

Kontrolka ActiveX Roger_RemoteMon.ocx wer. 4.2.1

Rejestracja kontrolki:

Kontrolkę należy zarejestrować w systemie: z linii komend poleceniem „RegSvr32 Roger_RemoteMon.ocx ”.

Import do środowiska programistycznego:

Następnie można ją zaimportować do dowolnego środowiska programistycznego obsługującego kontrolki ActiveX np.:

- Delphi, C++Builder (z menu Component/Import ActiveX Control...)
- MS Visual Studio (dla VS.NET - menu Tools/Customize Toolbox/COM Components).

Specyfikacja interfejsu API kontrolki:

Własności:

- property _PRMasterHostIP: WideString - IP komputera na którym znajduje się serwer PRMaster
- property _PRMasterHostPort: SYSUINT - Port na komputerze z serwerem
- property _Login: WideString - login użytkownika
- property _Password: WideString - hasło użytkownika
- property _Connected: WordBool - ustawienie na true powoduje połączenie z serwerem, false - rozłącza
- property _ShowMessage: WordBool - true jeśli mają się pokazywać okienka komunikatów
- property _ShowErrorMessage: WordBool - true jeśli mają się pokazywać okienka z komunikatami błędów
- property _GET_ReaderNamesXML: WideString - pobiera string w formacie XML zawierający listę czytników. Odpowiada on zawartości wyświetlanej w oknie wyboru kontrolera (wszystkie podmenu menu: Commands/Controller Commands/...)

Struktura XML (z przykładowymi danymi):

```
<tables>
  <table>
    <row>
      <col>Podsystem
      A</col>
      <col>Domyślna</col>
    >
    <col>PR3 01</col>
  </row>
</table>
```

```

        <col>Podsystem
    A</col>
        <col> Domyślna</col>
        <col>PR401</col>
    </row>
</table>
</tables>

```

- property `_GET_ZonesXML`: `WideString` - pobiera string w formacie XML zawierający listę stref. Odpowiada on zawartości wyświetlanej w oknie wyboru strefy (wszystkie podmenu menu: *Commands/Zones Commands/...*)

Struktura XML (z przykładowymi danymi):

```

<tables>
  <table>
    <row>
      <col>Domyślna</col>
    </row>
  </table>
</tables>

```

- property `_GET_UsersXML`: `WideString` - pobiera string w formacie XML zawierający listę użytkowników. Odpowiada on zawartości wyświetlanej w oknie użytkowników (menu: *View/View Users*)

Struktura XML (z przykładowymi danymi):

```

<tables>
  <table>
    <row>
      <col>True</col>
      <col>0</col>
      <col>Master Arnold</col>
      <col>False</col>
      <col>MASTER</col>
      <col></col>
      <col>No group</col>
    </row>
    <row>
      <col>True</col>
      <col>2</col>
      <col>Orłowska Janina</col>
      <col>False</col>
      <col>SWITCHER Full (1..49)</col>
      <col></col>
      <col>No group</col>
    </row>
    <row>
      <col>True</col>
      <col>1</col>
      <col>Woźniak Jarek</col>
      <col>False</col>
      <col>SWITCHER Full (1..49)</col>
      <col></col>
      <col>No group</col>
    </row>
  </table>
</tables>

```

- property `_GET_GroupsXML`: `WideString` - *pobiera string w formacie XML zawierający listę grup. Odpowiada on zawartości wyświetlanej w oknie grup (menu: View/View Groups)*

Struktura XML (z przykładowymi danymi):

```
<tables>
  <table>
    <row>
      <col>No group (No group)</col>
      <col>0</col>
    </row>
    <row>
      <col>New group (1)</col>
      <col>1</col>
    </row>
    <row>
      <col>New group (2)</col>
      <col>2</col>
    </row>
  </table>
</tables>
```

- property `_GET_SubsystemsXML`: `WideString` - *pobiera string w formacie XML zawierający listę podsystemów. Odpowiada on zawartości wyświetlanej w oknie podsystemów (menu: View/View Subsystems)*

