A1330 Samples Programmer User Manual

A guide to using the Allegro A1330 Samples Programmer

By Robert Bate
Allegro MicroSystems, LLC

Table of Contents

Welcome

Welcome to the Allegro A1330 Samples Programmer.

The A1330 is a 360° angle sensor IC that provides contactless high resolution angular position information based on magnetic Circular Vertical Hall (CVH) technology. It has a system-on- chip (SoC) architecture that includes: a CVH front end, digital signal processing, and an analog output driver. It also includes on-chip EEPROM technology, capable of supporting up to 100 read/write cycles, for flexible end-of-line programming of calibration parameters. Broken ground wire detection and user selectable output voltage clamps make the A1330 ideal for high reliability applications requiring 0° to 360° angle measurements.

The A1330 supports a Low RPM mode for slower rate applications and a High RPM mode for high speed applications. High RPM mode is for applications that require higher refresh rates to minimize error due to latency. Low RPM mode is for applications that require higher resolution operating at lower angular velocities.

With programmable angle scaling the A1330 supports applications requiring short angular displacements, while maintaining full dynamic range on the output. Programmable min/max angle thresholds allow diagnosis of mechanical failures.

Version

2019-07-09 Version 2.4

Short Stroke Enhancements

Support

Support on our website: https://registration.allegromicro.com/

We would like to make this the best A1330 programmer possible. Your input is valuable to us. Please, do not hesitate to contact us with problems or suggestions.

Hardware Requirements

To run this application you need one of the following:

An ASEK-1330-T-KIT which contains:

- ASEK-20-T-KIT
  - ASEK-20 Chassis with main Motherboard inside (85-0540-004)
  - USB Communications Cable
  - DC Power Supply/Cable with AC Outlet Adapters
  - Proto Board (Part # 85-0540-103) (Not required with A1330 communication)

    - Ribbon Cable (Part # 85-0540-300)
- ASEK-1330-SUBKIT-T
    - A1330 Daughterboard (Part #: TED-0002219 / Stenciled A1330)
    - A1330 Socketed grand-daughter board (Part #: 85-0808-002 / Stenciled ASEK1330-SMT)
    - A1330 Surface mount grand-daughter board (Part #: 85-0808-004)

Or

An ASEK21-T-KIT which contains:

- ASEK-21 Main Board(Part #: 85-0712)
- ASEK-21 Calibration Board(Part #: 85-0712-100)
- ASEK-21 Daughter Board(Part #: 85-0712-103)
- ASEK-21 Cable(Part #: 85-0712-301)
- USB A male to B male 3-foot cable (Part#: 3021001-03)
- ASEK-21 Power Supply (Part #: EMMA050400-P5P-IC)

ASEK Setup

Setting up ASEK-20 or ASEK-21

Connecting the ASEK-20 to the PC

1. Connect one end of the USB communications cable to a personal computer
2. Connect the other end of the USB communications cable to the “USB” port on the ASEK-20 chassis.
3. Connect the ribbon cable to the J2 connector on the daughterboard (TED-0002219)
4. Connect the other end of the ribbon cable to the “Device Connection” port on the ASEK-20 chassis
5. Connect the DC Power Supply/Cable to the 5V port on the ASEK-20 chassis
6. Plug in the DC Power Supply to a 110V/220AC 60/50Hz outlet with the proper adapter

Figure 1: ASEK-20 Setup

Connecting the ASEK-21 to the PC

1. Connect one end of the USB communications cable to a personal computer
2. Connect the other end of the USB communications cable to the “USB” port on the ASEK-21 chassis.
3. Connect the cable from the module to the “Device Connection” port on the ASEK-21 chassis
4. Connect the DC Power Supply/Cable to the 5V port on the ASEK-21 chassis
5. Plug in the DC Power Supply to a 110V/220AC 60/50Hz outlet with the proper adapter

Figure 2: ASEK-21 Setup

Setting Up Hardware Communication

Installing the Software

Place the zip file where you want the software to be run. Unpack the zip file. A folder called **Allegro A1330 Samples Programmer** should have been created. All the files need to run should be in the folder. The application is the file called **Allegro A1330 Samples Programmer.exe** in this folder.

