Mobility

CirPark Scada

Scada software for parking management

Devices Manual

V4.2.2

MU610105-16A-EN-CirPark Scada-4-Devices

1 Devices

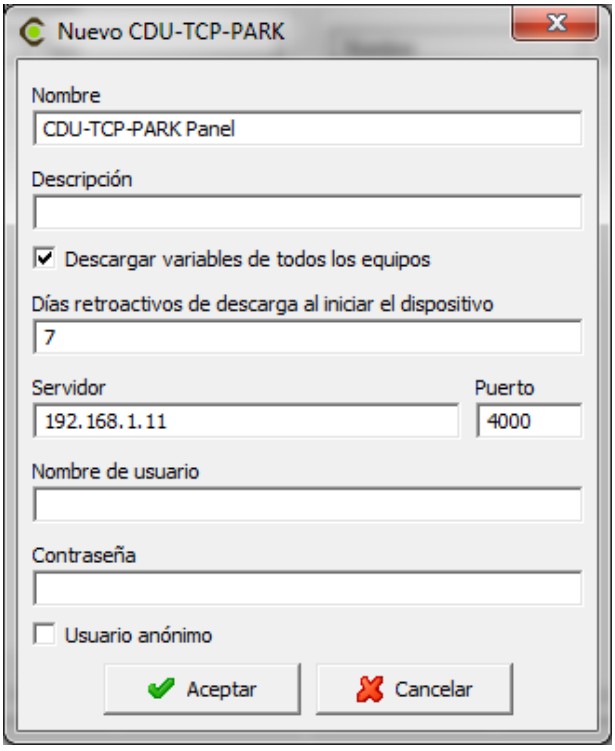
1.1 CirPark Client

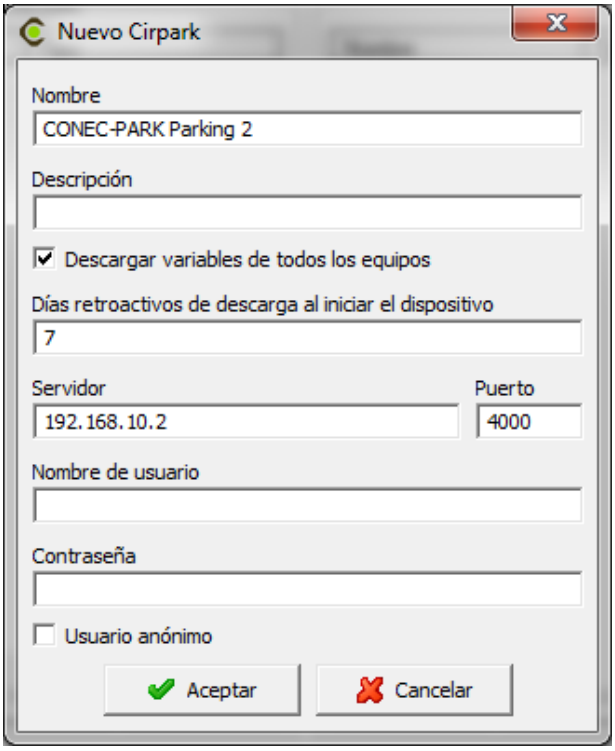
The CirPark client driver, CONEC-PARK, CDU or another CIRPARK, enables access to devices and variables from a CirPark that is being executed in a remote machine.

Once the variables that need to be read are configured, the CirPark client will behave like the rest of the devices in communication with the program.

1.1.1 Cirpark Client Device configuration

New CirPark clients must be registered using the device menu and added as a top level device.



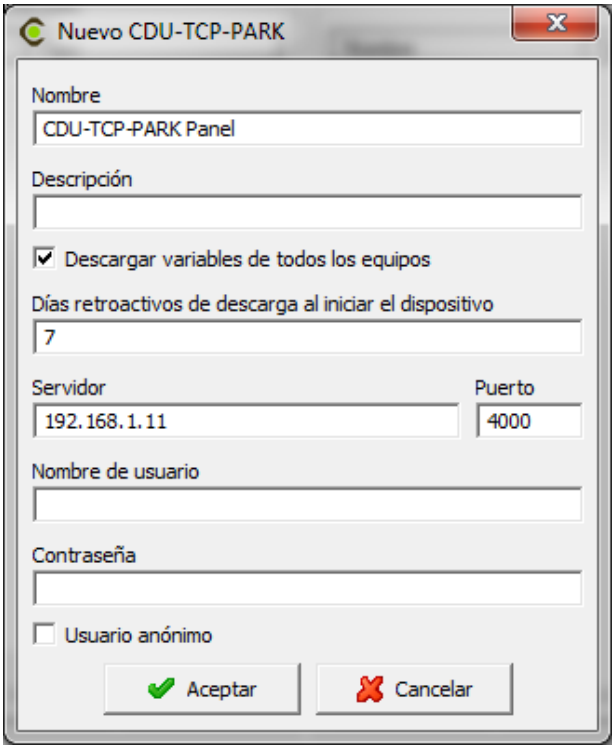


As was seen in the previous dialogue box, it is necessary to enter the address and gateway of the CirPark remote server that is going to be connected, as well as the user and password if the authentication is activated.

There is a "Download all equipment variables" option that can be marked to automatically download all the equipment data and variables that can be accessed by the remote CirPark. If this option is not marked each variable must be entered manually.

1.2 CDU-TCP-PARK as a Cirpark client

A new CirPark client must be registered using the device menu and added as a top level device.

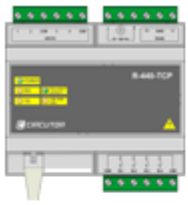


As was seen in the previous dialogue box, it is necessary to enter the address and gateway of the CirPark remote server that is going to be connected, as well as the user and password if the authentication is activated.

A "Download all equipment variables" option is available, which will automatically download all the equipment data and variables that can be accessed by CirPark remote. If this option is not marked, each variable must be entered manually.

1.2.1 Driver Options

The following menu options will be available:



Dispositivo: CDU-TCP-PARK

Nombre: cdu

Descripción:

Servidor: 192.168.1.34

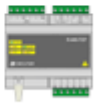
Puerto: 80

Comunicación: Sí

Histórico: Sí

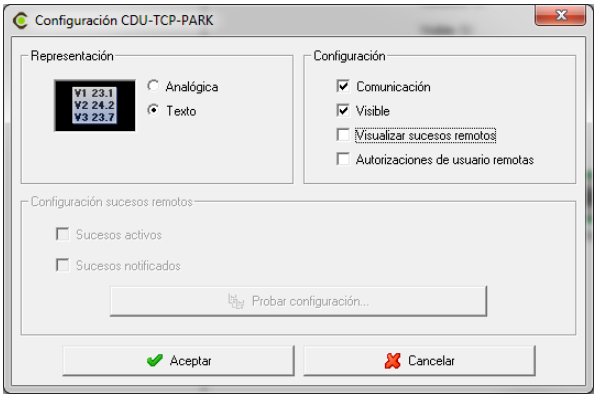
Visible: Sí

Parámetros del driver



Parámetros del dispositivo

1.2.1.1 Driver Parameters



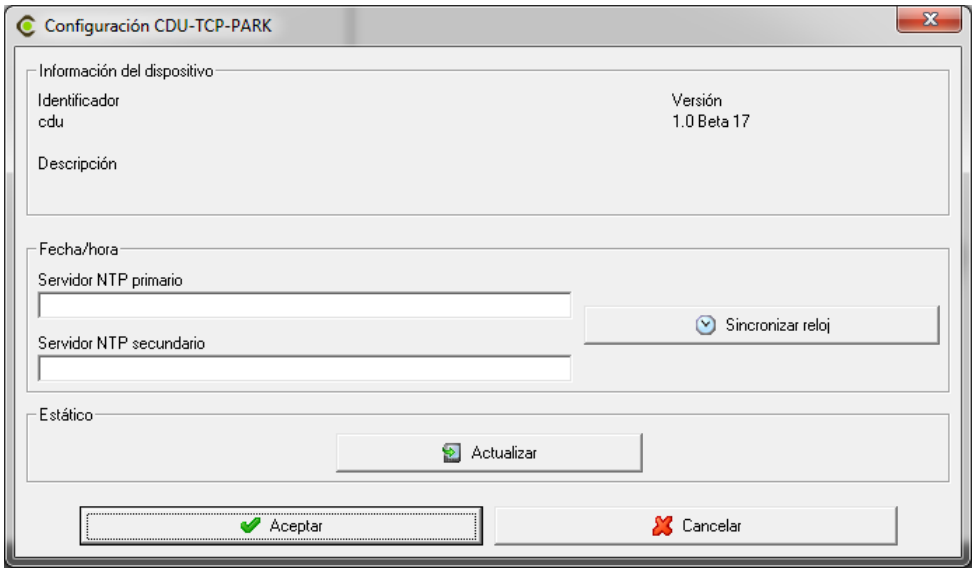
Enables define the variables analogically or with text.
The driver configuration can be adjusted for:

- Establishing communication from the main motor.
- A viewable device from the main motor.
- Displaying remote incidents configured in the Embedded Cirpark system of the device.
- Accessing with remote user authorisations from the Cirpark Embedded system.

By enabling the "Display remote incidents" option the remote incidents configuration can be activated because they now become accessible through the main system.

1.2.1.2 Device Parameters

When the dialogue is opened the software will read the device configuration and display it in the top section of the screen.
The middle section is used for configuring the internal device parameters. In this area the NTP server can be defined for setting a schedule or synchronising the motor clock through which communications with the CDU are made. Finally, there is an option for updating the device.



1.3 CDU-TCP-PARK as a device

The CDU driver enables accessing devices and variables in an Embedded Cir Park that is being executed in the device.

Once the variables that need to be read are configured, the CirPark client will behave like the rest of the devices in communication with the program.

1.3.1 Driver Options

This enables defining the numerical representation device mode, configuring communications, visibility and the logged data history log, as well as the device capture period.

1.3.1.1 Driver Parameters

Displays:

- General device information and the value of the counters associated with its digital inputs.
- Enables resetting the counter values.
- Offers the option of synchronising the internal device clock with the motor.

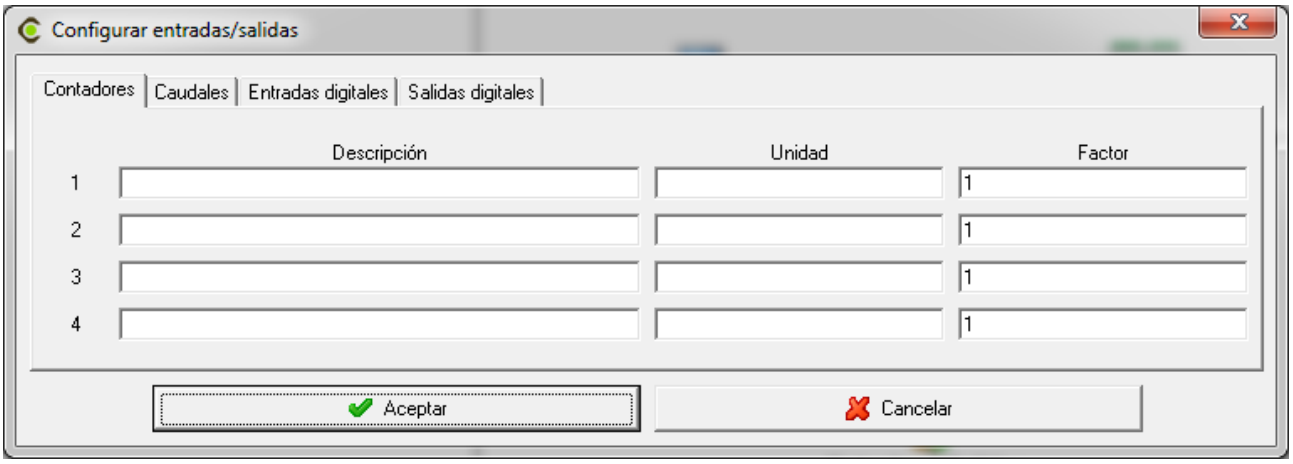
1.3.1.2 Inputs/Outputs Configuration

The unit is equipped with a total of 4 digital inputs, which have the function of counting energy pulses coming from external sensors or detecting the logic state of the input. The contacts associated with the unit's digital inputs must be voltage free contacts.

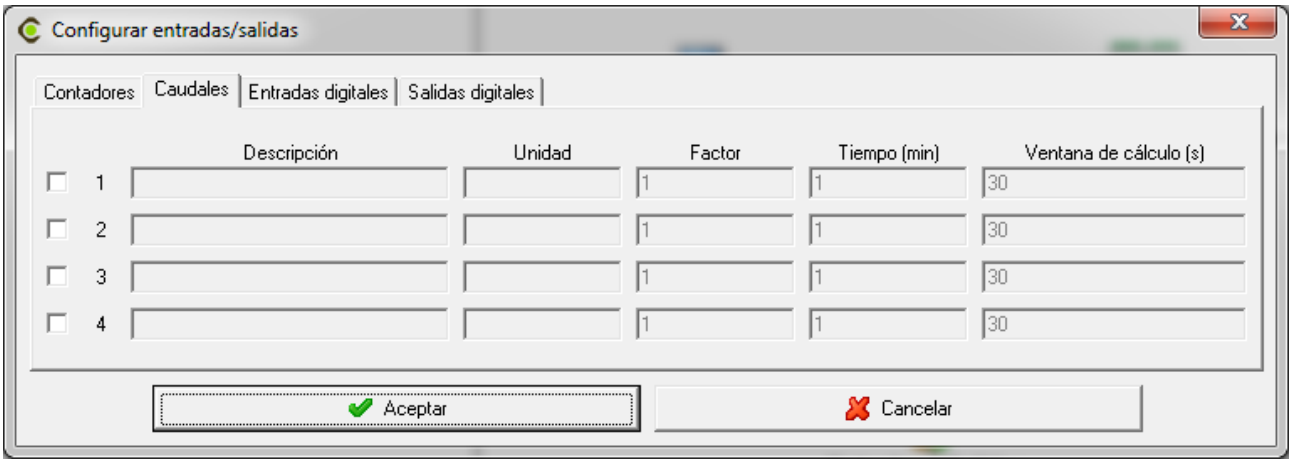
The electronic counters have a pulse output proportional to the energy logged. The CDU-TCP-PARK is a centralised unit equipped with 4 digital inputs (opto-coupled) for reading pulses through its inputs sent by step counters and electric, water, and gas meters, etc.

