguiente diagrama de flujo esta la secuencia del programa con los saltos que debe hacer y las temporizaciones. Es posible hacer este mismo programa de forma más sencilla pero lo que se busca con este ejercicio es poner en práctica los comandos de salto, etiqueta y temporización.

⁴⁹ Fuente: Elaboración propia.





3.10.5 Conclusiones

3.11 PRÁCTICA DE LABORATORIO

3.12 DEFINIR VARIABLES

3.12.1 Objetivo general de aprendizaje

Crear un programa haciendo uso de variables.

3.12.2 Objetivos específicos de aprendizaje

- Utilizar el comando Ser Variable.
- Editar las opciones del comando Set Variable.
- Editar las opciones del comando Go to Position.

3.12.3 Concepto importante: Comando Set Variable

Muchas veces resulta más conveniente trabajar con variables en lugar de valores constantes, esto permite un mejor manejo y control de programas extensos. El comando Set Variable permite crear una variable y asignar un valor numérico entero o el resultado de una operación matemática o de cómputo.

3.12.4 Procedimiento

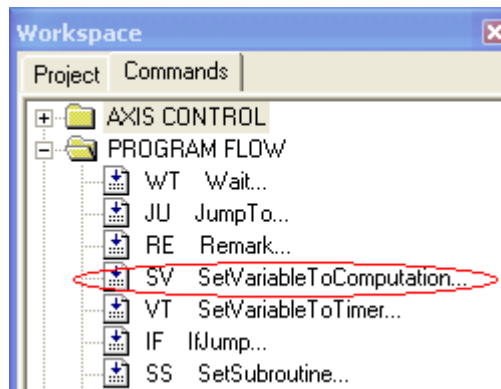
En este laboratorio se enseñará el manejo de comandos que permiten la utilización de variables creadas por el mismo usuario. Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.

➤ Banda deslizante.

1. Ingrese a Scorbace, asegúrese de tener el **control ON** y de estar en el nivel **Pro**.
2. Grabe dos posiciones de banda, 1 y 2.
3. En el **Workspace** y en los comandos de **PROGAM FLOW** haga doble clic en **SetVariableToComputation**.

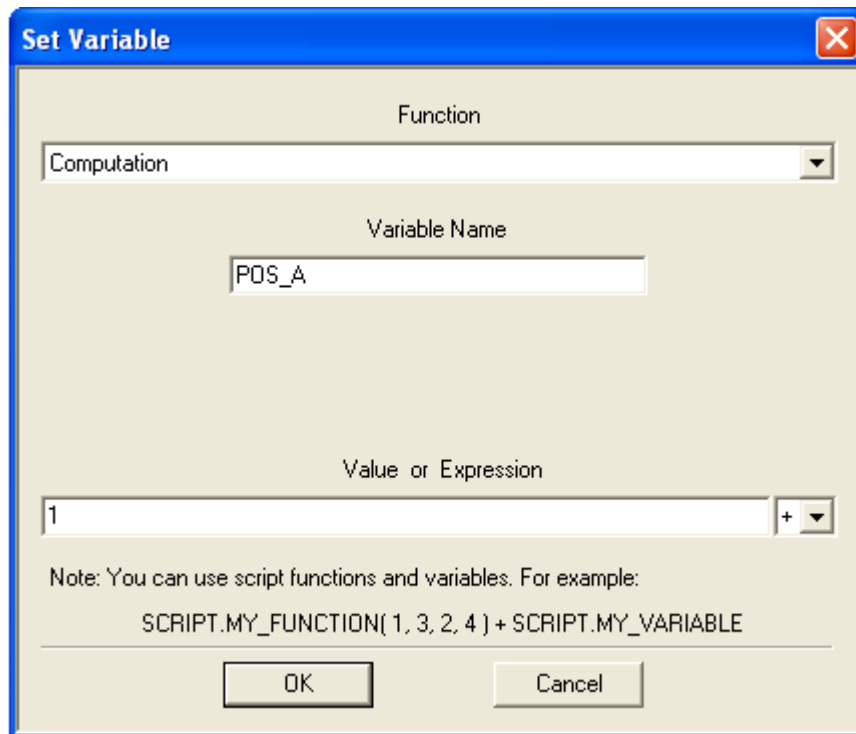
Figura 3.12.1. Comando Set Variable⁵⁰.



Al hacer doble clic en este comando aparecerá la siguiente ventana:

⁵⁰ Fuente: Imagen capturada del software SCORBASE for NXC100.

Figura 3.12.2. Ventana de opciones del comando Set Variable⁵¹.



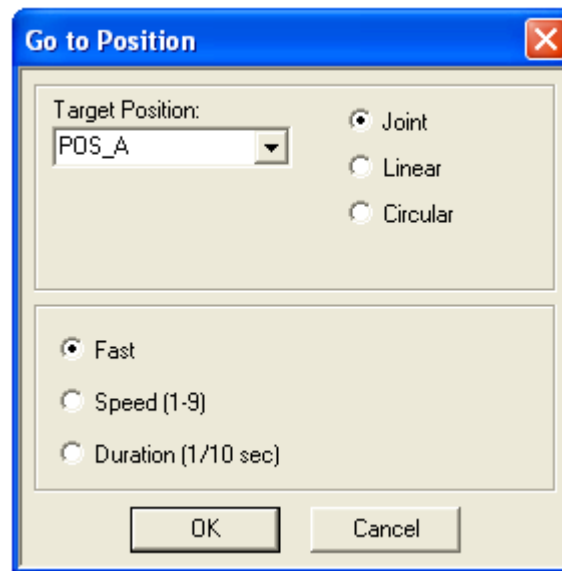
La función es de tipo computacional para trabaja con operaciones matemáticas. Donde dice Nombre de Variable simplemente escriba el nombre de la variable. En Valor o expresión escriba el número 1, el cual coincide con la primera posición de periférico o de banda que se guardó. Haga clic en OK, enseguida se inserta este comando en el área de programación como: **Set Variable POS_A = 1.**

4. Repita el paso 3 para agregar otra variable de usuario **Set Variable POS_B = 2.**

⁵¹ Fuente: Fuente: Imagen capturada del software SCORBASE for NXC100.

5. En el **Workspace** y en los comandos de **AXIS CONTROL** haga doble clic en **GoToPosition#_Speed**. En el cuadro de cargar posición escriba el nombre de la primera variable de usuario que se creó, **POS_A**.

Figura 3.12.3. Ventana de opciones del comando Go to Position⁵².



Haga clic en OK, la línea de código que se inserta deberá ser **Go to Position POS_A Fast**, observe que esta instrucción no tiene un valor constante sino una variable con relación a las posiciones en la banda, que puede ser modificada haciendo doble clic sobre la instrucción **Set Variable POS_A = 1**, el cual abre la ventana donde puede modificar el valor de la variable.

6. Repita el paso 5 para insertar la instrucción **Go To Position POS_B Fast**, el programa final deberá ser:

⁵² Fuente: Imagen capturada del software SCORBASE for NXC100.

Set Variable POS_A = 1

Set Variable POS_B = 2

Go to Position POS_A Fast

Go to Position POS_B Fast

7. Ejecute el programa.

3.12.5 Conclusiones

3.13 PRÁCTICA DE LABORATORIO

3.14 SALTO CONDICIONAL

3.14.1 Objetivo general de aprendizaje

Observar el comportamiento del salto condicional en un programa.

3.14.2 Objetivos específicos de aprendizaje

- Utilizar el comando If Jump.
- Editar las opciones del comando If Jump.

3.14.3 Concepto importante: If Jump

Este es un comando de salto condicional, que se utiliza para comprobar el valor de una variable. Si la condición es verdadera el programa saltará hasta la Etiqueta especificada.

3.14.4 Procedimiento

Para este laboratorio debe contar con:

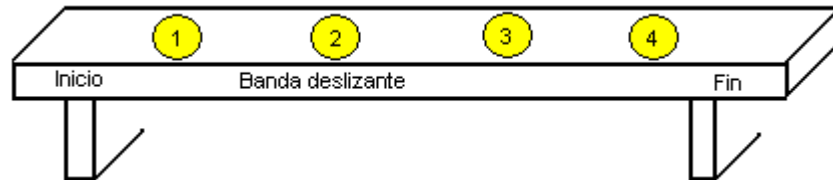
- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.

Siga los siguientes pasos.

1. Ingrese a Scorbace, asegúrese de tener el **control ON** y de estar en el nivel **Pro**.

2. Abra un nuevo proyecto.
3. Grabe 4 posiciones de periférico o de banda y numérelas de 1 a 4. Trate de que las posiciones tengan un orden coherente a lo largo de la banda deslizante.

Figura 3.14.1. Posiciones a lo largo de la banda deslizante⁵³.

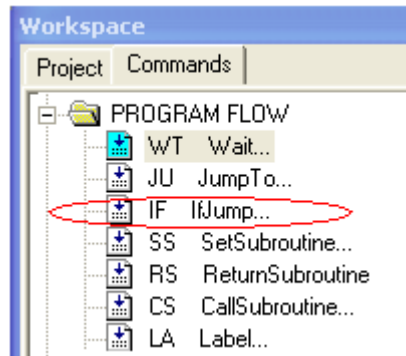


4. En el **Workspace** y en los comandos de **PROGAM FLOW** haga doble clic en **SetVariableToComputation** para inicializar una variable. En **Variable Name** escriba por ejemplo **POS**, en **Value or Expression** escriba 0.
5. En la siguiente línea de código agregue una Etiqueta o **Label**, de nombre **INICIO**.
6. Agregue en la siguiente línea una instrucción con el comando **SetVariableToComputation**, en **Variable Name** escriba **POS** y en **Value or Expression** escriba **POS + 1**. La línea deberá ser **Set Variable POS = POS + 1**, esta expresión incrementa el valor de la variable **POS**.
7. En el **Workspace** y en los comandos de **AXIS CONTROL** haga doble clic en **GoToPosition#_Speed**. En la ventana que aparece, en la parte de cargar posición escriba **POS**.
8. Agregue una temporización de 3 segundos con el comando **Wait**.

⁵³ Fuente: Elaboración propia.

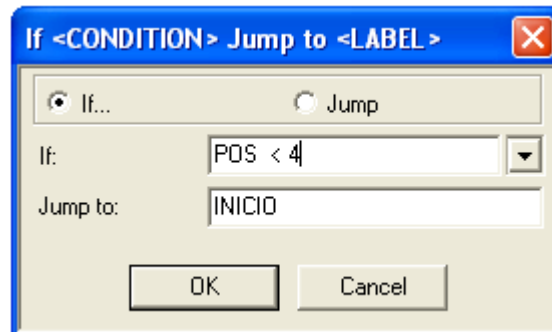
9. En el **Workspace** y en los comandos de **PROGRAM FLOW** haga doble clic en **IfJump**.

Figura 3.14.2. Comando If Jump⁵⁴.



Al hacer doble clic en este comando aparecerá la siguiente ventana:

Figura 3.14.3. Opciones del comando If Jump⁵³.



