



# **ACTinBOX**

## **ACTinBOX CLASSIC KNX**

**ZN110-AB46**

## INDICE:

1. INTRODUCCIÓN .....	1
1.1. PRODUCTO.....	4
1.2. OBJETOS DE COMUNICACIÓN .....	4
2. Salidas .....	1
2.1. CANALES PERSIANA .....	7
2.1.1. TIPO .....	7
2.1.2. TIEMPOS .....	7
2.1.3. FUNCIONES.....	9
2.1.3.1. OBJETO DE ESTADO .....	9
2.1.3.2. CONTROL PRECISO.....	9
2.1.3.3. ESCENAS .....	9
2.1.3.4. BLOQUEO .....	10
2.1.3.5. ALARMAS.....	11
2.1.3.6. MOVIMIENTO INVERTIDO .....	12
2.1.3.7. POSICIONAMIENTO DIRECTO.....	12
2.1.3.8. CONFIGURACIÓN INICIAL .....	13
2.2. SALIDAS INDIVIDUALES.....	13
2.2.1. TIPO .....	13
2.2.2. FUNCIONES.....	14
2.2.2.1. OBJETO DE ESTADO .....	14
2.2.2.2. TEMPORIZACIONES.....	16
2.2.2.3. ESCENAS .....	17
2.2.2.4. BLOQUEO .....	19
2.2.2.5. ALARMA.....	19
2.2.2.6. CONFIGURACIÓN INICIAL .....	20
3. 3. ENTRADAS .....	1
3.1. PULSADOR.....	22
3.1.1. FUNCION “0/1”.....	22
3.1.2. FUNCION PERSIANAS.....	23
3.1.3. FUNCION DIMMER.....	23
3.1.4. FUNCION ESCENA.....	24
3.1.5. CONFIGURACIÓN ADICIONAL.....	24
3.2. INTERRUPTOR/SENSOR .....	25
4. FUNCIONES LÓGICAS .....	1

4.1.	LLAMADA .....	28
4.2.	OPERACIONES.....	28
4.3.	RESULTADO .....	32
5.	OBJETOS DE COMUNICACIÓN.....	1
5.1.	NOMENCLATURA: .....	34

# 1. INTRODUCCIÓN

## 1.1. PRODUCTO

El **ACTinBOX CLASSIC**, es un actuador KNX que combina en un mismo dispositivo:

- **4 x 10A SALIDAS** binarias multifunción (**INDIVIDUALES** o **CANALES PERSIANA**).
- **6 ENTRADAS** binarias multifunción para la conexión de sensores y pulsadores libres de potencial.
- **FUNCIONES LÓGICAS** Avanzadas multi-operación.

Estos 3 bloques trabajan independientemente; y cualquiera de ellos puede interactuar con los otros como si se tratase de 3 aparatos autónomos conectados al **BUS KNX**.

## 1.2. OBJETOS DE COMUNICACIÓN

El **ACTinBOX** está provisto de **151 Objetos de Comunicación** organizados en cuatro grupos principales:

**SALIDAS INDIVIDUALES**

**ENTRADAS**

**CANALES PERSIANA**

**FUNCIONES LÓGICAS**

**NOMENCLATURA:** Para una fácil localización durante la asignación de direcciones de grupo, cada Objeto de Comunicación se nombra en función del grupo al que pertenece de la siguiente manera:

**["Grupo al que pertenece"] "Función que desempeña"**

Las abreviaturas asociadas a los diferentes grupos son las siguientes:

### - Salidas Individuales:

[S1] → Salida 1  
[S2] → Salida 2  
[S3] → Salida 3  
[S4] → Salida 4

### - Entradas:

[E1] → Entrada 1  
... ..  
... ..  
[E6] → Entrada 6

### - Canales Persiana:

[CA] → Canal A  
[CB] → Canal B

### - Funciones Lógicas:

[FL] → Función Lógica

Ejemplos: (Ver Figura 1.2)

- **[S1] Estado:** Objeto de Estado de la Salida 1
- **[CB] Parar:** Control de Parada de la Persiana del Canal B
- **[E3] Bloqueo:** Objeto que bloquea el control sobre la entrada 3
- **[FL] RESULTADO Función 2 (2 bytes):** Objeto que almacena el resultado (2 bytes) de la Función Lógica 2.

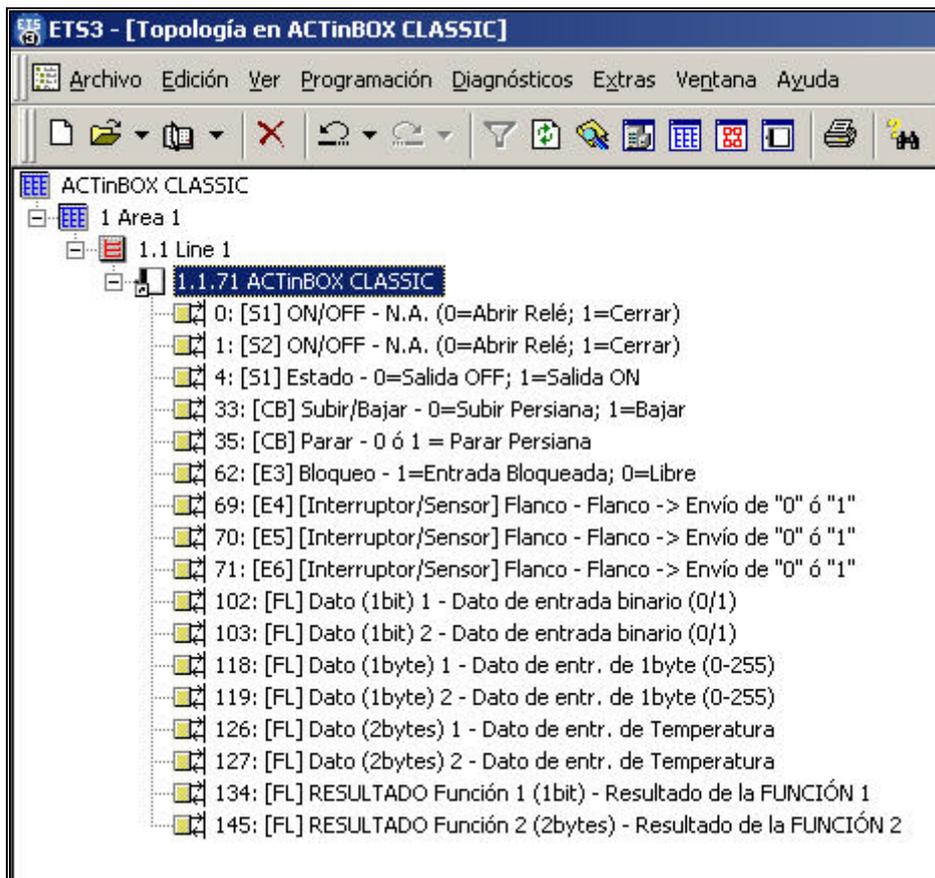


Figura 1.2

## 2. SALIDAS

El ACTinBOX CLASSIC está provisto de **4 salidas binarias de 10A** (ver hoja técnica de producto). Estas 4 salidas se nombran como Salida 1, Salida 2, Salida 3 y Salida 4 y **están divididas en 2 grupos de 2 salidas cada uno**, denominados **Canales**:

- **Canal A:** “Salida 1” y “Salida 2”
- **Canal B:** “Salida 3” y “Salida 4”

Cada uno de los dos canales puede ser parametrizado independientemente como:

- **Salidas Individuales:** Cada una de las salidas se puede manejar de manera independiente. Se utilizan para el control de cargas eléctricas, como puntos de luz,....
- **Canales Persiana:** Las dos salidas pertenecientes a cada canal se gestionan conjuntamente. Pueden ser usadas para controlar persianas motorizadas (con o sin lamas), toldos,....

- **En el caso de Canal de Persiana:**

- ✓ **La Salida 1 (3 en el Canal B):** Se convierte en la fase encargada de Subir la Persiana.
- ✓ **La Salida 2 (4 en el Canal B):** Se convierte en la fase encargada de Bajar la Persiana.

***Ejemplo de Configuración:** Consideremos una instalación en la que se necesita manejar una persiana y un punto de luz.*

*En este caso el ACTinBOX podría parametrizarse de la siguiente manera:*

- **Canal A = Canal de Persiana**

- **Canal B = Salidas Individuales**

- ✓ **Salida 3 = Normalmente Abierta**
- ✓ **Salida 4 = Deshabilitada**

## 2.1.CANALES PERSIANA

El ACTinBOX permite instalar en sus Canales, cualquier tipo de persiana u otros cerramientos motorizados. Para manejarlos, se pone a disposición del usuario, los objetos de control básicos “Subir/Bajar” y “Parar”, y también un conjunto de funciones adicionales, cada una con sus propios objetos de comunicación.

