

CONTROL SECUENCIAL DE UN PROCESO SIMULADO DE 5 CILINDROS HIDRÁULICOS MEDIANTE UN PLC.

Autores: David Steven Galvis Arévalo 1802584, Johan Andres Sebastian Gómez Espinosa 1802322,
Eric Xavier Buitrago Forero 1802410.

Resumen: En este informe se presenta el desarrollo de un proceso secuencial implementado en un PLC en el cual se recrean curvas de fase realizadas por actuadores (cilindros) que contengan características físicas reales y que sean simulados en el software Automation Studio. Mediante el uso de temporizadores y contadores se cumplen los perfiles asociados a cada uno de los cilindros, los temporizadores permiten que se cumpla el tiempo entre segmento por segmento de cada perfil, los cambios de posición del mismo y los contadores permiten que estos ciclos se puedan repetir la cantidad de veces que se indique.

Finalmente al realizarse este procedimiento se obtienen las curvas de fase en gráficas para cada uno de los actuadores al tiempo que el usuario desee, se trabaja en simultáneo como un proceso de aplicación industrial en el cual se encuentren sincronizados distintos actuadores y esto gracias al controlador lógico programable.

Abstract: *In this paper is presented the development of a sequential process programmed in PLC which recreates given phase curves made by actuators (cylinders) and with physical features simulated in Automation Studio. Through use of timers and counters the cylinder's profiles can be done. The timers allow the user to have the right time between segments of each profile and changes of position of itself, and the counters allow the process to repeat the cycles that the user indicates.*

Finally when the process is done there are presented the phase curves graphs for the 5 actuators. It works simultaneously as an industrial process where actuators are synchronized and this by using the programmable logic controller.

INGENIERÍA MECATRÓNICA UMNG

Introducción:

Los procesos secuenciales proponen estados (operaciones a realizar para la transformación de la materia prima en producto) y transiciones (información relativa a sensores o elementos lógicos como temporizadores o contadores) en una secuencia ordenada que identifica la evolución dinámica de un proceso controlado. [1]

Los contadores se usan cuando es necesario contabilizar las veces que se acciona un contacto. Entre las funciones de los PLC están las de conteo y disponen de los circuitos adecuados para tal fin. Existen contadores progresivos y regresivos, significa que cuenta ascendente o descendientemente y cuando alcanzan el valor deseado, el contacto del contador cambia de estado.

Los temporizadores permiten la conexión o desconexión de un contacto luego de un determinado tiempo, cuando se quieren lograr tiempos de retrasos mayores a los que se pueden obtener con un solo temporizador, se conectan entre sí varios temporizadores, lo cual se conoce como conexión en cascada. [2]

Un controlador lógico programable es un dispositivo electrónico digital que usa una memoria programable para guardar instrucciones y llevar a cabo funciones lógicas, de configuración de secuencia, de sincronización, de conteo y aritméticas, para el control de maquinaria y procesos. [3]

Objetivos

- General

*Diseñar una solución de software para un proceso secuencial utilizando el PLC.

- Específicos

*Identificar los actuadores necesarios acorde al perfil que cada uno debe realizar.

*Emplear un programa en el PLC que permita desarrollar un proceso secuencial utilizando temporizadores y contadores.

*Desarrollar un entorno en Automation Studio que permite simular el proceso previamente establecido.

Desarrollo

Teniendo los perfiles y la carga para cada actuador, se debe buscar entonces en catálogos comerciales los cilindros que cumplan las especificaciones como velocidad, carga, fuerza de empuje y retroceso.

Materiales:

- PLC Siemens S7-1500
- Realtek RTL8188EU Wireless LAN 802.11n USB 2.0 Network Adapter.
- TIA Portal V13
- OPC IBH
- Automation Studio 6.1

Procedimiento

Se identifican las curvas de fase a recrear para el grupo de trabajo correspondiente

GRUPO 2.

Grupo	Secuencia	Perfiles de Carga				
		A	B	C	D	E
2	D(XE(XB(XA(XC))))	1	5	2	3	4

Se realiza la secuencia enseñada anteriormente D(xE(xB(xD(xC)))) donde < x > es la cantidad de repeticiones solicitadas por el usuario.

INGENIERÍA MECATRÓNICA UMNG

Donde las curvas de fase de los perfiles a realizar se encuentran en su respectivo orden de la figura 1 a 5.