La condición es que la variable **POS** sea menor a 4, si esto se cumple habrá un salto hasta la etiqueta **INICIO**, de lo contrario el apuntador de programa sigue a la siguiente instrucción.

⁵⁴ Fuente: Imagen capturada del software SCORBASE for NXC100.

10. Agregue **END** al programa ubicando el apuntador en la última línea, escriba *end*, en seguida se agrega la instrucción final. El programa final deberá ser el siguiente:

```
Set Variable POS = 0  
INICIO:  
Set Variable POS = POS + 1  
Go to Position POS Fast  
Wait 30 (10ths of seconds)  
If POS < 4 Jump to INICIO  
End
```

11. Ejecute el programa.

12. Pregunta.

a) Observe el programa anterior y las posiciones y describa el proceso.

3.14.5 Conclusiones

3.15 PRÁCTICA DE LABORATORIO

3.16 VISUALIZACIÓN DE VALORES DE VARIABLE

3.16.1 Objetivo general de aprendizaje

Visualizar los valores que puede tomar una variable en un programa mientras se ejecuta.

3.16.2 Objetivos específicos de aprendizaje

- Utilizar el comando Print to screen.
- Visualizar el estado de más de una variable.

3.16.3 Concepto importante: Print to screen

Este comando es muy útil porque permite visualizar el valor que toma cada variable mientras el programa se está ejecutando.

3.16.4 Procedimiento

Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.

Siga los siguientes pasos.

1. Retome el programa del laboratorio anterior.

Set Variable POS = 0

INICIO:

Set Variable POS = POS + 1

Go to Position POS Fast

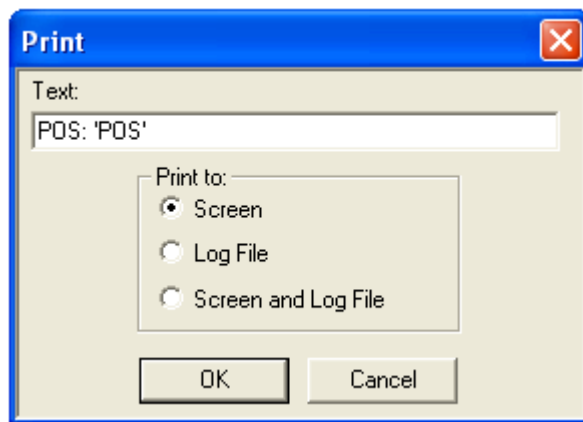
Wait 30 (10ths of seconds)

If POS < 4 Jump to INICIO

End

2. Ubique el apuntador de programa sobre ***Go to Position POS Fast***.
3. En el **Workspace** y en los comandos de **PROGRAM FLOW** haga doble clic en **PrintToScreenLog**. En la ventana que aparece, elija la opción **Screen** y en la barra de texto escriba: **POS: 'POS'**.

Figura 3.16.1. Opciones del comando Print to screen⁵⁵.



Haga clic en OK, enseguida la instrucción se agrega.

Set Variable POS = 0

INICIO:

Set Variable POS = POS + 1

⁵⁵ Fuente: Imagen capturada del software SCORBASE for NXC100.

Print to Screen: POS: 'POS'

Go to Position POS Fast

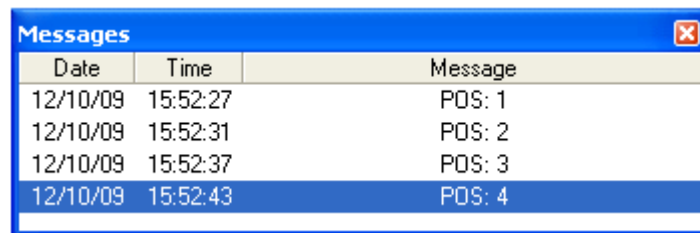
Wait 30 (10ths of seconds)

If POS < 4 Jump to INICIO

End

4. Ejecute el programa y observe el cuadro que aparece.

Figura 3.16.2. Ventana que muestra los valores que va tomando la variable POS⁵⁶.



Date	Time	Message
12/10/09	15:52:27	POS: 1
12/10/09	15:52:31	POS: 2
12/10/09	15:52:37	POS: 3
12/10/09	15:52:43	POS: 4

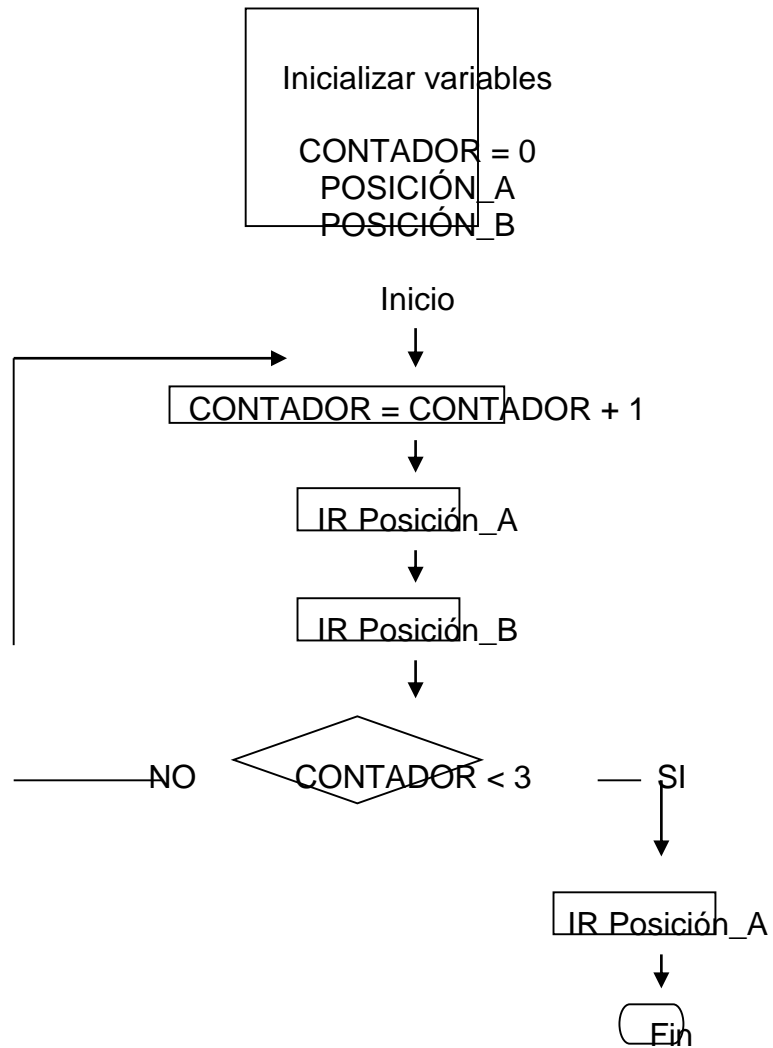
Suponga que quiere visualizar mas variables, en la barra de texto se debe escribir por ejemplo: **V1,V2,V3,.....Vn : 'V1', 'V2', 'V3'... 'Vn'**.

5. Ejercicio.

a) Utilizando variables haga un programa en Scorbace que permita desplazar 3 veces el brazo de un extremo al otro. Deberá mostrar el estado de cada variable, es por esto que tendrá que usar variables para las posiciones.

⁵⁶ Fuente: Imagen capturada del software SCORBASE for NXC100.

b) Utilice el siguiente diagrama de flujo como referencia.



3.16.5 Conclusiones

3.17 PRÁCTICA DE LABORATORIO

3.18 COMANDOS DEL CONTROLADOR NXC100

3.18.1 Objetivo general de aprendizaje

Utilizar los comandos más importantes del controlador NXC100 para ejecutar trabajos del Teach pendant de manera remota desde Scorbace.

3.18.2 Objetivos específicos de aprendizaje

- Utilizar el comando Startjob para ejecutar un trabajo del Teach pendant.
- Utilizar el comando Jobwait.
- Manejar el modo remoto del Teach pendant.
- Activar los servos con el comando Control ON.
- Desactivar los servos con el comando Control OFF.

3.18.3 Concepto importante: Startjob y JobWait

Los comandos para el controlador NXC100 permiten interactuar directamente con el brazo robótico mediante SCORBASE, estas instrucciones funcionan en el modo profesional, el comando Start job permite llamar un trabajo directamente del Teach pendant. El comando Jobwait es sumamente importante, por lo general va después de una instrucción StartJob, la función de JobWait es dar el tiempo necesario para que se ejecute un trabajo del Teach pendant para posteriormente ejecutar la siguiente línea de código en el Scorbace.

3.18.4 Procedimiento

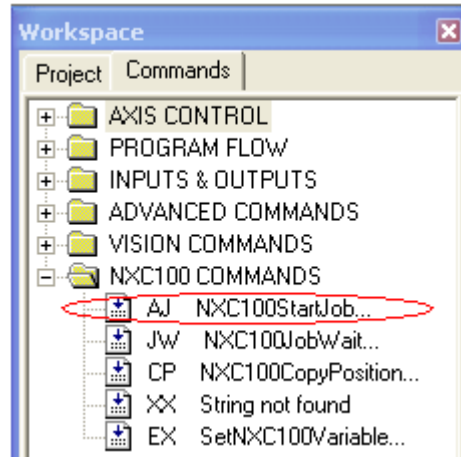
Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Llave MOTOCOM.

Siga los siguientes pasos.

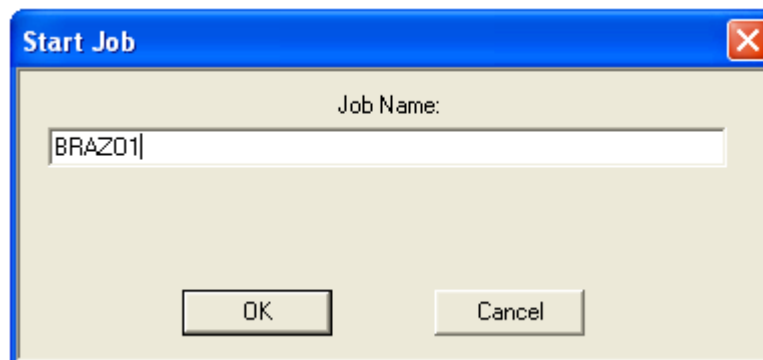
1. Asegúrese que la llave está conectada al PC o SW1.
2. Encienda el controlador NXC100 y tome el **Teach pendant**, cambie al modo **{REMOTE}** con el Switch modal.
3. Abra Scorbace, los Servos del brazo deberán activarse esto lo indica un pitido agudo y el indicador luminoso de SERVO ON del **Teach pendant**.
4. Abra un nuevo proyecto en Scorbace.
5. Lleve el robot mediante la banda deslizante a una posición central en la banda.
6. En el **Workspace** y en los comandos de **NXC100 COMMANDS** haga doble clic en **NXC100Startjob**.

Figura 3.18.1. Comando Startjob del controlador NXC100⁵⁷.



En la ventana que aparece escriba BRAZO1.

Figura 3.18.2. Nombre del trabajo que se va a ejecutar desde el Teach pendant⁵⁶.



