



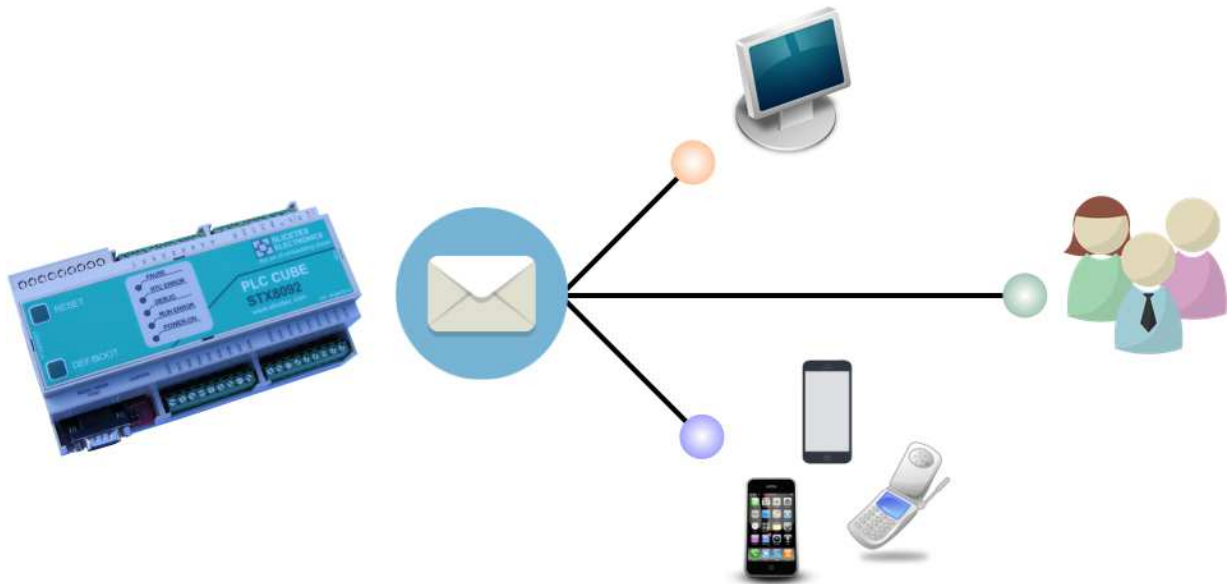
Slicetex Ladder Designer Studio

NOTA DE APLICACIÓN

AN020

Enviar E-MAILS desde el PLC

Autor: Ing. Boris Estudiez



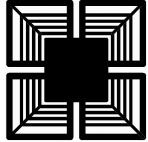
Modelos Aplicables	AX, CX y DX
--------------------	-------------

1 Descripción General

La presente nota de aplicación explica como configurar y enviar E-MAILS o correo electrónicos desde el PLC.

La capacidad de enviar emails desde nuestros PLC le permitirá enviar mensajes de textos o datos a diferentes destinatarios para informales posibles eventos, alarmas, valores de parámetros o simplemente información.

El destinatario podrá leer el mensaje desde su computadora, teléfono móvil (Smartphone), tablet, etc.



2 Lecturas Recomendadas

Antes de leer este documento, recomendamos que se familiarice con el software StxLadder y el PLC adquirido. Sugerimos leer los siguientes documentos:

1. Manual de Usuario del software StxLadder.
2. Manual de Programación Pawn del PLC (si utiliza el ejemplo en lenguaje Pawn)
3. Hoja de datos técnicos del PLC.

Más documentación puede encontrar en la página del producto: www.slicetex.com.

Para consultas y soporte, ponemos a disposición un foro de discusión en: www.slicetex.com/foro donde puede leer preguntas de otros usuarios y realizar también sus propias preguntas.

3 Requerimientos

Para enviar emails desde el PLC debe tener instalado el entorno de Programación **StxLadder** (Slicetex Ladder) y utilizar un firmware actualizado en el PLC.

Es necesario que el PLC esté conectado a internet para poder enviar los e-mails.

Se recomienda utilizar las últimas versiones del entorno **StxLadder** y versión actualizada del firmware en su PLC.

Opcional: Es posible utilizar un servidor SMTP (Simple Mail Transfer Protocol) externo en red local o internet, para ello debe contar con una cuenta en el servidor SMTP que admita autenticación AUTH-LOGIN.



4 Teoría de Funcionamiento

Los PLC de Slicetex Electronics permiten el envío de e-mails en sus últimas versiones de firmware. Para enviar un email hay dos opciones:

1. Enviar el mail utilizando un servidor **SMTP** (Simple Mail Transfer Protocol) definido por el PLC: Simple, sin configuración. Solo requiere conexión a internet.
2. Enviar el mail utilizando un servidor SMTP externo: Para ello es necesario contar con una cuenta de e-mail en un servidor **SMTP** y configurarla en el PLC. Como ventaja, se puede usar un servidor propio en la red Local sin conexión a internet.

El PLC internamente cuenta con un cliente de e-mail que se conecta al servidor SMTP para enviar el mensaje a los destinatarios finales (direcciones de correo). El proceso se muestra en la siguiente figura:

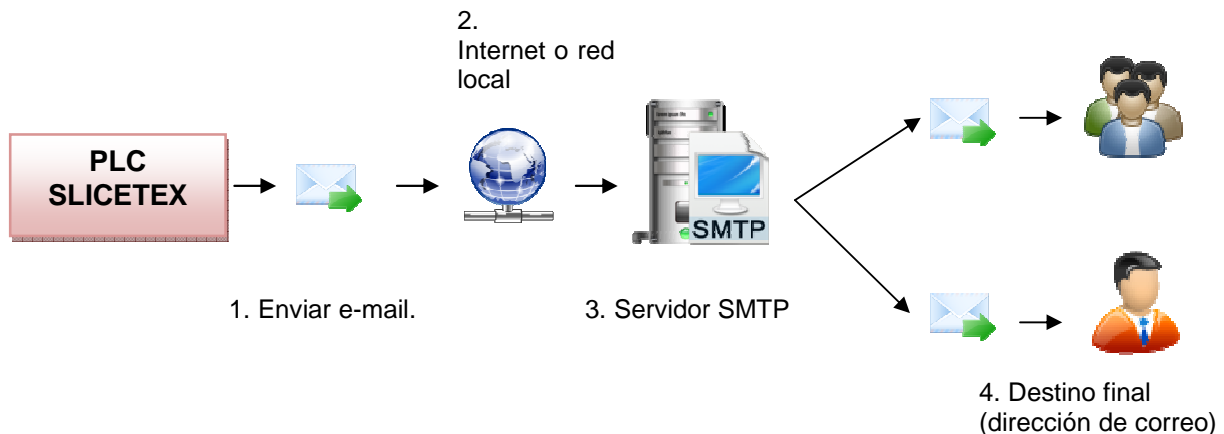
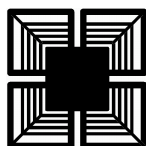


Fig. 1: Proceso al enviar un E-mail desde el PLC.

