

For eWON® with a modem

1 Summary

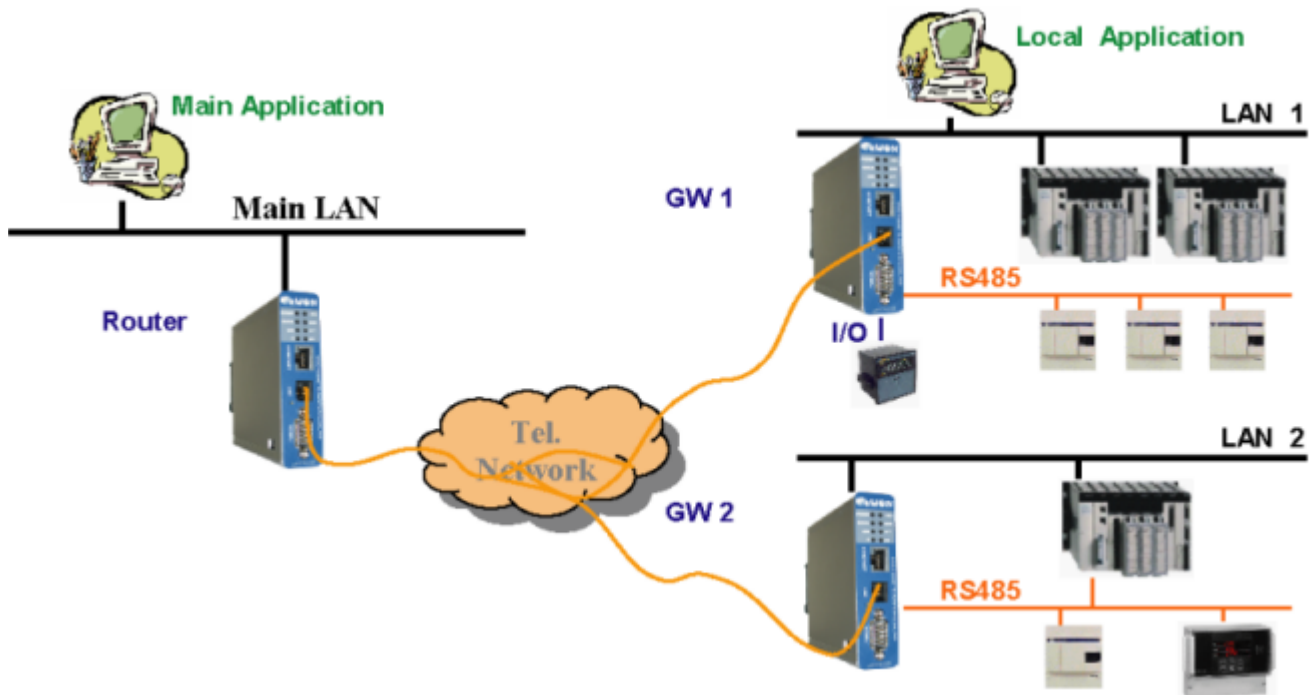


Figure 1: Multiple networks with eWON as a gateway

Consider the architecture which is drawn above.

Assume you have a central application which has to be temporarily connected to various LANs (Local Area Networks) via telephone lines.

This Technical Note introduces by an example how to configure eWON devices in order to let the Main Application be connected to various applications once at a time.

The eWON connected to the Main LAN will act as a Router able to call any of the other eWONs.

The whole set of resources behind a called eWON becomes accessible for the Main LAN Applications.

2 Example refinement

In order to refine our example, we will assume that the Main Application is a monitoring system with either a MODBUS-RTU master driver or a MODBUS-TCP client one. This is of course absolutely not mandatory to take benefit of this Technical Note by using the explained technique.

Remember that eWON is a full MODBUS router (RTU slave units are accessible from TCP clients). Furthermore, you can use a MODBUS-RTU master driven Windows application as it was a MODBUS-TCP client driven one, provided you use the Virtual COM utility (MbsTcpCOM) that you can download from the eWON website www.ewon.biz (Support/Download Software).



