DOCUMENTACIÓN ESPECÍFICA

Funciones Lógicas X10

Módulo Zennio

de 10 funciones lógicas

Edición: 1.a

ÍNDICE

1. Introducción 3
1.1. Módulo de Funciones lógicas 3
1.2. Dispositivos Zennio con módulo X10 4
2. Configuración 5
2.1. Funcionamiento general 5
2.2. Objetos de entrada 6
2.3. Variables internas 6
2.4. Objeto de Gate 6
2.5. Objetos de Llamada 7
2.6. Operaciones 7
2.7. Resultado 7
3. Parametrización ETS 8
3.1. Pantalla general 8
3.2. Llamada 9
3.2.1. Objeto de Gate 9
3.2.2. Objetos de llamada 11
3.3. Operaciones 12
3.3.1. Tipo de operaciones 13
3.4. Resultado 14
3.5. Objetos de comunicación 16
Anexo I. Operaciones lógicas 17
Anexo II. Operaciones de conversión 19

1. INTRODUCCIÓN

1.1. MÓDULO DE FUNCIONES LÓGICAS

Ciertos dispositivos Zennio, como los actuadores, llevan implementado un módulo de funciones lógicas que les permiten realizar operaciones en **lógica binaria** con datos procedentes del bus KNX y enviar el resultado a través de otros objetos de comunicación, de diferentes longitudes (1 bit, 1 byte y 2 bytes), específicamente habilitados a tal efecto.

Los datos con los que trabajan las funciones lógicas pueden proceder de dos sitios diferentes:

- Del **bus KNX**, a través de objetos de comunicación especialmente habilitados para las funciones lógicas.
- De **variables internas**, donde se guardan resultados parciales de operaciones intermedias.

Existen dos módulos de funciones lógicas:

- **Módulo de 5 funciones lógicas (X5).** Permite habilitar y configurar hasta 5 funciones lógicas diferentes, independientes entre sí, que podrán realizar hasta 4 operaciones cada una.
- **Módulo de 10 funciones lógicas (X10).** Permite habilitar y configurar hasta 10 funciones lógicas diferentes, independientes entre sí, que podrán realizar hasta 4 operaciones cada una.

En el presente documento se explicará en detalle el módulo de 10 funciones lógicas (X10).

1.2. DISPOSITIVOS ZENNIO CON MÓDULO X10

En la tabla 1.1 pueden conocerse los dispositivos Zennio (nombre del dispositivo y versión del programa de aplicación) que actualmente tienen incorporado el módulo de 10 funciones lógicas (X10) que se describe en el presente documento. Esta tabla se irá actualizando a medida que aumente el número de versiones y de dispositivos en los que se implemente este módulo.

Dispositivo	Versión del Programa de Aplicación
MAXinBOX 16	1.0
MAXinBOX 16	2.0
ACTinBOX MAX6	3.0
ACTinBOX QUATRO	2.0

Tabla 1.1. Dispositivos Zennio con módulo de funciones lógicas X10

Nota: *Para una información más detallada sobre la parametrización del módulo X10 de funciones lógicas en los dispositivos Zennio, por favor consultar la sección de* ***"Parametrización ETS"****, en el apartado 3 de este documento.*

2. CONFIGURACIÓN

A continuación se presentan una serie de conceptos generales relacionados con el módulo X10 de funciones lógicas.

2.1. FUNCIONAMIENTO GENERAL

El funcionamiento del módulo de funciones lógicas se resume en los siguientes puntos:

- **LLAMADA**. El primer paso es llamar a la función que se desea ejecutar. Para ello, se habilitarán uno o varios objetos de comunicación (hasta un total de 8). Cada vez que su valor se actualice, se ejecutará la función lógica correspondiente.

- **OPERACIONES**. Cada función lógica puede realizar hasta un total de 4 operaciones de cálculo diferentes. Para cada una de ellas, hay que definir el tipo de la operación, los operandos (valores con los que se realizan las operaciones, que pueden ser objetos de comunicación o variables internas) y la variable interna donde se almacenará el resultado parcial de cada operación.

- **RESULTADO**. Finalmente, el resultado de la función se recoge de una variable interna y se envía a través de un objeto de comunicación al bus KNX. Pueden configurarse distintas opciones para este envío.