Job Name debe estar definido con anterioridad en el **Teach pendant**. Haga clic en OK.

⁵⁷ Fuente: Imagen capturada del software SCORBASE for NXC100.

7. Ejecute la instrucción **NXC100 Start Job BRAZO1**, el brazo deberá moverse. En caso contrario revise que el SERVO ON este encendido en caso contrario cierre el programa y repita los pasos del 1 al 7.
8. En el mismo trabajo, haga clic sobre **NXC100 Start Job BRAZO1** y luego diríjase a **Workspace** en **AXIS CONTROL**, haga doble clic en el comando **ControlON**.
9. En la línea 3 debajo de **NXC100 Start Job BRAZO1** agregue el comando **NXC100 Job Wait 60 (seconds)** que se encuentra en **NXC100 COMMANDS**, digite 60 para esta instrucción, esto equivale a 60 segundos, este rango puede variar dependiendo de qué duración tenga la ejecución del trabajo en el Teach pendant, para todos los trabajos se estimará 60 segundos.
10. En la secuencia inserte al final el comando **ControlIOF** que se encuentra en **AXIS CONTROL**. El comando **Control OF** y **Control ON** desactiva o activa los servos del brazo robótico. El programa tuvo que haber quedado de la siguiente forma:

Control On

NXC100 Start Job BRAZO1

NXC100 Job Wait 60 (seconds)

Control Off

11. Ejecute todo el programa.

12. Ejercicio.

a) Con la tecla suprimir del teclado elimine la línea de instrucción **NXC 100 Job Wait 60 (seconds)** del programa que acabó de hacer.

b) Ejecute el programa varias veces y describa lo sucedido.

c) Describa la importancia de utilizar el comando **NXC 100 Job Wait** después que se ejecuta un trabajo del **Teach Pendant**.

3.18.5 Conclusiones

3.19 PRÁCTICA DE LABORATORIO

3.20 SUBRUTINAS

3.20.1 Objetivo general de aprendizaje

Ejecutar un programa haciendo uso de subrutinas.

3.20.2 Objetivos específicos de aprendizaje

- Utilizar el comando Set subroutine para crear una subrutina.
- Utilizar el comando Call subroutine para llamar una subrutina.
- Utilizar el comando Return subroutine para retornar al programa principal.
- Ejecutar trabajos del Teach pendant desde Scorbace.
- Utilizar el sistema neumático de la celda de manufactura.
- Utilizar el gripper del robot.
- Utilizar las plantillas de la celda de manufactura.

3.20.3 Concepto importante: Call Subroutine, Set Subroutine y Return Subroutine

La instrucción set subroutine permite definir una subrutina, recuerde que el concepto de subrutina no es más que un proceso que se puede tomar varias veces en un programa. El comando call subroutine es muy importante a la hora de elaborar procesos complejos debido a que permite llamar las subrutinas creadas por el programador. Return subroutine simplemente permite retornar a la ejecución normal del programa al finalizar el proceso de la subrutina y siempre será la última línea de código de una subrutina.

3.20.4 Procedimiento

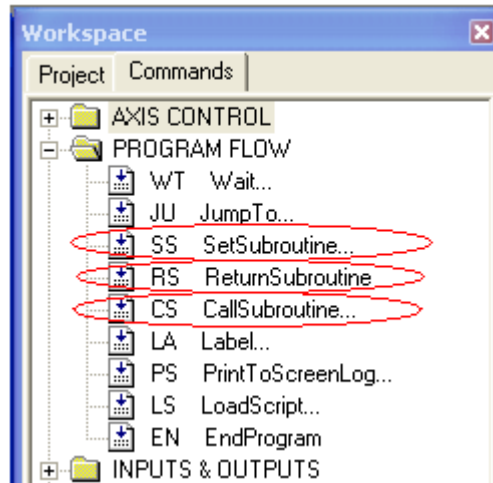
Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Llave MOTOCOM.
- Compresor de aire.

Siga los siguientes pasos.

1. Asegúrese que la llave está conectada al PC o SW1.
2. Encienda el controlador NXC100 y tome el **Teach pendant**, cambie al modo **{REMOTE}** con el Switch modal.
3. Abra Scorbace, los Servos del brazo deberán activarse esto lo indica un pitido agudo y el indicador luminoso de SERVO ON del **Teach pendant**.
4. Abra un nuevo proyecto en Scorbace.
5. En el **Workspace** y en los comandos de **PROGAM FLOW** ubique la instrucción **CallSubroutine**.

Figura 3.20.1. Comandos relacionados con subrutinas⁵⁸.



6. Haga doble clic en **CallSubroutine** en seguida debe aparecer un cuadro el cual pide escribir el nombre de la subrutina que se va a llamar, haga clic en OK y se agregará un línea de código como:

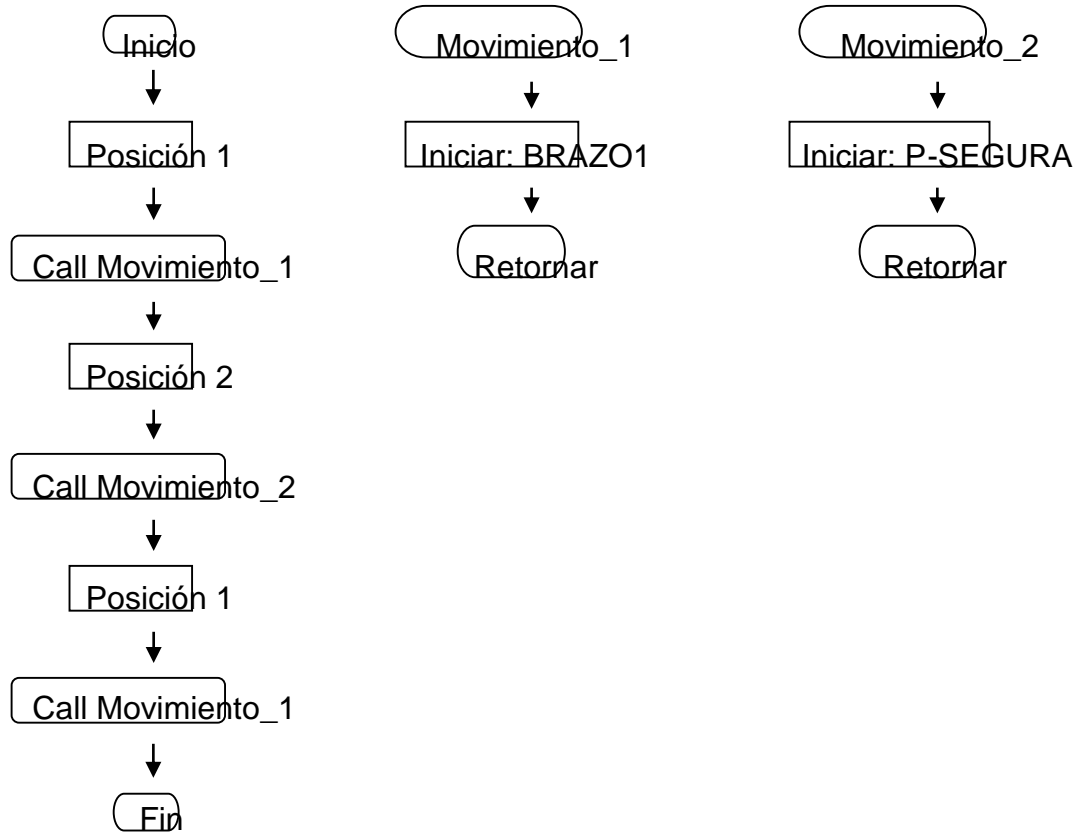
Call Subroutine NAME_SUBROUTINE

7. En el **Workspace** y en los comandos de **PROGAM FLOW** ubique la instrucción **SetSubroutine**.
8. Haga doble clic sobre la instrucción **SetSubroutine**, en seguida aparece un cuadro que pide ingresar el nombre de Subrutina, *Enter a name for the subroutine* y haga clic en OK, la instrucción se deberá mostrar cómo:

Set Subroutine NAME_SUBROUTINE

⁵⁸ Fuente: Imagen capturada del software SCORBASE for NXC100.

9. Observe e interprete el siguiente diagrama de flujo:



Las posiciones 1 y 2 se refieren a dos puntos específicos de la banda deslizante, BRAZO1 y P-SEGURA son programas o trabajos ya creados anteriormente en el **Teach pendant**.

10. Guarde dos posiciones arbitrarias y numérelas como 1 y 2.

11. El diagrama de flujo llevado al lenguaje de SCORBASE sería:

Control On

Go to Position 1 Fast

Call Subroutine MOVIMIENTO_1

Go to Position 2 Fast

Call Subroutine MOVIMIENTO_2

Go to Position 1 Fast

Call Subroutine MOVIMIENTO_1

Control Off

End

Remark: *****

Set Subroutine MOVIMIENTO_1

NXC100 Start Job BRAZO1

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine MOVIMIENTO_2

NXC100 Start Job P-SEGURA

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Impleméntelo en el software y ejecútelo, si no funciona asegúrese que los trabajos BRAZO1 y P-SEGURA estén en el JOB LIST del **Teach pendant** o reinicie SCORBASE. Hay que tener claro que las subrutinas van siempre al final del programa y la última línea de código debe ser un **Return from Subroutine**.

12. Ejercicio 1.

- a) Como ejercicio se mostrará un proceso pero primero debe asegurarse de que el sistema neumático este funcionando correctamente y todos los dispositivos que componen la celda.
- b) En la celda de ensamblaje las plantillas o Template se colocan únicamente en el Buffer, se quiere capturar las plantillas del Index 9 e Index 10 del **ASRS** o armario y colocarlas en el Buffer 1.
- c) Diríjase a SCORBASE y abra el proyecto PROCESO_1 de la carpeta LABORATORIOS ESTUDIANTES que se encuentra en el escritorio. El proyecto deberá ser el siguiente:

Control On

INICIO:

Set Variable C = 0

Set Variable PB = 9

Set Variable PB1 = 20

Remark: *****

SEGUIR:

Set Variable C = C+1

Call Subroutine POSICION_SEGURA

Call Subroutine PLANTILLA_IR

Call Subroutine CERRAR_GRIPER

Call Subroutine SACAR_PLANTILLA

Call Subroutine BUFFER_PONER

Call Subroutine ABRIR_GRIPER

Set Variable PB = PB+1

Set Variable PB1 = PB1+1

If C == 2 Jump to FIN

Jump to SEGUIR

FIN:

Call Subroutine POSICION_SEGURA

Control Off

End

Remark: *****

Set Subroutine PLANTILLA_IR

Go to Position PB Fast

NXC100 Start Job PROC_1A

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine SACAR_PLANTILLA

Go to Position PB Fast

NXC100 Start Job PROC_1B

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine BUFFER_PONER

Go to Position PB1 Fast

NXC100 Start Job PROC_1C

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine CERRAR_GRIPER

NXC100 Start Job CGRIP

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine ABRIR_GRIPER

NXC100 Start Job OGRIP

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

Set Subroutine POSICION_SEGURA

NXC100 Start Job P-SEGURA

NXC100 Job Wait 60 (seconds)

Return from Subroutine

Remark: *****

d) Ejecute el proceso y obsérvelo.

e) Cuantas subrutinas tiene este programa y mencione cuales son.

f) Explique la función de las variables ***C = 0, PB = 9 y PB1 = 20.***

- g) Explique el propósito de las líneas de código: ***Set Variable C = C+1, Set Variable PB = PB + 1, Set Variable PB1 = PB1 + 1, If C==2 Jump to FIN.***

13. Ejercicio 2.

- a) Capturar el Template del Index 5, llevarlo al Buffer 1 Index 2 luego capturar la pieza con agujero desde el Index 1 del Buffer 1 y llevarla al Index 2 del Rack1
- b) Describa cada variable y cada proceso con variable que utilice en el programa.