The value of said pulses is associated with 4 memory logs stored in a non-volatile memory. Each log is 32 bits (4 bytes), and consequently can count up to a maximum of 4,294,967,295 pulses. When a memory log reaches said value, the counter resets to zero. The minimum duration of the pulse or state change must be 50 ms, and the minimum time between two successive pulses through the same input must be 50 ms. This represents a maximum sample frequency of 10 Hz.

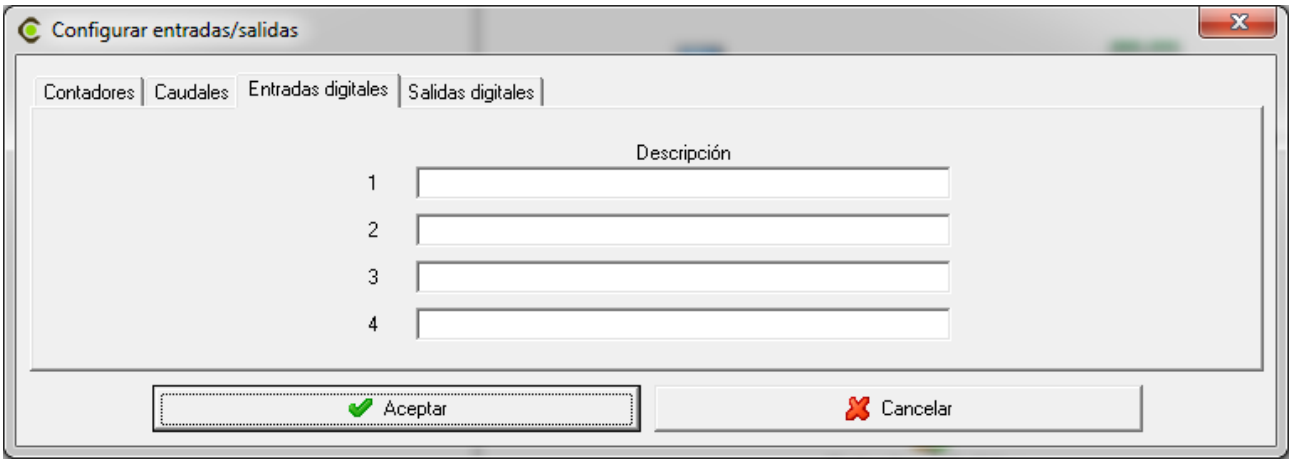
Its internal memory enables the user to extract graphs and tables of the pulses received in a given period.



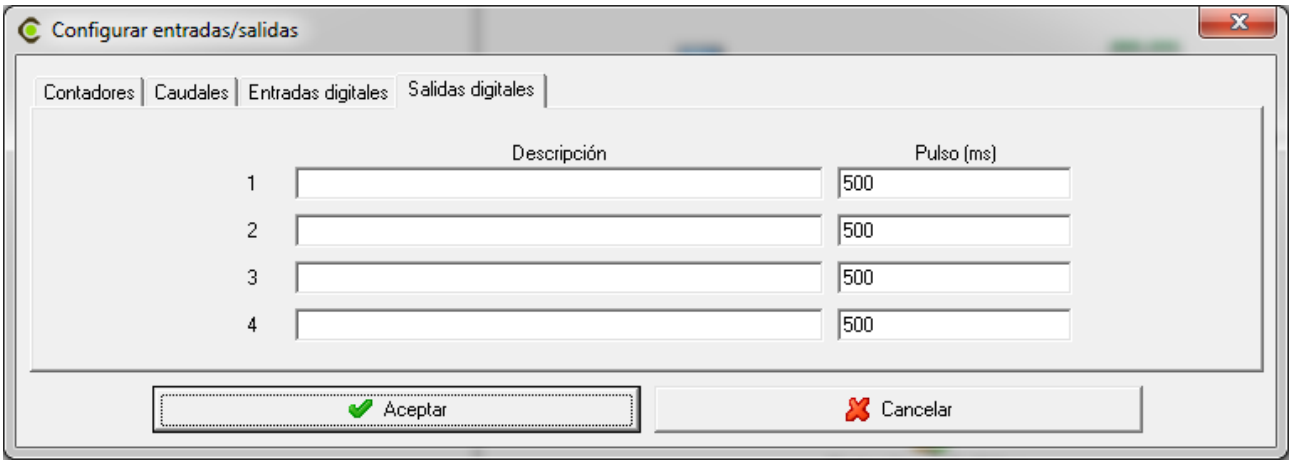
Configuration of the system to log flows by acquiring pulses at its digital inputs. The Internal configuration of the system with counters is based on the time to establish flow meters.
A description of each flow meter is defined, including the unit of measurement, the multiplication factor, the acquisition time and the calculation window.



The digital inputs can be managed as independent signals, for which a description of each one can be defined. The 4 available inputs on the unit are voltage free and have function for detecting the input logic state. This means that when a bridge is made between the common input and one of the digital inputs, the unit closes the input after viewing the state through both communication servers.

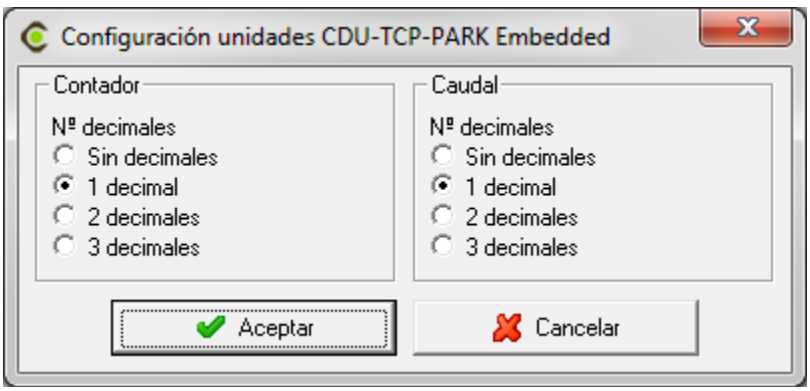


The unit has 4 relay controlled digital outputs. The user can perform telecontrol actions on the outputs through the communication servers (open, close or generate a pulse). These actions can be carried out manually or programmed in the event section of the device (see CirPark Scada manual).

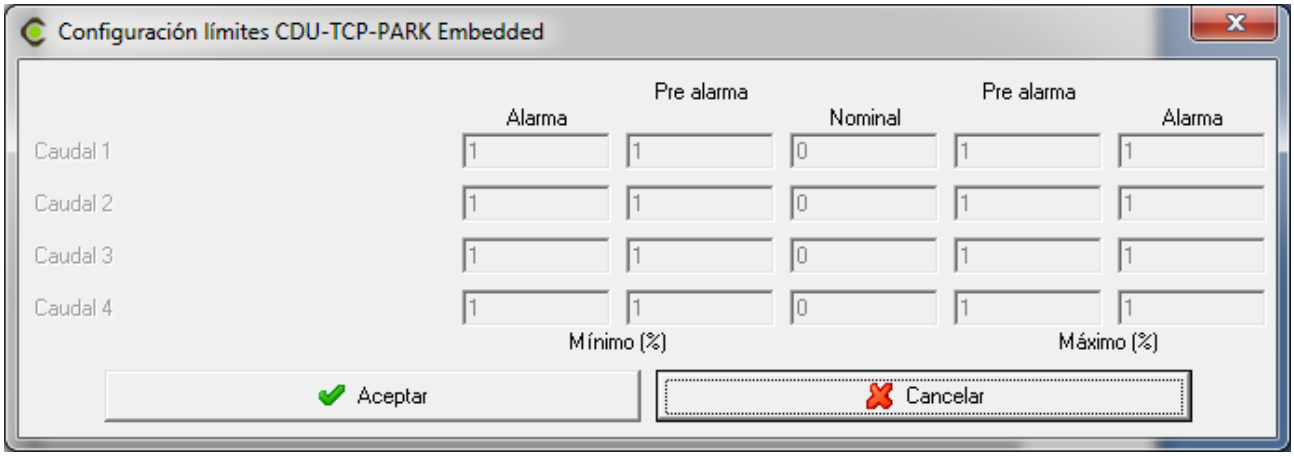


1.3.1.3 Variables Units

The number of decimals is defined by counter and by flow meter.



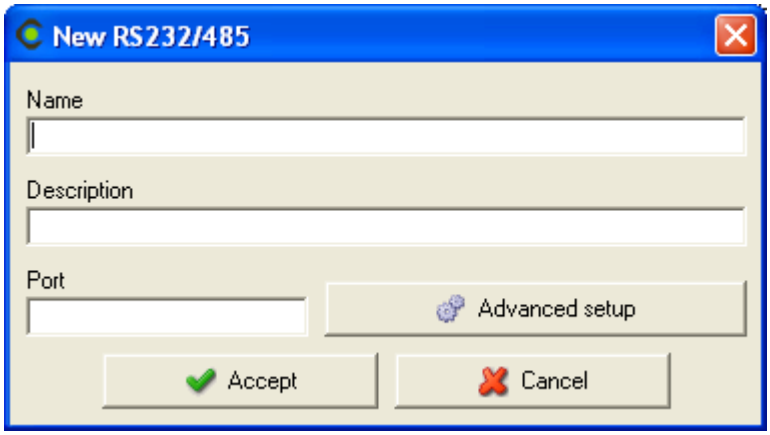
1.3.1.4 Variables limits The alarm and pre-alarm statuses are defined as above and below a nominal value of each counter.



1.4 RS232/485 converter

The converter RS232/485 connects equipment which works with an RS-485 serial connection to the port of the PC with an RS-232.

By adding or modifying this device, see “Editor Manual”, the following dialogue will appear:

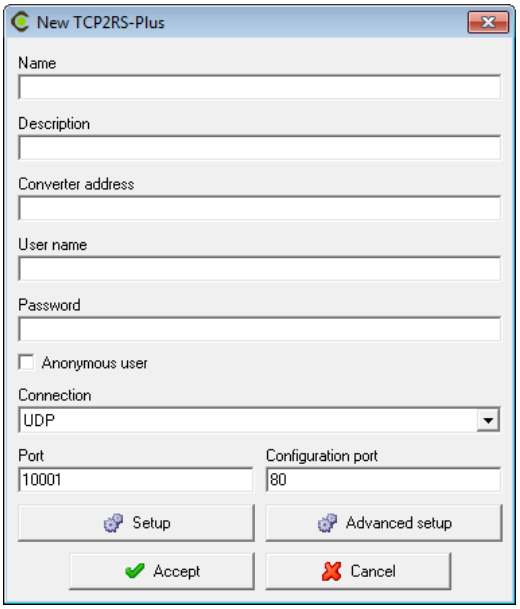


- **Name:** Alphanumeric field which uniquely identifies the device throughout the program. There are no two devices in the configuration with the same name.
- **Description:** Alphanumeric type data to insert a brief description of the device.
- **Port:** Number of the serial port that the program uses for communication with the device.
- **Advanced setup** Permits the configuration of a series of additional parameters of the device. See “Editor Manual”.

1.5 TCP2RS+ Converter

The TCP2RS+ converter connects units that work with serial communications (RS-232 / RS-485) to an Ethernet network using IP protocol.

When adding or changing this device, the following dialogue appears (Consult "Manuel Editor"):



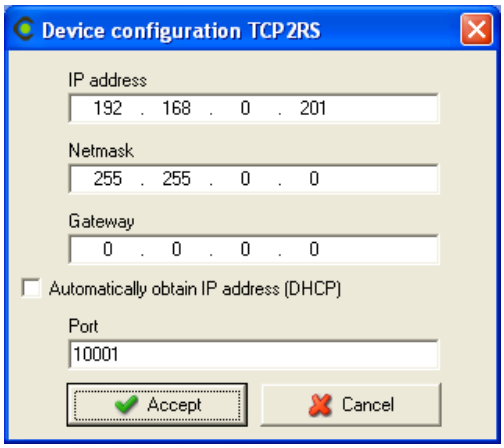
- **Name:** Alphanumeric field which uniquely identifies the device throughout the program. There are no two devices in the configuration with the same name.
- **Description:** Alphanumeric type data to insert a brief description of the device.
- **Converter address:** Corresponds to the address through which the program can communicate with the device. This parameter can be an IP address or a name. This direction should not be confused with the MAC address.
- **User name:** There is a possibility to assign a username and password to the device to prevent unauthorized access to the device.
- **Anonymous user:** if not assign username and password, this check box is marked to indicate that you access as an anonymous user
- **Connection**: Protocol communication between the platform and the device.
- **Port:** Corresponds to the converter communications port:
- **Configuration port:** Corresponds to the communication port, through which the program can configure the device. This port is 30718 by defect.