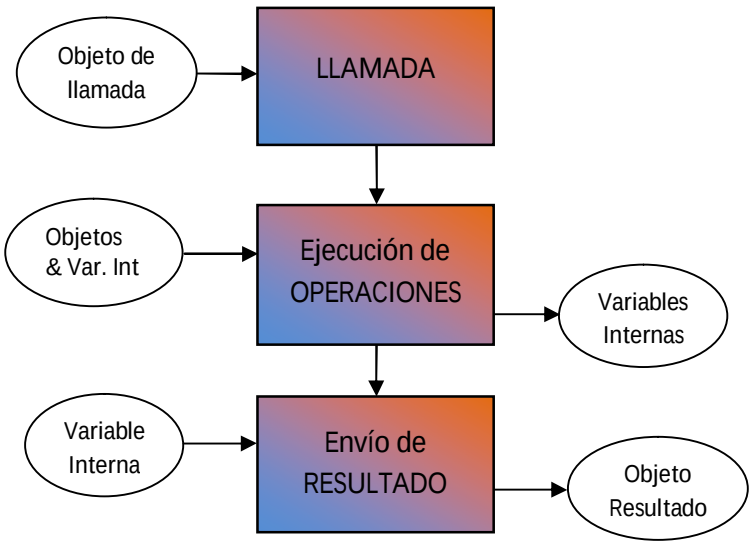


Figura 2.1. Funcionamiento general módulo funciones lógicas

2.2. OBJETOS DE ENTRADA

Los objetos de comunicación específicos para funciones lógicas pueden ser de tamaño 1 bit, 1 byte o 2 bytes. Será necesario habilitar uno a uno todos los objetos de cada tipo que sean necesarios como entrada de datos en las operaciones en lógica binaria a realizar.

Para el módulo X10 de funciones lógicas se pueden habilitar los siguientes objetos:

- **32 objetos** de comunicación de **1 bit**.
- **16 objetos** de comunicación de **1 byte**.
- **16 objetos** de comunicación de **2 bytes**.

2.3. VARIABLES INTERNAS

Las variables internas son variables propias del módulo de funciones lógicas, que maneja de manera interna y que permiten almacenar resultados parciales de las operaciones intermedias.

El módulo X10 de funciones lógicas dispone de las siguientes variables internas:

- **32 variables** para almacenar resultados de **1 bit**, nombradas como [b1 … b32]
- **16 variables** para almacenar resultados de **1 byte**, nombradas como [n1 … n16]
- **16 variables** para almacenar resultados de **2 bytes**, nombradas como [x1 … x16]

2.4. OBJETO DE GATE

Para cada una de las 10 funciones lógicas existentes se puede utilizar uno de los 32 objetos de entrada de 1 bit disponibles como ***gate*** de la función. Este objeto permite activar o desactivar la ejecución de la función lógica al recibir el valor binario correspondiente (“0” ó “1”, según lo parametrizado).

Ver apartado 3.2 para más información sobre este objeto.

2.5. OBJETOS DE LLAMADA

Los objetos de llamada son aquellos que, con una actualización de su valor, desencadenan la ejecución de la función lógica en donde se hayan habilitado.

Pueden configurarse hasta un total de 8 objetos de llamada (de cualquier tipo, de entre los objetos de entrada establecidos) por cada función lógica habilitada.

Los objetos de llamada no tienen por qué estar incluidos en la función lógica, es decir, pueden no utilizarse como valores en las operaciones que realiza la función.

2.6. OPERACIONES

Las operaciones son un conjunto de pasos que permiten obtener una serie de resultados a partir de los objetos de entrada establecidos.

El módulo X10 de funciones lógicas permite habilitar y configurar hasta 4 operaciones de cálculo diferentes por cada función lógica.

Las operaciones que pueden realizarse son de 4 tipos:

- **Lógica** (AND/OR/NOT/...)
- **Aritmética** (Suma/División/Máximo/…)
- **Comparación** (Mayor/Menor/Distinto/…)
- **Conversión** (a 1 bit /a 1 byte /a 2 bytes)

Las opciones asociadas a cada tipo de operación se verán con más detalle en el apartado 3.3.1.

2.7. RESULTADO

Los resultados son los valores que enviará cada función lógica una vez realizadas todas las operaciones definidas.

Se puede establecer cuándo y cómo se realizará el envío de dichos resultados al bus KNX, parametrizando un envío periódico, o con un determinado retardo, por ejemplo.

Pueden ser de tres tipos: resultados de 1 bit, de 1 byte y de 2 bytes.

3. PARAMETRIZACIÓN ETS

A continuación se presenta la configuración del módulo X10 de funciones lógicas a través de la herramienta ETS.

Las capturas de pantalla proceden del actuador MAXinBOX 16 (versión 2.0), pero todas las opciones, parámetros y el aspecto de la ventana de configuración del módulo de funciones lógicas en ETS son iguales para todos los dispositivos Zennio que lo incorporan (ver Tabla 1.1).

