

## PROGRAMACIÓN DE UNA CINTA TRANSPORTADORA MEDIANTE VARIADOR OMRON Y SOFTWARE CX-DRIVE



**Eloy Fernandez**

Alumno de 2º curso del Centro "I.E.S Palau Ausit" de Ripollet (Barcelona)

**Rubén Gea**

**INDICE**

	<b>PÁGINA</b>
1. Finalidad del presente manual.....	<b>3</b>
2. Configuración del nuevo controlador.....	<b>4</b>
3. ¿Qué práctica vamos a desarrollar y porqué?.....	<b>9</b>
4. Desarrollo del programa.....	<b>10</b>
5. Configuración parámetros del variador.....	<b>17</b>
6. Envío del programa y simulación.....	<b>18</b>


## **1. Finalidad del presente manual**

Lo que se pretende con este manual es dar a conocer las diferentes herramientas que el software cx-drive nos facilita para poder automatizar un proceso mediante un lenguaje de programación propio de este software sobre el mismo variador y evitar así la instalación de un plc.

No hace falta que el usuario que disponga de un variador OMRON y el programa CX-drive tenga un nivel avanzado para realizar esta práctica, simplemente ha de tener nociones de programación con estructuras tipo (if , then, while...wend, do loop, until, etc..) ya que si queremos realizar una aplicación más compleja de la que en este manual se va a desarrollar, se ha de entender este tipo de lenguaje de programación.

Evidentemente habrán aplicaciones donde lo expuesto en este documento no nos sirva, pero todo motor que sea controlado a través de un variador podremos programarlo como a continuación paso a explicar.

## 2. Configuración del nuevo controlador

Lo primero que haremos será abrir el Cx-drive, (programa que incorpora la suite de OMRON Cx-one) y pulsaremos, una vez abierto, sobre el icono  que se muestra en la barra de herramientas. Una vez pulsado veremos la siguiente imagen (Figura 1.0)

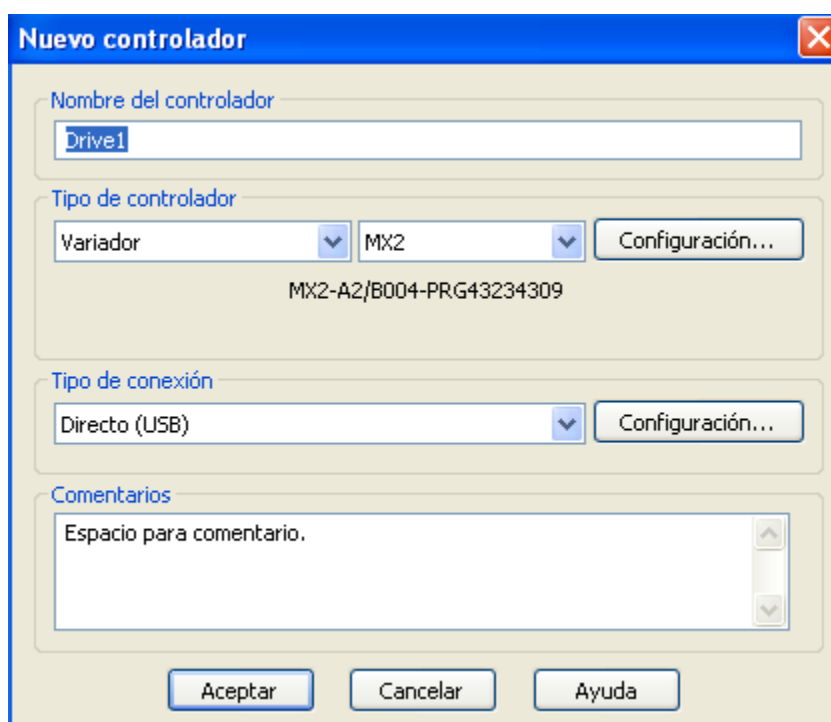


Figura 1.0

**Nombre del controlador:** Aquí le pondremos el nombre a nuestro proyecto. Por defecto Drive1.

**Tipo de controlador:** Podremos elegir entre variador o servo simplemente pulsando sobre el desplegable. Una vez elegido el tipo de controlador la casilla de al lado la podremos desplegar para elegir el modelo exacto de nuestro variador o servo.

**Configuración (Tipo de controlador):** Al pulsar sobre este botón accederemos a la siguiente pantalla (Figura 1.1)

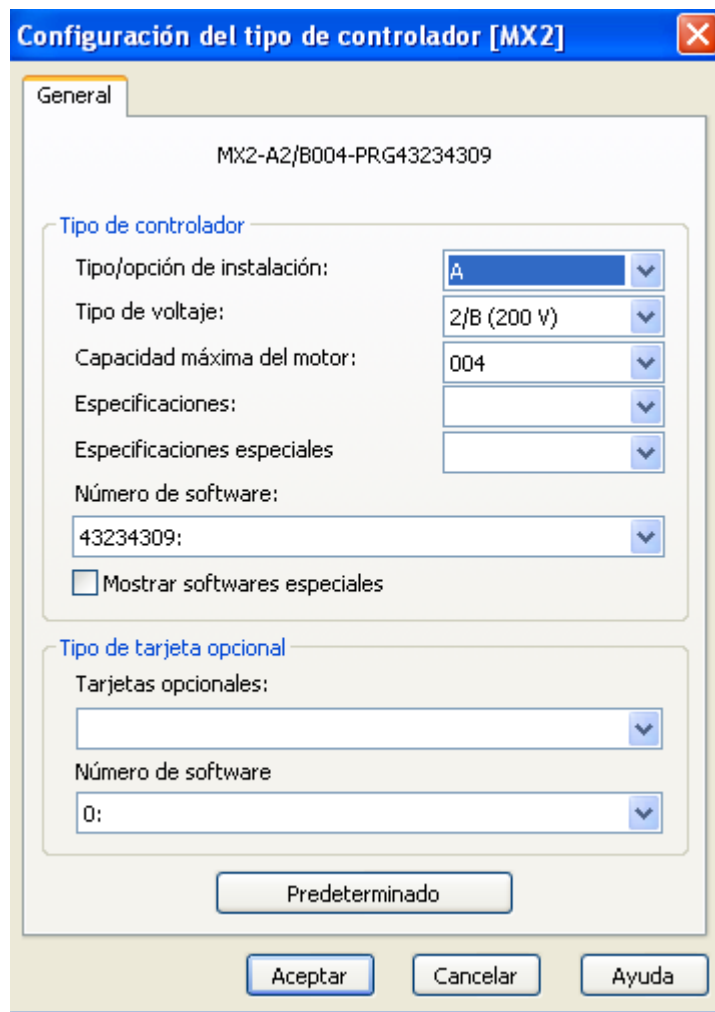


Figura 1.1

Donde podremos configurar dependiendo de nuestro controlador (variador o servo) una serie de opciones que incluye cada modelo, como por ejemplo, el tipo de voltaje, la capacidad máxima del motor, etc...

Una vez configurado todo correctamente pulsaremos sobre aceptar para volver a la anterior pantalla.

**Tipo de conexión:** Al pulsar sobre el desplegable de esta opción podremos ver los diferentes tipos con los que nos podremos conectar al controlador, en este caso con un variador. (Figura 1.2).