To be able to communicate a TCP2RS device through a router carry out the following steps:

1-In the "Converter Address" field inserts the IP router address.
2-In the "*Port*" field inserts the communications port and redirects this port on the router to the converter communications port.
3 - Redirect the configuration port on the router to port 30718 of the converter.

NOTE: For information on re-addressing ports consult the user manual.

- **Setup:** By pressing the button a dialogue will appear where a number of parameters of the device can be configured.

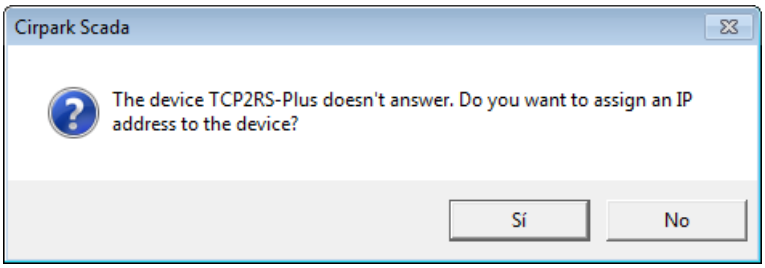


Where:

- *IP address* Corresponding to the IP address through which we can communicate with the device.
- *Subnet Mask:* Corresponds to the subnet mask used on the network where the device is connected
- *Predetermined Gateway:* Corresponds to the address of the gateway if the device is not on the same network as the computer containing the program.
- *Automatically obtaining IP address (DHCP):* This option will be activated when we want the device to automatically receive the IP address via a DHCP server.
- *Port:* Corresponds to the communications port of the device.

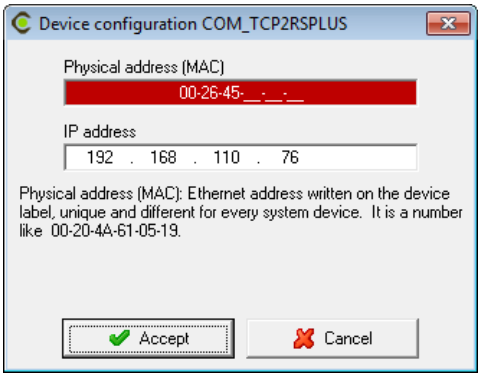
- **Advanced setup:** Permits the configuration of a series of additional parameters of the device. See "Editor Manual".

On adding or modifying the device, the software will try to detect it. If it is unable to detect it, if the device is new or not been allocated an IP address or has an allocated IP different to that inserted in the *'Converter Address*"), it will ask if we want to assign a new IP address to the converter.



It will only be possible to assign an IP address to the device on the same network as the computer running the program.

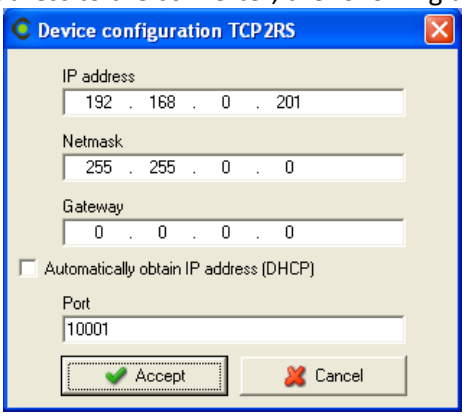
If the answer *'Yes'*, the following dialogue box will appear, which will permit an IP address to be assigned to the device.



- **Physical address (MAC):** Ethernet address that each device has, which is unique and distinct on all network devices. The hardware addresses which each network interfaces possess. Will be type 00-20-4A-61-05-19.

- **IP address** IP Address to be allocated to the device that has the physical address introduced in the previous field.

If it is possible to assign an IP address to the converter, the following dialogue box will appear:

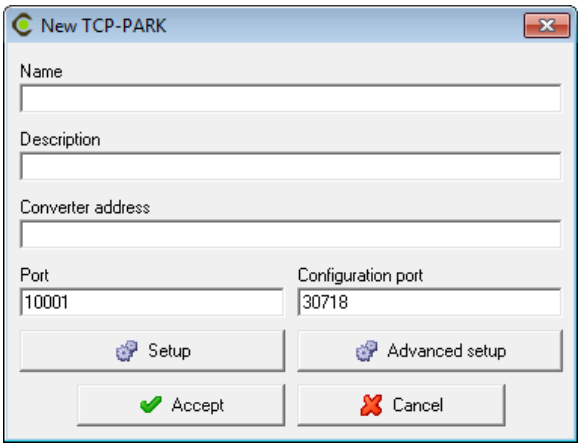


To finish set up with the rest of the parameters to communicate with the device.

1.6 TCP-PARK concentrator

The TCP-PARK concentrator connects the car park devices that connect by serial communications (RS-485) to an Ethernet network with IP protocol. This includes the converter features that the TCP2RS+ offers plus the features of the signal concentrator of all the devices on the RS-485 line that it controls.

When adding or editing a TCP-PARK concentrators, see "Editor Manual", following window appears:



- **Name:** Alphanumeric field which uniquely identifies the device throughout the program. There are no two devices in the configuration with the same name.

- **Description:** Alphanumeric type data to insert a brief description of the device.

- **Converter address:** Corresponds to the address through which the program can communicate with the device. This parameter can be an IP address or a name. This direction should not be confused with the MAC address.

- **Port:** Corresponds to the converter communications port:

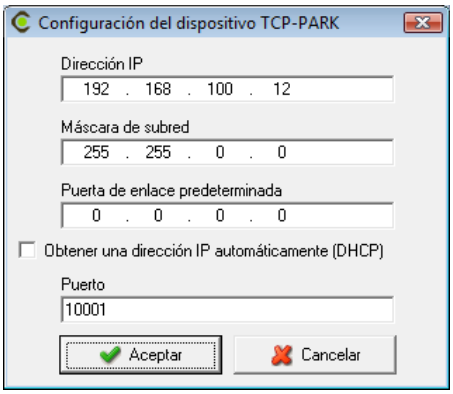
- **Configuration port:** Corresponds to the communication port, through which the program can configure the device. This port is 30718 by defect.

To be able to communicate a TCP2RS device through a router carry out the following steps:

1-In the "Converter Address" field inserts the IP router address.
2-In the "*Port*" field inserts the communications port and redirects this port on the router to the converter communications port.
3 - Redirect the configuration port on the router to port 30718 of the converter.

NOTE: For information on re-addressing ports consult the user manual.

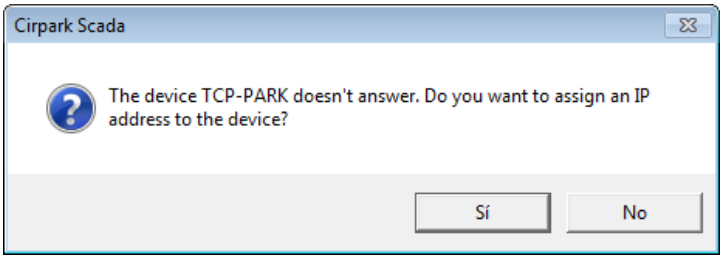
- **Setup:** By pressing the button a dialogue will appear where a number of parameters of the device can be configured.



- *IP address* Corresponding to the IP address through which we can communicate with the device.
- *Subnet Mask:* Corresponds to the subnet mask used on the network where the device is connected
- *Predetermined Gateway:* Corresponds to the address of the gateway if the device is not on the same network as the computer containing the program.
- *Automatically obtaining IP address (DHCP):* This option will be activated when we want the device to automatically receive the IP address via a DHCP server.
- *Port:* Corresponds to the communications port of the device.

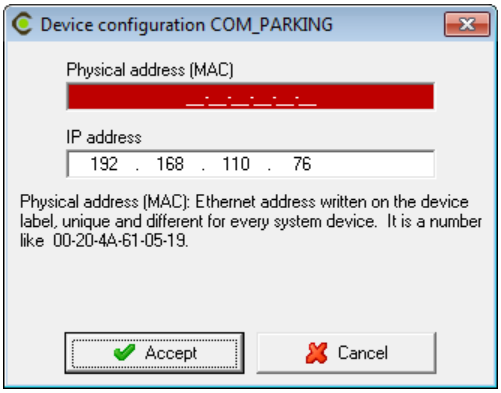
- **Advanced setup:** Permits the configuration of a series of additional parameters of the device. See "Editor Manual".

On adding or modifying the device, the software will try to detect it. If it is unable to detect it, if the device is new or not been allocated an IP address or has an allocated IP different to that inserted in the *'Converter Address*"), it will ask if we want to assign a new IP address to the converter.



It will only be possible to assign an IP address to the device on the same network as the computer running the program.

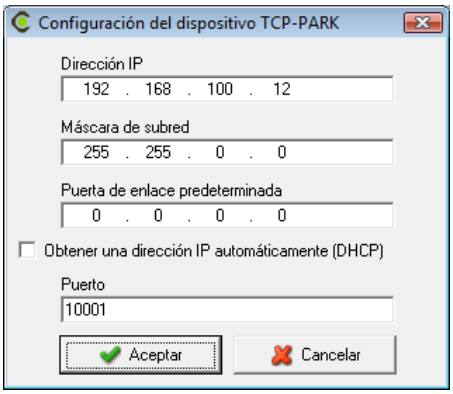
If the answer *'yes'*, the following dialogue box will appear, which will permit an IP address to be assigned to the device.



- **Physical address (MAC):** Ethernet address that each device has, which is unique and distinct on all network devices. The hardware addresses which each network interfaces possess. Will be type 00-20-4A-61-05-19.

- **IP address** IP Address to be allocated to the device that has the physical address introduced in the previous field.

If it is possible to assign an IP address to the converter, the following dialogue box will appear:



To finish set up the rest of the parameters for communication with the device.

1.7 TCP-PARK+ Concentrator

The TCP-PARK concentrator connects the car park devices that connect by serial communications (RS-485) to an Ethernet network with IP protocol. It includes the converter features that the TCP2RS+ offers plus the features of the TCP-PARK and new functions such as the exchange, plus those in the signal concentrator of all the devices on the RS-485 line that it controls.

When adding or changing this device, the following dialogue appears (Consult "Manuel Editor"):

- **Name:** Alphanumeric field which uniquely identifies the device throughout the program. There are no two devices in the configuration with the same name.

- **Description:** Alphanumeric type data to insert a brief description of the device.

- **Converter address:** Corresponds to the address through which the program can communicate with the device. This parameter can be an IP address or a name. This direction should not be confused with the MAC address.

- **User name:** There is a possibility to assign a username and password to the device to prevent unauthorized access to the device.

- **Anonymous user:** if not assign username and password, this check box is marked to indicate that you access as an anonymous user

- **Port:** Corresponds to the converter communications port:

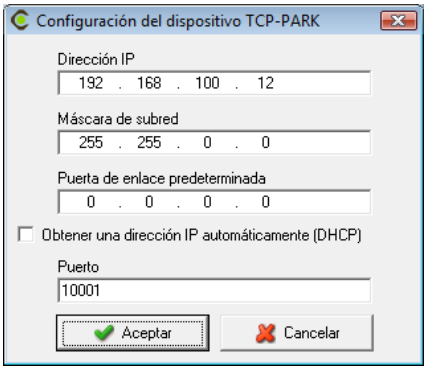
- **Configuration port:** Corresponds to the communication port, through which the program can configure the device. This port is 80 by defect.

The following steps must be carried out to enable a TCP-PARK to communicate via a router:

1.- Enter the router IP address in the *Converter address* field.
3. - Redirect the configuration port in the router to converter port 30718.

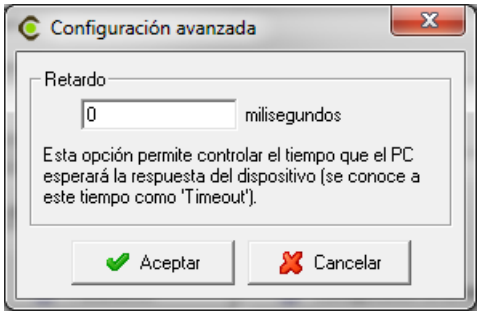
NOTE: Consult the router manual on how to redirect the ports.

- **Parameters:** Pushing the button opens a dialogue box for configuring various device parameters.

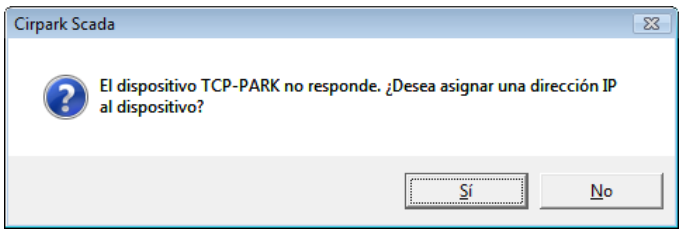


  - *IP Address*: Identifies the address through which the program can communicate with the device.
  - *Subnet mask*: Identifies the subnet mask used in the network where the device is connected.
  - *Default gateway*: Identifies the gateway port address in the event the device is not in the same network as the PC with the program.
  - *Obtaining an IP address automatically (DHCP):* This option is activated when it is necessary for the device to receive the IP address automatically via a DHCP server.
  - *Port:* Identifies the device communication port.