Starting the Application

1. Double click on the application icon. The main window will appear (see Figure 3)

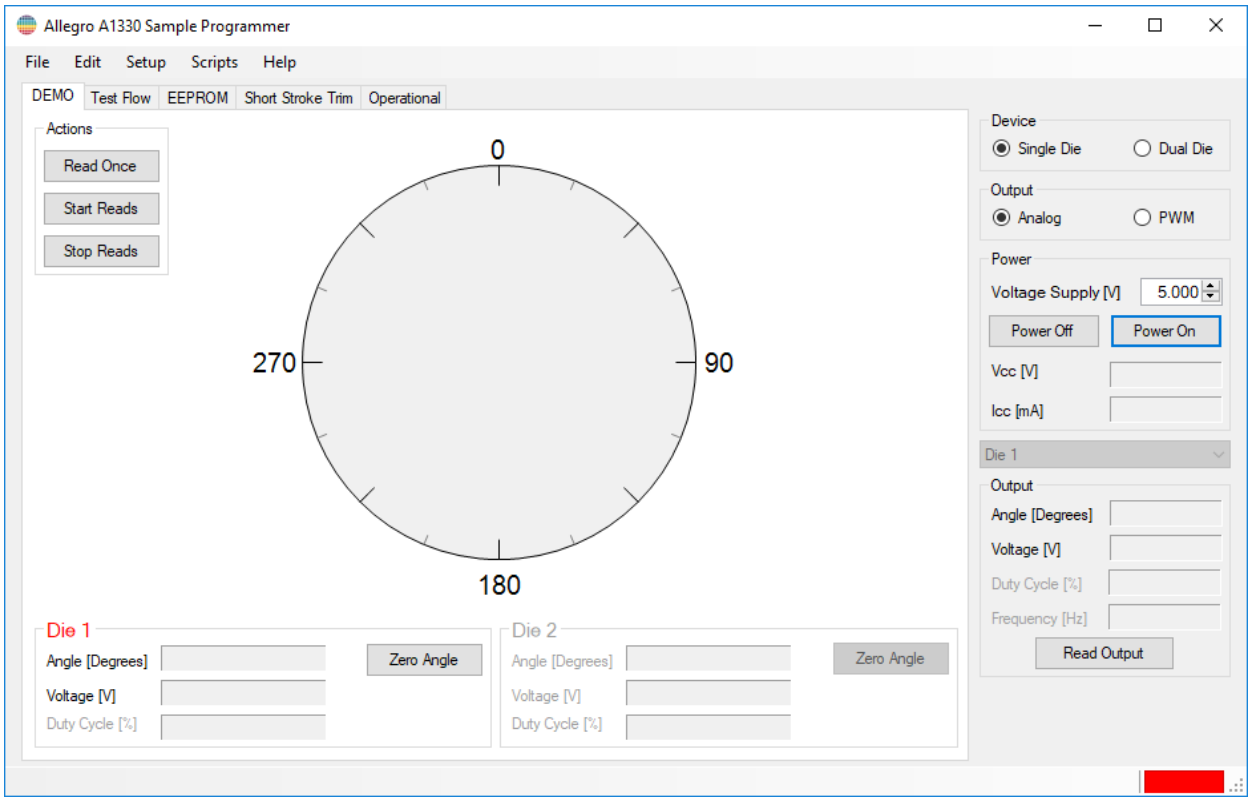


Figure 3: Main Window

2. The application is running but is not connected with the ASEK-20. To connect to the ASEK-20, click on Setup in the menu bar.

3. On the Setup pull down menu, select **Communications Setup...**. The Communications Setup dialog box will appear (see Figure 4)

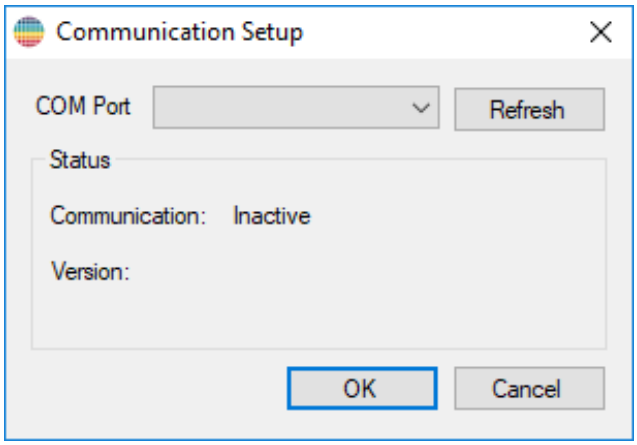


Figure 4: Communication Setup Dialog Box

4. Click the COM port pull down and select the COM port that is the ASEK-20. If you do not know which COM port the ASEK-20 is connected to, then do the following:

   i. Unplug the USB cable to the ASEK-20.

   ii. Click **Refresh** at the top right of the communication setup dialog box.

   iii. Click on the **COM Port** pull down menu and note which ports are in the menu.

   iv. Plug the USB cable back into the ASEK-20.

   v. Click **Refresh** again.

   vi. Click the **COM Port** popup menu again.

   vii. The new COM port in the menu is the ASEK-20. Click the COM port for the ASEK-20.

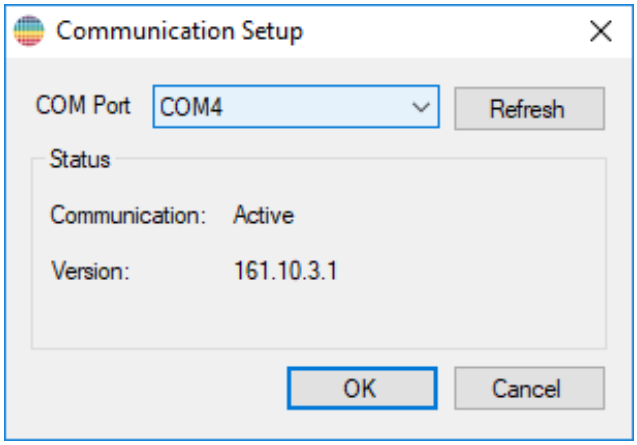


Figure 5: ASEK-20 Connected

5. If the ASEK-20 is turned on, the Status field on the Communication Setup dialog box will change from Inactive to Active, and the firmware version will be displayed (see Figure 5). If the ASEK-20 is not on, do the following:

    a. Turn the ASEK-20 on.

    b. Click **Refresh** at the top right of the Communications Setup dialog box. The Communication line in the Status field should change to Active.

6. Click **OK**. The com port will be saved and used the next time the application is started.

Status Bar

The Green or Red colored rectangle on the right side of the status bar indicates the status of the communication with the ASEK. If Red, the communication is not active and if green the application is communicating with the ASEK. The COM port that is currently set is overlaid on the colored rectangle. Clicking on the rectangle will open the Communication Setup dialog.

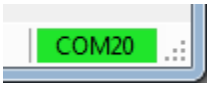


Figure 6: Status Bar with ASEK-20

Troubleshooting

If the driver to communicate to the ASEK-20 or ASEK-21 was not loaded automatically, a copy of the driver is available at http://www.ftdichip.com/Drivers/VCP.htm

Device Setup

Connecting the Device to the ASEK-20 or ASEK-21

Connecting the Device to the ASK-20 with the Grand-Daughterboard

1. Insert a device into the socket on the 85-0808.
2. Mount the socketed grand-daughterboard (85-0808) onto the daughter board (TED-0002219)

Figure 7: ASEK-20 with grand-daughterboard installed

Connecting the Device to the ASK-20 without the Grand-Daughterboard

When connecting the A1330 to the TED-0002219 without the using the grand-daughterboard (85-0808), there are a number of important pins that need to be connected in order to read/write EEPROM correctly. J3 is the while Molex connector next to the test points and pin 1 on the connector is the pin that is the furthest from the ribbon cable connector.