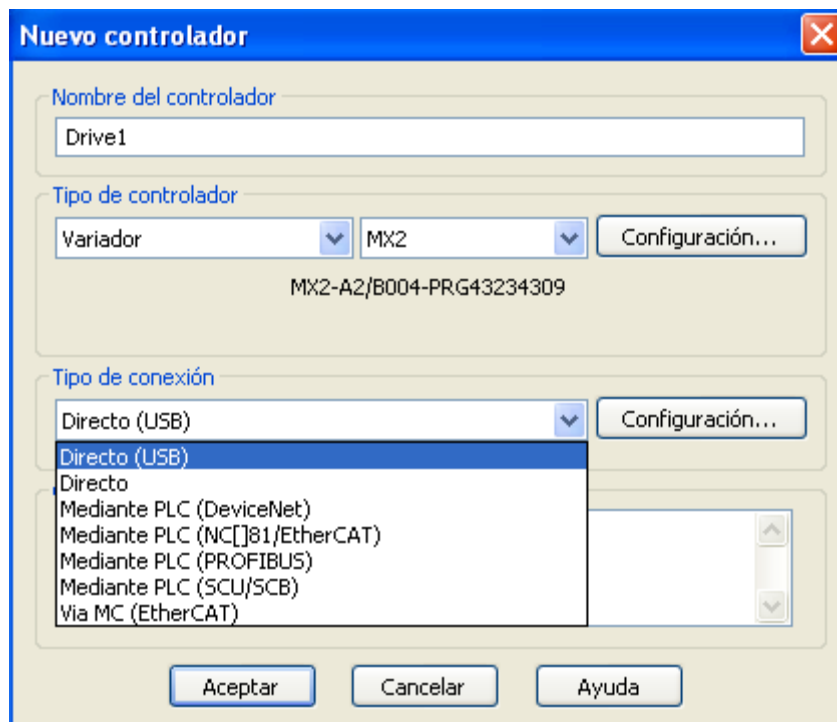
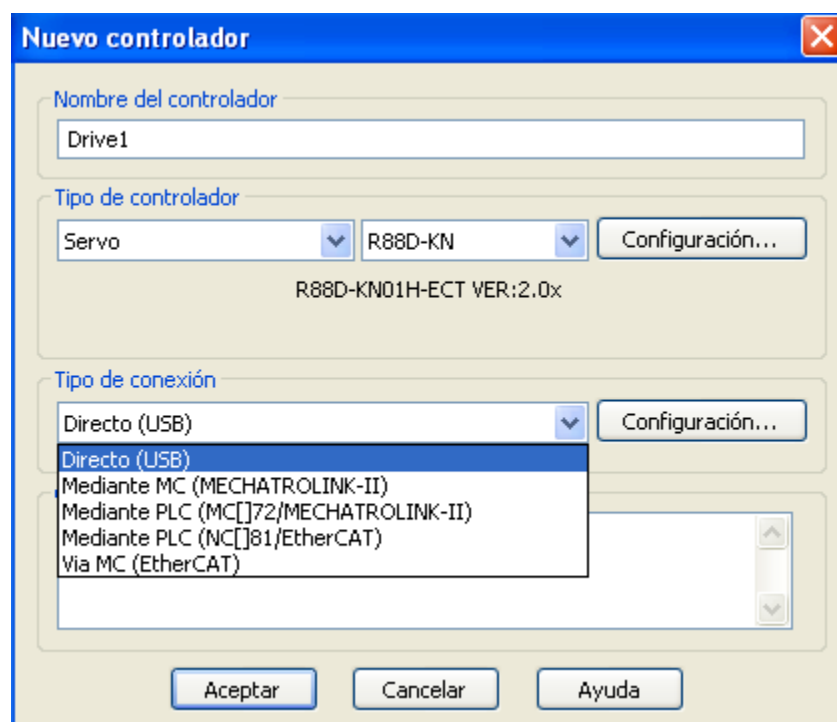


Figura 1.2

Si por el contrario hemos configurado como “**Tipo de controlador**” servo, veremos los diferentes tipos de conexión con los que podremos comunicar con el servo (Figura 1.3).



**Figura 1.3**

**Configuración (Tipo de conexión):** Al pulsar sobre este botón accederemos a la siguiente pantalla (Figura 1.4).



**Figura 1.4**

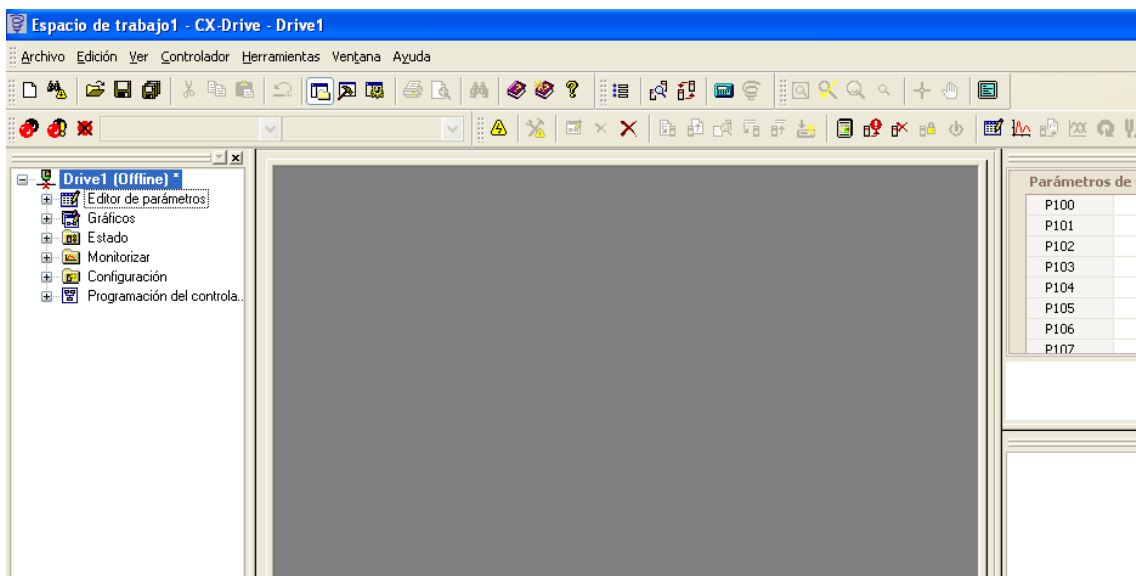
Donde podremos configurar nuestra red.

Esta pantalla será diferente según el tipo de controlador que tengamos y según el tipo de conexión que hayamos elegido previamente. En nuestro ejemplo, lo hemos hecho con un variador MX2 y conexión directa por USB.

Por último si queremos añadir algún comentario para nuestro proyecto lo colocamos sobre el espacio que esta ventana nos reserva para tal efecto.

Con esto tendríamos realizada la configuración del tipo de controlador con el que vamos a trabajar y el tipo de conexión con la que nos vamos a comunicar, por lo que tan solo tendríamos que pulsar el botón aceptar.

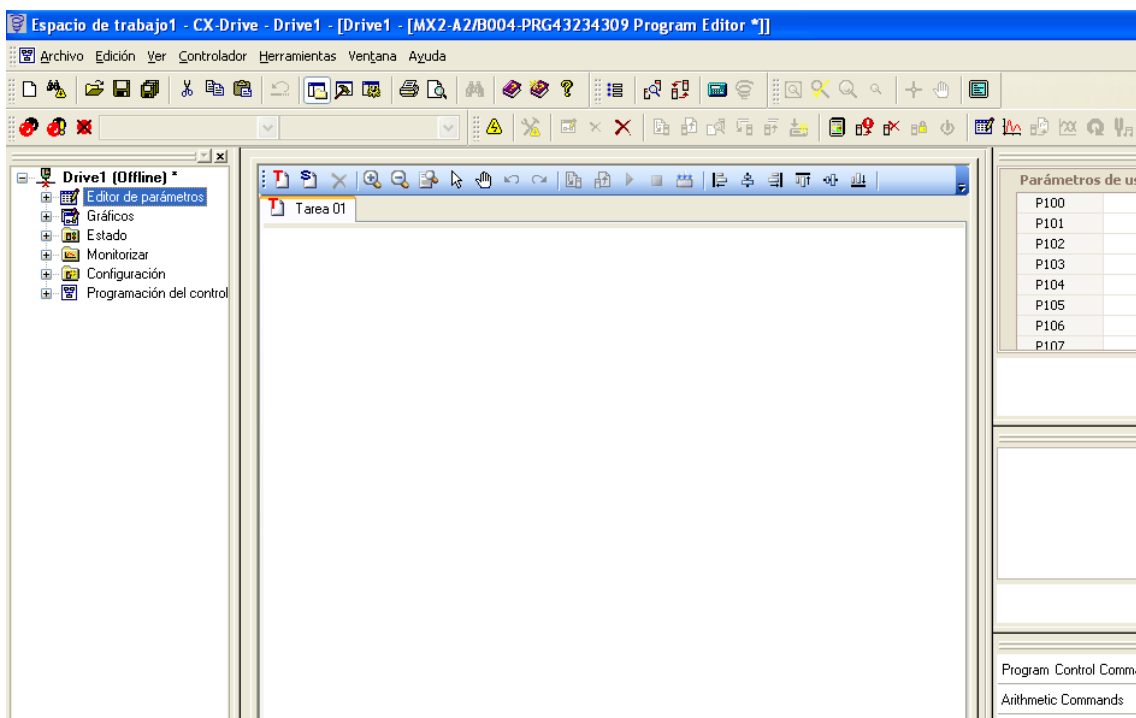
A continuación visualizaremos la siguiente pantalla (Figura 1.5)



**Figura 1.5**

Donde haremos doble click sobre la última opción que tenemos al desplegar nuestro proyecto (Programación del controlador).