La figura 1, muestra el proceso que lleva a cabo el PLC cuando un e-mail debe ser enviado a uno o más destinatarios. Los pasos son los siguientes:

1. **Enviar e-mail:** El PLC ejecuta un componente Ladder o instrucción Pawn que envía un e-mail a un correo electrónico.
2. **Internet o red local:** El PLC utiliza una conexión en la red para buscar el servidor SMTP.
3. **Servidor SMTP:** El PLC se conecta al servidor SMTP, ingresa al mismo con los datos de la cuenta de email, es decir el nombre de usuario y contraseña. Una vez validado dentro del servidor, ingresa las direcciones de correo electrónico de destino y el mensaje.
4. **Destino final:** Si el servidor SMTP acepta la conexión del PLC, procede a despachar los e-mails a las direcciones de correo electrónico especificadas. Luego, los dueños de las direcciones de correo electrónicos pueden bajar el mensaje enviado desde el PLC.



El proceso de envío de email es una operación que puede tardar decenas de segundos o minutos, ya que las conexiones por internet están sujetas a retardos y congestión.

4.1 Requerimientos del Servidor SMTP Externo

La forma más simple de enviar un mail, es utilizando un servidor SMTP definido por el PLC, pero si usa un servidor SMTP externo, el mismo debe cumplir con las siguientes características.

El servidor SMTP debe permitir la autenticación AUTH LOGIN, sin encriptación. **Algunos servidores SMTP como hotmail.com o gmail.com, no están soportados**, ya que requieren una conexión encriptada SSL/TLS.

A continuación pasamos algunos servidores SMTP gratuitos que pueden ser utilizados con el PLC para enviar datos, ya que cumplen los requisitos:

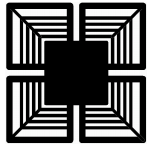
Tabla: Algunos servidores SMTP que admiten conexión no encriptada

Servidor	Puerto	Descripción	Gratuito
smtp.aol.com	587	Servidor de AOL (America On-Line). Puede registrarse gratuitamente en: webmail.aol.com	SI
smtp.arnet.com.ar	25	Servidor de servicio internet de Telecom Argentina, común en servicios de internet por banda ancha.	NO
nxnethosting.com	587	Servidor de NxNet / Latin Cloud Argentina. Puede contratar el servicio por unos cuantos dólares al año y registrar un dominio a medida.	NO
hostgator.com	25 / 587	Servidor hosting en EEUU, planes económicos de calidad.	NO

La lista es más extensa, puede buscar en www.google.com servidores que admitan este tipo de conexión. Si ya tiene un servidor SMTP propio, consulte con el soporte técnico para ver compatibilidad o pedirla.

Con la cuenta habilitada en el servidor SMTP, **puede enviar e-mails a cualquier dirección de correo** (incluida hotmail.com, gmail.com, etc).

Tip: Si necesita tener una copia del email enviado desde el PLC, asegúrese de agregar una dirección de correo en el campo **CC** del mensaje.



5 Enviar E-Mail con Lenguaje Ladder

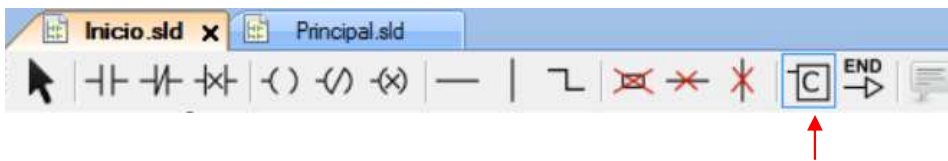
En esta sección explicaremos a modo general como enviar un e-mail utilizando el lenguaje Ladder.

El proyecto completo puede descargarse desde nuestra página Web, con el nombre “**MailExample1Simple.zip**”, el mismo contiene el código necesario para probar el cliente de correo electrónico e imprimir el progreso del envío del e-mail en la pantalla de **VirtualHMI**.

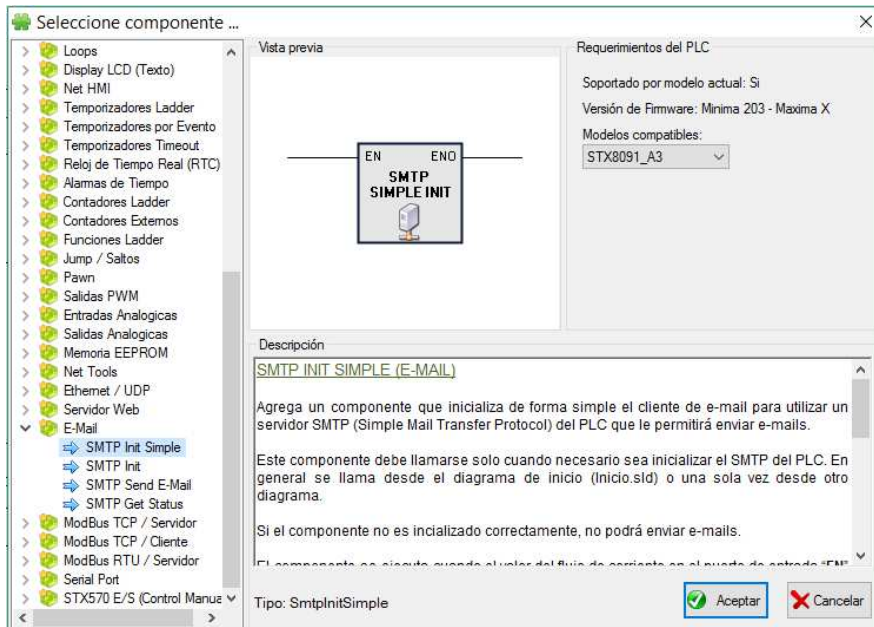
Hay tres componentes claves para el envío del E-mail en Ladder:

- **SmtplnitSimple**: Inicializa los parámetros del PLC para utilizar un servidor SMTP propio.
- **SmtplibSend**: Envía el e-mail a los destinatarios especificados.
- **SmtplibGetStatus**: Obtiene el estado del envío del e-mail (útil para comprobar si existe un error o si el mail fue entregado con éxito).

Para acceder a los componentes, debemos ir a un diagrama en **StxLadder** e insertar componente con el icono:



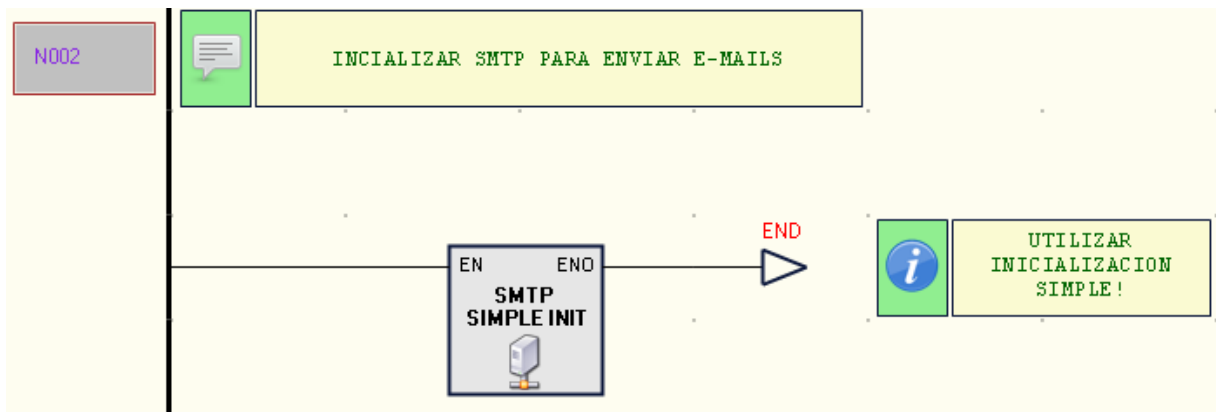
Los componentes de e-mail se ubican en el grupo **E-Mail**, como muestra la sig. Imagen:





Entonces, primero debemos inicializar los parámetros del servidor **SMTP** con el componente **SmpInItSimple**. Es conveniente llamar a este componente desde el diagrama **Inicio.sld**, ya que la configuración en general se realiza una vez al comienzo del programa del PLC.