3.20.5 Conclusiones

CAPÍTULO 4

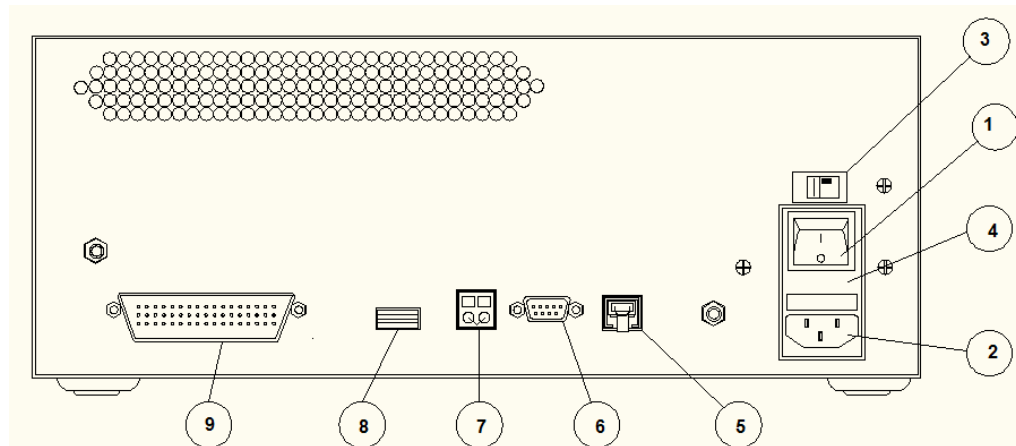
4. CONTROLADOR LÓGICO PROGRAMABLE

4.1 INTRODUCCIÓN DEL CAPÍTULO

Este capítulo tratará particularmente sobre el manejo del controlador lógico programable de Intelitek mediante el software SCORBASE. El PLC o controlador lógico programable es un dispositivo electrónico digital que usa una memoria programable para almacenar internamente instrucciones para implementar funciones específicas, como lógicas, secuencia, sincronización, conteo y operaciones aritméticas para controlar, a través de módulos digitales o analógicos de entrada y salida, diversas clases de máquinas o procesos.

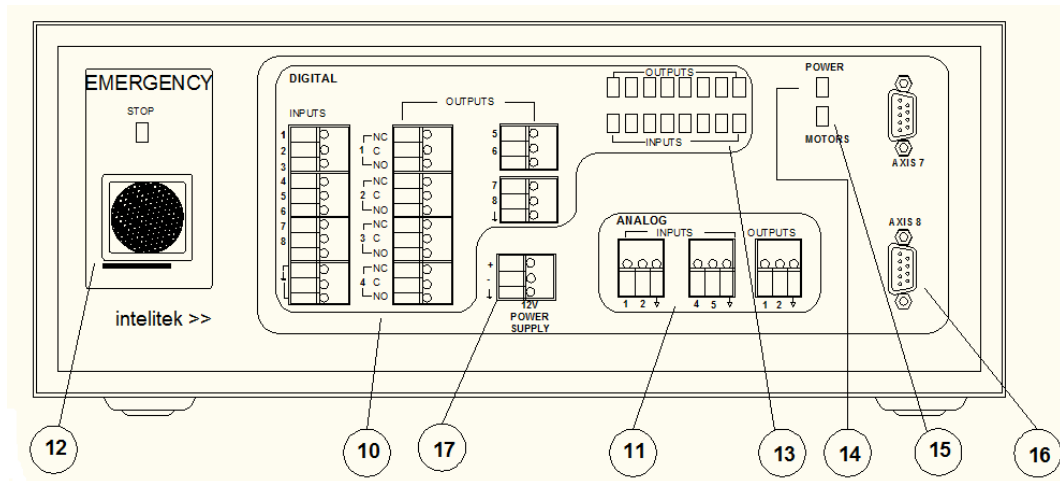
Los controladores lógicos programables se han adoptado ampliamente en los sistemas y operaciones de manufactura. Sus funciones básicas son activar y desactivar salidas, movimiento, operaciones secuenciales y control con retroalimentación. En la siguiente figura se muestra el controlador lógico programable que posee la celda de manufactura del laboratorio de procesos industriales.

Figura 4.1.1. Esquema de la parte posterior del controlador lógico programable.



1	Interruptor de encendido y apagado
2	Línea de potencia 110/220 V
3	Selector de línea de potencia
4	Fusibles de protección 110/220V 1A
5	Puerto de conexión con el Teach Pendant
6	Puerto RS232 para futuras aplicaciones
7	Switch para emergencias remotas
8	Puerto USB para conexión con ordenador
9	Puerto 62 pines tipo D de alta densidad

Figura 4.1.2. Esquema de la parte frontal del controlador lógico programable.



10	Terminales digitales de entrada y salida
11	Terminales analógicas de entrada y salida
12	Botón de emergencia y LED indicador
13	Indicadores de estado de salidas y entradas digitales
14	Indicador de encendido
15	Indicador de actividad de control
16	Conectores D9 para los ejes 7 y 8 (separado para cada dispositivo)
17	Fuente auxiliar de 12V 0,1 A

4.2. ENTRADAS Y SALIDAS

Las terminales de salidas permiten al controlador lógico programable gestionar los dispositivos en el entorno del robot. Este posee 10 salidas que son las siguientes:

- 4 salidas digitales por relé.
- 4 salidas digitales de colector abierto.
- 2 salidas analógicas.

Cada salida tiene asociado un diodo emisor de luz indicador que refleja el estado de cada una de ellas, e individualmente ilumina cuando cualquier salida digital esta activa.

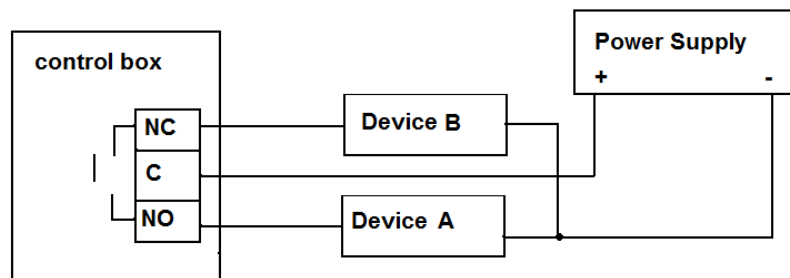
4.2.1. Salida de Relé

Las salidas digitales 1 al 4 son salidas por relé. Estas son controladas por los comando del software Scorbace. Entrega un voltaje máximo de 24 V y una corriente máxima de 1.5 A. cada una de las salidas de relé posee tres puntos de contactos:

- Común (C)
- Normalmente Cerrado (NC).
- Normalmente Abierto (NO).

Normalmente (antes de que los relé estén energizado por los comandos de Scorbace) la terminal NO está conectada de la línea común, y la terminal NC está conectada al extremo COM. Cuando el relé esta activado el extremo NO es conectado a la Terminal COM, y NC es desconectada.

Figura 4.2.1. Esquema de interconexión para dispositivos por Relé.



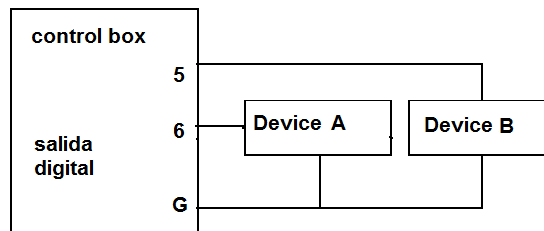
4.2.2. Salida por Colector Abierto

Las salidas 5 al 8 incluye un transistor con colector abierto en su etapa final. Estas salidas pueden ser conectadas a la carga tal como resistores, selenoides, relés y motores. Cuando se utiliza cargas inductivas se recomienda conectar diodos de protección para contra restar los retorno de corrientes.

Se puede conectar directamente la salida de colector abierto del controlador a una entrada digital de este mismo. Este tipo de salidas pueden generar un voltaje de salida de 24V y una corriente de salida Máxima de 0.5 A.

Nunca conecte el voltaje desde una fuente directamente a cualquier salida por colector abierto (terminales 5-8). La salida por colector abierto debe ser siempre conectada a la carga que está por encima de la escala.

Figura 4.2.2. Esquema de interconexión para la salida por colector abierto.



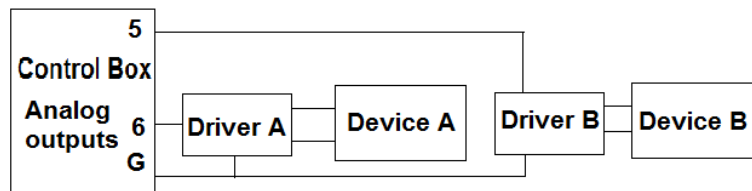
4.2.3. Salidas Análogas

Las salidas análogas 1 y 2 permiten controlar dispositivos que operan en acorde con una entrada análoga de voltaje, tales como un Led o motores.

Las salidas análogas tienen un conversor análogo digital de 8 bit de resolución y una salida de voltaje de 0 -10 V. el voltaje es controlado por medio del software Scorbace (Comando output Word 0-255 0 – 10 V).

Las salidas análogas están limitadas a 20mA. Utilice una etapa de amplificación para energizar dispositivos que requieran potencias más altas.

Figura 4.2.3. Esquema de interconexión para las salidas análogas.



4.2.4. Terminales de entradas y Leds

El controlador lógico programable posee 8 entradas digitales y cuatro entradas análogas, posee 8 leds verdes los cuales tienen como fin mostrar el estado de las entradas. Individualmente se encienden cuando cualquier entrada digital esta activa.

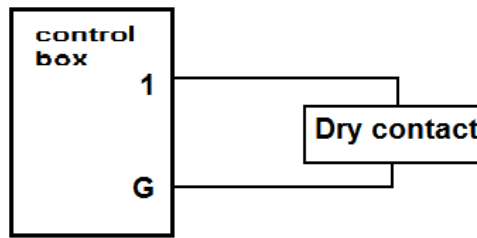
Cuando conecte dispositivos externos asegure que la salida del periférico sea tipo NPN.

Dos tipos de dispositivos pueden ser conectados a las entradas de digitales 1-8.

- Un switch de contacto seco o sensor.

Conecte el dispositivo a una de las terminales de entrada del controlador y la entrada puesta a tierra del controlador. Al cerrar el contacto del swicht o sensor la entrada se activará.

