

**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARIA
SEDE CONCEPCIÓN - REY BALDUINO DE BELGICA**

GUÍA PARA LA DOCENCIA DE SERVOMOTORES SEW

Trabajo de Titulación para optar al Título
de Técnico Universitario en Robótica y
Mecatrónica.

Alumnos:

Jairo Nicodemo Crisóstomo Monsalve

Nicolás Francisco Millavil Oñate

Profesor Guía:

Msc. Felipe Benavides Pantoja

Concepción, Chile

2018

Resumen

SERVOMOTOR SEW

“GUÍA PARA LA DOCENCIA Y BASE PARA ESTUDIANTES ”.

POR

Jairo Nicodemo Crisóstomo Monsalve.

Nicolás Francisco Millavil Oñate.

Profesor Guía: Felipe Benavides Pantoja.

Concepción, Chile.

2018

En la actualidad las industrias manufactureras están incorporando nuevas tecnologías como lo son los servomotores.

Con el objetivo de inculcar mayor conocimiento sobre este, se proporcionará una amplia información sobre los servomotes SEW, tales como su principio de funcionamiento, su estructura y ventajas de este. Por otra parte, es necesario conocer el software con el cual funciona el servomotor, la plataforma Movidrive, donde es analizado las principales características del programa, normas de seguridades entre otros.

Para concretar con las nociones básicas, diferentes laboratorios de trabajo forjaran las habilidades necesarias para el buen uso del servomotor formando futuros proyectos con este.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO I	2
1.1. ¿QUÉ ES UN SERVOMOTOR?	3
1.2. PRINCIPIO DE FUNCIONAMIENTO	3
1.3. CARACTERÍSTICA DEL SERVOMOTOR	5
1.3.1. Alta velocidad:	5
1.3.2. Baja Inercia:	6
1.3.3. Minimizar Jload/Jmotor:	8
1.3.4. Alta densidad de torque:	9
1.3.5. Requerimientos de espacio:	10
1.4. ESTRUCTURA Y PARTES DE UN SERVOMOTOR:	11
1.5. RETROALIMENTACIÓN	11
1.5.1. Encoder incremental	12
1.5.2. Encoder absoluto	12
1.6. VENTAJAS Y DESVENTAJAS DEL SERVOMOTOR	12
CAPÍTULO II	13
2.1. ¿QUÉ ES MOVIDRIVE?	14
2.2. PRINCIPIO DE CONTROL DE POSICIÓN	15
2.3. CARACTERÍSTICAS MOVIDRIVE:	15
2.4. FUNCIONES	16
2.5. CARACTERÍSTICAS DE TARJETA DE MEMORIA	16
2.5.1. Reemplazo tarjeta de memoria.	17
2.6. SEGURIDAD DE MOVIDRIVE	17
2.6.1. Movidrive B safety stop	18
2.7. EFICIENCIA ENERGÉTICA	18
2.8. EXISTEN DOS TIPOS DE MOVI-PLC	18
2.8.1. MOVI-PLC basic DHP11B	18
2.8.2. MOVI-PLC advanced	19
2.9. INTERFACES DE ENCODER	20
2.9.1. Interface de encoder DEH11B:	20
2.9.2. Interface de encoder DEH21B:	20
2.9.3. Interface de encoder DER11B:	21
2.10. INTERFASE BUS DE CAMPO	21

2.11. DIAGNOSTICO – MEMORIAS DE FALLAS	23
Conclusión	27
Bibliografía	28
Anexos	29
LABORATORIO N°1: “CONFIGURACIÓN SPEED CONTROLLER”	30
1.1 CONEXIÓN SERVOMOTOR-PC	31
1.2 ESCANEADO DE COMUNICACIÓN SERVOMOTOR	32
1.3 CONFIGURACIÓN PRINCIPAL DEL SERVOMOTOR	35
LABORATORIO N°2: “ANÁLISIS VARIABLES SPEED CONTROLLER”	45
2.1 CONFIGURACIÓN	46
2.2 ANÁLISIS VARIABLES MOTOR PARÁMETROS	47
LABORATORIO N°3: “CONTADOR CON IPOS COMPILER”	52
3.1 CONFIGURACIÓN	53
3.2 CONSTRUCCIONES EN C	59
If e if else	59
For.....	59
While.....	60
Do while.....	60
Switch case.....	60
LABORATORIO N°4: “CONTROL DE MOTOR UTILIZANDO GOABS, GOREL Y GO0”	62
4.1 COMPARADORES LÓGICOS	63
4.2 FUNCIONES DE COMANDO DE POSICIÓN	63
4.3 DECLARACIÓN DE VARIABLES	66
LABORATORIO N°5: “PROBLEMA REAL”	68
5.1 PROBLEMÁTICA EMPRESARIAL	69
5.2 ANÁLISIS DEL PROBLEMA	69
5.3 SOLUCIÓN A LA PROBLEMÁTICA	70
5.4 CREACIÓN DEL PROGRAMA	71
LABORATORIO N°6: “ENTRADAS Y SALIDAS DIGITALES”	72
6.1 DECLARACIÓN DE ENTRADAS Y SALIDAS DIGITALES	73
6.2 COMANDOS	73
6.3 VISUALIZACIÓN DE VARIABLES	74
6.4 EJERCICIO	75
LABORATORIO N°7: “LECTURA DEL PUERTO ANÁLOGO ”	77

7.1	CONEXIONADO DEL POTENCIÓMETRO AL SERVOMOTOR.	78
7.2	COMANDOS	78
7.3	PROGRAMACIÓN	80
	LABORATORIO N°8: “COMUNICACIÓN ETHERNET”	82
8.1	PROTOCOLO CONFIGURACIÓN IP	83

ÍNDICE DE FIGURAS

CAPITULO I

Ecuación 1. 1	5
Ecuación 1. 2	6
Ecuación 1. 3	7
Ecuación 1. 4	7
Ecuación 1. 5	9
Gráfica 1.1	9
Imagen 1. 1 Material conductor por el cual pasa una corriente formando un campo magnético.	3
Imagen 1. 2 Bobina la cual se le intrduce una corriente sinusoidal generando un campo magnetico en la dirección.....	4
Imagen 1. 3 representacion de un iman permanente por el cual pasa un campo magnetico produciendo torque.....	5
Imagen 1. 4 representacion de un iman permanente por el cual pasa un campo magnetico produciendo torque.....	5
Imagen 1. 5	8
Imagen 1. 6 Comparación de tamaño entre un servomotor y motor de inducción	10
Imagen 1. 7 Partes internas del servomotor.	11
Tabla 1. 1	6
Tabla 1. 2	8
Tabla 1. 3	10

CAPITULO II

Imagen 2. 1 Lazo de control de posicionamiento por Movidrive.	15
Imagen 2. 2 Tarjeta de memoria y ubicación en el controlador.....	17
Imagen 2. 3 MOVI-PLC basic DHP11B.....	19
Imagen 2. 4 MOVI-PLC advanced DHE/DHR/DHF41B.....	20
Imagen 2. 5 Interface de encoder DEH11B.	20
Imagen 2. 6 Interface de encoder DEH21B.	21
Imagen 2. 7 Interface de encoder DER11B.	21

Imagen 2. 8 Interface de bus de campo para servomotores SEW.....	22
Imagen 2. 9 Interface de bus de campo Ethernet para servomotores SEW.	23
Tabla 2. 1 Significado de siglas de seguridad.	18

ANEXOS

Imagen 1, 1 Conexión ethernet Pc-Servomotor.	31
Imagen 1, 2 Configuración de propiedades ethernet.....	31
Imagen 1, 3 Configuración de IP del PC	32
Imagen 1, 4 Inicio plataforma MOVITOOLS.....	33
Imagen 1, 5 Configuración de la comunicación en MOVITOOLS	33
Imagen 1, 6 Selección del tipo de comunicación.....	34
Imagen 1, 7 Visualización de la comunicación.....	34
Imagen 1, 8 Escaneo del controlador.	34
Imagen 1, 9 Reconocimiento del servomotor.	35
Imagen 1, 10 Selección de “Startup”.	35
Imagen 1, 11 Configuración inicial para el control del servomotor.	36
Imagen 1, 12 Configuración inicial para el control del servomotor.	36
Imagen 1, 13 Configuración inicial para el control del servomotor.	37
Imagen 1, 14 Configuración inicial para el control del servomotor.	37
Imagen 1, 15 Configuración inicial para el control del servomotor.	38
Imagen 1, 16 Configuración inicial para el control del servomotor.	38
Imagen 1, 17 Configuración inicial para el control del servomotor “Cualquier parámetro modificado inapropiadamente en este ítem podría dañar severamente el servomotor”	39
Imagen 1, 18 Configuración inicial para el control del servomotor.	39
Imagen 1, 19 Configuración inicial para el control del servomotor.	40
Imagen 1, 20 Configuración inicial para el control del servomotor.	40
Imagen 1, 21 Configuración inicial para el control del servomotor.	41
Imagen 1, 22 Configuración inicial para el control del servomotor.	41
Imagen 1, 23 Configuración inicial para el control del servomotor. "No Apretar Download".	42
Imagen 1, 24 Configuración inicial para el control del servomotor.	42
Imagen 1, 25 Para los parámetros del servomotor ingresar en “Parameter tree”	43
Imagen 1, 26 Configuración de parámetros de velocidad del servomotor.....	43
Imagen 1, 27 Switch de la maqueta del servomotor.	44

Imagen 2, 1 Diferentes modos de operación.	46
Imagen 2, 2 Parámetros para configuración del servomotor.....	46
Imagen 2, 3 Configuración de velocidad del servomotor.	47
Imagen 2, 4 Parámetros de velocidad.	48
Imagen 2, 5 configuración para el servomotor.....	49
Imagen 2, 6 Parámetros de protección del servomotor.	50
Imagen 2, 7 configuración de dirección del servomotor.....	51
Imagen 3, 1 Configuración por IPOS.....	53
Imagen 3, 2 Opciones de comunicación por IPOS.	53
Imagen 3, 3 Opciones de compilación en IPOS.....	54
Imagen 3, 4 Crear nueva planilla.	54
Imagen 3, 5 Creación de planilla de programa..	55
Imagen 3, 6 Selección de parámetros para la creación de programas.....	55
Imagen 3, 7 constante preestablecidas	56
Imagen 3, 8 Estructura de una sola ejecución de programa.	56
Imagen 3, 9 estructura del programa que se ejecuta periódicamente a través de una condición.....	56
Imagen 3, 10 Opción para visualizar variables.	57
Imagen 3, 11 Visualización de variables.....	57
Imagen 3, 12 Compilación del programa.....	57
Imagen 3, 13 Correcta compilación del programa.	58
Imagen 3, 14 Errónea compilación del programa.	58
Imagen 3, 15 Compilación y carga del programa.	59
Imagen 3, 16 Ejecución del programa.....	59
Imagen 3, 17 Ejemplo de programación.	61
Imagen 4, 1 Comandos de Go0.....	64
Imagen 4, 2 Comandos GoAbs.	65
Imagen 4, 3 Comandos GoRel.	65
Imagen 4, 4 Código de ejemplo.	67
Imagen 5, 1 Código del ejercicio.	71
Imagen 6, 1 Comando BitClear.....	74
Imagen 6, 2 Visualización de las variables.	75

Imagen 6, 3 Solución ejercicio.....	76
Imagen 7, 1 - Imagen 7, 2 Diagramas de conexión.....	78
Imagen 7, 3 Declaración de variables.	80
Imagen 7, 4 Código del programa.....	81
Imagen 7, 5 Código de lectura y conversión del programa.....	81
Imagen 8, 1 Switch “DEF IP” on y off.	83
Imagen 8, 2 direccionándose hacia la carpeta de movitools.	83
Imagen 8, 3 abriendo ubicación de Addr Tool.....	84
Imagen 8, 4 Programa Addr Tool.	84
Imagen 8, 5 Configuración IP a través de Addr Tool.	85
Imagen 8, 6 Configuración IP a través de Addr Tool.	85
Imagen 8, 7 Configuración IP a través de Addr Tool	86
Imagen 8, 8 Configuración IP a través de Addr Tool	86
Imagen 8, 9 Verificación IP	86
Imagen 8, 10 Diagrama de comunicación.....	87

INTRODUCCIÓN

Para la comunidad universitaria, especialmente para los alumnos y profesores, es fundamental la motivación e iniciativa en nuevos proyectos, forjando nuevas habilidades versátiles.

El trabajo que realizaremos está enfocado principalmente como una ayuda tanto a docentes como alumnos de la universidad, ya que será una guía enfocada en servomotores de la marca SEW, los cuales posee la sede concepción y son utilizados por profesores y estudiantes del área encargada. La guía constara con una base introductoria al servomotor con su respectivo software y el completo funcionamiento de este, además de un anexo con una lista de 8 laboratorios orientados al aprendizaje de una mejor forma para quienes estudien sobre este.

Objetivo General:

Facilitar el entendimiento y estimular la creatividad, con la creación de una guía de servomotores SEW apoyado de 8 laboratorios, así brindándole al alumno o docente un material académico extra con el cual puede tener apoyo en todo momento que lo necesite.

Objetivos específicos:

- Estudiar y analizar la estructura del servomotor SEW.
- Estudiar y analizar el driver movitools correspondiente al servomotor.
- Trabajar diferentes tipos de laboratorios.
- Sintetizar información para el óptimo aprendizaje de los alumnos.
- Estudiar las falencias del servomotor.

CAPÍTULO I

1.1. ¿QUÉ ES UN SERVOMOTOR?

Un servomotor es un tipo de motor eléctrico que tiene la capacidad de controlar el movimiento de su eje, en velocidad, aceleración, torque y posición. Su bajo momento de inercia y su elevada capacidad de sobrecarga le permite hacer movimientos muy rápidos, con grandes aceleraciones y frenadas. Este puede ser bifásico o trifásico, ya que los motores de corriente continua no son capaces de dar determinada cantidad de vueltas o detenerse en una posición fija. Estos giran indefinidamente, hasta que se interrumpa el suministro de corriente. Lo cual hace que no sea posible utilizarlos para asuntos de robótica, y en este tipo de aplicaciones se necesita movimientos precisos y mantener posiciones fijas.

En cambio, con los servomotores podemos crear toda clase movimientos de una forma controlada, por ejemplo, en robótica para el control del movimiento del brazo de un robot o en los sistemas de radio control.

1.2. PRINCIPIO DE FUNCIONAMIENTO

La mayor parte de motores eléctricos operan a través de la interacción entre un campo magnético y corriente sinusoidal para generar fuerza dentro del propio motor.

El motor de corriente alterna y los de corriente continua se basan en el mismo principio de funcionamiento, el cual establece que si un conductor, por el que circula una corriente eléctrica, se encuentra dentro de la acción de un campo magnético, éste tiende a desplazarse perpendicularmente a las líneas de acción del campo magnético. (Ver Imagen 1.1)

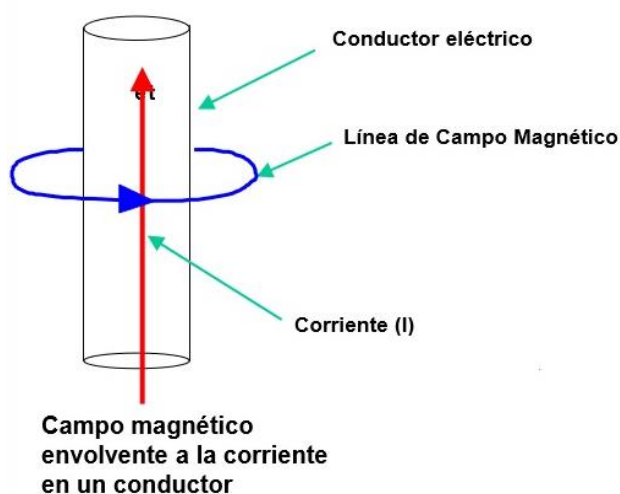


Imagen 1. 1 Material conductor por el cual pasa una corriente formando un campo magnético.

En los motores, la electricidad crea campos magnéticos opuestos entre sí, que provocan que la parte giratoria de éste (el rotor) se mueva.

A grandes rasgos, podemos decir que donde exista electricidad habrá magnetismo, y que la polaridad de un objeto se puede alterar haciendo circular electricidad en una u otra dirección a su alrededor, mientras más alta sea la corriente creará un campo magnético más fuerte (Vector más grande), y viceversa mientras más pequeña es la corriente el campo magnético será más débil (Vector más pequeño). (Ver Imagen 1.29)

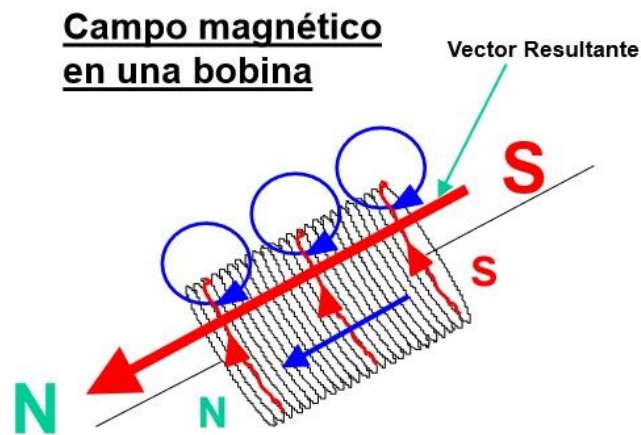


Imagen 1. 2 Bobina la cual se le intrduce una corriente sinusoidal generando un campo magnetico en la dirección

La importancia de la posición del rotor en un motor sincrónico:

Constituye la parte móvil del motor. El rotor es el elemento de transferencia mecánica, ya que de él depende la conversión de energía eléctrica a mecánica. Los rotores, son un conjunto de láminas de acero al silicio que forman un paquete.

El ángulo de ubicación del imán permanente con respecto al campo magnético del estator nos permite obtener la fuerza del torque:

Si el imán se encuentra en un ángulo de 90° con respecto al campo magnético del estator, el torque será máximo.

Si se encuentra entre 90° y 0° con respecto al campo magnético del estator, el torque será intermedio. (Ver Imagen 1.3)

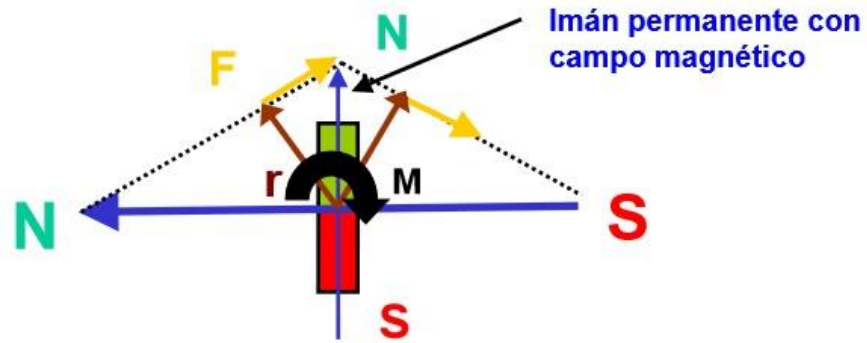


Imagen 1. 3 representacion de un iman permanente por el cual pasa un campo magnetico produciendo torque

Si el imán se encuentra a 0° con respecto al campo magnético del estator, el torque producido es 0, no genera fuerza. (Ver Imagen 1.4)

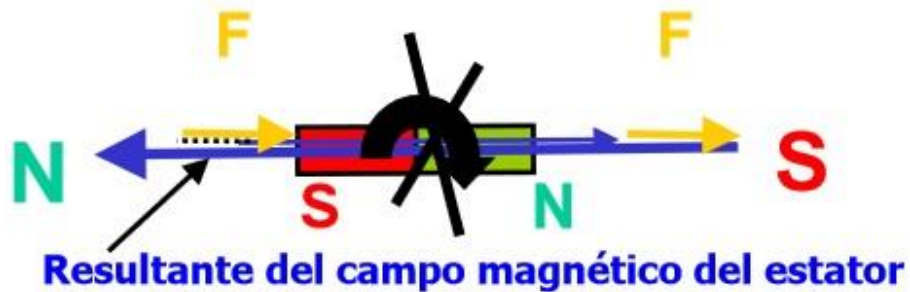


Imagen 1. 4 representacion de un iman permanente por el cual pasa un campo magnetico produciendo torque

1.3. CARACTERÍSTICA DEL SERVOMOTOR

1.3.1. Alta velocidad:

En un motor de inducción su velocidad nominal está dada por:

$$\text{M. Inducción} \quad \uparrow n = \frac{f^* \overset{\leftarrow}{2} \overset{\leftarrow}{60}}{p \downarrow}$$

Ecuación 1. 1

Donde “n” es la velocidad nominal “f” la frecuencia suministrada y “p” el número de polos. En un motor de inducción la frecuencia suministrada es fija por lo cual para aumentar su velocidad se requiere disminuir su cantidad de polos.

Al contrario de un servo motor, en el cual su cantidad de polos se mantiene fija y se le varia la frecuencia suministrada para aumentar o disminuir su velocidad según se requiera a través de un inversor, el cual se maneja a través de un driver que posee cada servomotor correspondientemente.

Servo

$$n = \frac{f * 2 * 60}{p}$$

Ecuación 1. 2

A continuación, se muestra una tabla donde compara lo anteriormente planteado

Motor Type	Speed
8 polos motor de inducción	750 rpm
6 polos motor de inducción	1000 rpm
4 polos motor de inducción	1500 rpm
2 polos motor de inducción	3000 rpm
6 polos servo	3000 rpm
6 polos servo	4500 rpm
6 polos servo	6000 rpm

Tabla 1. 1

1.3.2. Baja Inercia:

La inercia, en física, es la propiedad que posee los cuerpos de oponerse a un cambio de su estado de reposo o movimiento en que se encuentran. Como tal, la inercia es la resistencia ofrecida por un cuerpo a la alteración de su estado en reposo o movimiento.

Los servomotores tienen una inercia más baja que los motores de inducción para el mismo torque nominal.

La siguiente formula puede determinar la inercia de un objeto.

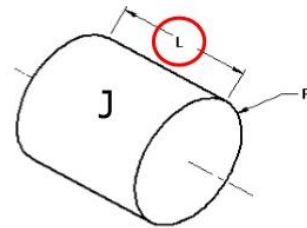
$$J = m * r^2$$

Ecuación 1. 3

Donde “J” es el símbolo para denominar la inercia en (Kgm²), “m” es la masa del objeto en (Kg), y “r” La distancia de la distribución de la masa con respecto a su centro de masa (m), o sea de determina la inercia por la masa y la forma de un objeto.