- **Advanced configuration:** Used to set the time of the data request from Cirpark to the TCP-PARK + device



The software tries to detect a new device or change to the existing one. If it fails to detect it, the device may be new and does not have an assigned IP address, or its assigned IP address is not the same as the address entered in the *Converter address* field, in which case the software will ask if an IP address should be assigned to the device.



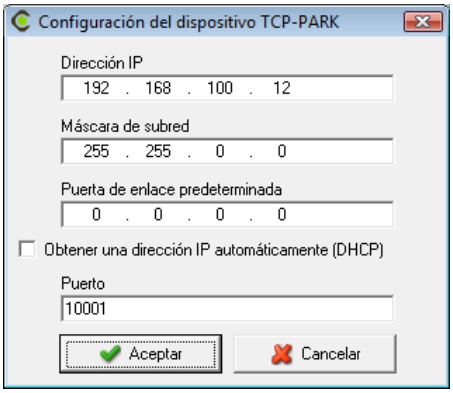
It is only possible to assign one IP address to the device if it is in the same network as the PC that executes the program.

If the reply is *Yes*, the following dialogue box will appear for assigning an IP address to the device.

- **Physical address (MAC):** This is the Ethernet address every device has, which is unique and distinct to each device on the network. It is the hardware address that every interface on the network has. It has a format such as 00-20-4A-61-05-19.

- **IP Address:** This is the IP address assigned to the device with the physical address entered in the previous field.

If it has been possible to assign the IP address to the converter, the following dialogue box will appear:



Complete the configuration of the rest of the parameters to communicate with the device.

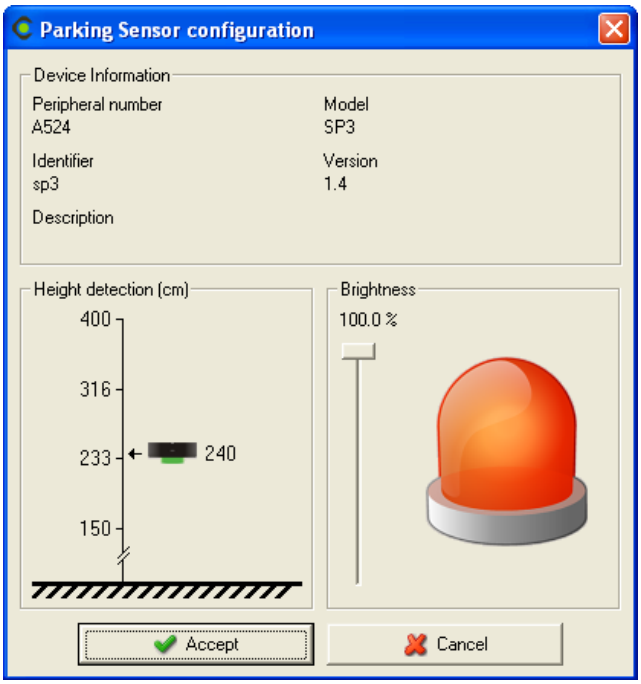
1.8 SP3 – Ultrasonic Parking Sensor

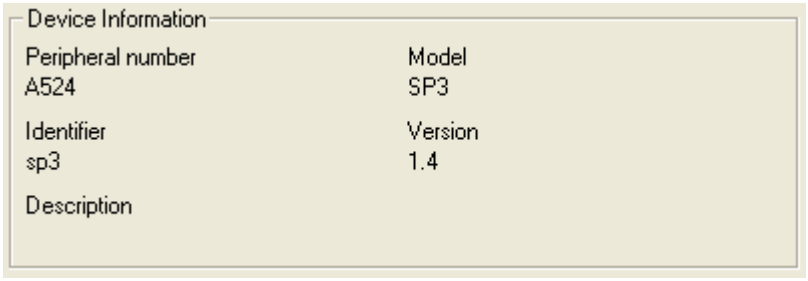
1.8.1 Driver Options

The following menu options will be available:

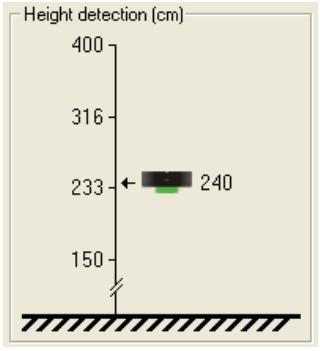
1.8.1.1 Device Parameters

This window gives access to the internal device parameters configuration. When opening this box, the CirPark Scada reads the device configuration. When "*Accept*" is pressed, changes are sent to the device. No information is saved on the Server PC.

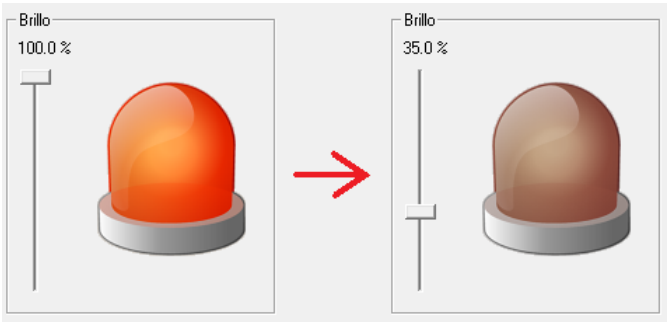




The top of this window shows the general device information such model type and firmware version.



From here the maximum detection distance can be established. Left click on the mouse and hold the button down over the sensor icon and slide it up or down to fix the height.

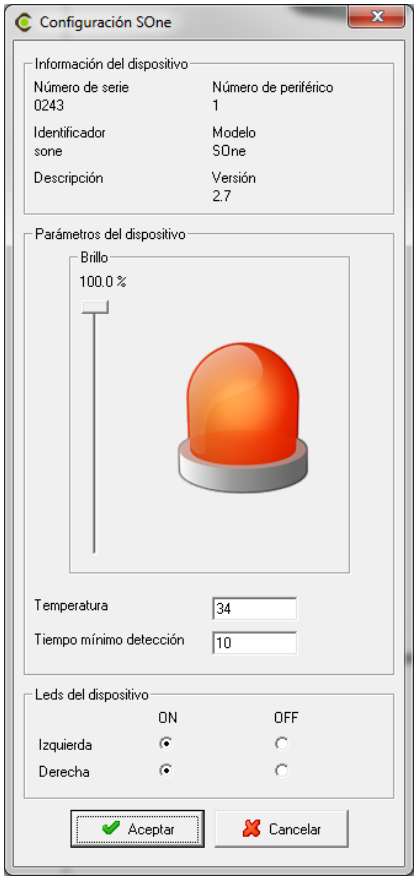


Here the brightness of the device can be adjusted. The brightness can be increased or decreased by sliding the vertical bar.

1.9 SOne – Optical Parking Sensor/Indicator/LedDriver/Tº Sensor

1.9.1 Driver options

The following menus options appear:

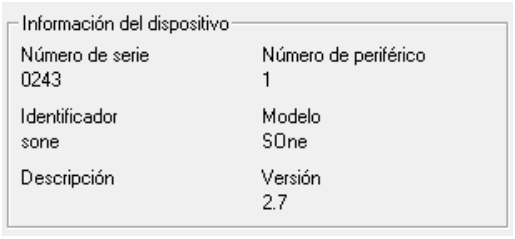


1.9.1.1 Device Parameters

This screen is used for configuring the internal device parameters. When the dialogue screen is opened the software reads the device configuration. When it finishes, pressing *Accept* will send any software changes made in the device. This information is never stored in the PC hard drive.

Here the general device parameters can be defined, including:

- Both sides of the sensor brightness indicator.
- Recalibrating the installed temperature sensor.
- Defining the speed change of the sensor status.
- Activating the bidirectional or unidirectional lighting in both directions.



Display general device information.

1.9.1.2 Advanced Configuration

Here the sensor measurement and its calibration system parameters can be set:

- Auto-calibration: Enables automatic calibrating by the sensor itself and the period, in seconds, of each auto-calibration.
- Displays the upper and lower free space threshold detection, as well as the margin established between both thresholds.
- This enables creating the reference calibration that the sensor needs to begin operating.
- This enables manual calibration.

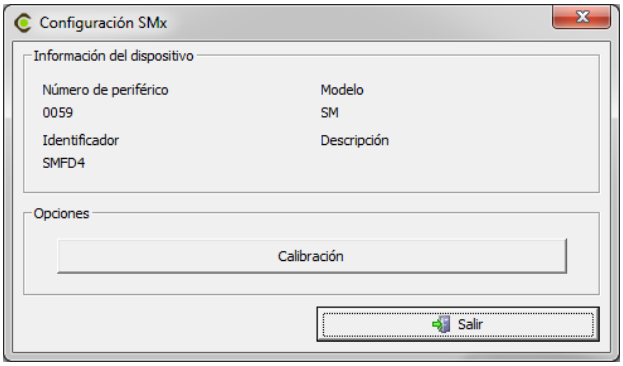
1.10 SM – Magnetic Surface Parking Sensor

1.10.1 Driver Options

The following menu options will be available:

1.10.1.1 Device Parameters

This screen is used for configuring the internal device parameters. When the dialogue screen is opened the software reads the device configuration. When it finishes, pressing *Accept* will send any software changes made in the device. This information is never stored in the PC hard drive.

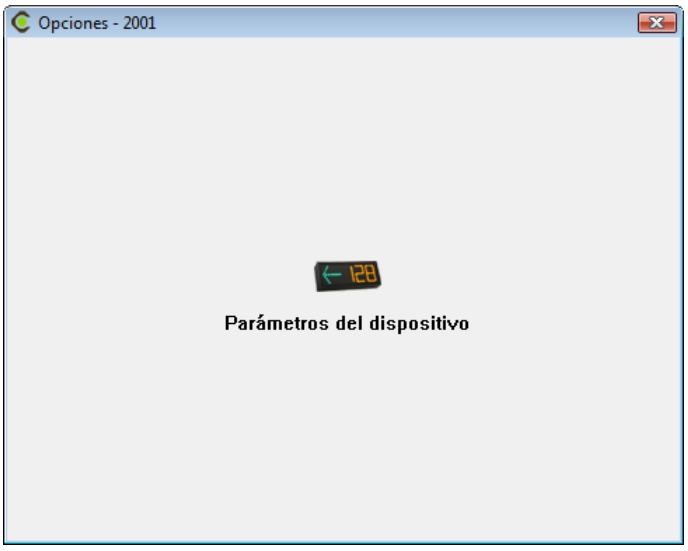


This screen displays information regarding the magnetic surface sensor, and also has an option for calibrating it.

1.11 Parking Display – basic range

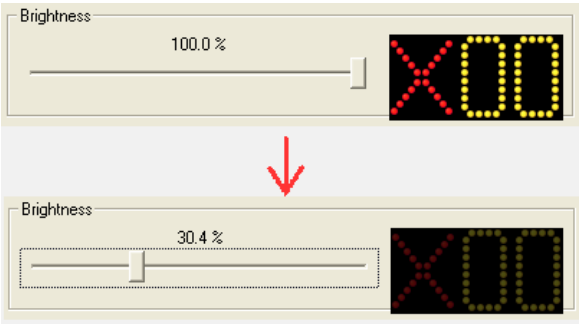
1.11.1 Driver Options

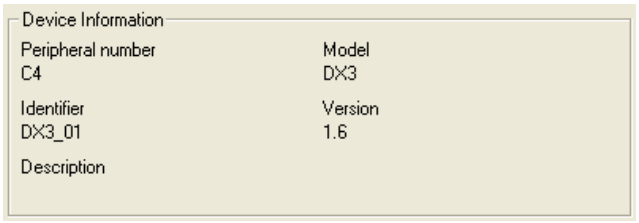
The following menu options will be available:



1.11.1.1 Device Parameters

This window gives access to the internal device parameters configuration. When opening this box, the CirPark Scada reads the device configuration. When "*Accept*" is pressed, changes are sent to the device. No information is saved on the Server PC.





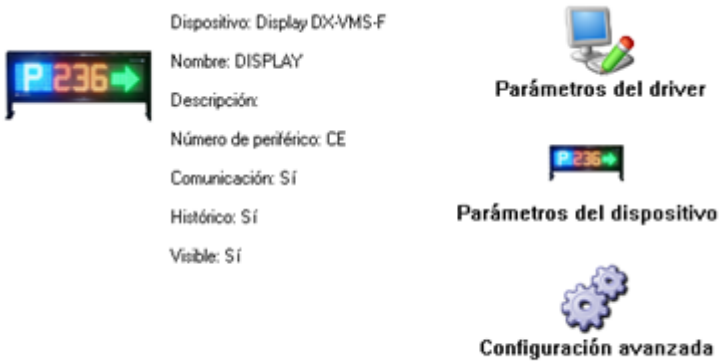
The brightness of the device can be adjusted. The brightness can be increased or decreased by sliding the horizontal bar like the above example.

The top of window shows the general device information such model type and firmware version.

