



MAXinBOX 66

Actuador multifunción con 6 entradas y 6 salidas

ZN1IO-MB66

Versión del programa de aplicación: 1.1
Edición del manual: [1.1]_a

www.zennio.com

CONTENIDO

Contenido	2
1 Introducción	4
1.1 MAXinBOX 66	4
1.2 Instalación	5
1.3 Inicialización y fallo de tensión	6
2 Configuración	7
2.1 General	7
Parametrización ETS	7
2.2 Entradas	8
Parametrización ETS	8
2.2.1 Entrada binaria	9
2.2.1.1 Pulsador	9
2.2.1.2 Interruptor/sensor	9
2.2.1.3 bloqueo de una entrada binaria	11
2.2.1.4 Estados iniciales	11
Parametrización ETS	12
2.2.2 Sonda de temperatura	18
Parametrización ETS	18
2.2.3 Detector de movimiento	21
2.3 Salidas	22
Parametrización ETS	22
2.3.1 Salidas individuales	23
Parametrización ETS	26
2.3.2 Canales de persiana	33
Parametrización ETS	36
2.3.3 Control manual	44
Parametrización ETS	46
2.4 Funciones lógicas	48

2.5	Termostatos	49
2.6	Control maestro de iluminación.....	50
	Parametrización ETS.....	50
ANEXO I.	Objetos de comunicación	53

1 INTRODUCCIÓN

1.1 MAXINBOX 66

El MAXinBOX 66 de Zennio es un actuador KNX versátil y con una amplia variedad de funciones:

- **6 salidas de relé**, configurables como:
 - Hasta 3 canales de persiana independientes (con o sin lamas),
 - Hasta 6 salidas ON/OFF individuales independientes,
 - Una combinación de las anteriores.
- **6 puertos de entrada multipropósito**, configurables como:
 - Sondas de temperatura,
 - Entradas binarias (es decir, pulsadores, interruptores, sensores),
 - Detectores de movimiento.
- **10 funciones lógicas multioperación personalizables.**
- **4 termostatos independientes.**
- **Control maestro de iluminación** para un control sencillo e inmediato de un conjunto de luminarias (o dispositivos funcionalmente equivalentes), una de las cuales actúa como luz principal y las otras como secundarias.
- **Control / supervisión manual** de las 6 salidas de relé a través de los pulsadores y LEDs incorporados.

1.2 INSTALACIÓN

El MAXinBOX 66 se conecta al bus KNX mediante el conector KNX incorporado.

Una vez que el dispositivo se alimenta con tensión a través del bus, se podrá descargar tanto la dirección física como el programa de aplicación asociado.

Este actuador no necesita fuente de alimentación externa, pues se alimenta enteramente a través del bus KNX.

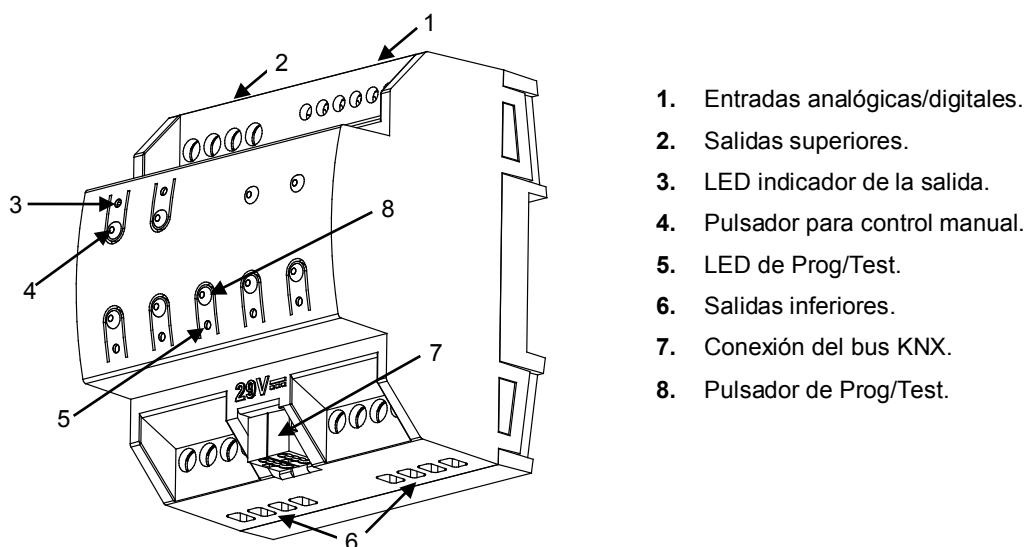


Figura 1. MAXinBOX 66. Elementos.

A continuación se describen los elementos principales del actuador:

- **Pulsador de Prog/Test (8):** una pulsación corta sobre este botón sitúa al dispositivo en modo programación. El LED asociado (5) se ilumina en rojo.

Nota: si este botón se mantiene pulsado en el momento en que se aplica la tensión de bus, el dispositivo entra en **modo seguro**. El LED reacciona parpadeando en rojo cada 0,5 segundos.

- **Salidas (2 y 6):** puertos de salida para la inserción de los cables (pelados) de los sistemas controlados por el actuador (ver sección 2.3). Deberá asegurarse la conexión por medio de los tornillos incorporados en la placa.
- **Entradas (1):** puertos de entrada para la inserción de los cables de accesorios externos como interruptores / detectores de movimiento / sondas de temperatura, etc. Uno de los cables de cada accesorio debe conectarse a

alguna de las entradas marcadas como “1” a “6”, mientras que el otro cable debe conectarse a la entrada etiquetada como “C”. Téngase en cuenta que todos los accesorios externos comparten la entrada “C” para uno de los dos cables. Deberá asegurarse la conexión a través de los tornillos incorporados.

Para obtener información más detallada de las características técnicas del dispositivo, así como información de seguridad y sobre su instalación, consúltese la **hoja técnica** incluida en el embalaje original del dispositivo, y que también se encuentra disponible en la página web: <http://www.zennio.com>.

1.3 INICIALIZACIÓN Y FALLO DE TENSION

Durante la inicialización del dispositivo, el LED de Prog/Test, parpadea en azul unos segundos antes de que el MAXinBOX 66 esté listo. Las órdenes externas no se ejecutarán durante este tiempo, pero sí después.

Dependiendo de la configuración, se ejecutarán además algunas acciones específicas durante la puesta en marcha. Por ejemplo, el integrador puede configurar si los canales de salida deben conmutar a un estado en particular y si el dispositivo debe enviar ciertos objetos al bus después de recuperar la tensión. Por favor, consulte las siguientes secciones de este documento para obtener más detalles.

Por otro lado, cuando se produce un fallo de tensión, el MAXinBOX 66 interrumpirá cualquier acción pendiente, y guardará su estado de forma que lo pueda recuperar una vez se restablezca el suministro de energía.

Por razones de seguridad, se detendrán todos los **canales de persiana** (es decir, se abrirán los relés) si se produce un fallo de tensión, mientras que las salidas individuales conmutarán al estado específico configurado en ETS (si se ha configurado alguno).

2 CONFIGURACIÓN

2.1 GENERAL

Después de importar la base de datos correspondiente en ETS y añadir el dispositivo a la topología del proyecto deseado, el proceso de configuración se inicia haciendo clic derecho en el dispositivo y seleccionando *Editar Parametros*.

PARAMETRIZACIÓN ETS

La única pantalla parametrizable disponible por defecto es General. Desde esta pantalla se pueden activar/desactivar todas las funciones necesarias.

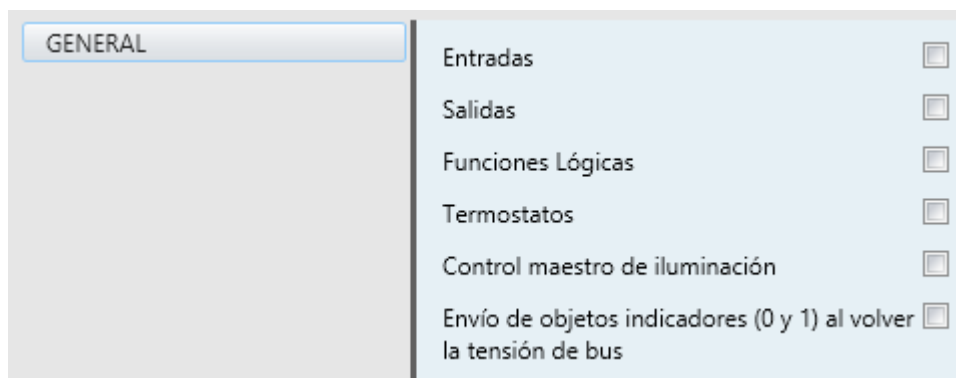


Figura 2. Pantalla General.

- Una vez activadas las funciones de **Entradas**, **Salidas**, **Funciones lógicas**, **Termostatos** y **Control maestro de iluminación**, se incluirán pestañas adicionales en el menú de la izquierda. Estas funciones y sus parámetros se explicarán en secciones posteriores de este documento.
- **Envío de objetos indicadores (0 y 1) al volver la tensión de bus**: este parámetro permite al integrador activar dos nuevos objetos de comunicación ("**Reset 0**" y "**Reset 1**"), que se enviarán al bus KNX con valores "0" y "1" respectivamente cada vez que el dispositivo comience a funcionar (por ejemplo, después de un fallo de tensión). Es posible parametrizar un cierto **retardo** para este envío (0 a 255 segundos).

2.2 ENTRADAS

El MAXinBOX 66 incorpora **6 puertos de entrada analógico-digitales**, cada uno de los cuales configurable como:

- **Entrada binaria**, para la conexión de un pulsador o interruptor/sensor.
- **Sonda de temperatura**, para conectar un sensor de temperatura (modelo ZN1AC-NTC68 S/E/F de Zennio).
- **Detector de movimiento**, para conectar un detector de movimiento (modelos ZN1IO-DETEC-P y ZN1IO-DETEC-X de Zennio).

Importante: los modelos antiguos del detector de movimiento Zennio (ej.: ZN1IO-DETEC y ZN1IO-DETEC-N) no funcionarán correctamente con el MAXinBOX 66.

PARAMETRIZACIÓN ETS

Tras activar **Entradas** en la pantalla de parámetros General, las siguientes listas desplegables estarán disponibles para la selección de las funciones específicas requeridas.

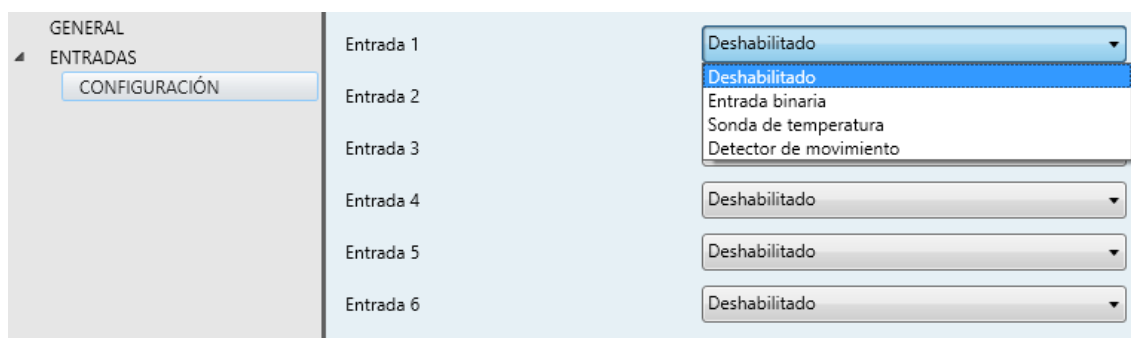


Figura 3. Entradas - Configuración.

Todas las entradas están inhabilitadas por defecto. Dependiendo de la función seleccionada para cada entrada, se incluirán pestañas adicionales en el menú de la izquierda.

2.2.1 ENTRADA BINARIA

Las entradas binarias llevan a cabo las siguientes tareas:

- Recuperar el **estado** (1/0) de la línea de entrada y detectar cambios (por ejemplo, pulsación de un botón, cambios de los sensores, etc.).
- **Informar** al bus KNX sobre los estados/cambios anteriores y activar las correspondientes **acciones**, según el caso.
- Detectar **sabotajes** (es decir, niveles inesperados de voltaje en la línea) en las entradas configuradas como interruptor/sensor.

Cada entrada binaria debe configurarse como uno de estos dos tipos: **pulsador** o **interruptor/sensor**.

2.2.1.1 PULSADOR

Las acciones que tendrán lugar ante una pulsación corta y ante una pulsación larga (e incluso al soltar el pulsador) son independientes y parametrizables. Es posible incluso configurar **cuánto tiempo** debe durar una pulsación para ser considerada como larga.

Estas acciones pueden consistir en **enviar al bus KNX** un valor binario (0, 1 o alternados), una orden de control de persianas, una orden de control de regulación, una orden de grabar/ejecutar una escena o un valor numérico constante (entero de un byte, valor de porcentaje, entero de dos bytes o coma flotante de dos bytes).

En algunos casos es posible configurar un cierto **retardo** antes de enviar el valor al bus. En el caso de valores binarios, se puede configurar también un reenvío periódico, que puede ser útil si se va a enlazar ese valor con un monitor de alarma o similar.

2.2.1.2 INTERRUPTOR/SENSOR

Se enviarán valores binarios (configurables) al bus cada vez que se detecten flancos de subida o de bajada en la línea de entrada. En este caso lo que importa es el hecho de pasar de un estado a otro, y no tanto la duración del tiempo del pulso. Un ejemplo sencillo podría ser un sensor que conmuta entre dos estados en función de ciertas condiciones físicas (iluminación, inundaciones, peso...), siendo uno de los cuales no deseado. Por regla general, se enviará un cierto valor binario al bus KNX cada vez que el interruptor/sensor conmute de un estado a otro.

Es posible introducir un cierto retardo antes de enviar estos valores al bus KNX (un retardo para el “0” y un retraso para el “1” sin importar cuál de los dos se envía tras cada uno de los flancos: subida o bajada). Además, es posible el **reenvío periódico** del último valor configurando el periodo deseado.

Se pueden realizar opcionalmente comprobaciones de **seguridad** para las entradas de tipo interruptor/sensor, siempre que se conecte una resistencia de fin de línea al interruptor/sensor. El valor de dicha resistencia debe ser configurado por parámetro (los valores disponibles son 2,2 k Ω , 2,7 k Ω , 3,3 k Ω , 4,7 k Ω y 10 k Ω), así como si se ha conectado en paralelo o en serie, lo que depende del tipo de interruptor/sensor (*normalmente abierto* o *normalmente cerrado*).

- Si se configura como **normalmente abierto**, la línea se mantendrá en un nivel bajo de tensión en ausencia de la condición no deseada. Sin embargo, si ocurre esta situación, se producirá un flanco de subida (el interruptor/sensor se cierra). Este tipo de sensor requiere la conexión de la resistencia de fin de línea **en paralelo**.
- Por el contrario, si se configura como **normalmente cerrado**, la línea se mantiene en un nivel de alta tensión hasta que ocurre la situación no deseada, lo que producirá un flanco de bajada (el interruptor/sensor se abre). Esto requiere la conexión de la resistencia de fin de línea **en serie**.