Para utilizar la formula en un servomotor la formula varía en función al cuerpo del objeto que en este caso será un cilindro por lo que quedaría de esta forma.

$$J = 1/2 m * r^2$$



Ecuación 1. 4

El radio de un rotor afecta drásticamente su inercia. Pero la longitud de un rotor cilíndrico no tiene efecto (a no ser que se incremente la masa por incremento de la longitud).

La tabla a continuación realiza comparaciones con diferentes tipos de motores y su factor de inercia.

Motor Type	Inertia J_{rotor} ($\times 10^{-4}$ Kgm ²)
6 pole induction	6,6
4 pole induction	4,4
2 pole induction	4,6
6 pole servo 3000 rpm	0,1
6 pole servo 4500 rpm	0,1
6 pole servo 6000 rpm	0,1

Tabla 1. 2

1.3.3. Minimizar Jload/Jmotor:

A diferencia de un motor de inducción el rotor de un servomotor es más largo, de menor radio y con agujeros a lo largo de este, esto se debe a que en la inercia solo influye la masa y el radio, como se demuestra en la ecuación anteriormente mencionada, por ello para mantener el torque en un rotor de servomotor este se alarga, disminuye el radio y se le hacen agujeros a lo largo de este para bajar su masa, en la siguiente imagen se compara un rotor de servomotor con uno de un motor de inducción con mismo torque (Ver Imagen 1.5)

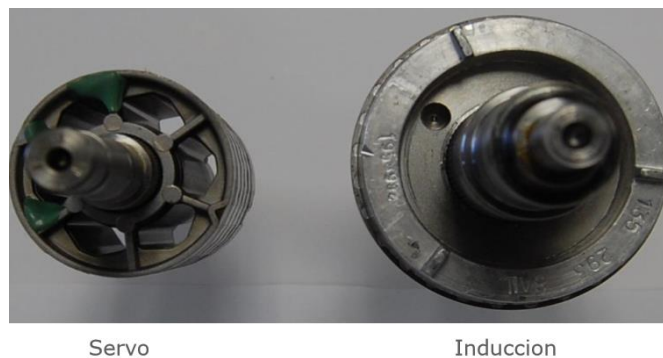
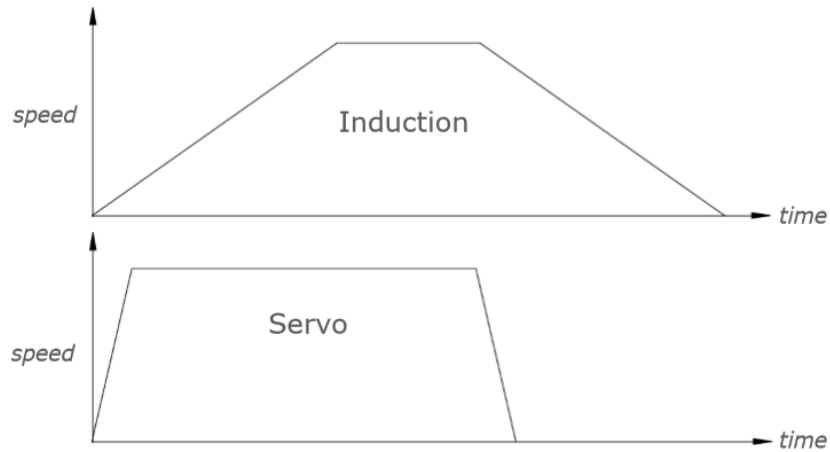


Imagen 1. 5

Además de bajar su inercia, su rampa de aceleración y de frenado son más cortas, como muestra a continuación:



Gráfica 1.1

1.3.4. Alta densidad de torque:

Para calcular la densidad de torque se utiliza la siguiente formula:

$$\rho_{torque} = \frac{T}{V}$$

Ecuación 1.5

Donde V es el volumen del eje del servomotor, T es el torque y ρ_{torque} es la densidad del torque resultante, esto explica como un servomotor puede tener la misma densidad de torque que un motor de inducción, aún con un volumen menor.

A continuación, se puede apreciar la comparación entre la cantidad de polos de inducción, el volumen del eje del servomotor.

Motor Type	Volume (cm ³)
6 pole induction	1696
4 pole induction	1696
2 pole induction	1696
6 pole servo 3000 rpm	188
6 pole servo 4500 rpm	188
6 pole servo 6000 rpm	188

Tabla 1. 3

1.3.5. Requerimientos de espacio:

La ventaja de un servomotor en comparación a un motor de inducción con la misma densidad de torque, es el espacio que abarca en el sistema, evitando así un problema de “ubicación especial”.

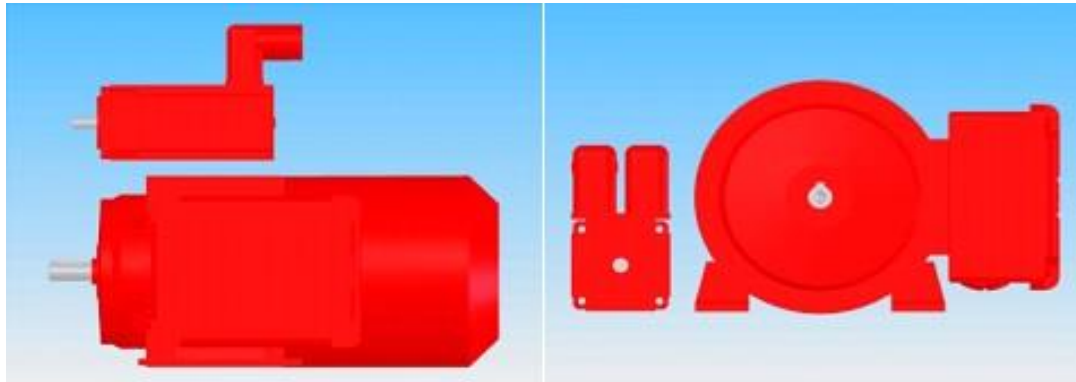


Imagen 1. 6 Comparación de tamaño entre un servomotor y motor de inducción

1.4. ESTRUCTURA Y PARTES DE UN SERVOMOTOR:

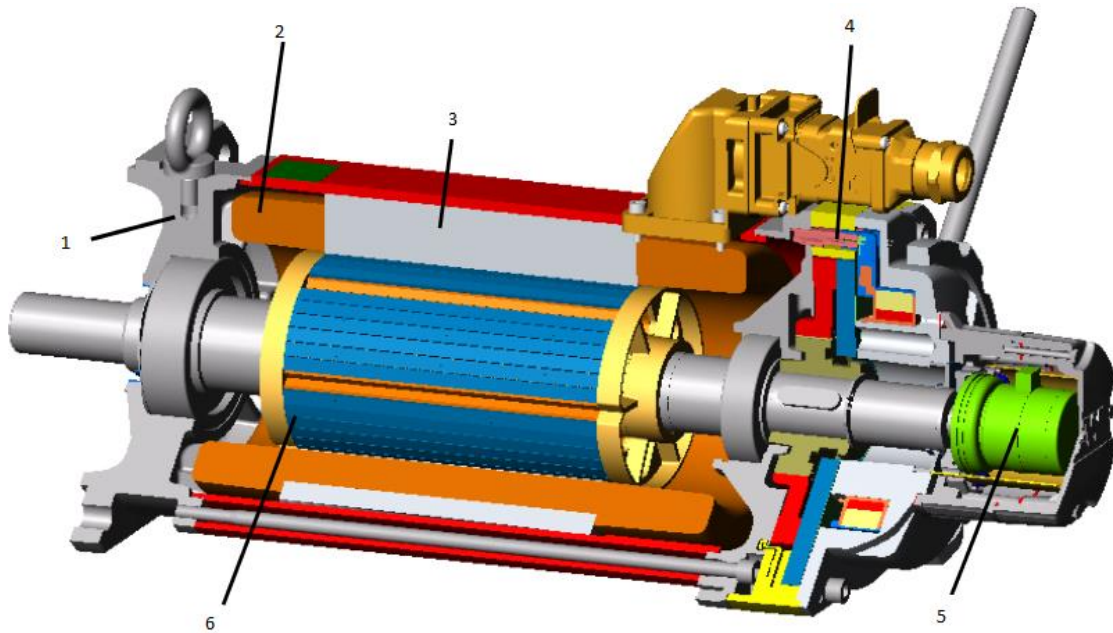


Imagen 1. 7 Partes internas del servomotor.

1. Estator.
2. Bobinado del estator.
3. Núcleo laminado del estator.
4. Freno.
5. Resolver o encoder de alta resolución.
6. Rotor con imanes permanentes.

1.5. RETROALIMENTACIÓN

Todo servomotor posee “dos cables principales”, uno que es para energizar (Bifásico o trifásico) y otro para la retroalimentación, estos van separados para que la señal del encoder que va por el cable de retroalimentación no se perturbe con el ruido cuando se energiza un servomotor.

Con respecto a la retroalimentación existen 2 formas de realizarlo, a través de un encoder el cual es acoplado de forma externa en un motor o por un resolver, el cual va incorporado en el servomotor.

En el caso del encoder puede ser incremental o absoluto, en el incremental existe el TTL/HTL y el Sin/Cos. En el absoluto puede ser codificado-óptico o Hiperface.

1.5.1. Encoder incremental

Los encoders incrementales generan un número exactamente definido de impulsos por revolución. Éstos indican la medida de la distancia angular y lineal recorrida. Debido al desplazamiento de fase entre las señales A y B (de aproximadamente 90 grados), se puede analizar el sentido de rotación.

1.5.2. Encoder absoluto

Los encoders absolutos asignan a cada posición angular un valor inequívoco, incluso durante varias revoluciones. Incluso después de una caída de tensión el encoder detecta de forma rápida y segura la posición momentánea.

En el caso del servomotor, posee este último, con un disco principal codificado en gray de 12 bits (4096 pasos por vuelta) indicando posición angular, además existen tres discos adicionales de 4 bits vinculados mediante reducciones $i=16$ puede definirse un rango de resolución absoluto de: $16 \times 16 \times 16 = 4096$ vueltas.

1.6. VENTAJAS Y DESVENTAJAS DEL SERVOMOTOR

Ventajas:

- Precisión en el posicionamiento.
- Precisión en control de la velocidad.
- Control en todo el rango de velocidad.
- Alto torque estático (velocidad = 0).
- Estabilidad en el torque.
- Capacidad de sobrecarga (hasta 4 veces).
- Performance dinámica (rápida aceleración y desaceleración).
- Compacto y Liviano.

Desventajas:

- Su costo es elevado.
- Se requiere personal capacitado para el mantenimiento del servomotor, ya que la calibración de este es muy sensible.

CAPÍTULO II

2.1. **¿QUÉ ES MOVIDRIVE?**

Movidrive es un controlador universal de servomotores que se puede utilizar tanto en:

- Motores asincrónicos sin encoder: Los cuales son utilizados en transportadores, ventiladores, bombas y cortadores.
- Motores asincrónicos con encoder: Los cuales son utilizados en sistemas de almacenamiento, control de velocidad preciso y posicionamiento.
- Motores sincrónicos: Los cuales son utilizados en manipulaciones dinámicas, sincronización de posición y máquinas de procesamiento.

MOVIDRIVE se puede utilizar en los siguientes sectores:

- Industria del automóvil
- Industria de alimentos y bebidas
- Transporte y logística

Uso de MOVIDRIVE en las siguientes aplicaciones:

- Sistemas de almacenamiento
- Alimentaciones
- Elevadores
- Aplicaciones sincronizadas
- Pórticos

2.2. PRINCIPIO DE CONTROL DE POSICIÓN

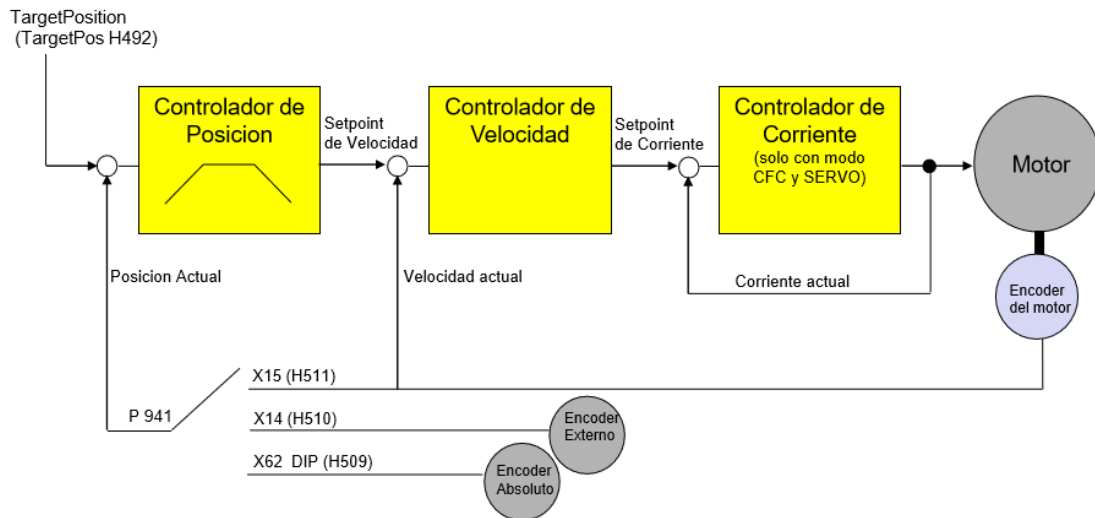


Imagen 2. 1 Lazo de control de posicionamiento por Movidrive.

El principio de control de posición a través de Movidrive es por un lazo de control cerrado en cascada, el cual en su entrada (TargetPosition) se le indica la posición deseada, esta se compara con la posición actual del servomotor en el controlador de posición; para el control de velocidad el cual viaja el servomotor se indica la velocidad deseada la cual se desea llegar, y es comparada con la velocidad actual del servomotor, este proceso es el controlador de velocidad; el mismo proceso es planteado para la corriente pero solo con los modos CFC y SERVO, a través de su respectivo controlador, de esta forma permitiendo al motor realizar el trabajo correspondiente, y corrigiendo sus errores a través de un encoder.

2.3. CARACTERÍSTICAS MOVIDRIVE:

- Controlador universal: Posibilidad de manejo de motores sincrónicos y asincrónicos.
- Plug & Play: Adicione las tarjetas y la unidad automáticamente las reconoce.
- Programación de movimientos y secuencias a través de IPOS Plus® incorporado.
- Función “Safety stop” integrada.
- Tarjeta de memoria integrada.
- Módulo de aplicaciones (posicionamiento, corte al vuelo, etc.).
- Programación rápida – Datos transferidos directamente desde el encoder Hiperface al equipo
- 8 Entradas digitales / 6 Salidas digitales.
- 1 Salida a relé / 1 Entrada analógica.

- Evaluación TF/TH.
- Borneras removibles.
- Control automático de freno.
- Circuito Chopper de frenado integrado.
- Control PID Integrado.
- Gran Capacidad de Sobrecarga:
 - 125 % IN en forma continua.
 - 150 % IN por 60 segundos.
 - 180 % IN por 5 segundos.
- Filtro de Entrada EMC Incorporado (EN 55011)
- Memoria de fallos (5 fallos).
- 6 Velocidades Fijas Programador extraíble.
- Puesta en Marcha desde panel o desde una PC (con Motion Studio)
- Interfaces de Comunicación: RS-485, SBus y CanOpen como estándar
Profibus, DeviceNet, Ethernet/IP, etc.

2.4. FUNCIONES

Las funciones integradas permiten una variedad de aplicaciones. Las siguientes funciones están integradas en el Movidrive B:

- Entradas y salidas digitales configurables libremente.
- Función de Elevador (Hoist).
- Función de arranque en vuelo (Flying start).
- Función de memorización de la consigna (Setpoint hold).
- Segundo juego de parámetros Función de Osciloscopio digital (SCOPE) vía PC.
- Versión de Aplicación: Leva Electrónica y Sincronismo Interno (ISYNC).
- Módulos de Aplicación: Sin programación solo configuración.
- Funciones de Protección Integradas:
 - Sobre-Corriente
 - Sobre-Temperatura
 - Falta de Fase
 - Monitoreo de Velocidad

2.5. CARACTERÍSTICAS DE TARJETA DE MEMORIA

- Tarjeta de memoria insertable
- Como estándar para cada unidad
- Conjunto de datos completo: Parámetros, IPOS, variables, curvas
- Reemplazo de la unidad sin startup
- Tiempos de inactividad (Down time) reducidos

2.5.1. Reemplazo tarjeta de memoria.

- 1 Remueva la unidad vieja
- 2 Instale la nueva unidad
- 3 Inserte la tarjeta de memoria
- 4 Arranque el motor y/o la maquina



Imagen 2. 2 Tarjeta de memoria y ubicación en el controlador.

2.6. SEGURIDAD DE MOVIDRIVE

- STO de acuerdo a PL-d en la unidad básica.
- Monitor se seguridad insertable en la unidad de acuerdo a PL-e.
- STO, SS1, SS2, SLS, SDI, SLP, SBC SLI, SLC y otros de acuerdo a PL-e (ver tabla 2.1)
- Lógica de operación Failsafe.
- Interfaces de comunicación seguras para ProfiBus Profisafe y Profinet Profisafe.

- Puede ser opcionalmente expandida por I/O para automatización completa segura de una celda.

Safe Stop and brake functions		Safe Motion Basics Functions	Safe Motion Advance Functions
STO	Safe Torque Off	SLS	Safety Limited Speed
SS1	Safe Stop 1	SS1	Safety Stop 1. maintaining power and after certain time delay OFF
SOS	Safe Operating Stop	SS2	(Stop function) like Stop Cat.2 of EN60204, Controlled shut down while maintaining the supply of power, and after Standstill Can includes SOS
SS2	Safe Stop 2	SLP	Safely-Limited position
SBC	Safe Brake Control	...	SafeEncoder. Combine to encoders
		SLA	Safety Limited Acceleration
		SAR	Safe Acceleration range
		SSR	Safe Speed Range
		SLT	Safely-Limited Torque
		STR	Safe Torque Range
		SLI	Safely-Limited Increment
		SDI	Safe Direction
		SCA	Safe Cam
		SSM	Safe Speed Monitor
		SMT	Safe Motor Temperature

Tabla 2. 1 Significado de siglas de seguridad.

2.6.1. Movidrive B safety stop

- Protección segura contra re arranques en conformidad a norma EN954-1, cat. 3, categoría de parada 0 o 1.
- Elimina la conexión de potencia a través de un contactor.
- PLC de seguridad o switch de seguridad externo como un dispositivo de accionamiento de la función.

2.7. EFICIENCIA ENERGÉTICA

- Óptimo “startup” para motores AC serie DR de eficiencia energética.
- Costos de Procesos reducidos ajustando los movimientos a los requerimientos.
- Funciones de eficiencia energética para reducción de la magnetización orientada al proceso.
- Unidades de alimentación regenerativas para estados de operación regenerativos (37 kW, 75 kW, 132 kW).
- Intercambio de energía vía conexión DC link.

MOVIDRIVE® MDR60A alimenta el DC link del circuito del Movidrive B conectado con energía eléctrica de la red de alimentación en operación normal del motor y retorna la potencia regenerativa a la red de alimentación en operación regenerativa del motor.

2.8. EXISTEN DOS TIPOS DE MOVI-PLC

2.8.1. MOVI-PLC basic DHP11B

Características de hardware:

- 8 terminales de I/O.
- 2 puertos CAN.
- Puerto Esclavo ProfiBus.
- 1 Puerto RS-485.
- Puerto opcional OST11B (RS-485).

Características funcionales:

- Hasta 12 ejes.
- Máximo 64 I/O por acoplador de bus.
- Idealmente para aplicaciones estándares.
- Funciones Tecnológicas.



Imagen 2. 3 MOVI-PLC basic DHP11B.

2.8.2. MOVI-PLC advanced

- 8 terminales digitales de I/O, 5 con capacidad de interrupción
- 2 puertos CAN
- 2 puertos RS-485
- 1 puerto Ethernet
- 1 puerto EtherCat system bus
- Puerto USB
- Tarjeta SD (leible en PC)
- 512 kB memoria para programa
- 128 kB memoria para datos
- 16 kB variables retentivas
- 8 kB variables del sistema (retentivas)
- Programación conforme a IEC 61131-3, conforme a PLCopen



Imagen 2. 4 MOVIE-PLC advanced DHE/DHR/DHF41B

2.9. INTERFACES DE ENCODER

2.9.1. Interface de encoder DEH11B:

- Encoder Hiperface.
- Encoder Sin/Cos.
- RS422.
- Encoder TTL.
- Encoder HTL (con DWE12B).
- Simulación de encoder incremental o encoder externo.



Imagen 2. 5 Interface de encoder DEH11B.

2.9.2. Interface de encoder DEH21B:

- Encoder SSI.
- Encoder Hiperface.
- Encoder Sin/Cos.

- RS422.
- Encoder TTL.
- Encoder HTL (con DWE12B).



Imagen 2. 6 Interface de encoder DEH21B.

2.9.3. Interface de encoder DER11B:

- Resolver.
- Simulación de encoder incremental o encoder externo.



Imagen 2. 7 Interface de encoder DER11B.