1. V_{cc_1} (and V_{cc_2} if using a dual die part) on the A1330 should be connected to Pin 1 on J3 or TPVCC.
2. V_{OUT_1} on the A1330 should be connected to Pin 3 on J3 or TPOUT1.
3. V_{OUT_2} on the A1330 should be connected to Pin 2 on J3 or TPOUT2 if using a dual die part.
4. All of the GND pins on the A1330 should be connected to Pin 4 on J3 or TPGND1
5. A jumper wire should be placed between TPGNDS and TPGND1.
6. Depending on the length of wire between the ASEK-20 and the A1330, it is recommended that you reduce to Manchester Speed.

Connecting the Device to the ASEK-21

When connecting the A1330 to the ASEK-21 the following guidelines must be used.

1. The VCC_F and VCC_S lines need to be connected together. If using the stable or accurate feedback power option, then the lines can be shorted together as close to the connecter as desired, otherwise when using the external feedback option the VCC_S line needs to come from the module and connect to the VCC_F line there.
2. The VOUT_F and VOUT_S lines need to be connected together either at the connecter or the module.
3. The GND_F and GND_S lines need to be connected together either at the connecter or the module.

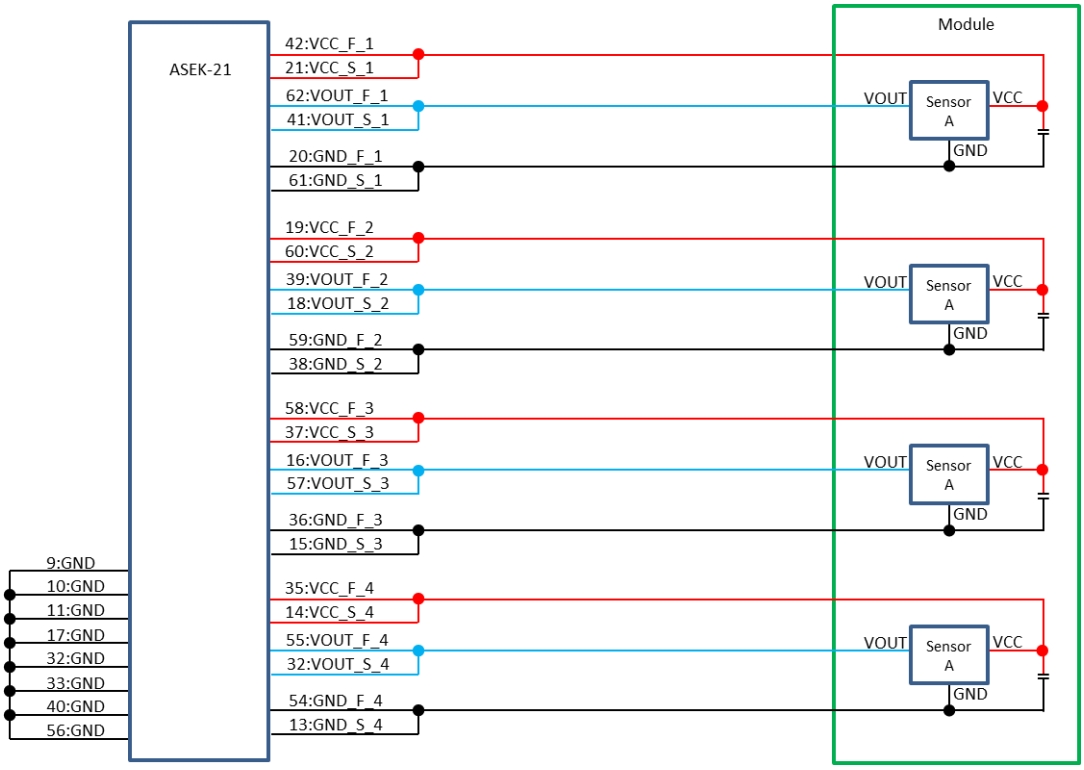


Figure 8: Connecting 4 Devices to ASEK-21

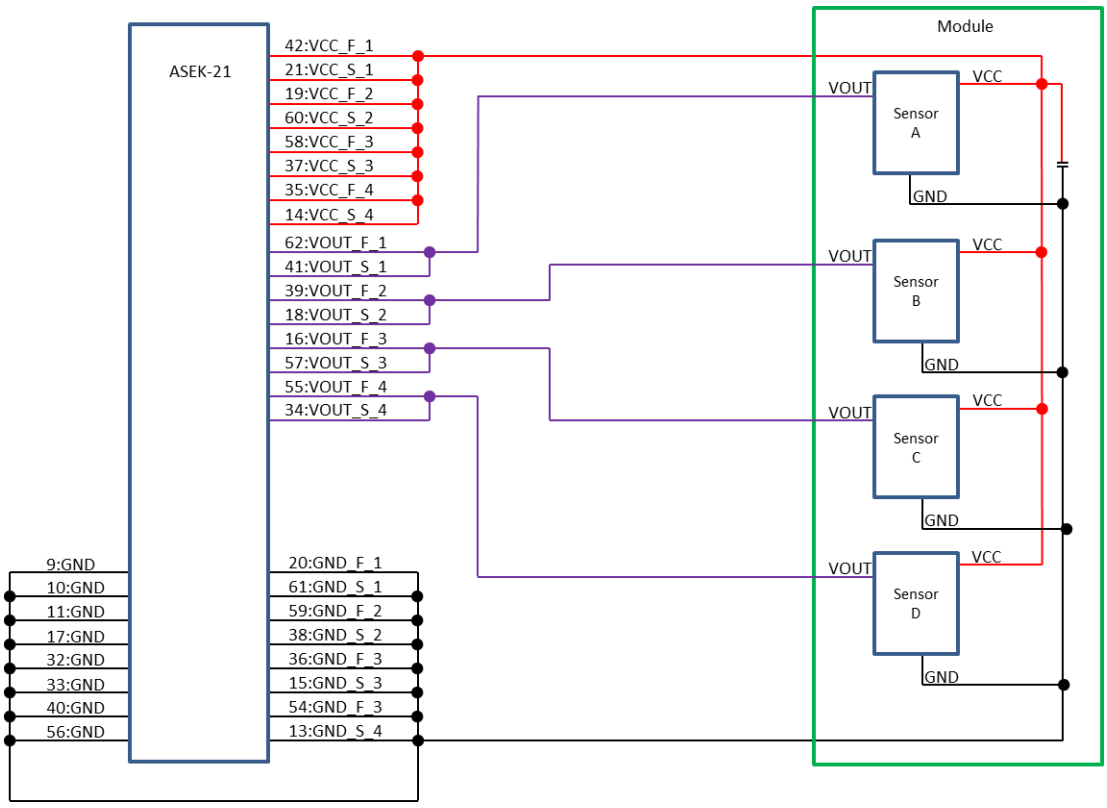


Figure 9: Connecting 4 Devices to ASEK-21, Common Vcc

Setting Up Communications

To set the various options when communicating with the device that are to be used, click **Device Setup…** from the Setup pull-down menu.

Manchester Protocol Options

