

# SCHNELLANLEITUNG FÜR SEAL FÜR RTU SENECA Z-GPRS3 Z-UMTS Z-LOGGER3

## SENECA s.r.l.

Via Austria 26, PADUA – ITALIEN

Tel. +39.049.8705355 – 8705359 Fax. +39.049.8706287

Web-Site: [www.seneca.it](http://www.seneca.it)

Kundendienst: [supporto@seneca.it](mailto:supporto@seneca.it) (IT), [support@seneca.it](mailto:support@seneca.it)  
(Sonstiges)

Kaufmännische Informationen: [commerciale@seneca.it](mailto:commerciale@seneca.it) (IT),  
[sales@seneca.it](mailto:sales@seneca.it) (Sonstiges)



Dieses Dokument ist Eigentum von SENECA srl. Ohne Genehmigung ist die Vervielfältigung und Wiedergabe (auch auszugsweise) untersagt. Die Inhalte des vorliegenden Dokuments beziehen sich auf die darin beschriebenen Produkte und Technologien. Obschon wir uns ständig bemühen, die Perfektion zu erreichen, können alle in diesem Dokument enthaltenen technischen Daten aufgrund von technischen oder handelstechnischen Anforderungen geändert werden; es ist nicht möglich, Fehler oder Abweichungen vollständig zu vermeiden. Die Inhalte des vorliegenden Dokuments unterliegen periodischen Aktualisierungen. Bitte schreiben Sie uns an die oben

MI004474-D

Datum	Version	Änderungen
29/06/2016	1.00	Erste Revision
22/07/2016	1.02	Kapitel zur Benutzung der Audio-Alarme hinzugefügt
02/08/2016	1.03	Kapitel zur Benutzung des Telefonbuchs und der Gruppen für das Senden der Alarmmeldungen hinzugefügt
14/12/2016	1.04	Support für RTU Z-UMTS und Z-LOGGER3 hinzugefügt  Kapitel gestrichen und Abschnitt Application Note hinzugefügt

<b>1. VORBEREITENDE INFORMATIONEN ZU SEAL .....</b>	<b>6</b>
1.1. EVENTS – LOGIK – AKTIONEN .....	6
1.2. EVENTS .....	7
1.3. LOGIK .....	8
1.3.1. SEC: SIMPLE EVENT CONTROL .....	9
1.4. AKTIONEN .....	11
<b>2. ERSTELLUNG UND PROGRAMMIERUNG DES ERSTEN SEAL-PROJEKTS .....</b>	<b>13</b>
<b>3. DIENSTE VON WEB-SERVER UND MODBUS-TCP-IP-SERVER .....</b>	<b>20</b>
3.1. AKTIVIERUNG VON WEB-SERVER UND MODBUS-TCP-IP-SERVER .....	20
3.2. ZUGANG ZUM WEB-SERVER UND ZUM MODBUS-TCP-IP-SERVER .....	21
3.2.1. KLASSISCHE SIM MIT DATENVERKEHR .....	22
3.2.2. KLASSISCHE SIM MIT DATENVERKEHR UND OFFENEN EINGANGSPORTS .....	22
3.2.3. SIM MIT PRIVATER APN .....	23

**ACHTUNG!**

*Wenden Sie sich hinsichtlich der Kosten der Dienste GSM und GPRS an Ihren Telefonanbieter; die Kosten für das Senden der Logs und der SMSs sollten vor der Konfigurierung und Installation von Z-GPRS3, Z-UMTS und Z-LOGGER3 geklärt werden.*

*Die Benutzung von Z-GPRS3 und Z-UMTS in der Modalität Daten-Roaming (zum Beispiel im Ausland mit deutscher SIM) kann zu unerwarteten Kosten führen. Wenden Sie sich für weitergehende Informationen an Ihren Telefonanbieter.*

**IN KEINEM FALL KÖNNEN SENECA ODER DIE LIEFERANTEN VON SENECA FÜR EVENTUELLE VERLUSTE VON EINGEHENDEN DATEN ODER PROFITEN VERANTWORTLICH GEMACHT WERDEN, ODER FÜR INDIREKTE SCHÄDEN, ZUFÄLLIGE SCHÄDEN ODER FOLGESCHÄDEN, FÜR SCHÄDEN (EINSCHLISSLICH NACHLÄSSIGKEIT) IM ZUSAMMENHANG MIT DER BENUTZUNG ODER DER UNMÖGLICHKEIT DER BENUTZUNG VON Z-GPRS3, Z-UMTS und Z-LOGGER3, AUCH WENN SENECA ÜBER DIE MÖGLICHKEIT DIESER SCHÄDEN INFORMIERT WORDEN IST.**

**SENECA, DIE TOCHTER- UND PARTNERGESELLSCHAFTEN DER GRUPPE SOWIE DIE HÄNDLER VON SENECA GARANTIEREN NICHT, DASS DIE FUNKTIONEN GENAU DIE ERWARTUNGEN ERFÜLLEN, UND DASS Z-GPRS3, Z-UMTS und Z-LOGGER3 SOWIE DEREN FIRMWARE FREI VON FEHLERN IST ODER DASS SIE UNTERBRECHUNGSFREI FUNKTIONIEREN.**

**SENECA HAT BEI DER ERSTELLUNG DES VORLIEGENDEN HANDBUCHES DIE GRÖSSTMÖGLICHE SORGFALT AUFGEWENDET; DENNOCH IST ES MÖGLICH, DASS FEHLER ODER AUSLASSUNGEN DARIN ENTHALTEN SIND. SENECA SRL BEHÄLT SICH DAS RECHT VOR, IM FALL VON FEHLERN ODER ÄNDERUNGEN DER EIGENSCHAFTEN DES PRODUKTS TEILE DES FOLGENDEN HANDBUCHES OHNE VORANKÜNDIGUNG ZU ÄNDERN.**

**ACHTUNG!**

*-Wenden Sie sich hinsichtlich der Kosten für die Dienste GSM und GPRS an Ihren Telefonanbieter, vor allem, wenn Z-GPRS3 oder Z-UMTS mit SIM im Ausland verwendet werden (internationales Roaming).*

*-Die Telefonkosten sollten vor der Konfigurierung von Z-GPRS3 und Z-UMTS geklärt werden.*

*-Die Kosten für eine SMS werden vom Telefonanbieter festgelegt.*

*-Die Kosten für Empfang und senden via GPRS mit den gesendeten/empfangenen Kbyte in Zusammenhang stehen, mit einer monatlichen Höchstmenge, die in einem Paket enthalten ist, oder mit der Zeit der GPRS-Verbindung; wenden Sie sich für weitergehende Informationen an den Telefonanbieter.*

*-Überprüfen Sie die Zahl der Sendungen von Daten via GPRS und via SMS vor der Inbetriebnahme von Z-GPRS3 und Z-UMTS.*

***Wir erinnern daran, dass die Mobilfunkanbieter jede Transaktion 2G/3G als Datenverkehr behandeln, sowie auch die gesamte Kommunikation, die die Übertragung der Datei gestattet (berechnet wird also auch der Overhead der Datenübertragung, die Anzahl der Verbindungsversuche usw.) und nicht nur die Größe der Datei.***

## 1. VORBEREITENDE INFORMATIONEN ZU SEAL

Dieses Handbuch hat den Zweck, in die Programmierung von SEAL für die RTU von Seneca einzuführen.