3.1. PANTALLA GENERAL

En el programa de aplicación de los dispositivos con módulo X10 existe una pantalla exclusiva para la configuración del mismo:

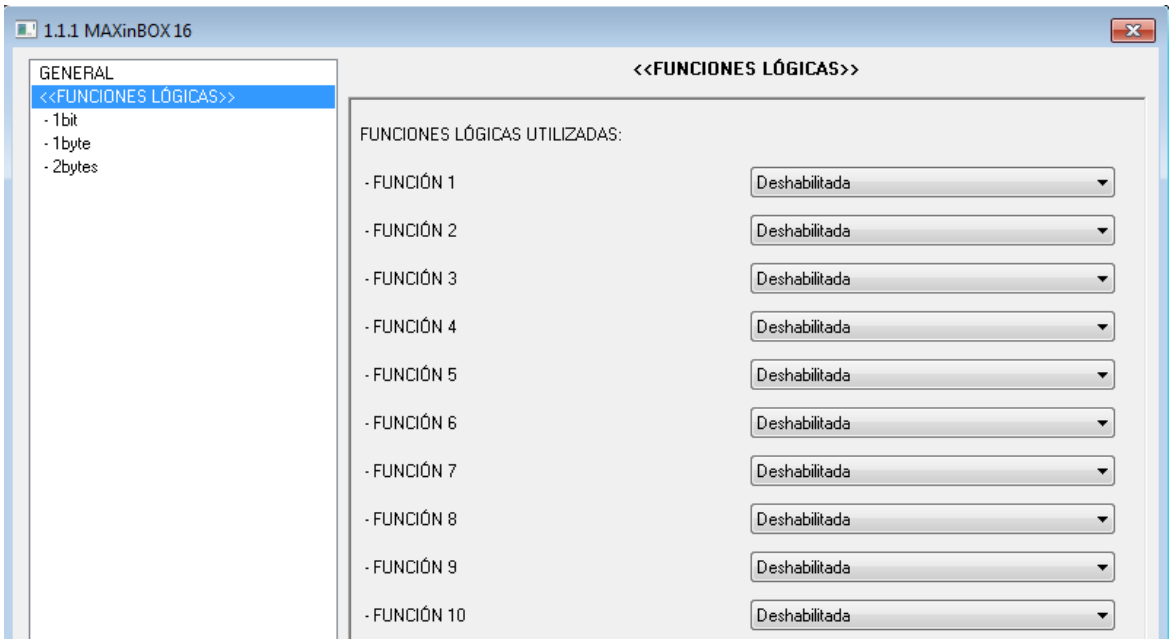


Figura 3.1. Pantalla general funciones lógicas

Como puede verse en la Figura 3.1, las 10 funciones lógicas disponibles aparecen deshabilitadas por defecto.

En las pestañas “1bit”, “1byte” y “2bytes” se podrán habilitar, uno a uno, los objetos de comunicación de distintos tamaños que vayan a usarse como entrada de datos para las funciones.

Como se vio en el apartado de Configuración, podrán habilitarse hasta 32 objetos de 1 bit, 16 objetos de 1 byte y otros 16 de 2 bytes.

Al habilitar cada una de las funciones lógicas, irán apareciendo una serie de pantallas que permitirán configurar las opciones de Llamada, Operaciones y Resultado correspondientes, y que se explicarán en detalle a continuación.

3.2. LLAMADA

Cada una de las funciones lógicas habilitadas dispone de la sección **Llamada**, en la que deberán seleccionarse los objetos de llamada de dicha función, así como habilitar el uso del objeto de ***gate*** de la función.

