

Descripción de Servicio

Alarm Control - HTTP Software Interface

Índice

1. Introducción al servicio
2. Condiciones de uso
3. Flujo de trabajo sugerido
4. Documentación de API
5. Recomendaciones de implementación



Contenido

1. Introducción al servicio

"Alarm Control - HTTP Software Interface", "HSI" en adelante, es un servicio web pensado para que sea posible la interacción con los sistemas de alarma que dispongan de un comunicador IP400 a través de una interfaz HTTP.

La conexión al servicio se realiza únicamente a través de HTTPS, lo que garantiza la privacidad de los datos transmitidos.

Este servicio no es de carácter público, sin embargo, aún siendo su documentación expuesta, no representaría ningún tipo de amenaza para la privacidad y seguridad de sus usuarios, ya que para poder utilizarlo, es necesario contar con las credenciales "applD" y "appKey", provistas de forma segura e independiente para cada usuario del servicio.

2. Condiciones de uso

Es de vital importancia que las credenciales que representan al usuario de HSI no sean comprometidas o compartidas de forma pública. El mal uso del servicio podría llevar a una inhabilitación temporal automática, que luego de someterse a una pericia podría traducirse en una inhabilitación permanente del usuario.

Si bien no hay una limitación comercial para el uso del servicio, existe una serie de consideraciones que apuntan a mantener la seguridad general del servicio y la sanidad de las comunicaciones que se den a través del mismo.

3. Flujo de trabajo sugerido

a. Login

El primer paso para empezar a usar el servicio es loguearse.

La mayoría de las rutas del servicio se encuentran protegidas por una autenticación a través de JSON Web Token's o JWT's. La manera de obtener un JWT válido es loguearse utilizando "appld" y "appKey". El JWT tiene un tiempo de expiración de 24hs luego del cual dejará de ser útil y será necesario renovarlo.

Una vez obtenido el JWT, ya puede accederse al resto de los endpoints utilizándolo como HTTP Header "x-access-token".

b. Cambiar AppKey y Re-Login

El segundo paso recomendado, para realizar por una única vez, es cambiar la AppKey que se entrega por defecto, por una de su preferencia, debe tener entre 8 y 24 caracteres. Luego de esto será necesario Re-Loguearse.


c. Asociar

Como paso inicial para controlar un sistema de alarma, es necesario asociar el "appld" del usuario con el mismo.

Este paso debe ser realizado una única vez para cada nuevo sistema que se desee controlar.

Para poder asociar se deberá contar con el "número de sistema" (Número que identifica a cada IP400 en el servicio) y la "clave de sistema" (Una clave configurable en IP400 que agrega un nivel más de seguridad).

Al enviar un request de asociar con estos datos correctos, y estando abierta la "ventana de tiempo para asociar terminal", se recibirá una respuesta satisfactoria que indicará que se asoció de forma correcta, sino, se enviará una respuesta negativa, indicando que algo en el proceso falló.



La "ventana de tiempo para asociar terminal" es de 15 minutos y se abre a partir de un comando disparado desde el sistema de alarma con la secuencia [PROG] + [8] + [3] + [Clave de usuario]

d. Obtener versiones

Antes de comenzar a interactuar con el sistema de alarma es recomendable conocer las versiones de Panel y Comunicador. Recordar que los comandos de configuración funcionan únicamente con versiones de comunicador a partir de "2.4".

e. Usar comandos de control y configuración

Una vez seguros de que el sistema está asociado y que se conocen las versiones ya es posible utilizar los comandos de control y configuración.

4. Documentación de la API

La documentación de la API explica los diferentes endpoints, los argumentos recibidos y las respuestas esperadas de cada uno.

Es importante saber que la API estará disponible en dos dominios

Production: **"https://www.serveralonsoalarmas.com"**

Development: **"https://app.serveralonsoalarmas.com"**

Y montada la siguiente ruta:

"/hsi/user_api/v1"

Todos los request/responses deben ser JSON, por eso, el header **'Content-Type': 'application/json'** deberá estar siempre presente.

El **response** siempre incluirá el atributo **"success"** siendo este un booleano que defina si la acción se concretó o no con éxito. **En caso de ser false, además incluirá un campo "message"** con información acerca de por qué no pudo concretarse exitosamente.

Para cada request realizado hacia la API, se obtiene además como respuesta un **status code** que indica el estado de la comunicación contra el servicio.

Status code **200**: Comunicación realizada de forma exitosa.