2.10. INTERFASE BUS DE CAMPO

- CANopen (DFC11B)

- DeviceNet (DFD11B)
- Interbus (DFI11B)
- Interbus/FO (DFI21B)
- Profibus (DFP21B)
- Profisafe/Profibus + CAN (DFS12B)
- Profisafe/Profibus (DFS11B)

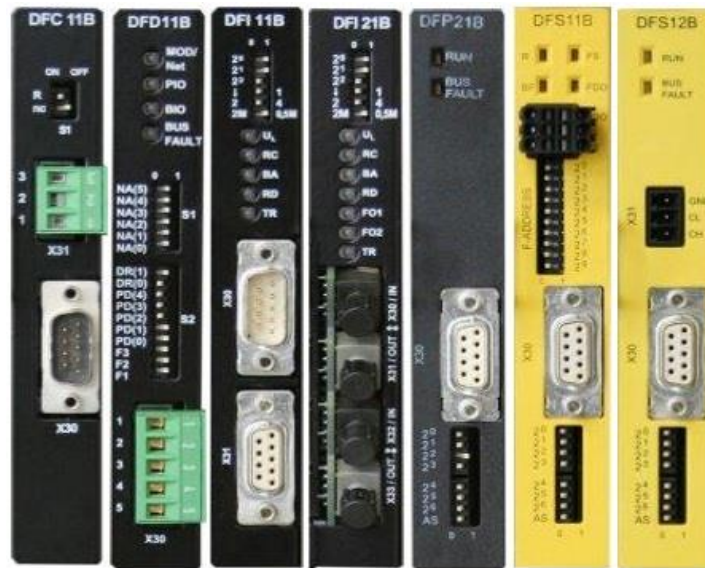


Imagen 2. 8 Interface de bus de campo para servomotores SEW.

Interfaces de Bus de campo Ethernet

- Modbus/TCP (DFE11B)
- PROFINET (DFE12B)
- Ethernet/IP (DFE13B)
- EtherCAT (DFE24B)
- PROFINET incl. switch (DFE32B)
- Ethernet/IP & Modbus/TCP incl. switch (DFE33B)
- Profisafe/PROFINET + CAN (DFS22B)
- Profisafe/PROFINET (DFS21B)



Imagen 2. 9 Interface de bus de campo Ethernet para servomotores SEW.

2.11. DIAGNOSTICO – MEMORIAS DE FALLAS

Display de 7 segmentos	Estado de la unidad (Byte alto en la palabra de estado 1)	Significado
0	0	Funcionamiento con 24V (controlador no está listo)
1	1	Bloqueo controlador activado
2	2	Sin habilitación
3	3	Corriente de parada
4	4	Habilitación
5	5	Regulación n
6	6	Regulación M
7	7	Regulador de posición
8	8	Ajuste de fábrica
9	9	Final de carrera alcanzado
A	10	Opción tecnológica
B	-	Libre
C	12	Búsqueda de referencia IPOSplus
D	13	Reconexión en marcha
E	14	Medición de encoder
F	11	Display de fallo (parpadeante)

H	-	Funcionamiento manual
T	16	Convertidor esperando datos
U	17	“parada segura” activa
. (Punto parpadeante)	-	El programa IPOSplus se está ejecutando
Display parpadeante	-	STOP desde DBG60B

Código de fallo	Denominación	Respuesta	Causa posible	Medida
00	Sin fallo	-		
01	Sobre corriente	Parada de emergencia	Cortocircuito en la salida. Motor demasiado grande. Etapa de salida defectuosa. Limitación de rampa desconectada y tiempo de rampa ajustado demasiado corto.	Eliminar el cortocircuito. Conectar un motor más pequeño. En caso de etapa de salida defectuosa consultar al servicio de SEW. Activar P 138 y/o alargar el tiempo de rampa.
03	Derivación a tierra en el cable de motor	Desconexión inmediata	Fallo a tierra. En la línea de alimentación del motor. En el convertidor. En el motor.	Eliminar el contacto a tierra. Consultar al servicio de SEW.
04	Freno chopper	Desconexión inmediata	Potencia regenerativa demasiado alta. Circuito de resistencia de frenado interrumpido. Cortocircuito en el circuito de resistencia de frenado.	Prolongar las rampas de deceleración. Comprobar las conexiones de la resistencia de frenado. Comprobar los datos técnicos de la

			Valor de resistencia de frenado demasiado alto. Freno chopper defectuoso.	resistencia de frenado. En el caso de freno chopper defectuoso, cambiar el MOVIDRIVE.
06	Fallo de fase de la red	Desconexión inmediata	Fallo de fase.	Comprobar la línea de alimentación de red.
07	Sobretensión Uz	Desconexión inmediata	Tensión del circuito intermedio demasiado alta.	Prolongar las rampas de deceleración. Comprobar la línea de alimentación a la resistencia de frenado. Comprobar los datos técnicos de la resistencia de frenado.
08	Vigilancia de velocidad	Desconexión inmediata	El regulador de velocidad o el regulador de corriente (en el modo de funcionamiento VFC sin encoder) trabaja a límite de corriente ajustado debido a la sobrecarga mecánica o fallo de fase en la red o en el motor. El encoder no está correctamente conectado o el sentido de giro es incorrecto. En la regulación del par se sobrepasa $N_{m\acute{a}x}$.	Reducir la carga. Aumentar el tiempo de retardo ajustado (P501 o P503). Comprobar la conexión del encoder, si fuera necesario cambiar los pares A/A y B/B. Comprobar la alimentación de tensión del encoder. Comprobar la limitación de corriente. Si fuera necesario, prolongar las rampas.

			<p>En modo de funcionamiento VFC: Frecuencia de salida ≥ 150 Hz</p> <p>En modo de funcionamiento U/f: Frecuencia de salida ≥ 600Hz.</p>	<p>Comprobar el motor y la línea de alimentación del motor.</p> <p>Comprobar las fases de alimentación.</p>
09	Puesta en marcha	Desconexión inmediata	El convertidor no se ha puesto en marcha aún para el modo de funcionamiento seleccionado.	Llevar a cabo la puesta en marcha para el modo de funcionamiento correspondiente.
10	IPOS-ILLOP	Parada de emergencia	Se ha detectado un comando erróneo en la ejecución del programa IPOSplus. Condiciones erróneas en la ejecución del comando.	<p>Comprobar el contenido de la memoria del programa y, si fuera necesario, corregirlo.</p> <p>Cargar el programa correcto en la memoria del programa.</p> <p>Probar el desarrollo del programa (Manual IPOSplus).</p>

Conclusión

Se han ido solucionado diferentes tipos de problemas mediante se ha progresado en la guía, en los que firmemente se puede asegurar una relación directa con los objetivos de esta, otorgando así una ayuda a quienes les está dirigida.

Los servomotores significan un avance tecnológico para las industrias manufactureras, no debe olvidarse la gran utilidad que éste puede presentar en grandes proyectos, organizando y uniendo diversas aplicaciones en diferentes formas de trabajo.

Con esta guía de trabajo se retroalimenta el conocimiento adquirido en clases sobre el servomotor, profundizándose en tareas y laboratorios además de la información necesaria para utilizar el servomotor junto con su respectivo software.

Bibliografía

L. PUCCIARIELLO. Entrenamiento de Electrónica Comandos del compilador SEW EURODRIVE [Diapositiva]. Argentina, TASC ARGENTINA. 2002. 103 diapositivas.

SEW EURODRIVE. Introducción a la Tecnología de Servomotores [Diapositivas]. 2004. 131 diapositivas.

SEW EURODRIVE. IPOS Positioning and Sequence Control System [DataSheet]. 11/2009, Bruchsal, Germany. 2009. 384 paginas.

SEW EURODRIVE. System Manual MOVIDRIVE [DataSheet]. Bruchsal, Germany. 2009. 101 paginas.

7GRAUS. Significados. [En línea]. 2013. <<https://www.significados.com/inercia/>>.

TECNOLOGIA. areatecnologia. [En línea]. 2010. <<http://www.areatecnologia.com/electricidad/servomotor.html>>.

Anexos

LABORATORIO N°1: “CONFIGURACIÓN SPEED CONTROLLER”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Realizar comunicación servomotor-pc a través de Ethernet.</i> • <i>Conocer la plataforma “MOVITOOLS”.</i> • <i>Realizar configuración principal para el servomotor.</i> • <i>Realizar configuración speed controller (Variar velocidades).</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>Para empezar a trabajar con el servomotor, se requiere un cable Ethernet y realizar la configuración de la IP del PC, al abrir el software movitools se podrá crear un nuevo proyecto y comenzar con la configuración principal de este y de speed controller.</i></p>		

1.1 CONEXIÓN SERVOMOTOR-PC

El servomotor necesita trabajar con la plataforma MOVITOOLS, por lo cual requiere de una conexión Ethernet y así comunicarse con el PC. (ver imagen 1.1)



Imagen 1, 1 Conexión ethernet Pc-Servomotor.

Para realizar esta comunicación se requerirá modificar la IP del computador, si es necesario. Lo primero a realizar para cambiar la IP es entrar a **panel de control**, ir a **redes e internet** y luego **conexiones de red**. Realizar clic secundario en “Ethernet” e ir a propiedades. (ver imagen 1.2)

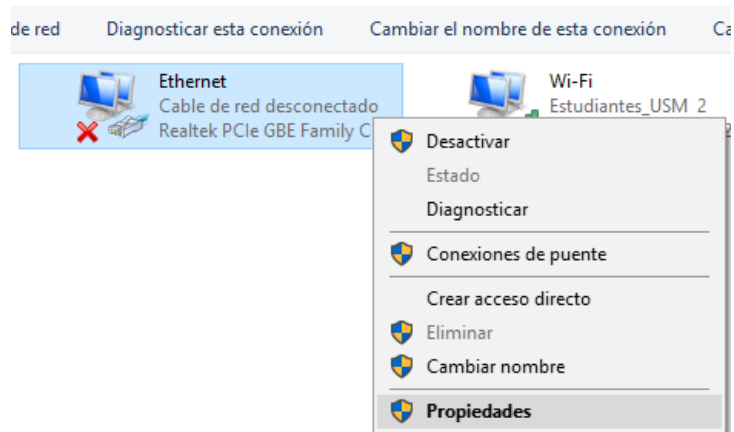


Imagen 1, 2 Configuración de propiedades ethernet.

En propiedades seleccionar la opción “Protocolo de internet versión 4 (TCP/IPv4)” Donde se deberá cambiar la IP del computador por la siguiente “192.168.10.4” donde el último número es a elección dependiendo del PC en donde se esté trabajando. (ver imagen 1.3)

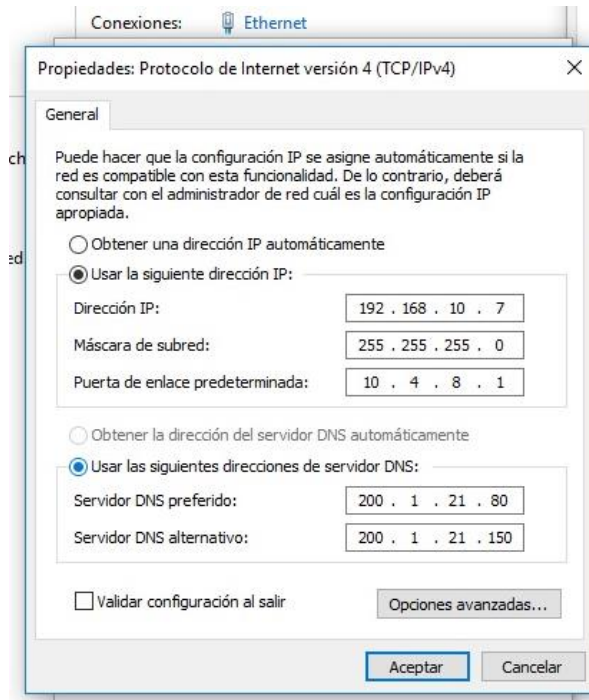


Imagen 1, 3 Configuración de IP del PC

1.2 ESCANEO DE COMUNICACIÓN SERVOMOTOR

Ya con la configuración IP lista, ejecutamos nuestro software MOVITOOLS en donde podremos empezar a trabajar con nuestro servomotor, el primer paso es crear un nuevo proyecto (ver imagen 1.4)

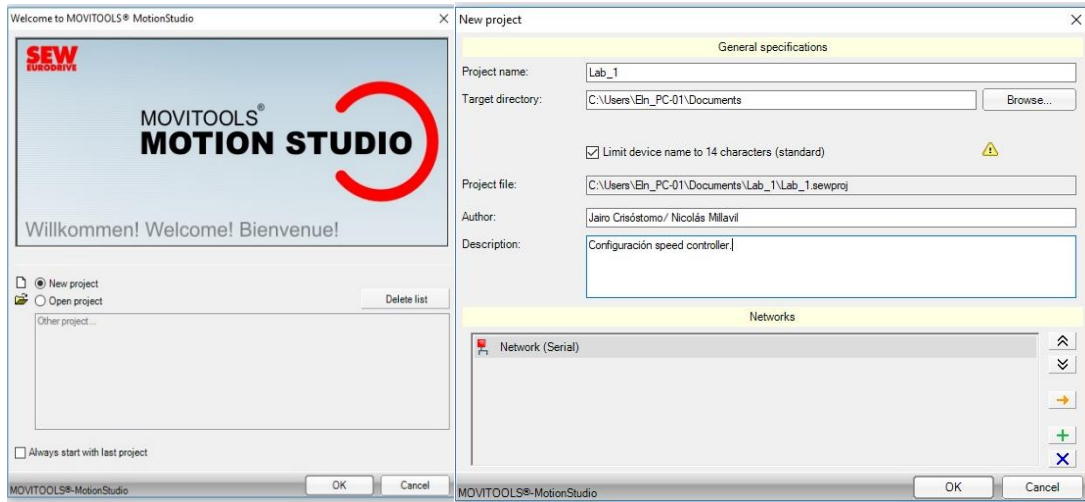


Imagen 1, 4 Inicio plataforma MOVITOOLS

Una vez creado el proyecto, se realizará la comunicación Ethernet por la configuración de MOVITOOLS, clic secundario en “Serial (COM)” (ver imagen 1.5)

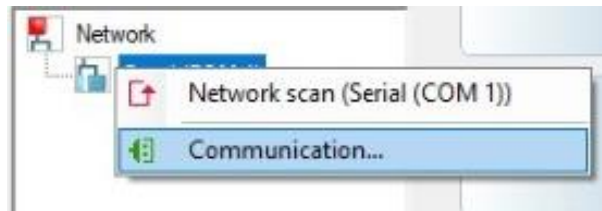


Imagen 1, 5 Configuración de la comunicación en MOVITOOLS

Se debera cambiar el puerto en el que se encuentra y seleccionar la opcion “Ethernet”. (ver imagen 1.6)

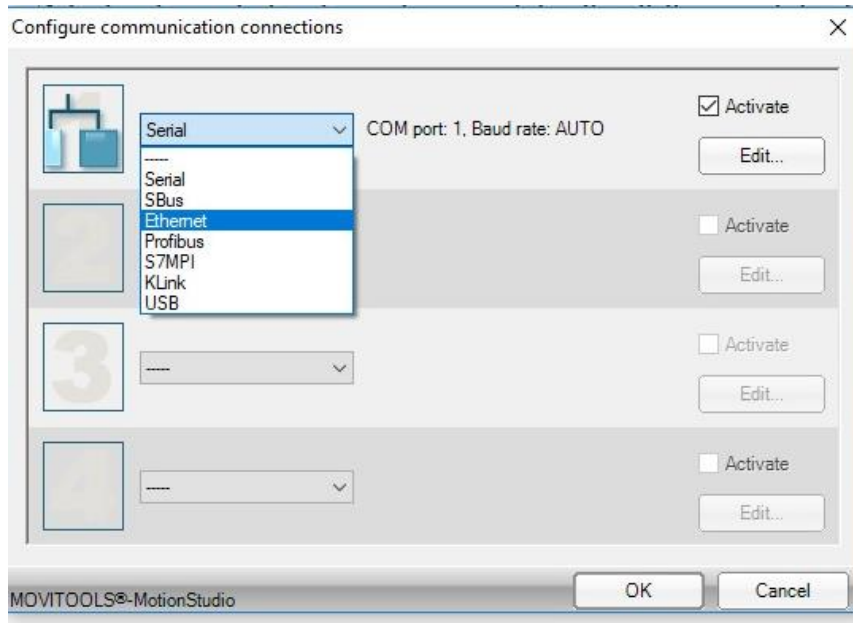


Imagen 1, 6 Selección del tipo de comunicación.

Se podrá visualizar en la primera interfaz que se cambio satisfactoriamente el puerto. (ver imagen 1.7)



Imagen 1, 7 Visualización de la comunicación.

Por ultimo se procedera a escanear el puerto para tener la comunicación completa. (ver imagen 1.8)

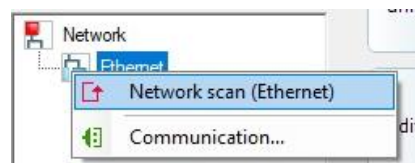


Imagen 1, 8 Escaneo del controlador.

El escaneo del puerto debió haber reconocido la IP anteriormente modificada y así quedaría comunicado el servomotor. (ver imagen 1.9)

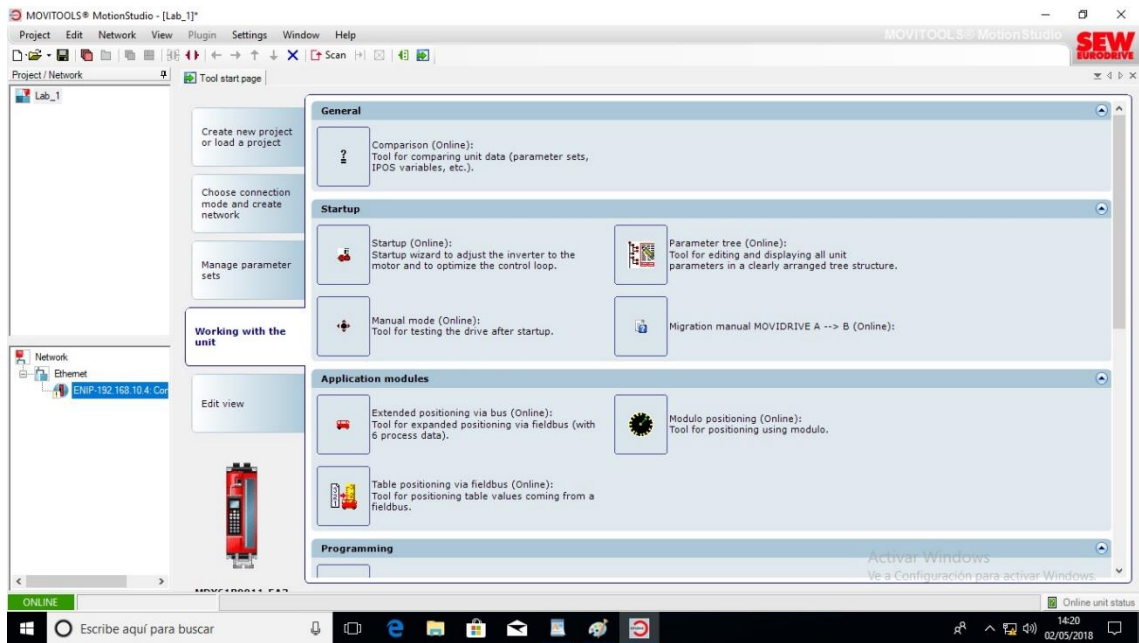


Imagen 1, 9 Reconocimiento del servomotor.

1.3 CONFIGURACIÓN PRINCIPAL DEL SERVOMOTOR

Seleccionar la opción “Startup” para proceder a la configuración principal del servomotor y trabajar con “speed controller”. (ver imagen 1.10)

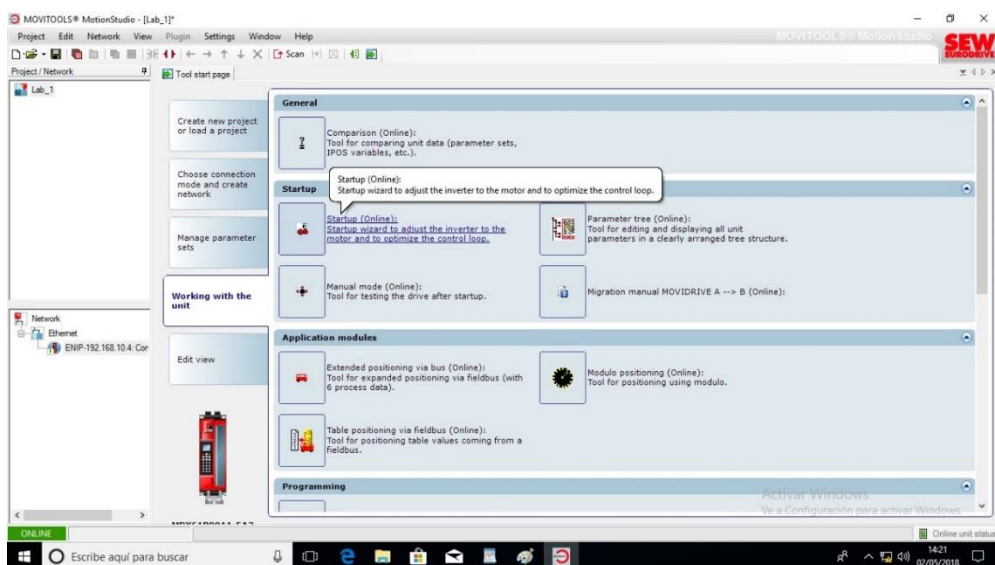


Imagen 1, 10 Selección de “Startup”.

Seleccionar “perform startup” y dar siguiente a todos los parámetros de las imágenes a continuación (ver imágenes de la 1.11 a la 1.24)

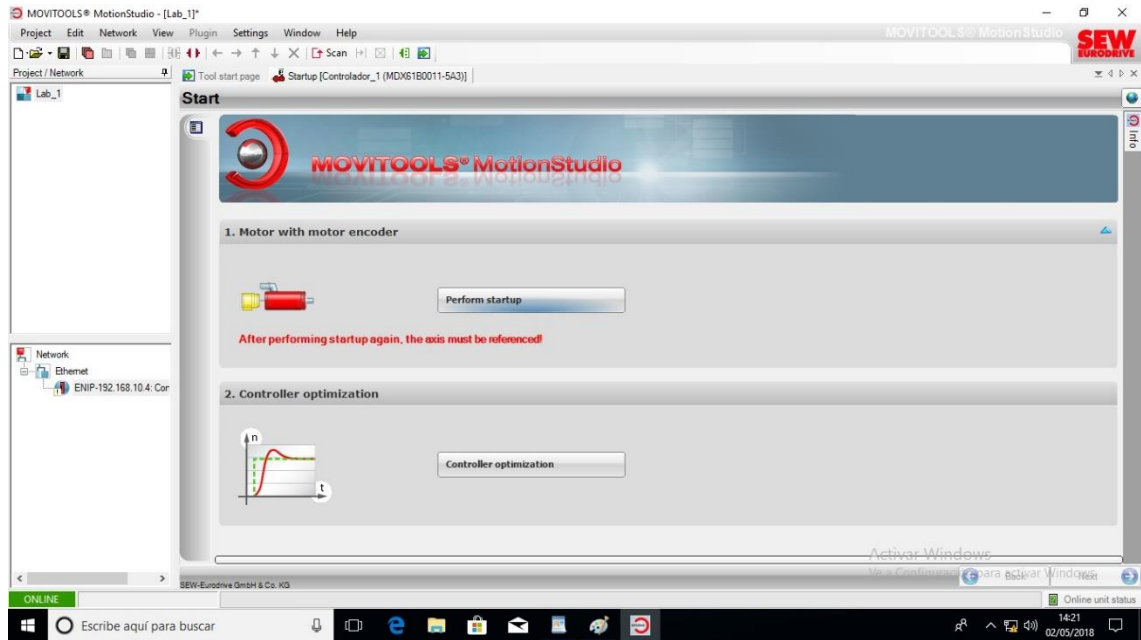


Imagen 1, 11 Configuración inicial para el control del servomotor.