Figura 1: Perfil 3, asignado a la etapa D

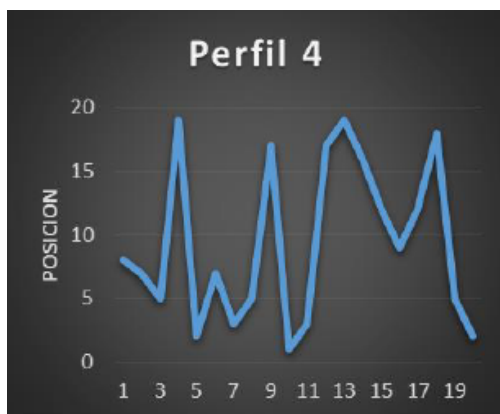


Figura 2: Perfil 4, asignado a la etapa E

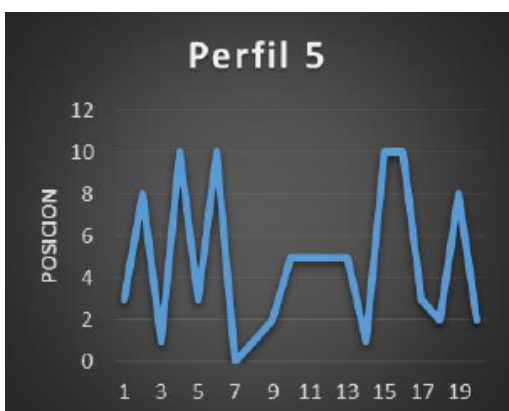


Figura 3: Perfil 5, asignado a la etapa B

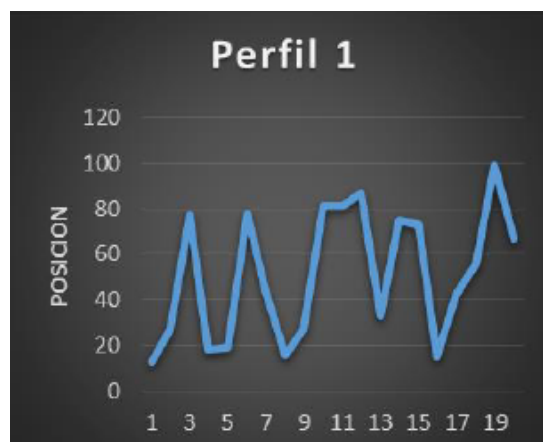


Figura 4: Perfil 1, asignado a la etapa A

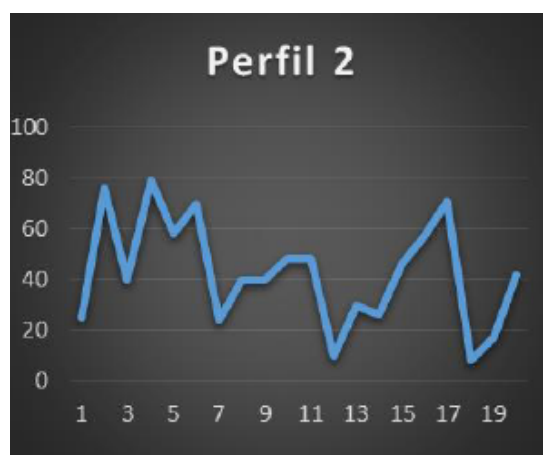


Figura 5: Perfil 2, asignado a la etapa C

Se debe tener en cuenta la carga para cada uno de los cilindros, en base a la tabla x.

A	B	C	D	E
14,4Tn	18,6Tn	8Tn	3,45Tn	12Tn

Tabla 1: Carga para los actuadores

Se muestran entonces los cilindros capaces de desarrollar el proceso y con sus características más relevantes.



Figura 6: Cilindros hidráulicos de doble efecto LARZEP

Para todos los cilindros, primero la característica principal que se identificó fue la carga que este va a soportar, las fuerzas de empuje y de retroceso y las velocidades máxima y mínima que puede alcanzar.[4]

Cil	Capacidad (Tn)	Carrera (mm)	Modelo	Capacidad (kN) Push - Pull	Volumen (cc)	Peso (kg)
D	20	160	D02016	214 - 53	499	11,7
E	100	160	D10016	911 - 189	2123	51,8
B	150	50	D15005	1380 - 370	1005	66
A	150	150	D15015	1380 - 370	3015	90
C	350	150	D35015	3368 - 1014	7359	247

Tabla 2: Referencia y características de los actuadores

Al momento de calcular la fuerza necesaria fue necesario calcular las velocidades maximas y minimas, para esto se usó la siguiente tabla

Velocidad del émbolo (mm/s)	Factor de carga máximo
8 a 100	70%
101 a 200	30%
201 a 300	10%

Tabla 3: Factor de carga en función de la velocidad.