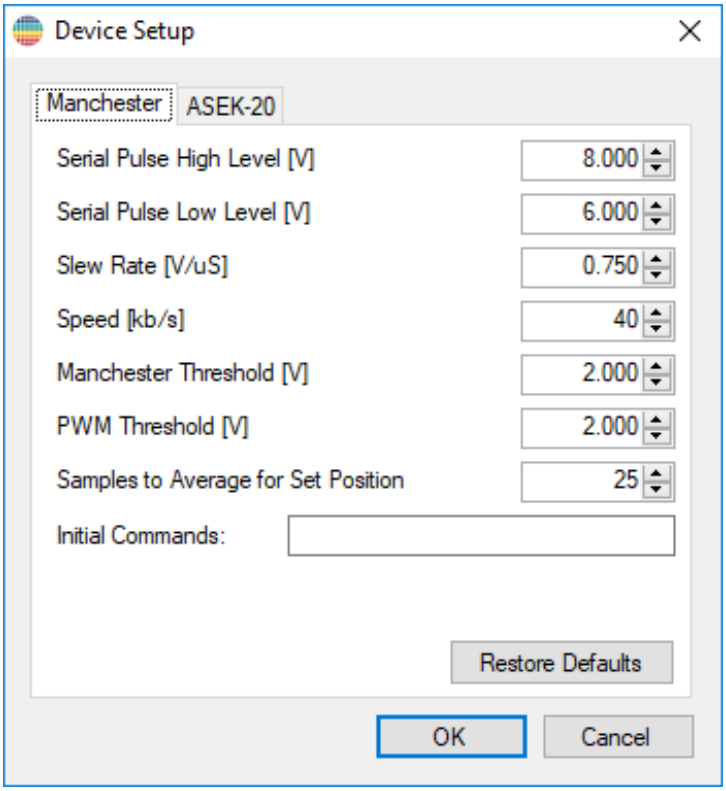


Figure 10: Device Setup of Manchester Protocol Options

Serial Pulse High Level [V] Text Entry Box

This is the voltage that will be used for the high level of the Manchester signal.

Serial Pulse Low Level [V] Text Entry Box

This is the voltage that will be used for the low level of the Manchester signal.

Slew Rate [V/µS] Text Entry Box

Used to set the speed at which the Manchester signal will take to get from one voltage to another. The default is 0.75 Volts/µS.

Speed [kb/s] Text Entry Box

This is the bit rate that will be used when communicating with the A1330.

Manchester Threshold [V] Text Entry Box

This is the voltage that will be used to determine if the input is true or false when reading the Manchester signals.

PWM Threshold [V] Text Entry Box

Used to set the threshold for determining the difference between a 1 and a 0 when performing a PWM read.

Initial Command Text Entry Box

This is used for commands that need to be sent to the ASEK-20/21 when it is initialized.

Restore Defaults

When clicked, all of the default values for the Manchester settings are restored.

ASEK-20 Options

When the programmer is connected to an ASEK-20, this tab is shown.

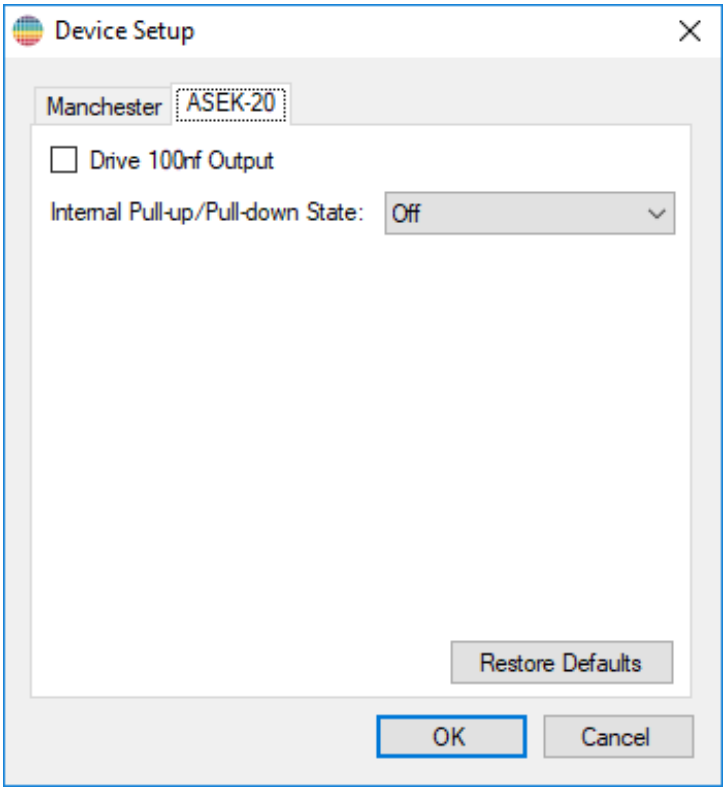


Figure 11: Device Setup with ASEK-20

Drive 100nf Output

If the Vcc line has about 100nF of capacitance then this option is used to control the overshoot.

Internal Pull-up/Pull-down State

The TED-002219 daughterboard has pull-up and pull-down resisters on the board and this option is used to control then.