Status code **400**: La comunicación no se ha realizado exitosamente debido a algún parámetro inválido.

Status code **401**: Sesión expirada. Debe volver a loguearse en el servicio para poder comunicarse.

Status code **403**: **access token** no provisto.

Status code **500**: Se ha generado un error que será procesado y tratado internamente.

Status code **504**: Se ha sobrepasado el tiempo máximo de espera para realizar la acción solicitada.

POST - /login

Esta es la ruta de logueo que **permite a un usuario obtener su access token.**

Request body:

"appld": **String** - ID provisto

"appKey": **String** - Clave de usuario

Response body:

"accessToken": **String** - Es el JWT que deberá ser usado como "x-access-token" en los request siguientes

A partir de aquí, todas las rutas necesitan el header "x-access-token".

POST - /cambiar_app_key

Esta es la ruta que **permite a un usuario cambiar su AppKey.**

Request body:

"appKey": **String** - Nueva clave de usuario

POST - /asociar_cuenta

Esta es la ruta que **permite a un usuario asociar su cuenta** con un sistema de alarma.

Request body:

"numeroSistema": **String** - Número de sistema de IP400

"claveSistema": **String** - Clave de sistema de IP400

POST - /panel_commands/:action/:option?

Esta es la ruta que **permite enviar comandos de control** a un sistema de alarma.

A continuación se detallan las diferentes variaciones que admiten los parámetros action y option:

"/state" -> **consulta el estado** actual del sistema

"/arm/away" -> **arma** el sistema en modo **ausente**

"/arm/delayed" -> **arma** el sistema en modo **presente demorado**

"/disarm" -> **desarma** el sistema

"/bypass/:nroZona" -> **inhibe** una zona

"/unbypass/:nroZona" -> **restaura la inhibición** de una zona

"/turn_output_on/:nroSalida" -> **enciende** una **salida**

"/turn_output_off/:nroSalida" -> **apaga** una **salida**

"/turn_bell_on" -> **enciende** la salida de **sirena**

"/turn_bell_off" -> **apaga** la salida de **sirena**



Request body:

"numeroSistema": **String** - Número de sistema de IP400

"seq": **String** - Número de secuencia de "000" a "255" para identificar comandos

"partKey": **String** - Clave de 4 dígitos relacionada con la partición a controlar

Response body:

"result": **String** - Puede ser "OK" o "ERROR", por ejemplo, si se desea anular una zona mientras el sistema de alarma se encuentra armado, el campo "success" será true, porque desde el punto de vista de la API el comando se concretó, sin embargo, "result" será "ERROR" ya que el panel no admite que se inhiban zonas mientras se encuentra armado.

"panelStatus": **String** - Es un string que indica el estado del sistema en el momento, al final del documento se explica en un anexo, cómo interpretar el estado.

POST - /panel_commands/start_programation

Genera el evento de ingreso en programación en panel

Request body:

"numeroSistema": **String** - Número de sistema de IP400

"seq": **String** - Número de secuencia de "000" a "255" para identificar comandos

POST - /panel_commands/end_programation

Genera el evento de salida de programación en panel, **además genera la re-inicialización del sistema de alarma**, para que se procesen las nuevas configuraciones realizadas.



Request body:

"numeroSistema": **String** - Número de sistema de IP400

"seq": **String** - Número de secuencia de "000" a "255" para identificar comandos

PUT - /panel_programation/:entity

Graba una entidad **en la programación del panel** de alarmas

Todas las entidades comparten los siguientes parámetros

Request params:

"numeroSistema": **String** - Número de sistema de IP400

"seq": **String** - Número de secuencia de "000" a "255" para identificar comandos

Y además tienen un response body similar

Response body:

"result" - **String** - "OK" / "ERROR"

Las entidades pueden ser las siguientes:

clave_usuario

Permite grabar un código de usuario de 1 a 32, ingresable por el teclado del panel, siempre de 4 dígitos, el valor '0000' anula el código.

Query params:

"numeroUsuario" - **Number** - Número de usuario a grabar, de 1 a 32

Request body:

"clave" - **String** - Clave de usuario a grabar

clave_instalador

Permite grabar el código de instalador, ingresable por el teclado del panel, siempre de 4 dígitos.

Request body:

"clave" - **String** - Clave de instalador a grabar

clave_coaccion

Permite grabar la clave de coacción de cualquiera de las 4 particiones, ingresable por el teclado del panel, siempre de 4 dígitos.

