

 

**Measuresoft Development Ltd.**

**INI And Text Manual**

**Version:** **6.4.12.0**

Table of Contents

INI Configuration File 3

Device Section 3

Example1 3

Module Section 3

Example 1 3

Example 2 3

Module Measurement Section 3

Example 1 3

Example 2 3

Example 3 3

Text Files 3

# INI Configuration File

A file called ‘AlphaConfig.ini’ will exist on Disk1 of the setup disk set. This file should be edited by distributor before shipment. It should detail the type of modules available for configuration and their properties.

**Note:**

**The Key = Value pairs defined below must not break over a single line in the INI file even if the examples do.**

The examples that follow are just examples and are not intended to define a working system.

## Device Section

The first section that must exist is the **[Device]** section. The following fields need to be completed for this section (\* indicates the field is optional).

|  |  |
| --- | --- |
| Key | Value |
| Modules | Lists modules with sections in this INI |
| RS485Baud | Provides a list of mappings between Sigma Baud codes and baudrates for RS485 |
| RS232Baud | Provides a list of mappings between Sigma Baud codes and baudrates for RS232 |
| MaxChanBlock | Specifies the maximum size of the Channel Blocks. Defaults to 48. |
| StdChanBlock | Specifies the size of the standard Channel Block. Defaults to 32.This value must be less then MaxChanBlock. **Note:** Auxiliary Channels = MaxChanBlock - StdChanBlock |

### Example1

**[Device]**

**Modules = 910, 933**

**RS232Baud = 5:38400, 4:19200, 3:9600, 2:4800, 1:2400, 0:1200**

**RS485Baud = 6:56000, 5:38400, 4:19200, 3:9600**

## Module Section

For each Module defined in the device section module list there should be a corresponding section in the INI file. The names of these sections are defined by the module list. Taking the example above we would look for the following sections **[910]** and **[933]**

The following fields need to be completed for these sections (\* indicates the field is optional).

|  |  |
| --- | --- |
| Key | Value |
| Description | Gives a brief description of the module |
| Channels | Details the type of channels supported by this module**Format :** nn:CTid-cs**nn** = Number of channels**CT** = Channel Type – AI,DI,AO,DO,CI**id** = If more than one CT of the same type exists for this module then a sequence number must be provided.**-cs** = If there is channel sharing it will be identified here. If channel pairs are being shared the value would be 2 if it every 3 channels are shared then the value would be 3 etc.**Note : id is optional, -cs is optional** |
| Auxiliary | \* Details the type of auxiliary channels supported by this module**Format :** nn:CTid **as above** |
| Alarm | \* Specifies whether the Aplha alarm scheme will be supported on this module. A value of **supported** means it is. |
| CI | \* Specifies that this module supports the CI command. Used during initialisation and reset functions. A value of **on** means it is supported. |
| CR | \* Specifies that this module supports the CR command. Used during Clear Initali Conditions. A value of **on** means it is supported. |
| EventTrace | \* Specifies that this module supports events. A value of **on** means it is supported. |
| Command0… Command6 | \* Up to six extra module commands can be configured.**Format :** ***cmd, descriptor, cntrl\_type, data\_type , DD\_list*** ***cmd*** = Two character field which gives the command string.***descriptor*** = Describes the functionality of the command. This will be displayed in the configuration window. This field should be kept to a limit of 19 characters.**cntrl\_*type*** = Specifies the control type to be used in the configuration screen: **DD** – Drop down list**EB** – Edit Box**CB** – Check box***data\_type*** = Specifies the data type to be used for the control:**UINT** – Unsigned Integer**INT** – Integer**FLOAT** – Float***DD\_list*** = This should only appear where **cntrl\_type** = DD. It describes the **value:descriptor** pairs that will appear in the drop down list. Right hand is the descriptor and the left hand is its associated sigma value. e.g. **0:50Hz , 1:60Hz** **, default:1** if the **default** option exists then the specified option will be used as the default. |

### Example 1

[910]