El control de la persiana con los objetos básicos se realiza como sigue:

- **Subir Persiana:** se envía un “0” al objeto “Subir/Bajar”.
- **Bajar Persiana:** se envía un “1” al objeto “Subir/Bajar”.

*Nota: Cuando el objeto “Subir/Bajar” recibe un “0” o un “1”, la persiana comenzará a moverse, y no se detendrá a menos que llegue al final de su recorrido o que haya recibido alguna otra orden anulando la anterior.*

- **Para parar una persiana en movimiento:** se debe enviar un “0” ó un “1” al objeto “Parar”.

### 2.1.1. TIPO

- **PERSIANA NORMAL/TOLDO** → Se trata de las persianas (u otros cerramientos) de uso habitual, las que poseen un único movimiento (de Subida o de Bajada) con el que se desplazan en bloque.
- **PERSIANA CON LAMAS ORIENTABLES** → Son persianas especiales que poseen lamas, cuyo movimiento puede manejarse con el mismo motor.El ACTinBOX en este caso, permite controlar los dos movimientos, el de **rotación de las lamas** (para conseguir una mayor o menor incidencia de la luz exterior), y el de **movimiento de la persiana** (el encargado de subirla o bajarla).Si se selecciona este tipo de control (persiana con lamas orientables), el objeto “Parar” es reemplazado por el objeto “Parar/Paso”. De manera que si la persiana se encuentra en movimiento al recibir un “0” ó un “1” por este objeto, la persiana se para; mientras que si se encuentra parada, al recibir un “0” da un pequeño paso hacia arriba (giro ascendente de las lamas), y al contrario con el “1” (giro descendente de las lamas”).

### 2.1.2. TIEMPOS

Existen 2 tiempos (3 en el caso de persiana con lamas orientables) que son necesarios establecer para el correcto funcionamiento del Canal.

- **RECORRIDO DE LA PERSIANA:** Es el tiempo que tarda la persiana en realizar su recorrido completo. Se puede utilizar el tiempo de bajada o bien el de subida. En caso de que ambos tiempos (el de subida y el de bajada) sean distintos, se debe poner aquí el tiempo de bajada y utilizar

la función “**Tiempos de Subida y Bajada distintos**” para indicar el de subida. No es necesario graduar este tiempo periódicamente, ya que el **ACTinBOX** recuerda siempre la posición exacta en que se encuentra la persiana (aunque se vaya la tensión en el BUS).

*Nota: Tras la programación con el ETS, el **ACTinBOX** considera que la persiana está completamente subida.*

- **DURACIÓN DEL GIRO DE LAS LAMAS:** (Solo para el tipo persiana con lamas orientables). Es el tiempo que tardan las lamas en realizar un giro completo.
- **TIEMPO DE SEGURIDAD ANTES DEL CAMBIO DE SENTIDO :** Es el tiempo de seguridad asignado como medida de protección ante un cambio de sentido en el movimiento de la persiana, para no sobrecargar el motor que la controla. Es decir, si la persiana se encuentra subiendo y se le envía una orden de bajar, el **ACTinBOX** detiene unos instantes la persiana antes de ponerla a bajar. Se recomienda un valor como el que hay puesto por defecto en parámetro 5 (décimas de segundo).
- **¿TIEMPOS DE SUBIDA Y BAJADA DISTINTOS?** : En el caso de persianas pesadas, los tiempos de subida y de bajada de la misma pueden ser distintos; en ese caso, se debe activar este parámetro e indicar aquí el tiempo de Subida y utilizar el parámetro "**Recorrido de la Persiana**" para establecer el tiempo de Bajada de la persiana como ya se ha comentado anteriormente.

*Ejemplo de Configuración: La persiana asociada al Canal B tarda 15 segundos en bajar, y 20 en subir. En este caso la parametrización queda como:*

TIEMPOS:	
- Recorrido de la Persiana (Tiempo de Bajada) [x 0.1s]	150
- Duración del giro de las Lamas [décimas de segundo]	20
- Tiempo de seguridad ante el cambio de sentido [décimas de segundo]	5
- Tiempos de Subida y Bajada distintos?	Sí
Tiempo de Subida [x 0.1s] (el de bajada es el de más arriba)	200

- **TIEMPO ADICIONAL AL LLEGAR AL LÍMITE:** Este parámetro garantiza que la persiana siempre alcance su final de carrera. Una vez la persiana ha consumido su tiempo de Subida o de Bajada, este parámetro establece un tiempo adicional durante el cual el motor seguirá girando para prevenir pequeños desajustes en el final de carrera de la persiana.

## 2.1.3. FUNCIONES

Cada uno de los siguientes parámetros añade funcionalidad o características especiales al control de cada uno de los Canales Persiana (A ó B).

### 2.1.3.1. OBJETO DE ESTADO

Esta función pone a disposición del usuario un objeto que indica en qué posición se encuentra la persiana en cada momento.

Se trata del objeto "**Posición Actual**". Es un objeto de 1 byte medido en tanto por ciento "%". De manera que el objeto vale 0 (0=%) cuando la persiana está completamente subida, y vale 255 (=100%) cuando está completamente bajada. El resto de valores representan las distintas posiciones intermedias.

***Nota:** El objeto de estado "Posición Actual" de la persiana ha sido programado para que cada vez que la persiana se pone en movimiento éste sea enviado al BUS cada segundo, actualizando así la posición de las persianas en tiempo real.*

### 2.1.3.2. CONTROL PRECISO

Esta función permite al usuario mover la persiana hasta cualquier posición de su recorrido, a través del objeto de 1 byte "**Posición Deseada**".

Cada vez que el **ACTinBOX** recibe por ese objeto un nuevo valor (por ejemplo 50%), mueve la persiana hasta la posición correspondiente (en el caso del ejemplo hasta la mitad).

### 2.1.3.3. ESCENAS

Esta función permite el uso de escenas (con formato de 1 byte) para controlar la persiana. Es decir, permite elegir la posición a la que debe moverse la persiana en función de la escena que llega al **ACTinBOX** a través del objeto "**Escenas**".

- **NÚMERO DE ESCENAS** → A través de este parámetro se designa el número de escenas que se van a utilizar. Hasta un máximo de 5.
- **ESCENA** → Indica el número de escena ante la cual reaccionará la persiana.
- **REACCIÓN** → Indica la posición a la que se moverá la persiana cuando llegue dicho número (el elegido en el parámetro ESCENA) por el objeto "Escenas".

**Ejemplo:** Se desean utilizar 4 escenas en cierta instalación (la 4, la 6, la 8 y la 9). Pero sólo se necesita que la persiana del ACTinBOX se mueva a una posición determinada para las tres primeras (la 4, la 6 y la 8). Las posiciones deseadas son:

- Ante la Escena 4 → Arriba

- Ante la Escena 6 → Abajo

- Ante la Escena 8 → A la mitad (50%)

La parametrización del canal en este caso queda del siguiente modo:

Escenas	
Nº DE ESCENAS	3
- Escena [1->0; 64->63]	4
- Reacción	Arriba
- Escena [1->0; 64->63]	6
- Reacción	Abajo
- Escena [1->0; 64->63]	8
- Reacción	Posición Determinada
Posición [0=0%; 255=100%]	127

#### 2.1.3.4. BLOQUEO

Gracias a esta función, es posible bloquear la salida. Es decir, deshabilitar su control (el control ON/OFF y el Temporizado). La salida se bloquea enviando un “1” al objeto “**Bloqueo**”; y se desbloquea enviando un “0”.

**Nota:** Tan solo la función de Alarma tiene mayor prioridad que el Bloqueo. Es decir, que si estando en estado bloqueado, ocurre una alarma, la salida se posicionará tal y como se indique en la Función de Alarma. Al cesar la Alarma la salida vuelve al estado Bloqueo.

### 2.1.3.5. ALARMAS

Esta función está diseñada para casos en los que el **ACTinBOX** debe reaccionar ante situaciones de Alarma.

Ante una Alarma, esta función obliga a la salida a colocarse en la posición (ON, OFF o intermitente) elegida por parámetro; y tras esto, la bloquea, impidiendo su control, hasta que cese la Alarma.

- **NÚMERO DE ALARMAS** → Este parámetro determina **SI SE VAN A DISTINGUIR 1 ó 2 Alarmas**. En caso de elegir 2 alarmas, cada una de ellas se controla de modo independiente con el objeto correspondiente ("**Alarma 1**" para la 1ª y "**Alarma 2**" para la 2ª).