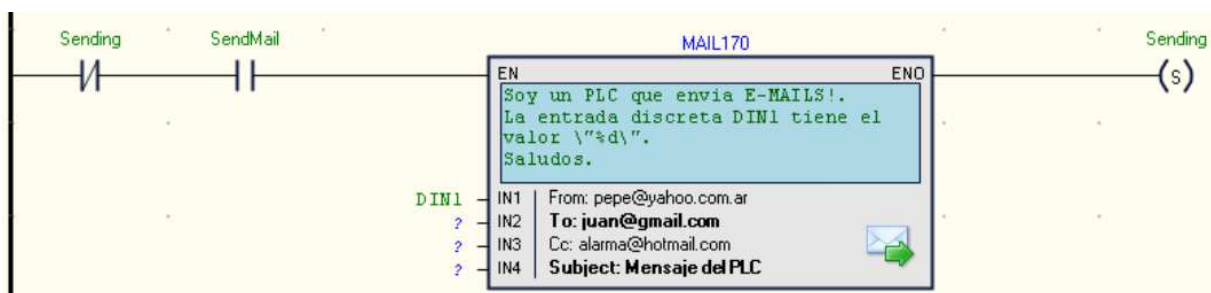
Insertamos el componente como se muestra a continuación:



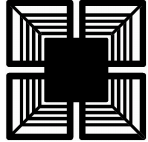
El componente SmpInItSimple no se necesita configuración por defecto ya que le dice al PLC que utilice un servidor SMTP propio del PLC. Si necesita utilizar un servidor SMTP externo, diríjase a la página 11.

Para enviar el e-mail, debemos utilizar el componente **SmpMailSend**. El componente cuando se ejecuta envía un e-mail mediante el servidor SMTP. Podemos insertar dicho componente para que se ejecute cuando una entrada discreta pasa de 0 a 1 en el diagrama **Principal.sld** o desde cualquier otra condición.

A continuación mostramos el componente insertado en el proyecto **"MailExample1Simple.zip"** disponible para bajar desde nuestra página Web:



Como se aprecia en la figura, el componente envía un e-mail cuando la variable **"Sending"** es "0" y la variable **"SendMail"** es "1". Si el mail es aceptado para envío, se establece la variable **"Sending"** a "1".



El mensaje a enviar es el siguiente:

```
Soy un PLC que envia E-MAILS!.  
La entrada discreta DIN1 tiene el valor \"%d\".  
Saludos.
```

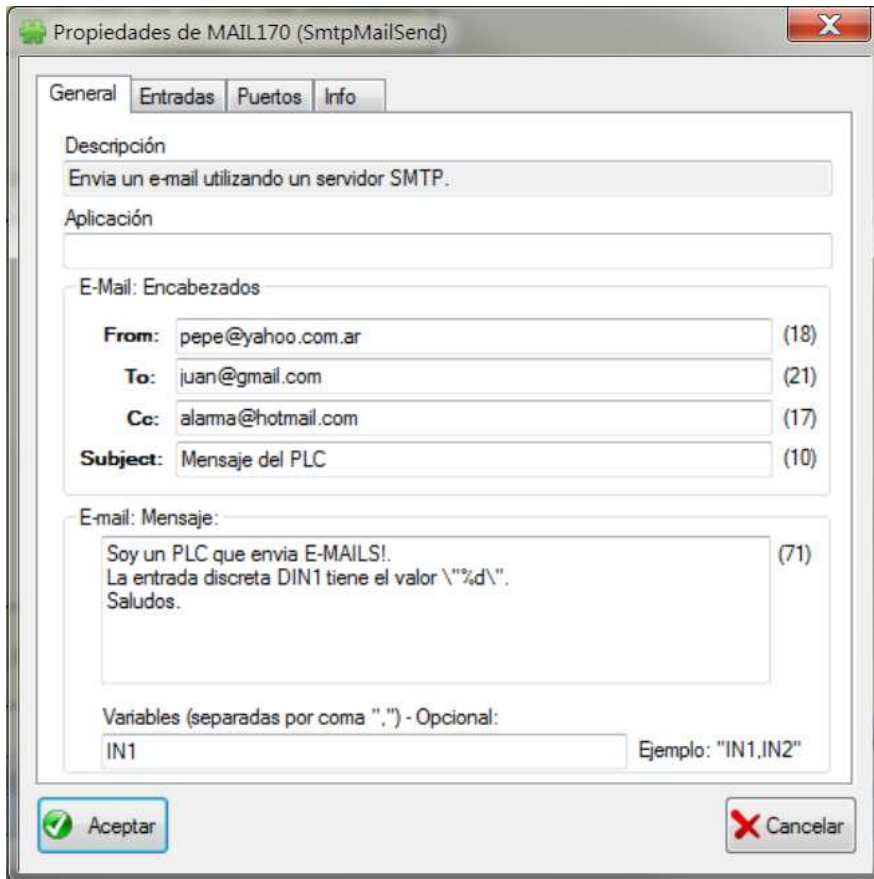
Además, el mensaje puede enviarse con formato, es decir el símbolo **%d**, será reemplazado por el valor de la entrada discreta DIN1, que se muestra conectada al puerto **IN1** del componente. También pueden pasarse variables Int32, float, etc. El mecanismo es similar a imprimir valores en el display LCD de **VirtualHMI** o del PLC.

El mensaje se enviará a la dirección **juan@gmail.com** (con copia a **alarma@hotmail.com**). El remitente será **pepe@yahoo.com.ar**.

El asunto del mensaje es: "**Mensaje del PLC**".

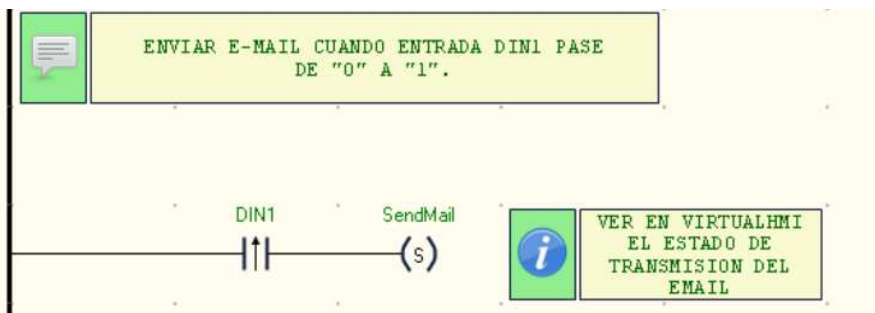


Si accedemos a las propiedades del componente **SmtMailSend** podemos especificar los parámetros del e-mail como se muestra a continuación:



Los números entre paréntesis, por ejemplo (71), son los caracteres remanentes, ya que la memoria para e-mail del PLC es limitada.