Accessing Multiple Networks with eWON Router-Gateway

TN 22

ver 1.1

03/11/2004

For eWON® with a modem

Assume we have the following Ethernet network IP configuration (pay attention to the gateWay dependencies):

Equipment	IP address	Gateway address	Subnet Mask
Main LAN PC	10.0.0.5	10.0.0.53	255.255.255.0
Main LAN eWON	10.0.0.53	0.0.0.0	255.255.255.0
LAN 1 eWON	20.0.0.26	0.0.0.0	255.255.255.0
LAN 1 PC	20.0.0.47	20.0.0.26	255.255.255.0
LAN 1 PLC1	20.0.0.104	20.0.0.26	255.255.255.0
LAN 1 PLC2	20.0.0.105	20.0.0.26	255.255.255.0
LAN 2 eWON	30.0.0.19	0.0.0.0	255.255.255.0
LAN 2 PLC1	30.0.0.67	30.0.0.19	255.255.255.0

Table 1: Ethernet network IP configuration

3 Tag Configuration of the LAN 1 eWON (Router)

Let us specify the expected behaviour of our system. We would like:

- The Main LAN PC writing and reading Modbus Tags in the LAN1 eWON to drive the whole system
- The LAN1 eWON to consequently call the local LAN eWON as configured in the LAN1 eWON
- The LAN1 eWON to inform the Main LAN PC about the connection status

We will set the following eWON Tags (see eWON User Guide on eWON web site for further information):

Tag Name	Type	Mode	Modbus Visibility (enabled)	Role and Behaviour
GWconnect	Boolean	R / W	2: Read Status Write Coil	Request connection
GWdisconnect	Boolean	R / W	4: Read Status Write Coil	Request Hang Up
GWnumber	Analog	R / W	6: Read Output Register Preset single Register	Change targeted eWON
GWonline	Boolean	R	8: Read Status ---	PPP connection activity
GWstatus	Analog	R	10: Read Input Register ---	Connection cycle status

Table 2: eWON Tags configuration

Remember that if you want to operate from a Modbus TCP Client, you have to configure the eWON MODBUS IO Server in order to enable the eWON MODBUS TCP Server (it will work by default with Unit number 100). Modbus TCP server is enabled from the Tag setup page.

If you do not operate with MODBUS, you can browse the eWON standard web site (or write your own custom page) to operate the GWxxx variables.



For eWON® with a modem

4 BASIC Program of the LAN 1 eWON (Router)

We add 5 Sections (**CheckPPP**, **ChooseGW**, **Connect**, **HangUp**, **Status**) to the default **Init** & **Cyclic** Sections:

Init Section

```
01 onchange "GWnumber","goto ChooseGW"
02 onchange "GWconnect","goto Connect"
03 onchange "GWdisconnect","goto HangUp"
04 onchange "GWonline", "goto Status"
05 tset 1,1
06 ontimer 1,"goto CheckPPP"
07 GWnumber@=1
```

As you can see, four sections are executed on GWxxx Tag value changes.

The checkPPP section will be executed every second, according to timer 1 trigger event.

By default (when eWON starts), the targeted gateway is the first one (ChooseGW section is called when GWnumber is set).

Cyclic Section (Not used)

CheckPPP Section

```
01 CheckPPP:
02 a$=getsys prg,"PPPIP"
03 if a$="0.0.0.0" then
04   GWonline@=0
05 else
06   GWonline@=1
07 endif
```

At each timer 1 trigger event (every second), the IP address of the PPP connection is tested. If the returned value is 0.0.0.0, there is no connection, otherwise the returned value contains the IP address assigned by the PPP Server of the targeted eWON of the local LAN.

(See GETSYS and SETSYS basic functions in the eWON User Manual)

Status Section

```
01 Status:
02 if GWonline@=1 then
03   GWstatus@=2
04 else
05   GWstatus@=0
06   setsys com,"load"
07   setsys com,"DialInOut",0
08   setsys com,"save"
09 endif
```

When the PPP connection becomes active, GWonline changes to 1 and GWstatus is set to 2.