De acuerdo a la x establecida se halló el factor de carga y se encontró el cilindro que contenía las características necesarias.

Programa TIA Portal

En esta parte del main inicializamos todos los bloques, que necesitamos que se cumplan cíclicamente puesto que el main es un ciclo repetitivo.

*Main

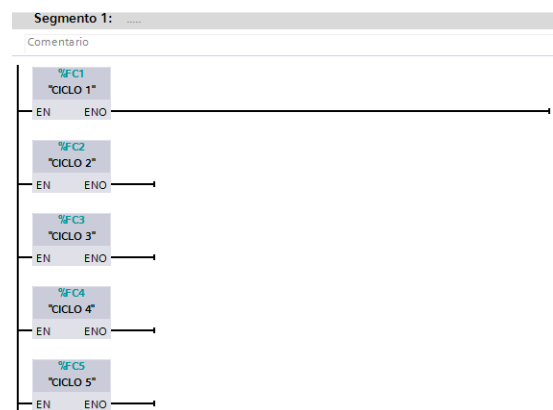


Figura 7: Se inicializan todos los ciclos al mismo tiempo.

En el segmento 1 del main inicializamos todas las partes del ciclo siendo el ciclo 1 la envolvente y el ciclo 5 el que más se repite.

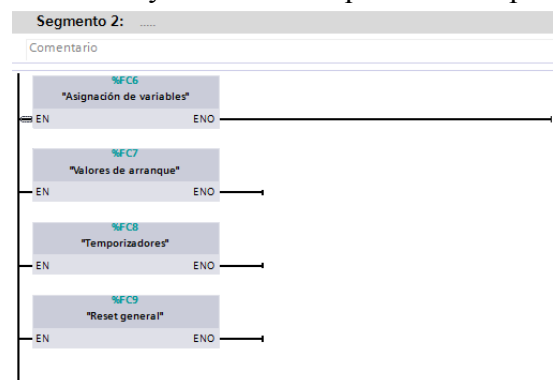


Figura 8: Se inicializan bloques de temporizadores, contadores y asignación de variables.

INGENIERÍA MECATRÓNICA UMNG

En el segmento 2 del main inicializamos las funciones como asignación de variables, valores de arranque, temporizadores, Reset general.

¿Cómo funciona cada bloque?

Asignación de variables: se hace la igualación de el bloque de variables del plc con las variables de bloque de datos con la función de que si hay algún cambio en el programa no se tenga que hacer un cambio completo en el programa si no solo con el renombramiento de una variable.

Variables de arranque: es un bloque en lenguaje scl y lo usamos para dar los valores de arranque a las estranguladoras y a los tiempos de cada ciclo.

Temporizadores: ponemos todos los contadores para cada ciclo ya que cambia su tiempo por lo tanto tienden a hacer diferentes los tiempos.

Reset general: cuando cumple su programa resetea contadores para cada parte del ciclo y vuelve a 0 para volver arrancar su ciclo completo.

*Bloque: Función de asignación de variables:

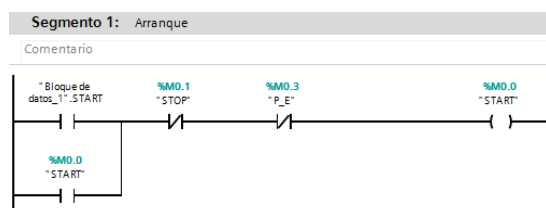


Figura 9: Inicio del programa.

En el segmento 1 del bloque de función de asignación de variables es el arranque del sistema donde igualamos el valor del

automation a la memoria de start del plc tiene una retención en su funcionamiento y lo único que hace que pare es que se de stop.