En el ejemplo para descargar, desde **Principal.sld** se envía un mail cuando la entrada **DIN1** pasa de 0 a 1, ya que la variable global **SendMail** se pone a 1 y luego es leída por el evento **OnRtcSecond**:

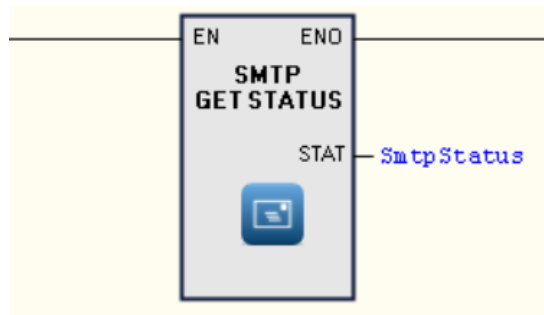




5.1 Ver Estado de Envío del E-Mail

Luego de enviar el e-mail con el componente **SmtplibSend**, debemos comprobar que no se produzcan errores, ya sea por pérdida de conexión, servidores ocupados, direcciones erróneas, puertos mal especificados o datos de acceso inválidos.

Para ver el estado de la transmisión y librería SMTP del PLC, debemos utilizar el componente **SmtplibGetStatus**:



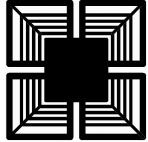
El componente retorna en una variable del tipo **Int32** un código de estado de la transmisión del e-mail, que podemos utilizar para verificar si hay errores, si el mail fue entregado o para imprimir en alguna pantalla.

En este caso el código de estado se devuelve en la variable apuntada por el puerto **STAT** del componente, es decir en **SmtplibStatus**:

Si el retorno es cero, el email fue enviado exitosamente. Si es negativo se produjo un error en el envío. Si es positiva, puede indicar que el mensaje está en proceso de envío o un caso particular.

Valores de retorno:

- (127): El email está en proceso de envío.
- (126): La librería no fue inicializada con el componente **SmtplibInit**.
- (0): **No hay errores**, el e-mail fue enviado con éxito.
- (-1): Error, la conexión al servidor fue cerrada o expiró.
- (-2): Error, no se recibió el código 220 del servidor.
- (-3): Error en proceso "EHLO".
- (-4): Error en proceso "AUTH".
- (-5): Error en proceso "USERNAME". Verifique nombre de usuario.
- (-6): Error en proceso "PASSWORD". Verifique contraseña.
- (-7): Error al enviar campo "FROM".
- (-8): Error al enviar campo "TO".
- (-9): Error al enviar campo "CC".
- (-10): Error al enviar campo "DATA".
- (-11): Error al enviar "." para finalizar mensaje.
- (-12): Error, conexión cerrada por el servidor.
- (-13): Error, el nombre del servidor no puede ser resuelto.
- (-14): Error, la conexión no pudo ser creada.
- (-15): Error. Ver código de error retornado por **SmtplibSendMail**.



En el proyecto "**MailExample1Simple.zip**" disponible para bajar desde nuestra página Web, verificamos cada un segundo desde el diagrama evento "**OnRtcSecond.sld**" el estado de la transmisión del e-mail y al mismo tiempo la imprimimos en la pantalla de **VirtualHMI**.

También comprobamos los casos particulares, por ejemplo si el código de estado es "**0**", decimos que el e-mail fue enviado con éxito. Si es "**127**", decimos que está en proceso de envío. Pero si es negativo, menor a 0, imprimimos en pantalla un cartel de error.

En el ejemplo de la pagina Web, el email se envía cuando la variable "**SendMail**" se establece a "1" gracias a la entrada **DIN1**. Luego comprobamos el estado de la transmisión desde el evento **OnRtcSecond.sld**, en segundo plano al programa **Principal.sld**, ya que como hemos dicho, el proceso de envío de un mail puede tardar minutos y seria ineficiente comprobar constantemente desde **Principal.sld** el progreso de envío del e-mail.

Tarea: Modificar "**MailExample1Simple.zip**", para que en caso de error de transmisión, el PLC intente nuevamente el e-mail automáticamente.

[Ver también pagina 21 \(Probando los Ejemplos en Ladder y Pawn\).](#)



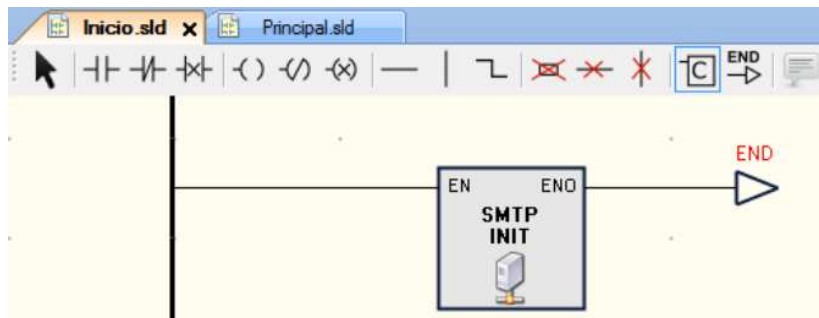
5.2 Enviar utilizando Servidor SMTP Externo

En caso de necesitar enviar emails utilizando un servidor SMTP externo (ver requerimientos en página 4) debemos utilizar y configurar el componente **Smtplnit** con los datos de acceso de nuestra cuenta en dicho servidor.

Desde nuestra página web es posible bajar el ejemplo "**MailExample1.zip**" que configura un servidor SMTP externo, el ejemplo es idéntico al utilizado en la sección 5 pág. 5 (ver **MailExample1Simple.zip**) pero reemplaza el componente **SmtplnitSimple** por **Smtplnit**.

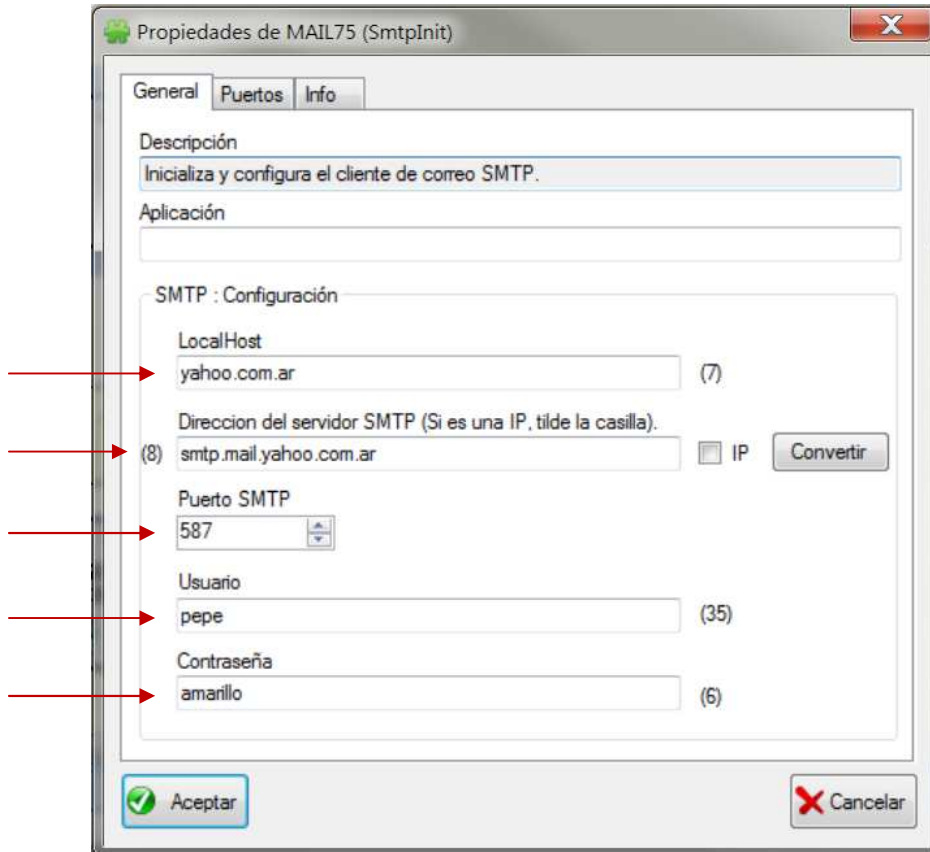
Entonces para utilizar el servidor externo, primero debemos configurar los parámetros del servidor **SMTP** con el componente **Smtplnit**. Es conveniente llamar a este componente desde el diagrama **Inicio.sld**, ya que la configuración en general se realiza una vez al comienzo del programa del PLC.