Weitergehende Informationen zu SEAL sind in der Online-Hilfe zu SEAL vorhanden, weitergehende Informationen zu Z-GPRS, Z-UMTS und Z-LOGGER3 sind im Benutzerhandbuch vorhanden.

Jeder Funktionsblock verfügt über eine Help-Taste, die direkt zur Erklärung des Blocks führt:

**1.6.8.1. Control Block SEC: Simple Event Logic Control**

Input Event 1	<input type="checkbox"/>	SEC1	<input type="checkbox"/>	Output
Input Event 2	<input type="checkbox"/>	SIMPLE	<input type="checkbox"/>	Output
Input Event 3	<input type="checkbox"/>	EVENT	<input type="checkbox"/>	
Input Event 4	<input type="checkbox"/>		<input type="checkbox"/>	

This block is a simplified logic state-machine for the SEAL program. Up to 32 Event Control blocks can be configured in the Graph Panel.

**SEC1 Event Control**

Simple Event Control

Input Logic: AND (4)

Send: SMS (Wait for Notification Acknowledge)

To Profile: Operator (Append to Info Events)

Block Info: SIMPLE EVENT

• **Input Logic:** up to 4 input events can be configured in AND/OR Logic

### 1.1. EVENTS – LOGIK – AKTIONEN

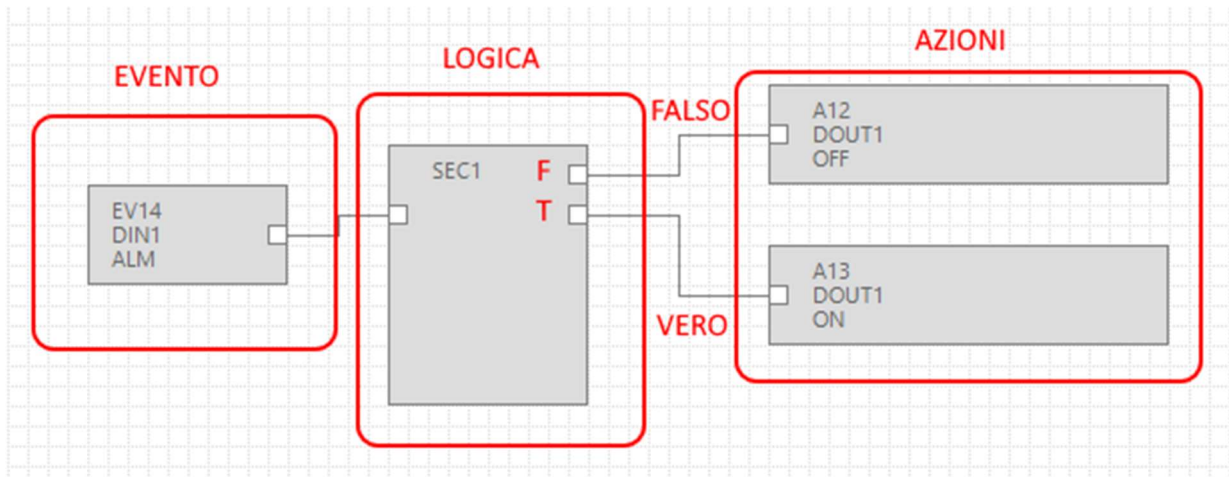
SEAL (SEneca Advanced Language) ist eine Software für die Konfigurierung und Programmierung für die RTU von Seneca.

SEAL gestattet die Erstellung einfacher Programme unter Nutzung eines Ansatzes vom Typ:

EVENTS – LOGIK – AKTIONEN

Die Events, Logiken und Aktionen werden **Funktionsblöcke** genannt und sie sind miteinander mit Verbindungen (Leitern) verbunden und bilden **Ketten**.

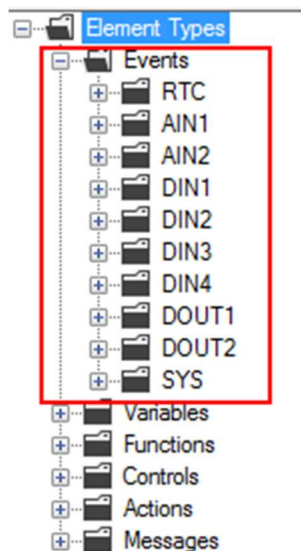
Es folgt ein Beispiel einer Kette:



Das Event DIN1 ALM (Alarm am Digitaleingang 1) ist mit der Steuerungslogik SEC (Simple Event Control) verbunden, wenn das Event falsch ist (also kein Alarm), wird die Aktion „FALSCH“ ausgeführt (der Relaisausgang 1 wird nicht erregt); wenn das Event wahr ist (also Alarm aktiv), wird die Aktion „WAHR“ ausgeführt (Relaisausgang 1 erregt):

## 1.2. EVENTS

Die **Events** in SEAL befinden sich im Ordner **Events**:

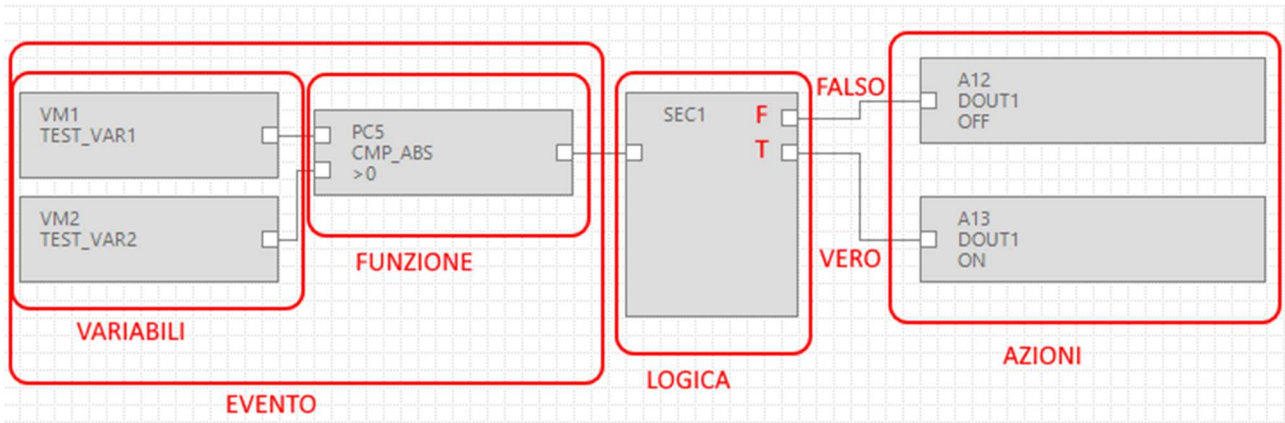


Zu den Events gehören: Die Alarme an den analogen Eingängen (AIN), an den digitalen Eingängen (DIN) usw.

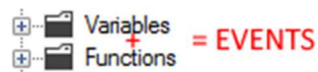
Ein Events ist nur vom bool'sche Typ (digital) und kann wahr (true) oder falsch (false) sein.