Struktura XML (z przykładowymi danymi):

```
<tables>
  <table>
    <row>
      <col>True</col>
      <col>0</col>
      <col>Podsystem A</col>
      <col>1</col>
    </row>
  </table>
</tables>
```

- property `_GET_UserXML(const userID: SYSUINT)`: `WideString` - *pobiera string w formacie XML zawierający dane użytkownika podsystemów. Odpowiada on zawartości wyświetlanej w oknie edycji użytkownika (menu: Tools/Quick User Update/[Edit User])*

Struktura XML (z przykładowymi danymi):

```
<user>
  <userId>999</userId>
  <firstName>RYSZARD</firstName>
  <lastName>ZIÓŁKOWSKI</lastName>
  <userName>ZIÓŁKOWSKI RYSZARD</userName>
  <official>False</official>
  <active>True</active>
  <custom1 />
  <custom2 />
  <custom3 />
  <custom4 />
  <evidence>34378</evidence>
  <status>NORMAL</status>
```

```

<transponderCode>0064446680005</transponderCode>
<keyboardCode>34378</keyboardCode>
<photo />
<group enable="True" selName="[No group]">
  <name>[No group]</name>
  <name>grup1</name>
</group>
<activityPeriodOn>True</activityPeriodOn>
<activitySince>2001/1/31</activitySince>
<activityUntil>2004/12/31</activityUntil>
</user>

```

- property `_GET_UsersLastLoginXML`: WideString - pobiera string w formacie XML zawierający dane o ostatnich logowaniach użytkowników na czytnikach. Odpowiada on zawartości wyświetlanej w oknie - tabeli ostatnich logowań użytkowników (menu: View/Find User)

Struktura XML (z przykładowymi danymi):

```

<tables>
  <table>
    <row>
      <col>0</col>
      <col>MASTER ALFRED</col>
      <col>Bez grupy</col>
      <col>2005-12-20</col>
      <col>13:43:40</col>
      <col>PR302 LCD/128.2/0002/348_T1</col>
      <col>32</col>
    </row>
    <row>
      <col>1</col>
      <col>ŚLUSARSKI BOGDAN</col>
      <col>Bez grupy</col>
      <col>2005-12-20</col>
      <col>14:24:20</col>
      <col>PR302 LCD/128.2/0002/348_T0</col>
      <col>32</col>
    </row>
  </table>
</tables>

```

- property `_GET_ControllersStatesXML`: WideString - pobiera string w formacie XML zawierający dane o statusie kontrolerów. Odpowiada on zawartości wyświetlanej w oknie - tabeli stanów kontrolerów (menu: View/Controllers States)

Struktura XML (z przykładowymi danymi):

```

<tables>
  <table>
    <row>
      <col>PR302 LCD/128.2/0001/837_T1</col>
      <col>-</col>
      <col>-</col>
      <col>- / -</col>
      <col>- / - / - / -</col>
      <col>-</col>
      <col>Brak</col>
      <col>??</col>
      <col>-</col>
    </row>
  </table>
</tables>

```

```

        </row>
        <row>
            <col>PR302 LCD/128.2/0002/348_T1</col>
            <col>-</col>
            <col>-</col>
            <col>- / -</col>
            <col>- / - / - / -</col>
            <col>-</col>
            <col>Brak</col>
            <col>??</col>
            <col>-</col>
        </row>
    </table>
</tables>

```

- property _GET_PRMaster Version: WideString -*pobiera string zawierający informacje o wersji połączonego PRMastera.*

Metody:

(Poniższe metody odpowiadają poszczególnym metodom wywoływanym z menu.)