Insertamos el componente como se muestra a continuación:





Luego accedemos a sus propiedades haciendo doble-click sobre el componente:

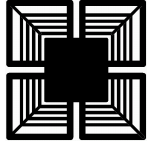


Las flechas indican los campos a completar para configurar el cliente de correo en el PLC. **Recuerde que puede acceder a la documentación completa del componente, desde el entorno StxLadder.**

En este caso utilizamos un servidor SMTP externo de ejemplo (como Yahoo, pero debería utilizar otro admitido) para enviar los e-mails. Por ello en la configuración especificamos:

- **LocalHost:** Nombre de nuestra red. En general utilizar el nombre de dominio del servidor SMTP. En este caso **yahoo.com.ar**
- **Dirección SMTP:** Nombre del servidor SMTP. En este caso **smtp.mail.yahoo.com.ar**.
- **Puerto SMTP:** Numero de puerto TCP del servidor SMTP. Utilice un puerto donde el servidor acepte conexiones no encriptados (por ejemplo: SSL / TLS, etc no están soportados). En este caso **587**.
- **Usuario:** Nombre de usuario para identificarse en el servidor. En este caso **pepe**.
- **Contraseña:** Clave o contraseña para identificarse en el servidor. En este caso **amarillo**.

Una vez configurado, puede utilizar el componente **SmtplibMailSend** para enviar los mails, como se mostró en la sección 5 pág. 5.



6 *Enviar E-Mail con Lenguaje Pawn*

En esta sección explicaremos a modo general como enviar un e-mail utilizando el lenguaje Pawn.

El proyecto completo puede descargarse desde nuestra página Web, con el nombre "**MailExample2Simple.zip**", el mismo contiene el código necesario para probar el cliente de correo electrónico y mostrar el progreso del envío en la pantalla de **VirtualHMI**.

El ejemplo en lenguaje Pawn, tiene el mismo funcionamiento que el ejemplo **MailExample1Simple.zip** para lenguaje Ladder.

Hay tres funciones claves para el envío del E-mail en Pawn:

- **SmtplnitSimple**: Inicializa los parámetros del PLC para utilizar un servidor SMTP propio.
- **SmtplibSend**: Envía el e-mail a los destinatarios especificados.
- **SmtplibGetStatus**: Obtiene el estado del envío del e-mail (útil para comprobar si existe un error o si el mail fue entregado con éxito).

En la página siguiente se describen dichas funciones en detalle.



6.1 Funciones Nativas en Pawn Disponibles

SmtplnitSimple(): Inicializa librería para que el PLC pueda enviar e-mails a través de un servidor SMTP del PLC.

Argumentos	Tipo	Descripción
-	-	
Retorno	Tipo	Descripción
0	S	Operación exitosa.
-1	S	Error, transmisión en progreso o librería en uso.
-2	S	Error, dirección de string errónea.
Notas		Descripción
1		Es necesario estar conectado a internet para luego poder enviar e-mails.

Ejemplo:

```
// Inicializar librería utilizando servidor SMTP del PLC.  
SmtplnitSimple()
```



Smtplnit(LocalHost[], Server[], Port, User[], Pass[], Opt): Inicializa y configura los parámetros para que el PLC pueda conectarse a un servidor SMTP externo cuando envíe un e-mail.

Argumentos	Tipo	Descripción
LocalHost	E	Nombre de nuestra red. En general utilizar el nombre de dominio del servidor SMTP. Máximo 19 caracteres.
Server	E	Nombre del servidor SMTP. Si es una dirección IP (ejemplo "192.168.1.15"), utilizar la constante SMTP_OPT_USE_IP en el argumento Opt . Máximo 30 caracteres.
Port	E	Numero de puerto TCP del servidor SMTP. Utilice un puerto donde el servidor acepte conexiones no encriptados (por ejemplo: SSL / TLS, etc no están soportados).
User	E	Nombre de usuario para identificarse en el servidor. Máximo 39 caracteres.
Pass	E	Clave o contraseña para identificarse en el servidor. Máximo 14 caracteres.
Opt	E	Opciones. Si no especifica ninguna, utilice la constante SMTP_OPT_DEFAULT.
Retorno	Tipo	Descripción
0	S	Operación exitosa.
-1	S	Error, transmisión en progreso o librería en uso.
-2	S	Error, dirección de string errónea.
Notas		Descripción
1		Recuerde utilizar un servidor SMTP que permita acceso del tipo "AUTH LOGIN", ver requerimientos en página 4.
2		Si el servidor externo se encuentra en la red local, no es necesario una conexión a internet del PLC para enviar los e-mail.

Ejemplo:

```
// Inicializar parámetros utilizando un servidor SMTP de Arnet.  
Smtplnit("arnet.com.ar", "smtp.arnet.com.ar", 587, "pepe", "amarillo",  
SMTP_OPT_DEFAULT)
```



SmtplibSend(To[], Cc[], From[], Subject[], Message[], ...): Envía un e-mail a través de un servidor SMTP previamente configurado.

Argumentos	Tipo	Descripción
To	E	Dirección de correo electrónico del destinatario. Máximo 35 caracteres.
Cc	E	Dirección de correo electrónico para copia. Utilizar "0" si no se especifica. Máximo 35 caracteres.
From	E	Dirección de correo electrónico del remitente. Máximo 35 caracteres.
Subject	E	Asunto del e-mail. Máximo 30 caracteres.
Message	E	Mensaje del e-mail, puede contener formato. Máximo 160 caracteres.
...	E	Lista de argumentos variables, de acuerdo a si se especifica formato.
Retorno	Tipo	Descripción
0	S	Operación exitosa. El e-mail fue aceptado para el envío, comprobar estado de librería con SmtplibGetStatus() para determinar si pudo ser transmitido.
-1	S	Error, transmisión en progreso o librería en uso. Intentar luego.
-2	S	Error, la conexión no pudo ser asignada.
-3	S	Error, el nombre del servidor SMTP no pudo ser resuelto. Comprobar DNS.
-4	S	Error, la librería no fue inicializada con SmtplibInit.
-5	S	Error, el Resolver esta en uso por otra función.
-20	S	Error, dirección de string errónea.
-21	S	No se pudo asignar memoria dinámica para la función.
Notas		Descripción
1		Cuidar extensión del mensaje cuando se utiliza formato y la lista de argumentos variables.
2		El formato de los mensajes, utiliza el mismo sistema que las funciones para imprimir mensajes en display LCD de VirtualHMI o del PLC.