Por medio de esta resistencia, será posible distinguir no sólo los dos estados del interruptor/sensor, sino también niveles inesperados de voltaje (por ejemplo, cortocircuitos y circuitos abiertos debido a una **avería** o un **sabotaje**), que serán comunicados al bus a través de objetos de alarma.



Figura 4. Izquierda: sensor normalmente abierto (resistencia en paralelo).
Derecha: sensor normalmente cerrado (resistencia en serie).

2.2.1.3 BLOQUEO DE UNA ENTRADA BINARIA

Es posible **bloquear/desbloquear** cada entrada independientemente escribiendo en los objetos apropiados. Mientras una entrada permanezca bloqueada, la aplicación ignorará las nuevas conmutaciones que puedan tener lugar en la línea; sin embargo, el envío periódico de valores, si está parametrizado, no se interrumpirá (el último valor seguirá reenviándose, incluso aunque la entrada cambie de estado). Por otro lado, cuando se produce el evento de **desbloqueo**:

- **Pulsador**: se realizará una nueva evaluación del estado actual (alto/bajo) de la línea, así como la correspondiente acción.
- **Interruptor/sensor**: si se habilita específicamente en ETS, el estado actual de la línea se comparará con el que hubiera antes del bloqueo. Si difieren, se asumirá que ha tenido lugar un flanco (de subida o de bajada) mientras la entrada estaba bloqueada, y por tanto se activará la acción asociada.

2.2.1.4 ESTADOS INICIALES

Finalmente, con respecto a los **estados iniciales**, es importante tener en cuenta que:

- El **bloqueo** se mantiene después de un fallo de bus. Sin embargo, una descarga desde ETS cambia cada entrada a *desbloqueado*.
- En interruptores/sensores con seguridad, las condiciones de **alarma** se evaluarán durante la inicialización del dispositivo, y los objetos de alarma se actualizarán cuando sea necesario.
- El **envío periódico** de valores, si está configurado en ETS, se reanudará después de un fallo de tensión.
- Cuando el dispositivo se recupera de un fallo de tensión, siempre se vuelve a evaluar el **nuevo estado de los pulsadores**, mientras que el de los **interruptores/sensores** sólo se comparará con el anterior si está así configurado por parámetro (al igual que para la función de bloqueo).
- La **conmutación** de objetos binarios (0, 1, 0, 1...) siempre comienza con el valor “1” después de una descarga desde ETS. Esto es aplicable a los interruptores/sensores configurados con respuesta “Conmutar 0/1”.

PARAMETRIZACIÓN ETS

Figura 5. Entrada binaria – Configuración.

Cuando una entrada se configura como entrada binaria, aparece el objeto “[Ex] **Bloquear entrada**” (cuando recibe un “1”, la entrada se bloquea, mientras que un “0” la desbloquea).

Además, se incluye una entrada específica en el menú de la izquierda de la ventana de parámetros. Esta entrada en sí consta de una pestaña llamada **Configuración**, que contiene el siguiente parámetro:

- **Tipo:** establece si la entrada es un “Pulsador” (por defecto) o un “Interruptor/sensor”.

Pulsador

Cuando en **Tipo** se selecciona “Pulsador”, están disponibles estos parámetros:

- **Tiempo de pulsación larga:** establece el tiempo mínimo (1 a 50 décimas de segundo) que debe durar una pulsación del usuario para que se considere una pulsación larga.
- **Pulsación Corta – Acción:** permite seleccionar la acción que se llevará a cabo cuando se detecta una pulsación corta. Las opciones disponibles son:
 - Nada
 - Envío de 0/1

- **Respuesta:** establece el valor (B) a enviar a través del objeto “[Ex] [Puls. Corta] B”, que puede ser “0”, “1” o “0” y “1” alternados.
- **Retardo:** establece un retardo entre la detección de la pulsación y el envío efectivo de la respuesta. El retardo se puede indicar en segundos (0-255), minutos (0-255) u horas (0-18).
- **Envío periódico:** establece si se deben enviar periódicamente o no los valores de respuesta. Las opciones son “Siempre”, “Sólo para 0”, “Sólo para 1” y “No” (por defecto). El tiempo de ciclo se debe establecer en segundos (0-255), minutos (0-255) u horas (0-18).

➤ Control de persianas.

- **Respuesta:** permite establecer la orden concreta que se enviará al actuador de la persiana. Las opciones disponibles son:
 - “Subir”: envía un “0” por “[Ex] [Puls. corta] Subir persiana”,
 - “Bajar”: envía un “1” por “[Ex] [Puls. corta] Bajar persiana”,
 - “Subir/Bajar (conmutado)”: se mandarán los valores “0” y “1” alternados a través de “[Ex] [Puls. corta] Subir/bajar persiana”. En este caso, estará disponible también un objeto de escritura (“[Ex] Estado de la persiana (entrada)”, que se deberá enlazar con el objeto de estado de la persiana del actuador, a fin de recibir retroalimentación sobre la posición de la persiana. Esto evitará enviar órdenes de subida si la persiana está ya al 0%, o de bajada si ya está al 100%.
 - “Parar/Paso arriba”: se enviará un “0” a través de “[Ex] [Puls. corta] Parar persiana / paso arriba”,
 - “Parar/Paso abajo”: se enviará un “1” a través de “[Ex] [Puls. corta] Parar persiana / paso abajo”,
 - “Parar/Paso conmutado”: se enviarán los valores “0” y “1” (alternados con cada pulsación) a través de “[Ex] [Puls. Corta] Parar persiana / paso conmutado”,
- **Retardo:** similar al de la opción “Envío de 0/1”.

➤ Regulación de luz.

- **Respuesta:** permite establecer la orden concreta que se enviará al regulador de luz. Las opciones disponibles son:
 - “Luz ON”: se mandará un “1” a través del objeto binario “[Ex] [Puls. corta] Dimmer ON”,
 - “Luz OFF”: se mandará un “0” a través del objeto binario “[Ex] [Puls. corta] Dimmer OFF”,
 - “Luz ON/OFF (conmutado)”: se mandarán los valores “0” y “1” alternados a través de “[Ex] [Puls. Corta] Dimmer ON/OFF”,
 - “Aumentar luz”: en cada pulsación par, se enviará una orden de cuatro bits (a través de “[Ex] [Puls. corta] Aumentar Luz”) para aumentar el nivel de luminosidad en un cierto porcentaje, que se configura a través del parámetro “**Paso de regulación**”. Por otro lado, en cada pulsación impar, se enviará una orden de parar la regulación. La secuencia es, por tanto, Aumentar → Parar → Aumentar → Parar → etc.
 - “Disminuir luz”: análogo a la opción anterior pero para disminuir el nivel de luminosidad.
 - “Aumentar/Disminuir luz (conmutado)”: análogo a los dos anteriores aunque en este caso las órdenes de regulación serán aumentar/disminuir alternadas. La secuencia será, por tanto, Aumentar → Parar → Disminuir → Parar → Aumentar → etc. Las órdenes se enviarán a través del objeto “[Ex] [Puls. corta] **Aumentar/disminuir luz**”. En este caso, estará disponible también un objeto de escritura (“[Ex] **Estado del dimmer (entrada)**”) que se debe enlazar con el objeto de estado del regulador, para retroalimentarse con el nivel de iluminación actual. Esto evitará enviar órdenes de aumentar si el nivel actual ya es del 100% u órdenes de disminuir si el nivel ya es del 0%.
- **Retardo:** similar al de la opción “Envío de 0/1”.

- Envío de escena.
 - **Respuesta:** establece la orden concreta que se enviará al bus. Las opciones son “Ejecutar escena” (se enviará la orden para ejecutar la escena especificada a través de “[Ex] [Puls. corta] Ejecutar escena”) y “Grabar escena” (se enviará la orden para grabar la escena especificada a través de “[Ex] [Puls. corta] Grabar escena”).
 - **Escena:** establece el número de escena deseado (1 a 64) para las anteriores órdenes de ejecutar/grabar.
 - **Retardo:** similar al de la opción “Envío de 0/1”.
- Constante de 1 byte (entero):
 - **Respuesta:** establece un valor constante que se enviará al bus (a través de “[Ex] [Puls. corta] Valor constante (entero)”), en el rango de 0 a 255.
- Constante de 1 byte (porcentaje):
 - **Respuesta:** establece un valor constante que se enviará al bus (a través de “[Ex] [Puls. corta] Valor constante (porcentaje)”), en el rango de 0 a 100.
- Constante de 2 byte (entero):
 - **Respuesta:** establece un valor constante que se enviará al bus (a través de “[Ex] [Puls. corta] Valor constante (entero)”), en el rango de 0 a 65535.
- Constante de 2 byte (coma flotante):
 - **Respuesta:** establece un valor constante que se enviará al bus (a través de “[Ex] [Puls. corta] Valor constante (coma flotante)”), en el rango de -671088,64 a 670760,96.
- **Pulsación larga – Acción:** permite seleccionar la acción que se llevará a cabo cuando se detecta una pulsación larga. Las acciones disponibles son totalmente análogas a las de pulsación corta.

Interruptor/Sensor

Cuando en **Tipo** se selecciona "Interruptor/Sensor", están disponibles los siguientes parámetros:

- **Seguridad:** marcando o desmarcando esta casilla se determina si la línea de entrada incluye una resistencia de fin de línea de manera que sea posible monitorizar un sabotaje o avería (que se notificarán enviando periódicamente el valor "1" a través del objeto "[Ex] Alarma: avería, sabotaje, línea inestable"; una vez terminen, se enviará un "0" por este objeto). Al seleccionarla aparecen dos parámetros más:
 - **Tipo de interruptor/sensor:** establece si el interruptor/sensor es de tipo normalmente abierto y, por tanto, con una resistencia conectada en paralelo ("N.A. (resistencia en paralelo)") o normalmente cerrado y, por tanto, con una resistencia conectada en serie ("N.C. (resistencia en serie)").
 - **Valor de resistencia:** establece el valor de la resistencia, que debe ser uno de los siguientes: 2,2 kΩ, 2,7 kΩ, 3,3 kΩ, 4,7 kΩ and 10 kΩ.
- **Acciones.**
 - **Flanco de subida:** permite seleccionar la acción que se llevará a cabo ante un flanco de subida en la línea. Las opciones son "Nada" (por defecto), "0", "1" y "Conmutar 0/1" (es decir, los valores "1" y "0" se alternarán con cada flanco de subida detectado). Estos valores se envían a través del objeto "[Ex] [Interruptor/sensor] Flanco".
 - **Flanco de bajada:** análogo al parámetro anterior. La respuesta a los flancos de bajada se enviará a través del mismo objeto ("[Ex] [Interruptor/sensor] Flanco").
- **Envío periódico.**
 - **Envío periódico de "0":** establece cada cuánto tiempo (0 a 255 segundos, 0 a 255 minutos, 0 a 18 horas) se enviará periódicamente el valor "0", una vez se ha detectado el flanco correspondiente. Si no es necesario el envío periódico, se debe dejar este parámetro a 0.

- **Envío periódico de “1”**: análogo al anterior, pero para el valor “1”.

- **Retardo.**

- **Retardo al enviar “0”**: establece un cierto retardo (0 a 255 segundos, 0 a 255 minutos, 0 a 18 horas) antes de mandar el valor “0”, una vez se ha detectado el flanco correspondiente. Para un envío inmediato, dejar este parámetro a 0.

- **Retardo al enviar “1”**: análogo al anterior, pero para el valor “1”.

- **Evaluar el estado de la entrada después de desbloquear o reiniciar**: establece si el estado de la línea de entrada debe ser evaluado o no cuando se desbloquea (a través del objeto “[Ex] Bloquear entrada”) y a la vuelta de un fallo de tensión, de forma que el nuevo estado pueda compararse con el último conocido, haciendo que el dispositivo ejecute la respuesta adecuada en caso de que sean diferentes.

- **Envío de estado (0/1) al volver la tensión**: determina si el estado de la línea (es decir, la correspondiente acción) debe enviarse siempre al bus cuando el dispositivo se recupera de un fallo de tensión, incluso si el estado es el mismo que antes del fallo de tensión.

2.2.2 SONDA DE TEMPERATURA

Conectando una sonda de temperatura (como los modelos ZN1AC-NTC68 S / E / F de Zennio) a una de las entradas será posible tomar medidas de temperatura y monitorizarlas en el MAXinBOX 66, que será capaz de **enviar estos valores al bus** así como **reportar situaciones de temperatura alta / baja**.

PARAMETRIZACIÓN ETS

Figura 6. Sonda de temperatura - Configuración.

Cuando una entrada se ha configurado como sonda de temperatura, aparecen los objetos “[Ex] Temperatura actual” (dos bytes) y “[Ex] Error de sonda” (un bit). El primero informará del valor actual de temperatura (periódicamente o después de un cierto incremento/decremento, según la configuración), mientras que el segundo advertirá (mandando el valor “1” periódicamente) de lecturas inesperadas en la línea de entrada debido a una **avería** o una conexión incorrecta de la sonda de temperatura.

Aparte de esto, aparece una entrada específica en el menú de la izquierda de la ventana de parámetros. Esta entrada en sí consta de una pestaña llamada **Configuración**, que contiene los siguientes parámetros:

- **Calibración del sensor de temperatura:** define un *offset* a aplicar a la medición recibida de la sonda para corregir las desviaciones debido a factores externos. El *offset* debe estar en el rango [-50, 50] décimas de grado.
- **Periodo de envío de la temperatura:** fija cada cuánto tiempo se debe enviar al bus el valor actual de temperatura (a través del objeto “[Ex] Temperatura actual”), en el rango 0 – 255 decenas de segundos. El valor “0” inhabilita el envío periódico.