- Off – Do not connect the pull-up or pull-down resisters to the output lines.
- Pull-up – Connect 4.75k ohm pull-up resisters from OUT to 5 volts.
- Pull-down – Connect 10k ohm pull-down resisters from OUT to ground.

ASEK-21 Setup Options

When the programmer is connected to an ASEK-21, these tabs are shown.

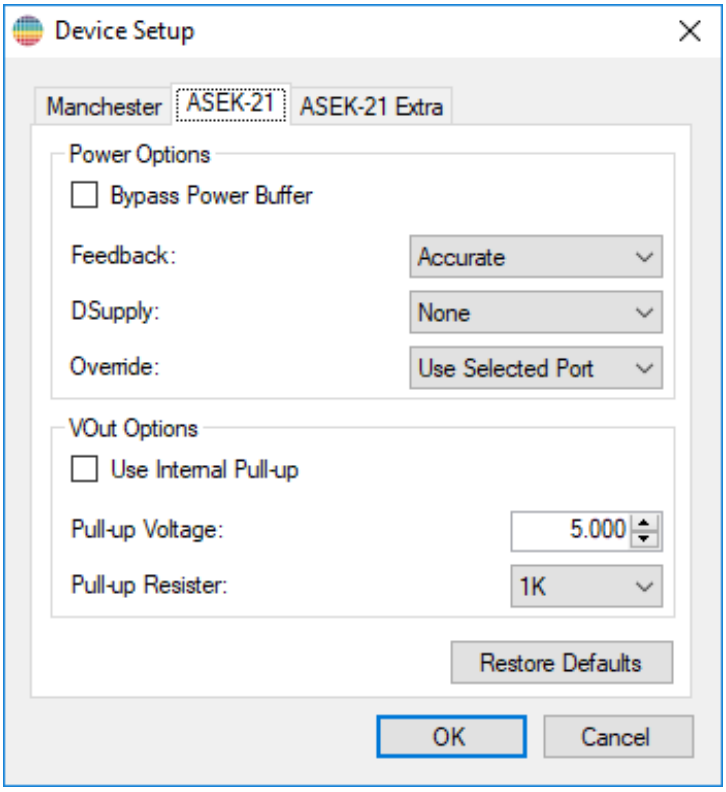


Figure 12: Device Setup with ASEK-21

Bypass Power Buffer Check Box

By default, the power supplied by the ASEK-20 is buffered internally by the ASEK-21. This allows higher current supply. This check box allows that buffer to be bypassed and the power to come directly from the ASEK-20. When checked, the power is limited to what the ASEK-20 can supply (about 40 mA).

Feedback Pull Down Menu

When the power is buffered by the ASEK-21, one of three feedback loops can be used. The first, **Stable** is the most stable but the voltage will decrease with the current draw (about 10mV per mA). The second is **Accurate** which is the default. The third is **External** which uses the VCC_S from the module under test. This is the most accurate but the least stable.

DSupply Pull Down Menu

The DSupply pin can supply power separate from the VCC lines. It can only be one of four values and they are:

- No Voltage Supplied
- 5 Volts
- 3.3 Volts
- 2.7 Volts

Override Pull Down Menu

The Override is used to select which supply power and ground lines to use and can be separate from the VOut lines. It can only be one of five values and they are:

- Use Selected Port – Always use the power and ground lines specified by the port control.
- Use Port 1 – Always use the power and ground lines on port 1
- Use Port 2 – Always use the power and ground lines on port 2
- Use Port 3 – Always use the power and ground lines on port 3
- Use Port 4 – Always use the power and ground lines on port 4

Use Internal Pull-up Check Box

When checked, the VOut line is pulled up by the designated resister by the chosen voltage.

Pull-up Voltage Text Entry Box

Selects the voltage that VOut will be pulled up to.

Pull-up Resister Pull Down Menu

Selects either the 1K or 4.75K pull-up resister.

Restore Defaults

When clicked, the default values for the ASEK-21 settings are restored.

ASEK-21 Extra Setup Options

When the programmer is connected to an ASEK-21, this tab is also shown.

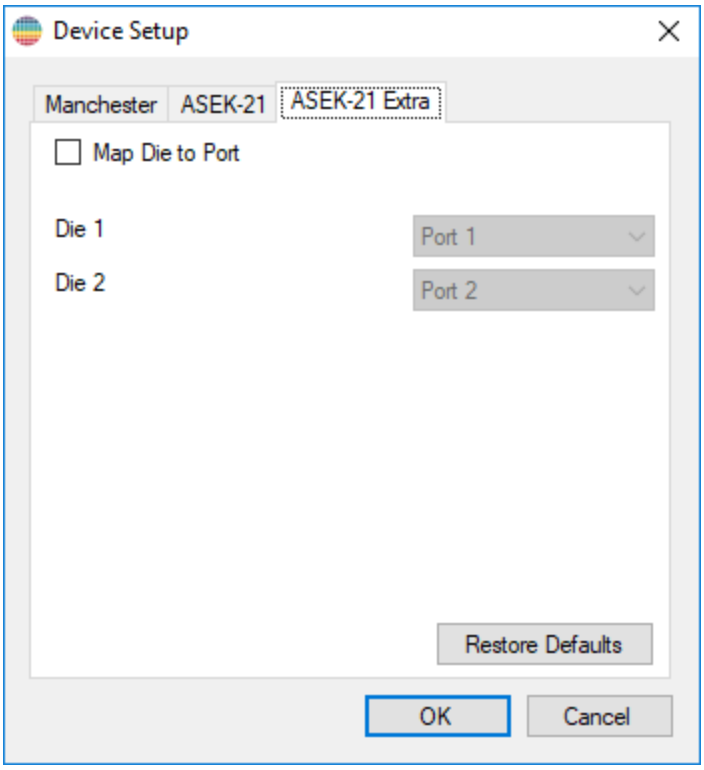


Figure 13: Device Setup with ASEK-21 Extra

Map Die to Port Check Box

By default, when using the ASEK-21 the **Die Selection** menu is disabled, and port selection is performed using the popup menu in the status bar at the bottom of the window. When this control is checked, the port selection menu is hidden, the Die Selection is enabled, and when the die is selected from the die selection menu the port is selected from the lower choices. The default is unchecked.