Ejemplo:

```
// Enviar un e-mail a juan@gmail.com con copia a alarma@hotmail.com.  
// Notar como transmitimos el valor de la entrada discreta DIN1 utilizando  
// formato en el mensaje del e-mail con el código %d.  
  
SmtplibSend("juan@gmail.com", "alarma@hotmail.com", "pepe@yahoo.com.ar",  
"Mensaje del PLC", "La entrada discreta DIN1 tiene el valor %d.",  
DinValue(DIN1))
```

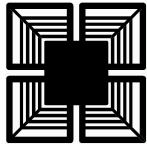



SmtGetStatus(): Obtiene el estado de la librería SMTP, que permite conocer el estado de envío de un e-mail o posibles errores.

Argumentos	Tipo	Descripción
		Ninguno
Retorno	Tipo	Descripción
127	S	El email está en proceso de envío.
126	S	La librería no fue inicializada con Smtplnit() .
0	S	No hay errores , el e-mail fue enviado con éxito.
-1	S	Error, la conexión al servidor fue cerrada o expiró.
-2	S	Error, no se recibió el código 220 del servidor.
-3	S	Error en proceso "EHLO".
-4	S	Error en proceso "AUTH".
-5	S	Error en proceso "USERNAME". Verifique nombre de usuario.
-6	S	Error en proceso "PASSWORD". Verifique contraseña.
-7	S	Error al enviar campo "FROM".
-8	S	Error al enviar campo "TO".
-9	S	Error al enviar campo "CC".
-10	S	Error al enviar campo "DATA".
-11	S	Error al enviar "." para finalizar mensaje.
-12	S	Error, conexión cerrada por el servidor.
-13	S	Error, el nombre del servidor no puede ser resuelto.
-14	S	Error, la conexión no pudo ser creada.
-15	S	Error. Ver código de error retornado por SmtSend() .

Ejemplo:

```
// Obtener código de estado.  
SmtStatus = SmtGetStatus()
```



SmtpNetFromName(FromName): Permite establecer el nombre del remitente del e-mail en el campo "From".

Argumentos	Tipo	Descripción
FromName	E	Nombre del remitente. Máximo 20 caracteres. Por defecto se utiliza "PLC DEVICE".
Retorno	Tipo	Descripción
0	S	Operación exitosa.
-1	S	Error, transmisión en progreso o librería en uso.
-2	S	Error, dirección de string errónea.
Notas		Descripción
-		

Ejemplo:

```
// Establecer el nombre del remitente como "Slicetex Electronics".  
SmtpNetFromName("Slicetex Electronics")
```



6.2 Ejemplo Completo en Pawn

El proyecto completo puede descargarse desde nuestra página Web, con el nombre "**MailExample2Simple.zip**", el mismo contiene el código necesario para probar el cliente de correo electrónico y mostrar el progreso del envío en la pantalla de **VirtualHMI**.

El ejemplo comienza inicializando los parámetros de conexión con **SmtplnitSimple()**, luego inicializa NetHMI para usar **VirtualHMI** y finalmente el evento **OnRtcSecond()**.

Se crean 4 variables globales:

```
// Flag, si el valor es 1 hay un mail pendiente de envío.
new Sending

// Flag, si el valor es 1, el PLC debe enviar un mail.
new SendMail

// Almacena el código de estado de una transmisión o librería SMTP.
new SntpStatus

// Flag, si el valor es 1, existe un error en la transmisión del mail.
new SendError
```

Entonces, desde el loop principal, comprueba si debe enviarse un e-mail cuando la entrada Discreta **DIN1** es "1" y la variable **Sending** es "0". En caso afirmativo establece **SendMail** a "1" para comenzar a enviar un mail desde el evento **OnRtcSecond()**.

```
for(;;)
{
    // Loop principal.

    //
    // Comprobar que la entrada discreta DIN1 sea igual 1
    // y no exista una transmisión en progreso.
    //
    // Si se cumple la condición, enviar un E-Mail.
    //

    if(DinValue(DIN1) == 1 && Sending == 0)
    {
        // Enviar E-Mail estableciendo variable a 1.
        // En VirtualHMI se mostrará el progreso del envío.

        SendMail = 1
    }
}
```

El evento **OnRtcSecond()** se llama constantemente cada 1 segundo y es el encargado de enviar el e-mail, comprobar estado de transmisión, imprimir mensajes en **VirtualHMI** y leer las 4 variables globales.



Utilizamos un evento, para liberar al PLC de comprobar constantemente la transmisión desde el loop principal, que puede tardar decenas de segundos a minutos. De esta forma comprobamos en “segundo plano” mientras podemos utilizar el loop principal para otras tareas.

Desde **OnRtcSecond()** enviamos el e-mail con **SmtplibSend()** cuando **Sending** es 0 y **SendMail** es 1:

```
//
// Enviar un E-MAIL si no hay una transmisión en progreso
// y si la variable SendMail es igual a 1.
//

if(Sending == 0 && SendMail == 1)
{
    if(SmtplibSend("...")
    {
        // Mail aceptado para envío, establecer a "1" variable.
        Sending = 1
    }
}
```

Y también obtenemos códigos de estado con **SmtplibGetStatus()** si hay un mail en progreso o intento de envío:

```
if(Sending == 1 || SendMail == 1)
{
    // Obtener código de estado.
    SmtplibStatus = SmtplibGetStatus()

    // Imprimir código en VirtualHMI.
    nLcdPrintf(0,0, LCD_CLRLINE, "Smtplib Stat = %d", SmtplibStatus)
}
```

[Ver también pagina 21 \(Probando los Ejemplos en Ladder y Pawn\).](#)

6.3 Enviar utilizando Servidor SMTP Externo

En caso de necesitar enviar emails utilizando un servidor SMTP externo (ver requerimientos en página 4) debemos utilizar y configurar la función **SmtplibInit()** con los datos de acceso de nuestra cuenta en dicho servidor.

Desde nuestra página web es posible bajar el ejemplo “**MailExample2.zip**” que configura un servidor SMTP externo, el ejemplo es idéntico al utilizado en la sección 6.2 pág. 19 (ver **MailExample2Simple.zip**) pero reemplaza la función **SmtplibInitSimple()** por **SmtplibInit()**.

Recuerde configurar correctamente el nombre de usuario y contraseña del servidor SMTP antes de enviar el e-mail.



7 Probando los Ejemplos en Ladder y Pawn

Para probar los ejemplos **MailExample1Simple.zip** (Ladder) o **MailExample2Simple.zip** (Pawn), primero los abrimos con StxLadder, luego debemos seleccionar nuestro modelo de PLC y finalmente compilar para transferir el programa al PLC.

Luego debemos configurar la IP de la computadora en la cual corre **VirtualHMI** para así poder ver el estado de la transmisión del mail y posibles errores.

Por lo tanto, los pasos son:

1. Configurar dirección IP donde se ejecuta **VirtualHMI**.
2. Ejecutar **VirtualHMI** y activarlo.
3. Compilar proyecto y transferirlo al PLC.
4. Resetear PLC.
5. **Activar la entrada DIN1 del PLC, así enviamos el e-mail.**
6. Ver en VirtualHMI el estado de la transmisión.
7. Si el mail fue transmitido, ver en la casilla de e-mail el mensaje.

