

ZES-22

Software

Librería de Control y Monitorización

Versión de Integración

v1.0.0.0



Índice de contenido

1. Requerimientos
2. Ficheros
3. Espacio de Nombres
4. Constructor
5. Destructor
6. Inicialización / Conexión
7. Parámetros Generales y de Red del Equipo
8. Lectura de Estados del Equipo
9. Lectura de Vúmetros de Entradas y Salidas
10. Lectura de Ganancias de Entradas y Salidas
11. Lectura de Entradas asignadas a Salidas
12. Lectura de Mute de Salidas
13. Cambio de Ganancia de Entradas y Salidas
14. Asignación de Entradas a Salidas
15. Muteo de Salidas

1. Requerimientos

- NET Framework 4.0

2. Ficheros

Ficheros que habrá que incluir como referencia en proyectos de integración.

ZES22CtrlLib.dll	Librería para control y monitorización de equipos LDA ZES-22
Socket.dll	Librería auxiliar.

3. Espacio de Nombres

```
LDA.Device.ZES22
```

4. Constructor

```
Dim zes As ZES22  
zes = New ZES22()
```

5. Destructor

```
zes.Dispose() ó zes.pClose()
```

- Ambos métodos son equivalentes, emplear sólo uno de ellos.
- Llamar siempre que deje de utilizarse cualquier instancia de la librería para liberar recursos y finalizar subprocesos.

6. Inicialización / Conexión

```
Public Function foConnect(  
    ByVal IP As String,  
    ByVal puerto As Integer  
) As Collection
```

- Especificar la IP y el puerto (habitualmente el 60000) del equipo remoto, por ejemplo:

```
res = zes.foConnect("192.168.3.110", 60000)
```

- Posibles valores de retorno:

- res("state") = 1
Ok.
- res("state") = 2
Dirección IP no válida.
- res("state") = 0
Error.

7. Parámetros Generales y de Red del Equipo

<u>PARÁMETRO</u>	<u>TIPO</u>	<u>DESCRIPCIÓN</u>
zes.FW_version	String	Versión de firmware del equipo
zes.Net_MAC	String	MAC del interfaz Ethernet
zes.Net_IP	String	IP del equipo
zes.Net_GW	String	Puerta de enlace del equipo
zes.Net_Mask	String	Máscara de red del equipo
zes.Num_Inputs	Byte	Número de entradas físicas configuradas en el equipo
zes.Num_Outputs	Byte	Número de salidas físicas configuradas en el equipo

8. Lectura de Estados del Equipo

<u>PARÁMETRO</u>	<u>TIPO</u>	<u>DESCRIPCIÓN</u>
zes.Sinc_Connect	Boolean	True: Hay conexión establecida con el equipo. False: No hay conexión establecida con el equipo.
zes.ACSI_Mode	Boolean	True: Modo ACSI Activo. Entrada 3 no disponible como entrada de audio, empleada internamente para distribución de audio de micrófonos ACSI. Sólo si en el equipo se han configurado 3 o 4 entradas físicas. False: Modo ACSI No Activo.

9. Lectura de Vúmetros de Entradas y Salidas

La información sobre el valor de los vúmetros de las entradas y salidas físicas lo podemos obtener con:

```
zes.Vum_Peak  
zes.Vum_RMS
```

Estos campos son arrays de longitud 4 y de tipo SByte donde cada posición representa el vúmetro de uno de los 4 canales físicos del equipo.

- Posición 0 : **CH1**
- Posición 1 : **CH2**
- Posición 2 : **CH3**
- Posición 3 : **CH4**

Si hay entradas configuradas en el equipo (`zes.num_inputs > 0`) éstas irán ocupando los índices menores. Es decir, si por ejemplo `zes.num_inputs = 2`, CH1 y CH2 serán entradas físicas y CH3 y CH4 serán salidas físicas.

El valor del vúmetro de una entrada es previo a cualquier cambio de ganancia o modificación en la señal de audio por parte del equipo.

En el valor del vúmetro de una salida sí se verá reflejado cualquier cambio de ganancia o modificación en la señal de audio por parte del equipo.