- **Envío tras un cambio de temperatura:** define un umbral (en el rango de 0 a 255 décimas de grado) de manera que, cada vez que se detecte una lectura de temperatura que difiera del último valor enviado al bus en más de dicho umbral, se llevará a cabo un envío extra.
- **Protección de temperatura:** lista desplegable con las siguientes opciones:
 - No: no se requiere protección de la temperatura.
 - Sobrecalentamiento: se requiere protección contra sobrecalentamiento. Aparecen dos nuevos parámetros:
 - **Temp. de sobrecalentamiento:** temperatura máxima permitida, en el rango de 30 a 125 °C. Las lecturas de temperatura superiores a esta serán consideradas sobrecalentamiento y, por tanto, se enviará periódicamente un “1” a través del objeto “[Ex] Sobrecalentamiento”. Una vez que no haya sobrecalentamiento, se enviará un “0” (una vez).
 - **Histéresis:** banda muerta o umbral (en el rango de 1 a 200 décimas de grado – es decir, dos grados–) alrededor de la temperatura de sobrecalentamiento definida anteriormente. Esta banda evita que el dispositivo envíe la alarma y la no alarma de sobrecalentamiento una y otra vez cuando la temperatura actual está oscilando en torno al límite de sobrecalentamiento (T): una vez que se ha activado la alarma de sobrecalentamiento, no se enviará no alarma hasta que la temperatura actual sea menor que T menos la histéresis. Después de esto, si la temperatura actual alcanza de nuevo T, se reenviará la alarma.
 - Sobreenfriamiento: se requiere protección contra sobreenfriamiento. Aparecen dos nuevos parámetros (análogos a los dos anteriores):
 - **Temp. de sobreenfriamiento:** temperatura mínima permitida, en el rango de 30 a 125°C. Las lecturas de temperatura inferiores a esta serán consideradas sobrecalentamiento y, por tanto, se enviará periódicamente un “1” a través del objeto “[Ex] Sobreenfriamiento”. Una vez que no haya sobreenfriamiento, se enviará un “0” (una vez).
 - **Histéresis:** banda muerta o umbral (en el rango de 1 a 200 décimas de grado, el valor por defecto es de 20 décimas de grado) alrededor de la temperatura de sobreenfriamiento definida anteriormente. Al igual que

para el sobrecalentamiento, una vez que se ha activado la alarma, no se enviará no alarma hasta que la temperatura actual sea mayor que T más la histéresis. Después de esto, si la temperatura actual alcanza de nuevo T , se reenviará la alarma.

- Sobrecalentamiento y sobreenfriamiento: se requieren ambas protecciones, sobrecalentamiento y sobreenfriamiento. Aparecen los siguientes tres parámetros:

- **Temp. de sobrecalentamiento.**
- **Temp. de sobreenfriamiento.**
- **Histéresis.**

Los tres son análogos a los que ya se explicaron por separado.

2.2.3 DETECTOR DE MOVIMIENTO

Es posible conectar detectores de movimiento a los puertos de entrada del MAXinBOX 66. Esto ofrece la posibilidad al actuador de monitorizar movimiento y presencia en la habitación, al mismo tiempo que el nivel de luminosidad. En función de la detección, es posible parametrizar diferentes acciones de respuesta.

Dado que la funcionalidad y la configuración de los parámetros de la función de detección de movimiento puede variar ligeramente en función de cada dispositivo Zennio, deberá consultarse el manual de usuario específico “**Detector de movimiento Zennio para MAXinBOX 66**” (disponible en la página web de Zennio, www.zennio.com).

Por otro lado, se debe tener en cuenta que sólo los detectores con referencias **ZN1IO-DETEC-P** y **ZN1IO-DETEC-X** son compatibles con el MAXinBOX 66, siempre y cuando se configure el correspondiente microinterruptor en el modo **Tipo B**.

Se recomienda asimismo la lectura de la **hoja técnica del modelo concreto** del detector de movimiento elegido.

Nota: *los detectores con referencias ZN1IO-DETEC y ZN1IO-DETEC-N, al ser de Tipo A, pueden reportar mediciones inexactas si se conectan al MAXinBOX 66.*

2.3 SALIDAS

El actuador MAXinBOX 66 incorpora **6 salidas de relé** que permiten el control independiente de hasta **6 cargas diferentes**, o hasta **3 persianas** (u otros sistemas de cerramientos equivalentes), denominadas A, B y C. Cada salida puede ser habilitada o deshabilitada por parámetro y realizar diferentes funciones.

El MAXinBOX 66 también cuenta con un conjunto de botones integrados que ofrecen la opción de controlar manualmente los canales de salida, por ejemplo para realizar pruebas. La forma de la que las salidas reaccionan a estos botones depende de cómo se haya parametrizado el dispositivo, como se explica en secciones posteriores.

PARAMETRIZACIÓN ETS

Cuando se han activado las **Salidas** en la pantalla General de parámetros, la siguiente pantalla estará disponible para la selección de la función específica de cada canal de salida.

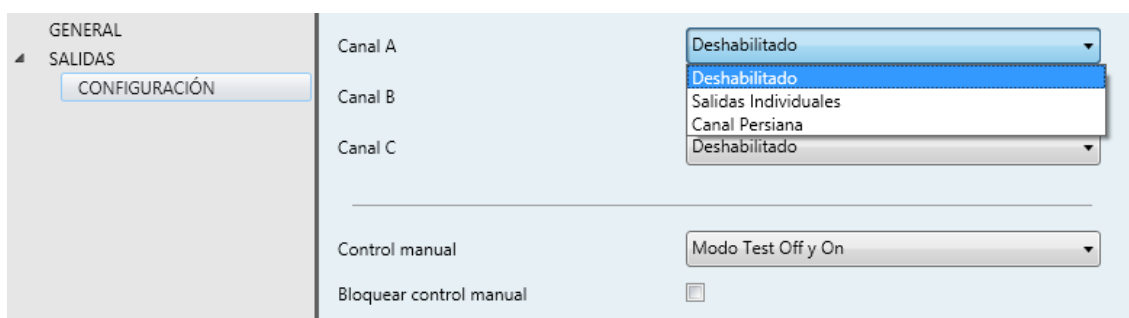


Figura 7. Salidas - Configuración.

Todas las salidas están inhabilitadas por defecto. En función de la opción seleccionada para cada salida, se incluirán algunas pestañas adicionales en el menú de la izquierda.

2.3.1 SALIDAS INDIVIDUALES

Cada salida individual se puede configurar como **normalmente abierta** (es decir, la activación de la salida provoca el cierre del relé) o **normalmente cerrada** (es decir, la activación de la salida provoca la apertura del relé).

Además del tipo, se pueden configurar las siguientes funcionalidades para cada salida individual:

- **Temporizaciones:** permite realizar acciones temporizadas sobre las salidas.

En primer lugar, la función de **intermitencia** consiste en realizar una secuencia continua de encendidos y apagados temporizados cuando se recibe un objeto determinado de activación.

Por otro lado, la función de **temporización simple** consiste en realizar una única secuencia temporizada de encendido/apagado cuando se recibe un “1” (o un “0”) a través de un objeto específico (“**[Sx] Temporización**”). Dependiendo de si la orden es de encendido o de apagado, se aplicará un cierto retardo (“**Retardo de encendido**” o “**Retardo de apagado**”). Estos retardos son parametrizables. Además, en el caso del encendido se debe definir una “**Duración de encendido**”. Este es el tiempo que la salida permanecerá encendida una vez que el Retardo de encendido ha expirado y la salida finalmente ha conmutado.

- La función de **multiplicación**, cuando está activada, permite multiplicar progresivamente (tantas veces como se reciban las ordenes de encendido o apagado a través de “**[Sx] Temporización**”) el retardo de encendido, el retardo de apagado o la duración de encendido definidos inicialmente, en función de cuál esté en marcha. Se distinguen diferentes situaciones:

- **Sin multiplicación:**
 - Si la cuenta del retardo de encendido ya está en marcha, se reiniciará cada vez que se reciba de nuevo un “1” por el objeto “**[Sx] Temporización**”.
 - Si el encendido ya se ha producido y está en curso la cuenta de la duración de encendido, ésta se reiniciará cada vez que se reciba de nuevo el valor “1”.

- Si la cuenta del retardo de apagado ya está en marcha, se reiniciará cada vez que se reciba un nuevo valor "0"
- **Con Multiplicación:**
 - Si la cuenta del retardo de encendido ya está en marcha y se recibe varias veces el valor "1" por el objeto "[Sx] **Temporización**", entonces el tiempo de retardo real será "n" veces el parametrizado, siendo n el número de veces que llegue el valor "1" (**5 veces** como máximo; las siguientes recepciones se ignorarán hasta la siguiente activación de la función de temporización simple).
 - Si el encendido ya se ha producido y está en curso la cuenta de la duración de encendido, entonces ésta pasará a ser "n" veces el valor parametrizado, siendo "n" el número de veces que llegue el valor "1" (de nuevo, **5** como máximo).
 - Si la cuenta del retardo de apagado ya está en marcha, el retardo real será "n" veces el parametrizado, siendo "n" el número de veces que se reciba el valor "0" (**5** como máximo).

Nota: la multiplicación puede resultar muy útil en ausencia de retardos de encendido y de apagado. No obstante, como ya se ha explicado y como se muestra en el próximo ejemplo, es posible aplicarla a los retardos en caso de haberse establecido un cierto valor para ellos.

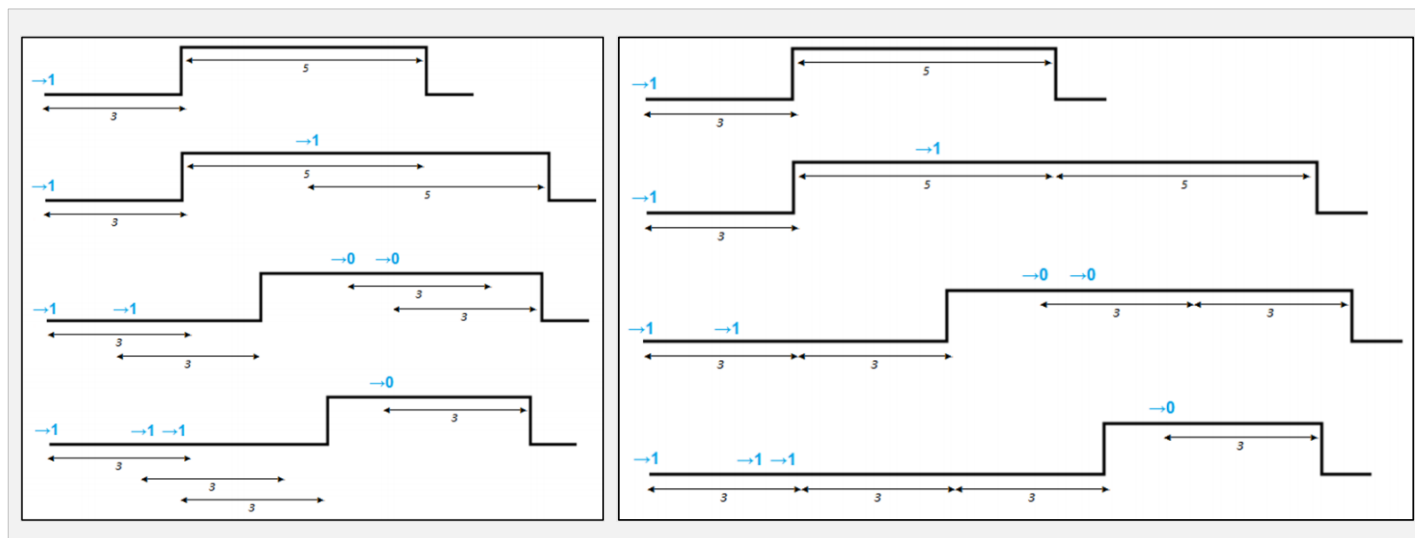


Figura 8. Sin multiplicación (izquierda). Con multiplicación (derecha).

El ejemplo anterior muestra cómo funciona la multiplicación cuando se parametrizan retardos de encendido y apagado de 3 segundos y una duración de encendido de 5 segundos. “→0” y “→1” representan llegadas sucesivas del valor “0” ó “1” a través de “[Sx] Temporización”.

- **Escenas:** permite ejecutar el encendido o el apagado de la salida ante la llegada de un objeto de escena.
- **Alarmas:** permite cambiar el estado de las salidas al llegar una señal de alarma desde el bus KNX. Puede configurarse el estado en que quedará la salida tanto cuando la alarma se activa como cuando se desactiva.

Es posible la **monitorización cíclica** de la señal de alarma definiendo un cierto periodo de tiempo. El actuador comprobará que el estado de alarma o no alarma se recibe al menos una vez antes de que el periodo expire (nota: esta comprobación no tiene lugar hasta que el objeto se haya recibido al menos una vez). En caso de que el objeto deje de ser actualizado (es decir, de que el actuador deje de recibir valores a través de él), se llevará a cabo la acción de alarma, por razones de seguridad.

Con respecto a la desactivación de la alarma, también es posible configurar una desactivación simple o una desactivación **con enclavamiento**.

- En el primer caso se desencadena la acción de desactivación tan pronto como el objeto de alarma recupere su valor normal.
- El segundo caso, en cambio, requiere que se reciba una orden de confirmación (a través de otro objeto) después de que el objeto de alarma tome su valor normal.

Nota: *las alarmas siempre prevalecen sobre el resto de las funciones (por ejemplo: las órdenes de bloqueo durante el estado de alarma se ignorarán; las órdenes de alarma durante el estado de bloqueo, no).*

- **Acción ante bloqueo:** permite conmutar la salida a un cierto estado (Encender / Apagar / No cambiar) al recibir el valor “1” a través del objeto de bloqueo específico (uno por salida).
- **Acción ante apagado:** permite conmutar la salida a un cierto estado

(Encender / Apagar / No cambiar) durante el apagado del actuador (por ejemplo, ante un fallo de bus).

- **Arranque:** ofrece la posibilidad de conmutar la salida a un cierto estado durante la puesta en marcha del actuador.
 - Por defecto: apagado después de una descarga de ETS, y sin cambios después de un fallo de bus.
 - Personalizado: apagado / encendido / último, bien después de una descarga de ETS o bien tras un fallo de bus (último será apagado en la primera puesta en marcha). Opcionalmente se puede mandar el objeto de estado al bus (con un retardo configurable).

PARAMETRIZACIÓN ETS

Figura 10. Habilitación de cada salida por separado.

Cuando se configura un canal con Salidas individuales, aparecen dos casillas para habilitar o inhabilitar independientemente las dos salidas de relé que lo constituyen.

Figura 11. Salida individual - Configuración

Por cada salida individual se incluye una pantalla específica de configuración en el menú de la izquierda. Esta pantalla contiene los siguientes parámetros:

- **Tipo:** se debe establecer en “normalmente abierto” (predeterminado) o “normalmente cerrado”, dependiendo del estado *natural* del relé.

- **Temporizaciones:** activa o desactiva las funciones de temporización, que se configuran desde una pantalla de parámetros específica (ver más adelante).
- **Escenas:** activa o desactiva la función de escenas, que debe configurarse desde una pantalla de parámetros específica (véase más adelante).
- **Alarmas:** activa o desactiva la función de alarmas, que debe configurarse desde una pantalla de parámetros específica (véase más adelante).
- **Acción ante bloqueo:** establece si dejar el relé como está (“No cambiar”, opción por defecto) o conmutarlo a apagado (“Apagar”) o a encendido (“Encender”) en caso de producirse una orden de bloqueo a través del objeto “[Sx] Bloquear” (1 = bloqueo; 0 = desbloqueo).
- **Acción ante apagado:** establece si dejar el relé como está (“No cambiar”, opción por defecto) o conmutarlo a *off* (“Apagar”) o a *on* (“Encender”) en caso de apagarse el dispositivo debido por ejemplo a un fallo de tensión.
- **Arranque:** establece si se debe realizar la acción por defecto (“Por defecto”) o una acción personalizada (“Personalizado”) sobre la salida durante el arranque del dispositivo. Ésta última opción debe configurarse desde una pantalla de parámetros específica (véase más adelante).