A continuación, mostramos algunas capturas de pantalla del proceso de envío:

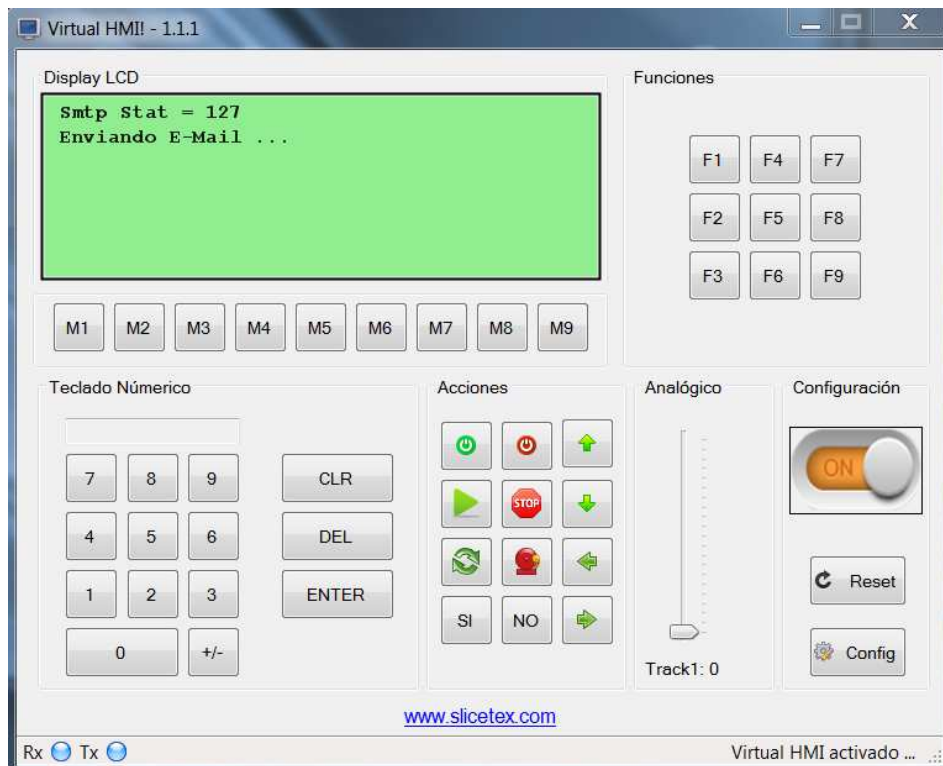


Fig 2: El PLC está enviando e-mail, código 127

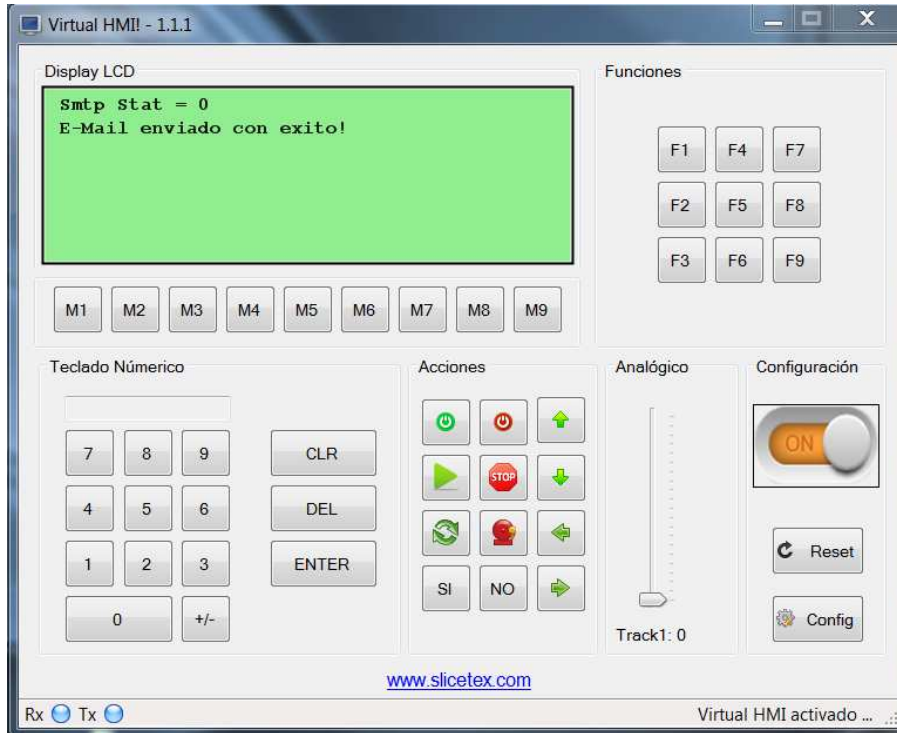
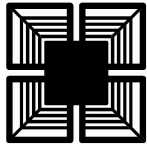


Fig 3: El PLC envió el mail con éxito, código 0.

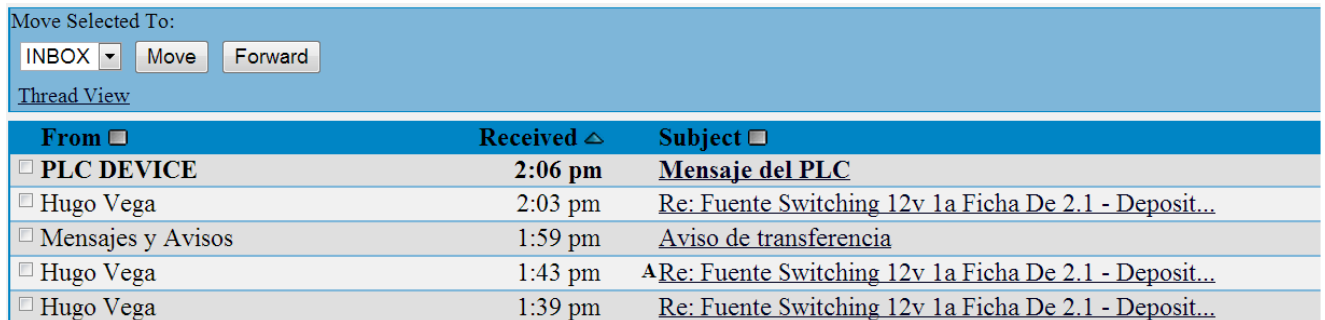


Fig 4: Bandeja de entrada, nuevo mail recibido de “PLC DEVICE” !



Subject: Mensaje del PLC
From: "PLC DEVICE" <boris_estudiez@yahoo.com.ar>
Date: Tue, December 18, 2012 12:05 pm
To: devel@slidetex.com
Priority: Normal
Options: [View Full Header](#) | [View Printable Version](#) | [Download this as a file](#)

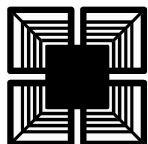
```
Soy un PLC que envia E-MAILS!.  
La entrada discreta DIN1 tiene el valor "0".  
Saludos.
```

Fig 5: Contenido del e-mail recibido. Notar como el valor de DIN1 fue reemplazado.