***Nota:** La Alarma1 tiene prioridad sobre la Alarma2. Es decir, si el Canal se encuentra en estado de Alarma2, y se produce una Alarma1, la persiana pasará a estado de Alarma1, y no volverá a estado de Alarma2 hasta que no cese la Alarma1. Mientras que si el Canal se encuentra en estado de Alarma1 y se produce una Alarma2, continuará en estado de Alarma1.*

- **DESENCADENANTE** → Este parámetro determina el valor que activa la Alarma. Es decir, se produce una Alarma cuando por el objeto "Alarma" llega el valor elegido como Desencadenante. Al valor opuesto (el que no activa la Alarma se le llama ValorPasivo).
- **MONITORIZACIÓN** → A veces es necesario conocer en cada momento, que no se ha producido una Alarma y que el sensor funciona correctamente. Para ello, el sensor que envía al **ACTinBOX** el Desencadenante, debe enviar continuamente el **ValorPasivo** cuando no se esta en estado de Alarma. En estos casos es cuando se debe activar el parámetro Monitorización. De manera que si el **ACTinBOX** no recibe el **ValorPasivo** durante el tiempo elegido en
- **TIEMPO DE MONITORIZACION**, se activará automáticamente la Alarma.

***Nota:** Se recomienda elegir un tiempo superior al doble del ciclo de envío del sensor, por si se perdiera algún mensaje.*

- **REACCIÓN** → Este parámetro determina cuál será la reacción de la salida (indica la posición a la que se moverá la persiana) al saltar la Alarma. (En caso de elegir "**Posición Determinada**", hay recordar que 0=0% y 255=100%).

- **DESACTIVACIÓN** → Existen 2 métodos para desactivar la Alarma:
  - ✓ **NORMAL** → Consiste en enviar al objeto "**Alarma**" el **ValorPasivo** (valor opuesto al Desencadenante).
  - ✓ **ENCLAVAMIENTO** → Consiste en aplicar el método Normal y después enviar un "1" al objeto "**Enclavamiento**". Este segundo método permite que, aunque la situación de alarma haya concluido, la salida continúe bloqueada hasta no ser habilitada manualmente desde otro punto.
- **ESTADO FINAL** → Este parámetro determina que hará la salida al terminar la Alarma (quedarse como esta, colocarse en ON, en OFF o en la posición que tuviera antes de producirse la Alarma).

### 2.1.3.6. MOVIMIENTO INVERTIDO

Esta función permite el control de la persiana de modo invertido a como se hace habitualmente. Es decir, subir la persiana con un "1", y bajarla con un "0".

Para ello se utiliza el objeto de comunicación "". Este tipo de control es compatible con el control normal. **Movimiento Invertido** Así pues, con el objeto "**Subir/Bajar**" se realiza el control normal y con el objeto "**Movimiento Invertido**" el control invertido.

Esto es útil, por ejemplo, para hacer un Apagado General en una instalación con el que se desea Apagar las luces y Bajar las persianas. En este caso se enviaría un "0" a los objetos "**ON/OFF**" de control de las luces y a los objetos "**Movimiento Invertido**" de control de las persianas.

### 2.1.3.7. POSICIONAMIENTO DIRECTO

Esta función permite mover la persiana a posiciones fijadas, utilizando objetos de 1 bit.

Al recibir un "1" por uno de estos objetos ("**Posicionamiento Directo**" o "**Posicionamiento Directo 2**"), la persiana se moverá a la posición elegida por parámetro.

- **Nº DE POSICIONES** → Este parámetro define si se van a utilizar 1 ó 2 posiciones.
- **POSICIÓN** → Con este parámetro se determina la posición elegida. (Recordar que 0=0% y 255=100%).

- **GRABACIÓN NUEVAS POSICIONES** → Este parámetro determina si se va a permitir almacenar nuevas posiciones de la persiana. Al habilitar esta opción aparecen dos nuevos Objetos de Comunicación:

- ✓ **Grabar Posición**

- ✓ **Grabar Posición 2**

Para grabar nuevas posiciones es necesario enviar un "1" a estos objetos.

### 2.1.3.8. CONFIGURACIÓN INICIAL

Esta función permite elegir el comportamiento del canal persiana del **ACTinBOX** tras una caída de tensión en el BUS, o tras descargarle una programación desde el ETS.

- **ESTADO** → A través de este parámetro se selecciona la posición exacta a la que debe colocarse la persiana tras una caída de tensión en el BUS. Tras una programación del dispositivo con el ETS, la opción "**Ultimo estado registrado**" significa que la persiana permanecerá en la posición exacta en la que se encontraba antes de la programación.
- **ACTUALIZACIÓN** → Este parámetro ofrece la posibilidad de enviar al BUS el Estado Inicial, para actualizar al resto de aparatos de la instalación.
- **RETARDO** → Como algunos de los aparatos en la instalación, pueden tardar más en iniciarse, es posible retrasar el envío del Estado Inicial a través de este parámetro, para asegurar que todos los aparatos están listos para recibir el mensaje. El envío del Estado Inicial se realiza siempre a través del objeto "Estado".

*Nota: El envío del estado inicial se realiza siempre a través del objeto "Posición Actual".*

## 2.2.SALIDAS INDIVIDUALES

En este caso cada una de las dos salidas del Canal controlará dispositivos diferentes e independientes. Si no se tiene previsto utilizarla, la salida puede ser deshabilitada

### 2.2.1. TIPO

Es necesario indicar si la salida es de tipo **Normalmente Abierta** o **Normalmente Cerrada**.

**- Normalmente Abierta:**

ON → Cerrar Relé  
OFF → Abrir Relé

**- Normalmente Cerrada:**

ON → Abrir Relé  
OFF → Cerrar Relé

***Nota:** En lo que respecta a las Salidas Individuales, **ON =1** y **OFF = 0**. Así, cada vez que se hace referencia a que por un objeto se envía o se recibe un “ON”, significa que se envía o se recibe el valor 1 (y viceversa). En ocasiones, al ON se le llama Encendido, y al OFF Apagado. El control normal de la salida, se realiza a través del Objeto de Comunicación “ON/OFF”.*

## 2.2.2. FUNCIONES

Cada uno de los siguientes parámetros añade funcionalidad o características especiales al control de cada uno de las Salidas Individuales.

### 2.2.2.1. OBJETO DE ESTADO

El objeto de comunicación “Estado” refleja el estado en que se encuentra la salida en cada momento.

- ⇒ Cuando la salida se encuentra a ON este objeto vale “1”.
- ⇒ Cuando la salida se encuentra a OFF este objeto vale “0”.

Cada vez que el estado de la salida cambia, el nuevo valor de este objeto es enviado al BUS.

- **ENLACES INTERNOS** → Esta función permite enlazar internamente el objeto “Estado” de la Salida con:

- ⇒ **Los objetos de 1 bit de las Entradas.**
- ⇒ **Los objetos de 1 bit de las Funciones Lógicas** (los de entrada de datos).

Es decir, que si se establece alguno de estos enlaces, cuando la salida cambia y se envía al **BUS** el nuevo valor del objeto “Estado” a través de la dirección de grupo correspondiente, internamente se produce el mismo efecto sobre los objetos elegidos de las Entradas y Funciones Lógicas. Esto es especialmente útil para actualizar las Entradas cuando éstas ejercen un control conmutado sobre las salidas.

- ✓ **ENTRADA 1** → Si se activa enlaza internamente el objeto "Estado" de la Salida con el objeto de 1 bit de la Entrada 1. Las opciones de enlace son:

- Nada

- Pulsación Corta
- Pulsación Larga
- Pulsación Corta y Larga
- Flanco

✓ ..... ..

✓ ..... ..

✓ **ENTRADA 6** → Si se activa enlaza internamente el objeto "Estado" de la Salida con el objeto de 1 bit de la Entrada 6.

Las opciones de enlace son:

- Nada
- Pulsación Corta
- Pulsación Larga
- Pulsación Corta y Larga
- Flanco

✓ **FUNCIONES LÓGICAS** → Si se activa enlaza internamente el objeto "Estado" de la Salida con el objeto de 1 bit de las Funciones Lógicas [FL] (entrada de datos). Las opciones de enlace en este caso son:

- Dato (1bit)1
- ..... ..
- ..... ..
- Dato (1bit)16

**Ejemplo de Configuración:** *En una instalación, la Salida 1 [S1] está conectada a un punto de luz.*

*Las Entradas 1 y 2 [E1] y [E2] controlan la Salida 1 con pulsaciones cortas: la **Entrada 1** apaga la luz, y la **Entrada 2** envía una conmutación (ON-OFF-ON alternativamente).*

*Si se pulsa la Entrada 2, la luz se encenderá. Si a continuación se pulsa la Entrada 1, se apagará. Pero si se vuelve a pulsar la Entrada 2, ésta enviará un OFF (ya que lo último que envió fue ON), y la luz no hará nada.*

*Para evitar esto, se enlaza internamente el objeto Estado de la Salida 1 con el objeto Pulsación Corta de la Entrada 2, y al ser apagada por la Entrada 1, la Entrada 2 será actualizada y no dará falsos conmutados.*

## 2.2.2.2. TEMPORIZACIONES

Es posible controlar la salida de manera temporizada.