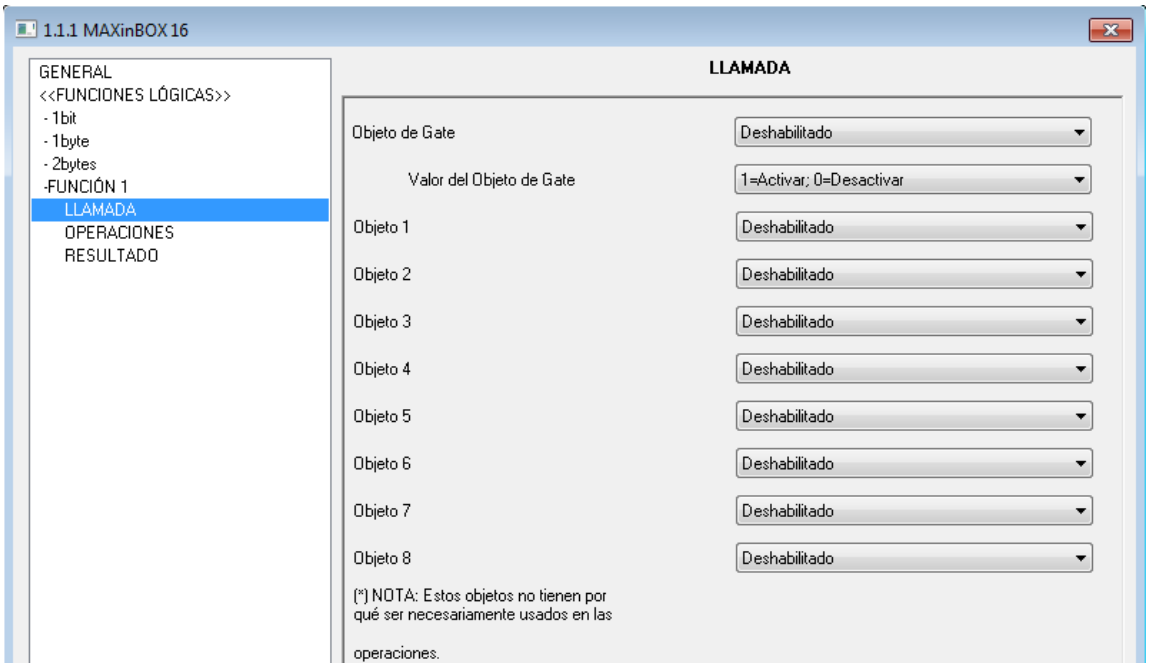
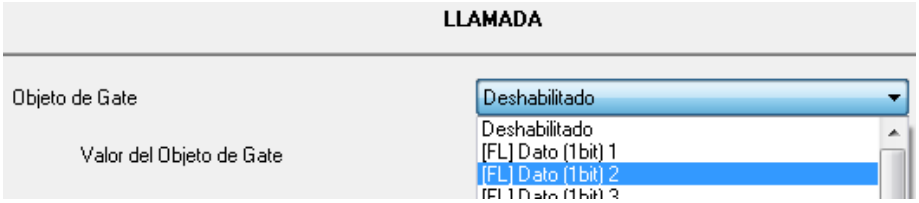


Figura 3.2. Pantalla de “Llamada”

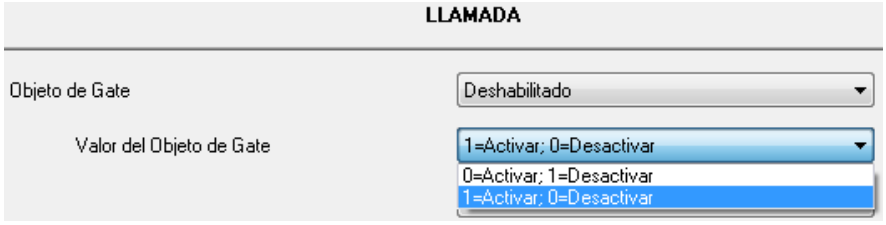
3.2.1. OBJETO DE GATE

En caso de querer controlar la ejecución de las funciones mediante objeto, será necesario habilitar el **objeto de *gate***, seleccionando en el menú desplegable uno de los 32 objetos de 1 bit disponibles.



Puede habilitarse un objeto de *gate* por función, por lo que podrá haber hasta 10 (iguales o diferentes). Si no se desea disponer de dicho objeto, habrá que seleccionar la opción "Deshabilitado"; en este caso, las funciones se ejecutarán al ser llamadas mediante los objetos de llamada correspondientes, porque siempre se encontrarán listas para ejecutarse.

Se podrá establecer el valor que deberá tomar el objeto para poder activar o desactivar la ejecución de la función correspondiente. Las opciones disponibles son: [0=Activar; 1=Desactivar] o [1=Activar; 0=Desactivar].