Außer den vordefinierten Events ist es möglich, eine **Variable** mit einer **Funktion** zu verbinden und ein neues **Event** zu erstellen.

In diesem Beispiel werden zwei Modbus-Variablen "Test\_VAR1" und "Test\_VAR2" definiert, wenn "TEST\_VAR1" > "TEST\_VAR2", wird die Aktion „WAHR“ ausgeführt, das heißt, der Relaisausgang 1 wird erregt, anderenfalls wird die Aktion „FALSCH“ ausgeführt, das heißt, der Relaisausgang 1 wird aberregt.



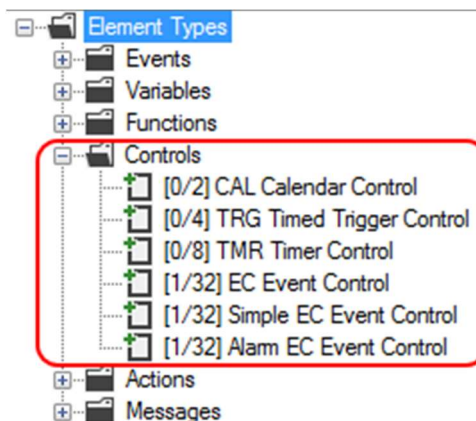
Die Variablen, die mit einer Funktion verbunden sind, bilden also ein **Event**.



### 1.3. LOGIK

Die Logik gestattet es auszuwählen, welche Aktionen ausgeführt werden müssen, wenn das Event wahr oder falsch ist.

Die Funktionsblöcke, die die Logik ausführen, sind im Ordner „Controls“ enthalten:



Die Nummer in eckigen Klammern [xx/yy] hat die folgende Bedeutung:

xx = Anzahl der Funktionsblöcke vom angegebenen Typ, die im aktuellen Projekt verwendet werden

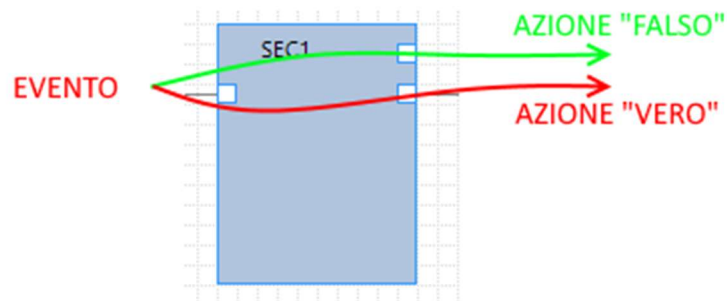


yy = max. Anzahl der Funktionsblöcke vom angegebenen Typ, die verwendet werden können

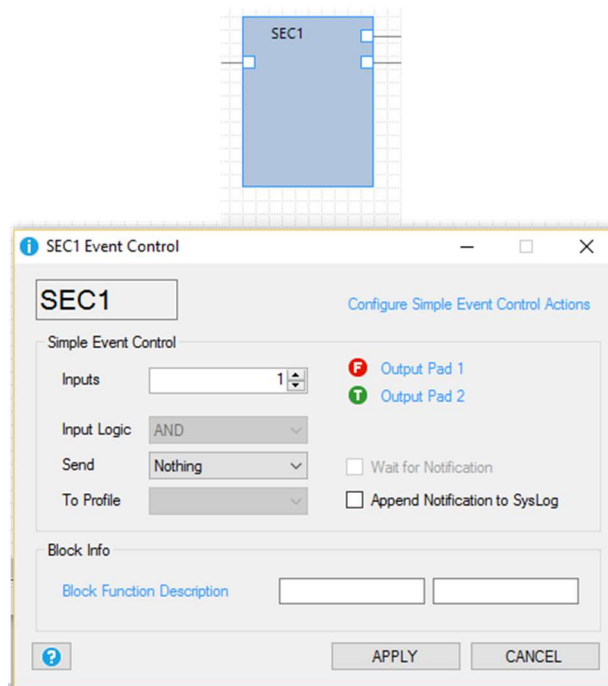
### 1.3.1. SEC: SIMPLE EVENT CONTROL

Der nützlichste Funktionsblock ist SEC (Simple EC Event Control); wie zu sehen ist, können max. 32 SEC verwendet werden.

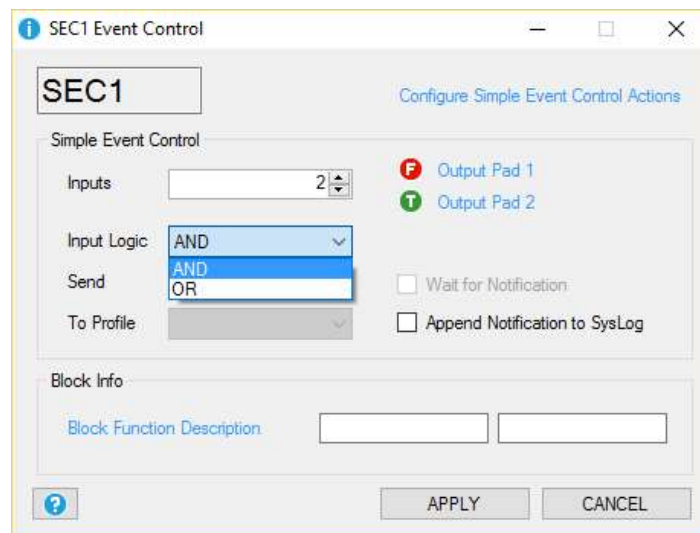
Der SEC führt eine Trennung zwischen dem Event „Falsch“ und dem Event „WAHR“ aus:



Bei einem Doppelklick auf den Funktionsblock erscheinen die Einstellungen des Simple Event Control:

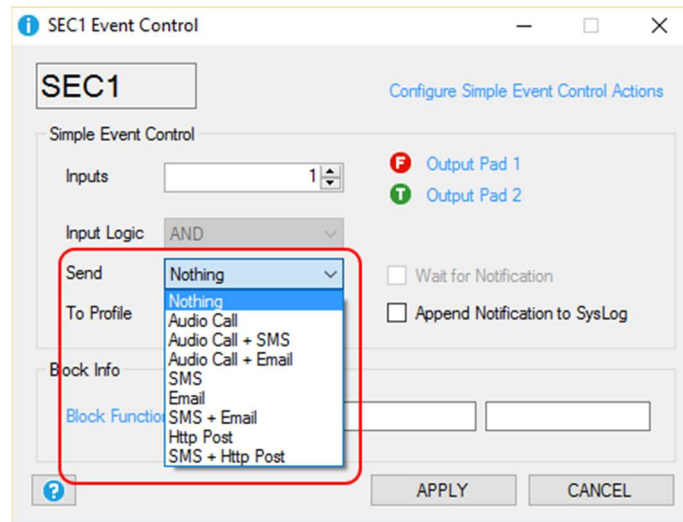


SEC kann 1 bis 4 Eingänge (Inputs) verarbeiten, im Fall von > 1 ist es möglich, den Input Logic auszuwählen:



Das heißt, die Eingangs-Events müssen alle wahr sein (AND), oder es reicht, dass nur einer wahr ist (OR).