Con esto entraremos en la interface de programación (Figura 1.6) donde podremos desarrollar nuestro programa y posteriormente enviárselo a nuestro variador o servo.



**Figura 1.6**



### 3. ¿Qué práctica vamos a desarrollar y porqué?

Antes de empezar a desarrollar la programación de nuestra aplicación, explicaré de qué trata, además de representarlo mediante una descripción grafica.

#### ¿ Qué programa vamos a desarrollar ?

Nuestra aplicación es bien sencilla, se trata de mover un contenedor a través de una cinta transportadora de un extremo a otro gracias a unos detectores que nos indicaran que el contenedor se encuentra al inicio o al final de la cinta. (Figura 1.7)

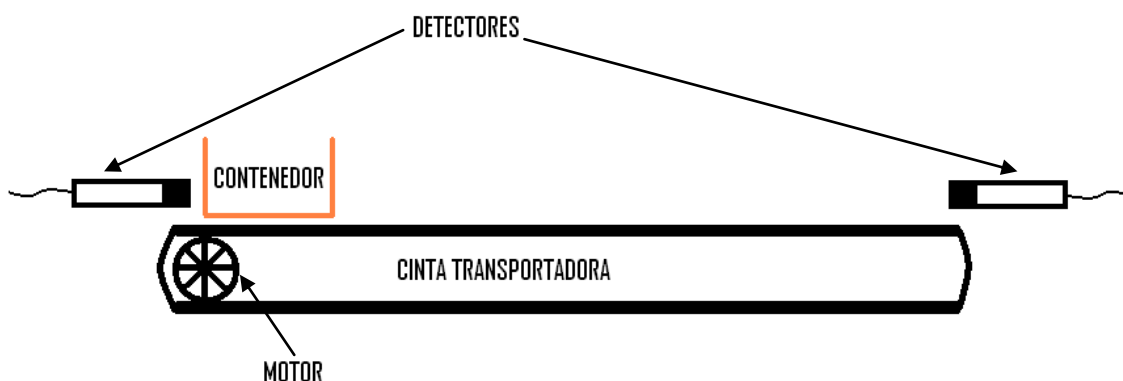


Figura 1.7

La secuencia sería la siguiente:

- 1) Al pulsar el botón de marcha, si está detectando el sensor del inicio de la cinta, ésta se pondrá en marcha.
- 2) Al detectar el sensor de final de la cinta, ésta se parara durante un tiempo (3 segundos).
- 3) Transcurrido este tiempo se pondrá en marcha de nuevo la cinta de forma inversa hasta detectar el sensor de inicio, donde se parará y no se pondrá en marcha de nuevo hasta que no pulsemos el marcha.

#### ¿ Porqué algo tan sencillo ?

La idea de este manual (como se ha dicho al principio de este documento) es la de enseñar las posibilidades que ofrece este software para controlar un proceso mediante un variador o servo sin la necesidad de instalar un PLC, para ello, se muestra una pequeña aplicación y partiendo

de esta base se puede experimentar un poco más para realizar un proceso complejo en el que intervengan varias entradas y varias salidas.

Dicho todo esto pasaremos al desarrollo de la práctica donde explicaremos en todo momento y con detalle los elementos que vamos introduciendo a la interface de programación.

#### 4. Desarrollo del programa

Antes de empezar, explicare brevemente cada una de las ventanas que componen la interface de programación (Figura 1.8).

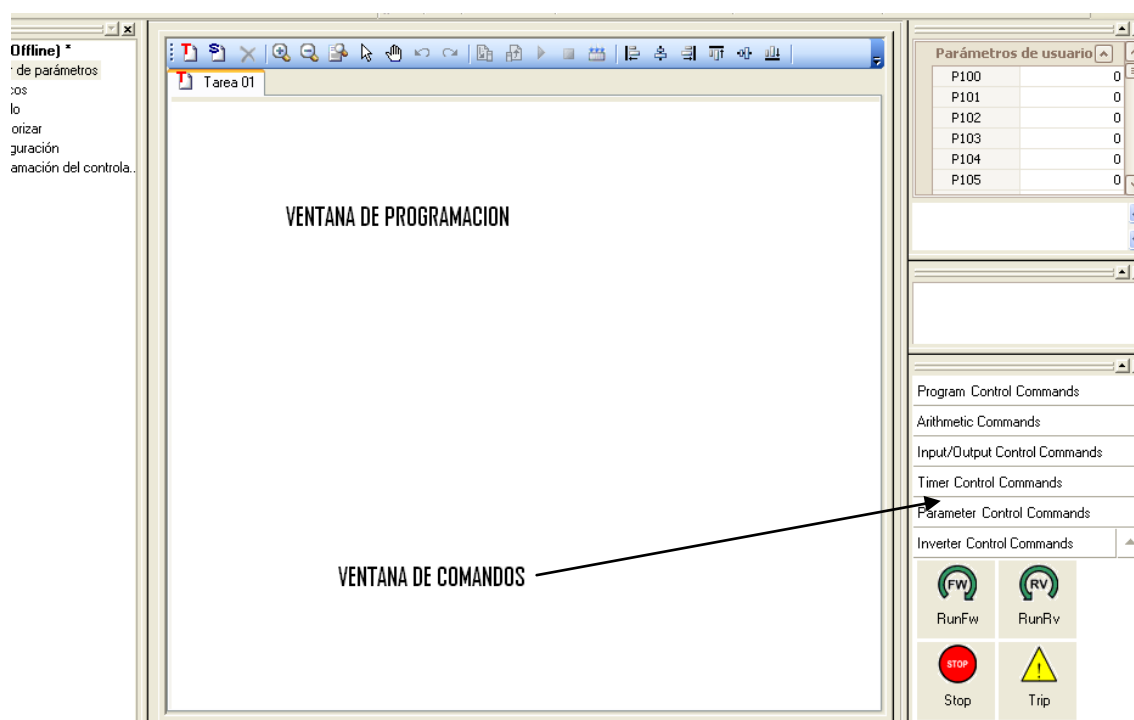


Figura 1.8

**Ventana de programación:** En esta ventana iremos colocando los diferentes comandos para el desarrollo de nuestro programa.