10. Lectura de Ganancias de Entradas y Salidas

La información sobre el valor de ganancia de las entradas y salidas físicas lo podemos obtener con:

```
zes.Inputs_Gain  
zes.Outputs_Gain
```

Estos campos son arrays de longitud entre 0 y 4 y de tipo SByte donde cada posición representa el valor de ganancia de una entrada física para `zes.Inputs_Gain` o de una salida física para `zes.Outputs_Gain`. La longitud de cada array viene determinada por la configuración de Entradas y Salidas del equipo.

Si no hay entradas físicas configuradas, `zes.Inputs_Gain` estará vacío (longitud = 0).

Si no hay salidas físicas configuradas, `zes.Outputs_Gain` estará vacío (longitud = 0).

11. Lectura de Entradas asignadas a Salidas

La información sobre qué entradas están asignadas a las salidas físicas la tenemos en:

`zes.Outputs_Input`

Este campo es un array con tantas posiciones como salidas físicas tenga configuradas el equipo. Cada posición representa una fuente que puede tomar alguno de estos valores:

- **0** : Ninguna entrada rutada.
- **1 a 4** : Se refieren a alguna de las 4 posibles **entradas físicas** según se haya configurado el equipo.
- **9 o mayor** : Se refiere a una entrada Cobranet.

Si no hay salidas físicas configuradas, este array estará vacío (longitud = 0).

12. Lectura de Mute de Salidas

La información sobre qué salidas físicas están muteadas la tenemos en:

`zes.Outputs_Mute`

Este campo es un array con tantas posiciones como salidas físicas tenga configuradas el equipo. Cada posición es un valor booleano que indica si el mute está activo (true) o no (false).

Si no hay salidas físicas configuradas, este array estará vacío (longitud = 0).

13. Cambio de Ganancia de Entradas y Salidas

```
Public Function foSetGain(  
    ByVal id As Byte,  
    ByVal e_s As Boolean,  
    ByVal gain As Byte  
) As Collection
```

- Establece la ganancia de una entrada o salida física con el parámetro gain. Éste puede tomar valores comprendidos entre 0 y -100.
- El parámetro e_s indica si vamos a cambiar la ganancia de una entrada (true) o de una salida (false).
- El parámetro id podrá tomar valores en función del parámetro e_s y el número de entradas y salidas físicas configuradas en el equipo.
- Por ejemplo, si queremos establecer la ganancia a -25 para la entrada física 2:

```
res = zes.foSetGain(2, true, -25)
```

- Posibles valores de retorno:

- res("state") = 1
Ok.
- res("state") = 2
Error en parámetros.

14. Asignación de Entradas a Salidas

```
Public Function foSetOutput(  
    ByVal output As Byte,  
    ByVal input As Byte  
) As Collection
```

- Establece la entrada que se va a asignar a una salida.
- El parámetro `output` indica a qué salida vamos a asignar la entrada.
- El parámetro `input` indica qué entrada vamos a asignar a la salida. Si el valor de este parámetro es `0`, la salida quedará sin entrada asignada.
- Los valores de `input` y `output` han de ser coherentes con la configuración de entradas y salidas del equipo.
- Por ejemplo, si queremos asignar la entrada 1 a la salida física 3:

```
res = zes.foSetOutput(3, 1)
```

- Posibles valores de retorno:
 - `res("state") = 1`
Ok.
 - `res("state") = 2`
Error en parámetros.

15. Muteo de Salidas

```
Public Function foSetMute(  
    ByVal id As Byte,  
    ByVal Mute As Boolean  
) As Collection
```

- Establece el muteo de una salida.
- El parámetro `id` indica a qué salida vamos a asignar la entrada.
- El parámetro `Mute` indica si se va a mutear la salida (`true`) o no (`false`).
- El valor de `id` ha de ser coherente con la configuración de entradas y salidas del equipo.

- Por ejemplo, si queremos mutear la salida física 2:

```
res = zes.foSetMute(2, True)
```

- Posibles valores de retorno:

- `res("state") = 1`
Ok.
- `res("state") = 2`
Error en parámetros.