- procedure _Entire_AllControllersToON; - *odpowiada menu Commands/Entire System/All Controllers to ON.*
- procedure _Entire_AllControllersToOFF; - *odpowiada menu Commands/Entire System/All Controllers to OFF*
- procedure _Entire_SetNormalMode; - *odpowiada menu Commands/Entire System/Set Normal Mode*
- procedure _Entire_SetUnlockMode; - *odpowiada menu Commands/Entire System/Set Unlock Mode*
- procedure _Entire_SetConditionalUnlockMode; - *odpowiada menu Commands/Entire System/Set Conditional Unlock Mode*
- procedure _Entire_SetBarriedMode; - *odpowiada menu Commands/Entire System/Set Barried Mode*
- procedure _Command_Cancel All Alarms; - *odpowiada menu Commands/Cancel All Alarms*
- procedure _Command_SetSystemClocks; - *odpowiada menu Commands/Set System Clock*
- procedure _Controller_CancelControllerAlarm(idx: SYSUINT); - *odpowiada menu Commands/Controller Commands/Cancel Controller Alarm. (Parametr idx – indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- procedure _Controller_OpenDoor(idx: SYSUINT); - *odpowiada menu*

Commands/Controller Commands/Open Door. (Parametr idx - indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property _GET _ReaderNamesXML.)

- *procedure _Controller_SwitchToON(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Switch to ON. (Parametr idx - indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property _GET _ReaderNamesXML.)*
- *procedure _Controller_SwitchToOFF(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Switch to OFF. (Parametr idx - indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property _GET _ReaderNamesXML.)*
- *procedure _Controller_CheckDoorMode(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Check Door Mode. (Parametr idx – indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- *procedure _Controller_SetNormalMode(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Set Normal Mode. (Parametr idx – indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- *procedure _Controller_SetUnlockMode(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Set Unlock Mode. (Parametr idx – indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- *procedure _Controller_SetConditionalUnlockMode(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Set Conditional Unlock Mode. (Parametr idx - indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- *procedure _Controller_SetBarriedMode(idx: SYSUINT); - odpowiada menu Commands/Controller Commands/Set Barried Mode. (Parametr idx – indeks kontrolera (pierwszy = 0) w/g listy zwróconej przez property GET_ReaderNamesXML.)*
- *procedure _Zone_SwitchZoneToON(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Switch Zone to ON. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property _GET _ZonesXML.)*
- *procedure _Zone_SwitchZoneToOFF(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Switch Zone to OFF. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property _GET _ZonesXML.)*
- *procedure _Zone_SetNormalMode(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Set Normal Mode. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property*

_GET_ZonesXML.)

- *procedure _Zone_SetUnlockMode(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Set Unlock Mode. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property _GET_ZonesXML.)*
- *procedure _Zone_SetConditionalUnlockMode(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Set Conditional Unlock Mode. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property _GET_ZonesXML.)*
- *procedure _Zone_SetBarriedMode(idx: SYSUINT); - odpowiada menu Commands/Zones Commands/Set Barried Mode. (Parametr idx - indeks strefy (pierwszy = 0) w/g listy zwróconej przez property _GET_ZonesXML.)*
- *procedure _ClearEventList; - odpowiada menu View/Clear Event Window*
- *procedure _ClearAlarmList; - odpowiada menu View/Clear Alarm Window*
- *procedure _Tools_AddNewUserXML(userStatus: SYSUINT; quickUpdate: WordBool; xml: BSTR); - odpowiada menu Tools/Quick User Update/[Add New User]. (Parametr user Status -wartości: 0-MASTER, 16-SWITCHER Full, 32-SWITCHER Limited, 48-NORMAL; parametr quickUpdate- określa czy zmiany mają być przesłane do urzędzeń; parametr xml -patrz wyżej: _Tools GET UserXML)*
- *procedure _Tools_UpdateUserXML(userID: SYSUINT; quickUpdate: WordBool; xml: BSTR); - odpowiada menu Tools/Quick User Update/[Update User]. (Parametr userID - ID użytkownika; parametr quickUpdate- określa czy zmiany mają być przesłane do urzędzeń; parametr xml-patrz wyżej: Tools GET UserXML)*
- *procedure _Tools_DeleteUser(userID: SYSUINT; quickUpdate: WordBool); - odpowiada menu Tools/Quick User Update/[Delete User]. (Parametr userID - ID użytkownika; parametr quickUpdate- określa czy zmiany mają być przesłane do urzędzeń)*
- *procedure _Tools_QuickUserUpdate(userID: SYSUINT; quickUpdate: WordBool); (Parametr userID - ID użytkownika) Przesyła dane z bazy danych do urzędzeń.*