Die 1 Pull Down Menu

When **Map Die to Port** is checked and Die 1 is selected in the **Die Selection** menu, the port is set to one of the choices selected. It can only be one of four values and they are:

- Port 1 (default)
- Port 2
- Port 3
- Port 3

Die 2 Pull Down Menu

When **Map Die to Port** is checked and Die 2 is selected in the **Die Selection** menu, the port is set to one of the choices selected. It can only be one of four values and they are:

- Port 1

- Port 2 (default)
- Port 3
- Port 3

Restore Defaults

When clicked, the default values for the ASEK-21 extra settings are restored.

Turning the Device On/Off and Reading the Output

1. Click **Angle** for an A1330 part setup for analog output and **PWM** for a part setup for PWM output.
2. Click **Power On.**
3. The Vcc [V] and Icc [mA] fields in the right hand side of the main window will be updated with the measured values. Verify that the voltage is what is desired and that the device is consuming approximately 10 mA. The Icc is an un-calibrated reading and should be used as a qualitative indicator.
4. Click **Power Off** to turn power off to the device. Icc will fall to 0.
5. Click **Power On** then click **Read Output**. All of the angle, voltage, duty cycle and frequency fields will be updated.
6. Clicking **Read Once** will also update all of the angle, voltage, duty cycle and frequency fields.
7. Clicking **Start Read** will continuously update all of the angle, voltage, duty cycle and frequency fields until **Stop Reads** is clicked.

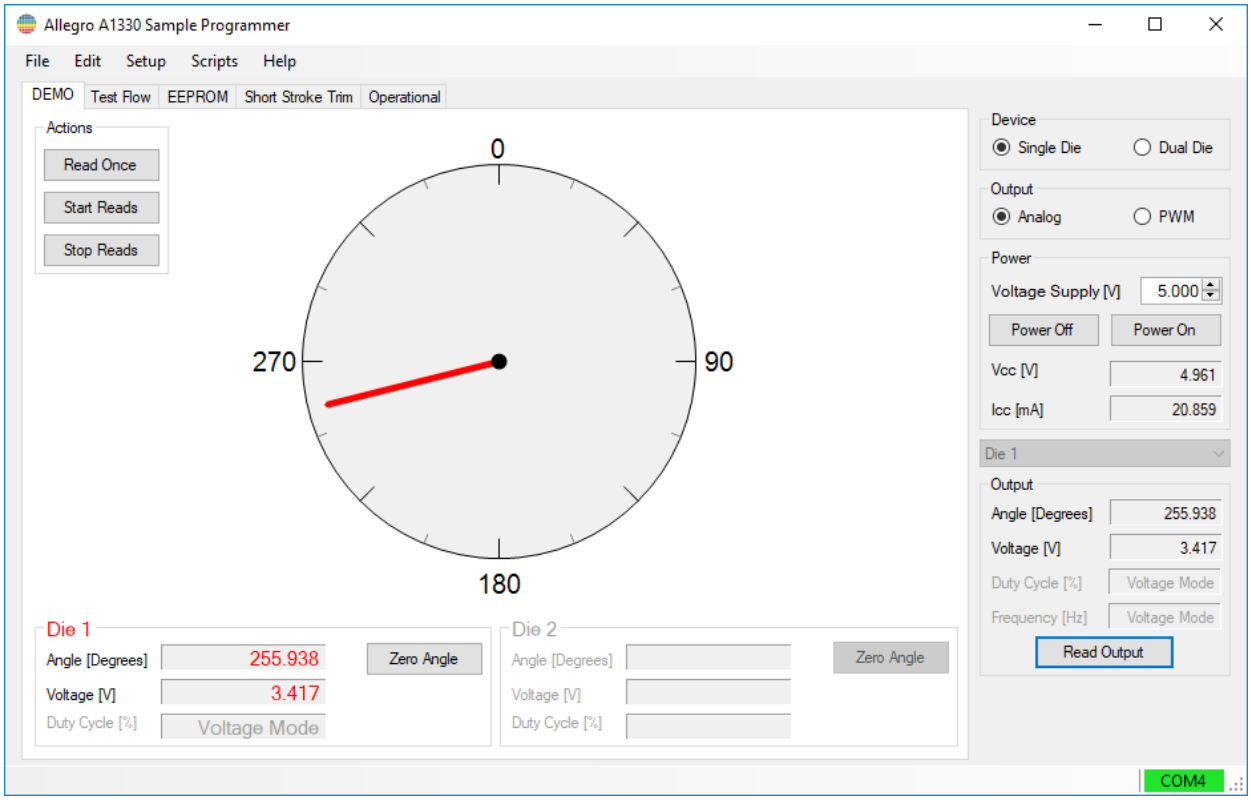


Figure 14: Powered On and Updated

Reading from the device

Reading a Field

1. On the Main Window, click on the **EEPROM** Tab. The **Show**: popup should be showing **All Fields**.
2. Scroll down the table until you see **Customer word**.
3. Click on the checkbox that is next to **Customer word**.
4. Click on the **Read Selected**, The EEPROM tab will appear similar to Figure 15.