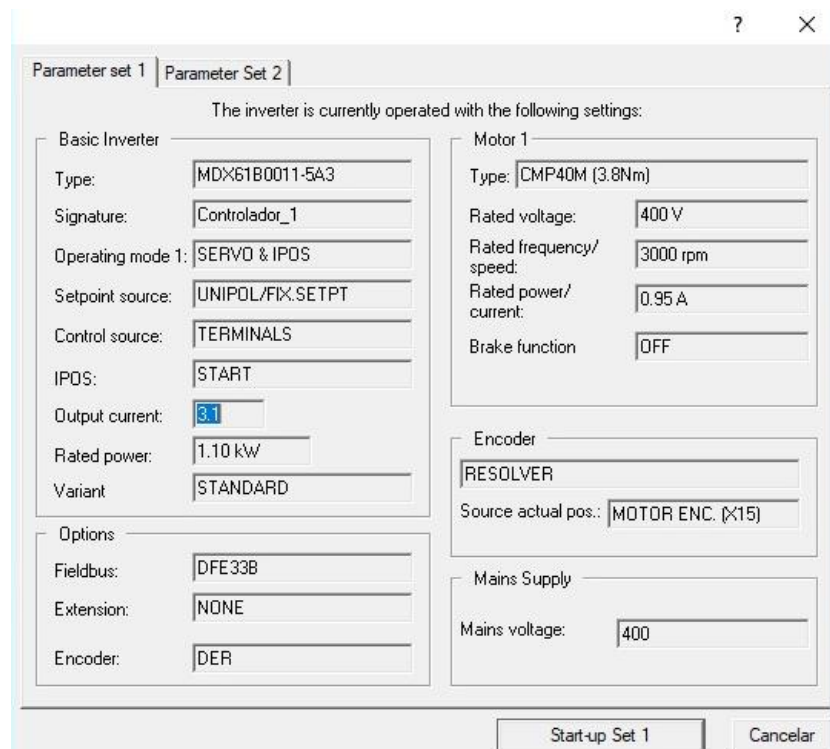


Imagen 1, 12 Configuración inicial para el control del servomotor.

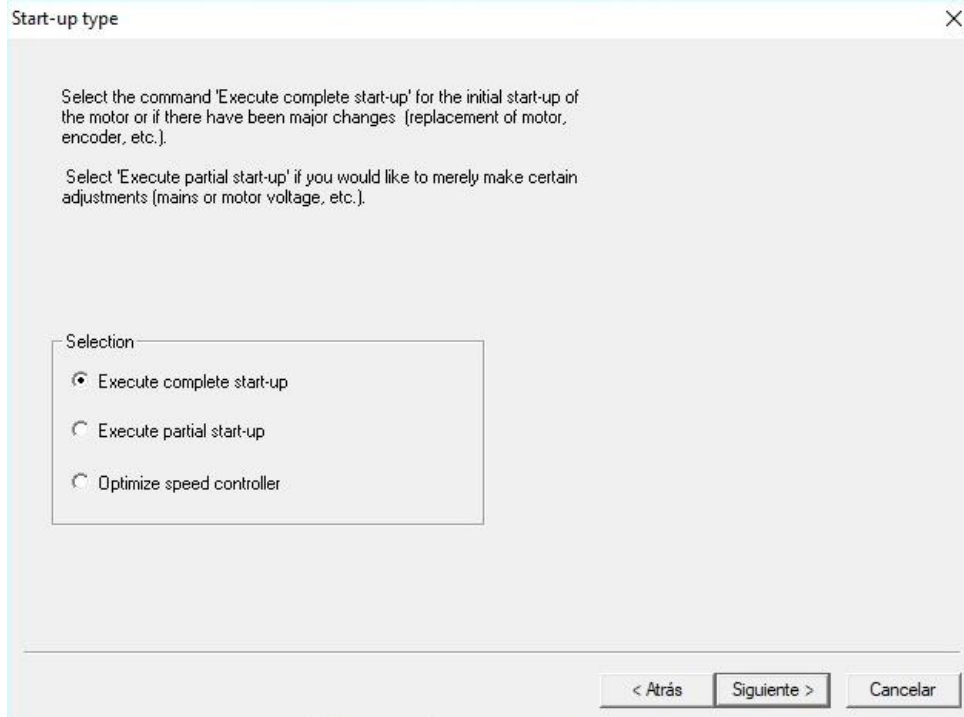


Imagen 1, 13 Configuración inicial para el control del servomotor.

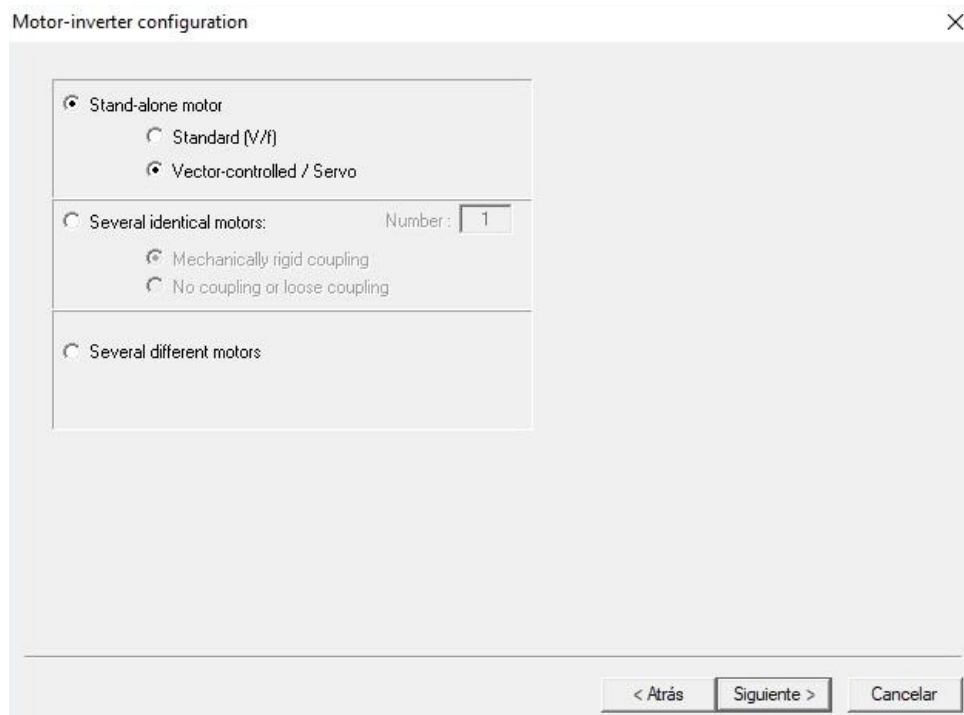


Imagen 1, 14 Configuración inicial para el control del servomotor.

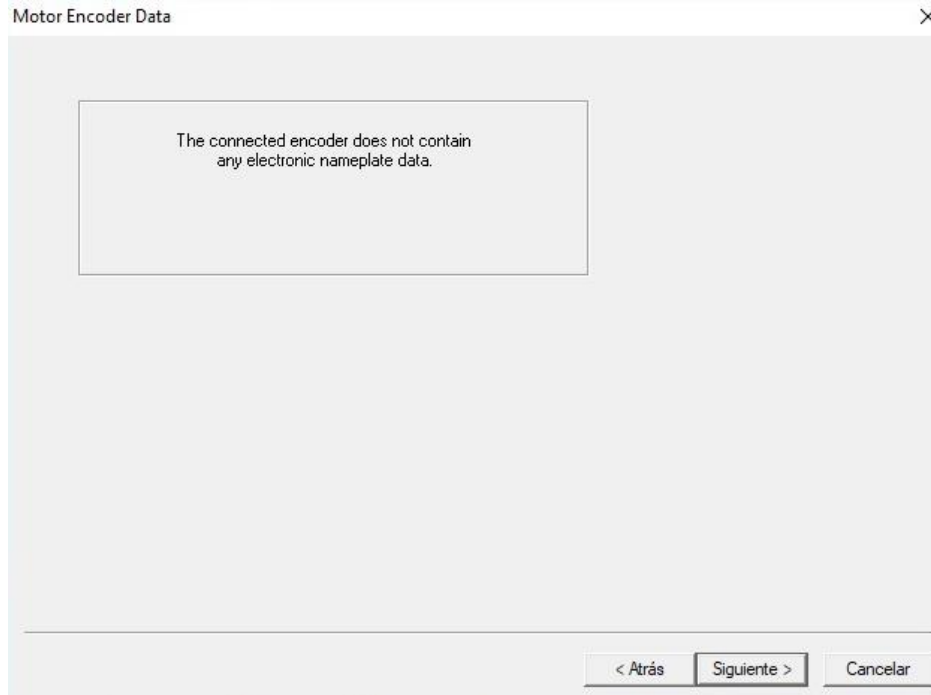


Imagen 1, 15 Configuración inicial para el control del servomotor.

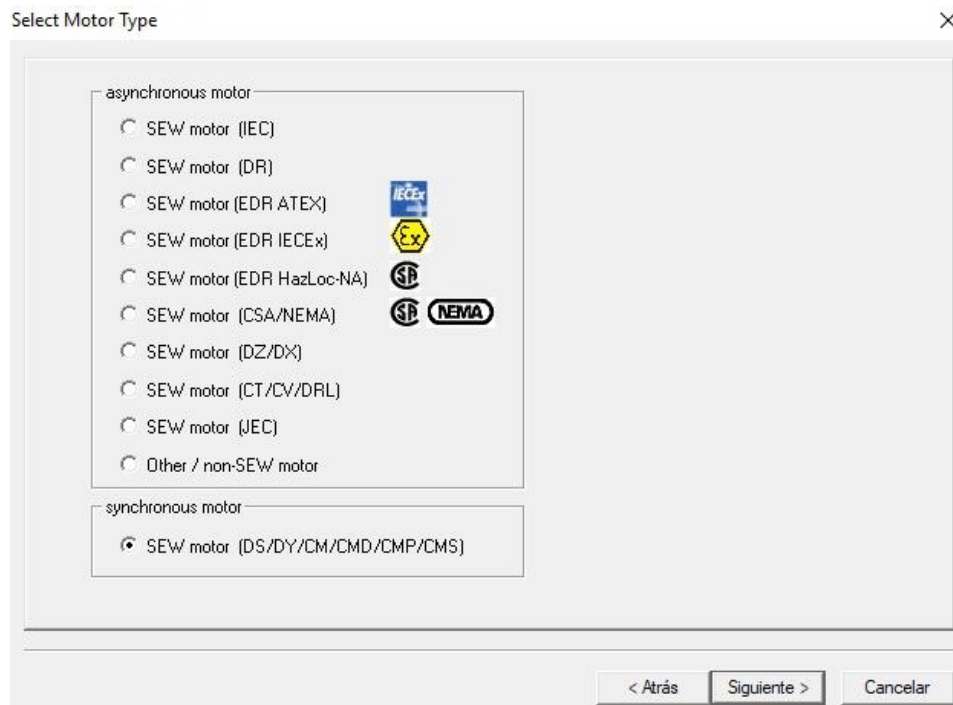


Imagen 1, 16 Configuración inicial para el control del servomotor.

servo ×

Motor type 1		CMP4QM (3.8Nm) ▾
Motor rated voltage	[V]	400 ▾
Motor rated speed	[rpm]	3000 ▾
Max. input speed gear unit	[rpm]	3000
Mains rated voltage	[V]	400
530 Sensor type 1		KTY ▾
832 Response MOTOR OVERLOAD		EMERG.STOP/FAULT ▾

Imagen 1, 17 Configuración inicial para el control del servomotor "Cualquier parámetro modificado inapropiadamente en este ítem podría dañar severamente el servomotor"

Possible applications ×

Operating mode

- Speed control
- Use hoist
- Use DC braking
- Use flying restart circuit
- Use synchronous operation control (DRS)
- Positioning with IPOS
- Use torque control

Imagen 1, 18 Configuración inicial para el control del servomotor.

Speed Controller Servo ×

Stiffness		<input type="text" value="1"/>	Testing determine automatically
Load inertia	[10e-4 kg]	<input type="text" value="0"/>	
Drive		WITHOUT BACI ▾	
Brake		WITHOUT ▾	
Shortest required ramp	[s]	<input type="text" value="1"/>	
Ext. control time ref.	[ms]	<input type="text" value="1"/>	
J0 motor	[10e-4 kg*m*m]	<input type="text" value="0.15"/>	
Number of slave axes		<input type="text" value="0"/>	

< Atrás Siguiete > Cancelar

Imagen 1, 19 Configuración inicial para el control del servomotor.

Parameters Servo ×

		Vorschlag	Download-Wert
303 Current limit 1	[%In]	<input type="text" value="193"/>	<input type="text" value="193"/>
304 Torque limit	[%In]	<input type="text" value="193"/>	<input type="text" value="193"/>
500 Speed monitoring 1		MOT. & REGEN.MODE	MOT. & REGEN.MODE ▾
501 Delay time 1	[s]	<input type="text" value="0.1"/>	<input type="text" value="0.1"/>
504 Encoder monitoring motor		NO	NO ▾
864 PWM frequency CFC	[kHz]	<input type="text" value="4"/>	<input type="text" value="4"/>

----->
Apply proposal

< Atrás Siguiete > Cancelar

Imagen 1, 20 Configuración inicial para el control del servomotor.

Speed Controller Parameters X

Parameters	Proposal	Download value
200 P gain speed controller	0.041	0.041
201 Time constant n-control.	460	460
202 Gain accel. feedforward	0.013	0.013
204 Filter speed actual	1.2	1.2
115 Filter setpoint [ms]	0	0
203 Filter accel.	1.19	1.19
210 P gain hold controller	7.62	7.62
910 Gain X controller	7.62	7.62

.....>
Apply proposal

< Atrás Siguiete > Cancelar

Imagen 1, 21 Configuración inicial para el control del servomotor.

Parameters X

	Proposal	Download value
100 Setpoint source		UNIPOL/FIX SETPT
101 Control signal source		TERMINALS
130 Ramp t11 UP CW [s]	1	1
131 Ramp t11 DOWN CW [s]	1	1
132 Ramp t11 up CCW [s]	1	1
133 Ramp t11 down CCW [s]	1	1
136 Stop ramp t13 [s]	1	1
137 Emergency ramp t14 [s]	1	1
301 Minimum speed 1 [rpm]	0	3000
302 Maximum speed 1 [rpm]	3000	3000
730 Brake function 1	OFF	OFF
731 Brake release time 1 [s]	0.1	0.1
732 Brake application time 1 [s]	0.1	0.1

.....>
Apply proposal

< Atrás Siguiete > Cancelar

Imagen 1, 22 Configuración inicial para el control del servomotor.

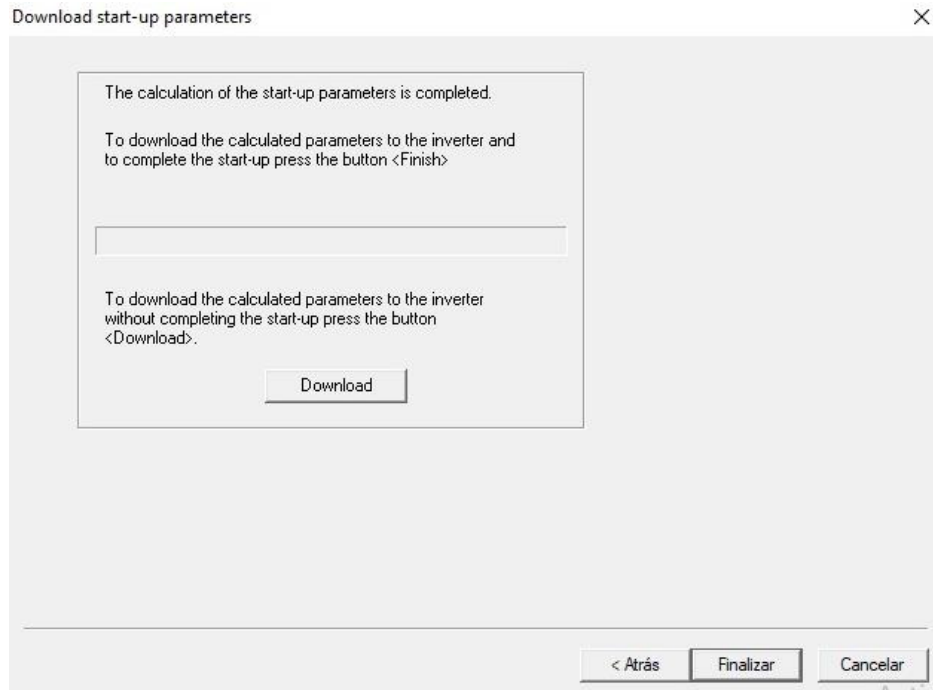


Imagen 1, 23 Configuración inicial para el control del servomotor. "No Apretar Download".

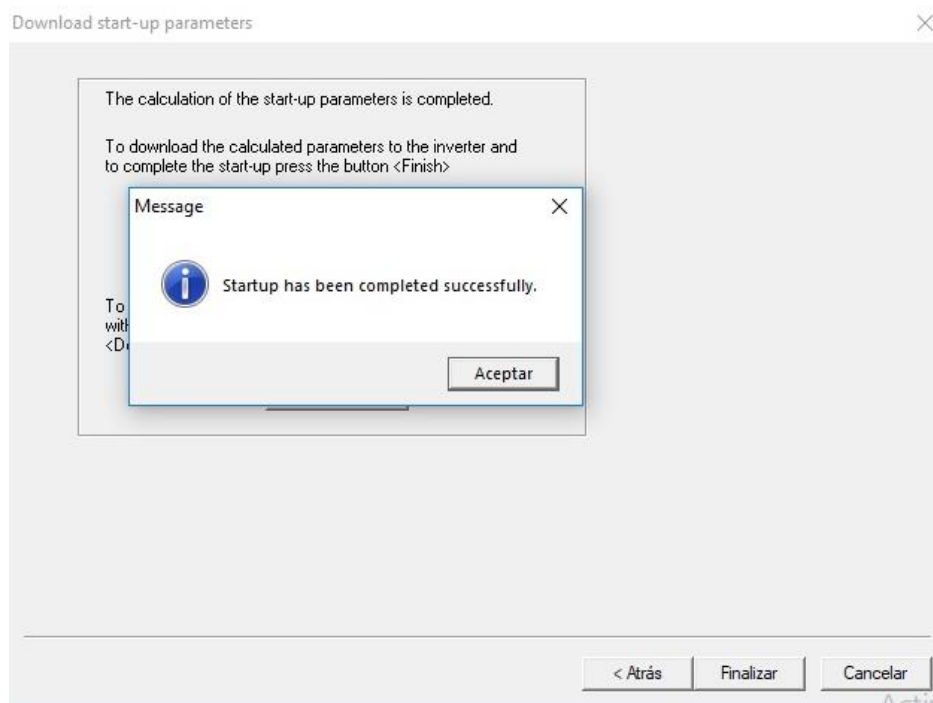


Imagen 1, 24 Configuración inicial para el control del servomotor.

A continuación, seleccionar la opción parameter tree en donde aparecerán una gran cantidad de opciones. (ver imagen 1.25)

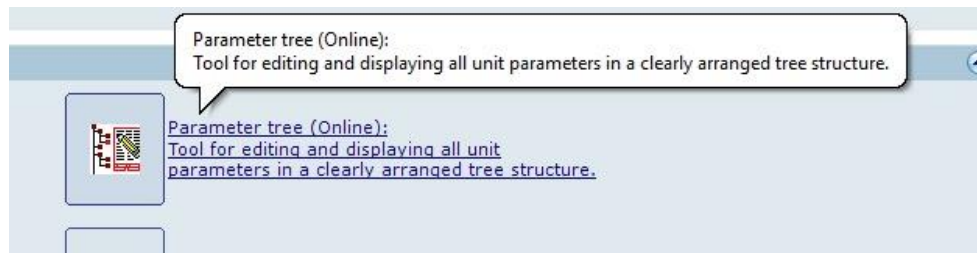


Imagen 1, 25 Para los parámetros del servomotor ingresar en "Parameter tree"

Ir al punto 3 y seleccionar "limits 1" en donde se puede configurar la velocidad máxima y mínima del servomotor en las opciones 301 y 302. (ver imagen 1.26)

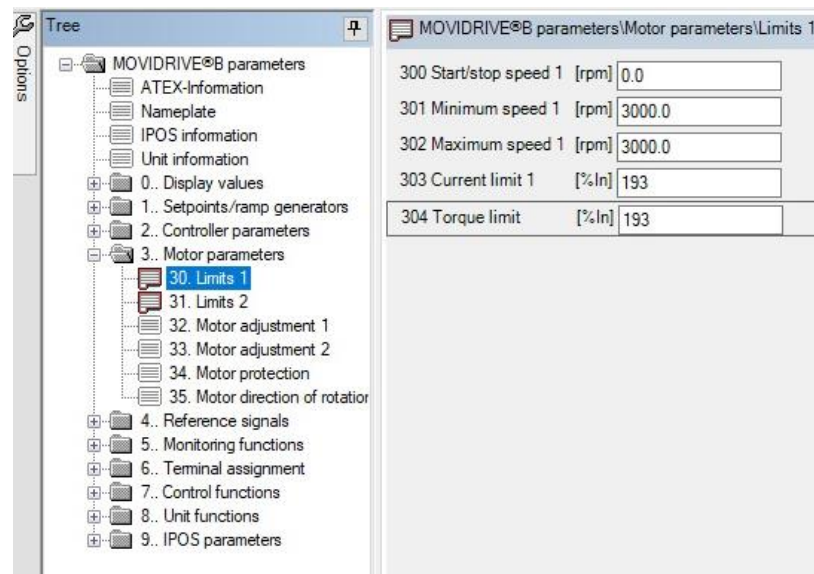


Imagen 1, 26 Configuración de parámetros de velocidad del servomotor.