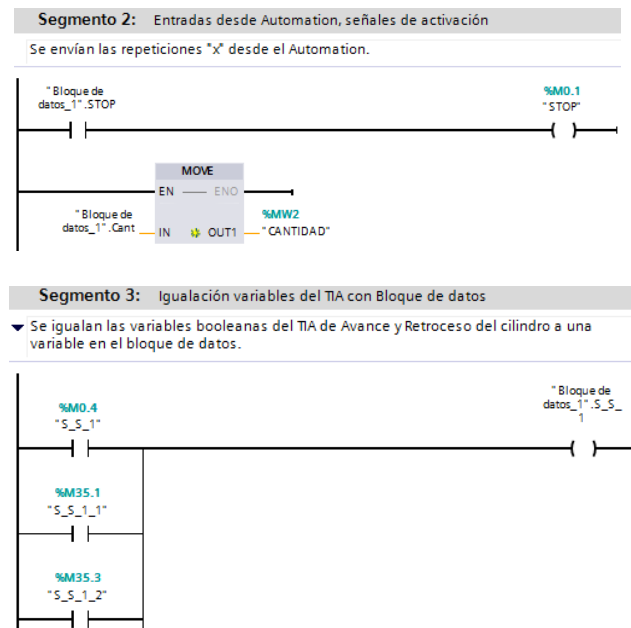


Figura 10: Igualación de variables del sistema con variables de bloque de datos.

En el segmento 2 vemos la igualación del stop del automation con el de stop del plc y también vemos la igualación de la cantidad de repeticiones por ciclo que lo traemos a una memoria del plc.

En estos primeros segmentos traemos valores del automation al plc los cuales son los que nos interesan traer en los siguientes segmentos igualamos cada sentido de cada ciclo de cada vástago y el valor de cada estrangulación.

En el segmento 3 nos damos cuenta que el sentido que mandamos al automation, de cada parte de cada ciclo del perfil depende de la unidad de tiempo que maneje cada uno entonces manejamos varios auxiliares que nos mandan la señal al bloque de datos puesto que no podemos

INGENIERÍA MECATRÓNICA UMNG

llamar la misma memoria en varias partes de una función para mandar el mismo sentido varias veces.

*Bloque: Reset general.

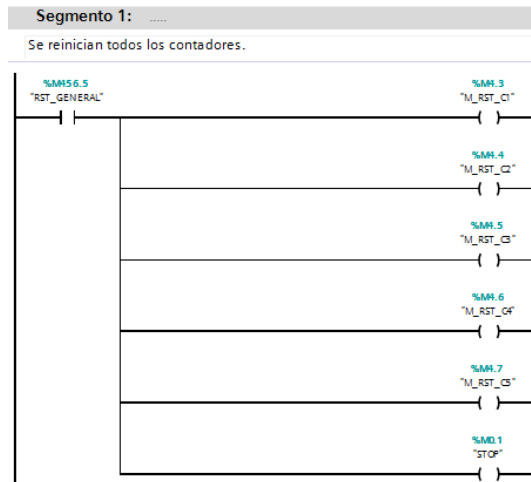


Figura 11: Reinicio de todos los contadores.

En el reset general damos que cuando cumpla la envoltente su ciclo significa que ya cumplió todo el proceso completo lo cual activa una memoria que se llama "RST_GENERAL" que en esta función resetea las memorias de los contadores de cada ciclo y prende el stop que para todo el proceso esperando que le den una nueva señal de arranque.

*Bloque: Temporizadores.

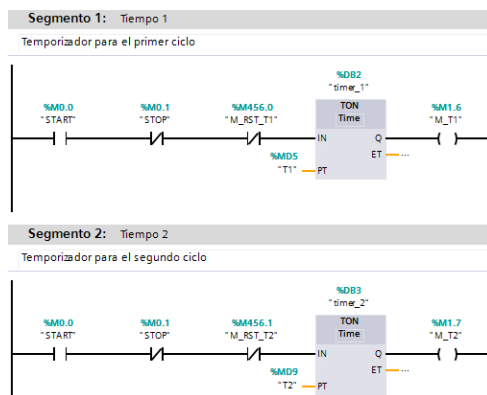


Figura 12: Temporizadores para los ciclos 1 y 2

Para mantener la señal de sentido y el valor de la estrangulación tenemos un bloque donde tenemos un temporizador que nos cuenta cada parte de cada ciclo

Se realiza el mismo procedimiento para los 3 ciclos restantes y así declarar los tiempos para los 5 ciclos.