Existen 2 tipos de controles temporizados:

- **Temporización Simple:** Se lleva a cabo a través del Objeto “Temporización”.
- **Intermitencia:** Se lleva a cabo a través del Objeto “Intermitencia”.

*Nota: Estos controles temporizados son independientes entre sí, e independientes del control normal “ON/OFF”, ya que se manejan desde tres objetos distintos.*

*Por ejemplo, si se envía un ON al objeto “Temporización”, comienza un ON temporizado en la salida. Si antes de terminar el ON Temporizado se envía un OFF al objeto “ON/OFF”, se aplicará un Apagado sobre la salida, y la temporización concluirá.*

- **TEMPORIZACIÓN SIMPLE** → Se trata de una temporización normal que el ACTinBOX aplica a la salida al recibir un ON o un OFF por el objeto "Temporización".
  - ✓ **RETARDO AL ENCENDIDO**→ A través de este parámetro, se determina el tiempo transcurrido entre la orden de ON (sobre el objeto “Temporización”) y el Encendido (ON) de la salida. Este tiempo se mide en décimas de segundo (por ejemplo, si se desea un retardo de 2 segundos y medio, habrá que elegir el valor 25 en el parámetro). Un “0” indica que no se aplicará retardo.
  - ✓ **RETARDO AL APAGADO**→ Este parámetro determina el tiempo transcurrido entre la orden de OFF (sobre el objeto “Temporización”) y el Apagado de la salida. Su funcionamiento es el mismo que el de Retardo al Encendido.
  - ✓ **DURACIÓN DEL ENCENDIDO**→ Este parámetro determina el tiempo que permanece la salida a ON antes de volver a OFF. Un “0” indica que la duración es ilimitada (que no se aplica temporización a la Duración del Encendido).

*Nota: El funcionamiento de los Retardos y la Duración se detalla a continuación:*

*- Cuando se envía un “1” al objeto “Temporización”, se ordena encender la salida aplicando el Retardo al Encendido y la Duración del Encendido.*

- Cuando se envía un “0” al objeto “**Temporización**”, se ordena el Apagado de la salida aplicando el Retardo al Apagado

- ✓ **MULTIPLICACIÓN** → Consiste en la multiplicación de una temporización tantas veces como se reciba un “1” o un “0” por el objeto “**Temporización**”.

Nota: El funcionamiento de la Multiplicación se detalla a continuación:

- Sin Multiplicación: Si durante un ON Temporizado se recibe un “1” por el objeto “**Temporización**”, el **ACTinBOX** empieza a contar de nuevo el ON Temporizado.

- Con Multiplicación: Si durante un ON Temporizado se recibe un “1” por el objeto “**Temporización**”, el **ACTinBOX** pasa a contar el doble de tiempo. Si recibe otro “1” antes de terminar, pasa a contar el triple, etc.

Lo mismo ocurre con el OFF Temporizado.

- **INTERMITENCIA** → La función de Intermitencia permite realizar una secuencia en la salida del tipo ON-OFF-ON-OFF....
  - ✓ **DURACIÓN DEL ENCENDIDO** → Determina la duración de cada repetición del ON a través de los parámetros correspondientes.
  - ✓ **DURACIÓN DEL APAGADO** → Determina la duración de cada repetición del OFF a través de los parámetros correspondientes.
  - ✓ **Nº DE REPETICIONES** → Selección del Nº de Repeticiones que se llevarán a cabo. Para un Nº de Repeticiones ilimitado, elegir el valor “0”.
  - ✓ **ESTADO FINAL** → Permite definir el estado en el que quedará la salida al terminar la última repetición.

Nota: La intermitencia se pone en marcha enviando un “1” al objeto “**Intermitencia**”, y se detiene enviándole un “0” al mismo.

### 2.2.2.3. ESCENAS

Esta función permite manejar la salida a través del objeto “**Escenas**”, utilizando para ello escenas BUS (tipo 1 byte).

- **Nº DE ESCENAS** → Este parámetro permite seleccionar hasta un máximo de 5 escenas

- ✓ **ESCENA**→ Indica el número de escena ante la cual reaccionará la salida.
- ✓ **REACCIÓN**→ Indica la posición en la que quedará la salida (ON u OFF) cuando llegue dicho número de escena por el objeto "Escenas".

***Ejemplo:** Se desea utilizar 4 escenas en cierta instalación (la 4, la 6, la 8 y la 9).*

*Pero sólo se necesita que la salida del **ACTinBOX** se sitúe en una posición concreta (ON u OFF) para las tres primeras (la 4, la 6 y la 8).*

*Esta posición deseada es:*

*- Ante la Escena 4 → ON*

*- Ante la Escena 6 → ON*

*- Ante la Escena 8 → OFF*

*Para ello, hay que parametrizar esta salida del ACTinBOX de la siguiente manera:*

Escenas	
Nº DE ESCENAS	3
- Escena [1->0; 64->63]	4
- Reacción	ON
- Escena [1->0; 64->63]	6
- Reacción	ON
- Escena [1->0; 64->63]	8
- Reacción	OFF

#### 2.2.2.4. BLOQUEO

Gracias a esta función, es posible bloquear la salida. Es decir, deshabilitar su control (el control ON/OFF y el Temporizado). La salida se bloquea enviando un "1" al objeto "Bloqueo"; y se desbloquea enviando un "0".

*Nota: Tan solo la función de Alarma tiene mayor prioridad que el Bloqueo. Es decir, que si estando en estado bloqueado, ocurre una alarma, la salida se posicionará tal y como se indique en la Función de Alarma. Al cesar la Alarma la salida vuelve al estado de Bloqueo.*

#### 2.2.2.5. ALARMA

Esta función está diseñada para casos en los que el **ACTinBOX** tiene que reaccionar ante situaciones de alarma.

Ante una Alarma, esta función obliga a la salida a colocarse en la posición (ON, OFF o intermitente) elegida por parámetro; y tras esto, la bloquea, impidiendo su control, hasta que cese la Alarma.

- **DESENCADENANTE** → Este parámetro determina el valor que activa la Alarma. Es decir, se produce una Alarma cuando por el objeto "Alarma" llega el valor elegido como Desencadenante. Al valor opuesto (el que no activa la Alarma se le llama **ValorPasivo**).
- **MONITORIZACIÓN** → A veces es necesario conocer en cada momento, que no se ha producido una Alarma. Para ello, el sensor que envía al **ACTinBOX** el Desencadenante, debe enviar continuamente el **ValorPasivo** cuando no se está en estado de Alarma. En estos casos es cuando se debe activar el parámetro Monitorización. De manera que si el **ACTinBOX** no recibe el **ValorPasivo** durante el tiempo elegido en
- **TIEMPO DE MONITORIZACION**, activará automáticamente la Alarma.

*Nota: Se recomienda elegir un tiempo superior al doble del ciclo de envío del sensor, por si se perdiera algún mensaje.*

- **REACCIÓN** → Este Parámetro determina cuál será la reacción de la salida (ON, OFF o Intermitencia) al saltar la Alarma. La reacción tipo Intermitencia hace que cuando ocurra una Alarma, la salida comience a realizar la secuencia ON-OFF-ON-OFF...., donde la duración de cada ON, de cada OFF y el nº de repeticiones se determinan por parámetro. En caso de elegir la opción "**Salida Intermitente**", aparecen las siguientes opciones en el entorno de parametrización en el ETS:

- ✓ **DURACIÓN DEL ENCENDIDO**→ Determina la duración de cada repetición del ON a través de los parámetros correspondientes.
  - ✓ **DURACIÓN DEL APAGADO**→ Determina la duración de cada repetición del OFF a través de los parámetros correspondientes.
  - ✓ **Nº DE REPETICIONES**→ Selección del Nº de Repeticiones que se llevarán a cabo. Para un Nº de Repeticiones ilimitado, elegir el valor "0".
  - ✓ **ESTADO FINAL**→ Permite definir el estado en el que quedará la salida al terminar la última repetición.
- **DESACTIVACIÓN** → Existen 2 métodos para desactivar la Alarma:
- ✓ **NORMAL**→ Consiste en enviar al objeto "Alarma" el **ValorPasivo** (valor opuesto al Desencadenante).
  - ✓ **ENCLAVAMIENTO**→ Consiste en aplicar el método Normal y después enviar un "1" al objeto "Enclavamiento". Este segundo método permite que, aunque la situación de alarma haya concluido, la salida continúe bloqueada hasta no ser habilitada manualmente desde otro punto.
- **ESTADO FINAL** → Este parámetro determina que hará la salida al terminar la Alarma (quedarse como esta, colocarse en ON, en OFF, o en la posición que tuviera antes de producirse la Alarma).