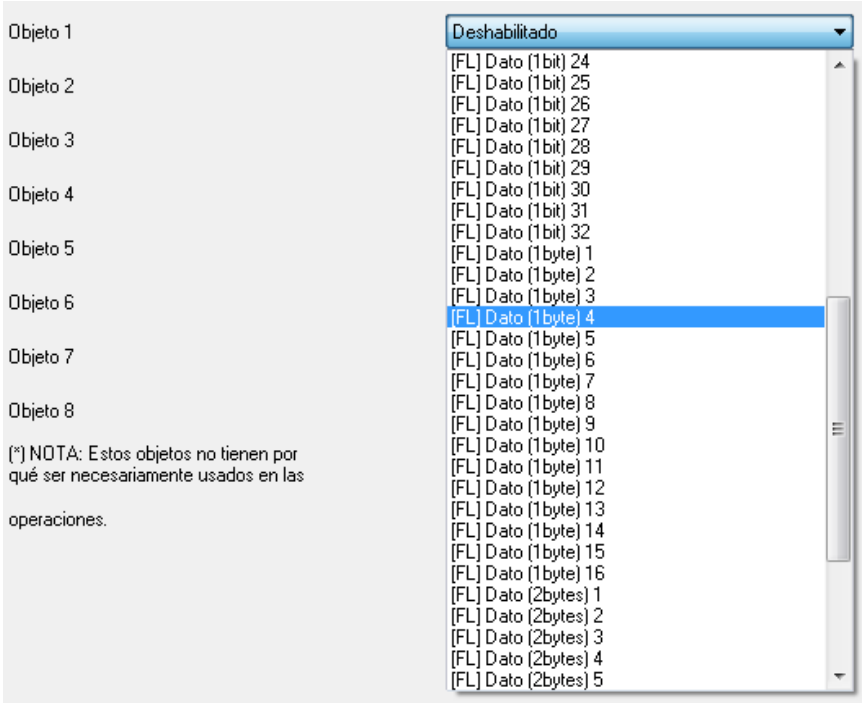
En el momento en que la función correspondiente sea llamada, ésta se ejecutará (siempre y cuando el objeto de *gate* tenga el valor de activación correspondiente). Cuando el objeto de *gate* tenga el valor complementario (valor de desactivación), la ejecución de la función quedará deshabilitada (la función no podrá ser llamada hasta que el objeto tome el valor de activación).

Nota I: *Tras una descarga del programa de aplicación, el objeto de gate está deshabilitado (valor "0" ó "1", según lo parametrizado). Tras un fallo de tensión, se mantendrá el valor que tuviera antes del mismo.*

Nota II: *Tener en cuenta que si se selecciona que el objeto de gate sea el mismo que uno de los utilizados como objetos de entrada en la función, debe prestarse especial atención al valor con que se vaya actualizando, pues podría habilitar/deshabilitar la función en un momento no deseado.*

3.2.2. OBJETOS DE LLAMADA

Los objetos de llamada son los objetos cuya actualización desencadena la ejecución de la función lógica. Pueden seleccionarse hasta 8 objetos de llamada diferentes, de entre los 64 disponibles, como objetos de llamada de función.



Para que la función se ejecute, será necesario que al menos uno de los objetos habilitados en esta sección actualice su valor y lo envíe al bus KNX. En caso de haber habilitado un objeto de *gate*, este también deberá recibir el valor de activación para que la función se ejecute al ser llamada.

3.3. OPERACIONES

En esta sección se definen las operaciones que componen la función. Para cada función lógica habilitada pueden configurarse hasta 4 operaciones de cálculo distintas.

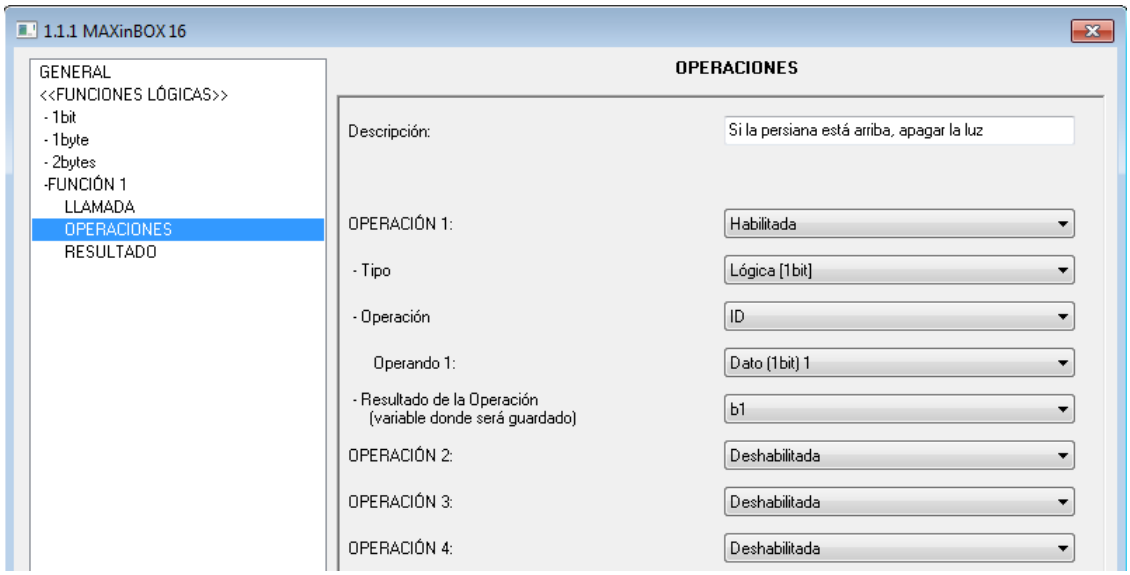


Figura 3.3. Pantalla de “Operaciones”

En el campo **“Descripción”** puede incluirse una breve descripción (espacio para 100 caracteres) de la función lógica: cuál es su finalidad, qué operaciones realiza, qué resultado se obtiene, etc.