Request params:

"numeroParticion" - **Number** - Número de partición de 1 a 4

Request body:

"clave" - **String** - Clave de coacción a grabar

clave_coaccion_control_acceso

Permite grabar la clave de coacción de control de acceso, ingresable por el teclado del control de acceso, siempre de 4 dígitos.

Request params:

"numeroParticion" - **Number** - Número de partición de 1 a 4

Request body:

"clave" - **String** - Clave de coacción a grabar

id_control_acceso

Permite grabar el id de un/a tarjeta/tag de control de acceso.

Request params:

"numeroDepartamento" - **Number** - Número de departamento de 1 a 255

"numeroTarjeta" - **Number** - Número de tarjeta de 1 a 16

Request body:

"idTarjeta" - **String** - ID de tarjeta a grabar

GET - /panel_programation/:entity

Lee una entidad en la programación del panel de alarmas

Todas las entidades comparten los siguientes parámetros

Request params:

"numeroSistema": **String** - Número de sistema de IP400

"seq": **String** - Número de secuencia de "000" a "255" para identificar comandos

Y además tienen un response body similar

Response body:

"result" - **String** contiene el valor leído

clave_usuario

Permite leer un código de usuario de 1 a 32, ingresable por el teclado del panel, siempre de 4 dígitos.

Query params:



"numeroUsuario" - **Number** - Número de usuario a grabar, de 1 a 32

clave_instalador

Permite leer el código de instalador, ingresable por el teclado del panel, siempre de 4 dígitos.

clave_coaccion

Permite leer la clave de coacción de cualquiera de las 4 particiones, ingresable por el teclado del panel, siempre de 4 dígitos.

Request params:

"numeroParticion" - **Number** - Número de partición de 1 a 4

clave_coaccion_control_acceso

Permite leer la clave de coacción de control de acceso, ingresable por el teclado del control de acceso, siempre de 4 dígitos.

Request params:

"numeroParticion" - **Number** - Número de partición de 1 a 4

id_control_acceso

Permite leer el id de un/a tarjeta/tag de control de acceso.

Request params:

"numeroDepartamento" - **Number** - Número de departamento de 1 a 255

"numeroTarjeta" - **Number** - Número de tarjeta de 1 a 16



version_ip_400

Permite leer la versión de un comunicador IP-400.

version_panel

Permite leer la versión de un panel de alarmas.

5. Recomendaciones de Implementación

Es importante que el cliente de HSI enmascare la API en su propia implementación para cumplir con las condiciones de uso, y que sus credenciales se mantengan seguras, como así también, puedan responder por todo el tráfico generado a través de este canal.

El servicio HSI no está pensado para ser distribuido al cliente final de forma directa.

Si se desea se puede agregar al servicio de HSI una regla de Firewall para admitir únicamente el uso del mismo desde una IP o pool de IP's específico.

La API cuenta con variados sistemas de protección ante DDoS y otros ataques informáticos.

Para realizar una programación, enviar siempre un comando de inicio antes de programar el sistema y uno de salida al finalizar. De esta manera el panel y todos los dispositivos conectados, iniciarán con la nueva programación y quedará almacenado cada ingreso y salida de programación en la memoria de eventos del sistema.}

Anexo

A continuación podrá observar a qué corresponde cada uno de los dígitos que se reciben ante una petición de estado.

La información que obtendrá necesitará transformarla a valores binarios para poder ser interpretada más adelante. Dicha transformación se hace utilizando el High Nibble y el Low Nibble (salvo en el primer dígito).

Una vez transformados los datos en un byte, debe comenzar a interpretarlos (con codificación **little-endian**) de derecha a izquierda siguiendo las indicaciones dadas en “Explicación de cada Campo (Byte)”.

<d000> En un dígito ASCII, la partición a la cual corresponde el código de 4 dígitos enviado como argumento del comando

<d01a> Registro de Problemas 1 (High Nibble del valor Hexa).

<d01b> Registro de Problemas 1 (Low Nibble del valor Hexa).

<d02a> Registro de Problemas 2 (High Nibble del valor Hexa).

<d02b> Registro de Problemas 2 (Low Nibble del valor Hexa).

<d03a> Estado de las Particiones (High Nibble del valor Hexa).

<d03b> Estado de las Particiones (Low Nibble del valor Hexa).

<d04a> Estado de las salidas (High Nibble del valor Hexa).

<d04b> Estado de las salidas (Low Nibble del valor Hexa).