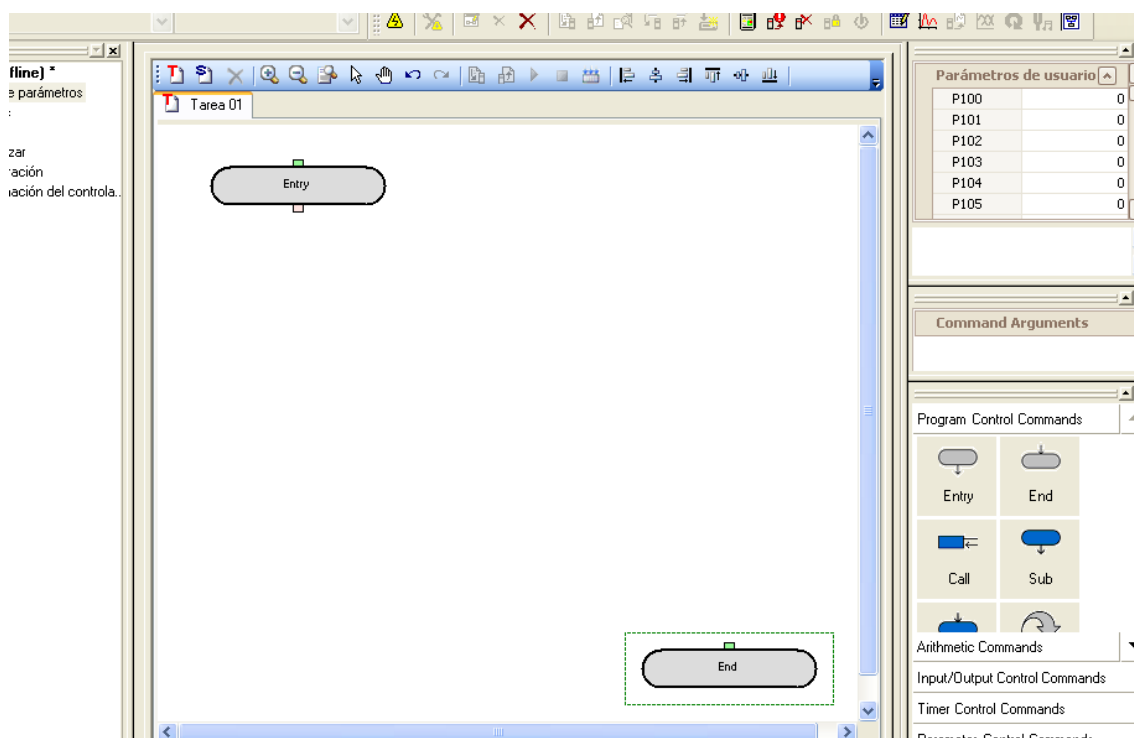
**Ventana de comandos:** Aquí encontraremos todos los comandos necesarios para poder realizar el proceso tales como, poner en marcha el variador, detenerlo, esperar durante un tiempo (temporizador), etc...

Después de esta breve explicación sobre las diferentes ventanas vamos a realizar la práctica que hemos expuesto anteriormente.

He de decir que el lenguaje de programación de este software puede parecer algo complicado pero después de realizar unas pruebas y entender su estructura se hace más entretenido el desarrollo del programa. Es un lenguaje secuencial aunque se pueden crear varias tareas para que se ejecuten al mismo tiempo.

Lo primero que haremos será introducir el comando “Entry” y “End” en nuestro programa y esto lo deberemos de hacer siempre ya que ha de haber un inicio y un final de programa. Estos comandos se encuentran en “Program control commands” situado en la ventana de comandos.

Quedara la interface de este modo (Figura 1.9).



**Figura 1.9**

A continuación introduciremos el comando “Waitcond” (Esperar condición) para esperar la señal del sensor del inicio de la cinta transportadora. Una vez introducido dicho comando debemos de pulsar sobre él y fijarnos en la parte derecha de la pantalla en el apartado “Command arguments” (Figura 1.10).

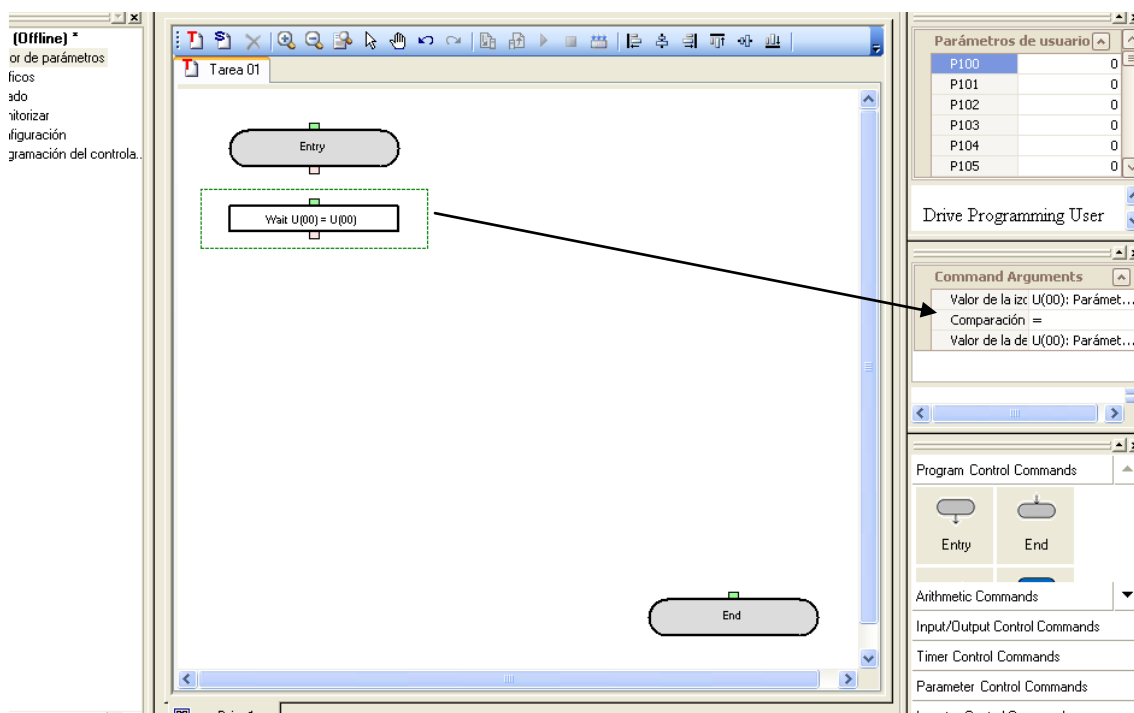


Figura 1.10

Donde pone **Valor de la izquierda** pulsaremos en **Parámetro** y se nos desplegará una lista donde tendremos que buscar un parámetro que dice **X(00): Contacto de entrada general**, es decir entrada física 0 del variador (en nuestro caso la que utilizaremos para el sensor inicial de la cinta transportadora).

En comparación lo dejamos tal cual.

Donde pone **Valor de la derecha** pulsaremos en **Parámetro** y colocaremos a través del teclado el valor 1.

¿Qué es lo que hemos hecho con estos pasos ?

Simplemente decirle que hasta que la entrada 0 no sea 1 no pase a la siguiente etapa.

Repetimos el proceso para la siguiente etapa, es decir, colocamos otro comando "Waitcond" pero ahora, al ser otra entrada la que queremos configurar (Pulsador de marcha) en parámetros hemos de buscar uno que se llama **X(01): Contacto de entrada general**, ya que estamos utilizando otra entrada física y por lo tanto no puede ser la misma.