Zdarzenia:

- *procedure _OnConnect(const info: WideString); - zdarzenie zachodzące w momencie uzyskania połączenia z serwerem PRMaster. W parametrze info string z informacją o połączeniu (np. „, Connected to PRMaster. Server TCP IP=127.0.0.1 Port=80”)*
- *procedure _OnDisconnect; - zdarzenie zachodzące w momencie rozłączenia z serwerem PRMaster.*

- procedure `_OnMessage(const str: WideString);` - zdarzenie w momencie wystąpienia komunikatu (w parametrze `str` - komunikat)
- procedure `_OnErrorMessage(const str: WideString);` - zdarzenie w momencie wystąpienia komunikatu błędu (w parametrze `str` - komunikat błędu)
- procedure `_OnMonitorEvent(const xml: WideString);` - zdarzenie w momencie wystąpienia zdarzenia monitorowania (informacje dopisywane w oknie monitorowania na liście `Events` i `Alarms`). W parametrze `xml` dane zdarzenia.

Struktura XML (z przykładowymi danymi):

```
<monEvent>
  <code>10</code>
  <par>3072</par>
  <str>;27-05-2003;13:09:30;Aux. input returned to normal
cond.;PR301;;;Podsystem A;Domyślna;Internal door</str>
  <type>0</type>
</monEvent>
```

- procedure `_OnUpdateSubsPanels(const xml: WideString);` - zdarzenie w momencie odświeżania zawartości paneli „Subsystem”. W parametrze `xml` dane zdarzenia.

Struktura XML (z przykładowymi danymi):

```
<subs>
  <sub nr="1" cl="-16777201">
    <dt>27-05-2003 13:07:43 Tuesday</dt>
    <ev>0</ev>
    <led>
      <al>8421504</al>
      <t>8421504</t>
      <c>8421504</c>
      <ba>8421504</ba>
      <ac>8421504</ac>
      <bu>8421504</bu>
    </led>
  </sub>
</subs>
```

- procedure `_OnUpdateUserLogin(const xml: WideString);` - zdarzenie w momencie odświeżania informacji o logowaniu użytkownika na czytniku. W parametrze `xml` dane zdarzenia.

Struktura XML (z przykładowymi danymi):

```
<tables>
  <table>
    <row>
      <col>4</col>
    </row>
    <row>
      <col>1</col>
    </row>
```



```

<row>
  <col>1</col>
</row>
<row>
  <col>PR302 LCD/128.2/0004/878_T1</col>
</row>
<row>
  <col>13:02:10</col>
</row>
<row>
  <col>Przyznanie dostępu</col>
</row>
<row>
  <col>ŚLUSARSKI BOGDAN</col>
</row>
<row>
  <col>Bez grupy</col>
</row>
<row>
  <col />
</row>
<row>
  <col>2005/12/20</col>
</row>
<row>
  <col>13:2:10</col>
</row>
<row>
  <col>32</col>
</row>
</table>
</tables>

```

- procedure `_OnUpdateControllerState(const xml: WideString);` - zdarzenie w momencie odświeżania zawartości tabeli „ *Controllers States* “. Wparametrze *xml* dane zdarzenia.

Struktura XML (z przykładowymi danymi):

```

<tables>
  <table>
    <row>
      <col>PR302 LCD/128.2/0007/689_T1</col>
    </row>
    <row>
      <col>-</col>
    </row>
    <row>
      <col>-</col>
    </row>
    <row>
      <col>- / -</col>
    </row>
    <row>
      <col>1 / - / - / -</col>
    </row>
    <row>
      <col>Otwarte</col>
    </row>
    <row>
      <col>Brak</col>
    </row>
  </table>
</tables>

```

```
<row>
  <col>Brak odpowiedzi z CPR</col>
</row>
<row>
  <col>Tryb ROZB.</col>
</row>
</table>
</tables>
```