**description = Thermocouple Module 20 Channels**

channels = 20:AI

**auxiliary = 3:DO,1:DI**

alarm = supported

**CI = on**

**CR = on**

**command0 = CJ , Set Cold Junction Temperature , EB , FLOAT**

**command1 = FR , Select Mains Frequency , DD , UINT , 0:50Hz , 1:60Hz , default:1**

**command2 = UT , Set Temperature Units , DD , UINT , 0:Celsius , 1:Fahrenheit**

### Example 2

[933]

description=Event Input Module

channels=4:DI1,4:DI2,10:DI3

**auxiliary=3:DO,1:DI**

**alarm = supported**

**EventTrace = on**

**command0=DB,Set Debounce Time 1-200mS,EB,INT**

**command1=MP,Multiple Period Mode,EB,INT**

**command2=VR,Read Event Data,CB,INT**

**command3=VC,Clear Event Buffer,CB,INT**

**command4=VM,Set as event Master,CB,UINT**

## Module Measurement Section

For each channel range that has been described by the **channels** field in the module section there must be a corresponding section in the INI file. The name of this section depends on what has been specified by the **channels** field.

In the example above we had a 910 module with **channels = 20:AI**. Therefore we would look for section **[910:AI]**

We also had a 933 with **channels=4:DI1,4:DI2,10:DI3.** Therefore in this case we would look for sections **[933:DI1], [933:DI2] and [933:DI3]**.

If the 910 had channel sharing on e.g. **channels = 20:AI-2** we would have looked for **[910:AI-2]**

The following fields need to be completed for this section (\* indicates the field is optional).

|  |  |
| --- | --- |
| Key | Value |
| CMn,n,n,nWhere n describes a sequence variable. See examples below. | **Format :** ***descriptor, cntrl\_type, data\_type , DD\_list*** ***descriptor*** = Describes the functionality of this parameter. This will be displayed in the configuration window. This field should be kept to a limit of 19 characters.**cntrl\_*type*** = Specifies the control type to be used in the configuration screen: **DD** – Drop down list**EB** – Edit Box**CB** – Check box***data\_type*** = Specifies the data type to be used for the control:**UINT** – Unsigned Integer**INT** – Integer**FLOAT** – Float***DD\_list*** = This should only appear where **cntrl\_type** = DD. It describes the **value:descriptor** pairs that will appear in the drop down list. Right hand is the descriptor and the left hand is its associated sigma value. e.g. **0:50Hz , 1:60Hz** **, default:1** if the **default** option exists then the specified option will be used as the default.**Digital Inputs as Counters**If the channel type is a DI and “/CI” is specified as the last three characters of the descriptor string then if the user selects this option in the drop down list the channel will be treated as a counter input.**Digital Inputs as Analogs**If the channel type is a DI and “/AN” is specified the last three characters of the descriptor string then if the user selects this option in the drop down list the channel will be treated as a analog input.**Channel Sharing**If the channel block supports channel sharing and ‘/SH’ is the last three characters of the descriptor string then if the user selects this option in the drop down list this channel and the following nn channels will be shared. Note : If the channel sharing is in pairs then only every odd channel will allow this option to be displayed. Similarly for all other shared groupings. |
| Command0… Command2 | \* Up to two extra module commands can be configured.**Format :** ***cmd, descriptor, cntrl\_type, data\_type , DD\_list*** ***cmd*** = Two character field which gives the command string.***descriptor*** = Describes the functionality of the command. This will be displayed in the configuration window. This field should be kept to a limit of 19 characters.**cntrl\_*type*** = Specifies the control type to be used in the configuration screen: **DD** – Drop down list**EB** – Edit Box**CB** – Check box***data\_type*** = Specifies the data type to be used for the control:**UINT** – Unsigned Integer**INT** – Integer**FLOAT** – Float***DD\_list*** = This should only appear where **cntrl\_type** = DD. It describes the **value:descriptor** pairs that will appear in the drop down list. Right hand is the descriptor and the left hand is its associated sigma value. e.g. **0:50Hz , 1:60Hz** **, default:1** if the **default** option exists then the specified option will be used as the default. |