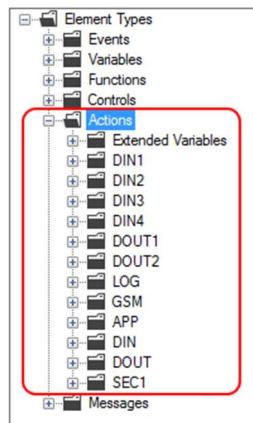
Mit „Send“ ist es möglich, das Senden von Textmeldungen oder Audio-Anrufen zuzuordnen:



Aktivieren Sie die Checkbox „Appen Notification to Syslog“, wenn die Events im **Event-Logger** erscheinen sollen.

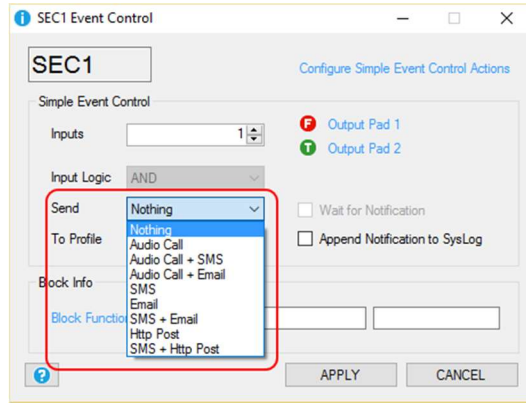
## 1.4. AKTIONEN

Die Aktionen sind in SEAL im Ordner Actions verfügbar:



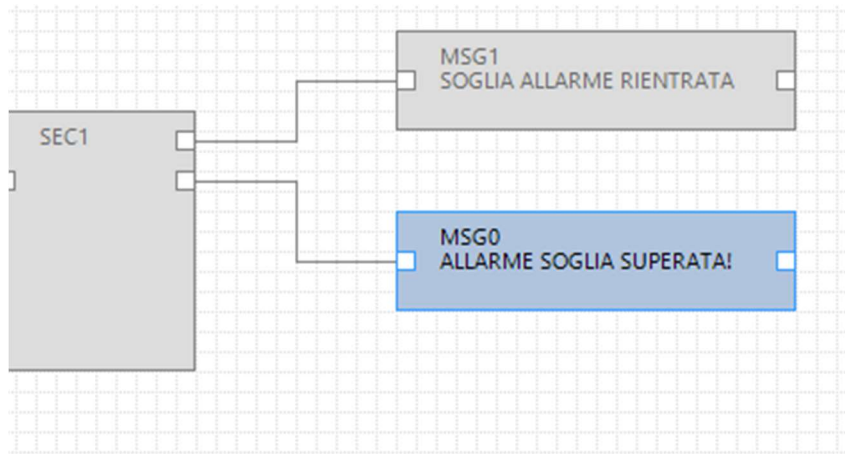
Es sind Aktionen an Zählwerken, an digitalen Eingängen, an Logs usw. verfügbar.

Die Aktion, eine Textnachricht (oder einen Audio-Anruf) zu senden, wird als eine besondere Aktion angesehen, und sie muss mit Nachrichten (Messages) und mit der Konfigurierung des Logikblock gesteuert werden:

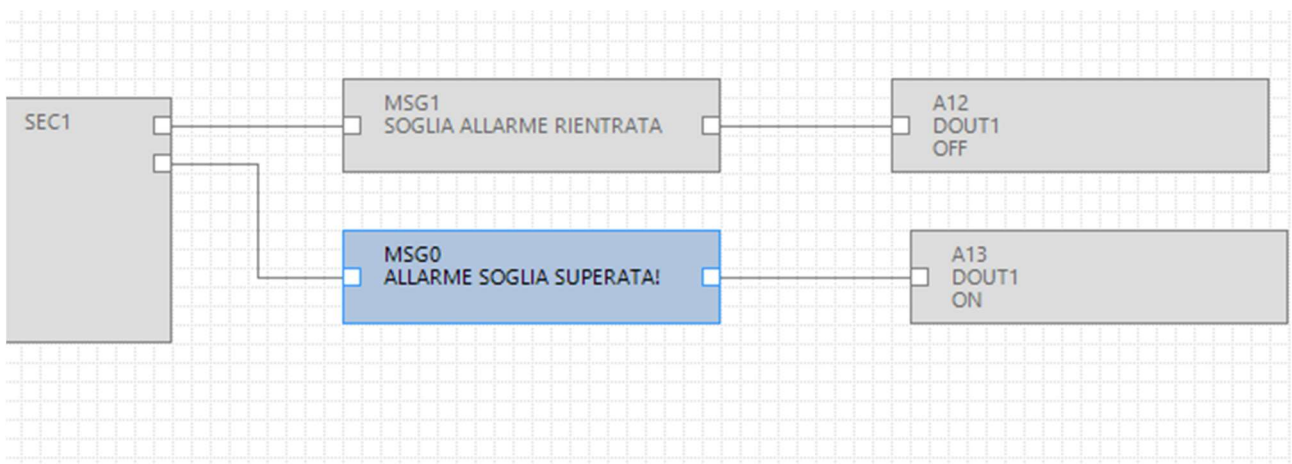


Die Nachrichten können ihrerseits mit der eigentlichen Aktion verbunden werden (die vorhanden oder nicht vorhanden sein kann).

Zum Beispiel kann das Senden einer SMS bei Überschreitung einer Schwelle am analogen Eingang 1 wie folgt erfolgen:



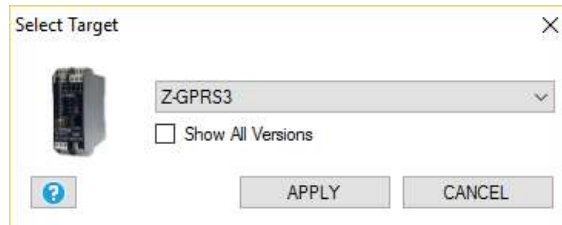
Es ist auch möglich, auf die Nachrichten den Befehl des digitalen Ausgangs 1 folgen zu lassen:




## 2. ERSTELLUNG UND PROGRAMMIERUNG DES ERSTEN SEAL-PROJEKTS

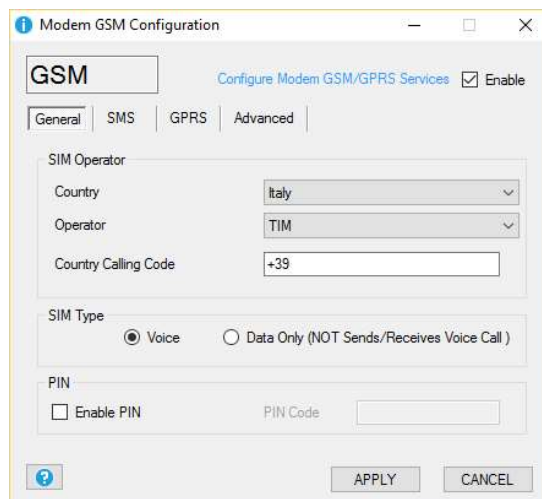
Als Beispiel erstellen wir Schritt für Schritt ein einfaches Projekt für Z-GPRS3, dessen Zweck es ist, eine Alarm-SMS zu senden, wenn sich der Status des digitalen Eingangs 1 (DIN1) ändert.