*Bloque: Valores de arranque.

```
1 //Valores de tiempos
2 //tiempo perfil mas bajo hasta la embolvente
3 // 15 segundos es el cambio de posición del vástago para el primer cilindro
4 "T1" := 15000;
5 "T2" := "T1" / "CANTIDAD";
6 "T3" := "T2" / "CANTIDAD";
7 "T4" := "T3" / "CANTIDAD";
8 "T5" := "T4" / "CANTIDAD";
9 // Valores de apertura de la valvula de estrangulación ciclo 1 envoltente
10 "EST_V_S_1_1" := (0.0067);
11 "EST_V_S_1_2" := (0.0035);
12 "EST_V_S_1_3" := (0.0213966666666667);
13 "EST_V_S_1_4" := (0.0092633333333333);
14 "EST_V_S_1_5" := (0.0227);
```

Figura 13: Valores de estrangulación.

En la figura x se muestran los valores de arranque con los que inicia el programa, este contiene variables las cuales son el porcentaje de estrangulación de las válvulas para cada una de las etapas de cada ciclo, estos datos son enviados a través de un bloque <MOVE> igualando a la variable del bloque de datos declarada en Automation.

*Bloque de datos.

1	Static		
2	START	Bool	false
3	STOP	Bool	false
4	Cant	Int	1
5	EST_S_1	Real	0.0
6	EST_S_2	Real	0.0
7	EST_S_3	Real	0.0
8	EST_S_4	Real	0.0
9	EST_S_5	Real	0.0
10	S_S_1	Bool	false
11	S_B_1	Bool	false
12	S_S_2	Bool	false
13	S_B_2	Bool	false
14	S_S_3	Bool	false
15	S_B_3	Bool	false
16	S_S_4	Bool	false
17	S_B_4	Bool	false
18	S_S_5	Bool	false
19	S_B_5	Bool	false

Figura 14: Variables del bloque de datos

INGENIERÍA MECATRÓNICA UMNG

Estas variables también se declaran en el Automation utilizando el OPC Client, con el fin de poder simular el proceso, <Start> es el inicio de los ciclos, los cilindros se activan realizando los perfiles correspondientes. El botón <Stop> tiene la función de pausar los contadores y por ende todo el proceso. <Cant> Se encarga de enviar desde el Automation la cantidad de repeticiones que el usuario desee, es la variables x de la secuencia. Las variables declaradas <EST_S_#> son el valor de estrangulación que necesita cada válvula para cumplir una determinada velocidad, se crean 5 variables para los 5 cilindros, y estas varía su valor conforme el paso del tiempo. Y finalmente las variables <S_S_#> y <S_B_#> son de tipo booleano y principalmente activan los solenoides para que cada cilindro avance o retroceda, esto gracias al uso de válvulas 4/3 implementadas en Automation Studio.

*Bloque: CICLO 1 y 2 ...5

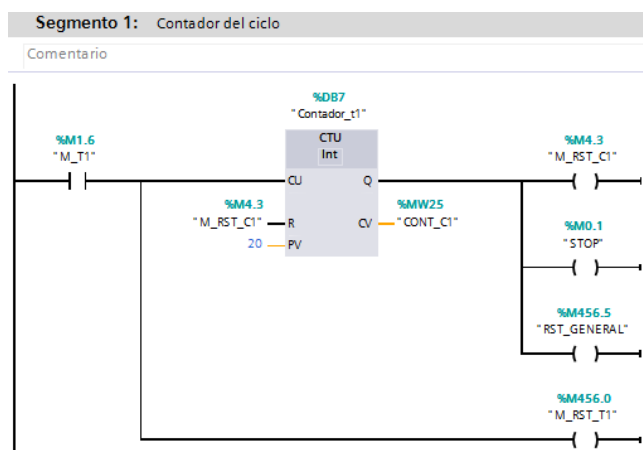


Figura 15: Contador del ciclo

Cada ciclo tiene un contador que contiene los 20 segmentos de cada perfil, y este se activa cuando el timer del ciclo también lo hace.

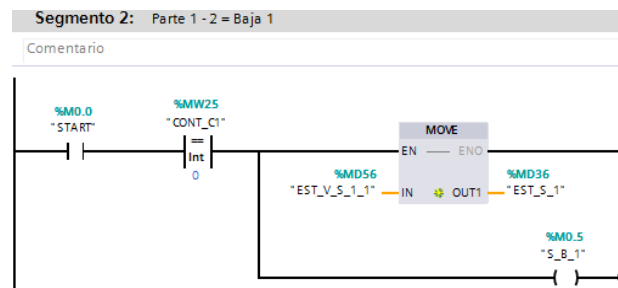


Figura 16: Envío de datos y comparación

En la figura x se tiene al inicio el ciclo el start general, lo que hace que todos los perfiles se realicen de manera simultánea, el contacto posterior a este es la comparación entre el valor del contador y el segmento en el que se encuentra. Que al activarse permite que el bloque “MOVE” envíe el porcentaje de estrangulación de la válvula al Automation a través del OPC.