1.12 Parking Display – advanced range

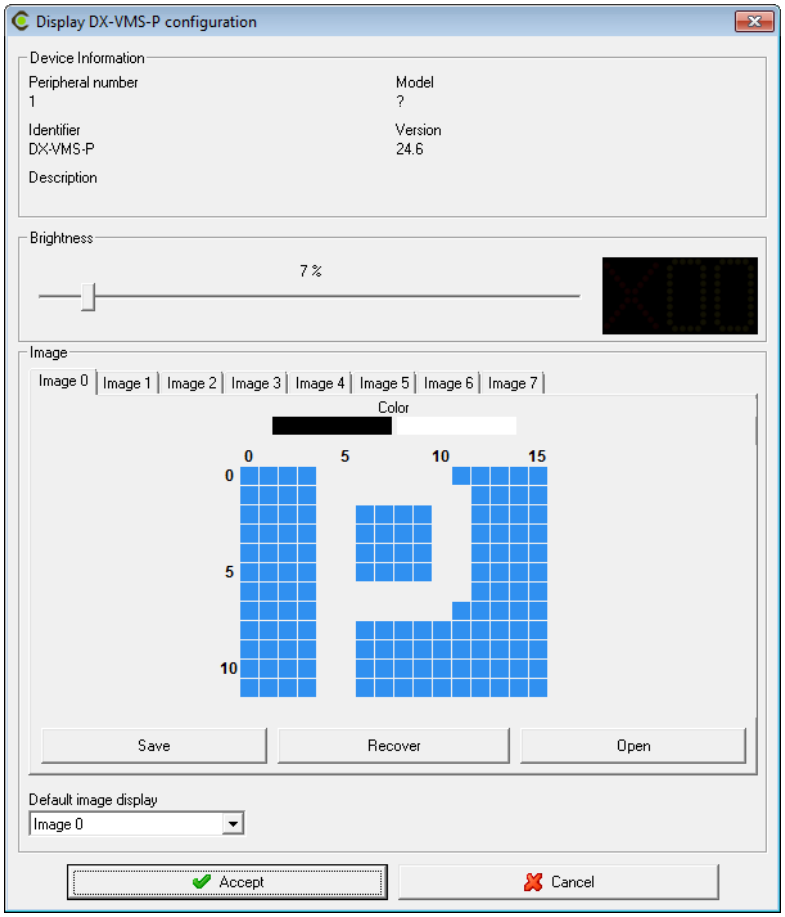
1.12.1 Driver Options

The following menu options will be available:



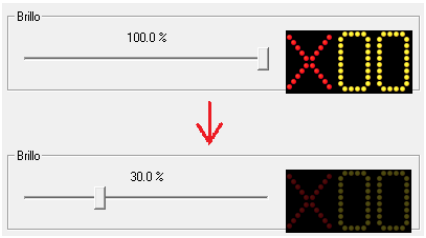
1.12.1.1 Device parameters

This screen is used for configuring the internal device parameters. When the dialogue screen is opened the software reads the device configuration. When it finishes, pressing *Accept* will send any software changes made in the device. This information is never stored in the PC hard drive.

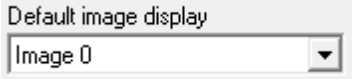


This window used is for:

- Adjusting the display brightness. The brightness can be increased or decreased by sliding the horizontal bar.
- Changing the default "P" for car parks. It is possible to save different formats in the file. The colour pallet can be used by choosing a colour for the left mouse button (black in the example screen), and a colour for the right button (white in the example screen), so that each pixel of the established area can be painted ("P" in the blue and white symbol background in the example screen).

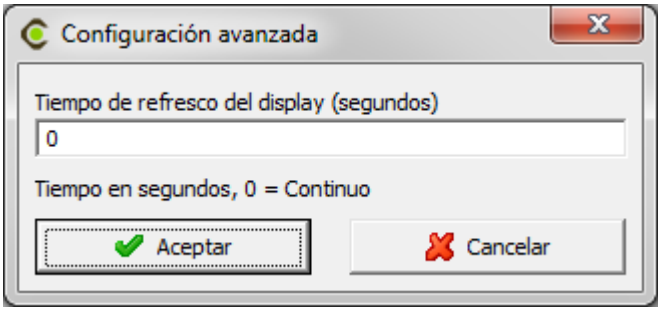


- You can save these drawn icons and able to recover. Also to choose an image file in the computer to transform it to icon format and display it on display. Giving the open button and choosing the desired image.
- 8 icons you have set so that they can exchange for the use of SCADA client. You can define which will be the default icon.



1.12.1.2 Advanced configuration

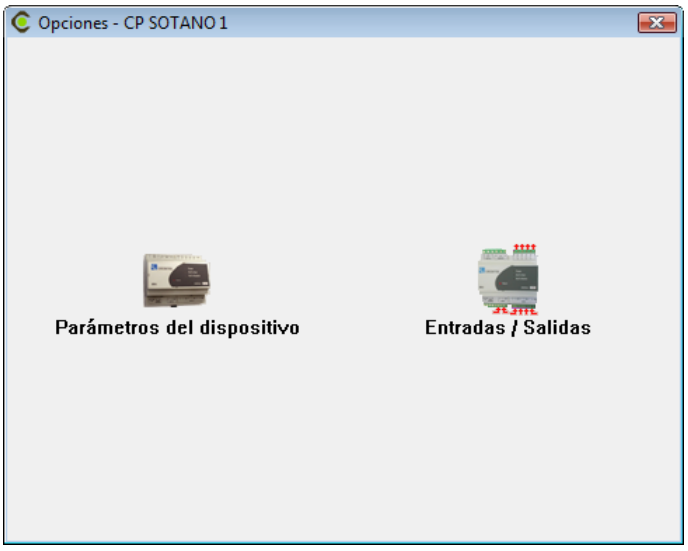
This screen is used for setting a delay time between the updating of the digits shown in the display, so that changes coming right after each other due to high traffic density are not shown on the display, which stabilises the values displayed.



1.13 MR4/dp – Parking Counter

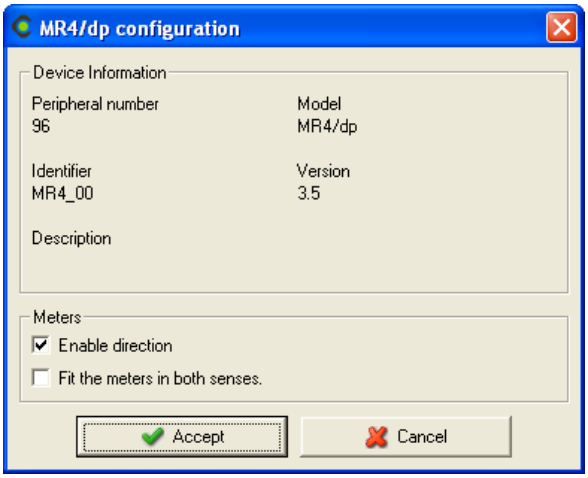
1.13.1 Driver options

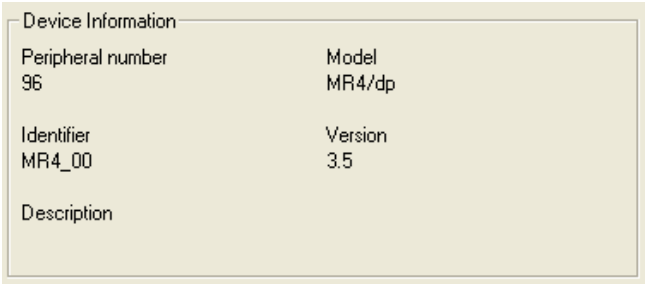
The following menu options will be available:



1.13.1.1 Device parameters

This window gives access to the internal device parameters configuration. When opening this box, the CirPark Scada reads the device configuration. When "*Accept*" is pressed, changes are sent to the device. No information is saved on the Server PC.





The top of this window shows the general device information such model type and firmware version.

The internal detection algorithm can be changed to adapt the MR4/dp to the detection type.

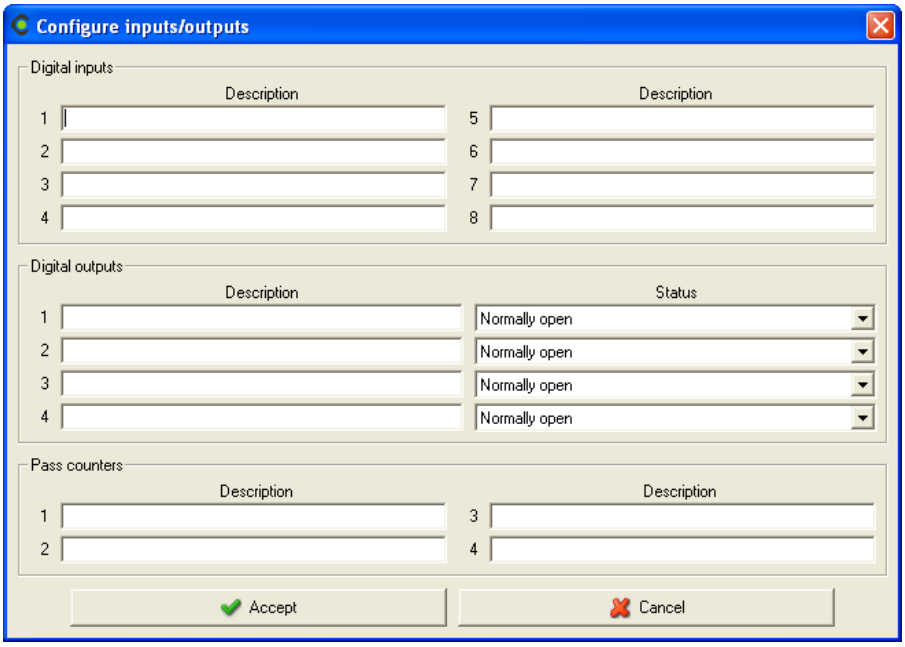
- **No option checked**: each input pair are functioning as a single input. This option is used when a potential free contact or single magnetic loop/photocell is connected.
  A jumper must unite both connectors of the pair.

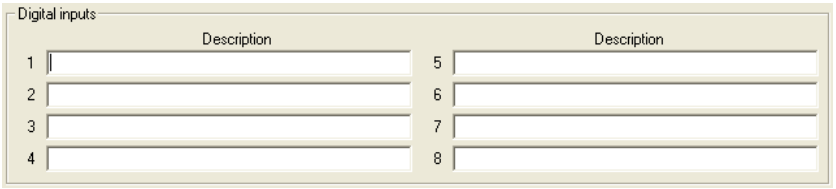
- **Enable direction:** This option modifies the internal detection algorithm allowing the directional detection of vehicles. This algorithm will detect a car when car is detected by the first input of the pair and then by the second one (maintaining the first). When the first one is released, the MR4/dp will count a crossing action. No other combination will count. Use this option when pair of magnetic loops or photocells is used. This is an unidirectional detection, only circulation from the lowest input to the highest of a pair detects detection (1->2 for pair 1, 3->4 pair 2, 5->6 pair 3 or 7->8 pair 4)

- **Fit the meters in both senses:** Only available when "Enable Direction" is enabled too. Enables bidirectional detection for each input pair. Different counters can be increased or decreased if the vehicle circulates in the normal direction (sense A) or in the reserved direction (sense B)

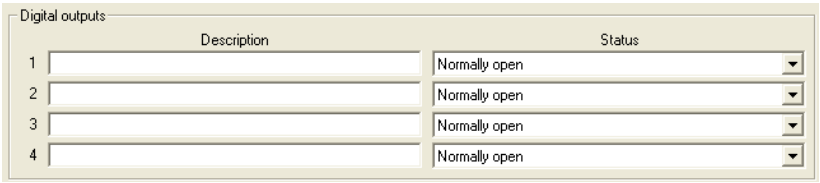
1.13.1.2 Input / outputs

This window is used to define name to the digital inputs and crossing detectors. Also, digital outputs name and default state (N.O, N.C) can be defined.





The descriptions of the digital inputs are used further in formulas and Scada Screens.
The four single inputs or four pairs can be defined depending of the configuration made in "Devices parameters".



Each input, a description is defined together with the default output state. Description is also used further in formulas and Scada Screens.

Each crossing pair a description can be defined. Description is used further in formulas and Scada Screens. Four pairs on DPU units are available.

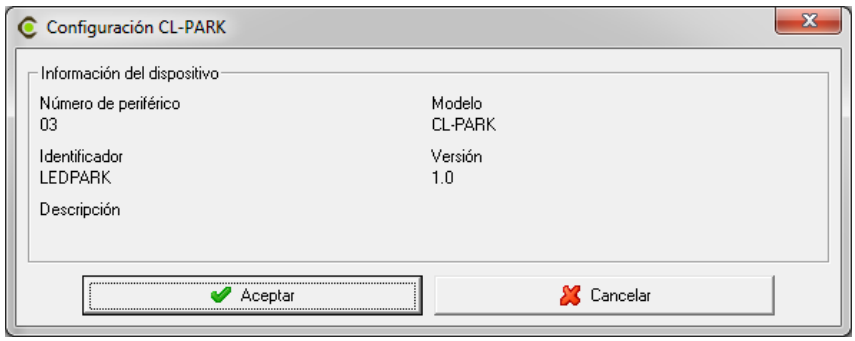
1.14 CL-PARK – Led Light Controller

1.14.1 Driver Options

The following menu options will be available:

1.14.1.1 Device parameters

This screen is used for configuring the internal device parameters. When the dialogue screen is opened the software reads the device configuration. When it finishes, pressing *Accept* will send any software changes made in the device. This information is never stored in the PC hard drive.



This device is not configurable, but it offers features for controlling the LED brightness from the Cirpark Scada platform for intelligent car park lighting management.