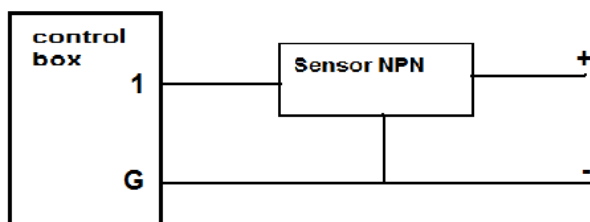
Figura 4.2.4. Esquema de conexión.



- Dispositivos Sensibles NPN

Conecte la salida del sensor a la entrada del controlador y la tierra del sensor a la tierra de entrada del PLC.

4.2.3. Figura muestra el diagrama de conexión.



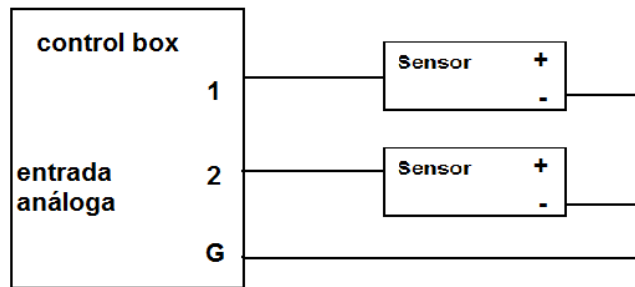
CONSIDERACIONES:

- Puede usar una fuente de 12 V corriente directa para alimentar al sensor.
- Las entradas del controlador son fabricadas en principio para sensores tipo NPN, para utilizar sensores tipo PNP cambie la configuración de los JUMPER en la entrada
- Si el rango de voltajes del sensor tipo NPN oscilan entre 0 -1.5V la entrada es considerada como activa.
- Si el intervalo de voltajes está entre 2.5 – 24V el estado de la entrada es inactiva.
- No conecte voltajes superiores a 24V a las entradas digitales del controlador lógico programable.

4.2.5. Entradas Análogas

Las entradas análogas permiten recibir información de sensores análogos. Estas entradas poseen una resolución de 8 bit y el rango de voltaje en el cual trabaja oscila entre 0 -10V. Esta información es leída por medio de los comandos del software.

Figura 4.2.5. Esquema de conexión.



Nota:

- Antes de conectar un sensor análogo asegúrese que el interruptor del controlador este en estado de apagado.
- No conecte voltajes a las entradas análogas que excedan los 10V.

4.3. PRÁCTICA DE LABORATORIO

4.4. SALIDAS Y ENTRADAS DEL PLC

4.4.1. Objetivo general de aprendizaje

Conocer la interfaz de salidas y entradas del PLC

4.4.2. Objetivos específicos de aprendizaje

- Programar las salidas del PCL con SCORBASE.
- Activar de manera manual las salidas digitales del PLC con SCORBASE.

4.4.3. Concepto importante: Salidas y entradas

El PLC tiene 8 salidas digitales y 8 entradas digitales el cual se pueden visualizar en una interfaz de SCORBASE. Este software permite programar las salidas del PLC y también leer las variaciones de los estados en las entradas.

4.4.4. Procedimiento

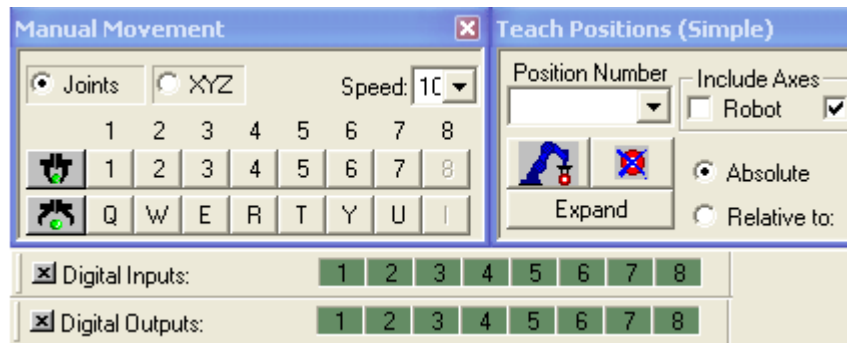
Para este laboratorio debe contar con:

- Software SCORBASE.
- Controlador lógico programable Intelitek.

Siga los siguientes pasos:

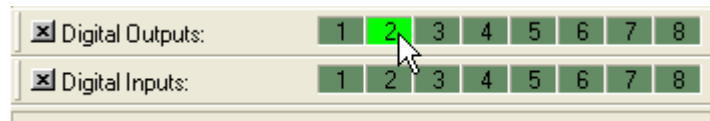
1. Ingrese a SCORBASE y haga clic en **View** en la barra de menú.
2. Diríjase a **Dialog Bars** y haga clic en **Digital Outputs** y **Digital Inputs**.
3. Observe el nuevo cuadro que muestra el estado actual de las salidas y entradas digitales del PLC.

Figura 4.4.1. Estado actual de entradas y salidas del PLC⁵⁹.



4. Haga clic en una de las salidas digitales, esta se activará en la interfaz y en el PLC.

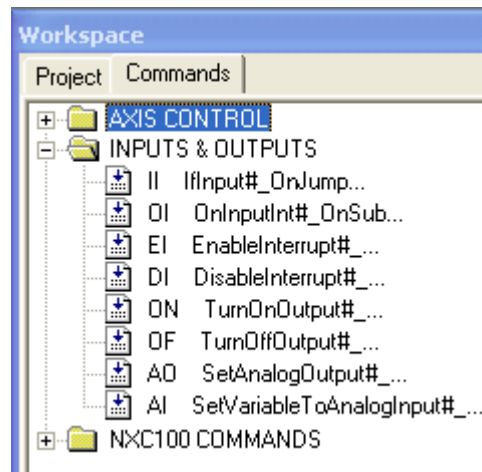
Figura 4.4.2. Salida digital numero 2 activada con el cursor del mouse.



5. Ingrese en el **Workspace** y despliegue los comandos de **INPUTS** y **OUTPUTS**.

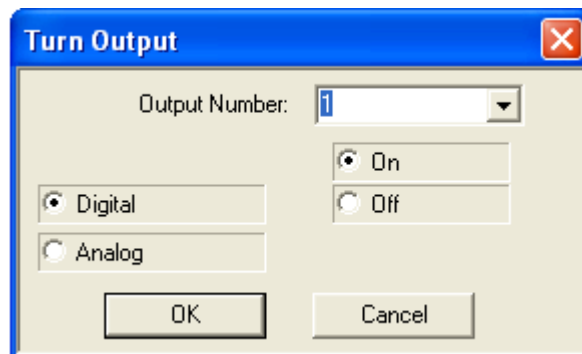
Figura 4.4.3. Comandos para el PLC.

⁵⁹ Fuente: Imagen capturada del software SCORBASE for NXC100.



Observe los comandos **TurnOnOutput#** y **TurnOffOutput#** estos sirven para activar o desactivar cualquier salida digital del PLC.

1. Abra un nuevo proyecto y haga doble clic en **TurnOnOutput#** deberá aparecer el siguiente cuadro:



En Output Number selecciona la salida y el estado, encendido o apagado. Haga clic en OK.

2. Agregue un temporizador de 5 segundos

3. Haga doble clic en **TurnOffOutput#** y seleccione la salida 1 en off. Deberá tener algo como:

Turn On Output 1

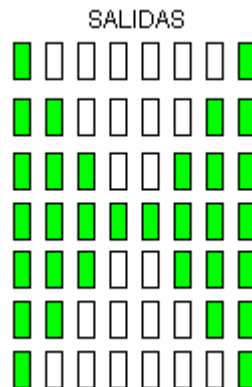
Wait 50 (10ths of seconds)

Turn Off Output 1

Ejecute el programa y observe la salida 1 de la interfaz de SCORBASE y del PLC, esta deberá encenderse y pasados 5 segundos apagarse.

4. Ejercicio.

- a) Con las salidas del PLC realice las siguientes secuencias:



El proceso debe ser repetitivo para ello utilice un Jump y temporizaciones de 0.5 segundos.

- b) ¿Las salidas de la secuencia del ejercicio 9 esta sincronizada? Explique.

4.4.5 Conclusiones

4.5. PRÁCTICA DE LABORATORIO

4.6. USO DE CONDICIONALES

4.6.1. Objetivo General de aprendizaje

- Comprender la funcionalidad del comando `IfInput#_OnJump` y su sintaxis dentro la ejecución de un programa.

4.6.2. Objetivos Específicos de aprendizaje

- Monitorear las entradas del controlador lógico programable con los comando de condicionalidad.
- Manipular subrutinas y saltos con el comando de condicionalidad **`IfInput#_OnJump`**.

4.6.3 Concepto importante:

SCORBASE permite realizar operaciones con base al estado de las entradas del PLC, por lo general utilizadas para tomar decisiones frente a eventos capturados por sensores.

Dentro de su repertorio de instrucciones Scorbases incluye comandos condicionales que evalúan las entradas para posteriormente tomar una acción que programemos previamente.

Los condicionales han jugado un papel importante a lo largo del desarrollo de la programación debido a que permiten evaluar al programador situaciones premeditadas para posteriormente realizar acciones que se requieran en el momento.

SCORBASE incluye dentro su repertorio de instrucciones el comando **IfInput#_OnJump**. El cual permite evaluar el estado de las entradas digitales del controlador y tomar alguna decisión que se requiera en la aplicación que el programador este realizando. Este comando puede ir acompañado de un salto o una subrutina en caso de que la entrada cumpla con la condición predetermina por el software.

Su sintaxis está dada de la siguiente forma.

If Input (número de la entrada digital) On/Off Call Subroutine [nombre subrutina]

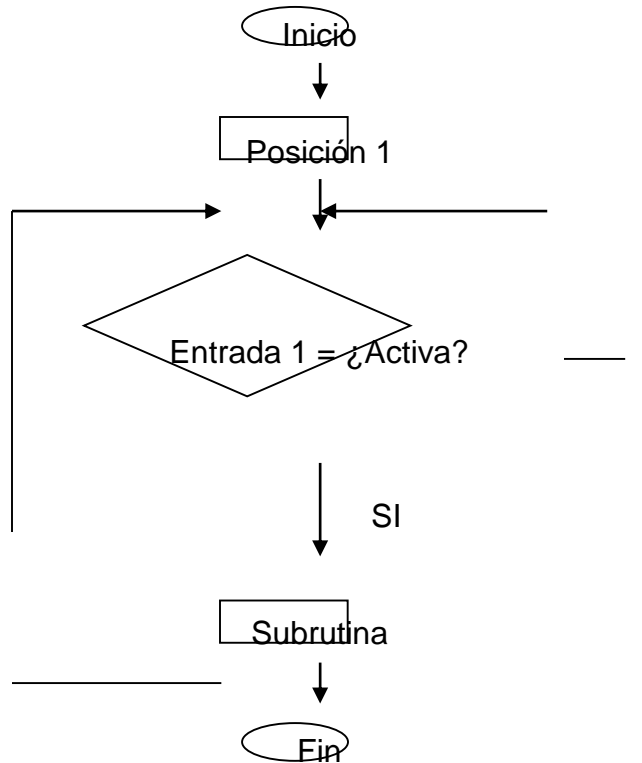
Esta sintaxis es válida siempre y cuando después que una de las entradas digital este activa/ inactiva proceda a realizar una subrutina previamente programada.