De momento nuestro programa quedará así (Figura 1.11)

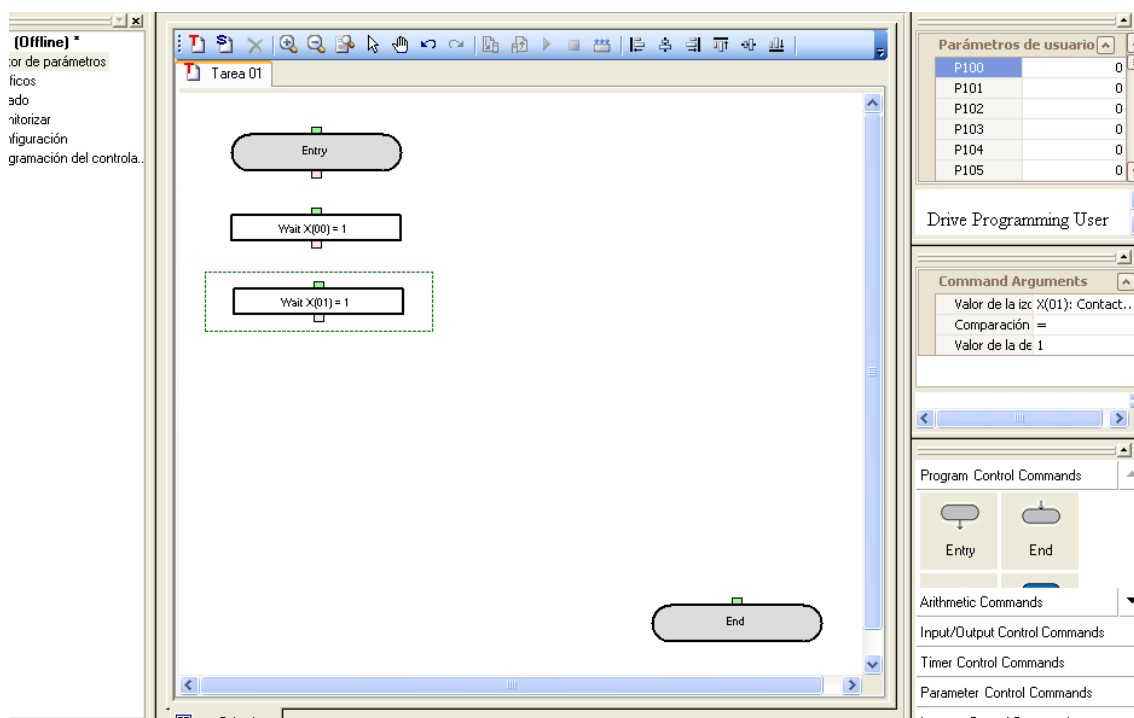


Figura 1.11

Si se cumple lo que de momento llevamos programado a continuación vendría el comando de poner en marcha la cinta. ¿ Como lo hacemos ? Pues muy fácil, nos fijamos en la ventana de comandos en un grupo que se llama "Inverter control commands" donde si lo desplegamos veremos una opción que se llama RunFw (Figura 1.12).

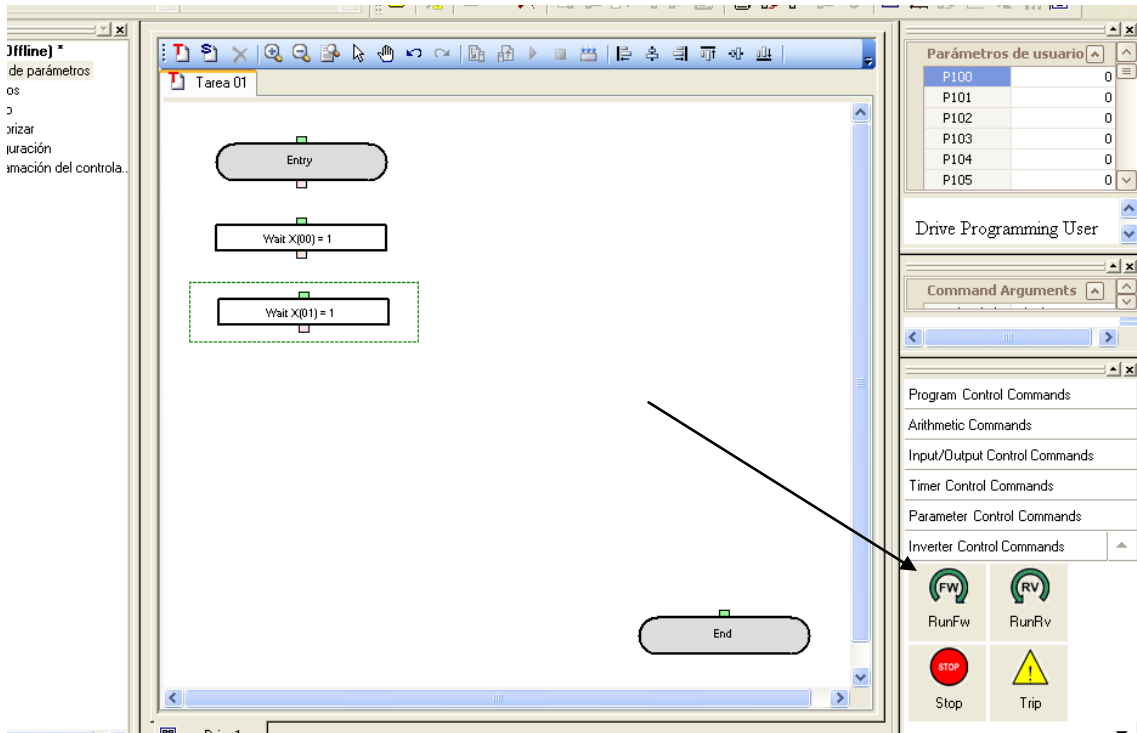


Figura 1.12

La añadimos a nuestro programa y nos quedará algo similar a la siguiente imagen (Figura 1.13)

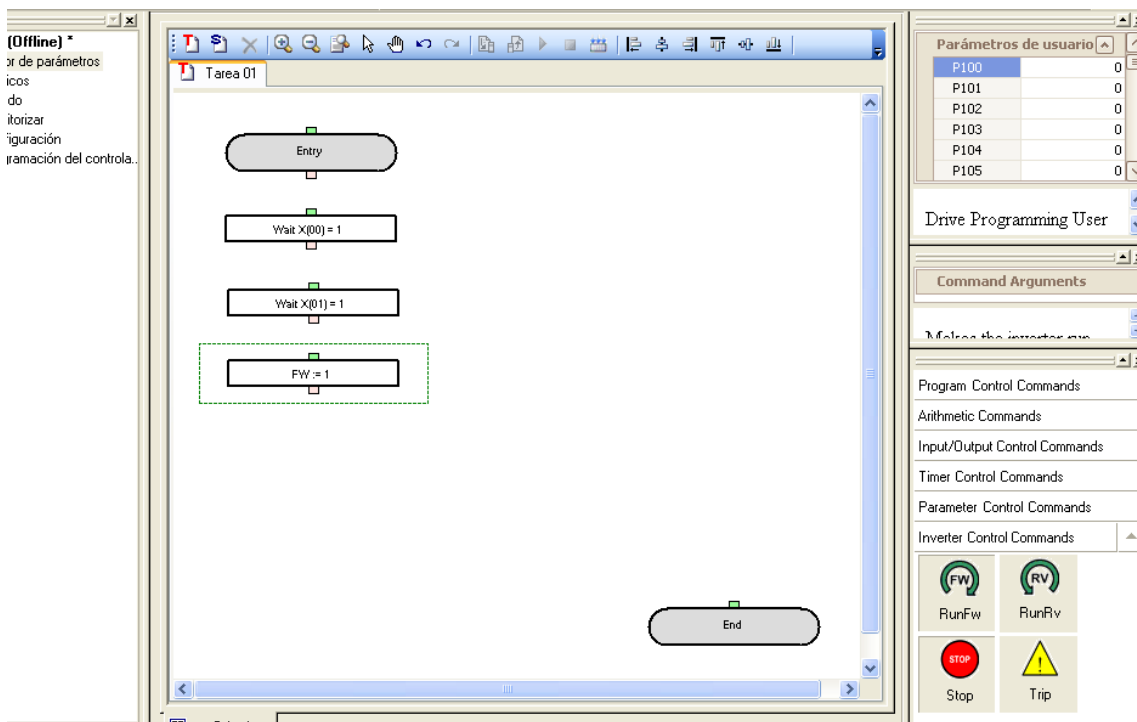
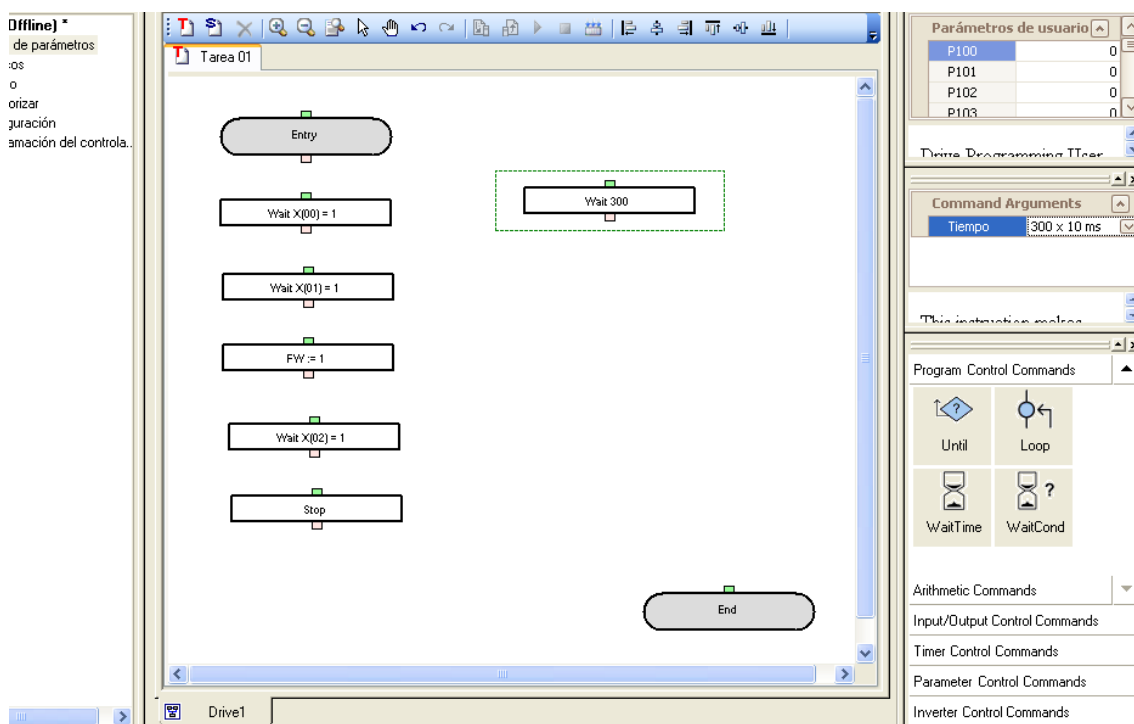