Para poder parametrizar una operación, es necesario habilitarla previamente. Al hacerlo, aparecen las siguientes opciones:

- **Tipo:** las funciones lógicas pueden realizar 4 tipos de operaciones distintas. Ver apartado 3.3.1 para conocer los tipos de operaciones disponibles.

- **Operación:** en función del tipo escogido, podrán realizarse diferentes operaciones, seleccionables en este campo. En consecuencia, se habilitarán uno o dos **operandos**, que pueden ser: objetos de comunicación (objetos de entrada), variables internas o valores constantes.

- **Resultado operación:** se elige la variable interna, del tamaño correspondiente, en donde se guardará el resultado parcial de la operación.

3.3.1. TIPO DE OPERACIONES

- **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** (ver Anexo I. Operaciones Lógicas, para conocer en detalle las operaciones lógicas que realizan cada una de ellas).

Todas trabajan con dos operandos (Operando 1 y Operando 2), excepto ID y NOT, que trabajan sólo con uno (Operando 1). Los operandos, como ya se adelantó en el apartado anterior, pueden ser elegidos de entre los 32 objetos de 1 bit disponibles (se muestran los 32, pero tener en cuenta que sólo sirven aquellos que hayan sido habilitados previamente como objetos de entrada) y las 32 variables internas de 1 bit (b1…b32).
El resultado de la operación de tipo Lógica es de 1 bit y puede ser almacenado en cualquiera de las 32 variables internas de 1 bit (b1…b32).

- **Aritmética [1 byte/2 bytes (entero sin signo)/2 bytes (coma flotante)]:** dependiendo del tipo elegido, estas operaciones trabajan con valores de 1 byte o de 2 bytes (entero sin signo o coma flotante). Se puede elegir entre las siguientes operaciones aritméticas: **ID**, **SUMA**, **RESTA**, **MULTIPLICACIÓN**, **DIVISIÓN**, **MÁXIMO** y **MÍNIMO**.

Todas ellas trabajan con dos operandos (Operando 1 y Operando 2), excepto ID, que trabaja sólo con uno (Operando 1). Estos operandos pueden ser elegidos de entre los 16 objetos de 1 byte/2 bytes (siempre que hayan sido habilitados previamente como objetos de entrada), las 16 variables internas de 1 byte (n1…n16), las 16 de 2 bytes (x1…x16) o un valor constante definido por parámetro (valor entre 0 y 255 para el caso de aritmética de 1 byte, entre 0 y 65535 para 2 bytes entero sin signo y entre 0 y 120.0 para 2 bytes coma flotante).
El resultado de la operación aritmética será de 1 byte o de 2 bytes (dependiendo de la operación elegida) y puede ser almacenado en cualquiera de las 16 variables internas de 1 byte (n1…n16) o en cualquiera de las de 2 bytes (x1…x16).

Nota I: *En las operaciones aritméticas de 2 bytes, si el resultado correspondiente excede del rango permitido, éste se convierte al extremo correspondiente.*

Nota II: *Una división entre "0" no genera ningún envío al bus KNX.*

- **Comparación [1 byte/2 bytes (entero sin signo)/2 bytes (coma flotante)]:** dependiendo del tipo elegido, estas operaciones trabajan con valores de 1 byte o de 2 bytes (entero sin signo o coma flotante). 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 (Operando 1 y Operando 2), que pueden ser elegidos de entre los 16 objetos de 1 byte/2 bytes (siempre que hayan sido habilitados previamente como objetos de entrada), las 16 variables internas de 1 byte (n1…n16), las 16 de 2 bytes (x1…x16) o un valor constante definido por parámetro (valor entre 0 y 255 para el caso de aritmética de 1 byte, entre 0 y 65535 para 2 bytes entero sin signo y entre 0 y 1200, en décimas, para 2 bytes coma flotante).

El resultado de la operación aritmética es de 1 bit (valor "1" si se cumple la comparación y valor "0" si no se cumple). Este resultado puede ser almacenado en cualquiera de las 32 variables internas de 1 bit (b1…b32).