<d05a> Zonas (abiertas/cerradas) 1-8 (High Nibble del valor Hexa).

<d05b> Zonas (abiertas/cerradas) 1-8 (Low Nibble del valor Hexa).

<d06a> Zonas (abiertas/cerradas) 9-16 (High Nibble del valor Hexa).

<d06b> Zonas (abiertas/cerradas) 9-16 (Low Nibble del valor Hexa).

<d07a> Zonas (abiertas/cerradas) 17-24 (High Nibble del valor Hexa).

<d07b> Zonas (abiertas/cerradas) 17-24 (Low Nibble del valor Hexa).

<d08a> Zonas (abiertas/cerradas) 25-32 (High Nibble del valor Hexa).

<d08b> Zonas (abiertas/cerradas) 25-32 (Low Nibble del valor Hexa).

<d09a> Alarma Zonas 1-8 (High Nibble del valor Hexa).

<d09b> Alarma Zonas 1-8 (Low Nibble del valor Hexa).

<d10a> Alarma Zonas 9-16 (High Nibble del valor Hexa).

<d10b> Alarma Zonas 9-16 (Low Nibble del valor Hexa).

<d11a> Alarma Zonas 17-24 (High Nibble del valor Hexa).

<d11b> Alarma Zonas 17-24 (Low Nibble del valor Hexa).
<d12a> Alarma Zonas 25-32 (High Nibble del valor Hexa).
<d12b> Alarma Zonas 25-32 (Low Nibble del valor Hexa).
<d13a> Zonas inhibidas 1-8 (High Nibble del valor Hexa).
<d13b> Zonas inhibidas 1-8 (Low Nibble del valor Hexa).
<d14a> Zonas inhibidas 9-16 (High Nibble del valor Hexa).
<d14b> Zonas inhibidas 9-16 (Low Nibble del valor Hexa).
<d15a> Zonas inhibidas 17-24 (High Nibble del valor Hexa).
<d15b> Zonas inhibidas 17-24 (Low Nibble del valor Hexa).
<d16a> Zonas inhibidas 25-32 (High Nibble del valor Hexa).
<d16b> Zonas inhibidas 25-32 (Low Nibble del valor Hexa).

Se incluye 1 caracteres más, correspondiente a las salidas PGMW's.

<d17a> Estado de las salidas inalámbricas 1-4
<d17b> Estado de las salidas inalámbricas 1-4

A continuación se incluyen 4 caracteres más, correspondientes al mismo IP400. Estos componen dos bytes.

Los primeros 4 bits dan indicación cuáles de las 4 salidas/entradas del IP-400 se encuentran configuradas como entradas. En los 4 siguientes se puede ver cuántas de esas entradas se encuentran en estado de alarma.

El byte siguiente indica con los primeros 4 bits que salidas/entradas del IP-400 se encuentran programadas como salidas y finalmente con los otros 4 bits cuáles de ellas se encuentran activadas.

En estos 4 caracteres, luego de componer los bytes deben leerse de izquierda a derecha para contar desde la salida/entrada 1 a la 4 (al revés de los bytes anteriores).

<d18a> Entradas/salidas configuradas como entradas (b7:b4) en "1".
<d18b> Entradas en estado de alarma (b3:b0) en "1", normal en "0".
<d19a> Entradas/Salidas configuradas como salidas (b7:b4) en "1".
<d19b> Salidas activas (b3:b0) en "1".

| Explicación de cada Campo.

Registro Problemas 1:

bit0:

0 = no hay fallo de Red

1 = hay fallo de Red

bit1:

0 = El panel no tiene baja batería

1 = El panel tiene baja batería

bit2:

0 = la fuente auxiliar no tiene baja batería

1 = la fuente auxiliar tiene baja batería

bit3:

0 = no hay fallo en supervisión de sirena

1 = hay fallo en sup. de sirena

bit4:

0 = no hay un fallo en la alimentación auxiliar del panel

1 = hay un fallo en la alimentación auxiliar del panel

bit5:

0 = no hay fallo alimentación Auxiliar bus datos

1 = hay un fallo en la alimentación auxiliar del panel

bit6:

0 = no hay un fallo de línea telefónica

1 = hay un fallo de línea telefónica

bit7:

0 = no hay un fallo en la salida de sirena

1 = hay fallo en la salida de sirena

Registro Problemas 2:

bit0:

0 = no hay fallo comunicación en teléfono 1

1 = hay fallo comunicación en teléfono 1

bit1:

0 = no hay fallo comunicación en teléfono 2

1 = hay fallo comunicación en teléfono 2

bit2:

0 = no hay fallo comunicación por gprs


1 = hay fallo comunicación por gprs

bit3:

0 = no hay fallo comunicación SMS

1 = fallo comunicación SMS

bit4:



0 = no hay fallo comunicación por Internet/ethernet
1 = hay fallo comunicación por Internet/ethernet

bit5:

0 = no hay fallo de link en gprs
1 = hay fallo de link en gprs

bit6:

0 = no hay fallo de link en internet/ethernet
1 = hay fallo de link en internet/ethernet

bit7:

0 = el reloj se encuentra programado
1 = el reloj no se encuentra programado

Estado de las Particiones:

bit0:

0 = la part4 se encuentra desarmada
1 = part4 armada

bit1:

0 = part3 desarmada
1 = part3 armada

bit2:

0 = part2 desarmada
1 = part2 armada

bit3:

0 = part1 desarmada
1 = part1 armada

bit4:

0 = part4 no lista para armar
1 = part4 lista para armar

bit5:

0 = part3 no lista para armar
1 = part3 lista para armar

bit6:

0 = part2 no lista para armar
1 = part2 lista para armar

bit7:

0 = part1 no lista para armar
1 = part1 lista para armar

Estado de las salidas:

bit0:
0 = sirena desactivada
1 = sirena activada

bit1:
0 = pgm1 desactivada
1 = pgm1 activada

bit2:
0 = pgm2 desactivada
1 = pgm2 activada

bit3:
0 = pgm3 desactivada
1 = pgm3 activada

bit4:
0 = pgm4 desactivada
1 = pgm4 activada

bit5:
0 =
1 =

bit6:
0 =
1 =

bit7:
0 =
1 =


Zonas Abiertas/cerradas 1-8

bit0:
0 = z1 cerrada
1 = z1 abierta

bit1:
0 = z2 cerrada
1 = z2 abierta

bit2:
0 = z3 cerrada
1 = z3 abierta

bit3:
0 = z4 cerrada
1 = z4 abierta



bit4:
0 = z5 cerrada
1 = z5 abierta

bit5:
0 = z6 cerrada
1 = z6 abierta

bit6:
0 = z7 cerrada
1 = z7 abierta

bit7:
0 = z8 cerrada
1 = z8 abierta

Zonas Abiertas/cerradas 9-16

bit0:
0 = z9 cerrada
1 = z9 abierta

bit1:
0 = z10 cerrada
1 = z10 abierta

bit2:
0 = z11 cerrada
1 = z11 abierta

bit3:
0 = z12 cerrada
1 = z12 abierta

bit4:
0 = z13 cerrada
1 = z13 abierta

bit5:
0 = z14 cerrada
1 = z14 abierta

bit6:
0 = z15 cerrada
1 = z15 abierta

bit7:
0 = z16 cerrada
1 = z16 abierta

Zonas Abiertas/cerradas 17-24

bit0:

0 = z17 cerrada

1 = z17 abierta

bit1:

0 = z18 cerrada

1 = z18 abierta

bit2:

0 = z19 cerrada

1 = z19 abierta

bit3:

0 = z20 cerrada

1 = z20 abierta

bit4:

0 = z21 cerrada

1 = z21 abierta

bit5:

0 = z22 cerrada

1 = z22 abierta

bit6:

0 = z23 cerrada

1 = z23 abierta

bit7:

0 = z24 cerrada

1 = z24 abierta

Zonas Abiertas/cerradas 25-32

bit0:

0 = z25 cerrada

1 = z25 abierta

bit1:

0 = z26 cerrada

1 = z26 abierta


bit2:

0 = z27 cerrada

1 = z27 abierta

bit3:

0 = z28 cerrada



1 = z28 abierta
bit4:
0 = z29 cerrada
1 = z29 abierta
bit5:
0 = z30 cerrada
1 = z30 abierta
bit6:
0 = z31 cerrada
1 = z31 abierta
bit7:
0 = z32 cerrada
1 = z32 abierta

Alarma Zonas 1-8

bit0:
0 = z1 normal
1 = z1 alarma
bit1:
0 = z2 normal
1 = z2 alarma
bit2:
0 = z3 normal
1 = z3 alarma
bit3:
0 = z4 normal
1 = z4 alarma
bit4:
0 = z5 normal
1 = z5 alarma
bit5:
0 = z6 normal
1 = z6 alarma
bit6:
0 = z7 normal
1 = z7 alarma
bit7:
0 = z8 normal