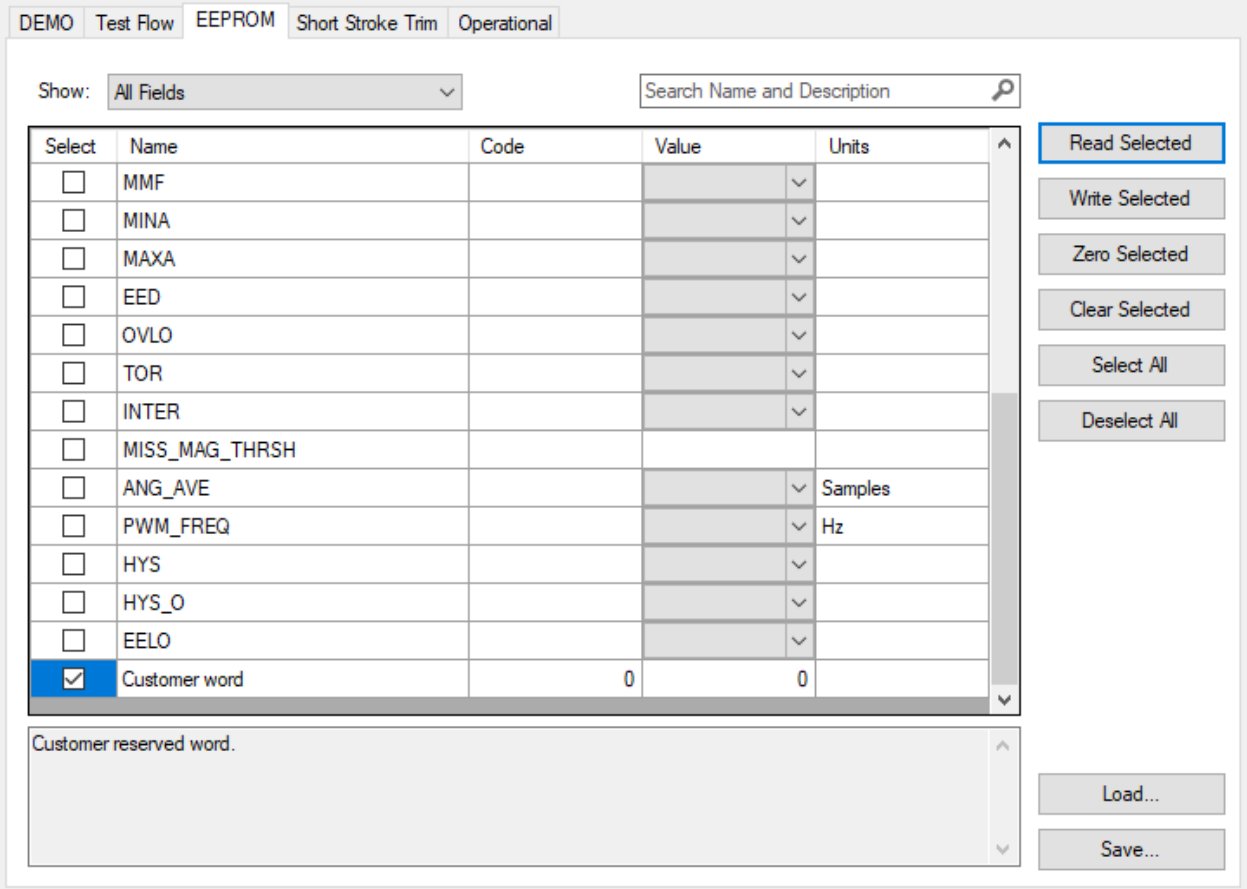


Figure 15: Reading a Field

Reading a Memory Location

1. On the EEPROM tab, click the **Show:** pull down menu.
2. On the **Show**: pull down menu, click **All Memory Locations**.
3. Click on the checkbox that is next to **0x3F**.
4. Click **Read Selected**. The window should appear as it does in Figure 16.