## 2.2.2.6. CONFIGURACIÓN INICIAL

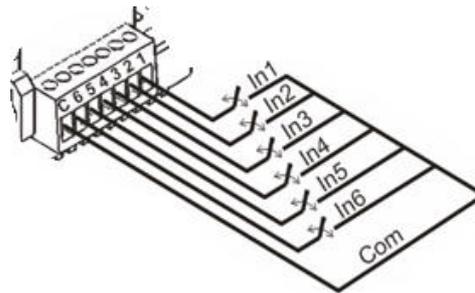
Esta función permite elegir en qué posición (ON u OFF) debe colocarse la salida al inicio, es decir, al volver la tensión de BUS al **ACTinBOX**, o tras descargarle una programación desde el ETS.

- **ESTADO** → A través de este parámetro se selecciona la posición inicial elegida. La opción "Último Estado Registrado" indica que la salida permanecerá tal y como hubiera quedado antes de irse la tensión de BUS. Tras una programación, la opción "Ultimo Estado Registrado" hace que la salida arranque en estado OFF, es decir, relé abierto si la salida es Normalmente Abierta (y viceversa).
- **ACTUALIZACIÓN** → Este parámetro ofrece la posibilidad de enviar al BUS el Estado Inicial, para actualizar al resto de aparatos de la instalación.
- **RETARDO**→ Como algunos de los aparatos en la instalación, pueden tardar más en iniciarse, es posible retrasar el envío del Estado Inicial a través de este parámetro, para asegurar que todos los aparatos están listos para recibir el mensaje. El envío del Estado Inicial se realiza siempre a través del objeto "Estado".

### 3. 3. ENTRADAS

El ACTinBOX dispone de **6 Entradas Binarias para contactos libres de potencial** que se pueden configurar individualmente. Cada una de ellas puede ser conectada a un **Pulsador o a un Interruptor/Sensor**.

Al unir físicamente cualquier entrada con el “**común**” de las entradas se produce un Contacto Cerrado en dicha entrada. Y al separarlos, se produce un Contacto Abierto.



A través de **una entrada tipo PULSADOR**, se puede realizar cualquiera de las siguientes funciones:

- **Envío de 0/1:** Envío al BUS del valor de (1 bit) "0" ó "1".
- **Control de Persianas:** Envío al BUS del objeto correspondiente para mover/parar una persiana (o varias).
- **Control de Dimmer:** Envío al BUS del objeto correspondiente para un Dimmer (o varios).
- **Envío de Escena:** Envío al BUS de una Escena en formato 1 byte (ejecución y grabación).

Se puede elegir que el ACTinBOX realice una de estas funciones ante una pulsación corta, e independientemente a que realice otra ante una pulsación larga.

***Ejemplo:** La Entrada 3 puede controlar una Persiana con la Pulsación Corta, y ejecutar una escena con la Larga.*

Esto hace que el ACTinBOX pueda manejar hasta 12 funciones independientes a través de sus entradas.

A través de **una entrada tipo SENSOR**, se realiza la siguiente función:

- **Envío de 0/1:** Para cada flanco (de subida y de bajada), se elige si se envía al BUS un "0", un "1", o una conmutación de "0" y "1".

### 3.1.PULSADOR

Un pulsador conectado a una entrada consiste en un mecanismo que en condiciones normales mantiene sus contactos abiertos. Y que al ser accionado, cierra el contacto durante el tiempo en que es accionado, para volver después a su estado normal. A este accionamiento se le llama “**pulsación**”, y al tiempo que dura la misma se le llama “**tiempo de pulsación**”.

Según el tiempo de pulsación se pueden distinguir dos acciones:

- **Pulsación Corta**
- **Pulsación Larga**

Suelen utilizarse como pulsadores, los mecanismos de pulsación convencionales como teclas o teclados.

*Nota: El ACTinBOX no acepta pulsadores Normalmente Cerrados, es decir, conectados de manera que en condiciones normales sean de Contacto Cerrado.*

#### 3.1.1. FUNCION “0/1”

Esta función tiene como resultado el envío al BUS de un objeto de 1 bit.

- **REACCIÓN** → En función de este parámetro se determina si el valor que se envía es “0”, “1” o una conmutación alternativa entre “0” y “1”.
- **TRANSMISIÓN CÍCLICA** → Con este parámetro se elige si se envía periódicamente el “0” ó el “1”, incluso ambos (si se selecciona “Siempre”).
  - ✓ **CICLO DE ENVÍO** → Este parámetro define el tiempo entre mensaje y mensaje al utilizar el envío periódico.
- **ENLACES INTERNOS** → A través de este parámetro se enlazan internamente el objeto asociado a la entrada con otros objetos del ACTinBOX. Esta asociación interna, permite que la información que circula por el BUS, también pueda tener efecto sobre otras partes del aparato A través de esta utilidad se pueden establecer enlaces con todos los objetos de control de 1 bit; tanto de **Salidas** como de **Funciones Lógicas** (excepto con los de movimiento y parada de la persiana, reservadas para la función de entradas tipo “**Control de Persianas**”).

*Nota: Los Enlaces Internos **sólo** tienen efecto sobre las partes habilitadas.*

*Por ejemplo: No sirve de nada enlazar con un objeto de la Salida 3, si el Canal B no ha sido seleccionado como Salidas Individuales.*

### 3.1.2. FUNCION PERSIANAS

Esta función tiene como resultado el envío al BUS de un objeto de 1 bit para el control de persianas.

- **REACCIÓN** → En función de este parámetro se determina si el "Objeto" es utilizado para el control de:
  - ✓ **Subir Persiana**
  - ✓ **Bajar Persiana**
  - ✓ **Subir/Bajar Persiana conmutado:** Permite el manejo de la persiana con una sola entrada.
  - ✓ **Parar Persiana**

***Nota:** Este último modo "Parar Persiana" incluye las 3 opciones de Paso Arriba, Paso Abajo y Paso Conmutado, para el control de lamas; pero si no se tienen lamas orientables, sirve cualquiera de las 3 para parar la persiana.*

- **ENLACES INTERNOS** → A través de este parámetro se enlazan internamente el objeto asociado a la entrada con el Canal de la Persiana del ACTinBOX elegido (o incluso con ambos), para que el mismo control que se tiene exteriormente con una persiana de cualquier otro aparato, se tenga también internamente. Con esta función solo se tiene enlace con los objetos de control "Subir/Bajar" ó "Parar" (**DEPENDEN DE LA REACCIÓN ELEGIDA**). Para enlazar con el resto de los objetos de 1 bit del Canal, se debe utilizar la entrada tipo 0/1.

***Nota:** El Control Interno solo tiene efecto si el canal (o canales) elegido ha sido seleccionado en el apartado "Salidas" como Canal de Persiana.*

### 3.1.3. FUNCION DIMMER

Esta función tiene como resultado el envío al BUS de un objeto de control de Dimmers.

- **REACCIÓN** → En función de la opción elegida, este objeto puede ser:
  - ✓ **Control ON/OFF**
    - ⇒ **Encender:** Envía "1".
    - ⇒ **Apagar:** Envía "0".
    - ⇒ **Encender/Apagar Conmutado:** Envía "0 y 1" alternativamente
  - ✓ **Control de Regulación:** Según la opción elegida, envía el objeto de 4 bits correspondiente.

- **PASO DE REGULACIÓN** → Si se elige una de las opciones de Control de Regulación, es necesario determinar este parámetro, que es el incremento o decremento de luminosidad que da el **DIMMER** en cada paso.

### 3.1.4. FUNCION ESCENA

Esta función se utiliza para manejar una escena del **BUS** con la entrada.

- **REACCIÓN** → Con este parámetro se determina si la acción a realizar es la Ejecución/Grabación de una escena (ordena al resto de aparatos que ejecuten la escena emitida).
- **ESCENA** → Determina el Número de Escena a ejecutar /Grabar.
- **ENLACES INTERNOS** → A través de este parámetro se enlazan internamente el objeto asociado a la entrada y los objetos de "**Escenas**" de las salidas del **ACTinBOX**. Esto permite tener internamente el mismo efecto que se tiene externamente sobre el resto de aparatos **BUS**.