Figura 1.13

A continuación colocaremos otro comando “Waitcond” para el sensor de final de cinta transportadora colocando en parámetro de la izquierda, el llamado **X(02): Contacto de entrada general**.

En parámetro del valor de la derecha, al igual que los dos anteriores, pondremos un 1.

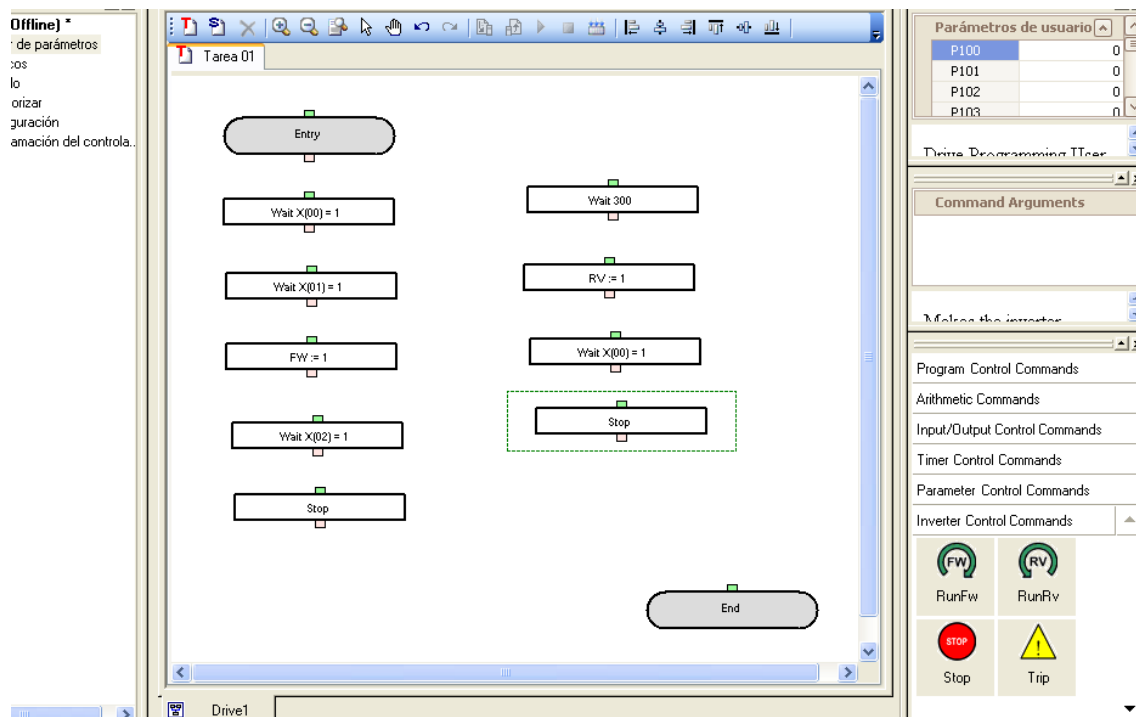
Al cumplirse esta condición se parara la cinta durante un tiempo (3 segundos) por lo que colocaremos un comando llamado “Waittime” después de indicarle a través de otro comando el Stop de la cinta. Nos quedará de la siguiente manera (Figura 1.14).



**Figura 1.14**

Como se puede observar al pulsar sobre el comando “Waittime” podemos ver como en la ventana de la derecha “Command arguments” podemos modificar el tiempo del temporizador.

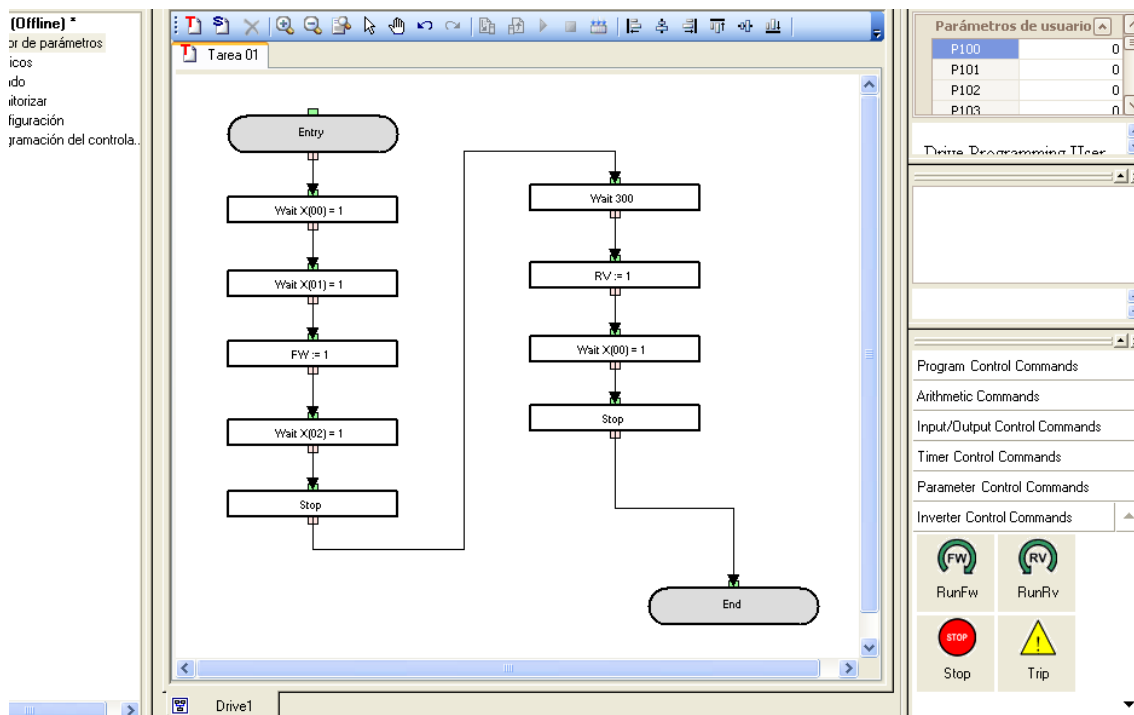
A continuación, transcurridos los 3 segundos se pondrá de nuevo en marcha la cinta, pero ahora de forma inversa y no se detendrá hasta detectar el sensor de inicio de cinta por lo que el programa quedaría de la siguiente manera (Figura 1.15)