If Input (número de la entrada digital) On/Off Jump to [nombre de la etiqueta]

Esta se utiliza cuando queramos alterar la secuencia de un programa con base al estado de las entradas.

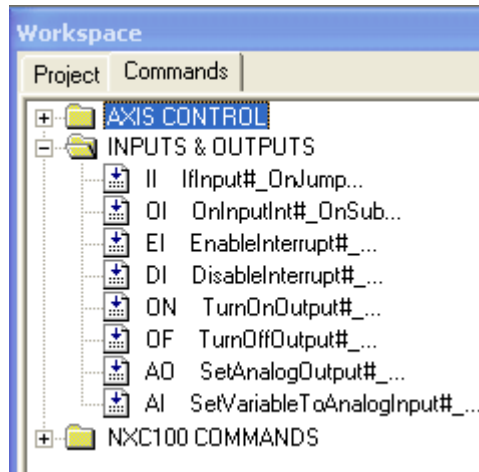
4.6.4 Procedimiento

El siguiente esquema representa la secuencia de un proceso en donde se evalúa el estado de la entrada digital 1 para posteriormente realizar una acción.



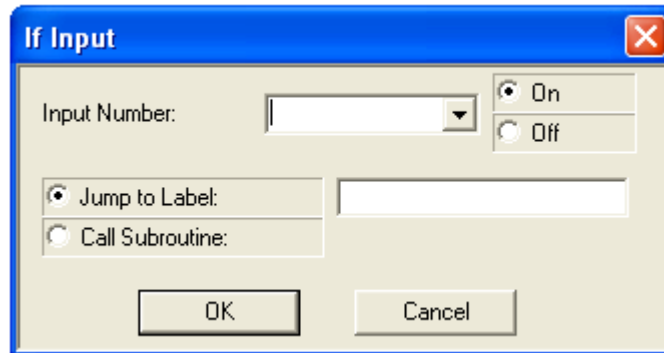
1. Cree una posición de banda segura el cual garantice que el manipulador no sufra alguna colisión con los demás elementos de la Celda. Posteriormente incluya en la secuencia del programa con el comando **Go to Position (n) Fast** la posición previamente guardada. Procedimiento explicado en anteriores laboratorios.
2. Diríjase a los comando que involucran entradas y salidas y seleccione el comando **IfInput#_OnJump**.

Figura 4.6.1. Comandos para el PLC.



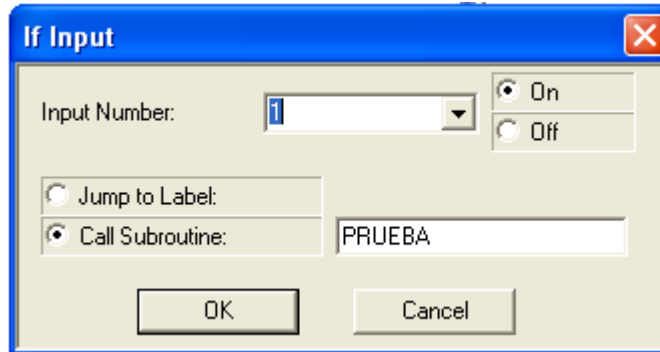
3. Al seleccionar la instrucción **IfInput#_OnJump** debe aparecer una ventana como la siguiente figura.

Figura 4.6.2. Comando **IfInput#_OnJump**.



4. Elija la entrada digital que va a validar y el estado en el cual el condicional va responder, en caso de existir una variación en las terminales de entrada. En este caso seleccione la entrada digital 1 y el estado activo (ON). Posteriormente elija que el caso de que la condición se cumpla llame a una

Subrutina, por esa razón escoja la opción **Call Subrutine** y escriba en frente el nombre de la subrutina. Presione ok.



Una vez terminado este procedimiento presione OK y automáticamente se genera la línea de código correspondiente.

5. Para que la entrada sea monitoreada en cada instante es necesario acudir a los saltos y las etiquetas. Para esto incluimos una etiqueta antes de la línea de código correspondiente al condicional y una instrucción de salto posterior a esta. Ejemplo:

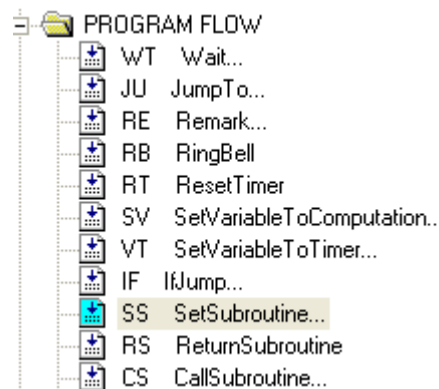
INICIO:

If Input 1 On Call Subroutine PRUEBA

Jump to INICIO

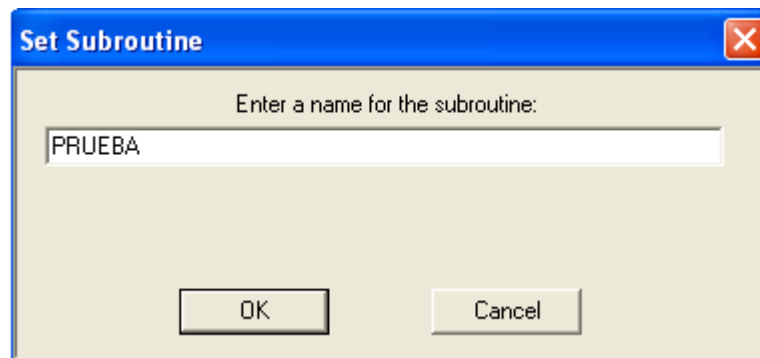
6. En caso de que la condición sea válida es necesario programar la subrutina la cual se va a ejecutar en el proyecto. Para esto nos dirigimos al **Workspace** y seleccionamos los comandos que gestionan el flujo del programa y elija la instrucción **SetSubrutine**.

Figura 4.6.3. Comando setSubrutine.



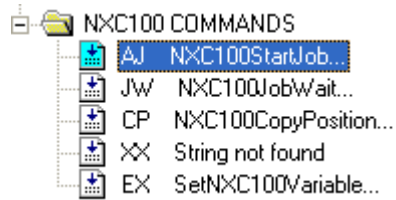
Una vez seleccionada no aparece una ventana en donde le puede asignar el nombre de tu subrutina para este ejemplo le asignamos PRUEBA para ir acorde al llamado que se realiza desde la línea de comando del condicional.

Figura 4.6.4. Nombre de la subrutina.



Elija un JOB o trabajo realizado previamente desde Teach Pendant. Para esta práctica se eligió un trabajo llamado EJEMPLO almacenado en el Teach Pendant. En el Workspace seleccione el comando NXC100 StartJob.

Figura 4.6.5. Comando NXC100StartJob.



Una vez escogida esta instrucción aparece una ventana en donde se asigna el nombre del trabajo realizado en el Teach Pendant, haga clic en ok y automáticamente se genera la línea de código. Para que el trabajo importado desde la interfaz del controlador de lenguaje avanzado se ejecute sin ningún tipo de problema se debe asignar un periodo de tiempo, para mayor detalle diríjase al laboratorio 3.18.

7. Para finalizar la subrutina se debe colocar un retorno para que el programa tome la secuencia normal antes de ejecutar el subprocesso. Para mayor detalle consulte las prácticas 3.18.

INICIO

Go to Position 1 Fast

INICIO1:

If Input 1 On Call Subroutine PRUEBA

Jump to INICIO1

End

Set Subroutine PRUEBA

NXC100 Start Job EJEMPLO

NXC100 Job Wait 40 (seconds)

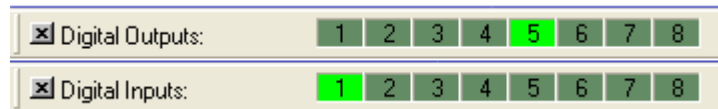
Return from Subroutine

Nota:

Observe que la subrutina esta por fuera de la estructura del programa principal, esta es la forma correcta de plantear un subprocesos en SCORBASE.

Hasta este punto se ha fundamentado el programa que monitorea el estado de una entrada en específico. Una forma de simular el evento en dado caso no se posean sensores, es forzando el estado de cada entrada y una manera sencilla es conectando directamente una salida de colector abierto a la entrada digital con la que desee trabajar y manipular la activación desde el cuadro de salidas que presenta Scorbace.

Figura 4.6.6. Entrada 1 activada.



En la figura anterior se observa que la salida de colector abierto está forzando el estado de la entrada digital 1 del controlador. Esto es útil puesto que con dar un solo clic sobre la salida se puede simular el comportamiento de un sensor.

Consideraciones:

- Solo es posible conectar salidas de colecto abierto (5-8) con entradas digitales del PLC para propósitos de simulación.
- Se recomienda mirar los esquemas de conexiones antes de proceder a articular dispositivos a las entradas o salidas del PLC.

4.6.5. Conclusiones

4.7. PRÁCTICA DE LABORATORIO

4.8. INTERACCIÓN CON LOS PUERTOS ANÁLOGOS DEL CONTROLADOR LÓGICO PROGRAMABLE

4.8.1 Objetivo general de aprendizaje

Programar los puertos análogos de controlador lógico programable.

4.8.2. Objetivos específicos de aprendizaje

- Utilizar el multímetro.
- Hacer mediciones de voltaje en las salidas de análogas del PLC.
- Relacionar valores análogos con valores discretos.

4.8.3. Concepto importante:

El mundo real está lleno de parámetros físicos que varían en el tiempo de forma continua, estas magnitudes en principio generan una labor un poco tediosa a la hora de medir, almacenar, y al recuperar la información que se desea extraer; es por esa razón que el formato digital ha tomado gran acogida en la actualidad porque facilita considerablemente los procesos anteriormente nombrados.

El controlador lógico programable con que cuenta la Celda de Manufactura posee dos salidas análogas y tres entradas de esta misma naturaleza, las cuales son comandadas por las instrucciones **SetVariableToAnalogInput#** y **SetAnalogOutput#**, que a continuación serán descritas en mayor detalle.

La Instrucción **SetAnalogOutput#**: es el comando que permite manipular las terminales de salidas análoga del PLC seleccionando el puerto con el cual se desea trabajar y además permite asignar el valor del voltaje asociando con una codificación

en binario que posee un rango de 0- 255 que corresponde al máxima cantidad de combinaciones posible en un grupo de 8 bits.