### 3.1.5. CONFIGURACIÓN ADICIONAL

- **TIEMPO DE PULSACIÓN** → Establece un umbral de tiempo entre la "pulsación corta" y la "pulsación larga".
- **RETARDO** → Se puede habilitar una temporización al envío del objeto asociado a la acción llevada a cabo en la entrada. Un Valor "0" en este parámetro implica un envío inmediato (**sin temporización**).

***Ejemplo:** Ante una pulsación corta, por ejemplo, el **ACTinBOX** esperará el tiempo designado en "**RETARDO (Pulsación Corta)**" antes de enviar el objeto correspondiente al **BUS**.*

- **BLOQUEO** → Este parámetro habilita el objeto "**Bloqueo**". Este objeto se utiliza para deshabilitar la entrada.
  - ✓ Al recibir un "1" por este objeto, el **ACTinBOX** bloquea cualquier pulsación sobre la entrada.
  - ✓ Al recibir un "0" por este objeto, la entrada vuelve a estar habilitada, pero no se tienen en cuenta las acciones realizadas durante el estado deshabilitado.

## 3.2.INTERRUPTOR/SENSOR

Un Interruptor/Sensor conectado a una entrada consiste en un mecanismo cuyo estado en condiciones normales puede ser tanto en Contacto Abierto como en Contacto Cerrado (y no vuelve automáticamente a la posición anterior como en el caso del Pulsador).

Al paso de un estado a otro se le denominan "**Flanco**":

- **Flanco de Bajada:** Paso de Contacto Cerrado a Contacto Abierto.
- **Flanco de Subida:** Paso de Contacto Abierto a Contacto Cerrado.

Suelen utilizarse como interruptor/sensor los contactos de salida libre de potencial de los sensores convencionales y de los relés, en general.

Al seleccionar una entrada como **Interruptor/Sensor**, se enviará al **BUS** el objeto de 1 bit "**[Interruptor/Sensor] Flanco**" cada vez que ocurra un Flanco de Bajada, un Flanco de Subida o ambos (según se seleccione por parámetro).

- **FLANCO DE SUBIDA** → A través de este parámetro se determina que valor se envía ante un Flanco de Bajada en la entrada.
- **FLANCO DE BAJADA** → A través de este parámetro se determina que valor se envía ante un Flanco de Subida en la entrada.
- **RETARDO "0"** → Tiempo de espera del **ACTinBOX** antes de enviar un "0" por el objeto "**[Interruptor/Sensor] Flanco**".
- **RETARDO "1"** → Tiempo de espera del **ACTinBOX** antes de enviar un "1" por el objeto "**[Interruptor/Sensor] Flanco**".
- **ENVÍO CÍCLICO "0"** → Este parámetro determina el ciclo de envío del valor "0". Es decir, cuando el objeto "**[Interruptor/Sensor] Flanco**" valga "0", enviará su valor al **BUS** constantemente. Si no se desea envío cíclico, elegir el valor "0".
- **ENVÍO CÍCLICO "1"** → Este parámetro determina el ciclo de envío del valor "1". Es decir, cuando el objeto "**[Interruptor/Sensor] Flanco**" valga "1", enviará su valor al **BUS** constantemente. Si no se desea envío cíclico, elegir el valor "0".
- **BLOQUEO** → Este parámetro habilita el objeto "**Bloqueo**". Este objeto se utiliza para deshabilitar la entrada:
  - Al recibir un "1" por este objeto, el **ACTinBOX** hace caso omiso ante cualquier flanco en la entrada.
  - Al recibir un "0" por este objeto, la entrada vuelve a estar habilitada (pero no se tienen en cuenta las acciones realizadas durante el estado deshabilitado)

- **ENLACES INTERNOS** → Esta utilidad, que se encuentra dentro de las funciones de las entradas, permite una independencia completa entre las partes de que se compone el **ACTinBOX**. Gracias a ella, el **ACTinBOX** se comporta como si hubiera una dirección de grupo interna que asociara el objeto de la Entrada y el objeto elegido correspondiente a las otras dos partes del **ACTinBOX** (Salidas y Funciones Lógicas). Estos enlaces se establecen a través de los parámetros **ENLACES INTERNOS** que poseen las funciones de las entradas.

*Nota: Si se establece un enlace interno entre una Entrada de tipo Conmutación y una Salida Individual, se recomienda establecer un enlace interno entre el objeto “Estado” de la Salida y la Entrada (utilizando los parámetros de la Salida), para evitar que se produzcan falsos conmutados.*

## 4. FUNCIONES LÓGICAS

Esta sección del **ACTinBOX** permite realizar operaciones en lógica binaria con datos procedentes del **BUS**, y enviar el resultado a través de otros Objetos de Comunicación específicamente habilitados al efecto en el Actuador. Los datos con los que trabajan las Funciones pueden proceder de 2 sitios diferentes:

- Del **BUS**, a través de **Objetos de Comunicación** especialmente habilitados para estas funciones.
- De **Variables Internas**, donde se guardan resultados parciales de operaciones intermedias.

➤ **FUNCIONES LÓGICAS UTILIZADAS** → Se pueden utilizar hasta 5 Funciones Lógicas distintas, independientes entre sí. Para utilizar cualquiera de ellas, es necesario habilitarlas.

○ FUNCIÓN 1.....5

➤ **Nº DE OBJETOS DE ENTRADA DE DATOS** → Es necesario definir el número de Objetos de Entrada de Datos de cada tipo, necesarios para utilizar entre todas las Funciones.

○ **1 BIT (16 Objetos disponibles)** → Es necesario definir el número de objetos de 1 bit que servirán de entrada de datos en las operaciones realizadas en lógica binaria.

○ **1 BYTE (8 Objetos disponibles)** → Es necesario definir el número de objetos de 1 byte que servirán de entrada de datos en las operaciones realizadas en lógica binaria.

○ **2 BYTES (8 Objetos disponibles)** → Es necesario definir el número de objetos de 2 byte que servirán de entrada de datos en las operaciones realizadas en lógica binaria.

*Nota I: También se ponen a disposición del integrador:*

*- 16 Variables de 1 bit*

*- 8 Variables de 1 byte*

*- 8 Variables de 2 bytes*

*Como variables internas para almacenar resultados parciales de las operaciones.*

**Nota II:** *Hasta que no se han definido el N° de Objetos de entrada de Datos por parámetro, los Objetos de Comunicación asociados a los mismos no aparecerán en el ETS.*

**Nota III:** *Se recomienda definir siempre algún objeto más de los necesarios, ya que una redefinición posterior implica el borrado de las posibles asociaciones de Direcciones de Grupo que ya se hubieran efectuado, con la consiguiente pérdida de tiempo al tener que volver a asociarlas.*

## 4.1.LLAMADA

Cada una de las Funciones Lógicas definidas o habilitadas, dispone de una sección **LLAMADA**, en la que han de seleccionarse los objetos que pueden desencadenar que la función en cuestión sea ejecutada.

Hasta un máximo de 8 Objetos pueden ser seleccionados como desencadenantes de la Función.

**Nota:** *Para que la Función se ejecute, será necesario que al menos uno de los Objetos habilitados en esta sección se actualice. No es necesario que el/los objetos desencadenantes de la función estén incluidos en la misma.*

## 4.2.OPERACIONES

En esta sección se definen las operaciones que componen la función. En cada Función se pueden utilizar hasta 4 Operaciones de cálculo distintas.

- **OPERACIÓN** → Para utilizar una operación, es necesario "**habilitarla**" primero.
- **TIPO** → Existen 4 tipos de operaciones distintas:
  - **Lógica** →: Este tipo de operación trabaja con valores de 1 bit. Se puede elegir entre las siguientes operaciones de tipo lógico **ID, AND, OR, XOR, NOT, NAND, NOR y NXOR**. Todas ellas trabajan con dos operandos (excepto **ID** y **NOT**, que trabajan sólo con uno). Los operandos pueden ser elegidos de entre los **16 objetos de 1 bit**, y las **16 variables internas de 1 bit**. El resultado de la operación de tipo Lógica es también de 1 bit y puede ser almacenado en cualquiera de las 16 variables internas de 1 bit.

- **Aritmética (1 byte/2bytes (entero sin signo)/2bytes (coma flotante)→**: Dependiendo del tipo elegido, estas operaciones trabajan con valores de 1 byte o de 2bytes. Se puede elegir entre las siguientes operaciones Aritméticas: **ID, SUMA, RESTA, MULTIPLICACIÓN, DIVISIÓN, MÁXIMO y MÍNIMO**. Todas ellas trabajan con dos operandos (excepto **ID** que trabaja sólo con uno), que pueden ser elegidos de entre los objetos, las variables o un valor constante elegido por parámetro. El resultado de la operación aritmética será de 1 byte, o 2 bytes (dependiendo de la operación elegida). Este resultado puede ser almacenado en cualquiera de las 8 variables correspondientes.