El switch DI01 es para energizar el servomotor (Ver imagen 1.27) por lo que se activa de forma on/off a la velocidad de nuestros parámetros, que en el ejemplo son 3000rpm (Minimum speed 1), este valor puede ser modificado, **Como ejercicio trabajar con diferentes velocidades**

1)300rpm

2)1500rpm

3)2500rpm

4)100 rpm



Imagen 1, 27 Switch de la maqueta del servomotor.

LABORATORIO N°2: “ANÁLISIS VARIABLES SPEED CONTROLLER”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Analizar cada variable de speed controller (motor parameter)</i> • <i>Profundizar en la plataforma “MOVITOOLS”.</i> • <i>Realizar configuración speed controller.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>A continuación ya teniendo conocimiento de la configuración principal, se analizaran de forma más profunda cada una de las variables speed controller (motor parameter) de la ventana “parameter tree”.</i></p>		

2.1 CONFIGURACIÓN

Al realizar la configuración general seleccionar Speed control. (Ver imagen 2.1)

Possible applications

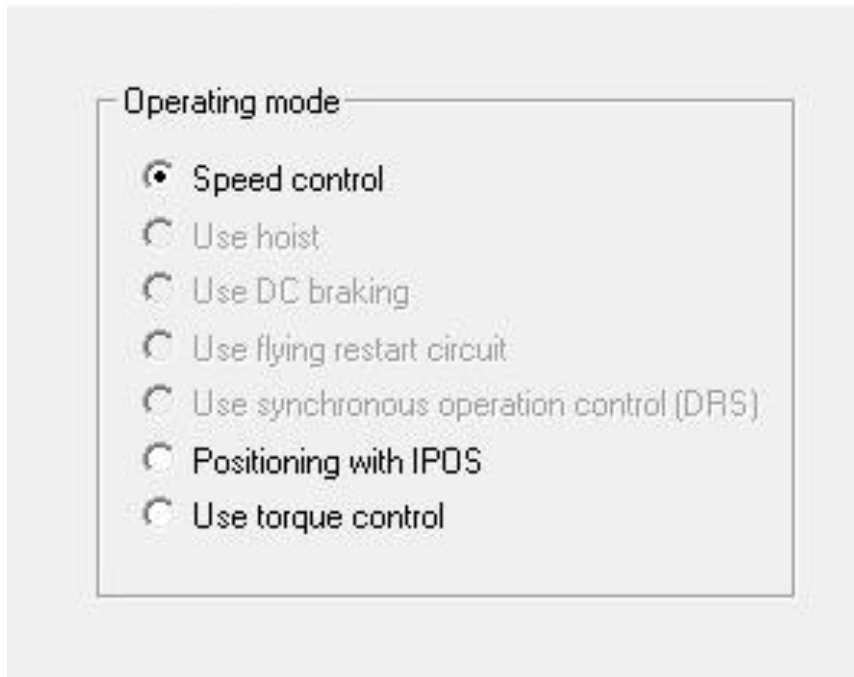


Imagen 2, 1 Diferentes modos de operación.

A continuación, seleccionar parameter tree en donde aparecerán una gran cantidad de opciones. (ver imagen 2.2)

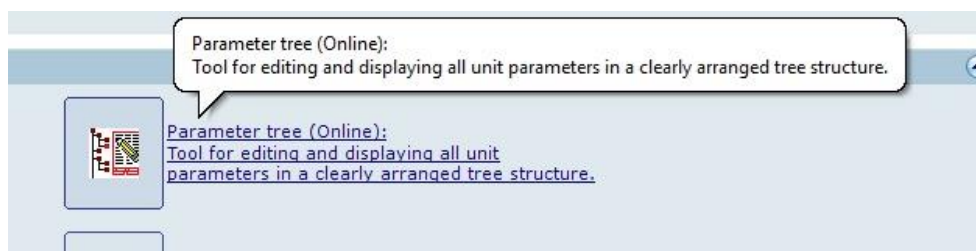


Imagen 2, 2 Parámetros para configuración del servomotor.

Donde nos enfocaremos principalmente en la sección 3 de “motor parameter”. Este grupo de parámetros se utiliza para ajustar el inversor al motor. Los parámetros se pueden configurar por separado para el conjunto de parámetros 1 y 2. Esto significa que se pueden operar dos motores diferentes alternativamente en el mismo inversor sin requerir un nuevo ajuste. (ver imagen 2.3)

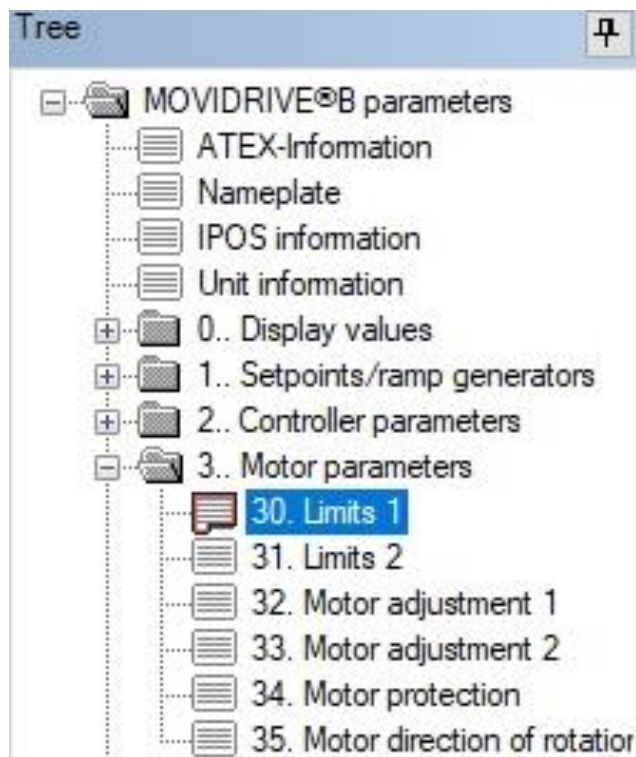


Imagen 2, 3 Configuración de velocidad del servomotor.

2.2 ANÁLISIS VARIABLES MOTOR PARÁMETROS

30/31. Limits 1/2: (Ver imagen 2.4)

300/310 Start/stop speed 1/2: Esta entrada define la solicitud de velocidad más pequeña que el inversor envía al motor cuando está habilitado. La transición a la velocidad determinada en la selección del punto de ajuste, se realiza usando la rampa de aceleración activa.

301/311 Minimum speed 1/2: Valor de velocidad, cuyo límite inferior no debe excederse incluso cuando se selecciona cero como punto de referencia. La velocidad mínima también se aplica cuando $n_{\min} < n_{\text{start/stop}}$ se establece.

302 /312 Maximum speed 1/2: El valor establecido aquí no puede excederse mediante una selección de punto de ajuste. Si se establece $n_{\min} > n_{\max}$, entonces se aplica n_{\max} . La velocidad máxima depende del modo de funcionamiento establecido

303/313 Current limit 1/2: La configuración de fábrica para la limitación de corriente se establece en 150% IN del motor correspondiente.

304 Torque limit 1: El parámetro limita el torque máximo del motor.

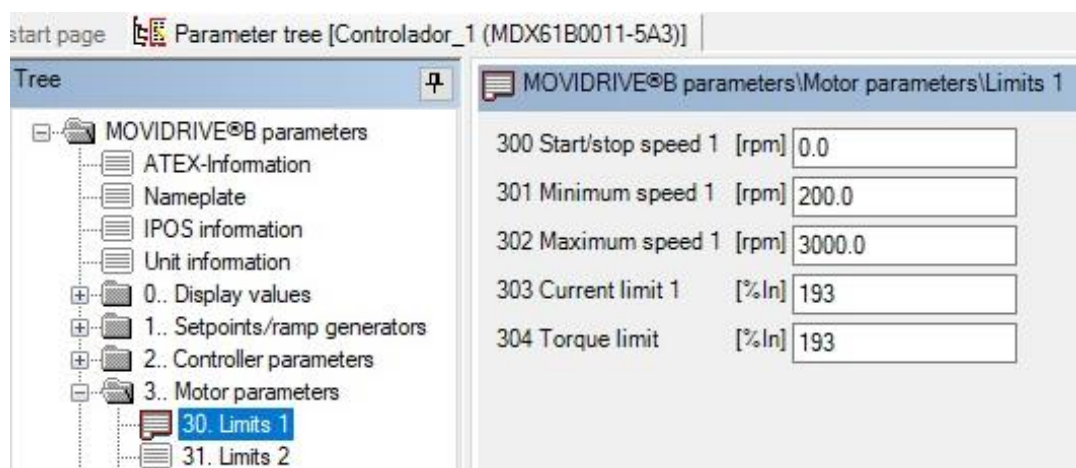


Imagen 2, 4 Parámetros de velocidad.

32/33. Adjustment 1/2: (Ver imagen 2.5)

320/330 Automatic adjustment 1/2: El inversor determina una configuración básica que es adecuada para una gran cantidad de aplicaciones de accionamiento.

321/331 Boost 1/2: Configuración manual generalmente no es necesaria. En casos excepcionales, la configuración manual puede ser necesaria para aumentar el torque de arranque.

322 /332 IxR compensation 1/2: Se realiza una configuración automática, la tensión de salida del inversor aumenta en 50 V cuando fluye la corriente nominal del motor.

323/333 Premagnetization time 1/2: La premagnetización sirve para establecer un alto torque del motor y se inicia cuando el inversor está habilitado.

324/334 Slip compensation 1: La compensación de deslizamiento aumenta la precisión de velocidad del motor. Si los valores se ingresan manualmente, deberá ingresar el deslizamiento nominal del motor conectado. Se permite un rango de ajuste de +/- 20% del deslizamiento nominal, si se ingresa un valor distinto del deslizamiento nominal para compensar las fluctuaciones entre varios motores.

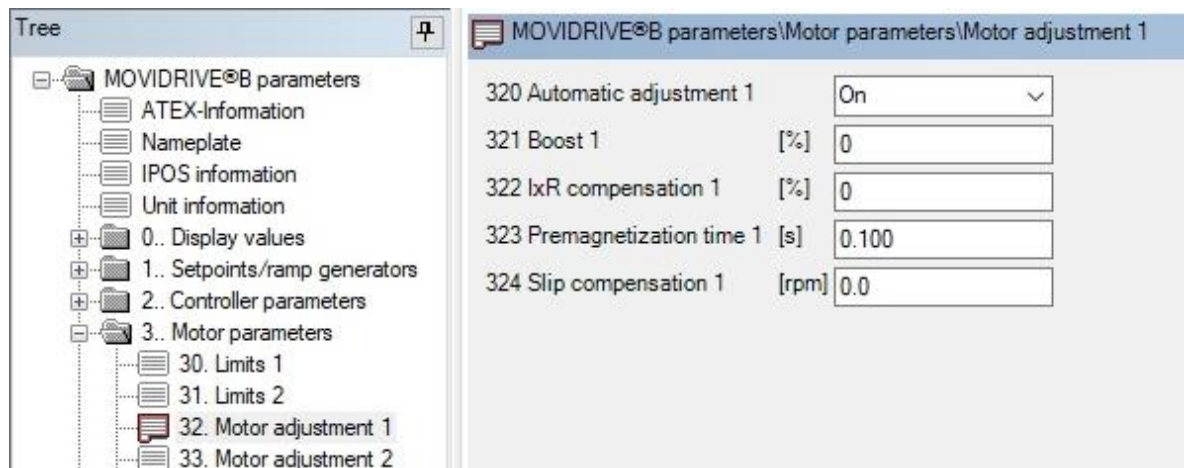


Imagen 2, 5 configuración para el servomotor.

34. Motor protection: (Ver imagen 2.6)

340/342 Motor protection 1/2: En la mayoría de los casos, la función de protección del motor es comparable a la protección térmica estándar (interruptor de protección del motor), además, tiene en cuenta la refrigeración dependiente de la velocidad del ventilador integrado. La utilización del motor se determina utilizando la corriente de salida del inversor, el tipo de refrigeración, la velocidad del motor y el tiempo.

341/343 Type of cooling 1/2: Rango de ajuste: **refrigeración por ventilación / reforzamiento forzado**. Debe conocer el tipo de refrigeración del motor para calcular la carga térmica del motor lo más exactamente posible.

344 Motor protection interval: No es relevante para motores asíncronos. Este parámetro corresponde al tiempo de ciclo del desplazamiento y se utiliza para la función P006 / P007. El rango de ajuste es de 100 ms a 20000 ms. (Siempre debe establecer el tiempo de avance y retroceso)

345/346 In-UL monitoring 1/2: La función no puede ser desactivada. La configuración de fábrica depende de la potencia nominal del MOVIDRIVE® B y está ajustada a la corriente nominal del motor SEW con la misma potencia

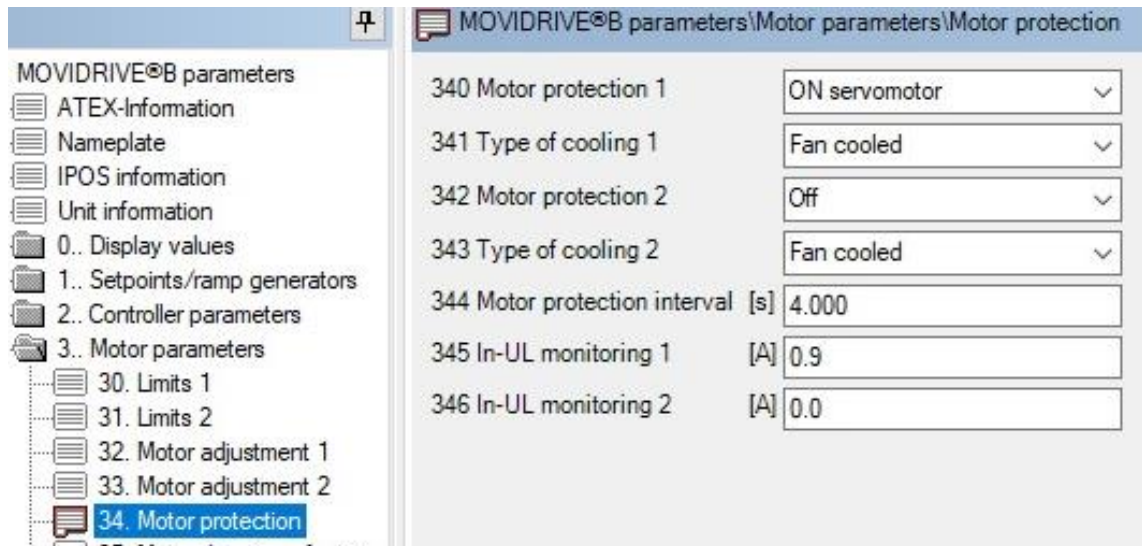


Imagen 2, 6 Parámetros de protección del servomotor.

35. Motor direction of rotation: (Ver imagen 2.7)

350/351 Motor direction of rotation 1/2: SEW-EURODRIVE especifica la dirección de rotación tal como se ve en el lado de accionamiento del motor. En sentido horario (positivo) se define como rotación hacia la derecha y en sentido anti-horario (negativo) como rotación hacia la izquierda. Esta definición se implementa cuando el motor está conectado de acuerdo con la designación SEW.

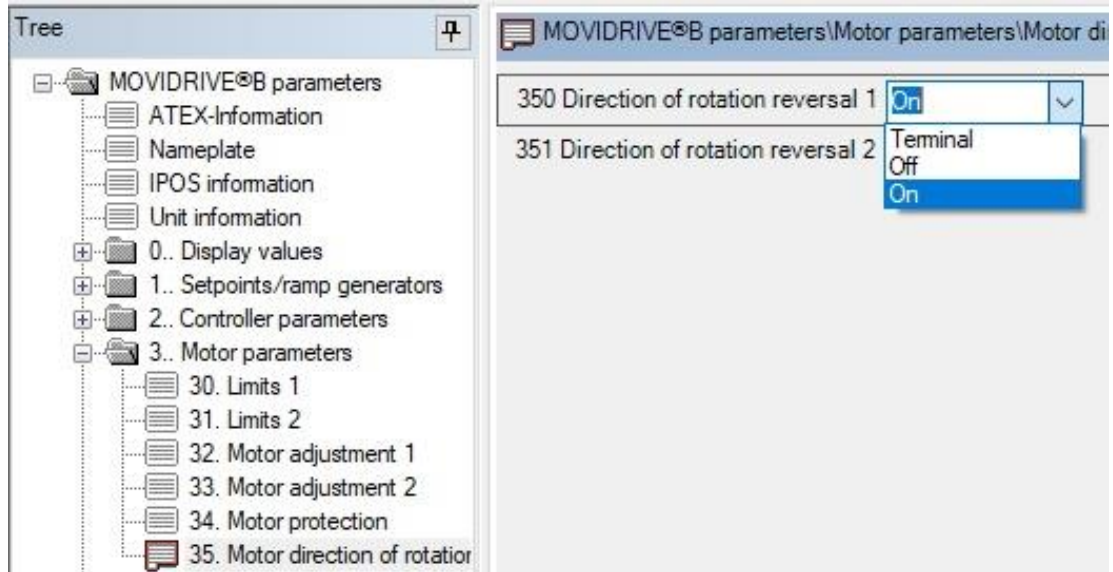


Imagen 2, 7 configuración de dirección del servomotor.

Como ejercicio trabajar con:

1)100rpm anti-horario 2)100rpm horario 3)2300rpm anti-horario

LABORATORIO N°3: “CONTADOR CON IPOS COMPILLER”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Realizar configuración IPOS.</i> • <i>Analizar ventana IPOSplus COMPILLER MOVITOOLS.</i> • <i>Crear código en IPOS compiler.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>A continuación ya teniendo conocimiento de la configuración principal, nos enfocaremos en el modo de operación Positioning with IPOS y realizar un ejemplo con las funciones del sistema.</i></p>		

3.1 CONFIGURACIÓN

Al realizar la configuración general, seleccionar “Positioning with IPOS”. (Ver imagen 3.1)

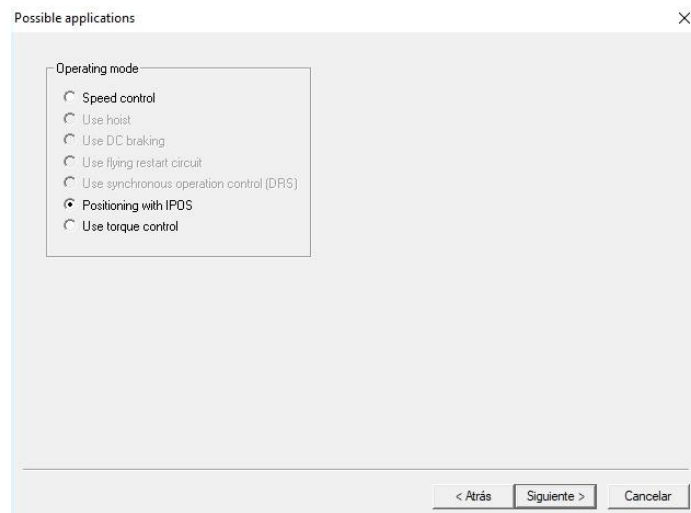


Imagen 3, 1 Configuración por IPOS.

A continuación, debajo de la opción Ethernet realizar clic secundario en el controlador, donde aparecerán una variedad de opciones. (Ver imagen 3.2)

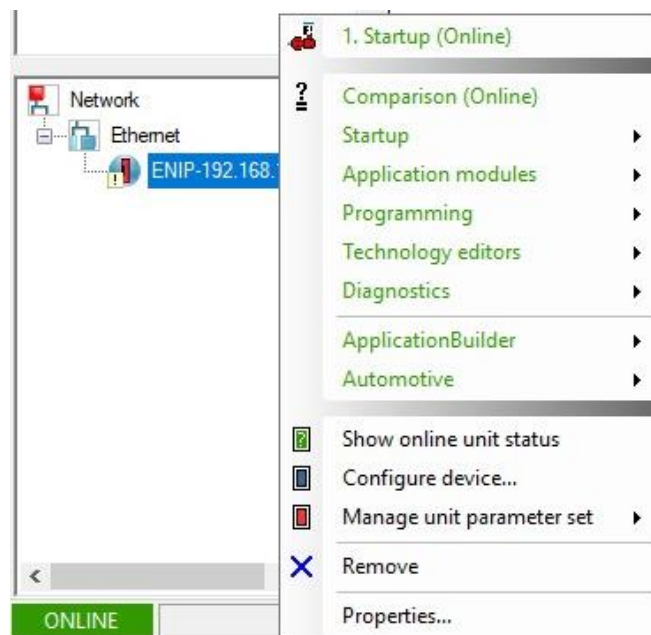


Imagen 3, 2 Opciones de comunicación por IPOS.

Ir a la opción “Programming” y seleccionar “IPOS Compiler(Online)” (Ver imagen 3.3)



Imagen 3, 3 Opciones de compilación en IPOS.

Se abrirá una nueva ventana (IPOSplus COMPILER MOVITOOLS) y seleccionar “new” en la barra de tareas. (Ver imagen 3.4)

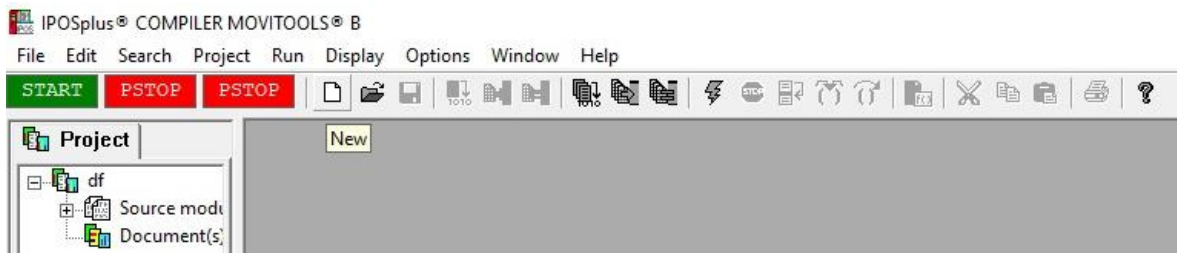


Imagen 3, 4 Crear nueva planilla.