(Figura 1.15)

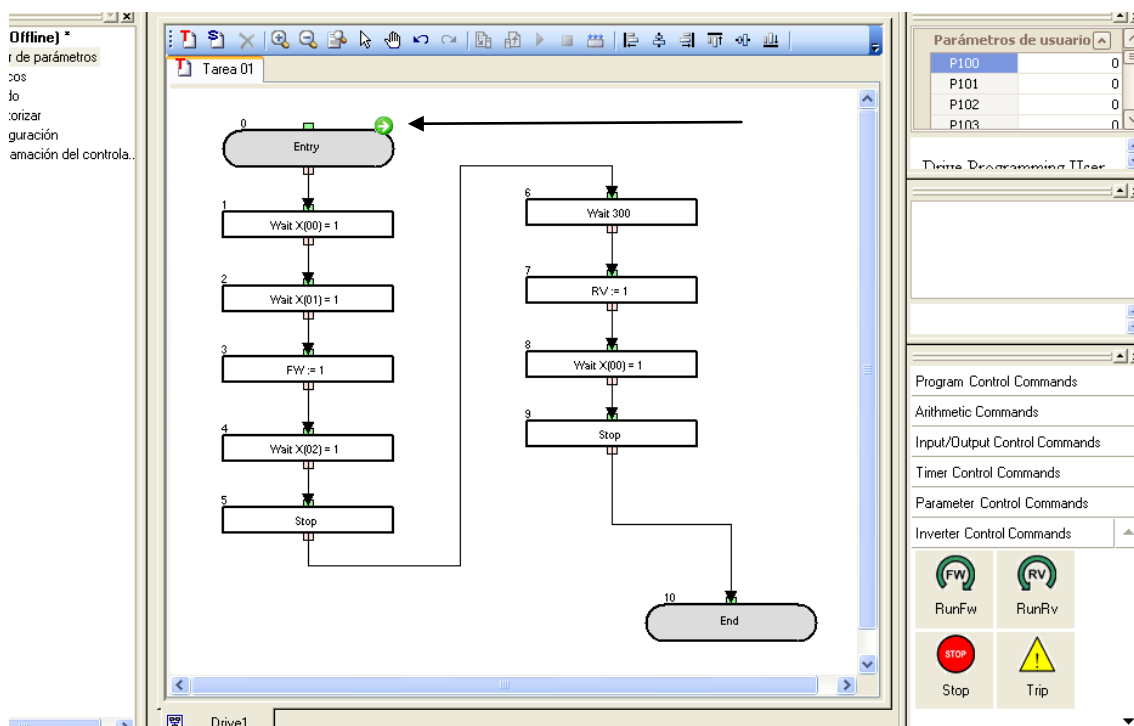
Por último nos quedaría unir toda las etapas para que al compilar el programa no nos de ningún error, para ello uniremos la salida del comando entry y la uniremos con la entrada de la entrada 0 y así sucesivamente hasta tener todo el programa unido (Figura 1.16)





**Figura 1.16**

Una vez hecho esto, compilaremos el programa para ver si tenemos algún error de programación. Si esta todo OK deberíamos de visualizar una flecha de color verde junto al comando entry. (Figura 1.17).



**Figura 1.17**

Nos quedaría por último, enviarle al variador el programa que hemos desarrollado pero antes de enviarlo, daremos un repaso de los parámetros del variador que se han de ajustar para que al realizar esta práctica no tengamos ningún problema.

## 5. Configuración parámetros del variador

Para el desarrollo y buen funcionamiento de esta práctica tan solo hemos de configurar cinco parámetros del variador, tales como:

A001 (Fuente de frecuencia): Donde entre todas las opciones, elegiremos la número 1 (Terminal de control).

A002 (Fuente de comando Run): Elegiremos también la opción número 1 (Terminal de control).


C001 (Función de entrada 1): En este parámetro elegiremos la opción 56 (MI1, entrada de propósito general 1). Esta es la entrada del sensor inicial de la cinta transportadora.

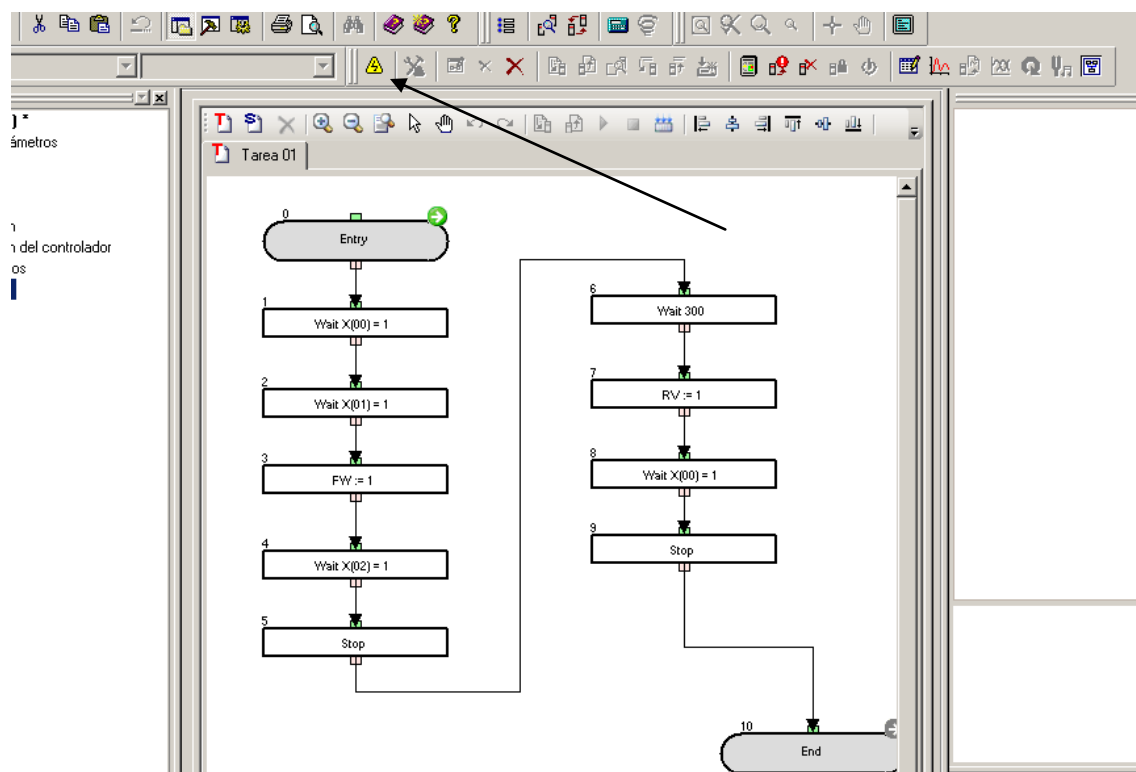
C002 (Función de entrada 2): En este parámetro elegiremos la opción 57 (MI2, entrada de propósito general 2). Esta es la entrada del pulsador de marcha.

C003 (Función de entrada 3): En este parámetro elegiremos la opción 58 (MI3, entrada de propósito general 3). Esta es la entrada del sensor final de la cinta transportadora.

Con estos pasos ya tenemos configurados los parámetros del variador por lo que pasaremos al último punto de este pequeño manual, el envío del programa al variador.