***Nota:** La operación Aritmética (2 bytes entero sin signo) opera con datos en el rango (0.....65535). Las constantes introducidas en el campo parametrizable correspondiente usan el formato IX (ej. Valor 4000 -> Parámetro 4000)*

***Nota I:** La operación Aritmética (2 bytes coma flotante) opera con datos en el rango (0.....120). Las constantes introducidas en el campo parametrizable correspondiente usan el formato 0.IX (ej. Valor 22.5 -> Parámetro 225)*

***Nota II:** En las Operaciones Aritméticas de 2 bytes, si el resultado excede del rango permitido, éste se convierte al extremo correspondiente. Dividir por "0" no envía nada al BUS*

- **Comparación (1 byte/2bytes (entero sin signo)/2bytes (coma flotante) →**: Estas operaciones trabajan con valores de 1 byte o de 2bytes, dependiendo del tipo elegido. Se puede elegir entre las siguientes operaciones de Comparación: **MAYOR, MAYOR O IGUAL, MENOR, MENOR O IGUAL, DISTINTO e IGUAL**. Todas ellas trabajan con dos operandos, que pueden ser elegidos de entre los objetos, las variables o un valor constante elegido por parámetro. El resultado de la operación es de tipo 1 bit ("1" si se cumple la comparación y "0" si no se cumple). Este resultado puede ser almacenado en cualquiera de las 16 variables de 1 bit disponibles.
- **Conversión (1 bit/1 byte/2bytes (entero sin signo)/2bytes (coma flotante)→**: Permite la conversión entre formatos de los Objetos de Comunicación.

## DESCRIPCIÓN DE FUNCIONES-CONVERSIÓN

A continuación se proporciona información detallada relativa a las funciones de conversión del ACTinBOX CLASSIC:

- **“CONVERTIR”** (1 bit → 1byte)

1bit	1byte
0	00000000
1	00000001

- **“CONVERTIR”** (1bit → 2bytes entero sin signo)

1bit	2bytes entero sin signo
0	00000000 00000000
1	00000000 00000001

- **“CONVERTIR”** (1 bit → 2 bytes coma flotante)

1bit	2 bytes coma flotante
0	0
1	0,1

- **“CONVERTIR”** (1 byte → 1 bit)

1byte	1bit
0	0
1..255	1

- **“CONVERTIR”** (1 byte → 2 bytes entero sin signo)

1byte	2bytes
\$00	\$00 00
\$01	\$00 01
...	...

\$FF	\$00 FF
------	---------

- **“CONVERTIR”** (1 byte → 2 bytes coma flotante)

1byte	2 bytes coma flotante
0	0
1	0.1
255	25.5

*Nota: El límite de conversión se encuentra en 25.5*

- **“CONVERTIR”** (2 bytes entero sin signo → 1 bit)

2bytes entero sin signo	1bit
0	0
1..65535	1

- **“CONVERTIR”** (2 bytes entero sin signo → 1 byte)

2bytes entero sin signo	1byte
\$00 00	\$00
\$00 01	\$01
...	...
\$00 FF	\$FF
> \$00 FF	\$FF

- **“CONVERTIR”** (2 bytes entero sin signo → 2 bytes coma flotante)

2bytes entero sin signo	2 bytes coma flotante
0	0
1	0.1
...	...
1200	120
>1200	120

- **“CONVERTIR”** (2 bytes coma flotante → 1 bit)

2 bytes coma flotante	1bit
0	0
0,1.....120	1

- **“CONVERTIR”** (2 bytes coma flotante → 1 byte)

2 bytes coma flotante	1byte
0	0
0,1... 25,5 .	1..255
> 25,5	255

- **“CONVERTIR”** (2 bytes coma flotante → 2 bytes entero sin signo)

2 bytes coma flotante	2bytes entero sin signo
0	0
0.1	1
...	
120	1200
>120	1200

- **RESULTADO DE LA OPERACIÓN**→ Permite definir la variable donde será guardado el resultado de la operación.

### 4.3.RESULTADO

En esta sección se determina dónde está y qué hacer con el resultado obtenido de las operaciones de la sección anterior.

- **TIPO**→ En primer lugar se determina el tipo de resultado, si es de 1 bit, 1 byte ó 2 bytes (Entero sin signo) / (Coma Flotante).
- **VALOR**→ Es necesario indicar en qué variable se va a almacenar el resultado proveniente de las operaciones de la sección anterior.

*Nota: Tenga en cuenta que las diferentes variables de almacenamiento son variables compartidas para todas las posibles funciones/operaciones, esto supone que una variable utilizada para almacenar el resultado parcial/total de una operación/función, no debería ser usada para almacenar otro resultado distinto.*

- **ENVIO**→ Define bajo qué condiciones se envía el resultado al BUS.
  - **Cambio de Resultado Final**→: El resultado será enviado cada vez que el Resultado Final proveniente de las Operaciones definidas en secciones anteriores cambie.
  - **Cada vez que se ejecuta la función**→: El resultado será enviado cada vez que se ejecute la función.

*Nota: Este parámetro está relacionado con la Sección **LLAMADA** (ver pag N° 35): Efectivamente el resultado será enviado cada vez que se ejecute la función, pero la función sólo será ejecutada cuando al menos uno de los objetos habilitados en la Sección llamada se actualice.*

- **Periódico**→: Esta opción implica que el Resultado será enviado repetidamente cada X segundos (tiempo determinado por el parámetro **PERIODO DE ENVÍO**).
- **RESTRICCIÓN**→ Para las **funciones cuyo resultado es de tipo 1 bit**, se puede restringir el envío del resultado de la función a uno de los dos valores posibles ("0" ó "1"), de modo que sólo se permite la salida al BUS del "0" o del "1", el que se elija como valor no restringido. **Para las funciones cuyo resultado es de tipo 1 byte o 2 bytes**, se puede restringir el envío siguiendo los siguientes parámetros
  - Valores Iguales al de Referencia
  - Valores distintos al de Referencia
  - Valores mayores que el de referencia
  - Valores menores que el de referencia
- **Valor de Referencia**→: Para el **Resultado Tipo = 1 byte**, el posible rango para el valor de referencia es [0.....255]. Para el **Resultado Tipo = 2 bytes**, el posible rango para el valor de referencia es [0.....65535]
- **RETARDO**→ Se puede establecer un tiempo de espera antes del envío del resultado al BUS. Si no se desea establecer RETARDO, elegir el valor "0".
- **ENLACES INTERNOS**→ A través de ellos se puede unir el resultado de la Función con el resto de Objetos de Control del ACTinBOX.
  - **Salidas**→: Si el resultado es de tipo 1 bit, se puede enviar a un objeto de control de 1 bit de cualquier salida.
  - **Canales Persiana**→: Si el resultado es de tipo 1 bit, se puede enviar a un objeto de control de cualquier Canal de Persiana.
  - **Entradas**→: Si el resultado es de tipo 1 bit, se puede enviar a un objeto de control de cualquier Canal de Persiana.
  - **Funciones Lógicas**→: El resultado de la función se puede enviar a cualquiera de los objetos de entrada de Datos de las Funciones Lógicas (y así puede ser reutilizado por cualquier otra función).

## 5. OBJETOS DE COMUNICACIÓN

Los Objetos de Comunicación de la parte de las Funciones Lógicas son de dos tipos

- **DATOS**→ Son los datos procedentes del BUS con los que trabajan las Funciones.
- **RESULTADOS**→ Son los Resultados de las Funciones (uno por cada una). Según el tamaño de estos objetos, se dividen en 3 tipos: 1 bit, 1 byte y 2 bytes.

### 5.1.NOMENCLATURA:

- **OBJETOS DE TIPO DATOS**

[FL] Dato (“tamaño”) “X” donde “tamaño” puede ser 1 bit, 1 byte o 2 bytes; y “X” es el número del Dato (es decir, entre 1 y 16 para los datos de 1 bit, y entre 1 y 8 para los de 1 byte y 2 bytes).

- **OBJETOS DE TIPO RESULTADO**

[FL] RESULTADO Función “X” (“tamaño”). Donde “tamaño” puede ser 1 bit, 1 byte o 2 bytes (depende del tipo de dato que tenga la función como resultado), y “X” es el número de la Función (es decir, entre 1 y 5).