Después de configurar una salida como Salida individual, aparecen por defecto tres objetos de comunicación más:

- **[Salidas] Escenas:** objeto de un byte para la recepción de los valores de escena (0-63 para ejecutar las escenas 1-64 y 128-191 para guardar las escenas 1-64) desde el bus KNX. Al tratarse de un objeto único para todas las salidas individuales que se habiliten, está disponible tan pronto se activa la primera de ellas..
- **[Sx] Encender/apagar:** objeto de un bit para la recepción de órdenes de apagado/encendido desde el bus. Un “1” hará que la salida se encienda y un “0” que se apague. Dependiendo de si la salida se ha marcado como normalmente cerrada o normalmente abierta, encender la salida consistirá en abrir o en cerrar el relé (y la acción opuesta para el apagado).
- **[Sx] Encender/apagar (estado):** objeto de un bit que informará del estado actual de la salida individual. Se puede leer en cualquier momento y

responderá con un “1” o un “0” en función de si la salida está encendida o apagada, respectivamente.

Pestaña Temporizaciones

Las dos funciones de temporización se pueden habilitar a través de sus correspondientes casillas:

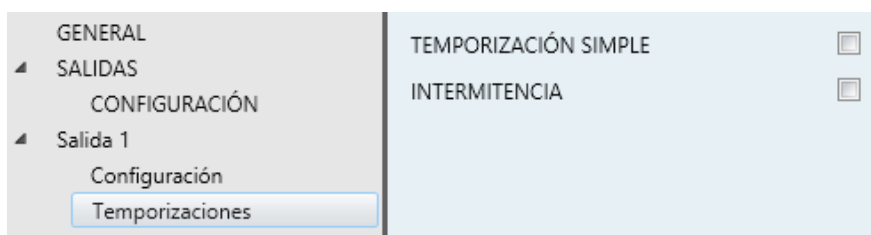


Figura 12. Salida individual - Temporizaciones.

- **Temporización simple:** activa o desactiva la función de temporización simple, es decir, la ejecución de un apagado/encendido temporizado tras la recepción de un “1” o un “0” a través de “[Sx] Temporización”, de acuerdo a los siguientes parámetros:

Figura 13. Salida individual – Temporización simple.

- **Retardo de encendido:** determina si la acción de “Encendido” debe ejecutarse inmediatamente (“0”; opción por defecto) después de recibir un “1” a través del objeto “[Sx] Temporización”, o si se debe retrasar un tiempo determinado (0 a 600 décimas de segundo; 0 a 3600 segundos; 0 a 1440 minutos; 0 a 24 horas).
- **Retardo de apagado:** análogo al anterior, pero referido únicamente a la orden de apagado que se lleva a cabo cuando se recibe un “0” a través del objeto “[Sx] Temporización”.
- **Duración de encendido:** establece el tiempo que la salida permanecerá encendida una vez que el retardo de encendido termina y la salida finalmente se enciende (0 a 600 décimas de segundo; 0 a 3600 segundos; 0 a 1440 minutos; 0 a 24 horas). Si se deja a cero, la salida no se apagará después de un tiempo.
- **Tiempo de aviso:** establece un tiempo de anticipación (0 a 600 décimas de segundo; 0 a 3600 segundos; 0 a 1440 minutos; 0 a 24 horas) antes de la acción de apagado de manera que se producirá una intermitencia en la salida (de dos segundos) para notificar que la temporización está a punto de terminar. Además, el objeto “[Sx] Tiempo de aviso (estado)” adquiere el valor de “1” mientras esa intermitencia está en marcha (y “0” en cualquier otro momento). Si se deja a cero, no se llevará a cabo la intermitencia de advertencia. Hay que tener en cuenta que este tiempo tiene que ser menor que el tiempo de duración de encendido.
- **Multiplicación:** habilita o inhabilita la función de multiplicación.
- **Intermitencia:** cuando se activa, se muestra el objeto “[Sx] Intermitencia” en el proyecto. Si se recibe un “1” a través de dicho objeto se activa la intermitencia de la salida, mientras que con un “0” se detiene. Esta intermitencia estará sujeta a los siguientes parámetros:
 - **Duración de encendido:** tiempo de cada fase de *on* (5 a 600 décimas de segundo; 1 a 3600 segundos; 1 a 1440 minutos; 1 a 24 horas).
 - **Duración Apagado:** tiempo de cada fase de *off* (5 a 600 décimas de segundo; 1 a 3600 segundos; 1 a 1440 minutos; 1 a 24 horas).

- **Repeticiones:** número de veces que se llevará a cabo la secuencia de encendido/apagado. Si se deja a “0”, la intermitencia sólo se detendrá cuando se reciba un “0” a través del objeto “[Sx] Intermitencia”.

INTERMITENCIA ☒

Duración Encendido 1 s

Duración Apagado 1 s

Repeticiones (0 = Sin límite) 0

Figura 14. Salida individual -Intermitencia.

Pestaña Escenas

GENERAL

▲ SALIDAS

CONFIGURACIÓN

▲ Salida 1

Configuración

Escenas

Número de escenas activas 0

Figura 15. Salida individual - Escenas.

Se puede configurar hasta cinco escenas. El número de escenas se debe introducir en **Número de escenas activas** (“0”, por defecto).

Nota: después de escribir el valor, es necesario mover el foco del cursor a otra parte de la ventana para que se actualice y aparezcan los parámetros requeridos.

Dependiendo del número anterior, los dos siguientes parámetros aparecen varias veces (una por cada escena):

- **Número de escena:** establece el número de escena deseado, de modo que cuando se recibe ese valor (decrementado en uno, de acuerdo al estándar KNX) a través del objeto “[Salidas] Escenas”, se activará la acción configurada a continuación.
- **Acción:** “Apagar” (por defecto) o “Encender”, dependiendo del estado que la salida debe adquirir cuando se activa la escena.

Pestaña Alarmas

Figura 16. Salida individual - Alarmas.

La pantalla de configuración “Alarma” contiene los siguientes parámetros:

- **Disparador:** establece el valor (“0” –por defecto– o “1”) que, al recibirse desde el bus KNX a través del objeto “[Sx] Alarma”, será interpretado por el actuador como un desencadenante de la alarma, por lo que se iniciará la acción configurada más abajo.
- **Periodo de monitorización cíclica:** establece cada cuánto tiempo, como máximo, cabe esperar la recepción del objeto de alarma desde el bus una vez haya tenido lugar una primera recepción. Si se excede, y por razones de seguridad, la acción de alarma también se activará. Si este parámetro se deja a cero (opción por defecto), la función de monitorización cíclica permanecerá inhabilitada. Los valores permitidos son 5 a 600 décimas de segundo; 1 a 3600 segundos; 1 a 1440 minutos; 1 a 24 horas.
- **Acción:** “No cambiar” (por defecto), “Apagar”, “Encender” o “Intermitencia”.
Seleccionando la última opción aparecen más parámetros:
 - “Duración de encendido”, “Duración de apagado” y “Repeticiones”: todos ellos análogos a los de la función de Intermitencia, dentro de Temporizaciones (ver los parámetros anteriores).
- **Desactivación – Modo:** “Normal” (por defecto) o “Con enclavamiento (necesario desenclavar)”. La segunda opción habilita un nuevo objeto de un bit, “[Sx] Desenclavar alarma”, que debe usarse para confirmar la alarma

externamente una vez que se ha recibido el valor de no alarma a través del objeto “[Sx] Alarma” (es decir, el valor inverso del de activación).

Nota: el desenclavamiento debe enviarse necesariamente después de que el objeto “[Sx] Alarma” haya adquirido el valor de no alarma. Si se envía mientras el valor de este objeto sea el de alarma, no tendrá efecto.

- **Desactivación – Acción:** establece el estado que debe adquirir la salida una vez se ha desactivado la alarma (y desenclavado, si se requiere). Puede ser: “No cambiar” (por defecto), “Apagado”, “Encendido” o “Último (antes de alarma)”.

Pestaña Arranque

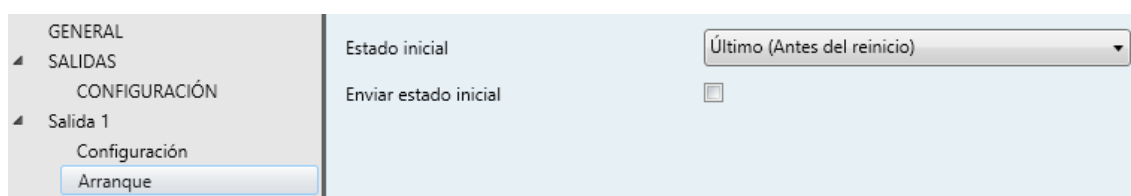


Figura 17. Salida individual - Arranque.

Contiene los siguientes parámetros:

- **Estado inicial:** permite establecer el estado que adquirirá la salida tras la puesta en marcha del actuador. Puede ser: “Último (antes del reinicio)” (por defecto; durante la primera puesta en marcha, la salida permanecerá apagada), “Apagado” o “Encendido”.
- **Enviar estado inicial:** establece si el objeto de estado de la salida se debe enviar al bus KNX (con el fin de actualizar otros dispositivos KNX) después de la puesta en marcha del actuador. Es posible imponer un retardo (0 a 600 décimas de segundo; 0 a 3600 segundos; 0 a 1440 minutos; 0 a 24 horas) para asegurar que este valor se envía una vez que los dispositivos están listos para recibirlo.

2.3.2 CANALES DE PERSIANA

Cada canal de persiana es capaz de controlar el movimiento de una persiana en el sistema domótico por medio de dos funcionalidades complementarias.

- **Control básico** (órdenes simples de subida/bajada).
- **Control preciso** de posicionamiento, tanto de la persiana como de las lamas (si las hay).

Cada canal de persiana (A, B, etc.) consiste en dos salidas de relé consecutivas (es decir, el canal A está formado por las salidas 1 y 2; el canal B está formado por las salidas 3 y 4; etc.). La primera salida de cada canal enviará pulsos eléctricos para subir la persiana; mientras que la segunda salida mandará pulsos para bajar la persiana. Los cables del motor de la persiana deben conectarse al actuador de acuerdo a lo anterior.

Canal	Salidas	Acción
A	Salida 1	Subir
	Salida 2	Bajar
B	Salida 3	Subir
	Salida 4	Bajar
(...)	(...)	(...)

Tabla 1. Canal persiana. Acciones de las salidas.

Cada canal puede configurarse como “**Persiana (sin lamas) / toldo**” o como “**Persiana con lamas orientables**”.

Además del tipo de persiana, es posible configurar lo siguiente para los canales persiana:

- **Tiempos:** tiempos que definen el movimiento de la persiana: la longitud del recorrido de subida, la longitud del recorrido de bajada y un retardo de seguridad para la inversión de marcha (para evitar problemas mecánicos y eléctricos debidos a un cambio súbito de dirección de la persiana). Además, es posible establecer un tiempo adicional de fin de carrera que se aplicará una vez que la persiana llega a su límite (superior o inferior).

Para las persianas con **lamas** es necesario configurar también un tiempo de giro completo y un tiempo de paso.

Con respecto a las lamas, es posible determinar si el actuador debe *forzar* a las lamas a mantener su posición después del movimiento de la persiana. Téngase en cuenta que, debido a que la persiana y las lamas se mueven al mismo tiempo, mantener la posición de las lamas en ocasiones requiere realizar una cierta corrección de la posición de la persiana.

- **Objetos de estado:** informan de la posición actual de la persiana y de las lamas (si existen). 0% significa que la persiana está totalmente subida (o las lamas totalmente abiertas, dejando pasar la luz), mientras que 100% significa que la persiana está totalmente bajada (o las lamas totalmente cerradas).
- **Objetos de control preciso:** ofrecen la posibilidad de subir/bajar la persiana o las lamas a la posición exacta deseada, en términos de porcentaje.
- **Escenas:** permiten situar la persiana (o las lamas) en una determinada posición al recibir un objeto de escena.
- **Alarmas:** permiten mover la persiana (o las lamas) a una posición concreta al recibir la señal de alarma por el bus KNX. Se dispone de dos alarmas (es decir, dos objetos de activación con posiciones finales independientes) por cada canal de persiana. Puede configurarse la posición de la persiana ante la **activación** de la alarma e igualmente ante la **desactivación**.

También es posible la **monitorización cíclica** de la señal de alarma definiendo un cierto periodo de tiempo. El actuador comprobará que el estado de alarma o no alarma se recibe al menos una vez antes de que el periodo expire (nota: esta comprobación no tiene lugar hasta que el objeto se haya recibido al menos una vez). En caso de que el objeto deje de ser actualizado (es decir, de que el actuador deje de recibir valores a través de él), se llevará a cabo la acción de alarma, por razones de seguridad.

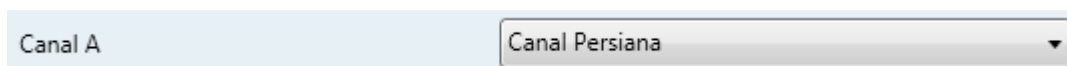
Con respecto a la desactivación de la alarma, también es posible configurar una desactivación simple o una desactivación **con enclavamiento**.

- En el primer caso se desencadena la acción de desactivación tan pronto como el objeto de alarma recupere su valor normal.
- El segundo caso, en cambio, requiere que se reciba una orden de confirmación (a través de otro objeto) después de que el objeto de alarma tome su valor normal.

Nota:

- **Las alarmas siempre prevalecen sobre el resto de las funciones** (por ejemplo: las órdenes de bloqueo durante el estado de alarma se ignorarán; las órdenes de alarma durante el estado de bloqueo, no).
- **La alarma 1 tiene preferencia sobre la alarma 2.** Si un canal se encuentra en estado de alarma 2 y se activa la alarma 1, ejecutará la acción de alarma 1 y esperará hasta que la alarma 1 sea desactivada (retomando el estado de alarma 2, aunque no volviendo a ejecutar la acción de alarma 2). Sin embargo, si un canal se encuentra en estado de alarma 1 y se activa la alarma 2, ésta será ignorada y el canal continuará en estado de alarma 1, hasta que esta primera alarma sea desactivada, momento en que el canal sí ejecutará la acción de alarma 2.
- **Movimiento inverso:** permite controlar una persiana con las órdenes habituales invertidas. Mientras que el objeto de control de persiana por defecto reacciona a un “1” y a un “0” subiendo y bajando la persiana respectivamente, se habilitará otro objeto específico para el control en el sentido inverso (“1” para bajar y “0” para subir).
- **Posicionamiento directo:** permite, por medio de un objeto de comunicación de un bit, mover la persiana (y las lamas, en su caso) a una posición predefinida. Por cada canal de persiana se implementan dos funciones de posicionamiento directo, cada una con la opción de *aprender* nuevas posiciones finales en tiempo de ejecución (sobrescribiendo la definida por parámetro) simplemente mandando un “1” al objeto concreto de “salvado”.
- **Arranque:** ofrece la posibilidad de situar la persiana en una posición particular durante la inicialización del actuador.
 - **Por defecto:** dejará la persiana como está. Tener en cuenta que en la primera inicialización (después de una descarga de ETS), el actuador asumirá que la persiana está completamente subida (al 0%).
 - **Personalizado:** Posición actual / Arriba / Abajo / Posición específica, tanto después de una descarga de ETS como de un fallo de tensión de bus. Opcionalmente se puede mandar el estado de los objetos al bus después de un retardo configurable.