7.1 En caso de Error

Si por algún motivo no recibe los e-mails, compruebe lo siguiente:

- Si el PLC envió con éxito el mail, es probable que su mail se encuentre en la carpeta de "Spam" o "Correo no deseado". Verifique dichas carpetas. Si lo encuentra, seleccione desde su cliente de correo el mail y la opción "No es spam" / "Marcar como deseado" o alguna similar que lo clasifique como correo deseado. Agregue también a su lista de contacto las direcciones info@slidetex.com, plc.device@slidetex.com para avisar a su cliente de correo que son direcciones válidas de correo para futuros mensajes. Es recomendable agregar también la dirección que está utilizando en el campo "**From**" para enviar su correo a su lista de contactos.
- Si el mensaje no puede ser enviado, compruebe que disponga de conexión a internet, que el PLC tenga configurado la dirección DNS correctamente (por ejemplo 8.8.8.8, desde menú "**PLC > Configurar PLC > Pestaña Red**"), o intente más tarde (puede que los servidores estén ocupados).
- Si utiliza un servidor SMTP externo, compruebe que sus datos de acceso sean correcto en la configuración de **Smtplnit**. También que cumpla los requerimientos de la sección 4.1 en página 4. Analice los códigos de estados retornados en los errores para determinar la causa.
- Si su PLC dispone de reloj RTC, configure la hora/fecha correctamente, será utilizada en el momento de enviar emails a menos que la opción **SMTP_OPT_NO_SEND_DATE** sea utilizada en Smtplnit(). Algunos servidores no exigen la hora/fecha y puede utilizar la opción **SMTP_OPT_NO_SEND_DATE** para no enviarla en caso de obtener errores relacionados.
- Por otros errores, puede consultar en nuestro foro de soporte técnico (www.slidetex.com/foro) suministrando su proyecto o código de prueba que da error, y toda la información posible para que podamos analizar la causa.



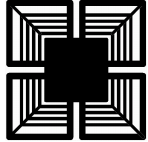
8 Abreviaciones y Términos Empleados

- **PLC:** Programable Logic Controller (Controlador Lógico Programable).
- **IP:** Dirección Internet, conformada por cuatro octetos, por ejemplo 192.168.1.81.
- **SMTP:** Simple Mail Transfer Protocol (Usualmente un servidor que permite enviar emails).
- **Ethernet:** Red de computadoras, que generalmente se utilizan el protocolo de internet TCP/IP o UDP/IP.
- **AUTH LOGIN:** Método de acceso a un servidor SMTP sin encriptación.

9 Historial de Revisiones

Tabla: Historia de Revisiones del Documento

Revisión	Cambios	Descripción	Estado
05 22/MAY/2017	1	1. Se menciona constante SMTP_OPT_NO_SEND_DATE en sección 7.1, pág. 23.	Estable
04 31/OCT/2016	1	1. Cambios para actualizar documento a versiones modernas de software y firmware.	Estable
03 21/SEP/2016	5	1. Agrega descripción de función y componente SmtplnitSimple() para utilizar servidor SMTP del PLC. 2. Cambio de ejemplos principales por MailExampleSimple1 y MailExampleSimple2. 3. Cambios en secciones 3, 4, 4.1, 5, 6 y 7. 4. Nueva sección "7.1 En caso de Error" en página 23. 5. Otros cambios menores de redacción y visuales.	
02 20/OCT/2013	1	6. Agrega descripción de función SmtplibSetFromName(). 7. Modifica longitud de caracteres en SmtplibSend(), argumento Subject.	Preliminar
01 18/DEC/2012	1	1. Versión preliminar liberada.	Preliminar



10 Referencias

Ninguna.

11 Información Legal

11.1 Aviso de exención de responsabilidad

General: La información de este documento se da en buena fe, y se considera precisa y confiable. Sin embargo, Slicetex Electronics no da ninguna representación ni garantía, expresa o implícita, en cuanto a la exactitud o integridad de dicha información y no tendrá ninguna responsabilidad por las consecuencias del uso de la información proporcionada.

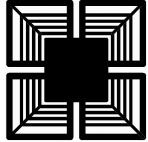
El derecho a realizar cambios: Slicetex Electronics se reserva el derecho de hacer cambios en la información publicada en este documento, incluyendo, especificaciones y descripciones de los productos, en cualquier momento y sin previo aviso. Este documento anula y sustituye toda la información proporcionada con anterioridad a la publicación de este documento.

Idoneidad para el uso: Los productos de Slicetex Electronics no están diseñados, autorizados o garantizados para su uso en aeronaves, área médica, entorno militar, entorno espacial o equipo de apoyo de vida, ni en las aplicaciones donde el fallo o mal funcionamiento de un producto de Slicetex Electronics pueda resultar en lesiones personales, muerte o daños materiales o ambientales graves. Slicetex Electronics no acepta ninguna responsabilidad por la inclusión y / o el uso de productos de Slicetex Electronics en tales equipos o aplicaciones (mencionados con anterioridad) y por lo tanto dicha inclusión y / o uso es exclusiva responsabilidad del cliente.

Aplicaciones: Las aplicaciones que aquí se describen o por cualquiera de estos productos son para fines ilustrativos. Slicetex Electronics no ofrece representación o garantía de que dichas aplicaciones serán adecuadas para el uso especificado, sin haber realizado más pruebas o modificaciones.

Los valores límites o máximos: Estrés por encima de uno o más valores límites (como se define en los valores absolutos máximos de la norma IEC 60134) puede causar daño permanente al dispositivo. Los valores límite son calificaciones de estrés solamente y el funcionamiento del dispositivo en esta o cualquier otra condición por encima de las indicadas en las secciones de Características de este documento, no está previsto ni garantizado. La exposición a los valores limitantes por períodos prolongados puede afectar la fiabilidad del dispositivo.

Documento: Prohibida la modificación de este documento en cualquier medio electrónico o impreso, sin autorización previa de Slicetex Electronics por escrito.



12 Información de Contacto

Para mayor información, visítenos en www.slicetex.com

Para información general, envíe un mail a: info@slicetex.com

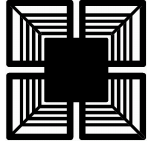
Slicetex Electronics
Córdoba, Argentina

© Slicetex Electronics, todos los derechos reservados.



13 Contenido

<u>1</u>	<u>DESCRIPCIÓN GENERAL.....</u>	<u>1</u>
<u>2</u>	<u>LECTURAS RECOMENDADAS.....</u>	<u>2</u>
<u>3</u>	<u>REQUERIMIENTOS</u>	<u>2</u>
<u>4</u>	<u>TEORÍA DE FUNCIONAMIENTO</u>	<u>3</u>
4.1	REQUERIMIENTOS DEL SERVIDOR SMTP EXTERNO	4
<u>5</u>	<u>ENVIAR E-MAIL CON LENGUAJE LADDER</u>	<u>5</u>
5.1	VER ESTADO DE ENVÍO DEL E-MAIL	9
5.2	ENVIAR UTILIZANDO SERVIDOR SMTP EXTERNO.....	11
<u>6</u>	<u>ENVIAR E-MAIL CON LENGUAJE PAWN</u>	<u>13</u>
6.1	FUNCIONES NATIVAS EN PAWN DISPONIBLES	14
6.2	EJEMPLO COMPLETO EN PAWN.....	19
6.3	ENVIAR UTILIZANDO SERVIDOR SMTP EXTERNO.....	20
<u>7</u>	<u>PROBANDO LOS EJEMPLOS EN LADDER Y PAWN.....</u>	<u>21</u>
7.1	EN CASO DE ERROR	23
<u>8</u>	<u>ABREVIACIONES Y TÉRMINOS EMPLEADOS.....</u>	<u>24</u>
<u>9</u>	<u>HISTORIAL DE REVISIONES.....</u>	<u>24</u>
<u>10</u>	<u>REFERENCIAS</u>	<u>25</u>
<u>11</u>	<u>INFORMACIÓN LEGAL</u>	<u>25</u>
11.1	AVISO DE EXENCIÓN DE RESPONSABILIDAD.....	25
<u>12</u>	<u>INFORMACIÓN DE CONTACTO</u>	<u>26</u>



13 **CONTENIDO** **27**

Copyright Slicetex Electronics 2016
www.slicetex.com