- **Conversión [1 bit/1 byte/2 bytes (entero sin signo)/2 bytes (coma flotante)]:** permite la conversión entre formatos de los objetos de comunicación. Todas las posibles conversiones se encuentran recogidas en el Anexo II. Operaciones de conversión del presente documento. Todas ellas trabajan con un único operando (Operando 1), a elegir entre los diferentes objetos de comunicación habilitados como objeto de entrada o alguna de las variables internas, en función del tipo de conversión. El tipo de resultado también dependerá del tipo de conversión.

3.4. RESULTADO

En esta sección se determina en qué variable se guarda el resultado de la función lógica obtenido de las operaciones previas y de qué manera se realizará su envío al bus KNX.

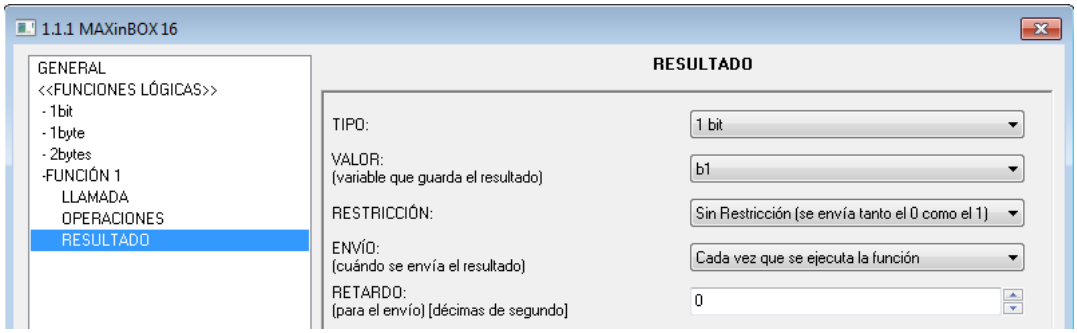


Figura 3.4. Pantalla de "Resultado"

Se muestran las siguientes opciones de configuración:

- **Tipo:** se podrá seleccionar el tipo del resultado final, si es de 1 bit, 1 byte, 2 bytes (entero sin signo) o 2 bytes (coma flotante).

- **Valor:** permite seleccionar la variable interna (en función del tipo del resultado) donde se almacena el resultado final de las funciones lógicas, el que será enviado al bus KNX.

- **Restricción:** puede restringirse el envío del resultado, es decir, la función será la encargada de decidir si envía el resultado o no al bus, en función del valor de este. Para resultados de 1 bit, se puede restringir el envío del resultado de la función a uno de los dos valores posibles (que sólo se envíe el "0" o que sólo se envíe el "1"). Para resultados de 1 byte o 2 bytes, se puede realizar las siguientes restricciones:

  - **Valores iguales al de referencia**
  - **Valores distintos al de referencia**
  - **Valores mayores al de referencia**
  - **Valores menores al de referencia**

Justo debajo, aparecerá un nuevo parámetro donde escribir el valor de referencia deseado. Para resultados de 1 byte, podrá establecerse un valor entre [0-255]; para resultados de 2 bytes (entero sin signo), podrá establecerse un valor entre [0-65535] y para resultados de 2 bytes (coma flotante), podrá establecerse un valor entre [0-1200, en décimas].

- **Envío:** en este campo se define bajo qué condiciones se enviará el resultado al bus KNX, pudiendo elegir entre:

  - **Cada vez que se ejecuta la función**. Este parámetro está relacionado con la sección "Llamada" (ver apartado 3.2), ya que la función se ejecutará siempre que alguno de los objetos habilitados en "Llamada" se actualice.

  - **Cambio de resultado final.** El resultado se enviará al bus KNX cada vez que el resultado final proveniente de las operaciones definidas previamente cambie su valor.

  - **Periódico.** Esta opción implica que el resultado será enviado al bus repetidamente cada "x" segundos (definidos en el campo "Periodo de envío" que se despliega al seleccionar esta opción).

- **Retardo:** se puede establecer un tiempo de espera antes del envío del resultado al bus KNX. Si no se desea un envío con retardo, deberá seleccionarse el valor 0 en este campo.

Tener en cuenta las siguientes consideraciones:

- Si por el objeto de habilitación se recibe el valor relativo a la deshabilitación de la función (“0” ó “1”, según lo parametrizado. Ver apartado 3.2) durante un envío periódico del resultado, éste se verá interrumpido, y no volverá a enviarse hasta que la ejecución de la función sea habilitada de nuevo.

- Si este valor de deshabilitación se recibe mientras se espera un resultado con retardo, el envío de resultado se interrumpirá y la función no volverá a ejecutarse hasta recibir el valor de habilitación por el objeto correspondiente y vuelva a ser llamada.