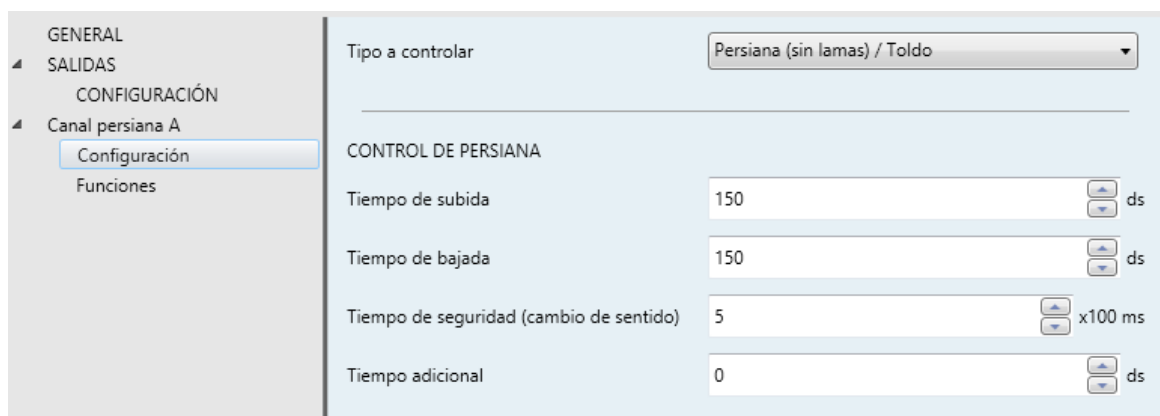
PARAMETRIZACIÓN ETS



Canal A Canal Persiana ▼

Figura 18. Habilitación de un canal persiana.

Pestaña Configuración



GENERAL
 SALIDAS
 CONFIGURACIÓN
 Canal persiana A
 Configuración
 Funciones

Tipo a controlar Persiana (sin lamas) / Toldo ▼

CONTROL DE PERSIANA

Tiempo de subida 150 ds

Tiempo de bajada 150 ds

Tiempo de seguridad (cambio de sentido) 5 x100 ms

Tiempo adicional 0 ds

Figura 19. Canal persiana - Configuración.

Esta pantalla contiene los siguientes parámetros:

- **Tipo a controlar:** define el canal de persiana como “Persiana (sin lamas) / toldo” o “Persiana con lamas orientables”.
- **Tiempo de subida:** tiempo total que necesita la persiana para subir (posición = 0%) cuando está completamente bajada (posición = 100%). El rango permitido va de 5 a 30000 segundos.
- **Tiempo de bajada:** tiempo total necesario para realizar el recorrido inverso (de 0% a 100%). El rango permitido va de 5 a 30000 segundos.
- **Tiempo de seguridad (cambio de sentido):** establece el tiempo (de 5 a 255 x100 milisegundos) que el actuador esperará si ya estaba en movimiento y llega una orden de cambio de dirección.
- **Tiempo adicional:** establece un tiempo de fin de carrera (de 0 a 255 décimas de segundo; por defecto, 0) con el fin de garantizar que la persiana alcance las posiciones finales del recorrido.

En el caso particular de configurar la persiana como “Persiana con lamas orientables”, aparecen algunos parámetros más:

- **Tiempo de giro completo:** tiempo total que tardan las lamas en realizar un recorrido completo entre “totalmente abierta” (0%; deja pasar la luz) y “totalmente cerrada” (100%). Los valores permitidos son de 1 a 255 décimas de segundo.
- **Tiempo de paso:** duración del pulso a enviar al actuador cuando se recibe una orden de paso arriba/abajo. Los valores permitidos son de 1 a 255 centenas de milisegundos.
- **Recuperar la posición al completar el movimiento:** determina si el actuador debe o no tratar de conservar la posición de las lamas tras el movimiento de la persiana.
- **Posición de lamas con la persiana abajo:** define la posición a la que se deben mover las lamas cuando la persiana está abajo (100%, completamente bajada)

Los siguientes objetos son visibles por defecto para cualquier canal de persiana:

- **[Persianas] Escenas:** objeto de un byte para la recepción de las órdenes de escena desde el bus KNX, cuya respuesta debe configurarse desde la pestaña concreta de Escenas para cada canal de persiana, como se explicará más abajo.
- **[Cx] Mover:** objeto de un bit para la recepción de órdenes de movimiento hacia arriba (“0”) o movimiento hacia abajo (“1”).
- **[Cx] Parar:** objeto de un bit para la recepción de órdenes de parada de la persiana (“0” o “1”) desde el bus KNX, que interrumpirán cualquier orden de movimiento arriba o abajo que se encuentren en ejecución. Si la persiana se ha configurado como “Persiana con lamas”, el nombre del objeto será “[Cx] Parar/Paso” e implementará exactamente la misma funcionalidad (interrumpir una orden de movimiento arriba o abajo en ejecución), además de la función de control de paso: si la persiana está ya parada, un “0” será interpretado como una orden de paso arriba, y un “1” se interpretará como una orden de paso abajo.

Nota: las órdenes sucesivas de paso de lamas recibidas antes de concluir un paso de lamas anterior reiniciarán la cuenta del tiempo de paso.

- **[Cx] Bloquear:** objeto de un bit para bloquear (“1”) o desbloquear (“0”) externamente la persiana. Cuando se recibe la señal de bloqueo, el actuador interrumpirá cualquier acción que esté en marcha e ignorará futuras órdenes recibidas desde el bus hasta que se reciba la señal de desbloqueo.

Nota: las órdenes de bloqueo se ignoran si la alarma está activada (aunque el estado de alarma también provoca que la persiana no responda a las órdenes externas).

Pestaña Funciones

Esta pestaña permite al integrador habilitar/inhabilitar una variedad de funciones adicionales relativas al control de canales de persiana.

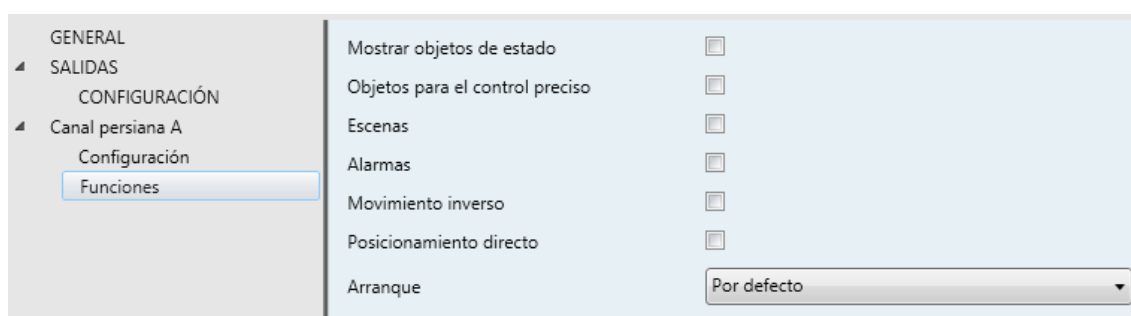


Figura 20. Canal persiana - Funciones.

- **Mostrar objetos de estado:** despliega dos objetos de un byte (“[Cx] Posición persiana (estado)” y “[Cx] Posición lamas (estado)”) que reflejarán, respectivamente, la posición instantánea de la persiana y de las lamas (si procede) en términos de porcentaje.
 - **Notificación periódica durante el movimiento [1s]:** indica si los objetos de estado se deben mandar (actualizados) cada un segundo mientras la persiana o las lamas se están moviendo, o sólo al finalizar el movimiento.
- **Objetos para el control preciso:** despliega dos objetos de un byte (“[Cx] Posicionar persiana” y “[Cx] Posicionar lamas”) que aceptarán órdenes de posición precisa (en términos de porcentaje) para la persiana y las lamas, respectivamente, desde el bus KNX.
- **Escenas:** activa o desactiva la función de Escenas, que deberá configurarse desde la pantalla de parámetros específica que muestra la Figura 21.

The screenshot shows the configuration interface for 'Canal persiana A' - 'Escenas'. On the left is a sidebar menu with the following items: GENERAL, SALIDAS, CONFIGURACIÓN, Canal persiana A, Configuración, Funciones, and Escenas (highlighted). The main area contains three configuration fields: 'Número de escena activas' with a value of 1, 'Número de escena (0=Deshabilitado)' with a value of 0, and 'Acción' with a dropdown menu set to 'Arriba'.

Figura 21. Canal persiana - Escenas.

Se pueden configurar hasta cinco escenas. El número de escenas se debe introducir en “**Número de escenas activas**” (“0”, por defecto).

Nota: después de escribir el valor, es necesario mover el foco del cursor a otra parte de la ventana para que se actualice y aparezcan los parámetros requeridos.

Dependiendo del número anterior, los dos siguientes parámetros aparecen varias veces (una por cada escena):

- **Número de escena:** establece el número de escena deseado, de modo que cuando se recibe ese valor (disminuido en uno, de acuerdo al estándar KNX) a través del objeto “[Persianas] Escenas”, se activará la acción configurada a continuación.
- **Acción:** “Arriba” (por defecto), “Abajo” o “Posición específica”, respectivamente en función de si la persiana debe moverse arriba, abajo o a una posición intermedia. La última opción permite configurar independientemente una posición para la persiana en sí o para las lamas, o bien para ambas.
- **Alarmas:** activa o desactiva la función de Alarmas, que deberá configurarse desde una pantalla de parámetros específica, como muestra la Figura 22.

Mientras que los parámetros de la alarma número 1 se muestran por defecto, los de la alarma número 2 sólo se muestran después de marcar la correspondiente casilla.

Figura 22. Canal persiana - Alarmas.

- **Disparador:** establece el valor (“1” –por defecto– o “0”) que, al recibirse desde el bus KNX a través del objeto “[Cx] Alarma”, será interpretado por el actuador como un desencadenante de la alarma, por lo que se iniciará la acción configurada debajo.
- **Periodo de monitorización cíclica:** establece cada cuánto tiempo, como máximo, se debe actualizar el objeto de alarma desde el bus después de que haya tenido lugar una primera recepción. Si se excede, y por razones de seguridad, la acción de alarma también se activará. Si este parámetro se deja a cero (opción por defecto), la función de monitorización cíclica permanecerá inhabilitada. Los valores permitidos son 5 a 600 décimas de segundo; 1 a 3600 segundos; 1 a 1440 minutos; 1 a 24 horas.
- **Acción:** “Parar” (por defecto), “Arriba”, “Abajo” o “Posición específica”. Seleccionando la última opción aparecen más parámetros:
 - “Posición de la persiana” y, en su caso, “Posición de las lamas”: definen, en términos de porcentaje, la posición específica en la que se deben situar la persiana y las lamas cuando se activa la alarma.
- **Desactivación – Modo:** “Normal” (por defecto) o “Con enclavamiento (requiere desenclavar)”. La segunda opción habilita un nuevo objeto de un bit, “[Cx] Desenclavar alarma”, que debe usarse para desenclavar la alarma externamente una vez que se ha recibido el valor de no alarma a

través del objeto “[Cx] Alarma” (es decir, el valor inverso del de activación).

Nota: el desenclavamiento debe enviarse necesariamente una vez el objeto “[Sx] Alarma” haya adquirido el valor de no alarma. Si se envía mientras que el valor de este objeto sea el de alarma no tendrá efecto.

- **Desactivación – Acción:** establece el estado que debe adquirir la salida una vez se ha desactivado la alarma (y desenclavado, si se requiere). Puede ser: “Posición actual” (por defecto), “Arriba”, “Abajo” o “Última posición (antes de alarma)”.
- **Movimiento inverso:** despliega el objeto de comunicación “[Cx] Movimiento inverso”, que es equivalente a “[Cx] Mover” salvo por que un “0” moverá la persiana hacia abajo y un “1” la moverá hacia arriba.
- **Posicionamiento directo:** activa o desactiva la función de Posicionamiento directo, que debe ser configurada desde una pantalla de parámetros específica.

Figura 23. Canal persiana - Posicionamiento directo.

Mientras que los parámetros del posicionamiento directo número 1 se muestran por defecto, los del número 2 sólo se muestran después de marcar la correspondiente casilla. Estos parámetros son:

- **Posición de la persiana:** establece la posición (en términos de porcentaje) a la que deberá moverse la persiana cuando se reciba un “1” a través de “[Cx] Posicionamiento directo” (o a través de “[Cx] Posicionamiento directo 2” para el número 2).

- **Posición de las lamas** (sólo para canales de persiana con lamas): análogo al parámetro anterior, pero para las lamas.

Independientemente del número de funciones de posicionamiento directo habilitadas (una o dos), se mostrará también el siguiente parámetro:

- **Guardar nuevas posiciones:** si está habilitado, se añadirá al proyecto un nuevo objeto de un bit: “[Cx] Posicionamiento directo (guardar)” (y “[Cx] Posicionamiento directo 2 (guardar)” si se habilita el posicionamiento directo número 2). Cuando recibe un “1” a través del bus KNX, se sobrescribe con la posición actual de la persiana (y de las lamas, si procede) la posición directa configurada. Las próximas llamadas a las funciones de posicionamiento directo llevarán a la persiana/lamas a la nueva posición.
- **Arranque:** establece si se lleva a cabo la acción por defecto (“Por defecto”) o una acción personalizada (“Personalizado”) durante la puesta en marcha del dispositivo. La segunda opción debe ser configurada desde una pantalla de parámetros específica:

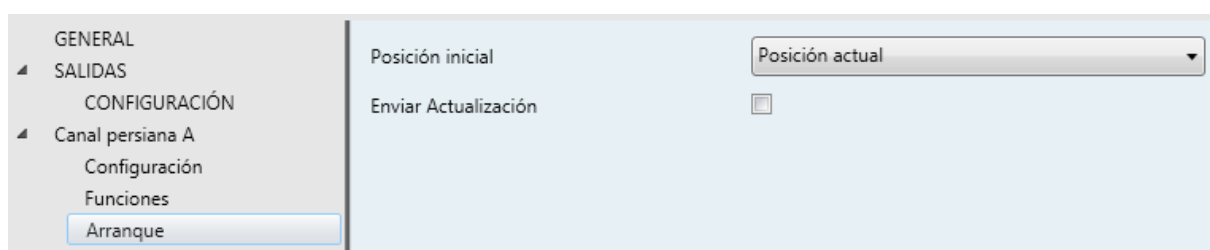


Figura 24. Canal persiana - Arranque.