Crear un nuevo proyecto con nombre a elección, de preferencia relacionado a lo que se realizara (Ver imagen 3.5)

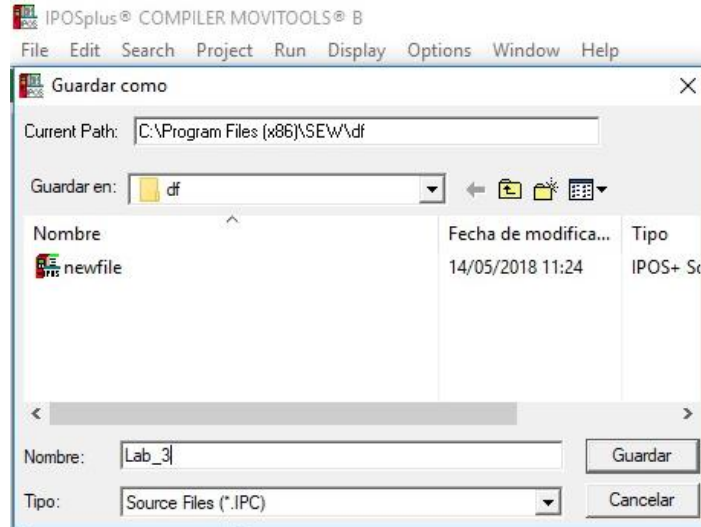


Imagen 3, 5 Creación de planilla de programa..

Para la inicialización de la planilla de programación en la opción de “create program structure” se debe seleccionar “initialisation part” en la opción principal y “OK”. (Ver imagen 3.6)

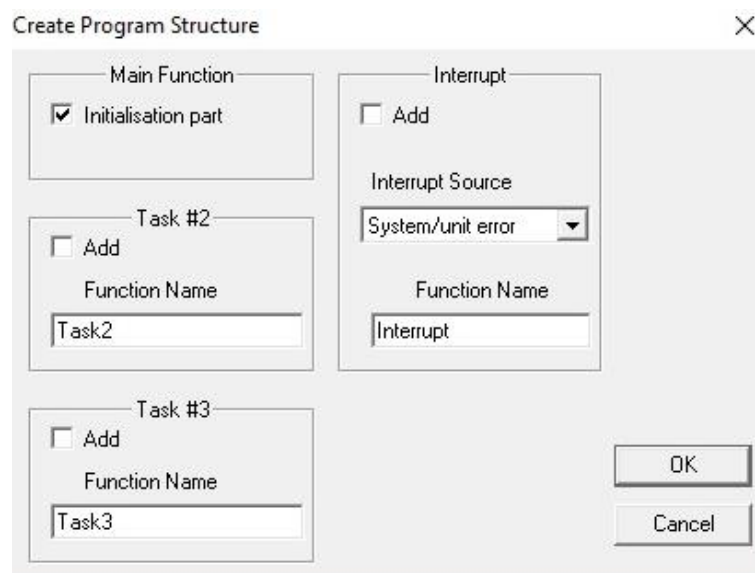


Imagen 3, 6 Selección de parámetros para la creación de programas..

Ya en la plantilla, esta posee una estructura predefinida, la cual en su encabezado se declaran las variables, aquí se encuentra ya definido `#include <const.h>` (incluye un archivo “header” llamado const.h) e `#include <io.h>` (incluye un archivo “header” llamado io.h) los cuales son archivos que ya tienen un formato fijo y no se deben cambiar. (Ver imagen 3.7)



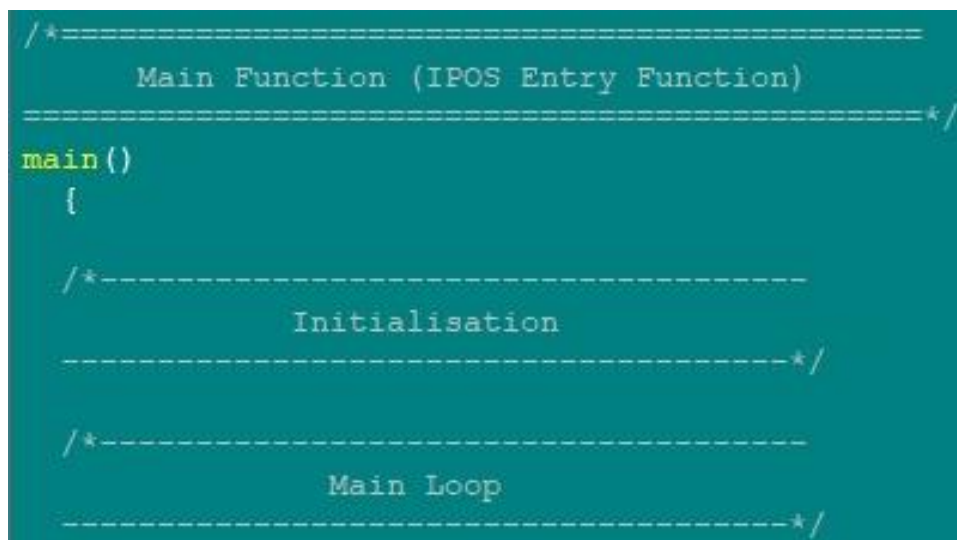
```

/*=====
IPOS Source File
=====*/
#include <constb.h>
#include <iob.h>

```

Imagen 3, 7 constante preestablecidas

En medio se encuentra el main, aquí se escriben las condiciones que solo se ejecutaran una vez en el programa. (Ver imagen 3.8)



```

/*=====
Main Function (IPOS Entry Function)
=====*/
main()
{

/*-----
Initialisation
-----*/

/*-----
Main Loop
-----*/

```

Imagen 3, 8 Estructura de una sola ejecución de programa.

Y finalmente el while (1) en el cual el código creado dentro de este, se ejecutará cíclicamente, de arriba hacia abajo, de izquierda a derecha. (Ver imagen 3.9)



```

while (1)
{

}

}

```

Imagen 3, 9 estructura del programa que se ejecuta periódicamente a través de una condición..

En la barra de tareas superior existe la opción “Display” en donde se puede mostrar, identificar y crear todas las variables. (Ver imagen 3.10)

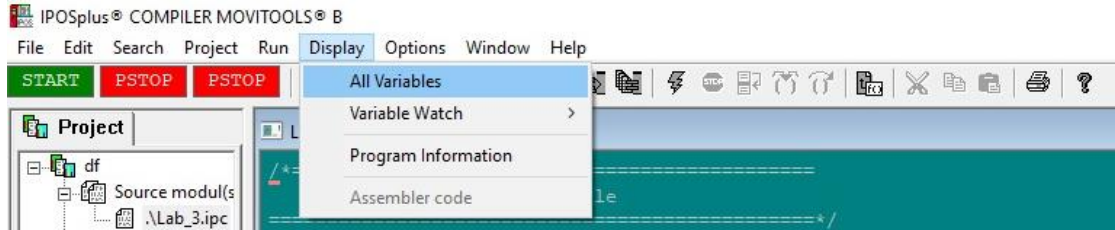


Imagen 3, 10 Opción para visualizar variables.

En la memoria existen variables predefinidas las cuales son de la H453 hasta la H560 y el resto son espacios libres para definir nuestras variables. (Ver imagen 3.11)

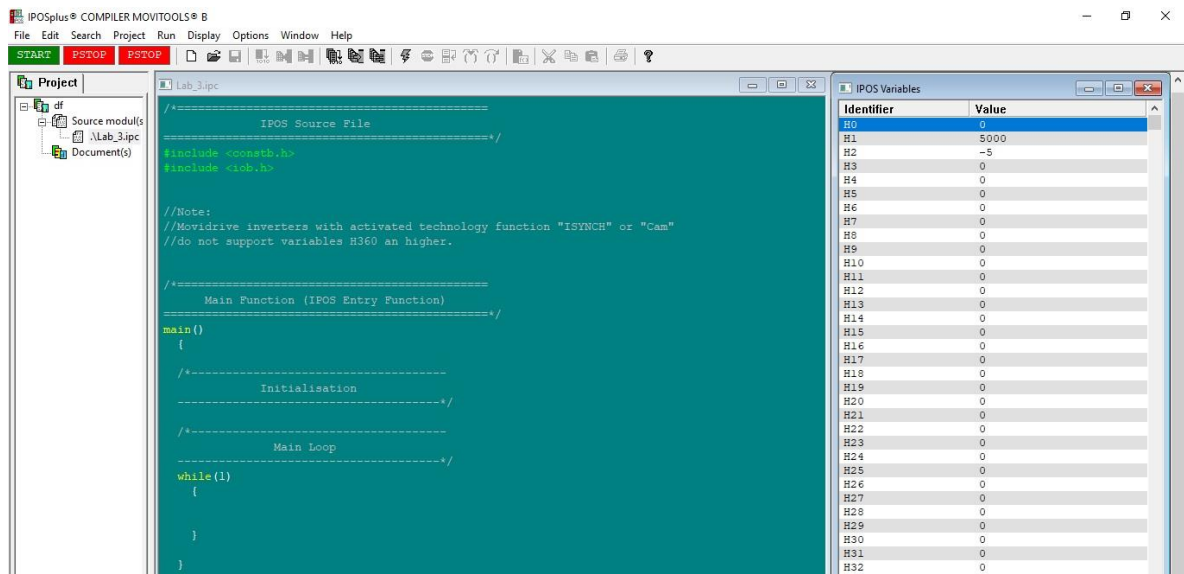


Imagen 3, 11 Visualización de variables.

Para compilar el programa y verificar que esto esté correcto, seleccionar la opción “Compile file”. (ver imagen 3.12)



Imagen 3, 12 Compilación del programa.

Al compilar el programa nos aseguramos de que no posea ningún error y aparecerá una ventana corroborando la información. (ver imagen 3.13)

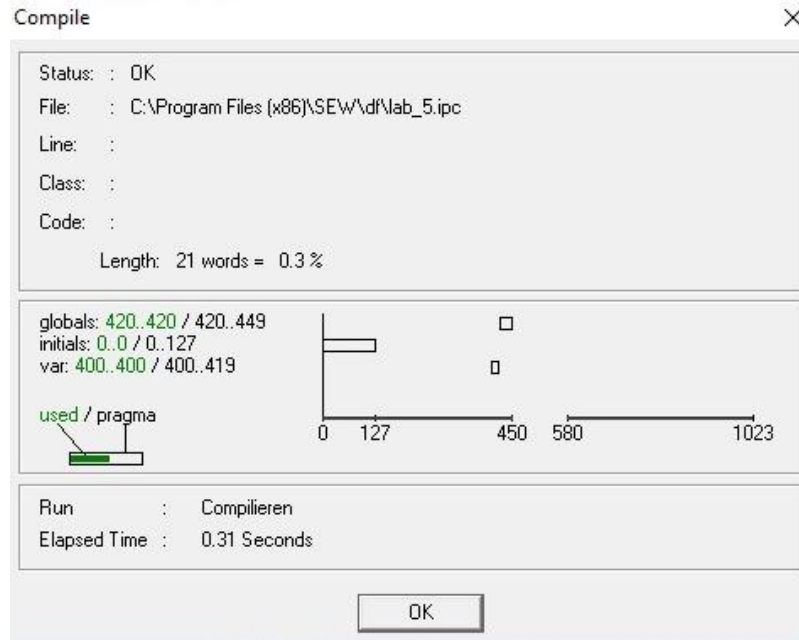


Imagen 3, 13 Correcta compilación del programa.

Si estamos equivocados nos señalará por qué de nuestro error y en qué línea estamos equivocados. (ver imagen 3.14)

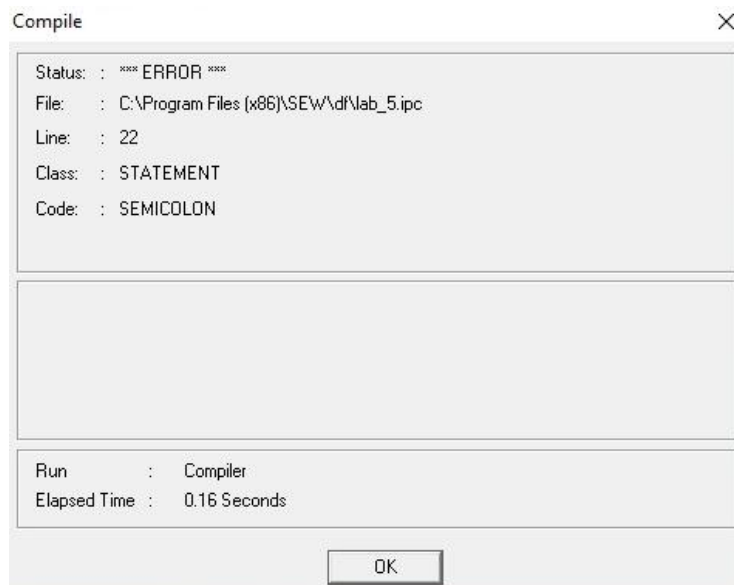


Imagen 3, 14 Errónea compilación del programa.

Si tenemos todo bien en nuestro código podremos cargar el programa sin ningún tipo de problema. (ver imagen 3.15)



Imagen 3, 15 Compilación y carga del programa.

Ya cargado nuestro programa, seleccionamos la opción “start program” para iniciarlo. (ver imagen 3.16)



Imagen 3, 16 Ejecución del programa.

3.2 CONSTRUCCIONES EN C

If e if else

if

- ¿Se ejecuta si la expresión dentro del paréntesis es verdadera ($! = 0$).
- La expresión puede contener varias condiciones.

else

- Se ejecuta si la expresión dentro del paréntesis es falsa ($= 0$).
- El else no tiene que especificarse si no es necesario.

For

- La declaración “For” permite hacer un bucle de programa

for (Expresion1; Expresion2; Expresion3)

- Expresion1= Se ejecuta al comienzo de bucle. Se inicializa la variable que determina el bucle.
- Expresion2 = Determina cuando se rompe el bucle (FALSO).

- Expresion3 = Se procesa luego de ejecutarse la declaración. Se utiliza para alterar la variable de la Expresion1.

While

- La declaración “while” es un bucle condicional que se repite cada vez que la expresión es verdadera.
- La condición se verifica al comienzo del bucle.

Do while

- La declaración “do” es un bucle condicional en donde la condición se verifica al final del bucle.
- La declaración al menos se ejecuta una vez.
- La declaración se ejecuta nuevamente si la expresión es verdadera al final del bucle.

Switch case

- La declaración “switch” hace posible crear múltiples bifurcaciones de programa dependiendo del valor de la expresión.
- Se puede utilizar una bifurcación por default como la última parte de la declaración de “switch”.

Ejercicio: Escriba un programa en el compilador para realizar un contador con una pausa de 3 segundos almacenado la variable (ver resultado en IPOS variable). Ya teniendo el conocimiento de las “Construcciones en C”.

Solución:

```
Lab_3.ipc
/*=====
IPOS Source File
=====*/
#include <constb.h>
#include <iob.h>

//Note:
//Movidrive inverters with activated technology function "ISYNCH" or "Cam"
//do not support variables H360 an higher.

/*=====
Main Function (IPOS Entry Function)
=====*/
main()
{
/*-----
Initialisation
-----*/
H2=0;
/*-----
Main Loop
-----*/
while(1)
{
for(H1=0; H1<= 100; H1++)
{
H2=H2+H1; // Add.
Wait( 3000 ); // Wait threeseconds.
}
}
```

Imagen 3, 17 Ejemplo de programación.

LABORATORIO N°4: “CONTROL DE MOTOR UTILIZANDO GOABS, GOREL Y GO0”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Realizar configuración IPOS.</i> • <i>Controlar motor mediante comandos.</i> • <i>Aprender a utilizar los comandos GoAbs, GoRel y Go0.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>El laboratorio a realizar consta principalmente de crear un programa por configuración IPOS con comandos GoAbs, Gorel y Go0, y así realizar distintos tipos de rutinas con el motor.</i></p>		

4.1 COMPARADORES LÓGICOS

Son muy utilizados en la electrónica para establecer condiciones y, por consiguiente, acciones que nos permiten desenvolvemos respecto a nuestras necesidades al momento de instalar o hacer un sistema de acuerdo a lo solicitado, en base tanto a operadores matemáticos como booleanos.

Operadores matemáticos

Son principalmente los comandos básicos de suma (+), multiplicación(*), resta(-) y división(/), además del módulo, que es el “resto” de una operación de división (%).

Operadores Lógicos

Son aquellos que establecen diferencias o igualdades entre 2 o más variables.

==	Igual que
!=	Distinto que
<	Menor que
>	Mayor que
<=	Menor o igual que
>=	Mayor o igual que

4.2 FUNCIONES DE COMANDO DE POSICIÓN

_Go0: Este comando es la búsqueda del “home” por lo que el servo hará un recorrido al punto de referencia. El argumento define el tipo de viaje de referencia. (Ver imagen 4.1)

El argumento puede adoptar diferentes valores

El significado de las letras son las siguientes:

- C (conditional) = Viaje de referencia solo si el viaje de referencia aún no se ha realizado.
- U (Unconditional)= Siempre referenciado, independientemente de si el eje ya está referenciado o no.

- W (Wait)= Espera en esta línea de declaración hasta que se realice el viaje de referencia.
- NW (NoWait)= Procesar la siguiente línea de extracto durante el recorrido de referencia realizado.
- ZP (Zero Pulse) = Recorrido de referencia a pulso cero.
- CAM= Recorrido de referencia a cámara.
- RESET= El recorrido de referencia que se inició se interrumpe y la llamada se restablece.

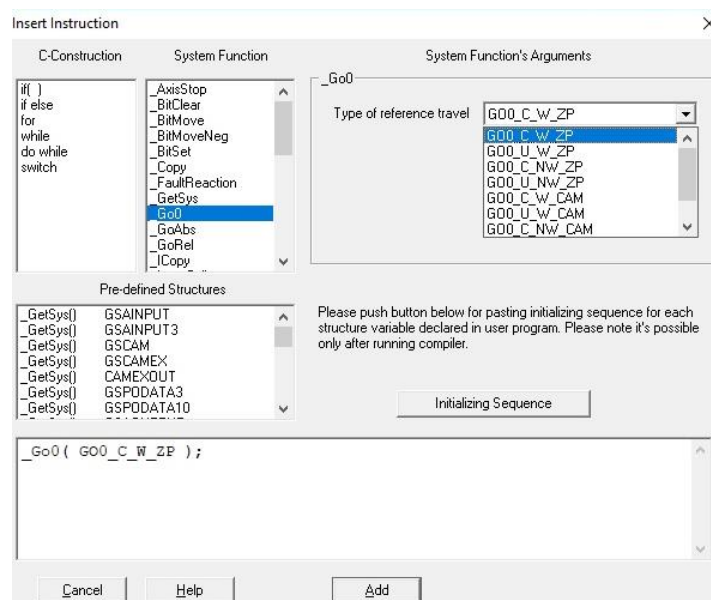


Imagen 4, 1 Comandos de Go0.

– **_GoAbs**: Este comando permite el posicionamiento en función de un punto de referencia. (Ver imagen 4.2)

La expresión para el tipo de comando de movimiento puede adoptar una de las siguientes opciones:

- GO_NOWAIT: Sin esperar se reanuda el procesamiento del programa en la siguiente línea de extracto inmediatamente después de enviar el comando de movimiento.
- GO_WAIT: Espera en esta línea de declaración hasta que se complete el viaje.

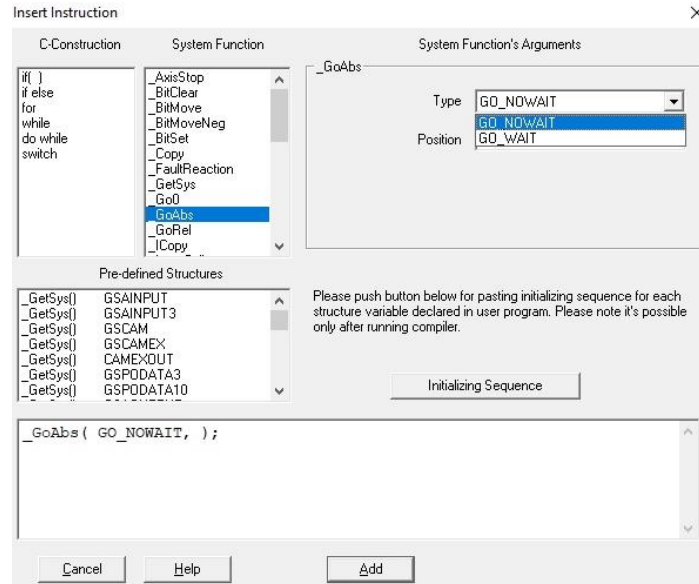


Imagen 4, 2 Comandos GoAbs.

– **_GoRel**: Posicionamiento relativo a una distancia basada en la posición actual. (Ver imagen 4.3)

- GO_NOWAIT: Sin esperar se reanuda el procesamiento del programa en la siguiente línea de extracto inmediatamente después de enviar el comando de movimiento.
- GO_WAIT: Espera en esta línea de declaración hasta que se complete el viaje.

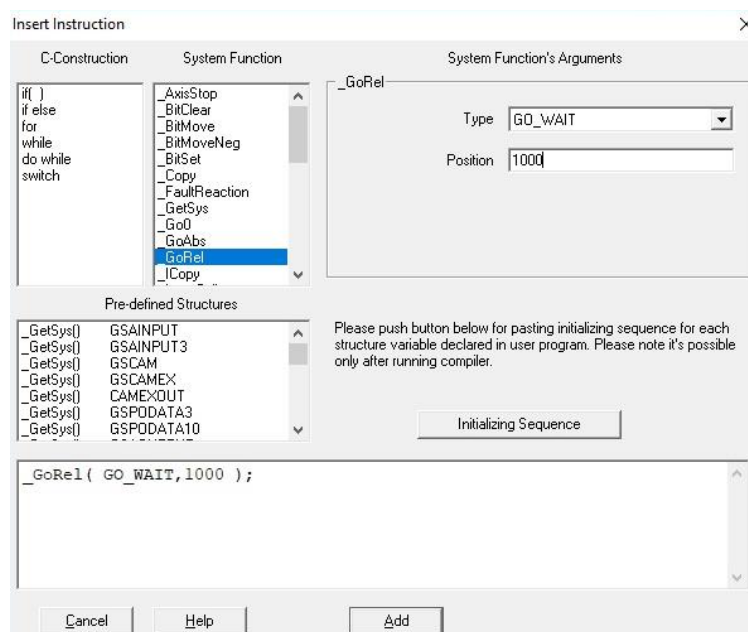


Imagen 4, 3 Comandos GoRel.