Zuerst wird das Target konfiguriert, indem zum Beispiel „Z-GPRS3“ ausgewählt wird:



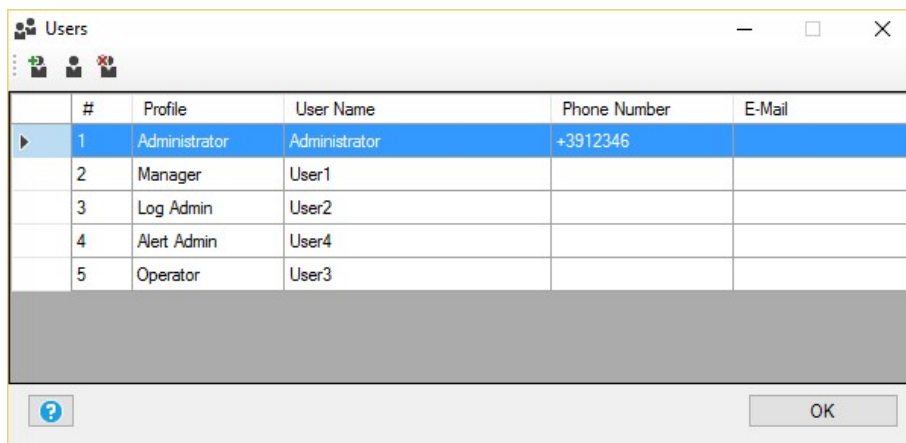
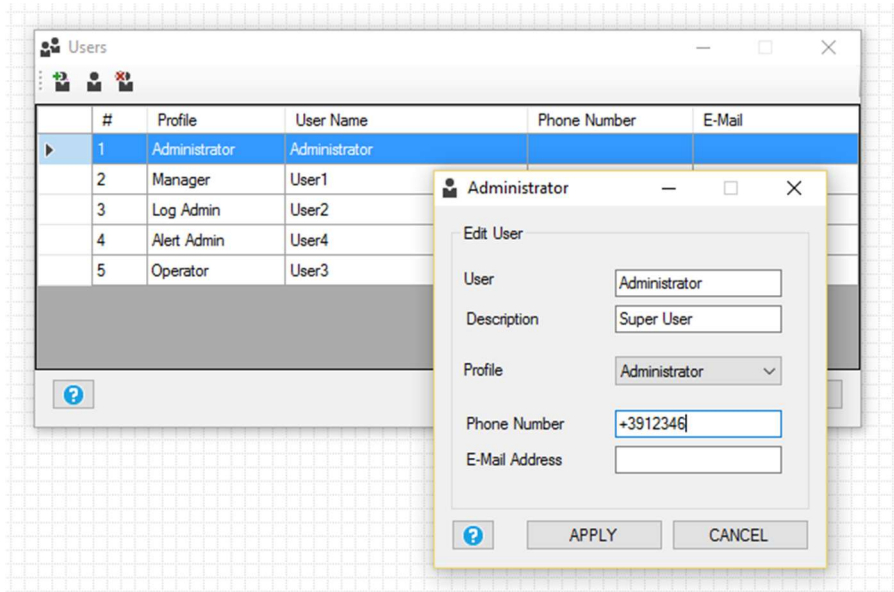
Dann folgt die Konfigurierung der SIM-Karte, die die Alarm-SMS beim anklicken des Symbols  GSM sendet.

Jetzt wählen wir die Parameter unseres Telefonanbieters aus:



Jetzt tragen wird die Telefonnummer, an die die SMSs beim Anklicken des Symbols  gesendet werden sollen, in das Telefonbuch ein:

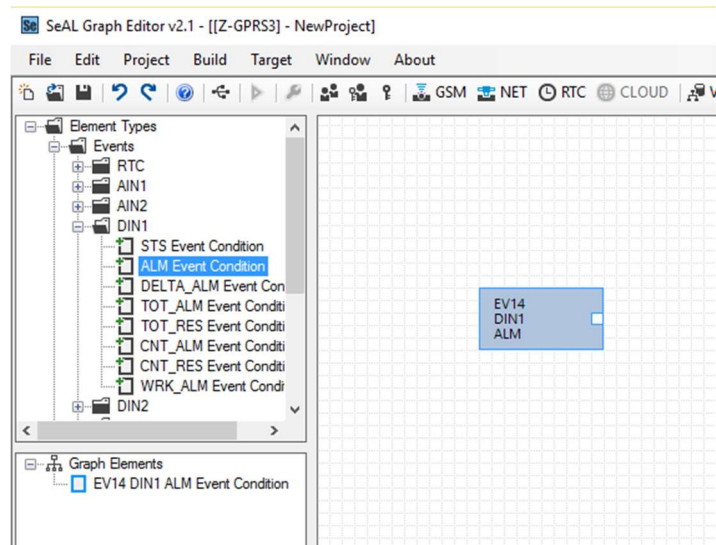
Wir führen einen Doppelklick auf „Administrator“ aus und geben die Telefonnummer +3912346 ein:



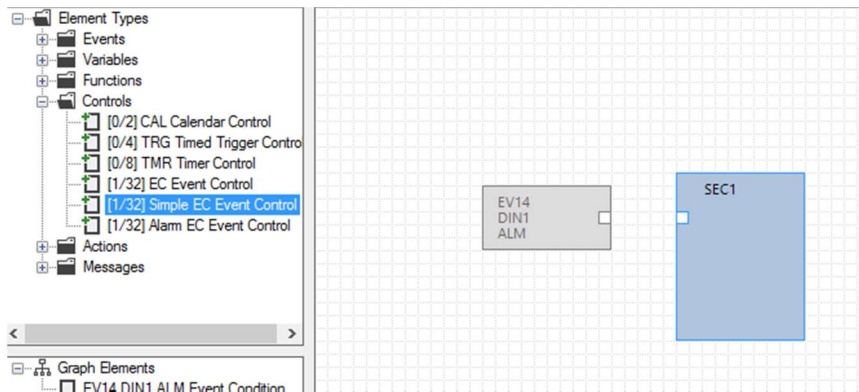
Damit ist die Konfigurierung abgeschlossen.

Jetzt muss die Alarmkette des digitalen Eingangs 1 programmiert werden.