1.15 CVM-MINI – Grid Analyser

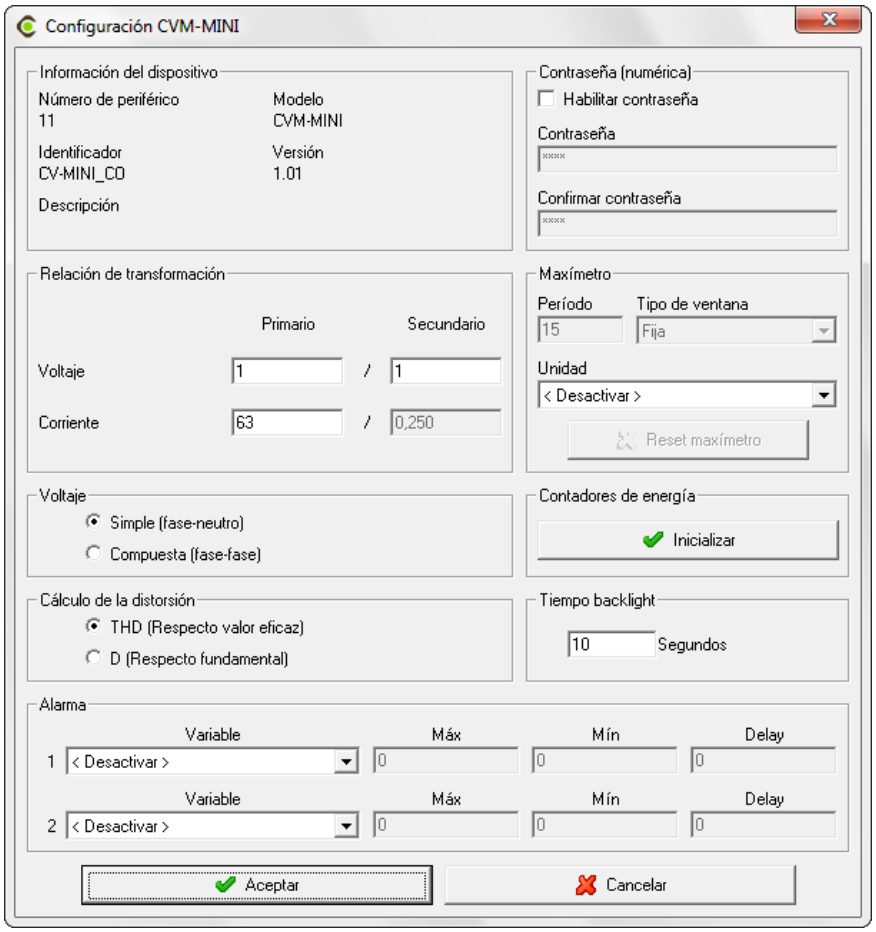
1.15.1 Driver options

The following menu options will be available:

The 'Variable Units' and 'Limit of the variables' are detailed in "1.3.1.3 Variables Units" and "0 Variables limits" respectively.

1.15.1.1 Device parameters

Internal devices parameters are defined in this window. This window shows the information received in real time from the device. If the device is not connected, the windows will not appear. When "accept" to leave this window, all changes will be sent to the device. No data is stored on the PC.



- Devices information**:** Displays general device info. The information available is: peripheral number, model ID, version, and description (if any).

- Password**:** Keyboard access to the device can be password protected. Check “Set password” and defined a password to block access to keyboard.

- Voltage relationship transformation**:** can be changed or disabled. This feature is only available in latest firmware version.

  - **Voltage Transformation Relationship**: Set the voltage transformation relationship between primary and secondary. The result of multiplying the primary value and the primary current must be equal or lower than 20.000.000. The primary value (depending of the full scale) must be between 1 and the corresponding value of the following table.

Full Scale	Maximal Value
110V	99.999
250V	70.000
275V	70.000
300V	70.000
500V	40.000

- **Current Transformation Relationship**: Device current primary value. Values oscillate between 1 and 10.000 Amps. Secondary current is 5A on the CVM-96.

- Maximeter**:** The Maximeter is responsible for registering the maximum power in a power supply for periods of 15 minutes, so you know whether or not exceeds the contracted capacity

  - **Period**: Integration period of the maxi meter oscillates between 1 and 60 minutes
  - **Window Type**: Window type used to save the maxi meter value can be fix or slicing.
  - **Unit**: Maximal demand is calculated using active power III, apparent power III, current III or current stage. This parameter is device dependent.
  - **Reset de maxi meter**: Press the button to reset the maxi meter.

- Voltage: Select the calculated voltage type.

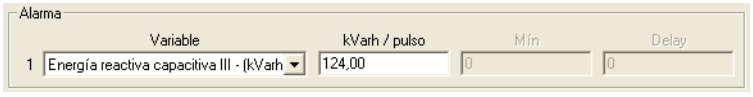
- Backlight time**:** Defines the "*backlight*" disconnection configuration type. Values must oscillate between 0 and 99 seconds.

- Calculated disturbed: Choose the distortion type used in the device calculation.

- Energy meter**:** Click this button to reset the device energy counters.

- Alarm The device relay output configuration is shown here. Values can be modified.
  Both relays outputs of the CVM-MINI devices can be configured.

  - Variable: This field shows the parameter linked to the relay. The output is programmed as an alarm or an energy pulse. Both parameters measured by the device can be controlled. Units are shown in parentheses. These units indicate the maximal and minimal values of the alarm.
  - Max: Defined the maximal control value. When an energy value is set, each pulse value will be indicated.
    Pulses values will be shown as follows:



  - **Min**: Defines the minimal control value. When an energy value is set, this value is disabled.
  - **Delay**: Alarm delay in seconds. The maximal value is 9999. This field is not present if an energy value is set.

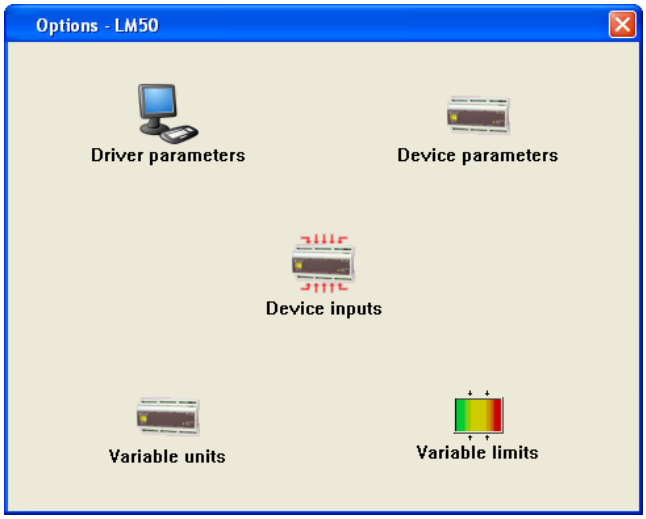
1.16 LM50-TCP

1.16.1 Driver options

You can access the LM50-TCP options from:

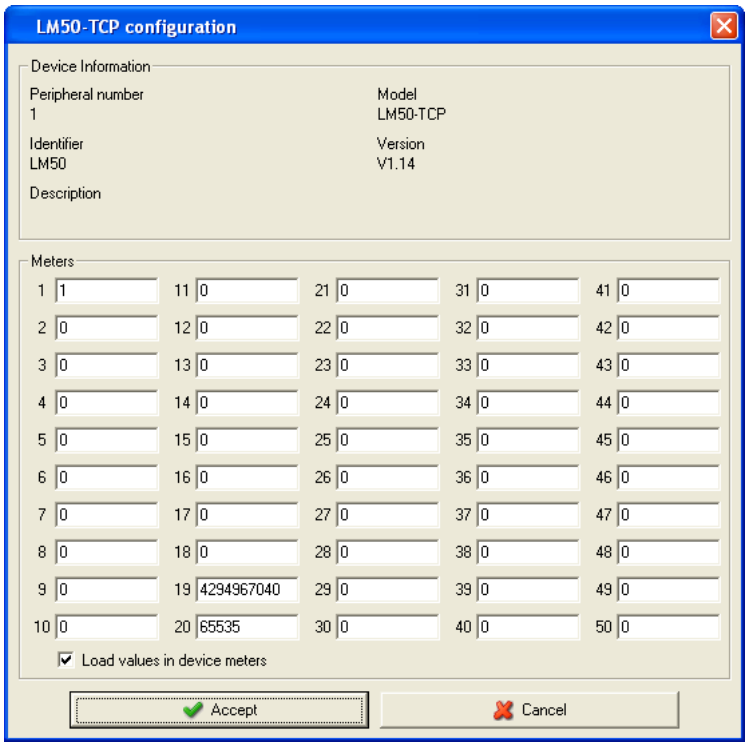
- *View Menu:* Refer to 'Editor manual'.
- *Toolbar*. Refer to 'Editor manual'.

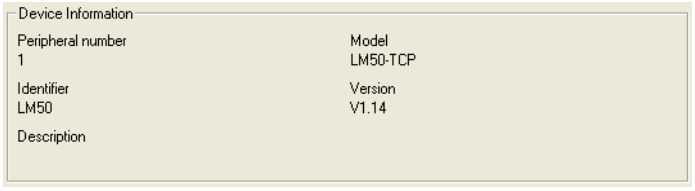
The following is the options menu:



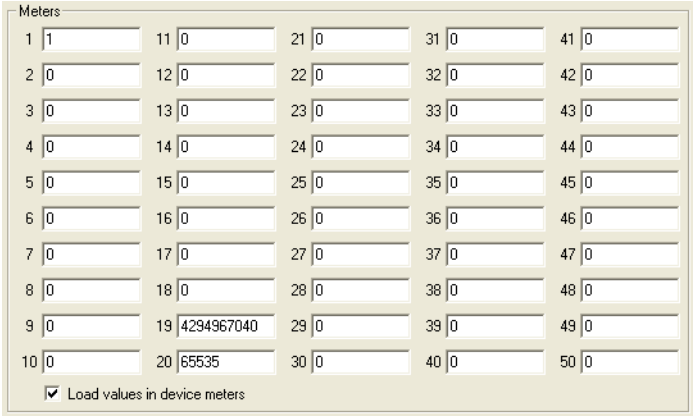
1.16.1.1 Device parameters

This screen allows the internal parameters of the device to be configured. On opening the dialogue box, the software will read the configuration of the device. When complete, click on '*Accept*' for the software to send information on changes to the device. In no case will the information be stored on the hard drive of the PC.





Show general information.

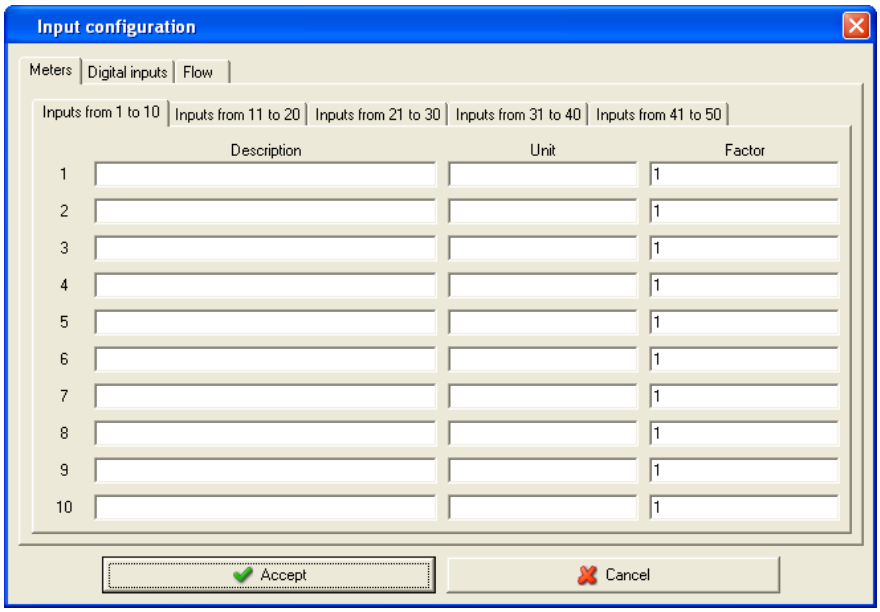


This part of the configuration corresponds to the device counters. If selector ☑ is checked, on sending the configuration the counters that have been modified will be loaded with the values entered into the corresponding boxes.

1.16.1.2 Inputs

For each of the LM50-TCP inputs three variables will be available, a digital one to show the input status (open / closed), a counter variable to show the number of pulses detected in the input and finally a flow variable calculated by the software as an estimate of the pulse frequency that occurs in the input during a period of time.