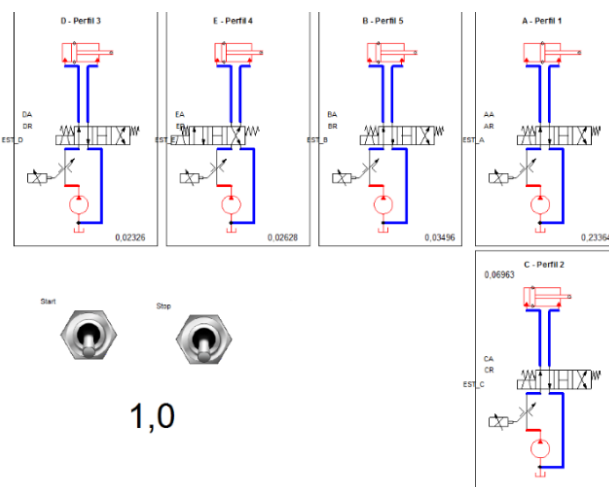


Figura 17: Diagrama del proceso en Automation Studio

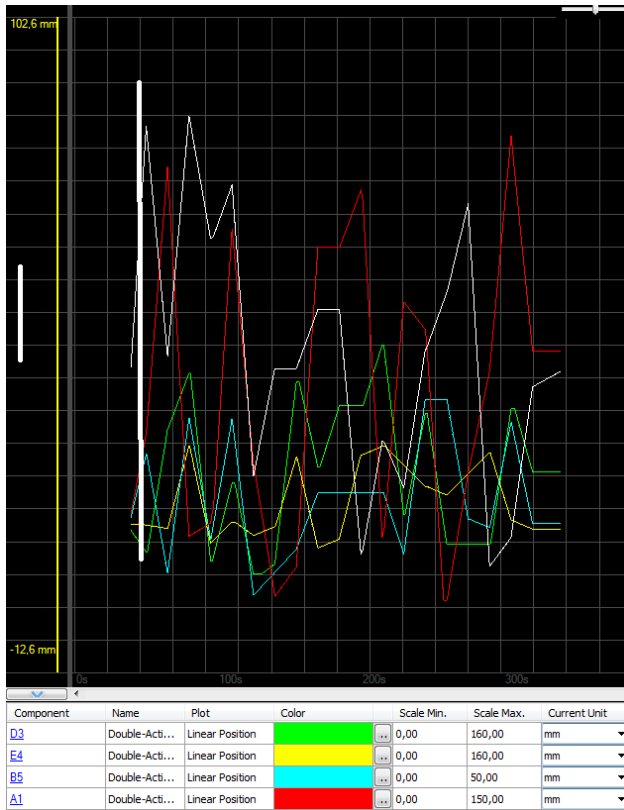


Figura 18: Curvas de fase de los cilindros

Conclusiones:

*Después de haber encontrado el cilindro correspondiente se llegó a la conclusión de que es uno de los múltiples que se tiene que tener en cuenta para una simulación exitosa ya que también hay que saber el circuito que se realizará y sus demás componentes como son

las válvulas direccionales, las estranguladoras y el compresor.

*Al momento de realizar el programa se observó que no se puede definir un variable en varios segmentos porque se genera un conflicto y la variable siempre se define en 0, sin embargo se logra el resultado deseado con la superposición de variables.

*Para desarrollar una simulación bastante cercana a la realidad es necesario tener en cuenta cada valor de cada componente.

Referencias:

[1]P. Ponsa, A. Granollers, Diseño y automatización industrial, Universitat Politècnica de Catalunya, Material y lecturas de diseño industrial, pp 4.

[2]W. Bolton, Mecatrónica: sistemas de control electrónico en ingeniería mecánica y eléctrica, Editorial Alfaomega, 2 ed, pp 436 - 439.

[3]W. Bolton, Mecatrónica: sistemas de control electrónico en ingeniería mecánica y eléctrica, Editorial Alfaomega, 2 ed, pp 423 - 425.

[4] Cilindros de doble efecto LARZEP:

<https://www.larzep.com/es/productos/cilindros-hidraulicos/d-doble-efecto>