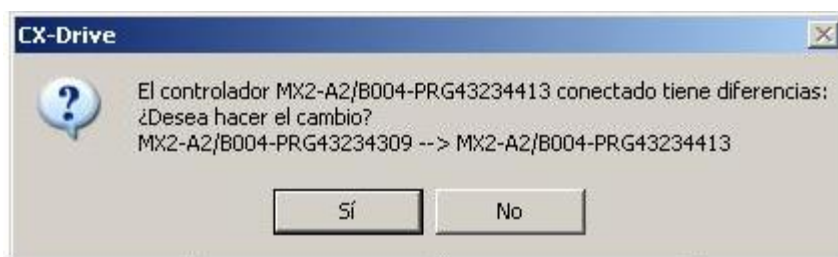
## 6. Envío del programa

El envío del programa es bien sencillo, deberemos de pulsar sobre el icono  que esta situado en el menú herramientas para la comunicación con el variador. (Figura 1.18).



**Figura 1.18**

Una vez pulsado, si el variador conectado tiene diferencias con el configurado se nos mostrará un mensaje de advertencia donde nos preguntará si deseamos realizar el cambio. (Figura 1.19)



**(Figura 1.19)**

Al aceptar los cambios, veremos cómo los botones que antes teníamos deshabilitados para el envío del programa, ahora los tenemos totalmente habilitados. (Figura 1.20).

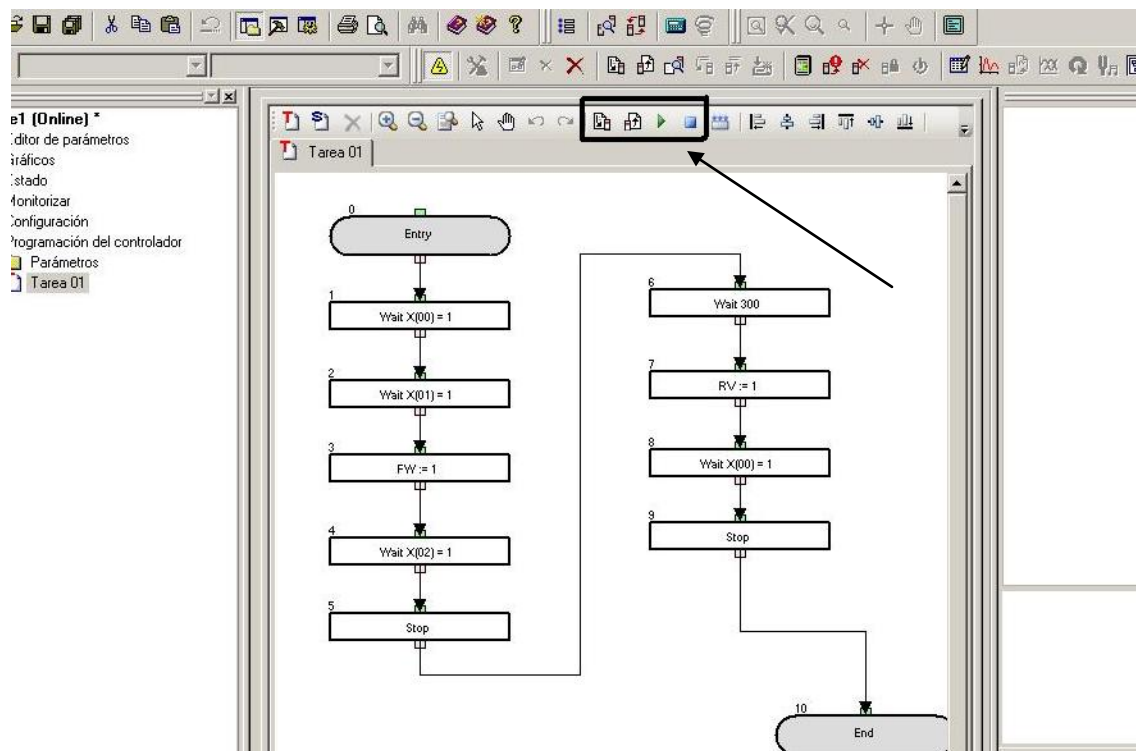

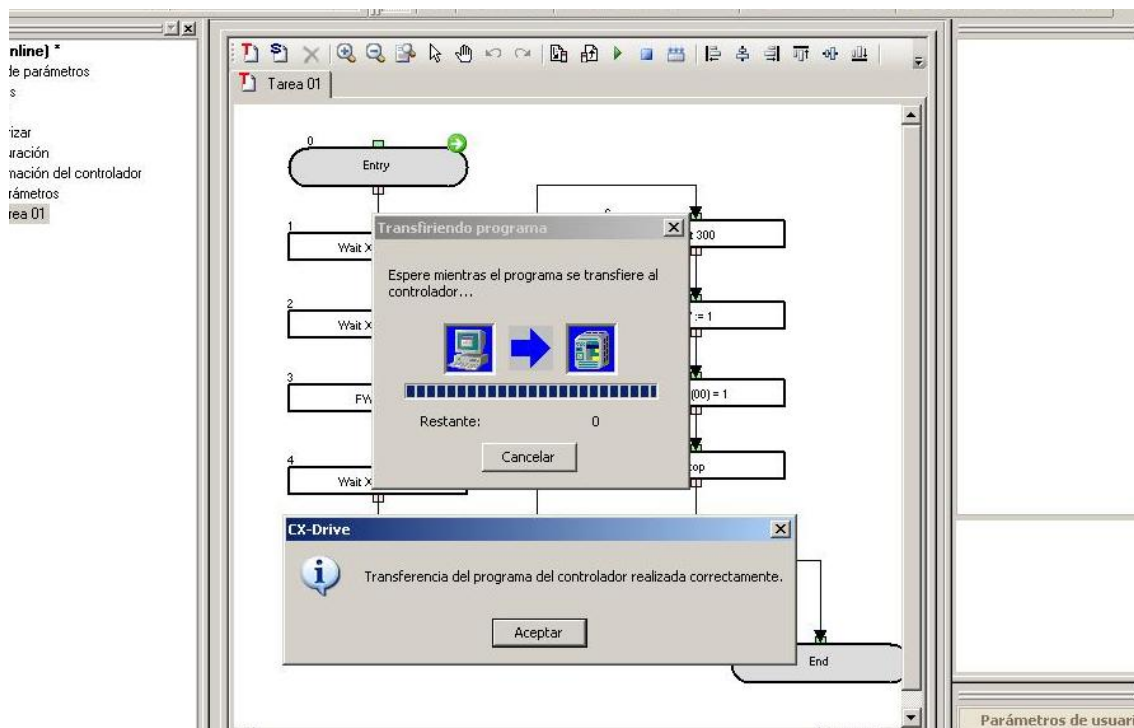



Figura 1.20



A continuación pulsaremos sobre el icono  transferir programa al controlador, donde se nos mostrará un mensaje indicándonos que el programa se está enviando y una vez finalizado nos saldrá otro mensaje diciéndonos que el programa ha sido enviado sin problemas. (Figura 1.21).



**Figura 1.21**

Por último, solo nos quedará simularlo, por lo que pulsaremos sobre el siguiente icono , que vendría a ser el “RUN” de cualquier simulador.

Si se ha seguido al pie de la letra este mini manual, la práctica nos debería de funcionar a la primera, por lo que pulsaremos el sensor de inicio de la cinta transportadora, luego el pulsador de marcha (donde se activará el motor de la cinta), a continuación activaremos el sensor de final de la cinta (el motor de la cinta se desactivará durante tres segundos), y posteriormente se activará de nuevo de forma inversa hasta pulsar el detector de inicio de la cinta donde el motor se desactivará de nuevo.

Este sería el ciclo de funcionamiento por lo que si queremos que funcione de nuevo deberíamos de pasar a modo “STOP”  y ponerlo de nuevo en modo “Run” , ya que al tener la instrucción END en el programa, una vez realizado un ciclo de funcionamiento, finalizará el proceso. Esto se puede evitar introduciendo un nuevo comando en la interface de programación, pero lo dejaremos para que lo investiguen lo más curiosos y sean ellos mismos quienes modifiquen el programa.

Por mi parte esto es todo, espero que este manual sea entendido por la mayoría de la gente que lo lea y si alguien tiene algún problema o se queda atascado en cualquier punto expuesto en este manual, estaremos por aquí para resolver cualquier duda.