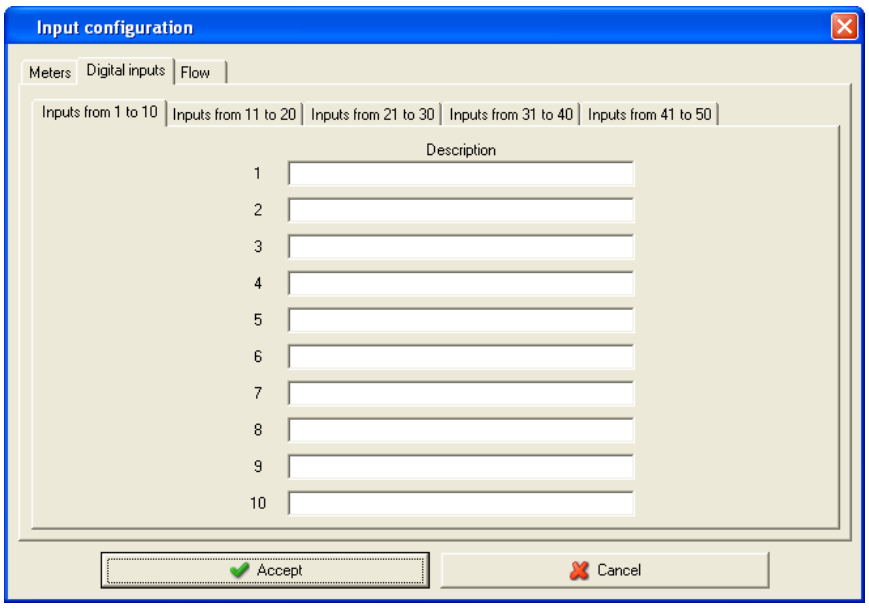
By means of this dialogue the LM50-TCP input can be configured.



Using the Contadores | Entradas digitales | Caudales display configuration of the different types of variables can be configured. The previous image shows the configuration of the counters, where:

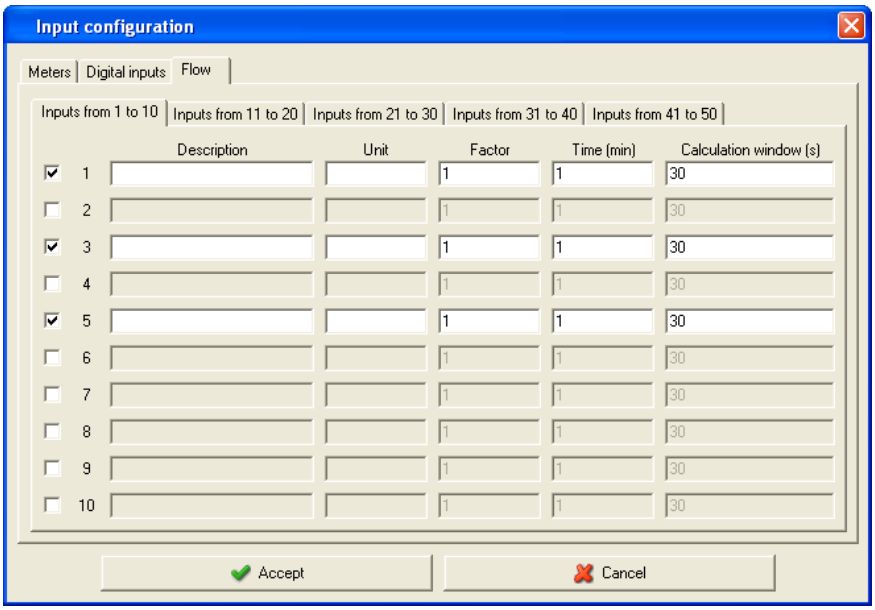
- **Description:** Alphanumeric type which permits a brief description of the counter so it can be identified better.
- **Unit:** Alphanumeric type data which permits a brief description of the units where the counter is displayed to be entered.
- **Factor:** Multiplier value of each input pulse.

For digital inputs:



- **Description:** Alphanumeric type data which permits a brief description of the digital input to be entered for better identification.

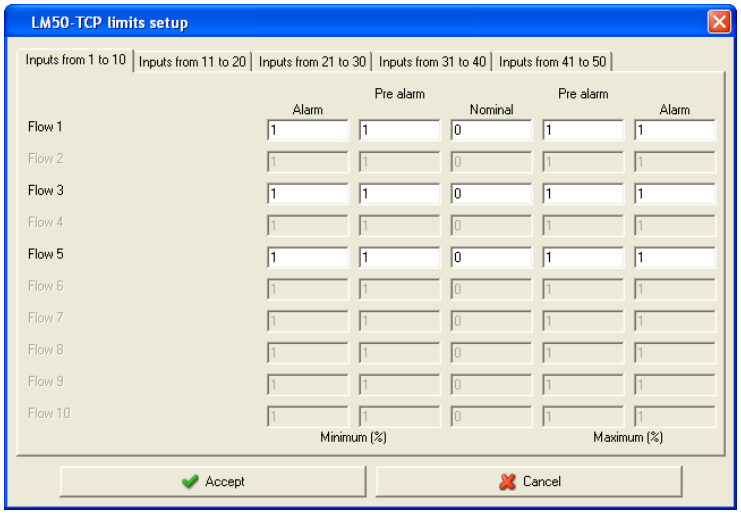
Finally for the flow configuration:



Checking the selector ☑ will activate the flow variable. Remember that if this selector is not checked it will not be possible to display the flow value of the corresponding LM50-TCP input.

- **Description:** Alphanumeric type data which permits a brief description of the flow to be entered for better identification.
- **Unit:** Alphanumeric type data which permits a brief description of the units where the flow is shown to be entered.
- **Factor:** Multiplier value of each input pulse.
- **Time:** Time in minutes for calculating the flow value.
- **Calculation window:** Time window, in seconds, that the software will use to calculate an estimate of the flow, taking into account the value of the device counter.

1.16.1.3 Variable limits



Limits can only be defined on flow type counters.

Through this dialogue the nominal value of flow type counters, as well as a series of margins to display on screen when a variable measures unusual values can be configured.

1.17 MR44-PARK

1.17.1 Driver options

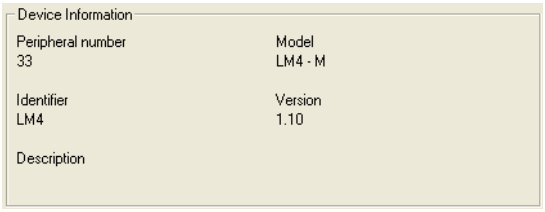
The MR44-PARK options can be accessed from:

- See Menu. Consult 'Manual Editor'.
- Tool bar. Consult 'Manual Editor'.

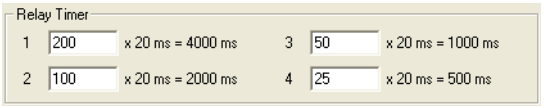
The following menu options will be available:

1.17.1.1 Device parameters

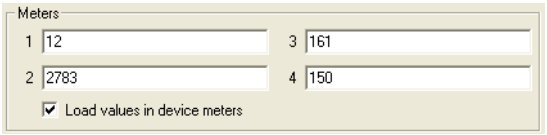
This screen allows the internal parameters of the device to be configured. On opening the dialogue box, the software will read the configuration of the device. When complete, click on "Accept" for the software to send information on changes to the device. In no case will the information be stored on the hard drive of the PC.



Show general information.



Configuration of the pulse duration can be generated with the digital outputs of the device.

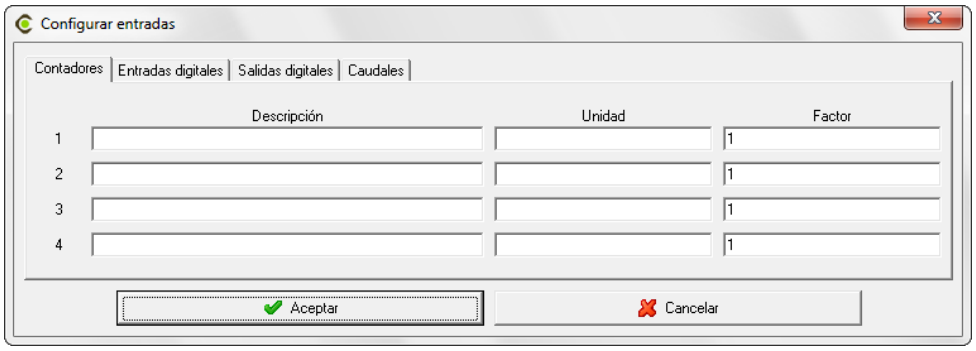


This part of the configuration corresponds to the device counters. If the selector is checked, on sending the configuration the energy counter will be loaded with the values entered into the corresponding boxes.

1.17.1.2 Device inputs

For each of the inputs of the LM4I-4O-M, three variables will be available: a digital display to show the status of the input (open / closed), a counter variable to show the number of pulses detected at the input, and finally a flow variable calculated by the software as an estimate of the pulse frequency on the input during a period of time.

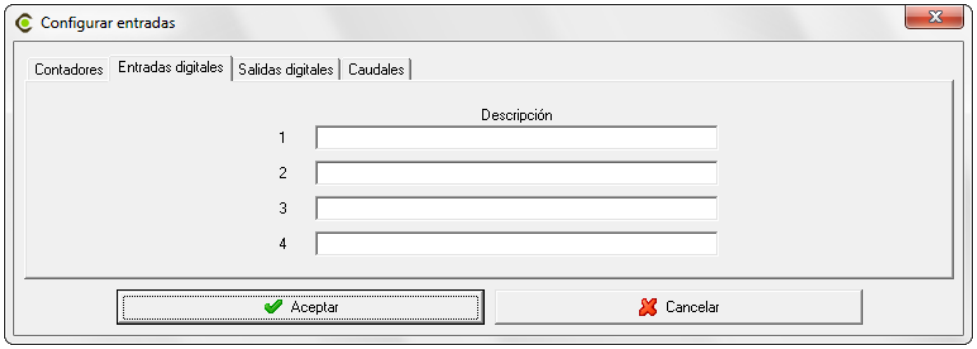
Using this dialogue the inputs and outputs of the LM4I-4O-M can be configured.



Using the  display configuration of the different types of variables can be configured. The previous image shows the configuration of the counters, where:

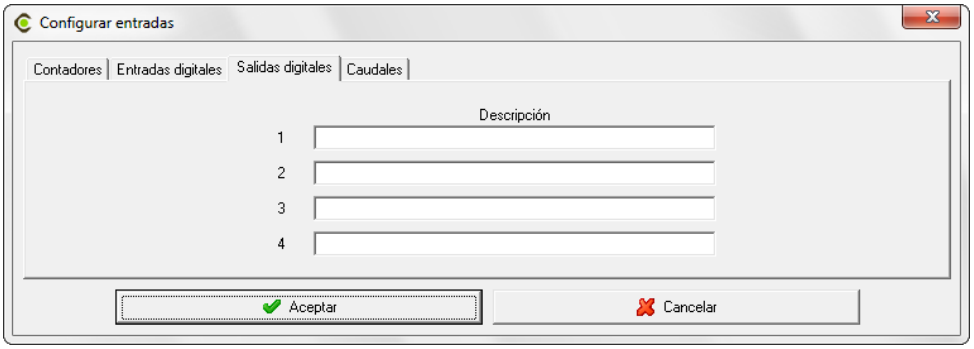
- **Description**: Alphanumeric type which permits a brief description of the counter so it can be identified better.
- **Unit**: Alphanumeric type data which permits a brief description of the units where the counter is displayed to be entered.
- **Factor**: Multiplier value of each input pulse.

For digital inputs:



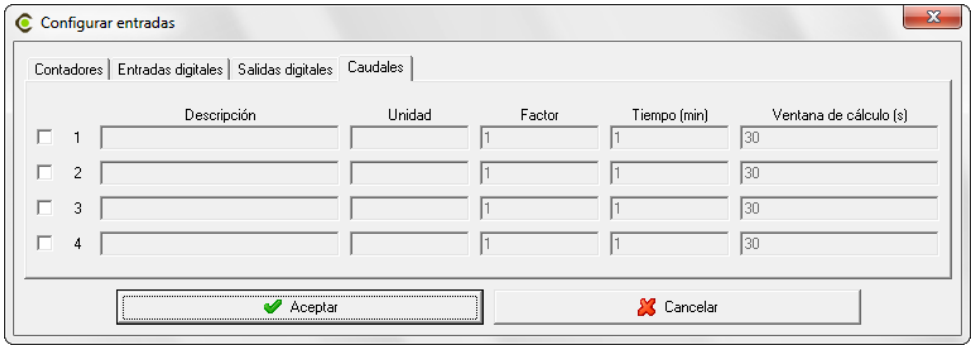
- **Description:** Alphanumeric type data which permits a brief description of the digital input to be entered for better identification.

For digital outputs:



- **Description:** Alphanumeric type data which permits a brief description of the digital output to be entered for better identification.

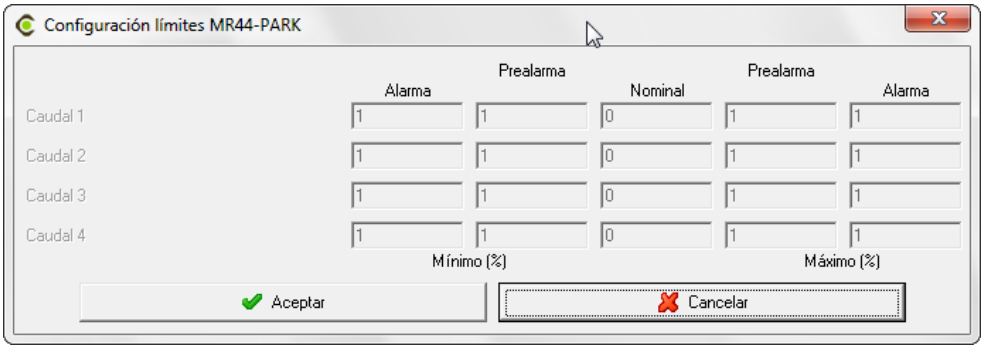
Finally for the flow configuration:



☑ Checking the selector it will activate the flow variable. Remember if this selector is not checked it will not be possible to view the flow value of the corresponding LM4I-4O-M input.