When the PPP connection becomes inactive, GWonline changes to 0 and GWstatus is reset to 0.

eWON DialInOut mode is set to 0 (Disabled) to avoid any undesired Dial Up of the eWON if the main LAN PC is trying to reach an IP address outside its subnet via the eWON (defined as the gateway of that main LAN PC)



For eWON® with a modem

ChooseGW Section

```

01 ChooseGW:
02 t% = GWnumber@
03 logevent "GATEWAY: Set GateWay"+str$(t%)+ " for connection"
04 if t%=1 then
05   setsys com,"load"
06   setsys com,"PPPCIPhone1","22"
07   setsys com,"PPPCUserName1","Adm"
08   setsys com,"PPPCIPassword1","adm"
09   setsys com,"save"
10   setsys sys,"load"
11   setsys sys,"SmtServerAddr","20.0.0.26"
12   setsys sys,"save"
13 endif
14 if t%=2 then
15   setsys com,"load"
16   setsys com,"PPPCIPhone1","24"
17   setsys com,"PPPCUserName1","Adm"
18   setsys com,"PPPCIPassword1","adm"
19   setsys com,"save"
20   setsys sys,"load"
21   setsys sys,"SmtServerAddr","30.0.0.19"
22   setsys sys,"save"
23 endif

```

The GWnumber is retrieved in the t% variable.
A message is sent to the event log file to trace the change of the targeted eWON (line 03)

Each Gateway configuration corresponds to a set of four parameters :

PPPCIPhone1: (lines 06, 16, ...)

phone number of primary server

PPPCUserName1: (lines 07, 17, ...)

name of a valid user of the targeted eWON

PPPCIPassword1: (lines 08, 18, ...)

password of the user of the targeted eWON

SmtServerAddr: (lines 11, 21, ...)

Ethernet IP address of the Targeted eWON

The first three parameters belong to the COM set of configuration parameters while the fourth one belongs to the SYS set.

(See eWON User Manual for details on COM and SYS sets of configuration parameters)

Add as many gateways as you need.

Connect Section

```

01 Connect:
02 onerror "goto ErrHandler"
03 if (GWconnect@=1) then
04   GWstatus@=1
05   setsys sys,"load"
06   a$= getsys sys,"SmtServerAddr"
07   g$= "tcp:"+a$+":25"
08   setsys com,"load"
09   setsys com,"DialInOut",2
10   setsys com,"save"
11   logevent"GATEWAY: request connect to GateWay"+str$(GWnumber@)+",120

```

When Gwconnect becomes 1 (a connection is requested to the currently defined GateWay), GWstatus is set to 1 (request acknowledge state).

The SmtServerAddr parameter is retrieved to build the g\$ string (lines 05 to 07).

The DialInOut parameter is set to 2 (Outgoing only) in order to allow the IP forwarding towards the targeted eWON (lines 08 to 10).

A trace is written in the even log file. (line 11)



For eWON® with a modem

```
12 close 1
13 open g$ for binary output as 1
14 gwstatus@=4
15 onchange "GWconnect", ""
16 GWconnect@=0
17 onchange "GWconnect", "goto Connect"
18 endif
19 end
20 ErrHandler:
21 gwstatus@=5
22 onchange "GWconnect", ""
23 GWconnect@=0
24 onchange "GWconnect", "goto Connect"
25 setsys com, "load"
26 setsys com, "DialInOut", 0
27 setsys com, "save"
28 logevent "GATEWAY: connect to GateWay"+str$(GWnumber@)+" failed", 120
```

Then, the eWON router opens a TCP socket on the port 25 of the targeted eWON. (remark the close 1 instruction at line 12 to ensure the channel is free).

If an error occurs, the code continues at ErrHandler.

If the request to open a socket is valid, the corresponding scheduled action is launched, the GWstatus becomes 4 (Scheduled action launched), and Gwconnect is reset to 0 (remark the onchange inhibition to avoid reexecution of the Connect section on Gwconnect change, see lines 15 to 17).