- **VARIABLES INTERNAS**

b1,……, b16 (para las de tipo 1 bit)

n1,……, n8 (para las de tipo 1 byte)

x1,……, x8 (para las de tipo 2 bytes)

## ANEXO I: OBJETO DE COMUNICACIÓN

SECCIÓN	NÚMERO	TAMAÑO	ENT/SAL	FLAGS	VALORES			NOMBRE	DESCRIPCIÓN
					RANGO	1ª VEZ	RESET		
SALIDAS INDIVIDUALES	0-3	1bit	I	W	0/1	Indiferente	Indiferente	[Sx] ON/OFF	N.A. (0=Abrir Relé; 1=Cerrar) N.C. (0=Cerrar Relé; 1=Abrir)
	4-7	1 bit	O	R-T	0/1	Seg. Param	Seg. Param	[Sx] Estado	0=Salida OFF; 1=Salida ON
	8-11	1 bit	I	W	0/1	Indiferente	Indiferente	[Sx] Temporización	0=OFF Temporizado;1=ON Tempor.
	12-15	1 bit	I	W	0/1	Indiferente	Indiferente	[Sx] Intermittencia	1=Intermittencia; 0=Fin Intern.
	16-19	1byte	I	W	0-63 128-192	Indiferente	Indiferente	[Sx] Escenas	0-63(Esc. 1-64);128-191(Grab.)
	20-23	1 bit	I	W	0/1	0	Anterior	[Sx] Bloqueo	1=Bloquear; 0=Desbloquear
	24-27	1bit	I	W	0/1	Seg. Param	Anterior	[Sx] Alarma	1=Alarma; 0=No Alarma 0=Alarma; 1=No Alarma
28-31	1 bit	I	W	0/1	Indiferente	Indiferente	[Sx] Enclavamiento	Alarma=0+Encl.=1 -> Fin Alarma	
SALIDAS DE PERSIANA	32-33	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Subir/Bajar	0=Subir Persiana; 1=Bajar
	34-35	1bit	I	W	0/1	Indiferente	Indiferente	[Cx] Parar	0 ó 1 = Parar Persiana
								[Cx] Parar/Paso	0=Parar/PasoArriba; 1=Par/P.Ab
	36-37	1byte	O	R-T	0/1	0	Calcular	[Cx] Posición Actual	0=0%=Arriba; 255=100%=Abajo
	38-39	1byte	I	W	0-255	Indiferente	Indiferente	[Cx] Posición Deseada	0=0%=Arriba; 255=100%=Abajo
	40-41	1byte	I	W	0-255	Indiferente	Indiferente	[Cx] Escenas	0-63(Esc. 1-64);128-191(Grab.)
	42-43	1 bit	I	W	0-63 128-192	0	Anterior	[Cx] Bloqueo	1=Bloquear; 0=Desbloquear
	44-45	1bit	I	W	0/1	Seg. Param.	Anterior	[Cx] Alarma	1=Alarma; 0=No Alarma 0=Alarma; 1=No Alarma
								[Cx] Alarma 2	1=Alarma; 0=No Alarma 0=Alarma; 1=No Alarma
	48-49	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Enclavamiento	Alarma=0+Encl.=1 -> Fin Alarma
	50-51	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Movimiento Invertido	0=Bajar Persiana; 1=Subir
	52-53	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Posicionamiento Directo	1=Ir a Posición; 0=Nada
	54-55	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Posicionamiento Directo 2	1=Ir a Posición 2; 0=Nada
56-57	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Grabar Posición	1=Grabar Posición; 0=Nada	
58-59	1 bit	I	W	0/1	Indiferente	Indiferente	[Cx] Grabar Posición 2	1=Grabar Posición 2; 0=Nada	
ENTRADAS	60-65	1 bit	I	W	0/1	0	Anterior	[Ex] Bloqueo	1=Entrada Bloqueada; 0=Libre
	66-71	1 bit	I	R-W-T	0/1	0	Anterior	[Ex] [Puls. Corta] "0"	Puls. Corta -> Envío de un "0"
								[Ex] [Puls. Corta] "1"	Puls. Corta -> Envío de un "1"
[Ex] [Puls. Corta] Conmutar								Puls. Corta -> Conmutación 0/1	
[Ex] [Puls. Corta] Subir Persiana								Puls. Corta -> Envía 0 (Subir)	
[Ex] [Puls. Corta] Bajar Persiana								Puls. Corta -> Envía 1 (Bajar)	
[Ex] [Puls. Corta] Subir/Bajar Persiana								Puls. Corta -> Conmutación 0/1	
[Ex] [Puls. Corta] Parar Persiana / Paso Arriba								Puls. Corta -> Envía 0	
[Ex] [Puls. Corta] Parar Persiana / Paso Abajo	Puls. Corta -> Envía 1								
[Ex] [Puls. Corta] Parar Persiana / Paso Conmutado	Puls. Corta -> Conmutación 0/1								
[Ex] [Puls. Corta] Dimmer ON	Puls. Corta -> Envía 1 (ON)								
[Ex] [Puls. Corta] Dimmer OFF	Puls. Corta -> Envía 0 (OFF)								
[Ex] [Puls. Corta] Dimmer ON/OFF	Puls. Corta -> Conmutación 0/1								
[Ex] [Interruptor/Sensor] Flanco	Flanco -> Envío de "0" ó "1"								
72-77	4bits	O	R-T	0-15	0	Anterior	[Ex] [Puls. Corta] Aumentar Luz [Ex] [Puls. Corta] Disminuir Luz [Ex] [Puls. Corta] Aumentar/Disminuir	PCorta->MásLuz; PCorta->Para PCorta->MenosLuz; PCorta->Para PCorta-> +/- Luz; PCorta->Para	

								Luz	
	78-83	1byte	O	R-T	0-63 128-192	Indiferente	Indiferente	[Ex] [Puls. Corta] Ejecutar Escena [Ex] [Puls. Corta] Grabar Escena	Puls. Corta -> Envío de 0-63 Puls. Corta -> Envía 128-191

ENTRADAS	84-89	1bit	O	R-W-T	0/1	0	Anterior	[Ex] [Puls. Larga] "0" ... [Ex] [Puls. Larga] Dimmer ON/OFF	Puls. Larga -> Envío de un "0" ... Puls. Larga -> Conmutación 0/1
	90-95	4bits	O	R-T	0-15	0	Anterior	[Ex] [Puls. Larga] Aumentar Luz [Ex] [Puls. Larga] Disminuir Luz [Ex] [Puls. Larga] Aumentar/Disminuir Luz	PLarga->MásLuz; PCorta->Para PLarga->MenosLuz; PLarga->Para PLarga-> +/- Luz; PLarga->Para
	96-101	1byte	O	R-T	0-63 128-192	Indiferente	Indiferente	[Ex] [Puls. Larga] Ejecutar Escena [Ex] [Puls. Larga] Grabar Escena	Puls. Larga -> Envío de 0-63 Fl. Subida -> Envía 128-191
FUNCIONES LÓGICAS	102-117	1bit	I	W	0/1	0	Anterior	[FL] Dato (1bit) 1 ... [FL] Dato (1bit) 16	Dato de entrada binario (0/1) ... Dato de entrada binario (0/1)
	118-125	1byte	I	W	0-255	0	Anterior	[FL] Dato (1byte) 1 ... [FL] Dato (1byte) 8	Dato de entr. de 1byte (0-255) ... Dato de entr. de 1byte (0-255)
	126-133	2bytes	I	W	0-FFFF	0	Anterior	[FL] Dato (2bytes) 1 ... [FL] Dato (2bytes) 8	Dato de entr. de Temperatura ... Dato de entr. de Temperatura
	134-138	1bit	O	R-T	0/1	0	Anterior	[FL] RESULTADO Función 1 (1bit) ... [FL] RESULTADO Función 5 (1bit)	Resultado de la FUNCIÓN 1 ... Resultado de la FUNCIÓN 5
	139-143	1byte	O	R-T	0-255	0	Anterior	[FL] RESULTADO Función 1 (1byte) ... [FL] RESULTADO Función 5 (1byte)	Resultado de la FUNCIÓN 1 ... Resultado de la FUNCIÓN 5
	144-148	2bytes	O	R-T	0°C-120°C	25°C	Anterior	[FL] RESULTADO Función 1 (2bytes) ... [FL] RESULTADO Función 5 (2bytes)	Resultado de la FUNCIÓN 1 ... Resultado de la FUNCIÓN 5
RESET	149	1bit	O	T	0	0	O	Reset 0	Vuelve la Tensión->Envía un 0
	150	1bit	O	T	1	1	1	Reset 1	Vuelve la Tensión->Envía un 1



**¡HAZTE USUARIO!**  
**<http://zennio.zendesk.com>**  
**SOPORTE TÉCNICO**