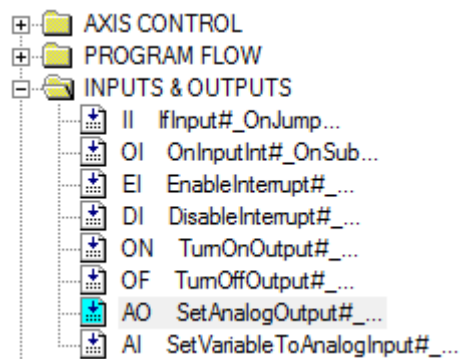
4.8.4. Procedimiento

Para esta práctica debe contar con:

- PLC Intelitek.
- Multímetro (Voltaje).

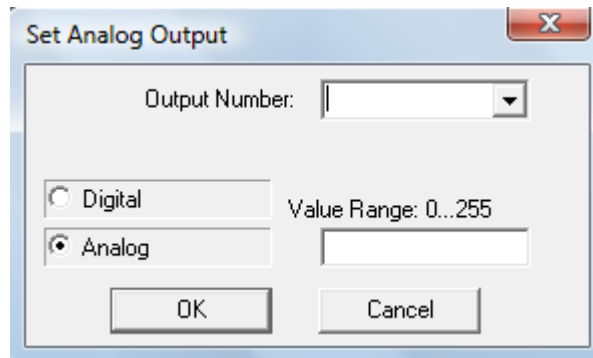
Siga los siguientes pasos:

1. Abra un nuevo proyecto en SCORBASE.
2. Como es una instrucción que está relacionado con terminales diríjase a **WorkSpace** y seleccione la opción **INPUTS & OUTPUTS** el cual despliega un repertorio de comandos que permiten gestionar las distintas terminales del controlador. La siguiente figura ilustra el procedimiento descrito anteriormente.



3. Elija el comando **SetAnalogOutput#**, aparecerá una ventana como en la siguiente figura.

Figura 4.8.1. Comando SetAnalogOutput#.



4. Escoja el tipo de salida que desea manipular, para este caso la opción **Analog** aparece seleccionada por defecto. En la caja de cuadro combinado Output Number seleccione la salida que desea controlar para el ejemplo elija la salida 1, y escriba un valor arbitrario dentro del rango de 0 -255 y presiones Ok.
5. Realice una medición de voltaje con un multímetro en la salida análoga la cual usted ha escogido previamente. ¿Qué valor obtuvo?
6. Ejercicio.

Realice 10 mediciones en las cuales considere los extremos del rango el cual permite el Software, complete el siguiente cuadro.

Numero del binario representado en decimal (0-255)	Valor del voltaje obtenido

7. Ejercicio.

a) Inicie desde cero e incremente en 25 unidades el decimal. ¿En qué proporción aumenta el valor del voltaje?

b) ¿Qué valor de voltaje toma cuando la cantidad llega a 255?

c) ¿En qué proporción aumenta el voltaje en la medida que incrementa cada unidad del valor en decimal?

4.8.5. Conclusiones

CAPÍTULO 5

5 APLICACIONES

5.1 INTRODUCCIÓN DEL CAPÍTULO

En este capítulo se estudiarán y se realizarán algunas aplicaciones que se pueden desarrollar con la celda de manufactura. Las aplicaciones estarán divididas en: aplicaciones de manipulador o brazo robótico y aplicaciones de Celda de manufactura. En la siguiente práctica de laboratorio se analizará un tipo de aplicación multipropósito con el brazo robotico

5.2 PRÁCTICA DE LABORATORIO

5.3 APLICACIÓN MULTIPROPÓSITO

5.3.1 Objetivo general de aprendizaje

Simular la trayectoria de una herramienta de soldadura de arco con el manipulador o brazo robótico.

5.3.2 Objetivos específicos de aprendizaje

- Utilizar movimiento lineal, circular y esférico para trazar trayectorias.
- Utilizar un marcador como representación de herramienta de soldadura.
- Hacer uso de un tablero acrílico para simular el material a soldar.

5.3.3 Concepto importante: Aplicaciones del manipulador

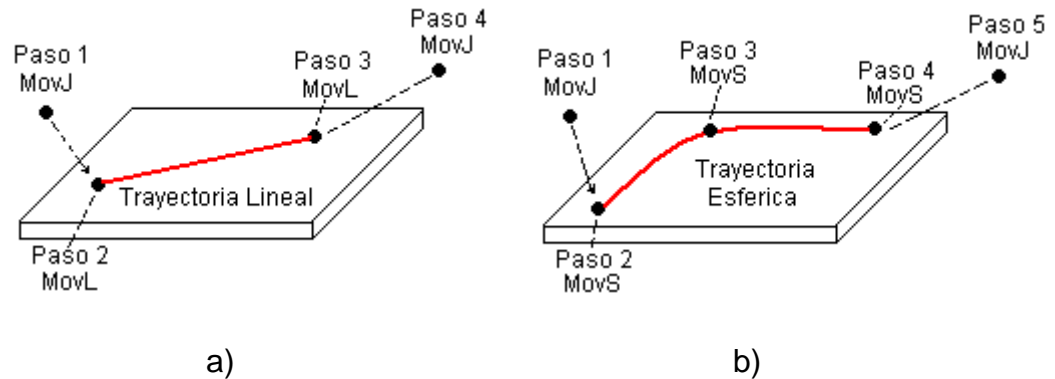
En las industrias hay variedad de aplicaciones que utilizan brazos robóticos tales como, la soldadura de punto, soldadura de arco, cortado de piezas, pintado, entre otras, esto es posible simplemente variando la herramienta del brazo o manipulador.

En el laboratorio de procesos industriales de la universidad del magdalena estas herramientas anteriormente mencionadas no se encuentran disponibles, pero esto no es impedimento para programar y crear una simulación para soldar, pintar o cortar un material.

En las aplicaciones multipropósito es muy importante tener en cuenta el comportamiento del robot con los diferentes tipos de movimientos LINEAL, CIRCULAR Y ESFERICO, para las aplicaciones con algún instrumento lo más conveniente es utilizar coordenadas de herramienta el cual brinda un mejor manejo y dominio de los movimientos del robot.

En una superficie plana el robot puede trazar una línea con el movimiento LINEAL y solo será necesario dos puntos o pasos. Para trazar una trayectoria esférica o circular es necesario mínimo 3 pasos en la superficie plana.

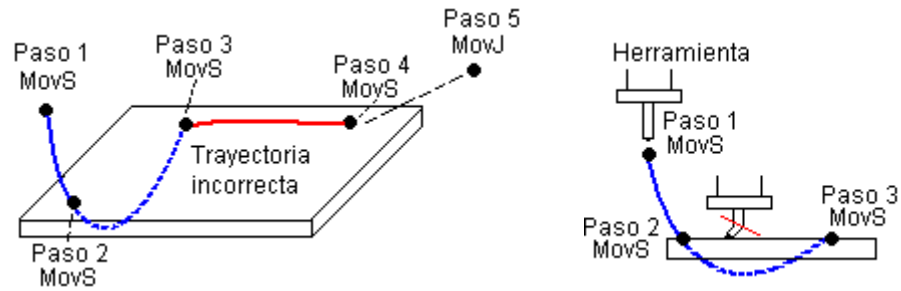
Figura 5.3.1. a) Trayectoria lineal con dos pasos en la superficie. b) Trayectoria esférica con 3 pasos en la superficie⁶⁰.



Observe que en la figura 5.3.1 el paso 1 está en movimiento Joint, esto es muy importante porque este primer punto no interfiere en la trayectoria que se quiere trazar en la superficie, en la figura 5.3.1 el paso 1 inicia en movimiento esférico lo que alteraría la trayectoria y por consecuencia la herramienta se ve afectada y fácilmente podría romperse.

⁶⁰ Fuente: Elaboración propia.

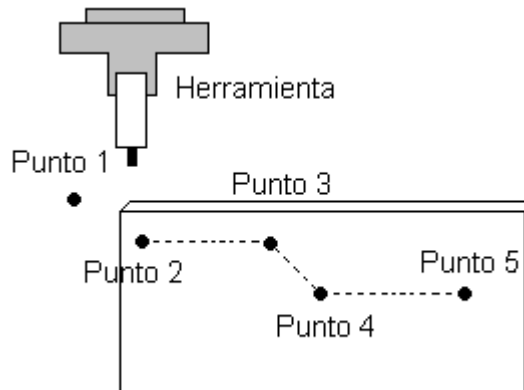
Figura 5.3.2. Error que no se debe cometer con el paso 1⁶¹.



5.4.4 Procedimiento

Para este laboratorio se utilizará una herramienta que consiste en un marcador borrable que representará un dispositivo para soldadura de arco. (marcador de 11.5 cm de longitud)

Figura 5.3.3. Trayectoria de soldadura⁶¹.



Para este laboratorio debe contar con:

- Controlador lógico programable.
- Banda deslizante.

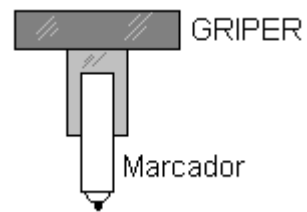
⁶¹ Fuente: Elaboración propia.

- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Marcador borrable.
- Tablero acrílico.

Siga los siguientes pasos:

1. Abra el griper del robot y coloque un marcador, luego cierre el griper, el marcador hará el papel de la herramienta para soldar.

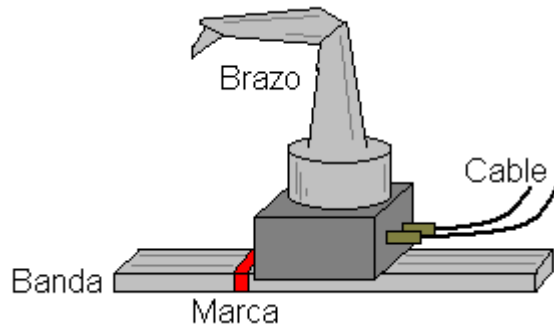
Figura 5.3.4. Marcador sostenido por el Gripper⁶².



2. Aproxime el robot mediante la banda deslizante hasta la marca que se muestra en la banda aproximadamente al frente del tablero acrílico.

⁶² Fuente: Elaboración propia.

Figura 5.3.5. Brazo robótico posicionado en la marca de la banda deslizante⁶³.



3. Utilice las coordenadas de herramienta del **Teach Pendant** para ubicar los puntos, guíese de la figura 5.3.3. El primer punto será el de partida y se ubicará en la esquina superior izquierda del Tablero acrílico. El segundo punto debe estar en la esquina superior izquierda del Tablero en blanco el cual representa la materia que será soldada, el marcador debe apenas tocar el tablero, grabe los demás puntos, todos con movimiento lineal.
4. El programa en el **Teach Pendant** deberá tener la siguiente estructura:

```
000  NOP
0001 MOVJ    VJ = 30.00  "Punto 1, posición inicial"
0002 MOVL    V = 10.0    "Punto 2"
0003 MOVL    V = 5.0     "Punto 3"
0004 MOVL    V = 5.0     "Punto 4"
0005 MOVL    V = 5.0     "Punto 5"
0006 MOVJ    V = 60.0    "Posición segura"
0007 END
```

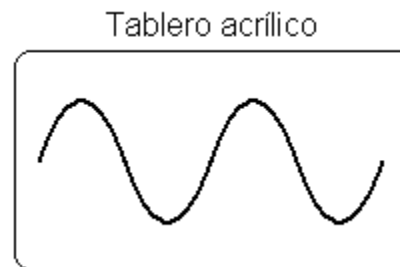
⁶³ Fuente: Elaboración propia.