- **Description:** Alphanumeric type data which permits a brief description of the flow to be entered for better identification.
- **Unit:** Alphanumeric type data which permits a brief description of the units where the flow is shown to be entered.
- **Factor:** Multiplier value of each input pulse.
- **Time:** Time in minutes for calculating the flow value.
- **Calculation window:** Time window, in seconds, that the software will use to calculate an estimate of the flow, taking into account the value of the device counter.

1.17.1.3 Variable limits



Limits can only be defined for flows that are enabled.

Through this dialogue the nominal value of flow type counters, as well as a series of margins to display on screen when a variable measures unusual values can be configured.

1.18 IP Camera

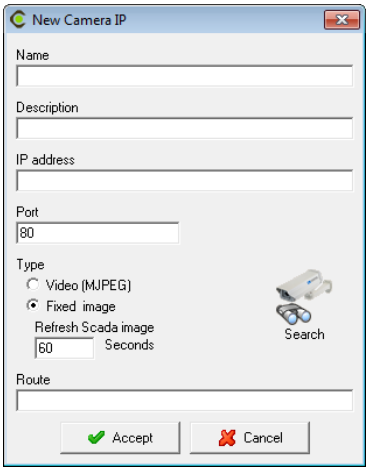
The IP Camera device only allows a captured image to be viewed, either by connecting directly with a camera with an Ethernet connection, a video recorder to which petitions can made or webcams.

It will not be possible to record, view recordings or carry out any other action on the image displayed, such as motion detection, from the sources described beforehand.

The IP Camera device can display images generated from different sources (IP cameras, videos, webcams, etc…) provided web requests can be made to these sources and the response is a still image in JPEG format or video streaming format MJPEG.

1.18.1 Add and IP camera

For more information on how to add a device see section 2.2.1 Devices The following are the different parameters to be configured to communicate with an IP camera.



- **Name:** Alphanumeric field which uniquely identifies the device throughout the program. There are no two devices in the configuration with the same name.

- **Description:** Alphanumeric type data to insert a brief description of the device.

- **IP Address** Corresponds to the address through which the program can communicate with the device. This parameter can be an IP address or a name.

- **Port:** Corresponds to the communications port:

- **Type:** Type of image returned by the device. We can choose between viewing a video in MJPEG or a still picture

- **Refresh Scada image:** Only needs to be set when it is a fixed image and it dictates how often in seconds the program should ask to obtain a new image.

- **Route:** Corresponds to the request that must be made to the device to obtain the video or image requested.

- **Search:** Pushing on this button the program will try to find the device at the address and port configured using the most frequent routes / requests.

For more information on which route or request should be used consult the device manual or contact your distributor.

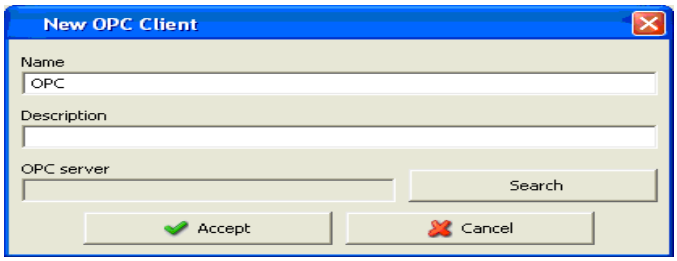
1.19 Client OPC

This Power Studio driver allows an OPC-DA client to be configured to access the variables accessible on an OPC-DA server.

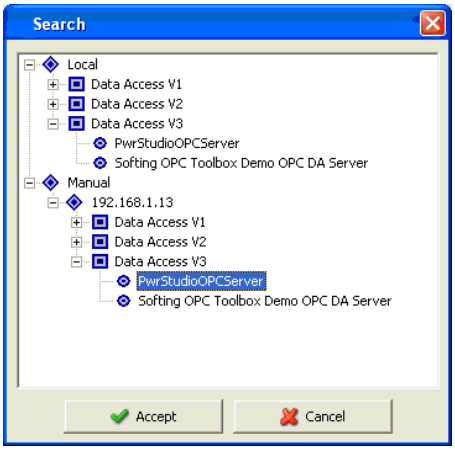
Once the variables to be read have been configured, the OPC client will behave in a manner similar to other devices with which the program communicates.

1.19.1 OPC connection configuration

To add a new OPC client the OPC server to which it has access must be indicated.



Pressing button Search brings up a dialogue to conveniently select the OPC server. We may select a local server that has previously been configured on our machine, or enter the IP address of a machine on the local network to seek all the OPC servers that it may have configured. The OPC specifications supported by the client are “Data Access V1”, “Data Access V2” and “Data Access V3”.



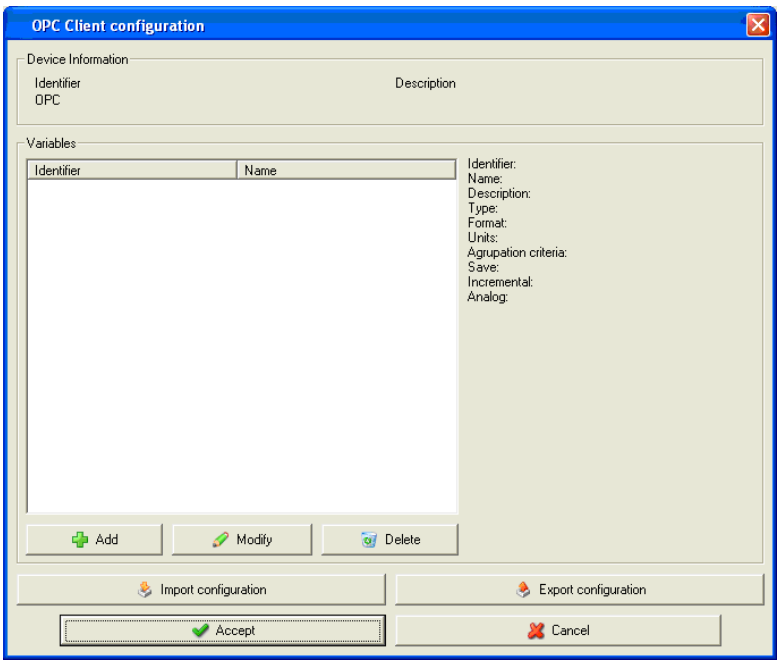
1.19.2 **Driver options**

The following menu options will be available:

The ‘Variable Units’ and ‘Limit of the variables’ are detailed in “1.3.1.3 Variables Units” and “0 Variables limits” respectively.

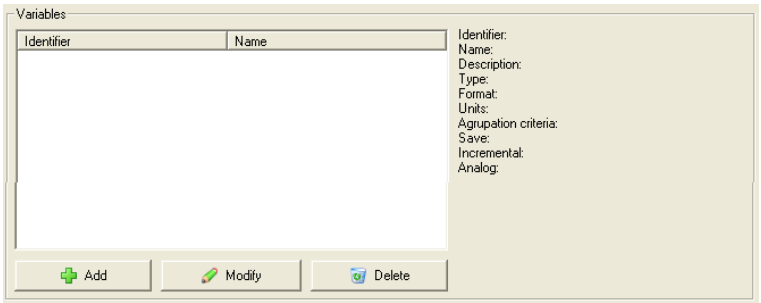
1.19.2.1 Device parameters

This screen is used to configure the variables that should be requested from the OPC server.



.

General OPC client information.



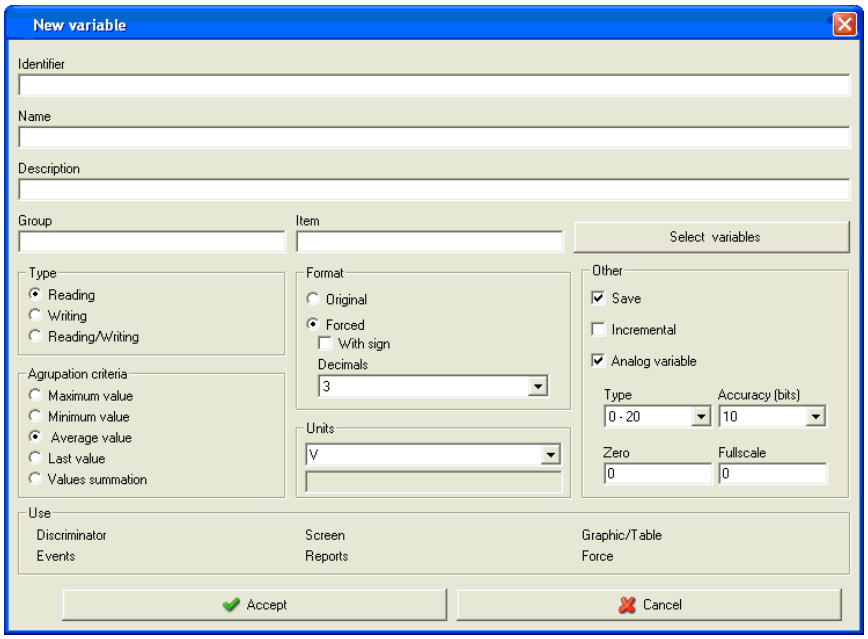
OPC client variable Configuration. See **¡Error! No se encuentra el origen de la referencia. ¡Error! No se encuentra el origen de la referencia.**.

Allow loading of a variable configuration that was previously saved to disk. This option is useful when the same variables have to be configured on several OPC clients.

Allow the variable configuration to be saved to disk, so that it can subsequently be loaded on another OPC client.

1.19.2.1.1 Variables

Pushing the “Add" or “Modify” button brings up a dialogue to configure the variable.



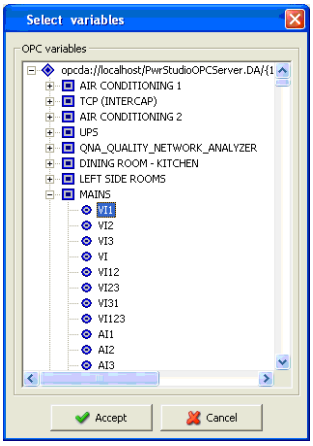
Where

- **Identifier:** Allows each of the binary variables to be used in expressions and conditions to be identified, see 'Editor Manual'.
- **Name:** Variable name, used to better identify it.
- **Description:** Short description of the variable.
- **Group:** Identifier of the group the variable belongs to in the OPC server.
- **Item:** Identifier of the item the variable belongs to in the OPC server.
- **Others:** Different variable configuration parameters
  - **Save:** Indicates whether or not the server value read should be saved in the value log.
  - **Incremental:** Indicates whether the value of the variable displayed is incremental. Typical energy variable option to see increments.
  - **Analogue variable:** Indicates whether or not the variable to be read is analogue type.
  - **Zero:** Value of the parameter to which the zero of the scale is assigned.
  - **Full-scale:** Value of the parameter to which the maximum value of the scale is assigned.
  - **Type:** Type of output, allows between 0-20 mA, 4-20 mA or 0 -? V, depending on the device.
- **Accuracy:** Number of bits in the sample.

- **Format:** Variable data format indicated.
  - Original: Based on how the variable is configured on the OPC server one conversion or another will be made.
    - VT_R4 handled as floating type value.
    - VT_UI4 handled as integer value with no symbol.
    - VT_R8 handled as double type value.
    - Any other type will display the value converted into a string.
  - Forced: Indicate whether or not the variable has a symbol and the number of decimals.
- **Type:** Selects the type of variable; i.e., read, write or both. If the variable is write or read/write the value can be forced on the device.
- **Grouping criteria:** Enables the variable values grouping criteria to be selected in graphs, tables, reports and the value to be saved in the values register. For example, if in the period between saving information we have 3 values (10, 12 and 7) these will be saved in the log register.
  - Maximum value. The maximum value of 3 (12) will be saved
  - Minimum value: The minimum value of 3 (7) will be saved.
  - Average value: The average value of 3 ((10 + 12 + 7) / 3 = 9.66) will be saved.
  - Last value: The last value will be saved (7).
  - Sum of the values: The last value will be saved (7).

  In the case of clusters of values in graphs, tables and reports the same criteria will be followed, except when it is the summation of values, where the sum of 3 values is shown (10 + 12 + 7 = 29).
- **Units:** Units in which the value is expressed. You can select one of the predefined units or define a new user unit.
- **Use:** Shows where the variable may be used in the rest of the program.
- **Select variables:** Pressing the 

button brings up the variables selection dialogue of the previously configured OPC server, and all the groups and items that can be accessed from it are displayed.



2 Appendix

2.1 Parking variables

In order to use references to the variables measured by devices, in Scada, reports, charts and tables, it will be necessary to know the encoding used by the software for each of them.

The basic encoding is made of the device name of the device and the variable code separated by one point.

name.variable

In this way the software will know which variable is involved and which device should be asked for the said variable.

This encoding (variable-name) can be used in formulas, graphs and tables; although in the case of reports, graphs and tables we can display variables saved in values log files.

On those occasions we wish to discriminate a variable, as long as the variable can be discriminated, for instance in the case of energy, the discriminator information will be added to the basic coding.

name_discriminator @ name_type_time: name.variable

Here the name of the discriminator and the name of the type of hour upon which the variable values will be discriminated are indicated.

This type of encoding with discriminators may only be used in reports, graphs and tables, and with variables that have been saved in values of log files.

NOTE:
All information of the variables of different devices, coding and remote access is available in the manual:
MU610105-[version]-[language]_Cirpark Scada_5_XML Protocol