Esta pantalla contiene los siguientes parámetros:

- **Posición inicial:** establece la posición que debe adquirir la persiana al inicializarse el actuador. Puede ser: “Posición actual” (es decir, dejarla como está; opción por defecto), “Arriba”, “Abajo” o “Posición específica”. Seleccionando la última opción aparecen más parámetros:
- “**Posición de la persiana**” y, en su caso, “**Posición de las lamas**”: definen, en términos de porcentaje, la posición específica en la que se deben situar la persiana y las lamas cuando el actuador se pone en marcha.

- **Enviar estado inicial:** establece si los objetos de estado de la persiana se deben enviar al bus KNX (con el fin de actualizar otros dispositivos KNX) después de la puesta en marcha del actuador. Es posible imponer un retardo (0 a 600 décimas de segundo; 0 a 3600 segundos; 0 a 1440 minutos; 0 a 24 horas) para asegurar que este valor se envía una vez que los dispositivos están listos para recibirlo.

2.3.3 CONTROL MANUAL

El MAXinBOX 66 permite controlar manualmente el estado de sus relés de salida mediante los pulsadores situados en la cara superior del dispositivo. Así pues, cada una de las salidas dispone de un pulsador asociado.

Este control manual puede ejercerse de dos modos diferentes, denominados **Test On** (destinado al testeo de la instalación durante la configuración del dispositivo) y **Test Off** (destinado al uso en cualquier otro momento). Desde ETS se podrá configurar si el control manual estará disponible y, en tal caso, cuál(es) de los dos modos estará(n) permitido(s). Así mismo, podrá habilitarse por parámetro un objeto binario destinado a bloquear o desbloquear el control manual en tiempo de ejecución.

Nota:

- *El modo **Test Off** (salvo que se haya inhabilitado por parámetro) se encuentra disponible en todo momento sin necesidad de activación específica tras descarga o reinicio.*
- *Por el contrario, para acceder al modo **Test On** (salvo, igualmente, que se haya inhabilitado por parámetro), será necesario mantener presionado el pulsador de Prog/Test durante tres segundos, hasta que el LED se vuelva amarillo. En ese momento, al soltar el pulsador, el LED adquiere el color verde para indicar que el modo Test Off ha dejado paso al modo Test On. Una nueva pulsación hará que el LED pase de nuevo a amarillo y después se apague (tras soltar el botón). De esta forma, el dispositivo abandonará el modo Test On. Téngase en cuenta que también se abandonará este modo si tiene lugar un fallo de bus.*

Modo Test Off

Mientras el control de las salidas del dispositivo se encuentra en este modo, éstas pueden controlarse no sólo mediante las órdenes enviadas a través de los objetos de comunicación, sino también utilizando los pulsadores físicos situados en el dispositivo.

Al presionar cualquiera de ellos, se actúa directamente sobre la salida como si se hubiese recibido una orden a través del objeto de comunicación correspondiente, dependiendo de la configuración de las salidas (salida individual o canal de persiana):

- **Salida individual:** una pulsación (corta o larga) hará que el dispositivo conmute el estado de la correspondiente salida, el cual se envía a través del objeto de estado asociado, si está habilitado.
- **Canal de persiana:** al presionar el pulsador, el dispositivo actuará sobre la salida según el tipo de pulsación y el estado actual:
 - Una **pulsación larga** hace que la persiana comience a moverse (hacia arriba o hacia abajo, dependiendo de cuál de los dos pulsadores del canal se haya presionado). El LED permanecerá en verde hasta el final del movimiento. Si el botón se pulsa estando la persiana ya en una posición final (arriba o abajo), no ocurrirá nada y el LED no se iluminará.
 - Una **pulsación corta** detendrá la persiana (en el caso de que se estuviese moviendo), tal y como habría sucedido si se hubiera recibido una orden de parar/paso desde el bus KNX. En el caso de que la persiana esté en reposo, la pulsación no desencadena ninguna acción, salvo que la persiana disponga de lamas orientables, en cuyo caso se provoca un movimiento de paso (arriba o abajo, dependiendo del botón pulsado). Los objetos de estado se enviarán al bus cuando corresponda.
- **Salida deshabilitada:** en el modo Test Off se ignorará cualquier pulsación sobre los botones de las salidas deshabilitadas por parámetro.

En cuanto a las funciones de bloqueo, temporizaciones, alarmas y escenas, el comportamiento del dispositivo durante el modo Test Off es el habitual. Las pulsaciones sobre los botones son totalmente equivalentes a la recepción desde el bus KNX de las órdenes de control análogas.

Modo Test On

Una vez activado el modo Test On, las salidas sólo se pueden controlar mediante la acción directa sobre los pulsadores de control. Todas las órdenes que lleguen a través de objetos de comunicación se ignorarán, independientemente del canal o la salida a la que vayan dirigidas.

Dependiendo de si una salida se encuentra configurada como salida individual o bien forma parte de un canal de persiana, el comportamiento ante una pulsación sobre el control manual provocará diferentes reacciones:

- **Salida individual:** una pulsación corta o larga sobre el botón correspondiente provocará una conmutación en el relé.
- **Canal de persiana:** una pulsación sobre el botón correspondiente pondrá en movimiento el motor de la persiana (arriba o abajo dependiendo del botón), hasta el momento en que cese la pulsación, ignorándose en todo caso la posición de la persiana y los tiempos de subida y bajada parametrizados.
- **Salida deshabilitada:** las pulsaciones cortas o largas tendrán en el modo Test On el mismo efecto para salidas deshabilitadas que las salidas individuales (es decir, el relé conmutará su estado con cada pulsación).

Las funciones de alarma, bloqueo y temporización, así como cualquier orden enviada desde el bus KNX hacia el actuador no afectarán a las salidas mientras el modo Test ON esté activo. Tampoco se enviarán objetos de estado.

Importante: en el estado de fábrica, el dispositivo se entrega con todas las salidas configuradas como canales de persiana (sin lamas) y con ambos modos de control manual (modos Test OFF y Test ON) habilitados.

PARAMETRIZACIÓN ETS

El **control manual** se configura desde la pestaña de Configuración, dentro de Salidas.

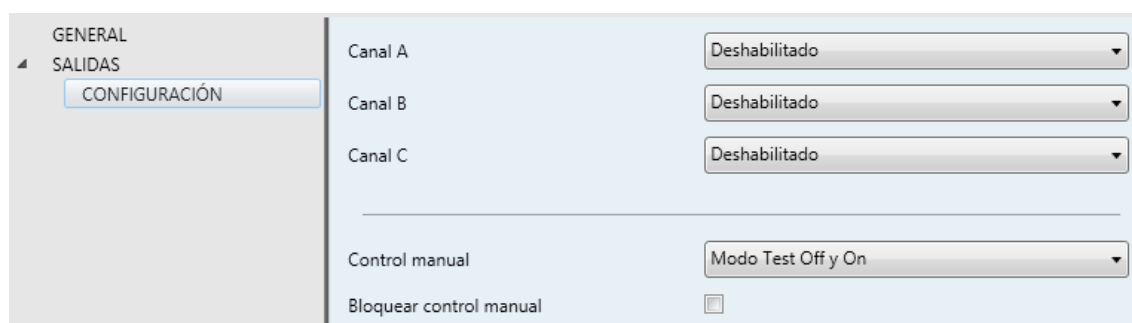


Figura 25. Control manual.

Los dos únicos parámetros son:

- **Control manual:** las opciones son “Deshabilitado”, “Sólo Modo Test Off”, “Sólo Modo Test On” y “Modo Test Off y On” (por defecto). Dependiendo de la seleccionada, el dispositivo permitirá usar el control manual en modo Test Off, en modo Test On o en ambos. Téngase en cuenta que, como ya se ha mencionado, para usar el modo Test Off no es necesaria ninguna acción

adicional, mientras que para cambiar al modo Test On es necesaria una pulsación larga en el botón de Prog/Test.

- **Bloquear control manual:** a menos que el parámetro anterior esté deshabilitado, el parámetro de bloqueo del control manual ofrece un procedimiento opcional para bloquear el control manual en tiempo de ejecución. Para ello, cuando se habilita esta casilla aparece el objeto **“Bloqueo de control manual”**, así como dos nuevos parámetros:
 - **Valor:** define si el bloqueo/desbloqueo del control manual debe tener lugar cuando se reciben los valores “0” y “1” respectivamente o viceversa.
 - **Inicialización:** especifica cómo debe permanecer el control manual tras la inicialización del dispositivo (tras descarga de ETS o fallo de bus): “Desbloqueado”, “Bloqueado” o “Último valor” (por defecto; en la primera inicialización se corresponderá con Desbloqueado).

2.4 FUNCIONES LÓGICAS

Este módulo permite la ejecución de operaciones numéricas o en lógica binaria con datos procedentes del bus KNX y enviar el resultado a través de objetos de comunicación específicamente habilitados a tal efecto en el actuador.

En el MAXinBOX 66 pueden implementarse **hasta 10 funciones lógicas diferentes e independientes entre sí**, completamente personalizables, que consisten en **un máximo 5 operaciones consecutivas para cada una**.

La ejecución de cada función puede depender de una **condición** configurable, que será evaluada cada vez que **active** la función a través de objetos de comunicación específicos y parametrizables. El resultado tras la ejecución de las operaciones de la función puede ser también evaluado de acuerdo a ciertas **condiciones** y después enviarse (o no) al bus KNX, todo lo cual podrá hacerse cada vez que la función se ejecute, periódicamente o sólo cuando el resultado difiera del anterior.

Consúltese el documento específico “**Módulo de funciones lógicas para MAXinBOX 66**” (disponible en la página web de Zennio: www.zennio.com) para obtener información detallada sobre el uso de las funciones lógicas y su parametrización en ETS.

2.5 TERMOSTATOS

MAXinBOX 66 implementa **cuatro termostatos Zennio** que pueden habilitarse y configurarse independientemente.

Consúltase el documento específico “**Termostato Zennio para MAXinBOX 66**” (disponible en la página web de Zennio: www.zennio.com) para obtener información detallada acerca de la funcionalidad y la configuración de los parámetros relacionados.

2.6 CONTROL MAESTRO DE ILUMINACIÓN

La función del control de maestro de iluminación ofrece la opción de controlar el estado de hasta 12 fuentes de luz –o cualquier otro elemento funcionalmente similar cuyo estado se transmita a través de un objeto binario– y, en función de estos estados, llevar a cabo una **orden maestra** cada vez que se reciba una cierta señal de disparo (de nuevo, un valor binario) a través de un objeto específico.

Esta orden maestra consistirá en:

- Una orden de **apagado general**, si al menos uno de los hasta doce objetos de estado se encuentra encendido.
- Una orden de **encendido de cortesía**, si ninguno de los hasta doce objetos de estado se encuentra encendido.

Téngase en cuenta que las órdenes de apagado y encendido anteriores no son necesariamente un valor binario a enviar al bus; el integrador puede decidir qué deberá enviarse al bus KNX en ambos casos: una orden de persiana, una consigna de termostato (o una orden de cambio de modo), un valor constante, una escena... Sólo el objeto de disparo y los doce objetos de estado deben necesariamente ser binarios.

El escenario más común para este control de maestro de iluminación podría ser una habitación de hotel con un pulsador maestro junto a la puerta. Al abandonar la habitación, el huésped tendrá la posibilidad de pulsar el botón maestro y hacer que todas las luces se apaguen juntas. Después, de vuelta a la habitación y con todas las luces apagadas, pulsando sobre el mismo botón sólo se encenderá una luz en particular (por ejemplo, la lámpara más cercana a la puerta). Esto es el encendido de cortesía.

PARAMETRIZACIÓN ETS

Al habilitar la función de Control de maestro de iluminación se incluirá una pestaña específica en el menú de la izquierda. Esta nueva pantalla de parámetros contiene las siguientes opciones:

- **Número de objetos de estado:** define el número de objetos de estado de un bit requeridos. El mínimo valor (por defecto) es “1” y el máximo es “12”. Estos objetos se llaman “[CMI] Objeto de estado *n*”.

- **Valor de disparo:** establece el valor (“0”, “1” o “0/1”, siendo este último el predeterminado) que activará, cuando se reciba a través de “[CMI] Disparo”, la acción correspondiente (apagado general o encendido de cortesía).

- **Apagado general.**

- **Retardo:** define un cierto retardo (que comienza una vez se ha recibido el disparo) antes de la ejecución del apagado general. El rango permitido es de 0 a 255 segundos.
- **Objeto binario:** si se habilita, aparece el objeto “[CMI] Apagado general: objeto binario”, que envía un “0” cuando se produce un apagado general.
- **Objeto de porcentaje:** si se habilita, aparece el objeto “[CMI] Apagado general: porcentaje”, que enviará un valor de porcentaje (configurable en “Valor”) cada vez que se produce el apagado general.
- **Escena:** si se habilita, aparece el objeto “[CMI] Apagado general: escena”, que enviará una orden de ejecutar/grabar escena (configurable en “Acción” y “Número de escena”) cada vez que se produce el apagado general.
- **Modo especial:** si se habilita, aparece el objeto “[CMI] Apagado general: modo especial”, que enviará un modo de termostato HVAC (configurable en “Valor”, siendo las opciones disponibles “Auto”, “Confort”, “Standby”, “Económico” y “Protección”) cada vez que se produce el apagado general.

Nota: las opciones anteriores no son mutuamente excluyentes; es posible mandar valores de diferente tipología al mismo tiempo.

- **Encendido de cortesía:**

Los parámetros disponibles aquí son completamente análogos a los relativos al apagado general. Sin embargo, en este caso los nombres de los objetos empiezan con “[CMI] Encendido cortesía (...)”. Por otro lado, no es posible enviar órdenes de salvado de escenas en el encendido de cortesía (sólo se permiten órdenes de ejecución de escenas).

Nota: el objeto “[CMI] Encendido de cortesía: objeto binario” envía el valor “1” (al tener lugar el encendido de cortesía), mientras que “[CMI] Apagado

general: objeto binario” enviará el valor “0” (cuando se produce el apagado general, como se explicó anteriormente).

GENERAL

CONTROL MAESTRO DE ILUMINACIÓN

CONFIGURACIÓN

Número de objetos de estado: 1

Valor de disparo: 0/1

Apagado general

Retardo: 0 x 1 s.

Objeto binario: ☒

Objeto de porcentaje: ☐

Escena: ☐

Modo especial: ☐

Encendido de cortesía

Retardo: 0 x 1 s.

Objeto binario: ☒

Objeto de porcentaje: ☐

Escena: ☐

Modo especial: ☐

Figura 26. Control de maestro de iluminación.