### Example 1

**[910:AI]**

**CMp1 = Measurement Type , DD , UINT , 0:skip, 1:Voltage, 2:Current, 3:ThermInt CJC**

**CM1,p2 = Units , DD , UINT , 0:uV , 1:mV**

**CM2,p2 = Units , DD , UINT , 0:0-20mA , 1:0-10%**

**CM3,p2 = Thermocouple Type , DD , UINT , 0:K , 1:B , 2:E , 3:J , 4:N , 5:R , 6:S , 7:T**

**CMp1,p2,p3 = Select Range , DD , UINT , 0:Auto, 1:23mV , 2:180mV , 3:1.5V , 4:10V**

**CMp1,p2,p3,p4 = Resolution , DD , UINT , 0:17bit , 1:13bit**

**command0 = CJ , Set Cold Junction Temperature , EB , INT**

**command1 = FR , Select Mains Frequency , DD , UINT , 0:50Hz , 1:60Hz , default:1**

In the above example **CMp1** identifies the display optionst be shown for the first parameter of the CM command. If the user selects **Voltage** then the configuration utility will present him with the details defined by **CM1,p2** for the second parameter . If **Current** were selected then the display options would be those defined by **CM2,p2** etc. In the case of **CMp1,p2,p3 and CMp1,p2,p3,p4** the values chosen for p1,p2 and p1,p2,p3 respectively are of no consequence for the selection of the display options.

### Example 2

**[933:DI1]**

**CMp1=Measurement Type, DD , UINT , 0:Skip , 30:Input Status , 31:Counter 16 bit/CI**

**, 32:High Freq. Meas., 34:Event**

**CM32,p2=Gate Time,DD,UINT,0:1 sec gate time,1:10 sec gate time**

**[933:DI2]**

**CMp1=Measurement Type, DD , UINT , 0:Skip , 30:Input Status , 31:Counter 16 bit/CI**

**, 32:Low Frequency Meas., 34:Event, 35:Interval Measure**

**CM32,p2=Gate Time,DD,UINT,0:1 sec gate time,1:10 sec gate time**

**[933:DI3]**

**CMp1=Measurement Type, DD , UINT , 0:Skip , 30:Input Status , 31:Counter 16 bit/CI**

**, 32:Low Frequency Meas., 34:Event**

**CM32,p2=Gate Time,DD,UINT,0:1 sec gate time,1:10 sec gate time**

In the above example the user would only be prompted for a second parameter to the CM command if **Low Frequency Meas.** has been chosen.

**Note:** If the user has chosen **Counter 16 bit/CI** then the channel would be treated as a Counter Input.

### Example 3

If we had the following section which supported channel sharing and a user were to select the **Full Bridge Strain/SH** then the channel it was selected for and the next channel would be shared. The **Full Bridge Strain/SH** option would only appear on every odd channel.

[910:AI-2]

**CMp1 = Measurement Type , DD , UINT , 0:skip, 1:Voltage, 2:Current, 14: Full Bridge Strain/SH**

**CM1,p2 = Units , DD , UINT , 0:uV , 1:mV**

**CM2,p2 = Units , DD , UINT , 0:0-20mA , 1:0-10%**

**CMp1,p2,p3 = Select Range , DD , UINT , 0:Automatic Range Selection, 1:23mV , 2:180mV**

**, 3:1.5V , 4:10V**

**CMp1,p2,p3,p4 = Resolution , DD , UINT , 0:17bit , 1:13bit**

**command0 = CJ , Set Cold Junction Temperature , EB , INT**

**command1 = FR , Select Mains Frequency , DD , UINT , 0:50Hz , 1:60Hz , default:1**

# Text Files

A number of text files have been provided which allow for extra commnds to be sent to the Sigma Device at specific times during the running of the Sigma Driver. These files are:

Open.txt Used when device enabled

Config,txt Used when device configured

Initialise.txt Used when analog input channels initialised

Reset.txt Used when counter input channels reset

Close.txt Used when device closes

The format of these files is as follows:

***resp\_required, timeout, command\_string***

***resp\_required*** – This should be set to 1 if a response is expected for this command string and 0 if not.

***timeout*** – Timeout in milliseconds. This can be provided even if no response is required.

***command\_string*** – The command string that you require to send to the sigma device. No STX, ETX or checksum is required.