Cuando se utilizan comandos de posición, los parámetros IPOS (P9xx) tienen que programarse.

Cuando se utilizan comandos de posición, P700 tiene que programarse como “IPOS mode”.

4.3 DECLARACIÓN DE VARIABLES

#define permite definir una macro para reemplazar símbolos en la fuente de texto, esta posee las siguientes limitantes:

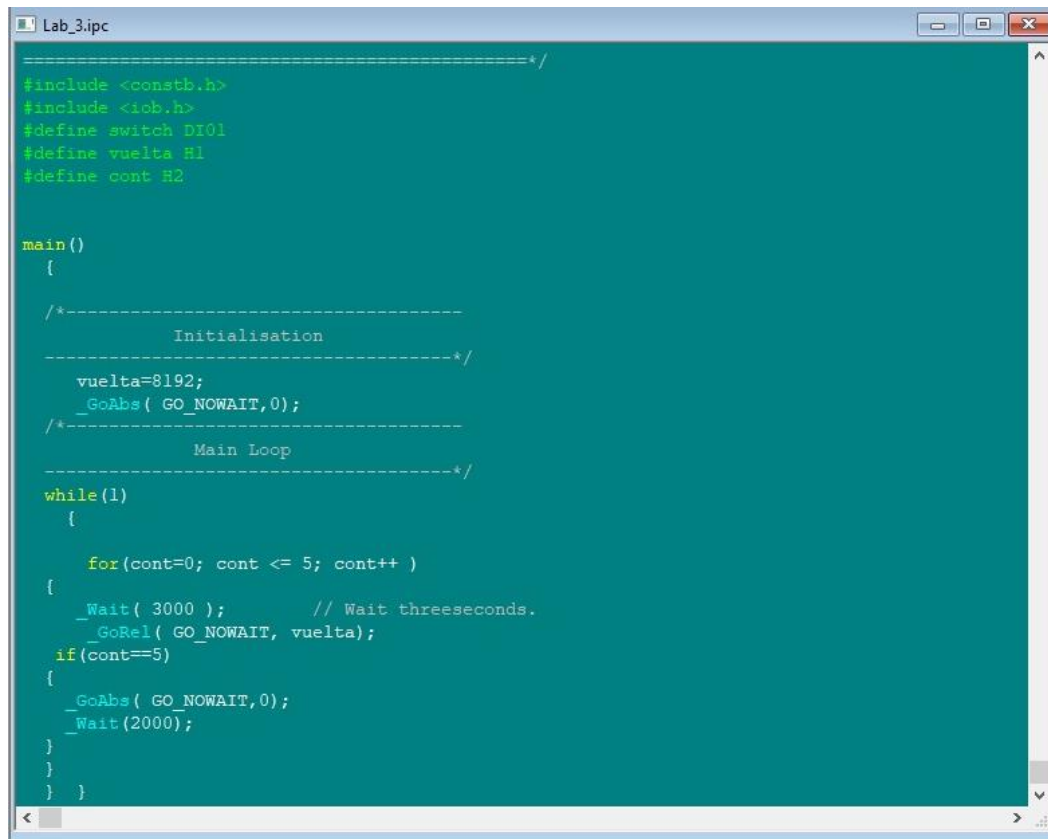
- Un nombre de variable simbólico o un nombre de función
- Solo letras, números, y el guion bajo (_) se pueden usar
- Debe empezar con una letra o un guion bajo
- Pueden ser de hasta 32 caracteres de longitud
- Use mayúscula para constantes (Ej. #define MAXRPM 3200)
- Use mayúscula para la primera letra de una variable

Ej. #define VarTime1 H320

Recordar:

El servomotor SEW posee un encoder absoluto codificado que posee 3 discos adicionales de 4 bits vinculados mediante reducciones $i=16$ puede definirse un rango de resolución absoluto de: $16 \times 16 \times 16 = 4096$ representaciones lo cual es 1 vuelta.

Ejercicio: Programar 2 giros cada 3 segundos horario, al realizar 10 vueltas esperar 2 segundos y volver a la posición inicial. (Ver imagen 4.4)



```
=====*/
#include <constb.h>
#include <iob.h>
#define switch DI01
#define vuelta H1
#define cont H2

main()
{
    /*-----
    Initialisation
    -----*/
    vuelta=8192;
    _GoAbs( GO_NOWAIT,0);
    /*-----
    Main Loop
    -----*/
    while (1)
    {
        for(cont=0; cont <= 5; cont++ )
        {
            _Wait( 3000 );           // Wait threeseconds.
            _GoRel( GO_NOWAIT, vuelta);
            if(cont==5)
            {
                _GoAbs( GO_NOWAIT,0);
                _Wait(2000);
            }
        }
    }
}
```

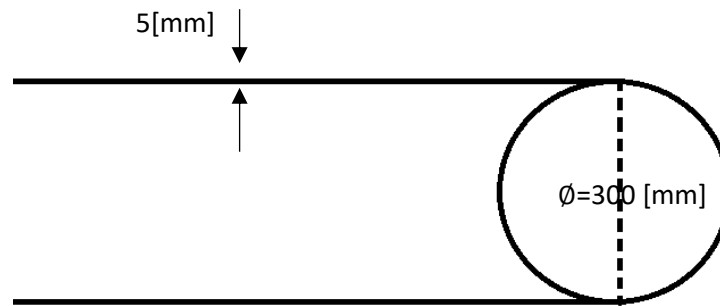
Imagen 4, 4 Código de ejemplo.

LABORATORIO N°5: “PROBLEMA REAL”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Analizar problemática real de empresa.</i> • <i>Dar solución mediante el conocimiento matemático.</i> • <i>Crear programa para su funcionamiento mediante conocimientos ya adquiridos.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>Entregar al estudiante una problemática real de una empresa para la preparación de este al mundo laboral, relacionándolo con el análisis matemático y la posible solución a este mediante el servomotor.</i></p>		

5.1 PROBLEMÁTICA EMPRESARIAL

Una empresa X tiene una celda flexible para su producción de botellas, el sistema posee un servomotor conectado a un reductor 20:1, el cual está conectado a una correa transportadora a través de un sistema de engranaje. La empresa solicita que la cinta avance 25[cm] y luego se detenga 1 segundo para mantener el control de sus productos.

Se sabe que el radio del disco son 150[mm] y el ancho de la cinta 5[mm].



5.2 ANÁLISIS DEL PROBLEMA

Lo que realmente nos interesa del problema es:

- Posee un reductor 20:1
- La cinta posee 5[mm]
- El disco posee un radio de 150[mm]
- La cinta debe avanzar 25[cm]

Se deben tomar en cuenta algunas fórmulas matemáticas para la comprensión de esta problemática, primero necesitamos calcular el perímetro, para eso se tomó en cuenta el ancho de la cinta y el radio del disco para calcular el radio total, ya con el radio total se pudo proceder al cálculo del perímetro.

Obtenido ya el perímetro procedemos a hacer la equivalencia entre las vueltas del servomotor con los 25[cm] de avance, ya con esto, calculamos a cantidad de representaciones necesarias para programar el servomotor.

5.3 SOLUCIÓN A LA PROBLEMÁTICA

Recordar que para cualquier calculo a realizar, se deben tomar en cuenta la mayor cantidad de decimales posible, para que el margen de error sea el mínimo debido a que la memoria del drive no acepta numero en formato flotante (decimales).

Calculo de perímetro

Tomar en cuenta para ello la suma del ancho de la cinta con el radio del disco.

Recordar que el perímetro de un circulo está dado por: $P = 2 \pi r$.

Ya con esto procedemos a calcular:

$$P = 2 \pi 15,5[\text{cm}]$$

$$P = 97,38937226 [\text{cm}]$$

Ya obtenido el perímetro procedemos a relacionarlo con las vueltas del servomotor

$$97,38937226 [\text{cm}] \longrightarrow 20 \text{ vueltas del servomotor}$$

$$25[\text{cm}] \longrightarrow x \text{ vueltas del servomotor}$$

Se realiza una regla de 3 simple obteniendo:

$$(25 [\text{cm}] \times 20[\text{vueltas}]) / 97,38937226 [\text{cm}]$$

$$= 5,134030422 [\text{vueltas}]$$

Una vez calculado la cantidad de vueltas requerida, lo relacionamos con la cantidad de representaciones necesarias para una vuelta según el driver.

$$1[\text{vuelta}] \longrightarrow 4096 [\text{representaciones}]$$

$$5,134030422 [\text{vueltas}] \longrightarrow x [\text{representaciones}]$$

Se realiza una regla de 3 simple obteniendo:

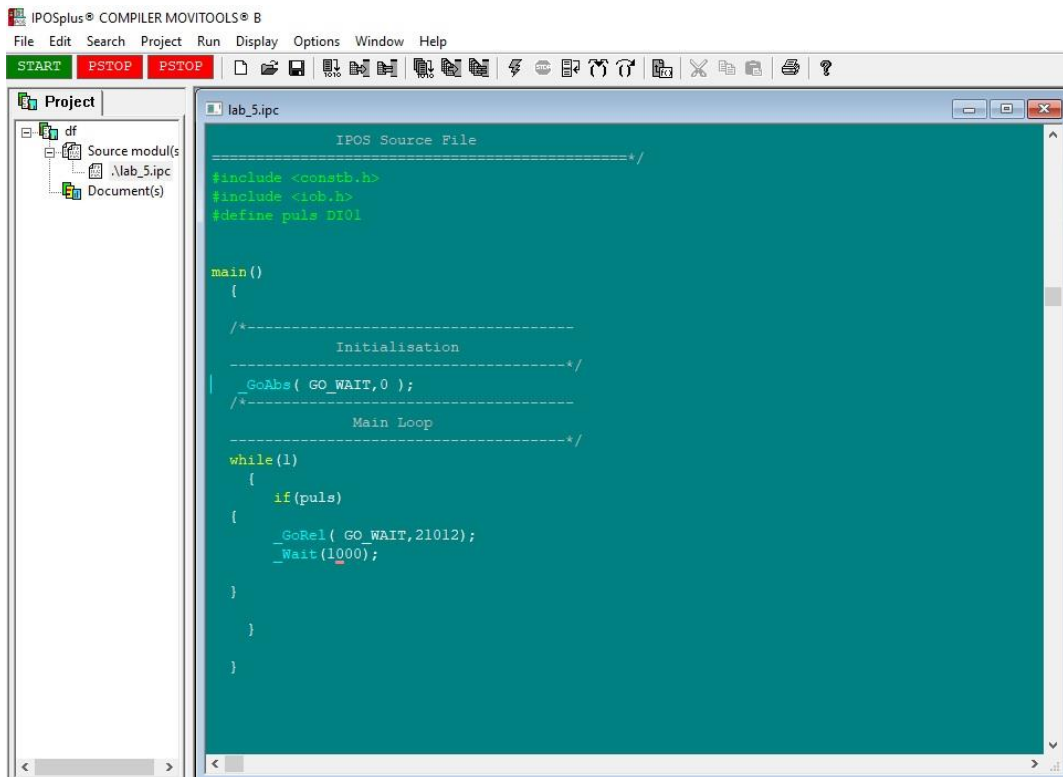
$$(5,134030422 [\text{vueltas}] \times 4096 [\text{representaciones}]) / 1[\text{vuelta}]$$

$$= 21028,98861[\text{representaciones}]$$

Ya listo nuestro ejercicio ocuparemos la cantidad de representaciones para que nuestro motor de las vueltas correspondientes, pero para ello debemos aproximar el resultado el cual sería =21029[representaciones]

5.4 CREACIÓN DEL PROGRAMA

Ya realizado el cálculo matemático procedemos a crear el programa correspondiente para solucionar el problema como se indica a continuación.



```

IPOSplus® COMPILER MOVITOOLS® B
File Edit Search Project Run Display Options Window Help
START PSTOP PSTOP

Project
  df
    Source modul(s)
      \lab_5.ipc
      Document(s)

lab_5.ipc
IPOS Source File
=====*/
#include <constb.h>
#include <lob.h>
#define puls DI01

main()
{
  /*-----
  Initialisation
  -----*/
  GoAbs ( GO_WAIT, 0 );
  /*-----
  Main Loop
  -----*/
  while(1)
  {
    if(puls)
    {
      _GoRel ( GO_WAIT, 21012);
      _Wait (1000);
    }
  }
}

```

Imagen 5, 1 Código del ejercicio.

No siempre la creación del código será la parte más difícil de algún problema como es representado en nuestro ejercicio, a veces se requiere un análisis matemático anteriormente.

LABORATORIO N°6: “ENTRADAS Y SALIDAS DIGITALES”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Encender luces del tablero del servomotor a través de salidas digitales.</i> • <i>Controlar entradas digitales a través de switch.</i> • <i>Sincronizar entradas digitales con salidas digitales.</i> • <i>Comprobar lectura de variables binarias.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>Se controlará el panel completo del servomotor, estudiando entrada y salidas digitales para una completa programación del tablero, corroborando su funcionamiento mediante la lectura de variables.</i></p>		

6.1 DECLARACIÓN DE ENTRADAS Y SALIDAS DIGITALES

Como ya sabemos para definir una variable ocupamos **#define** y para declarar una entrada o salida digital ocupamos la misma función, con la diferencia que no se le asigna un espacio memoria, si no, un bit en específico de un espacio de memoria.

En nuestro caso para definir una entrada digital como lo es un switch, se define de la siguiente manera: “**#define-nombre variable-DI01**”. Donde DI01 corresponde a digital input y su bit asociado.

En el tablero existen 3 switch, donde el primero energiza el servomotor utilizando el bit 0, mientras que el segundo y el tercero están disponibles para el trabajo con sus respectivos bits 1-2.

En el otro caso para definir una salida digital como lo es un LED, se define como: “**#define-nombre variable-DO02**”. Donde DO02 corresponde a digital output y su bit asociado del espacio de memoria H481, ya que en este se encuentra predeterminado toda salida digital.

La maqueta de este servomotor posee una falencia en el orden de sus LEDs, ya que la primera luz de la maqueta está asociada al bit 2, por consiguiente, los bits ocupados por estas son el 2-3-4.

6.2 COMANDOS

En esta ocasión hay que tener en cuenta los siguientes comandos para un mejor desarrollo de la actividad:

- **_BitClear**: pone un bit individual de una variable a 0.
- **_BitMove**: Copia un bit individual de una variable en una posición de bit de una variable diferente.
- **_BitMoveNeg**: Copia el valor negado de un bit en una variable de una posición de bit de una variable diferente.
- **_BitSet**: Ajusta un bit individual de una variable a 1.

(Ver imagen 6.1)

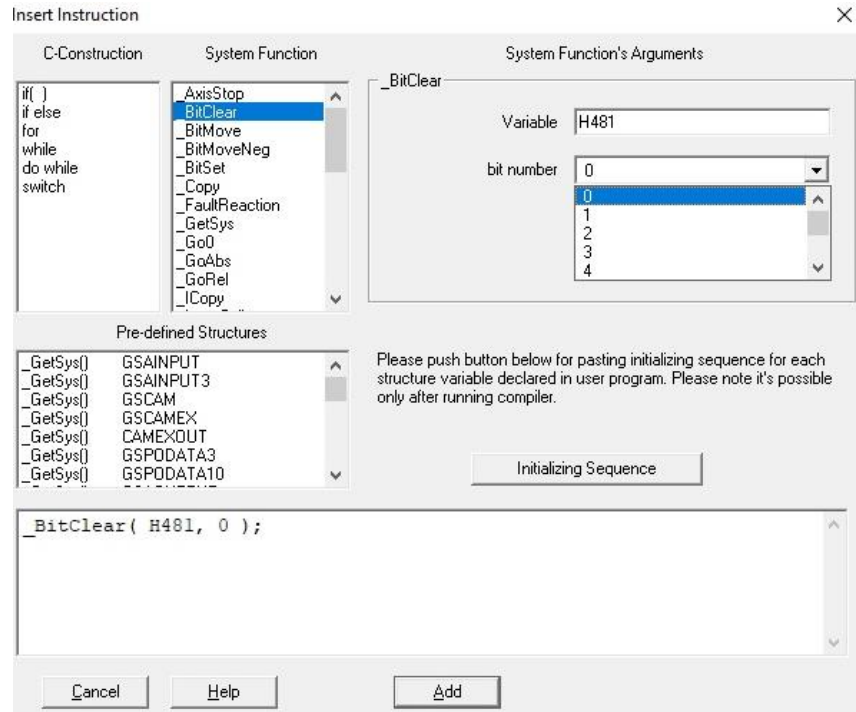


Imagen 6, 1 Comando BitClear.

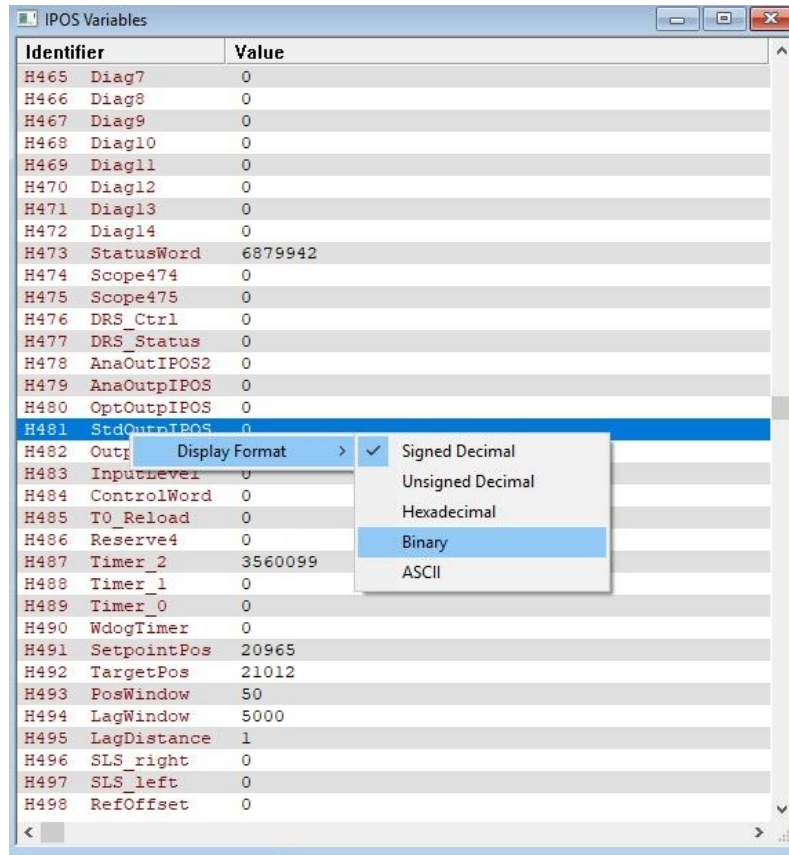
6.3 VISUALIZACIÓN DE VARIABLES

Para visualizar el correcto funcionamiento de una entrada y salida digital, se puede hacer desde la opción “**Allvariable**” mencionada anteriormente e ir al espacio de memoria H481 Y H483, en donde podremos apreciar los bits de entrada y salida respectivamente.

Para divisar los bits asociados se requiere realizar los siguientes pasos:

- 1- Ir a la variable H481 o H483.
- 2- Realizar clic secundario sobre ella.
- 3- Escoger opción “Binary”

(Ver imagen 6.2)



Identifier	Value
H465 Diag7	0
H466 Diag8	0
H467 Diag9	0
H468 Diag10	0
H469 Diag11	0
H470 Diag12	0
H471 Diag13	0
H472 Diag14	0
H473 StatusWord	6879942
H474 Scope474	0
H475 Scope475	0
H476 DRS_Ctrl	0
H477 DRS_Status	0
H478 AnaOutIPOS2	0
H479 AnaOutpIPOS	0
H480 OptOutpIPOS	0
H481 StdOutpIPOS	0
H482 OutpIPOS	0
H483 InputLevel	0
H484 ControlWord	0
H485 T0_Reload	0
H486 Reserve4	0
H487 Timer_2	3560099
H488 Timer_1	0
H489 Timer_0	0
H490 WdogTimer	0
H491 SetpointPos	20965
H492 TargetPos	21012
H493 PosWindow	50
H494 LagWindow	5000
H495 LagDistance	1
H496 SLS_right	0
H497 SLS_left	0
H498 RefOffset	0

Imagen 6, 2 Visualización de las variables.

6.4 EJERCICIO

Como actividad cree un programa donde cada switch encienda y apague una luz correspondientemente teniendo en cuenta que el switch para energizar igual puede ser utilizado, además de que al encender la luz2 el servomotor gire de forma horaria y anti-horario al encender la luz3.

Solución ver imagen 6.3

```
#include <constb.h>
#include <iob.h>
#define led H481
#define Swich_1 DI00
#define Swich_2 DI01
#define Swich_3 DI02

main()
{
    _GoAbs( GO_NOWAIT,0 );
    while(1)
    {
        if(Swich_1){
            _BitSet(led, 2 );
        }
        else{
            _BitClear(led,2);
        }
        if(Swich_2){
            _BitSet(led, 3 );
            _GoRel( GO_NOWAIT,40960 );
            _Wait( 1000 );
        }
        else{
            _BitClear(led,3);
        }
        if(Swich_3){
            _BitSet(led,4);
            _GoRel( GO_NOWAIT,-40960 );
            _Wait( 1000 );
        }
        else{
            _BitClear(led,4);
        }
    }
}
```

Imagen 6, 3 Solución ejercicio.