The socket is opened on the port 25 of the targeted eWON to ensure the scheduled action will fail

In case of error while requesting to open the TCP socket, the GWstatus becomes 5, GWconnect is reset to 0, the DialInOut mode is disabled and a trace is written in the event

HangUp Section

```
01 Hangup:
02 if GWonline@=0 then
03   GWstatus@ = 0
04 else
05   if GWdisconnect@=1 then
06     GWstatus@=3
07     logevent "GATEWAY: request HangUp"
08     setsys prg, "PPPIP", 0
09     setsys com, "load"
10     setsys com, "DialInOut", 0
11     setsys com, "save"
12   endif
13 endif
14 onchange "GWdisconnect", ""
15 GWdisconnect@=0
16 onchange "GWdisconnect", "goto HangUp"
```

If HangUp is requested while off line (GWonline=0), the Tag Gwstatus is reset to 0.

If HangUp is requested, GWstatus becomes 3 (HangUp requested while on line).

A trace is written in the event log file.

The DialInOut mode is disabled.

Then, GWdisconnect is reset to 0 while the onchange action is inhibited (lines 14 to 16).

For eWON® with a modem

5 Targeted eWONs Configuration (Gateways)

The parameters that are relevant to set the eWON gateway settings are:

- **The PPP Server IP address (202.0.0.240 by default, but you may use any suited value)**
- **The PPP Client IP address (202.0.0.1 by default, but you may use any suited value)**
- **The DialInOut must be set to "Incoming Only" (see eWON User Manual for further details)**
- **The IP Forwarding must be enabled (see eWON User Manual for further details)**

(remember that the IP Forwarding must also be enabled for eWON router on the Main LAN).

All the Ethernet connected devices behind an eWON gateway must refer to it as their own gateway.

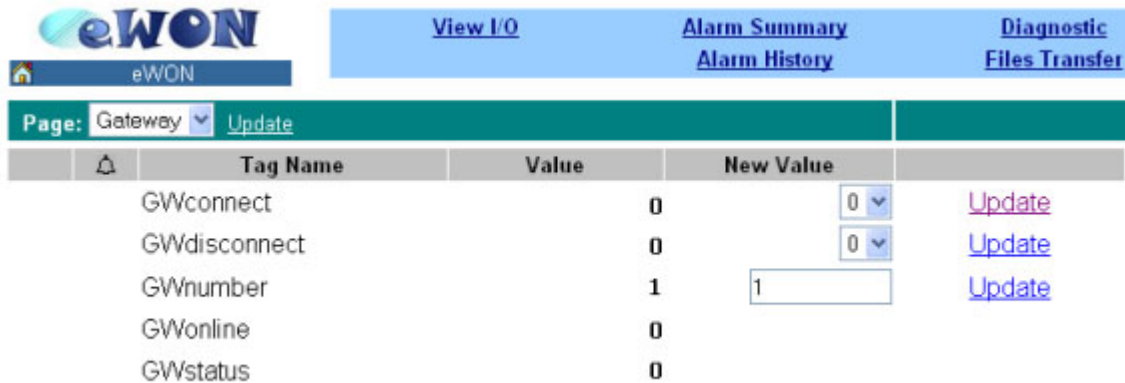
If you want to reach RTU Modbus devices, ensure the unit numbers are not conflictual.

If you want the targeted eWON gateway to publish Tags as MODBUS TCP variables, ensure that the MODBUS IO Server is enabled and that its unit is not in conflict with the potentially connected RTU units. By default, the eWON unit number for public Modbus TCP Tags is 100. (See eWON User Manual for further details on how to configure the eWON Modbus IO Server).

6 Example of use

We will now show how the system behaves (from the standard web site and with a MODBUS TCP Client).

When booting the eWON, one has in the View I/O page:



	Tag Name	Value	New Value	
	GWconnect	0	0	Update
	GWdisconnect	0	0	Update
	GWnumber	1	1	Update
	GWonline	0		
	GWstatus	0		

Figure 2: View I/O page at eWON boot

For eWON® with a modem

To change the Targeted eWON Gateway, simply set GWnumber to a valid number (for instance: 2):

0	<input type="text" value="0"/>	<input type="button" value="Update"/>
0	<input type="text" value="0"/>	<input type="button" value="Update"/>
1	<input type="text" value="2"/>	<input type="button" value="Update"/>
0		
0		

You can operate the same with your MODBUS TCPClient by requesting a write (Preset Single Register) on GWnumber Public MODBUS address:

IP address : 10.0.0.53

Unit : 100

Reg Address : 6 or 5

Remark : instead of 6 you might need to use 5 as the Tag Modbus address because of the 0 or 1 start address in the Modbus table conventions.