Figura 5.3.4. Trazo del marcador en el tablero acrílico⁶⁴.



5. Ejercicio 1.

a) Trace la siguiente trayectoria con movimiento esférico.



b) ¿Cuántos puntos se deben utilizar?

c) ¿Cuál es la diferencia de trazar las trayectorias con movimiento lineal o movimiento esférico?

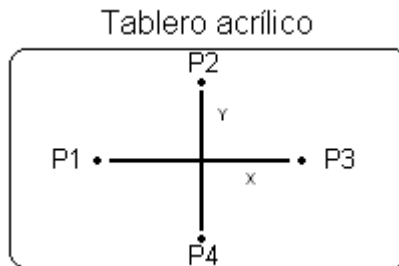
⁶⁴ Fuente: Elaboración propia.

- d) Repita el proceso con los mismos puntos y utilice movimiento lineal, dibuje la trayectoria resultante en el tablero acrílico.



6. Ejercicio 2.

- a) Trace la siguiente trayectoria con movimiento circular.



- b) ¿Cuál es la figura o trayectoria resultante?

5.4.5 Conclusiones

5.4 APLICACIONES DE ENSAMBLAJE

La principal aplicación de la celda de manufactura en el laboratorio de procesos industriales de la Universidad del Magdalena es de ensamblaje, por esta razón se desarrollarán algunos procesos o modelos representativos de la producción en las industrias que manejan la robótica industrial.

Una de las partes fundamentales de la celda de manufactura es el **ASRS** o sistema de almacenaje y recuperación automatizada, que funciona como la principal fuente de materia prima en la celda, el **ASRS** contiene plantillas o Templates donde se inserta o guarda la materia prima. Las plantillas con el material se llevan hacia el buffer es ahí donde se toma la materia prima y se lleva hacia el rack, en este paso ahora se llama materia en proceso, el producto final se obtiene ensamblando las partes en la prensa de la celda.

5.5 PRÁCTICA DE LABORATORIO

5.6 APLICACIÓN DE ENSAMBLAJE 1

5.6.1 Objetivo general de aprendizaje

Ensamblar las materias primas que se encuentran en la plantilla 9 y 11 del ASRS.

5.6.2 Objetivos específicos de aprendizaje

- Identificar las partes del proceso de ensamblado.
- Describir las posiciones y subrutinas del programa de ensamblado.
- Desarrollar un proceso de ensamblado con el Index 5 e Index 7.

5.6.3 Concepto importante: Partes de la celda de manufactura

En la celda de manufactura del laboratorio de procesos industriales de la Universidad del Magdalena existen tres partes fundamentales que son:

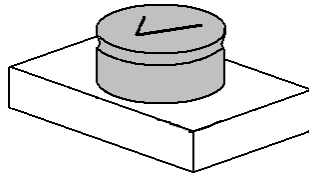
Almacenamiento de Materia prima

Materia en proceso

Ensamblaje o producto final

El almacenamiento de la materia prima se hace en el **ASRS**, se le llama materia en proceso únicamente cuando las partes o materia prima se encuentran en el Buffer y Rack. La combinación de las materias primas llega a ser un producto terminado cuando es ensamblado en la prensa.

Figura 5.6.1. Producto ensamblado o terminado⁶⁵.



5.6.4. Procedimiento

En este laboratorio se mostrará el proceso de las tres paradas que hace la materia prima en la celda para llegar al producto final. Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Prensa.
- Llave MOTOCOM.
- Materia prima.

El proceso se describe a continuación:

1. Se captura con el brazo robótico las plantillas que contienen la materia prima para posteriormente llevarlas al Buffer.
2. Desde el Buffer se lleva cada materia prima al Rack correspondiente.

⁶⁵ Fuente: Elaboración propia.

3. Desde el Rack se captura las materias primas y se llevan a la prensa, es aquí donde se obtiene el producto final.
4. Sacar el producto final de la prensa y mostrarlo.
5. Para ver este proceso ingrese a SCORBASE, luego abra el archivo **ENSAMBLE_1** de la carpeta **LABORATORIO ESTUDIANTES** en el escritorio del PC.
Ver Código de ENSAMBLAJE_1 en anexos.

6. Ejercicio.

- a) Haga una descripción de cada subrutina del programa ENSAMBLAJE_1.

- b) Haga una descripción de cada posición del programa ENSAMBLAJE_1.

- c) Desarrolle un programa similar a ENSAMBLAJE_1 con la condición de ensamblar las materias primas del Index 5 e Index 7.

d) Describa las subrutinas y posiciones que utilizó.

5.7.5 Conclusiones

5.7 PRÁCTICA DE LABORATORIO

5.8 APLICACIÓN DE ENSAMBLAJE 2

5.8.1 Objetivo general de aprendizaje

Llevar todas las materias primas que se encuentran almacenadas en el armario o ASRS al RACK1 y RACK2.

5.8.2 Objetivos específicos de aprendizaje

- Comprender el programa que lleva las materias primas al RACK 1 y 2.
- Describir las posiciones y subrutinas del programa de ensamblado.
- Desarrollar un proceso de abastecimiento de materia prima en los RACKs.

5.9.3 Procedimiento

Para este laboratorio debe contar con:

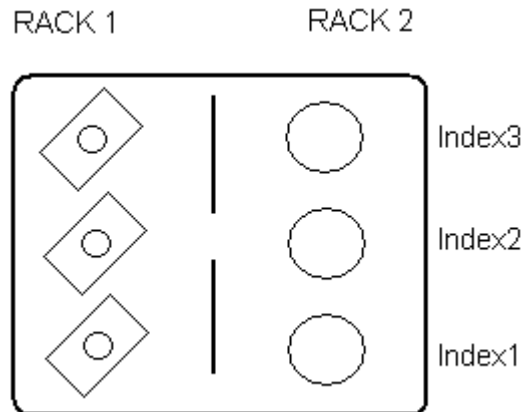
- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Llave MOTOCOM.
- Materia prima.

El proceso se describe a continuación:

1. El brazo robótico captura las plantillas con materias primas y las lleva al Index1 e Index2 del BUFFER.

2. En el BUFFER, el brazo captura las materias primas y las lleva hasta el RACK1 y RACK2, las materias son ubicadas de forma ascendente en los Index, de 1 hasta 3.

Figura 5.8.1. Ubicación correcta de materia prima en el RACK 1 y 2⁶⁶.



3. Las plantillas deben ser llevadas de vuelta al **ASRS** por el robot. El proceso se repite tres veces hasta que el RACK 1 y 2 queden abastecidos.
4. Para ver este proceso ingrese a SCORBASE, luego abra el proyecto **ENSAMBLE_2** de la carpeta **LABORATORIO ESTUDIANTES**.
Ver Código de ENSAMBLAJE_2 en anexos.
5. Ejercicio.
 - a) Haga una descripción de cada subrutina del programa ENSAMBLAJE_2.

⁶⁶ Fuente: Elaboración propia.

- b) Desarrolle un nuevo proceso similar a EMSANBLAJE_2 con la condición de colocar las materias primas en orden ascendente de 3 a 1 en el RACK.
- c) Describa las subrutinas y posiciones que utilizó.

5.9.4 Conclusiones

5.9 PRÁCTICA DE LABORATORIO

5.10 APLICACIÓN DE ENSAMBLAJE 3

5.10.1 Objetivo general de aprendizaje

Ensamblar las materias en proceso que soporta el RACK1 y RACK2.

5.10.2 Objetivos específicos de aprendizaje

- Comprender el programa de ensamblado de las materias en proceso.
- Describir las posiciones y subrutinas del programa de ensamblado.

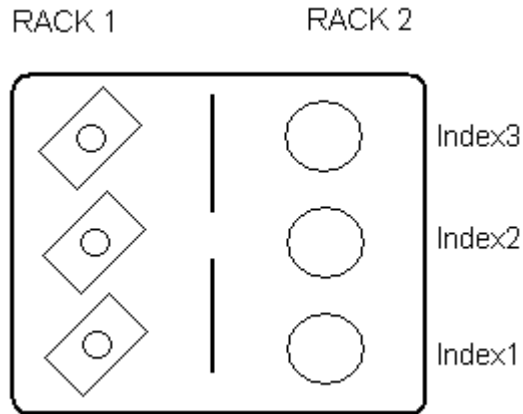
5.11.3 Procedimiento

Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Llave MOTOCOM.
- Materia prima.
- Prensa.

1. Basteza de forma manual el RACK1 y RACK2 con materia prima.
2. La materia prima en el RACK 1 y 2 debe quedar como lo muestra la figura 5.13.1.

Figura 5.13.1. RACK1 y RACK2 mostrando la correcta ubicación de la materia prima⁶⁷.



3. Para ver este proceso ingrese a SCORBASE, luego abra el proyecto **ENSAMBLE_3** de la carpeta **LABORATORIO ESTUDIANTES**.

Ver Código de ENSAMBLAJE_3 en anexos.

4. Ejercicio.

- a) Describa las posiciones y subrutinas del programa ENSAMBLE_3.

⁶⁷ Fuente: Elaboración propia.

5.11.4 Conclusiones

5.11 PRÁCTICA DE LABORATORIO

5.12 APLICACIÓN DE ENSAMBLAJE 4

5.12.1 Objetivo general de aprendizaje

Desarrollar una línea de ensamblaje completa de la celda de manufactura.

5.12.2 Objetivos específicos de aprendizaje

- Comprender el programa de ensamblaje.
- Describir las posiciones y subrutinas utilizadas en el programa de ensamblado.
- Determinar los tiempos de ensamblado.
- Mejorar los tiempos de ensamblado.

5.12.3 Procedimiento

Para este laboratorio debe contar con:

- Computadora 1.
- Controlador lógico programable.
- Banda deslizante.
- Controlador NXC100.
- Control de mando del NXC100.
- Compresor de aire.
- Llave MOTOCOM.
- Materia prima.
- Prensa.

1. Prepare la celda de manufactura ubicando las plantillas y las materias primas en los lugares que corresponde.
2. Prepare los dispositivos que se necesitan.
3. Ejecute el programa de nombre **ENSAMBLE_4** de la carpeta **LABORATORIO ESTUDIANTES**.
4. Observe y analice el proceso de ensamblado.
5. Ejercicio.

a) Describa las posiciones y subrutinas del programa ENSAMBLE_4.

b) Calcule los tiempos en segundos de: sacar las plantillas con la materia prima de los ASRS, colocar la materia prima en los RACKs y ensamblar toda la materia en proceso. ¿Cuál fue el tiempo total de ensamblado?

c) Desarrolle un proceso como el de ENSAMBLE_4 optimizando los tiempos.

d) Describa las posiciones y subrutinas que utilizó en el programa.

e) Describa los JOBs que utilizó en el **Teach pendant**.

5.12.4 Conclusiones