1 = z8 alarma

Alarma Zonas 9-16

bit0:

0 = z9 normal

1 = z9 alarma

bit1:

0 = z10 normal

1 = z10 alarma

bit2:

0 = z11 normal

1 = z11 alarma

bit3:

0 = z12 normal

1 = z12 alarma

bit4:

0 = z13 normal

1 = z13 alarma

bit5:

0 = z14 normal

1 = z14 alarma

bit6:

0 = z15 normal

1 = z15 alarma

bit7:

0 = z16 normal

1 = z16 alarma

Alarma Zonas 17-24

bit0:

0 = z17 normal

1 = z17 alarma


bit1:

0 = z18 normal

1 = z18 alarma

bit2:

0 = z19 normal



1 = z19 alarma
bit3:
0 = z20 normal
1 = z20 alarma
bit4:
0 = z21 normal
1 = z21 alarma
bit5:
0 = z22 normal
1 = z22 alarma
bit6:
0 = z23 normal
1 = z23 alarma
bit7:
0 = z24 normal
1 = z24 alarma

Alarma Zonas 25-32

bit0:
0 = z25 normal
1 = z25 alarma
bit1:
0 = z26 normal
1 = z26 alarma
bit2:
0 = z27 normal
1 = z27 alarma
bit3:
0 = z28 normal
1 = z28 alarma
bit4:
0 = z29 normal
1 = z29 alarma
bit5:
0 = z30 normal
1 = z30 alarma
bit6:
0 = z31 normal
1 = z31 alarma
bit7:
0 = z32 normal



1 = z32 alarma

Zonas Inhibidas 1-8

bit0:

0 = z1 normal
1 = z1 inhibida

bit1:

0 = z2 normal
1 = z2 inhibida

bit2:

0 = z3 normal
1 = z3 inhibida

bit3:

0 = z4 normal
1 = z4 inhibida

bit4:

0 = z5 normal
1 = z5 inhibida

bit5:

0 = z6 normal
1 = z6 inhibida

bit6:

0 = z7 normal
1 = z7 inhibida

bit7:

0 = z8 normal
1 = z8 inhibida

Zonas Inhibidas 9-16

bit0:


0 = z9 normal
1 = z9 inhibida

bit1:

0 = z10 normal
1 = z10 inhibida

bit2:

0 = z11 normal
1 = z11 inhibida



bit3:
0 = z12 normal
1 = z12 inhibida

bit4:
0 = z13 normal
1 = z13 inhibida

bit5:
0 = z14 normal
1 = z14 inhibida

bit6:
0 = z15 normal
1 = z15 inhibida

bit7:
0 = z16 normal
1 = z16 inhibida

Zonas Inhibidas 17-24

bit0:
0 = z17 normal
1 = z17 inhibida

bit1:
0 = z18 normal
1 = z18 inhibida

bit2:
0 = z19 normal
1 = z19 inhibida

bit3:
0 = z20 normal
1 = z20 inhibida

bit4:
0 = z21 normal
1 = z21 inhibida

bit5:
0 = z22 normal
1 = z22 inhibida

bit6:
0 = z23 normal
1 = z23 inhibida

bit7:

- 0 = z24 normal
- 1 = z24 inhibida

Zonas Inhibidas 25-32

bit0:

- 0 = z25 normal
- 1 = z25 inhibida

bit1:

- 0 = z26 normal
- 1 = z26 inhibida

bit2:

- 0 = z27 normal
- 1 = z27 inhibida

bit3:

- 0 = z28 normal
- 1 = z28 inhibida

bit4:

- 0 = z29 normal
- 1 = z29 inhibida

bit5:

- 0 = z30 normal
- 1 = z30 inhibida

bit6:

- 0 = z31 normal
- 1 = z31 inhibida

bit7:

- 0 = z32 normal
- 1 = z32 inhibida

Estado de las salidas inalámbricas:

bit0:

- 0 = PGMW4 Apagada
- 1 = PGMW4 Encendida

bit1:

- 0 = PGMW3 Apagada
- 1 = PGMW3 Encendida

bit2:



0 = PGMW2 Apagada
1 = PGMW2 Encendida
bit3:
0 = PGMW1 Apagada
1 = PGMW1 Encendida
bit4:
0 =
1 =
bit5:
0 =
1 =
bit6:
0 =
1 =
bit7:
0 =
1 =