Figure 3: Change GWnumber Tag value

The result is:

GWconnect	0	<input type="text" value="0"/>	<input type="button" value="Update"/>
GWdisconnect	0	<input type="text" value="0"/>	<input type="button" value="Update"/>
GWnumber	2	<input type="text" value="2"/>	<input type="button" value="Update"/>
GWonline	0		
GWstatus	0		

Figure 4: GWnumber Tag value change result

Suppose you select again the first Gateway and ask for a connection:

0	<input type="text" value="1"/>	<input type="button" value="Update"/>
0	<input type="text" value="0"/>	<input type="button" value="Update"/>
1	<input type="text" value="1"/>	<input type="button" value="Update"/>
0		
0		

You can operate the same with your MODBUS TCP Client by requesting a write (Preset Single Register) on GWnumber Public MODBUS address:

IP address : 10.0.0.53

Unit : 100

Reg Address : 2 or 1

Remark : instead of 2 you might need to use 1 as the Tag Modbus address because of the 0 or 1 start address in the Modbus table conventions.

Figure 5: Selection of the first gateway

For eWON® with a modem

the result is that the Gateway status is set to 1 (the request for connection is acknowledged):

Page: Gateway Update				
	Tag Name	Value	New Value	
	GWconnect	1	1	Update
	GWdisconnect	0	0	Update
	GWnumber	1	1	Update
	GWonline	0		
	GWstatus	1		

Figure 6: Gateway status set to 1

Update the Page and observe the following (if you obtain GWstatus = 5, there was an error in the basic code)

GWconnect	0	0	Update
GWdisconnect	0	0	Update
GWnumber	1	1	Update
GWonline	0		
GWstatus	4		

Figure 7: View I/O page update result

If you look at the scheduled actions table:

File Name	Description
Events.htm	Events log as table
sstat.htm	Scheduled status as table
estat.htm	eWON system status as table
rt_alm.bt	Real time alarms
inst_val.bt	Instantaneous values as text

Figure 8: Selecting the Scheduled Actions Table

You will read "TCP/IP Connect" in the **ActionType** field.



For eWON® with a modem

You will read "In Progress" in the **StatusText** field.

ActionId	ActionType	StatusCode	StatusText	Start	End
0	TCP/IP Connect	-1	In Progress	28/10/2004 16:10:16	28/10/2004 16:11:30

Figure 9: sstat.htm file sample

Update the View I/O page several times until you observe GWstatus = 2:

GWconnect	0	<input type="text" value="0"/>	Update
GWdisconnect	0	<input type="text" value="0"/>	Update
GWnumber	1	<input type="text" value="1"/>	Update
GWonline	1		
GWstatus	2		

Figure 10: GWstatus value worth 2

Now, the PPP communication is established with the targeted eWON. Look again in the scheduled action table:

ActionId	ActionType	StatusCode	StatusText	Start	End
0	TCP/IP Connect	21703	Connect socket failed	28/10/2004 16:10:16	28/10/2004 16:11:07

Figure 11: Connect socket failed

The socket connection failed (indeed eWON is not able to serve a TCP client on the port 25). If you used another IP address to open a socket (line 13 of Connect Section), actually any IP address outside the Main LAN eWON subnet range, the scheduled action would remain "In Progress" until all TCP/IP time-out processes are finished. Using the IP address of the Targeted eWON ensures the Scheduled Action cycle is closed as soon as the communication is established.

As of the moment GWstatus = 2 and Gwonline = 1, the route is established between the Main Lan Network and the Targeted Lan Network. You can use it to reach other devices (PCs, PLCs, controllers, ...).