Zuerst muss das Alarm-Event auf das Projekt gezogen werden, das heißt, der Funktionsblock DIN1 ALM:

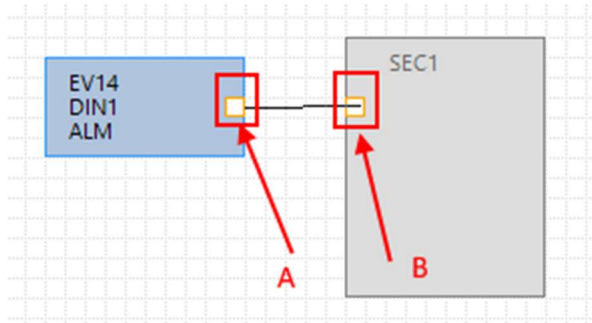


Wir ziehen den Funktionsblock der Logik Simple Event Control:



Dann verbinden wir das Event wie folgt mit der Logik:

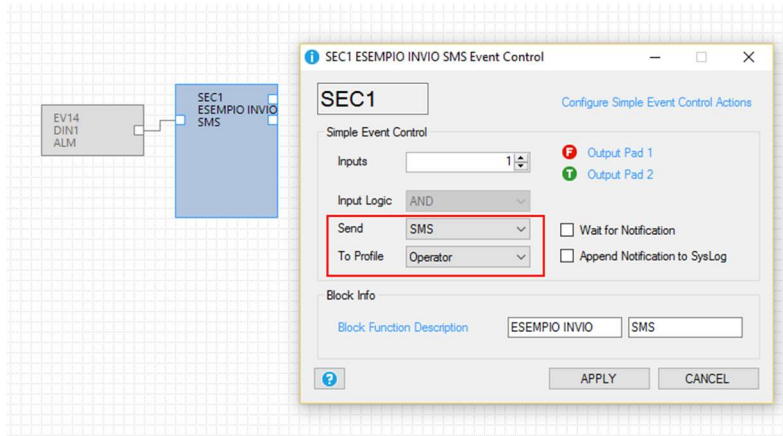
Mit der linken Maustaste auf den Punkt A klicken, die Taste gedrückt halten und am Punkt B loslassen:



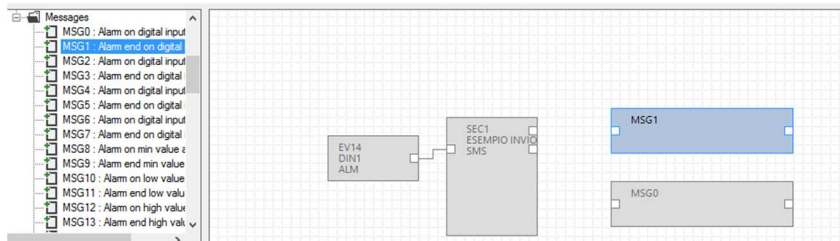
Nun führen wir einen Doppelklick auf den Funktionsblock der Logik „SEC1“ aus:

Wir stellen das Senden „SMS“ im Operator-Profil auf „Send“.

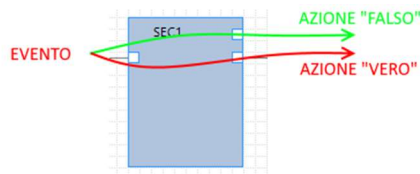
Da unsere Benutzer Administrator ist, ist er in allen Profilen vorhanden und erhält daher die SMSs:



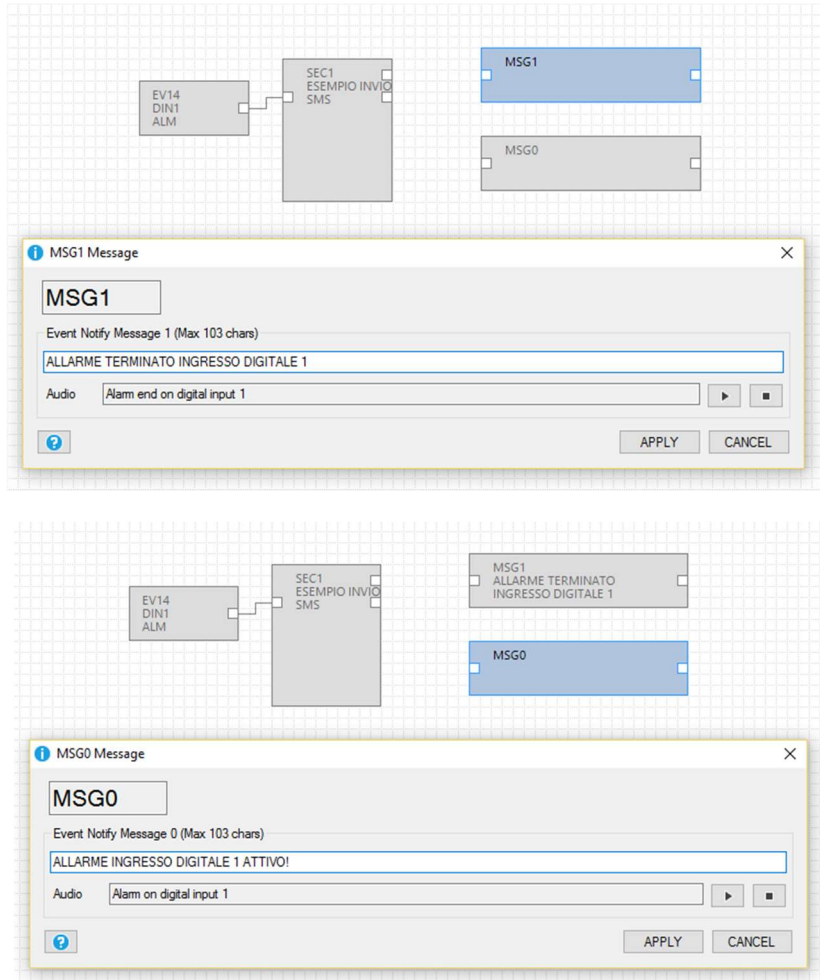
Nun fügen wir die Aktionen hinzu, die in diesem Fall Textnachrichten sind; wir fügen die ersten beiden Messages hinzu: MSG0 und MSG1:



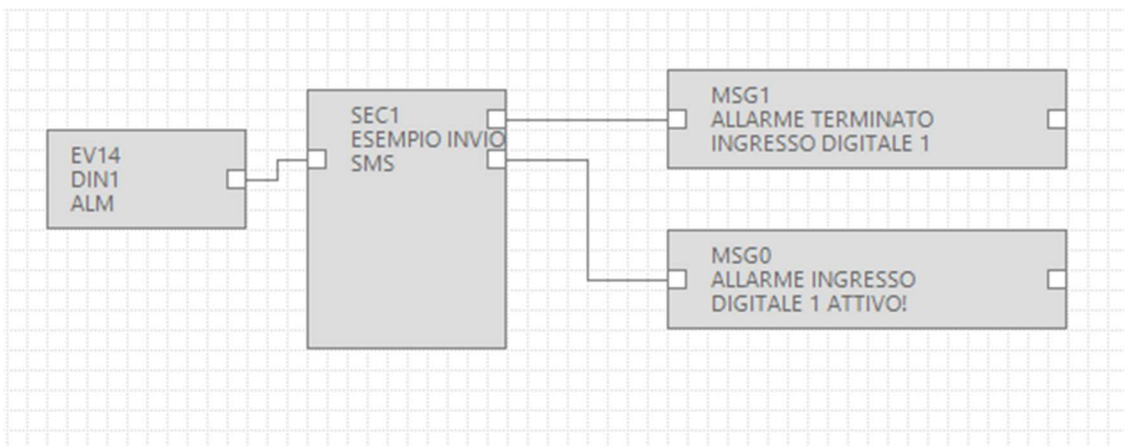
Wir führen einen Doppelklick auf die Funktionsblöcke MSG0 und MSG1 aus, um den Default-Text unserer Nachricht zu ändern. SEC1 hat im PAD 1 den Fall, dass das Event falsch ist, und im PAD 2 den Fall, dass das Event wahr ist:





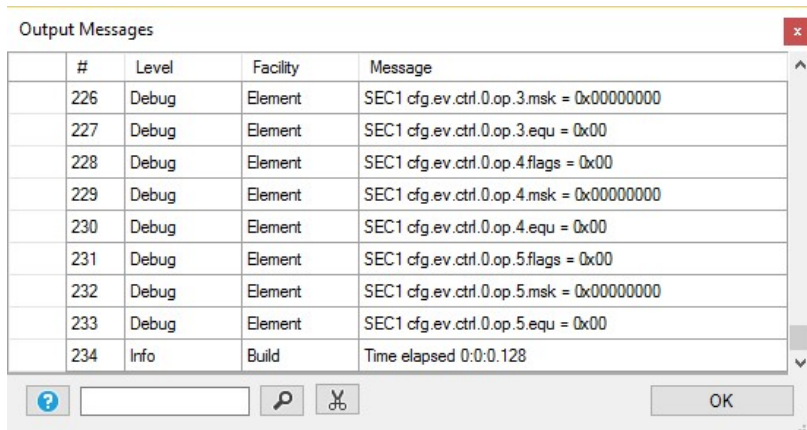
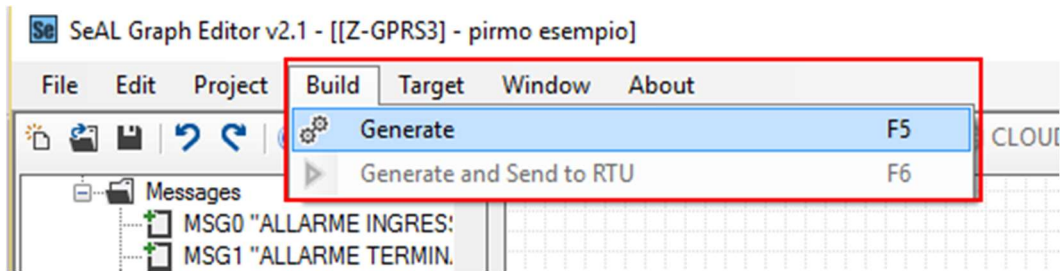


So erzielen wir die finale Kette durch Verbindung des Funktionsblocks der Logik mit den beiden Nachrichten:



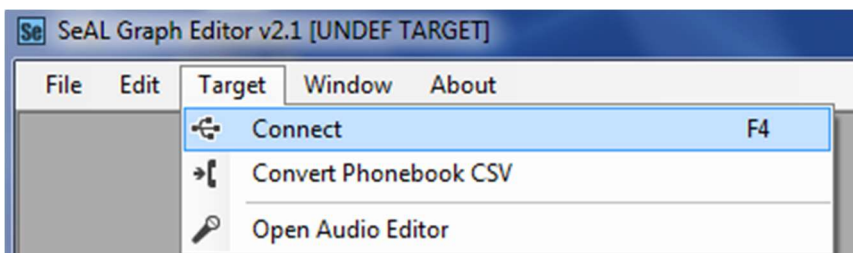
Sowohl die Konfiguration, als auch das Projekt sind abgeschlossen; nun muss das Projekt kompiliert und an die RTU gesendet werden.

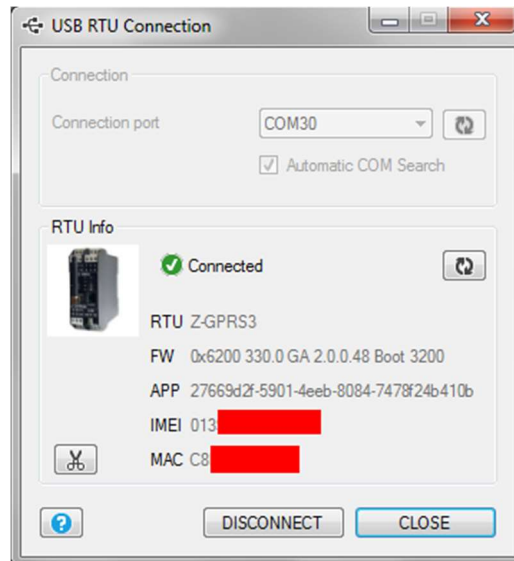
Bei Kompilieren stellen wir sicher, dass keine Fehler vorhanden sind; zum Kompilieren verwenden wir das Menü Build:



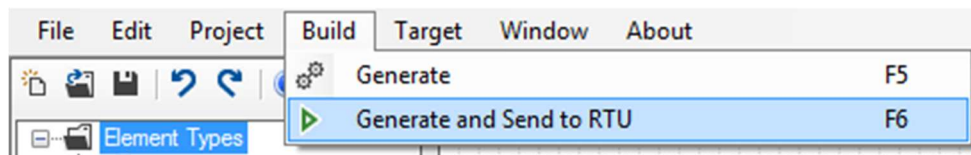
Die Kompilierung hat keine Fehler angezeigt.

Nun können wir das Projekt an die RTU senden, indem wir das USB-Kabel anschließen und die Verbindung mit „Connect“ herstellen:





jetzt senden wir das Projekt mit -> „Generate and Send to RTU“:



An diesem Punkt müssen die Klemmen 15 und 13 kurzgeschlossen werden, um die Alarm-SMS zu senden:

„ALARM DIGITALER EINGANG 1 AKTIV“

an die konfigurierte Nummer.

Beim Aufheben des Kurzschlusses der Klemmen 15 und 13 wird eine andere Alarm-SMS gesendet:

„ALARM BEENDET DIGITALER EINGANG 1“

### 3. DIENSTE VON WEB-SERVER UND MODBUS-TCP-IP-SERVER

Die RTU weisen zwei dienst-Server auf, den Web-Server und den Modbus-TCP-IP-Server:

-WEB SERVER: Gestattet das Anzeigen der Werte der Variablen des Loggers in Echtzeit durch einfache Verwendung eines Web-Browsers auf einem PC, Smartphone oder Tablet. Es ist möglich, den Status der RTU und eventuelle Fehler anzuzeigen, die auf SD-Karte vorhandenen Log-Dateien herunterzuladen usw.


-MODBUS-TCP-IP-SERVER: gestattet den Zugriff in Echtzeit auf die Werte der Variablen, die über SCADA oder Modbus-TCP-IP-Client registriert werden.

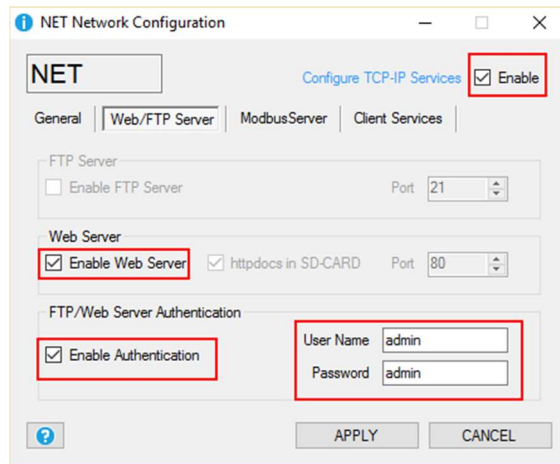
nehmen Sie für die Adressen Mobus-TCP-IP der internen Variablen der RTU auf das Benutzerhandbuch Bezug.