LABORATORIO N°7: “ LECTURA DEL PUERTO ANÁLOGO ”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Analizar puertos de conexión del servo motor.</i> • <i>Realizar circuito para la conexión del potenciómetro al servomotor.</i> • <i>Crear programa para manejo del servomotor a través del potenciómetro.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>Realizar una conexión entre el potenciómetro y el servomotor y crear un programa en donde el este controle la posición del servo, a través de la lectura del puerto análogo.</i></p>		

7.1 CONEXIONADO DEL POTENCIÓMETRO AL SERVOMOTOR.

Antes de comenzar este procedimiento se requiere de un desatornillador y un potenciómetro de 10 k Ω . (Ver imagen 7.1)

Ya con estos materiales procedemos al conexionado entre potenciómetro y servomotor, sin olvidar que todo debe estar desenergizado, para ello debemos informarnos sobre las entradas que hay en la regleta del servomotor (Ver imagen 7.2)

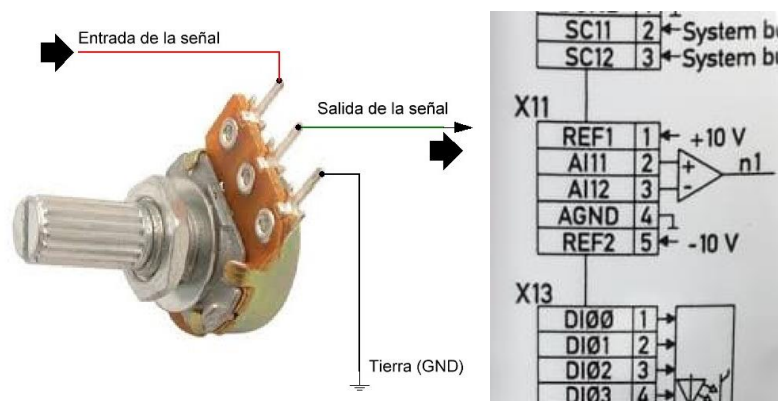


Imagen 7, 1 - Imagen 7, 2 Diagramas de conexión.

El potenciómetro para su conexionado con el servomotor requiere el uso de sus tres pines de conexión, los cuales son señal de entrada, señal de salida y tierra, estos deben ir conectados en la regleta x11 del servomotor, donde encontraremos 2 puertos analógicos disponibles. Nos conectaremos al puerto 1 (REF1, de la cual se obtienen 10 [V]), 2 (AI11, es una entrada de lectura analógica que posee rangos de [-10...0...+10 V] [0...+10 V] [0...+20 mA] [4...+20 mA]) y 4 (AGND, tierra) correspondientemente.

7.2 COMANDOS

GetSys: Carga el valor de un sistema interno, en una o más variables, Esta incluye:

- Corriente activa
- Velocidad actual
- Código de falla
- Status de sistema
- Posición actual
- Status de entradas/salidas
- Status entradas/salidas analógicas

- Datos PO.

Expresiones que puede aceptar el comando:

- GS_ACTCUR: Corriente activa en 0.1% de corriente nominal de la unidad.
- GS_ACTSPEED: Velocidad real en 0.1 rpm.
- GS_SPSPEED: Setpoint en 0.1 rpm.
- GS_ERROR: Código de error según la tabla "Mensajes de error y lista de errores" en el manual del sistema.
- GS_SYSSTATE: valor de la visualización de 7 segmentos de acuerdo con la tabla "Visualización del modo de operación" en el manual del sistema.
- GS_ACTPOS: posición real en función del codificador seleccionado en P941 H509, H510 o H511.
- GS_SPPOS: Setpoint de H491.
- GS_TPOS: posición de destino del generador de perfiles.
- GS_INPUTS: entradas binarias H483 (MOVIDRIVE® A) / H520 (MOVIDRIVE® B) de la unidad básica y opciones.
- GS_SYSSTATE: Idéntico a la palabra de estado 1 del perfil de la unidad de bus de campo (código de falla y estado de funcionamiento).
- GS_OUTPUTS: salidas binarias H482 unidad básica y opciones.
- GS_IxT: Utilización de la unidad en 0.1% nominal de la corriente de la unidad.
- GS_ACTPOS / GS_SPPOS / GS_TPOS: la resolución depende del codificador seleccionado en P941:
 - Codificador del motor: 4096 Inc./revolution
 - Codificador externo X14: resolución del codificador \times P944
 - DIP (codificador SSI): resolución del codificador \times P955
- GS_ANINPUTS: Valor de tensión / valor actual de las entradas analógicas 1 y 2
Entrada de tensión: -10 V ... 0 ... +10 V = -10000 ... 0 ... 10000 representaciones.
 - Entrada de corriente: 0 ... 20 mA = 0 ... 5000/4 ... 20 mA = 1000 ... 5000 representaciones.
 - H + 0 = entrada analógica 1
 - H + 1 = Entrada analógica 2
- GS_CAM: se usa para implementar un controlador de cámara
 - Con el comando GETSYS, se puede usar un controlador de levas estándar con 4 salidas por unidad. Para unidades MOVIDRIVE®, puede usar un controlador de cámara expandido con 8 salidas.
- GS_ANOUTPUTS: salidas analógicas opcionales, con -10 V ... 0 ... +10 V = -10000 ... 0 ... 10000 representaciones.

- h = Salida analógica 1
- h + 1 = Salida analógica 2
- GS_TIMER0: valor del contador de TIMER 0 H489 en ms
- GS_TIMER1: valor del contador de TIMER 1 H488 en ms
- GS_PODATA: Lee el buffer de datos de PO. Independientemente de la cantidad de elementos de datos de PO, se leen 2 elementos de datos PO o 10 elementos de datos PO (datos enviados desde el maestro a la unidad).
- GS_DCVOLT: tensión de enlace de CC [V]
- GS_RELTORQUE: Par relativo. El valor está disponible en los modos de funcionamiento CFC ... y SERVO
- GS_RELTORQUEVFC: El torque relativo es el valor de visualización basado en la corriente nominal de la unidad para el torque en el eje de salida del motor en una corriente nominal de la unidad del 0,1%.
- GS_ACTSPEEDEXT: velocidad real del codificador externo (X14)

GSINPUT: para definir variables análogas.

7.3 PROGRAMACIÓN

Para el siguiente código es necesario declarar el potenciómetro como una variable análoga como se muestra a continuación. (Ver imagen 7.3)

```
#include <constb.h>
#include <iob.h>
GSINPUT pot;
#define factor H7
```

Imagen 7, 3 Declaración de variables.

Como medida de seguridad, llevar el servomotor siempre a su posición inicial. (Ver imagen 7.4)

```

main ()
{
    /*-----
        Initialisation
    -----*/
    _GoAbs ( GO_WAIT, 0 );
    /*-----
        Main Loop
    -----*/
}

```

Imagen 7, 4 Código del programa.

Se lee nuestra entrada analoga en primer lugar, luego para obtener un rango se crea un factor de conversión el cual se obtienen de H420 que es la variable donde se guarda la lectura del potenciómetro multiplicado por 4096 (que es la cantidad de representaciones de una vuelta del servomotor) dividido en 10000 que son las representaciones que se considera como 10 volts de una entrada.

Finalmente con GoAbs logramos que el potenciómetro controle la posición del servomotor. (Ver imagen 7.5)

```

while (1)
{
    _GetSys ( pot, GS_ANINPUTS );
    factor = (H420 * 4096) / 10000;
    _GoAbs ( GO_NOWAIT, factor );
}

```

Imagen 7, 5 Código de lectura y conversión del programa.

LABORATORIO N°8: “COMUNICACIÓN ETHERNET”		
<i>Autores: Jairo Crisóstomo Nicolás Millavil</i>	<i>Profesor guía: Felipe Benavides</i>	<i>Fecha:2018</i>
<p><i>Objetivos:</i></p> <ul style="list-style-type: none"> • <i>Investigar y analizar configuración de IP de los servomotores.</i> • <i>Estudiar plataforma “Address Editor”</i> • <i>Crear una red de comunicación entre dos o más servomotores.</i> 		
<p><i>Descripción del laboratorio:</i></p> <p><i>El laboratorio consistirá en crear una red ethernet entre los servomotores asignando una IP propia a cada uno para obtener un mayor orden y así poder controlarlos desde un mismo PC</i></p>		

8.1 PROTOCOLO CONFIGURACIÓN IP

Para realizar optimamente el conexionado a través de Ethernet, se debe cambiar el switch “DEF IP” al valor 0, ya que esto permitira poder cambiar la IP al driver del servomotor, y la posicion 1 representara la IP definida de fabrica. (ver imagen 8.1)



Imagen 8, 1 Switch “DEF IP” on y off.

Ya cambiado a la posicion 0, realizar la conexión Ethernet normal PC/Servomotor para proceder a cambiar la IP del servomotor, para ello dirigirse al icono de la plataforma movitools y realizar clic secundario en ella . (ver imagen 8.2)

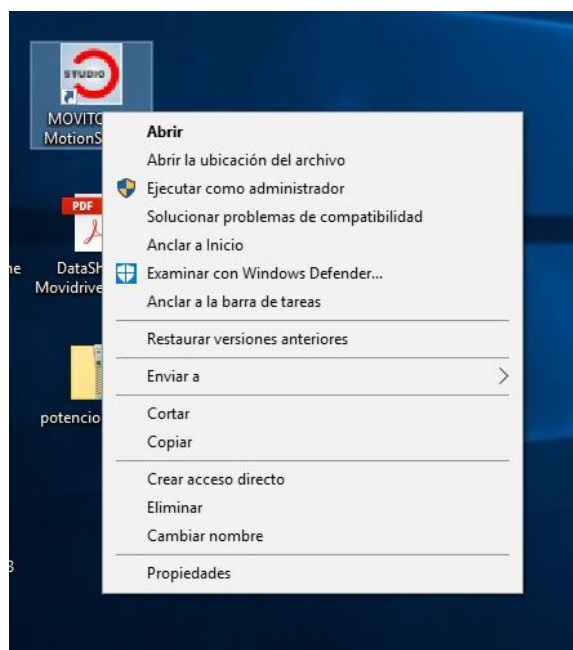


Imagen 8, 2 direccionándose hacia la carpeta de movitools.

Ir a la opción propiedades y seleccionar “abrir ubicación”. (Ver imagen 8.3)

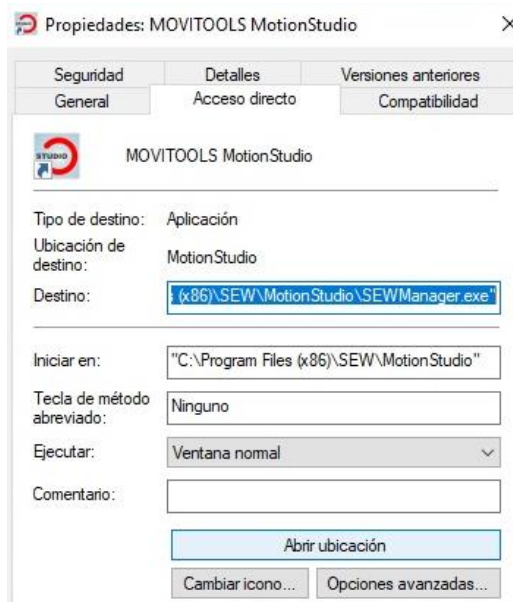


Imagen 8, 3 abriendo ubicación de Addr Tool.

Ya en la carpeta del programa, buscar y abrir la aplicación “addrtool”. (Ver imagen 8.4)

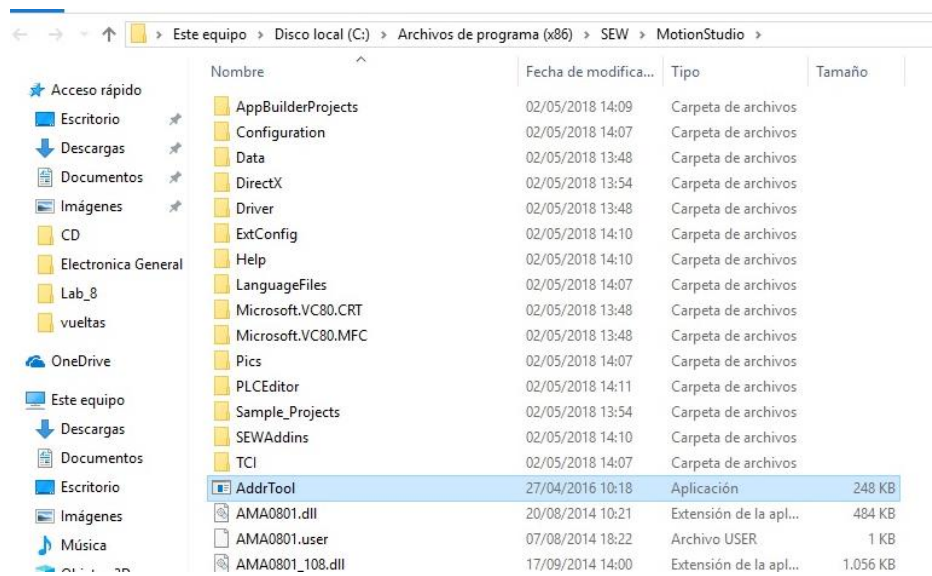


Imagen 8, 4 Programa Addr Tool.

Ya en la interfaz de la aplicación, escogemos nuestro tipo de comunicación y seleccionamos “next”. (ver imagen 8.5)

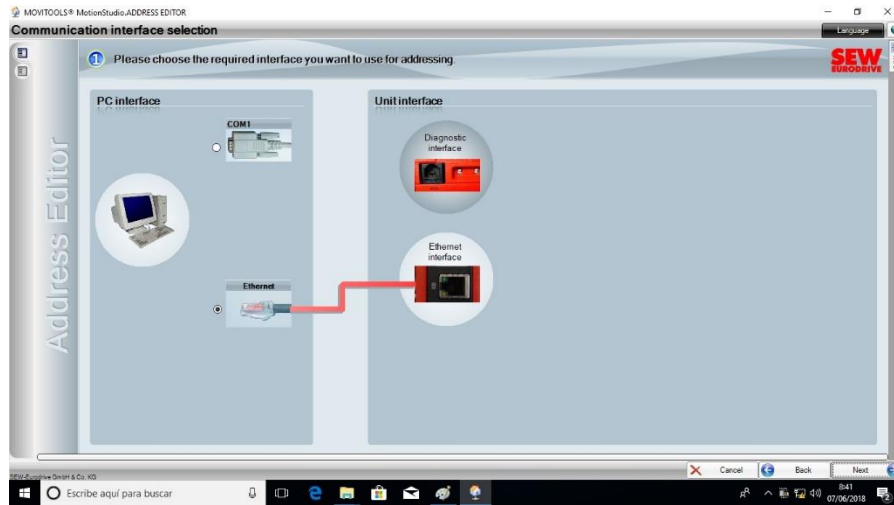


Imagen 8, 5 Configuración IP a través de Addr Tool.

Se escaneara nuestro servomotor y apareciera con sus respectivas configuraciones de fabricas las cuales pueden ser modificadas. (ver imagen 8.6)

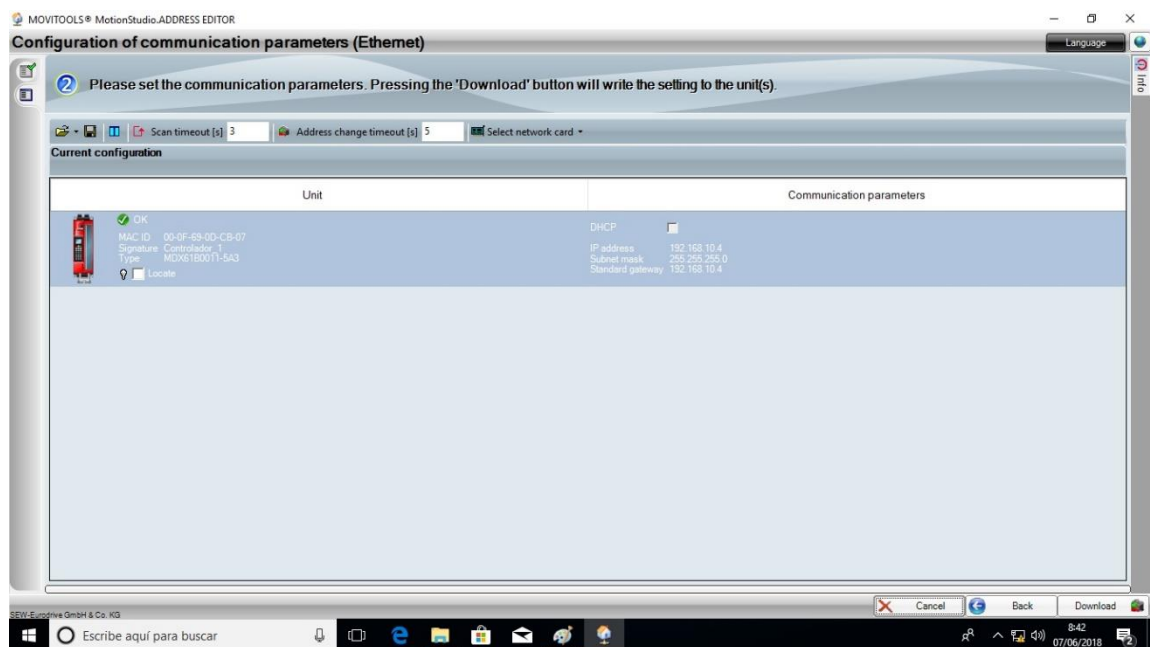


Imagen 8, 6 Configuración IP a través de Addr Tool.

Ir a comunicación parameters y podra escoger la IP que uno dese al servomotor, estableciendo un orden en nuestra red, teniendo en cuenta que cada vez que se cambien parametros presionar “Enter”. (Ver imagen 8.7)

Communication parameters

IP address	192.168.10.4
Subnet mask	255.255.255.0
Standard gateway	192.168.10.4

Imagen 8, 7 Configuración IP a través de Addr Tool

Ya asignada la IP, para guardar nuestros parametros de forma definitiva hay que seleccionar “Download”, y reiniciar¹ el servomotor (Ver imagen 8.8)

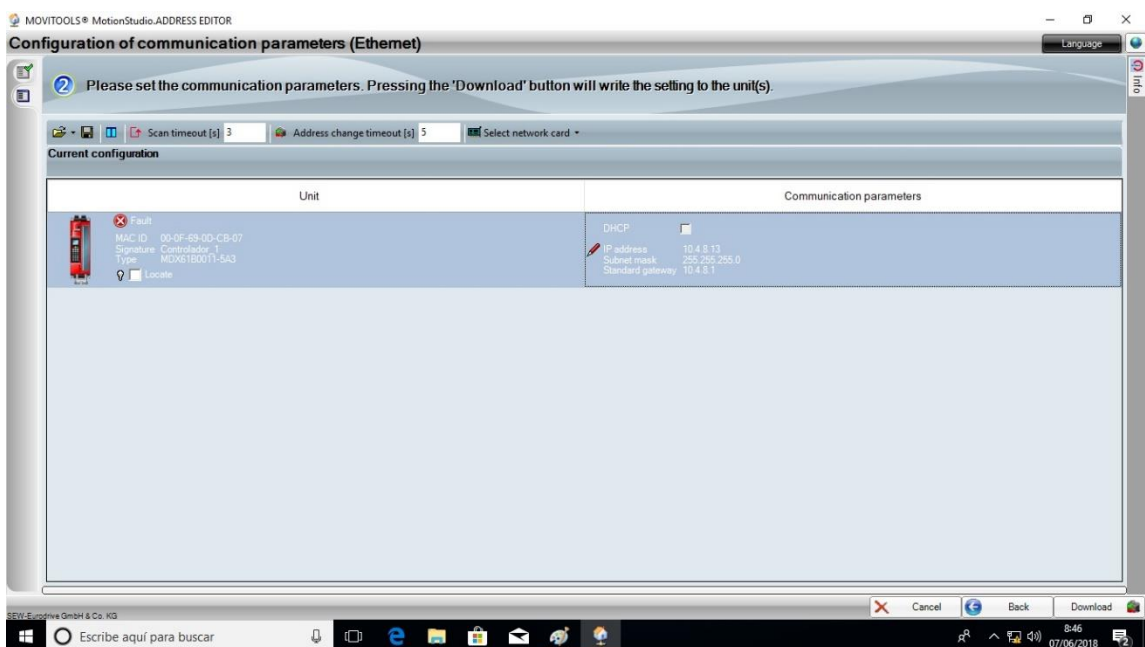


Imagen 8, 8 Configuración IP a través de Addr Tool

Luego procedemos a conectar el pc y el servomotor a la red mediante cables ethernet. Para comprobar el cambio de IP entrar al programa movitools y escanear. (Ver imagen 8.9)



Imagen 8, 9 Verificación IP

¹ Bajar y subir automático trifásico

Repetir el procedimiento con los demás servomotores formando así una red, la cual permitirá controlar cada servomotor con solo un PC, como indica el siguiente diagrama: (Ver imagen 8.10)

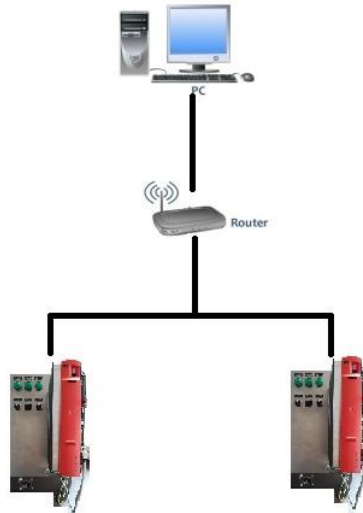


Imagen 8, 10 Diagrama de comunicación.

Para realizar comunicación Servomotor-PLC a través de Ethernet el tipo de PLC debe ser Allen Bradley ya que este posee un protocolo de Ethernet/IP, por otro lado, el PLC de SIEMENS no trabaja con este tipo de protocolo.

Para otras conexiones tipo Ethernet está la opción de cambiar la placa de comunicación por la DFE32B que posee el protocolo Profinet.

En el caso que se comunique PLC-Servomotor, se debe tener en cuenta el descargar las extensiones **GSDML** para Profinet, **GSD** para ProfiBus y **EDS** para DeviceNet, y ser incorporadas en el PLC.

Si se busca otro tipo de conexión aparte de Ethernet, existe la manera de comunicarse por RS485 o SBus Siempre y cuando se respete su protocolo de comunicación.