3.5. OBJETOS DE COMUNICACIÓN

Los objetos de comunicación del módulo de funciones lógicas son de dos tipos:

- **Datos:** procedentes del bus KNX, con los que trabajan las funciones. La nomenclatura usada en ETS para este tipo de datos es la siguiente:

  - **[FL] Dato (“*tamaño*”) “Y”**: donde “***tamaño***” puede ser “1 bit”, “1 byte” o “2 bytes” e “**Y**” es el número del dato, es decir, de 1-32 para los objetos de 1 bit y entre 1-16 para los objetos de 1 byte y 2 bytes.

- **Resultados:** resultados de las operaciones realizadas por las funciones. Un resultado por función. La nomenclatura usada en ETS para este tipo de datos es la siguiente:

  - **[FL] RESULTADO Función “X” (“*tamaño*”)**: donde “***tamaño***” puede ser “1 bit”, “1 byte” o “2 bytes” y “**X**” es el número de la función correspondiente, valor entre 1 y 10.

El número concreto asociado a cada uno de estos objetos de comunicación relativos a las funciones lógicas varía en cada dispositivo Zennio en que se implementa el módulo X10. Para conocerlo, consultar los manuales de usuario correspondientes, el Anexo de “Objetos de Comunicación”, disponibles en: http://www.zennio.com.

ANEXO I. OPERACIONES LÓGICAS

En este Anexo se describen las distintas operaciones lógicas que pueden realizarse con los módulos de funciones lógicas Zennio (Ver apartado 3.3.1 para información sobre parametrización).

- **ID (Identidad)**

Operando 1	Resultado
0	0
1	1

- **AND (Multiplicación Lógica)**

Operando 1	Operando 2	Resultado
0	0	0
0	1	0
1	0	0
1	1	1

- **OR (Suma Lógica)**

Operando 1	Operando 2	Resultado
0	0	0
0	1	1
1	0	1
1	1	1

- **XOR (OR-Exclusiva)**

Operando 1	Operando 2	Resultado
0	0	0
0	1	1
1	0	1
1	1	0

- **NOT (Negación)**

Operando 1	Resultado
0	1
1	0

- **NAND (AND Negada)**

Operando 1	Operando 2	Resultado
0	0	1
0	1	1
1	0	1
1	1	0

- **NOR (OR Negada)**

Operando 1	Operando 2	Resultado
0	0	1
0	1	0
1	0	0
1	1	0

- **NXOR (XOR Negada)**

Operando 1	Operando 2	Resultado
0	0	1
0	1	0
1	0	0
1	1	1

ANEXO II. OPERACIONES DE CONVERSIÓN

En este Anexo se describen las distintas operaciones de conversión que pueden realizarse con los módulos de funciones lógicas Zennio (Ver apartado 3.3.1 para información sobre parametrización).

- **Conversión [1 bit → 1 byte]**

Operando 1	Resultado
0	00000000
1	00000001

- **Conversión [1 bit → 2 bytes (entero sin signo)]**

Operando 1	Resultado
0	$00 00
1	$00 01

- **Conversión [1 bit → 2 bytes (coma flotante)]**

Operando 1	Resultado
0	0
1	0.1

- **Conversión [1 byte → 1 bit]**

Operando 1	Resultado
0	0
1-255	1

- **Conversión [1 byte → 2 bytes (entero sin signo)]**

Operando 1	Resultado
0	$00 00
1	$00 00
…	…
255	$00 FF

- **Conversión [1 byte → 2 bytes (coma flotante)]**

Operando 1	Resultado
0	0
1	0.1
…	…
255	25.5

- **Conversión [2 bytes (entero sin signo) →1 bit]**

Operando 1	Resultado
0	0
1-65535	1

- **Conversión [2 bytes (entero sin signo) →1 byte]**

Operando 1	Resultado
0	0
1	1
…	…
255	255
>255	255

- **Conversión [2 bytes (entero sin signo) →2 bytes (coma flotante)]**

Operando 1	Resultado
0	0
1	0.1
…	…
1200	120.0
>1200	120.0

- **Conversión [2 bytes (coma flotante) →1 bit]**

Operando 1	Resultado
0	0
0.1-120.0	1

- **Conversión [2 bytes (coma flotante) →1 byte]**

Operando 1	Resultado
0	0
0.1-25.5	1-255
…	…
>25.5	255

- **Conversión [2 bytes (coma flotante) →2 bytes (entero sin signo)]**

Operando 1	Resultado
0	0
0.1	1
…	…
120.0	1200

¡HAZTE USUARIO!

http://zennio.zendesk.com

SOPORTE TÉCNICO

ZENNIO DOCUMENTACIÓN TÉCNICA