ANEXO I. OBJETOS DE COMUNICACIÓN

- **“Rango funcional”** muestra los valores que, independientemente de los permitidos por el bus dado el tamaño del objeto, tienen utilidad o un significado específico, porque así lo establezcan o restrinjan el estándar KNX o el propio programa de aplicación.
- **“1ª vez”** muestra el valor por defecto que es asignado por el programa de aplicación a cada objeto tras una descarga o un reinicio completo desde ETS (se muestra un **guion** cuando por defecto el objeto no se inicializa expresamente, por lo que puede suponerse a cero o al valor que corresponda en caso de depender de elementos externos: sensores, etc.). Si existe algún parámetro para personalizar el valor del objeto tras descarga o reinicio, se mostrará el símbolo ✓ en la columna **“P”** contigua. Por último, si el objeto se envía al bus (bien como escritura o como petición de lectura) tras descarga o reinicio desde ETS o bien existe un parámetro para que lo haga, se indicará como **W** (envío del valor al bus) o como **R** (petición de lectura).
- **“Fallo”** muestra el valor por defecto que es asignado por el programa de aplicación a cada objeto a la vuelta de un fallo de bus. Se muestra un **guion** cuando el objeto no se inicializa expresamente, por lo que puede suponerse que recupera el valor anterior al fallo de bus o bien que adquiere el valor que corresponda en el caso de depender de elementos externos: sensores, etc. Si existe algún parámetro para personalizar el valor del objeto tras fallo de bus, se mostrará el símbolo ✓ en la columna **“P”** contigua. Por último, si el objeto se envía al bus (bien como escritura o como petición de lectura) tras fallo de bus o bien existe un parámetro para que lo haga, se indicará como **W** (envío del valor al bus) o como **R** (petición de lectura).

Número	Tamaño	E/S	Banderas	Tipo de dato (DPT)	Rango funcional	1ª vez	P	Fallo	P	Nombre	Función
1	1 Bit		C T - - -	DPT_Trigger	0/1	0 W		0 W		Reset 0	Vuelve la tensión -> Envía 0
2	1 Bit		C T - - -	DPT_Trigger	0/1	1 W		1 W		Reset 1	Vuelve la tensión -> Envía 1
3	1 Bit	E	C - - W -	DPT_Switch	0/1	-		-		Bloqueo de Control Manual	0 = Bloquear; 1 = Desbloquear
	1 Bit	E	C - - W -	DPT_Switch	0/1	-		-		Bloqueo de Control Manual	0 = Desbloquear; 1 = Bloquear
4	1 Byte	E	C - - W -	DPT_SceneControl	0-63; 128-191	-		-		[Termostato] Escenas: entrada	Valor de escena
5, 35, 65, 95	2 Bytes	E	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	25,00		-		[Tx] Fuente de Temperatura 1	Temperatura de sensor externo
6, 36, 66, 96	2 Bytes	E	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	25,00		-		[Tx] Fuente de Temperatura 2	Temperatura de sensor externo
7, 37, 67, 97	2 Bytes	S	C T R - -	DPT_Value_Temp	-273,00 - 670760,00	-		-		[Tx] Temperatura Efectiva	Temperatura efectiva de control
8,38, 68, 98	1 Byte	E	C - - W -	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	-	✓	-		[Tx] Modo Especial	Valor de modo de 1 byte
9, 39, 69, 99	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	✓	-		[Tx] Modo Especial: confort	0 = Nada; 1 = Disparo
	1 Bit	E	C - - W -	DPT_Switch	0/1	-	✓	-		[Tx] Modo Especial: confort	0 = Apagado; 1 = Encendido
10, 40, 70, 100	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	✓	-		[Tx] Modo Especial: standby	0 = Nada; 1 = Disparo
	1 Bit	E	C - - W -	DPT_Switch	0/1	-	✓	-		[Tx] Modo Especial: standby	0 = Apagado; 1 = Encendido
11, 41, 71, 101	1 Bit	E	C - - W -	DPT_Trigger	0/1	1	✓	-		[Tx] Modo Especial: económico	0 = Nada; 1 = Disparo
	1 Bit	E	C - - W -	DPT_Switch	0/1	1	✓	-		[Tx] Modo Especial: económico	0 = Apagado; 1 = Encendido
12, 42, 72, 102	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	✓	-		[Tx] Modo Especial: protección	0 = Nada; 1 = Disparo
	1 Bit	E	C - - W -	DPT_Switch	0/1	-	✓	-		[Tx] Modo Especial: protección	0 = Apagado; 1 = Encendido

13, 43, 73, 103	1 Bit	E	C - - W -	DPT_Window_Door	0/1	-	-	[Tx] Estado de Ventana (entrada)	0 = Cerrado; 1 = Abierto
14, 44, 74, 104	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[Tx] Prolongación de Confort	0 = Nada; 1 = Confort Temporizado
15, 45, 75, 105	1 Byte	S	C T R - -	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	3 W	✓ - W	[Tx] Modo Especial (Estado)	Valor de modo de 1 byte
16, 46, 76, 106	2 Bytes	E	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	-	-	[Tx] Consigna	Consigna del termostato
	2 Bytes	E	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	-	-	[Tx] Consigna Básica	Consigna de referencia
17, 47, 77, 107	1 Bit	E	C - - W -	DPT_Step	0/1	-	-	[Tx] Consigna (Paso)	0 = -0.5°C; 1 = +0.5°C
18, 48, 78, 108	2 Bytes	E	C - - W -	DPT_Value_Tempd	-670760,00 - 670760,00	-	-	[Tx] Consigna (Offset)	Valor de coma flotante
19, 49, 79, 109	2 Bytes	S	C T R - -	DPT_Value_Temp	-273,00 - 670760,00	- W	✓ - W	[Tx] Consigna (Estado)	Consigna actual
20, 50, 80, 110	2 Bytes	S	C T R - -	DPT_Value_Temp	-273,00 - 670760,00	- W	✓ - W	[Tx] Consigna Básica (Estado)	Consigna básica actual
21, 51, 81, 111	2 Bytes	S	C T R - -	DPT_Value_Tempd	-670760,00 - 670760,00	- W	✓ - W	[Tx] Consigna (Estado de Offset)	Valor actual del offset
22, 52, 82, 112	1 Bit	E	C - - W -	DPT_Reset	0/1	-	-	[Tx] Reinicio de Consigna	Reinicio a valores por defecto
	1 Bit	E	C - - W -	DPT_Reset	0/1	-	-	[Tx] Reiniciar Offsets	Reiniciar offset
23, 53, 83, 113	1 Bit	E	C - - W -	DPT_Heat_Cool	0/1	-	-	[Tx] Modo	0 = Enfriar; 1 = Calentar
24, 54, 84, 114	1 Bit	S	C T R - -	DPT_Heat_Cool	0/1	- W	✓ - W	[Tx] Modo (Estado)	0 = Enfriar; 1 = Calentar
25, 55, 85, 115	1 Bit	E	C - - W -	DPT_Switch	0/1	-	-	[Tx] On/Off	0 = Apagado; 1 = Encendido
26, 56, 86, 116	1 Bit	S	C T R - -	DPT_Switch	0/1	- W	✓ - W	✓ [Tx] On/Off (Estado)	0 = Apagado; 1 = Encendido
27, 57, 87, 117	1 Byte	S	C T R - -	DPT_Scaling	0% - 100%	-	-	[Tx] Variable de Control (Enfriar)	Control PI (Continuo)
28, 58, 88, 118	1 Byte	S	C T R - -	DPT_Scaling	0% - 100%	-	-	[Tx] Variable de Control (Calentar)	Control PI (Continuo)
29, 59, 89, 119	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Variable de Control (Enfriar)	Control de 2 puntos
	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Variable de Control (Enfriar)	Control PI (PWM)
30, 60, 90, 120	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Variable de Control (Calentar)	Control de 2 puntos
	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Variable de Control (Calentar)	Control PI (PWM)
31, 61, 91, 121	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Frío Adicional	Temp >= (Consigna+Banda)=> "1"
32, 62, 92, 122	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Calor Adicional	Temp <= (Consigna-Banda)=> "1"
33, 63, 93,	1 Bit	S	C T R - -	DPT_Switch	0/1	-	-	[Tx] Estado de PI (Enfriar)	0 = Señal PI a 0%; 1 = Señal PI mayor

123									que 0%
34, 64, 94, 124	1 Bit	S	CTR--	DPT_Switch	0/1	-	-	[Tx] Estado de PI (Calentar)	0 = Señal PI a 0%; 1 = Señal PI mayor que 0%
125, 129, 133, 137, 141, 145	2 Bytes	S	CTR--	DPT_Value_Temp	-273,00 - 670760,00	-	-	[Ex] Temperatura actual	Valor del sensor de temperatura
126, 130, 134, 138, 142, 146	1 Bit	S	CTR--	DPT_Alarm	0/1	-	-	[Ex] Sobreenfriamiento	0 = No alarma; 1 = Alarma
127, 131, 135, 139, 143, 147	1 Bit	S	CTR--	DPT_Alarm	0/1	-	-	[Ex] Sobrecalentamiento	0 = No alarma; 1 = Alarma
128, 132, 136, 140, 144, 148	1 Bit	S	CTR--	DPT_Alarm	0/1	-	-	[Ex] Error de sonda	0 = No alarma; 1 = Alarma
149	1 Byte	E	C--W-	DPT_SceneControl	0-63; 128-191	-	-	[Detec. Mov.] Escenas: entrada	Valor de escena
150	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	-	-	[Detec. Mov.] Escenas: salida	Valor de escena
151, 175, 199, 223, 247, 271	1 Byte	S	CTR--	DPT_Scaling	0% - 100%	-	-	[Ex] Luminosidad	0-100%
152, 176, 200, 224, 248, 272	1 Bit	S	CTR--	DPT_Alarm	0/1	-	-	[Ex] Error de circuito abierto	0 = No error; 1 = Circuito abierto
153, 177, 201, 225, 249, 273	1 Bit	S	CTR--	DPT_Alarm	0/1	-	-	[Ex] Error de cortocircuito	0 = No error; 1 = Cortocircuito
154, 178, 202, 226, 250, 274	1 Byte	S	CTR--	DPT_Scaling	0% - 100%	-	-	[Ex] Estado de presencia (Porcentaje)	0-100%
155, 179, 203, 227, 251, 275	1 Byte	S	CTR--	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	-	-	[Ex] Estado de presencia (HVAC)	Auto, Confort, Standby, Económico, Protección
156, 180, 204, 228, 252, 278	1 Bit	S	CTR--	DPT_Occupancy	0/1	-	-	[Ex] Estado de presencia (Binario)	Valor binario
	1 Bit	S	CTR--	DPT_Trigger	0/1	-	-	[Ex] Detector de presencia: salida de esclavo	1 = Movimiento detectado
157, 181, 205, 229, 253, 279	1 Bit	E	C--W-	DPT_Trigger	0/1	-	-	[Ex] Disparador de detección de presencia	Valor binario para disparar la detección de presencia
158, 182, 206, 230, 254, 280	1 Bit	E	C--W-	DPT_Trigger	0/1	-	-	[Ex] Detección de presencia: Entrada de esclavo	0 = Nada; 1 = Detección desde dispositivo esclavo
159, 183, 207, 231, 255, 281	1 Bit	E	C--W-	DPT_Trigger	0/1	-	-	[Ex] Detección de movimiento externa	0 = Nada; 1 = Detección de un sensor externo

160, 165, 170, 184, 189, 194, 208, 213, 218, 232, 237, 242, 256, 261, 266, 280, 285, 290	1 Byte	S	CTR--	DPT_Scaling	0% - 100%	-	-	[Ex][Cy] Estado de detección (Porcentaje)	0-100%
161, 166, 171, 185, 190, 195, 209, 214, 219, 233, 238, 243, 257, 262, 267, 281, 286, 291	1 Byte	S	CTR--	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	-	-	[Ex][Cy] Estado de detección (HVAC)	Auto, Confort, Standby, Económico, Protección
162, 167, 172, 186, 191, 196, 210, 215, 220, 234, 239, 244, 258, 263, 268, 282, 287, 292	1 Bit	S	CTR--	DPT_Switch	0/1	-	-	[Ex][Cy] Estado de detección (Binario)	Valor binario
163, 168, 173, 187, 192, 197, 211, 216, 221, 235, 240, 245, 259, 264, 269, 283, 288, 293	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[Ex][Cy] Bloqueo	Dependiente de los parámetros
164, 169, 174, 188, 193, 198, 212, 217, 222, 236, 241, 246, 260, 265, 270, 284, 289, 294	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[Ex][Cy] Forzar estado	0 = No detección; 1 = Detección
295	1 Byte	E	C--W-	DPT_SceneControl	0-63; 128-191	-	-	[Persianas] Escenas	0 - 63 (Ejecutar 1 - 64); 128 - 191 (Guardar 1 - 64)

296, 311, 326	1 Bit	E	C - - W -	DPT_UpDown	0/1	-	-	[C] <input type="checkbox"/> Mover	0=Subir; 1=Bajar
297, 312, 327	1 Bit	E	C - - W -	DPT_Step	0/1	-	-	[C] <input type="checkbox"/> Parar/Paso	0=Parar/Paso arriba; 1=Parar/Paso abajo
	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/> Parar	0=Parar; 1=Parar
298, 313, 328	1 Bit	E	C - - W -	DPT_Enable	0/1	-	-	[C] <input type="checkbox"/> Bloquear	0=Desbloquear; 1=Bloquear
299, 314, 329	1 Byte	S	C T R - -	DPT_Scaling	0% - 100%	- W	✓ - W	[C] <input type="checkbox"/> Posición	0%=Arriba; 100%=Abajo
300, 315, 330	1 Byte	S	C T R - -	DPT_Scaling	0% - 100%	- W	✓ - W	[C] <input type="checkbox"/> Estado posición lamas	0%=Abiertas; 100%=Cerradas
301, 316, 331	1 Byte	E	C - - W -	DPT_Scaling	0% - 100%	-	-	[C] <input type="checkbox"/> Posicionar per	0%=Arriba; 100%=Abajo
302, 317, 332	1 Byte	E	C - - W -	DPT_Scaling	0% - 100%	-	-	[C] <input type="checkbox"/> Posicionar lamas	0%=Abiertas; 100%=Cerradas
303, 318, 333	1 Bit	E	C - - W -	DPT_Alarm	0/1	-	-	[C] <input type="checkbox"/> Alarma	0=Sin alarma; 1=Alarma
	1 Bit	E	C - - W -	DPT_Alarm	0/1	1	-	[C] <input type="checkbox"/> Alarma	0=Alarma; 1=Sin alarma
304, 319, 334	1 Bit	E	C - - W -	DPT_Alarm	0/1	-	-	[C] <input type="checkbox"/> Alarma 2	0=Sin alarma; 1=Alarma
	1 Bit	E	C - - W -	DPT_Alarm	0/1	1	-	[C] <input type="checkbox"/> Alarma 2	0=Alarma; 1=Sin alarma
305, 320, 335	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/> Desenclavar alarma	Alarma=0 + Desenclavar=1 => Fin de alarma
306, 321, 336	1 Bit	E	C - - W -	DPT_UpDown	0/1	-	-	[C] <input type="checkbox"/> Tiempo inverso	0=Bajar; 1=Subir
307, 322, 337	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/> Posiciona	0=Ignorado; 1=Ir a posición
308, 323, 338	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/> Posiciona	0=Ignorado; 1=Ir a posición
309, 324, 339	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/>	0=Ignorado; 1=Guardar posición actual
310, 325, 340	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[C] <input type="checkbox"/>	0=Ignorado; 1=Guardar posición actual
341	1 Byte	E	C - - W -	DPT_SceneControl	0-63; 128-191	-	-	[Salidas] Escenas	0 - 63 (Ejecutar 1 - 64); 128 - 191 (Salvar 1 - 64)
342, 350, 358, 366, 374, 382	1 Bit	E	C - - W -	DPT_BinaryValue	0/1	-	-	[Sx] Encender/Apagar	N.A. (0=Abrir relé; 1=Cerrar relé)
	1 Bit	E	C - - W -	DPT_BinaryValue	0/1	-	-	[Sx] Encender/Apagar	N.C. (0=Cerrar relé; 1=Abrir relé)
343, 351, 359, 367, 375, 383	1 Bit	S	C T R - -	DPT_BinaryValue	0/1	- W	✓ - W	[Sx] Encendido/Apagado (Estado)	0=Salida apagada; 1=Salida encendida
344, 352, 360, 368, 376, 384	1 Bit	E	C - - W -	DPT_Enable	0/1	-	-	[Sx] Bloquear	0=Desbloquear; 1=Bloquear
345, 353, 361, 369, 377, 385	1 Bit	E	C - - W -	DPT_Start	0/1	-	-	[Sx] Temporización	0=Apagar; 1=Encender