Die Adressen Mobus-TCP-IP der erweiterten Variablen der RTU werden bei der Kompilierung errechnet; nehmen Sie auf den Output der Kompilierung Bezug:

#	Level	Facility	Message
251	Debug	Element	APP cfg.app.status_notify_flags.1 = 0x00000000
252	Debug	Element	APP cfg.app.status_notify_flags.2 = 0x00000000
253	Debug	Element	APP cfg.app.status_notify_flags.3 = 0x00000000
254	Debug	Element	APP cfg.app.debug_enable = 0x00000000
255	Debug	Element	PRT2 1:INVERTER_A 31793:Current String1 var.w.0 = 0x0000
256	Debug	Element	PRT2 1:INVERTER_A 31793:Current String1 var.w.1 = 0x0000
257	Debug	Element	PRT2 1:INVERTER_A 31795:Current String2 var.w.2 = 0x0000
258	Debug	Element	PRT2 1:INVERTER_A 31795:Current String2 var.w.3 = 0x0000
259	Debug	Element	PRT2 1:INVERTER_A 31797:Current String3 var.w.4 = 0x0000
260	Debug	Element	PRT2 1:INVERTER_A 31797:Current String3 var.w.5 = 0x0000
261	Notice	Build	Slave Address Current String1:PRT2 1:INVERTER_A 31793:Current String1 Mapped to 41003
262	Notice	Build	Slave Address Current String2:PRT2 1:INVERTER_A 31795:Current String2 Mapped to 41005
263	Notice	Build	Slave Address Current String3:PRT2 1:INVERTER_A 31797:Current String3 Mapped to 41007
264	Info	Build	Time elapsed 0:0:0.114

#### 3.1. AKTIVIERUNG VON WEB-SERVER UND MODBUS-TCP-IP-SERVER

Klicken Sie zur Aktivierung des Web-Servers und des Modbus-TCP-IP-Servers auf das Symbol  und geben Sie die Parameter für den Zugang zum Web-Server ein:



Geben Sie für den Zugang zum Web-Server mit Werkseinstellung ein:

<http://IP/maintenance/index.html>

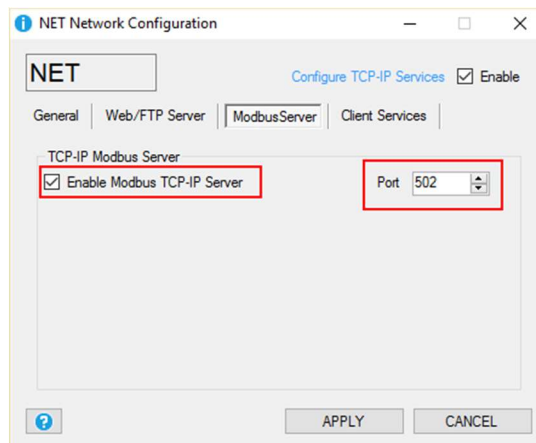
wobei IP die aktuelle IP-Adresse der RTU ist (Default 192.168.90.101).

Geben Sie für den Zugang zum personalisierten Web-Server auf SD-Karte über einen Browser ein:

<http://IP/index.html>

wobei IP die aktuelle IP-Adresse der RTU ist (Default 192.168.90.101).

Zur Aktivierung des Modbus-TCP-IP-Servers:



Das Protokoll Modbus TCP-IP verwendet per Default den Port 502; nehmen Sie zur Einstellung der richtigen Port-Nummer auf Ihr SCADA Bezug.

### 3.2. ZUGANG ZUM WEB-SERVER UND ZUM MODBUS-TCP-IP-SERVER

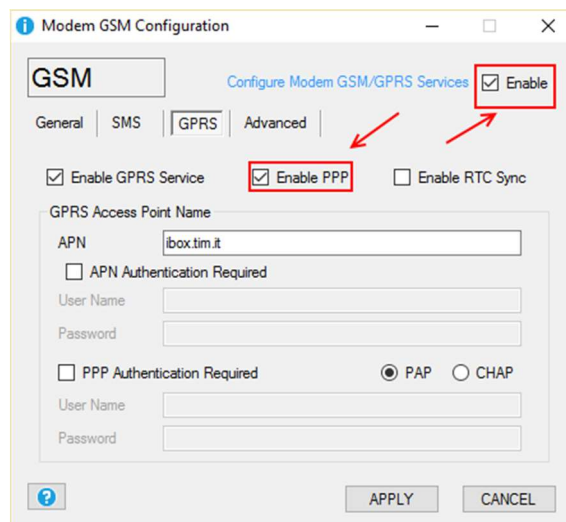
Auf folgende Weise ist es möglich, remote auf den Modbus-TCP-IP-Server und auf den Web-Server zuzugreifen:

**-VERBINDUNG ÜBER ADSL / Xdsl**

Für diesen Verbindungstyp muss die Ethernet-Verbindung verwendet werden und das Modem ADSL / xDSL muss so konfiguriert werden, dass die Ports der Server-Dienste für das Internet sichtbar sind (je nach Modell des Modems Port Forwarding oder Virtual Server)

**-VERBINDUNG ÜBER MODEM GSM/GPRS**

Für diesen Verbindungstyp muss die Verbindung GSM/GPRS PPP verwendet werden:



Es ist daher nicht möglich, E-Mails über Server EMAIL SMTP zu senden, die eine SSL-Verbindung erforderlich machen.

Für die Remote-Verbindung sind verschiedene SIM-Typen verfügbar:

**3.2.1. KLASSISCHE SIM MIT DATENVERKEHR**

Dies ist die gewöhnlichste SIM, die normalerweise in Smartphones verwendet wird. Diese SIM kann typischerweise nicht direkt erreicht werden, da alle Eingangs-Ports gefiltert werden. Es ist daher erforderlich, sich an den eigenen Telefonanbieter zu wenden, um „die Eingangs-Ports der SIM öffnen zu lassen“. Mit diesem SIM-Typ ist es nur möglich, die Log-Dateien und die Alarmer zu senden, es ist jedoch nicht möglich, auf den Modbus-TCP-IP-Server oder den WEB-Server zuzugreifen.

**3.2.2. KLASSISCHE SIM MIT DATENVERKEHR UND OFFENEN EINGANGSPORTS**

Diese SIM ist eine klassische SIM für den Datenverkehr, deren Eingangs-Ports vom Telefonanbieter geöffnet worden sind. Mit diesem SIM-Typ ist es möglich, auf den Modbus-TCP-IP-Server und auf den Web-Server zuzugreifen, aber bei jeder Verbindung ändert sich die IP-Adresse (dynamisch).

Zur Lösung des Problems kann die RTU auf die SMS (deren Text „NET“ ist) durch Senden der aktuellen IP-Adresse antworten. Es besteht auch die Möglichkeit, das Seneca-Produkt OPC Server für die Verbindung mit dem Modbus-TCP-IP-Server zu verwenden, so dass ein FTP-Server verwendet wird, der immer die letzte gültige IP-Adresse behält.

### ***3.2.3. SIM MIT PRIVATER APN***

Diese SIM gestattet den Zugriff auf ein privates Netz (außerhalb des Internets) über eine feste und statische IP-Adresse. Es ist daher möglich, direkt auf den Web-Server und den Modbus-TCP-IP-Server zuzugreifen.