If you use a MODBUS TCP Client, you can monitor these two GW... variables to start a Topic polling the Targeted Lan desired variables.

You are now working as you were crossing two ordinary gateways...

Any Targeted LAN application is also able to perform TCP or UDP requests on the MAIN LAN network.

(IP Forwarding is bilateral).

For eWON® with a modem

Now, we can proceed for disconnecting the route...

GWconnect	0	<input type="text" value="0"/>	Update
GWdisconnect	0	<input type="text" value="1"/>	Update
GWnumber	1	<input type="text" value="1"/>	Update
GWonline	1		
GWstatus	2		

Figure 12: Disconnecting the routing

You get the following page result with GWstatus = 3 (Hang Up requested while online):

Page: Gateway Update				
	Tag Name	Value		New Value
	GWconnect	0	<input type="text" value="0"/>	Update
	GWdisconnect	1	<input type="text" value="1"/>	Update
	GWnumber	1	<input type="text" value="1"/>	Update
	GWonline	1		
	GWstatus	3		

Figure 13: GWstatus = 3 (Hang Up requested while online)

After several Updates of the page, you finally close the connection cycle.

GWconnect	0	<input type="text" value="0"/>	Update
GWdisconnect	0	<input type="text" value="0"/>	Update
GWnumber	1	<input type="text" value="1"/>	Update
GWonline	0		
GWstatus	0		

Figure 14: Enclosure of the connection cycle

For eWON® with a modem

Now have a look at the Events log file:

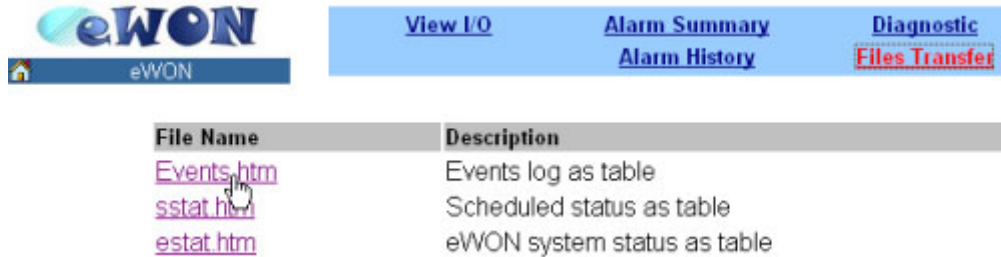


Figure 15: Opening the eWON Events file

You will be able to trace all the GATEWAY transactions

EventTimeInt	EventTimeStr	EventStr	ThreadStr	ThreadId	Event
1098979816	28/10/2004 17:28:43	GATEWAY : request connect to GateWay1.00@ tcp:20.0.0.76:25	scp	7	1073766944
1098979816	28/10/2004 17:28:43	ppp-Dial up required to connect to (To: 20.0.0.76)	unact	2	1073763855
1098979848	28/10/2004 17:29:10	smtp-Connect socket failed	unact	2	21703
1098979859	28/10/2004 17:33:50	GATEWAY : request connect to GateWay1.00@ tcp:20.0.0.76:25	scp	7	1073766944
1098979859	28/10/2004 17:33:50	smtp-Connect socket failed	unact	2	21703
1098979880	28/10/2004 17:34:32	GATEWAY : request HangUp	scp	7	25100
1098979883	28/10/2004 17:34:45	GATEWAY : request connect to GateWay1.00@ tcp:20.0.0.76:25	scp	7	1073766944
1098979883	28/10/2004 17:34:45	ppp-Dial up required to connect to (To: 20.0.0.76)	unact	2	1073763855
1098979916	28/10/2004 17:35:12	smtp-Connect socket failed	unact	2	21703
1098979984	28/10/2004 17:44:19	GATEWAY : request HangUp	scp	7	25100
1098980008	28/10/2004 17:46:12	GATEWAY : set GateWay2 for connection	scp	7	25100
1099996620	28/10/2004 17:48:00	GATEWAY : set GateWay2 for connection	scp	7	25100

Figure 16: Gateway transactions tracking in the Events file