346, 354, 362, 370, 378, 386	1 Bit	E	C - - W -	DPT_Start	0/1	-	-	[Sx] Intermitencia	0=Parar; 1=Comenzar
347, 355, 363, 371, 379, 387	1 Bit	E	C - - W -	DPT_Alarm	0/1	-	-	[Sx] Alarma	0=Normal; 1=Alarma
	1 Bit	E	C - - W -	DPT_Alarm	0/1	-	-	[Sx] Alarma	0=Alarma; 1=Normal
348, 356, 364, 372, 380, 388	1 Bit	E	C - - W -	DPT_Trigger	0/1	-	-	[Sx] Desendclavar alarma	Alarma=0 + Desendclavar=1 => Fin de alarma
349, 357, 365, 373, 381, 389	1 Bit	S	C T R - -	DPT_Trigger	0/1	-	-	[Sx] Tiempo de aviso (Estado)	0=Normal; 1=Aviso
390, 396, 402, 408, 414, 420	1 Bit	E	C - - W -	DPT_Switch	0/1			[Ex] Bloquear Entrada	1 = Bloqueada; 0 = Desbloqueada
391, 397, 403, 409, 415, 421	1 Bit		C T - - -	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] 0	Envío de 0
	1 Bit		C T - - -	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] 1	Envío de 1
	1 Bit	E	C T - W -	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] Conmutar 0/1	Conmutación 0/1
	1 Bit		C T - - -	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Corta] Subir persiana	Envío de 0 (Subir)
	1 Bit		C T - - -	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Corta] Bajar persiana	Envío de 1 (Bajar)
	1 Bit		C T - - -	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Corta] Subir/Bajar persiana	Conmutación 0/1 (Subir/Bajar)
	1 Bit		C T - - -	DPT_Step	0/1	- W	- W	[Ex] [Puls. Corta] Parar persiana / paso arriba	Envío de 0 (Parar/Paso arriba)
	1 Bit		C T - - -	DPT_Step	0/1	- W	- W	[Ex] [Puls. Corta] Parar persiana / paso abajo	Envío de 1 (Parar/Paso abajo)
	1 Bit		C T - - -	DPT_Step	0/1	- W	- W	[Ex] [Puls. Corta] Parar persiana / paso conmutado	Conmutación 0/1 (Parar/Paso arriba/abajo)
	4 Bit		C T - - -	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dism. 100%) 0x2 (Dism. 50%) 0x3 (Dismin. 25%) 0x4 (Dismin. 12%) 0x5 (Dismin. 6%) 0x6 (Dismin. 3%) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) 0xA (Aumen. 50%) 0xB (Aumen. 25%) 0xC (Aumen. 12%) 0xD (Aumen. 6%) 0xE (Aumen. 3%) 0xF (Aumen. 1%)	- W	- W	[Ex] [Puls. Corta] Aumentar luz	Aumentar luz
	4 Bit		C T - - -	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dismin. 100%)	- W	- W	[Ex] [Puls. Corta] Disminuir Luz	Disminuir luz

				(...) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) (...) 0xF (Aumen. 1%)					
	4 Bit		CT----	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dismin. 100%) (...) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) (...) 0xF (Aumen. 1%)	- W	- W	[Ex] [Puls. Corta] Aumentar/Disminuir luz	Conmutación aumentar/disminuir luz
	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] Dimmer ON	Envío de 1 (ON)
	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] Dimmer OFF	Envío de 0 (OFF)
	1 Bit	E	CT-W-	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Corta] Dimmer ON/OFF	Conmutación 0/1
	1 Byte		CT----	DPT_SceneControl	0-63; 128-191	- W	- W	[Ex] [Puls. Corta] Ejecutar escena	Envío de 0-63
	1 Byte		CT----	DPT_SceneControl	0-63; 128-191	- W	- W	[Ex] [Puls. Corta] Grabar escena	Envío de 128-191
	1 Bit	E/S	CTRW-	DPT_Switch	0/1	- W	- W	[Ex] [Interruptor/Sensor] Flanco	Envío de 0 o 1
	1 Byte		CT----	DPT_Value_1_Ucount	0 - 255	- W	- W	[Ex] [Puls. Corta] Valor constante (entero)	0 - 255
	1 Byte		CT----	DPT_Scaling	0% - 100%	- W	- W	[Ex] [Puls. Corta] Valor constante (porcentaje)	0% - 100%
	2 Bytes		CT----	DPT_Value_2_Ucount	0 - 65535	- W	- W	[Ex] [Puls. Corta] Valor constante (entero)	0 - 65535
	2 Bytes		CT----	9.xxx	-671088.64 - 670760.96	- W	- W	[Ex] [Puls. Corta] Valor constante (coma flotante)	Valor en coma flotante
392, 398, 404, 410, 416, 422	1 Byte	E	C--W-	DPT_Scaling	0% - 100%	- W	- W	[Ex] [Puls. Corta] Estado de la persiana (entrada)	0% = Arriba; 100% = Abajo
	1 Byte	E	C--W-	DPT_Scaling	0% - 100%	- W	- W	[Ex] [Puls. Corta] Estado del dimmer (entrada)	0% - 100%
393, 399, 405, 411, 417, 423	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] 0	Envío de 0
	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] 1	Envío de 1
	1 Bit	E	CT-W-	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] Conmutar 0/1	Conmutación 0/1
	1 Bit		CT----	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Larga] Subir persiana	Envío de 0 (Subir)
	1 Bit		CT----	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Larga] Bajar persiana	Envío de 1 (Bajar)
	1 Bit		CT----	DPT_UpDown	0/1	- W	- W	[Ex] [Puls. Larga] Subir/Bajar persiana	Conmutación 0/1 (Subir/Bajar)
	1 Bit		CT----	DPT_Step	0/1	- W	- W	[Ex] [Puls. Larga] Parar persiana / paso arriba	Envío de 0 (Parar/Paso arriba)
	1 Bit		CT----	DPT_Step	0/1	- W	- W	[Ex] [Puls. Larga] Parar persiana / paso abajo	Envío de 1 (Parar/Paso abajo)
	1 Bit		CT----	DPT_Step	0/1	- W	- W	[Ex] [Puls. Larga] Parar persiana / paso conmutado	Conmutación 0/1 (Parar/Paso arriba/abajo)

	4 Bit		CT----	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dismin. 100%) 0x2 (Dismin. 50%) 0x3 (Dismin. 25%) 0x4 (Dismin. 12%) 0x5 (Dismin. 6%) 0x6 (Dismin. 3%) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) 0xA (Aumen. 50%) 0xB (Aumen. 25%) 0xC (Aumen. 12%) 0xD (Aumen. 6%) 0xE (Aumen. 3%) 0xF (Aumen. 1%)	- W	- W	[Ex] [Puls. Larga] Aumentar luz	Puls. Larga -> Aumentar; Soltar -> Detener regulación
	4 Bit		CT----	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dismin. 100%) (...) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) (...) 0xF (Aumen. 1%)	- W	- W	[Ex] [Puls. Larga] Disminuir Luz	Puls. Larga -> Disminuir; Soltar -> Detener regulación
	4 Bit		CT----	DPT_Control_Dimming	0x0 (Parar) 0x1 (Dismin. 100%) (...) 0x7 (Dismin. 1%) 0x8 (Parar) 0x9 (Aumen. 100%) (...) 0xF (Aumen. 1%)	- W	- W	[Ex] [Puls. Larga] Aumentar/Disminuir luz	Puls. Larga -> Aumentar/Disminuir; Soltar -> Detener regulación
	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] Dimmer ON	Envío de 1 (ON)
	1 Bit		CT----	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] Dimmer OFF	Envío de 0 (OFF)
	1 Bit	E	CT-W-	DPT_Switch	0/1	- W	- W	[Ex] [Puls. Larga] Dimmer ON/OFF	Conmutación 0/1
	1 Byte		CT----	DPT_SceneControl	0-63; 128-191	- W	- W	[Ex] [Puls. Larga] Ejecutar escena	Envío de 0-63
	1 Byte		CT----	DPT_SceneControl	0-63; 128-191	- W	- W	[Ex] [Puls. Larga] Grabar escena	Envío de 128-191
	1 Bit	S	CTR--	DPT_Alarm	0/1	- W	- W	[Ex] [Interruptor/Sensor] Alarma: avería, sabotaje, línea inestable	1 = Alarma; 0 = No Alarma
	2 Bytes		CT----	9.xxx	-671088.64 - 670760.96	- W	- W	[Ex] [Puls. Larga] Valor constante (coma flotante)	Valor en coma flotante
	2 Bytes		CT----	DPT_Value_2_Ucount	0 - 65535	- W	- W	[Ex] [Puls. Larga] Valor constante (entero)	0 - 65535
	1 Byte		CT----	DPT_Scaling	0% - 100%	- W	- W	[Ex] [Puls. Larga] Valor constante (porcentaje)	0% - 100%
	1 Byte		CT----	DPT_Value_1_Ucount	0 - 255	- W	- W	[Ex] [Puls. Larga] Valor constante	0 - 255

								(entero)	
394, 400, 406, 412, 418, 424	1 Bit		CT---	DPT_Trigger	0/1	- W	- W	[Ex] [Soltar Puls. Larga] Parar persiana	Soltar -> Parar persiana
395, 401, 407, 413, 419, 425	1 Byte	E	C--W-	DPT_Scaling	0% - 100%	50% W	- W	[Ex] [Puls. Larga] Estado del dimmer (entrada)	0% - 100%
	1 Byte	E	C--W-	DPT_Scaling	0% - 100%	50% W	- W	[Ex] [Puls. Larga] Estado de la persiana (entrada)	0% = Arriba; 100% = Abajo
426	1 Bit	E	C--W-	DPT_Trigger	0/1	-	-	[CMI] Disparo	Dispara el control maestro de iluminación
427	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 1	Estado binario
428	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 2	Estado binario
429	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 3	Estado binario
430	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 4	Estado binario
431	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 5	Estado binario
432	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 6	Estado binario
433	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 7	Estado binario
434	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 8	Estado binario
435	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 9	Estado binario
436	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 10	Estado binario
437	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 11	Estado binario
438	1 Bit	E	C--W-	DPT_Switch	0/1	-	-	[CMI] Objeto de estado 12	Estado binario
439	1 Bit		CT---	DPT_Switch	0/1	-	-	[CMI] Apagado general: objeto binario	Envío de off
440	1 Byte		CT---	DPT_Scaling	0% - 100%	-	-	[CMI] Apagado general: porcentaje	0-100%
441	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	-	-	[CMI] Apagado general: escena	Envío de escena
442	1 Byte		CT---	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	-	-	[CMI] Apagado general: modo especial	Auto, Confort, Standby, Económico, Protección
443	1 Bit		CT---	DPT_Switch	0/1	-	-	[CMI] Encendido de cortesía: objeto binario	Envío de on
444	1 Byte		CT---	DPT_Scaling	0% - 100%	-	-	[CMI] Encendido de cortesía: porcentaje	0-100%
445	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	-	-	[CMI] Encendido de cortesía: escena	Envío de escena
446	1 Byte		CT---	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	-	-	[CMI] Encendido de cortesía: modo especial	Auto, Confort, Standby, Económico, Protección
447-478	1 Bit	E	C--W-	DPT_Bool	0/1	-	-	[FL] (1 bit) Dato de entrada x	Binary Dato de entrada (0/1)
479-494	1 Byte	E	C--W-	DPT_Value_1_Ucount	0 - 255	-	-	[FL] (1 byte) Dato de entrada x	1 byte Dato de entrada (0-255)
495-510	2 Bytes	E	C--W-	DPT_Value_2_Ucount	0 - 65535	-	-	[FL] (2 bytes) Dato de entrada x	2 bytes Dato de entrada
				DPT_Value_2_Count	-32768 - 32767				
				DPT_Value_Temp	-273,00 - 670760,00				
511-518	4 Bytes	E	C--W-	DPT_Value_4_Count	-2147483648 -	-	-	[FL] (4 bytes) Dato de entrada x	4 bytes Dato de entrada

				2147483647					
519-528	1 Bit	S	CTR--	DPT_Bool	0/1	-	-	[FL] Función x - Resultado	(1 bit) Booleano
	1 Byte	S	CTR--	DPT_Value_1_Ucount	0 - 255	-	-	[FL] Función x - Resultado	(1 byte) Sin signo
	2 Bytes	S	CTR--	DPT_Value_2_Ucount	0 - 65535	-	-	[FL] Función x - Resultado	(2 bytes) Sin signo
	4 Bytes	S	CTR--	DPT_Value_4_Count	-2147483648 - 2147483647	-	-	[FL] Función x - Resultado	(4 bytes) Con signo
	1 Byte	S	CTR--	DPT_Scaling	0% - 100%	-	-	[FL] Función x - Resultado	(1 byte) Porcentaje
	2 Bytes	S	CTR--	DPT_Value_2_Count	-32768 - 32767	-	-	[FL] Función x - Resultado	(2 bytes) Con signo
	2 Bytes	S	CTR--	DPT_Value_Temp	-273,00 - 670760,00	-	-	[FL] Función x - Resultado	(2 bytes) Flotante

Únete y envíanos tus consultas
sobre los dispositivos Zennio:
<http://zennio.zendesk.com>

Zennio Avance y Tecnología S.L.
C/ Río Jarama, 132. Nave P-8.11
45007 Toledo (Spain).

Tel. +34 925 232 002.
Fax. +34 925 337 310.
www.zennio.com
info@zennio.com



RoHS