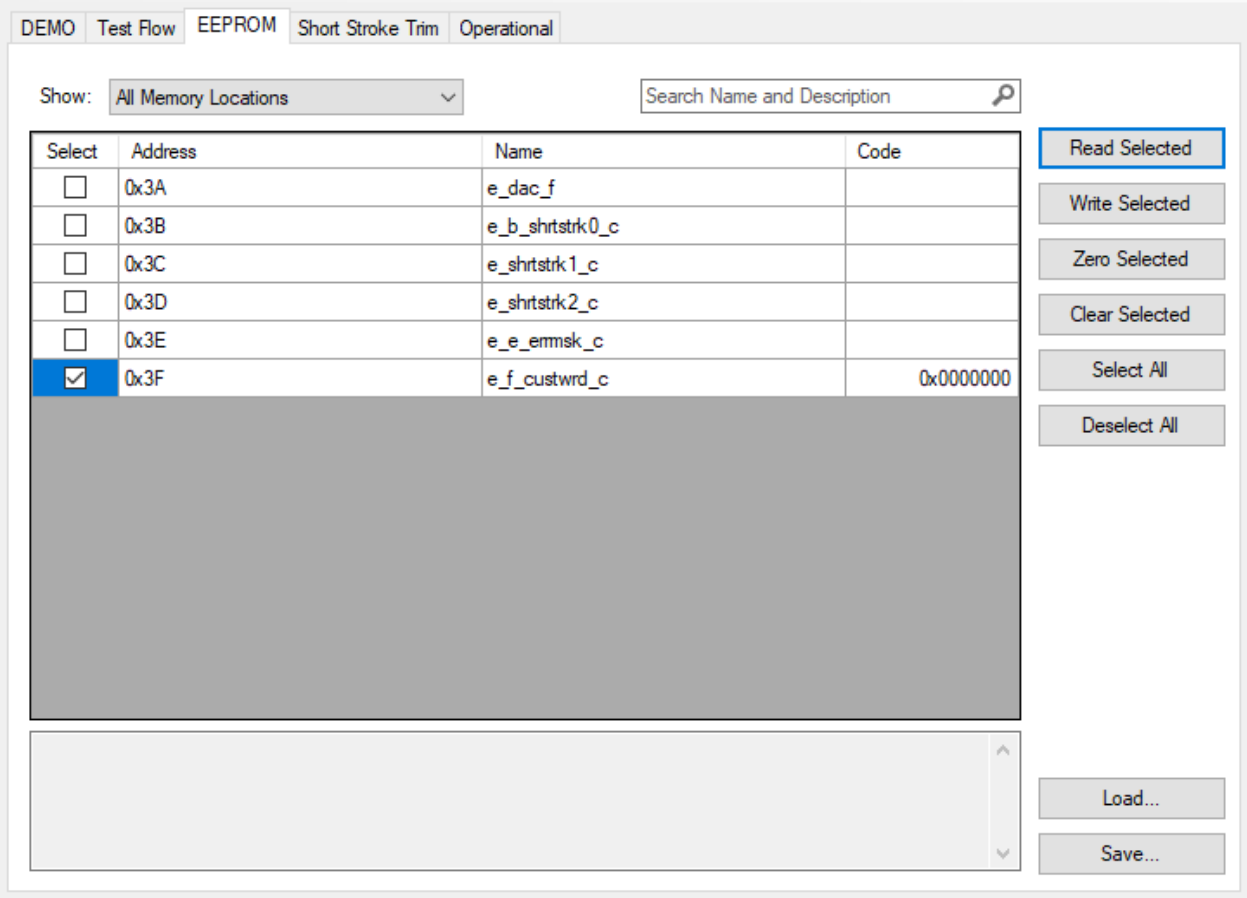


Figure 16: Reading a Memory Location

5. To verify that the reading a field and reading a memory location are reading the same location, look at **Customer word**. If you look at location 0x3F the value in the field and the value in the memory location are the same (in this case 0x0).

Writing to the device

Writing an Field

1. Click the **EEPROM** Tab.
2. Select **All Fields** from the **Show:** pull down menu.
3. Scroll down the table to the **Customer word** row.
4. Double click in the text entry box in the **Code** column of the **word** row.
5. Type **291** and press **Enter**. The Selected checkbox will be checked and the cell in the Value column will be set to 291.
6. Click **Write Selected**. The window should appear as it does in Figure 17.

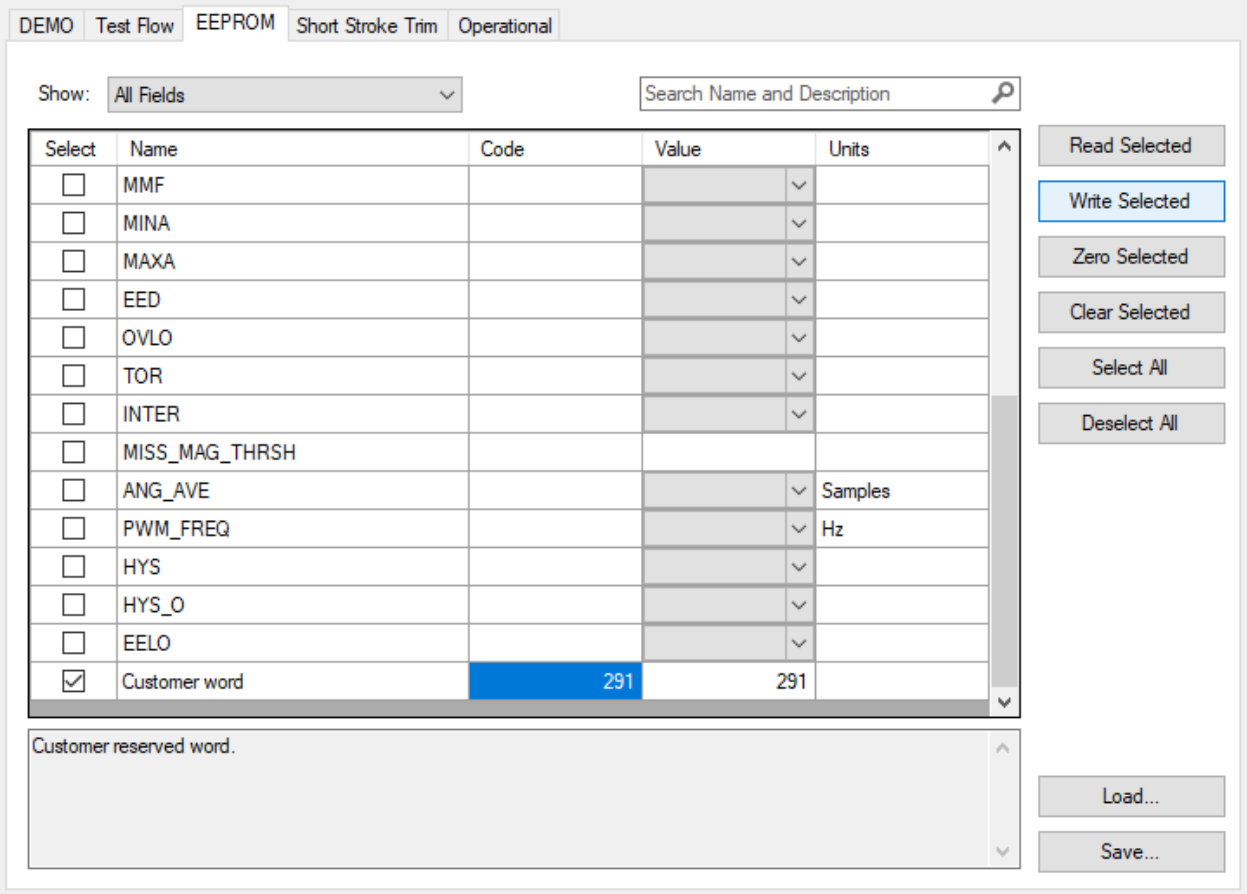


Figure 17: Writing a Field

7. To verify that the **Customer word** field was written to the device, do the following:
    a. Click **Clear Selected**. The values in the Code and Value cells should disappear.
    b. Click **Read Selected**. The values that were written will reappear in the Code and Value cells.

Writing a Memory Location

Writing to a Memory Location is done the same way.

1. From the **Show:** pull down menu, select **All Memory Locations**.
2. To ensure that there is not any memory locations selected, click **Deselect All**.
3. Scroll down the table and click the checkbox in the **Select** column next to **0x3F**.
4. Click **Read Selected**. The value in the Code column should be something like 0x000123.
5. Change the 0x000123 to 0x000321 by double-clicking in the field, typing **0x000321**, and pressing **Enter**.
6. Click **Write Selected**. The memory location is now changes to 0x000321.

7. To verify that the memory location has been changed, select **All Fields** from the **Show:** pull down menu and scroll down to the **Customer word** row. It should now read 801.

Die Selection

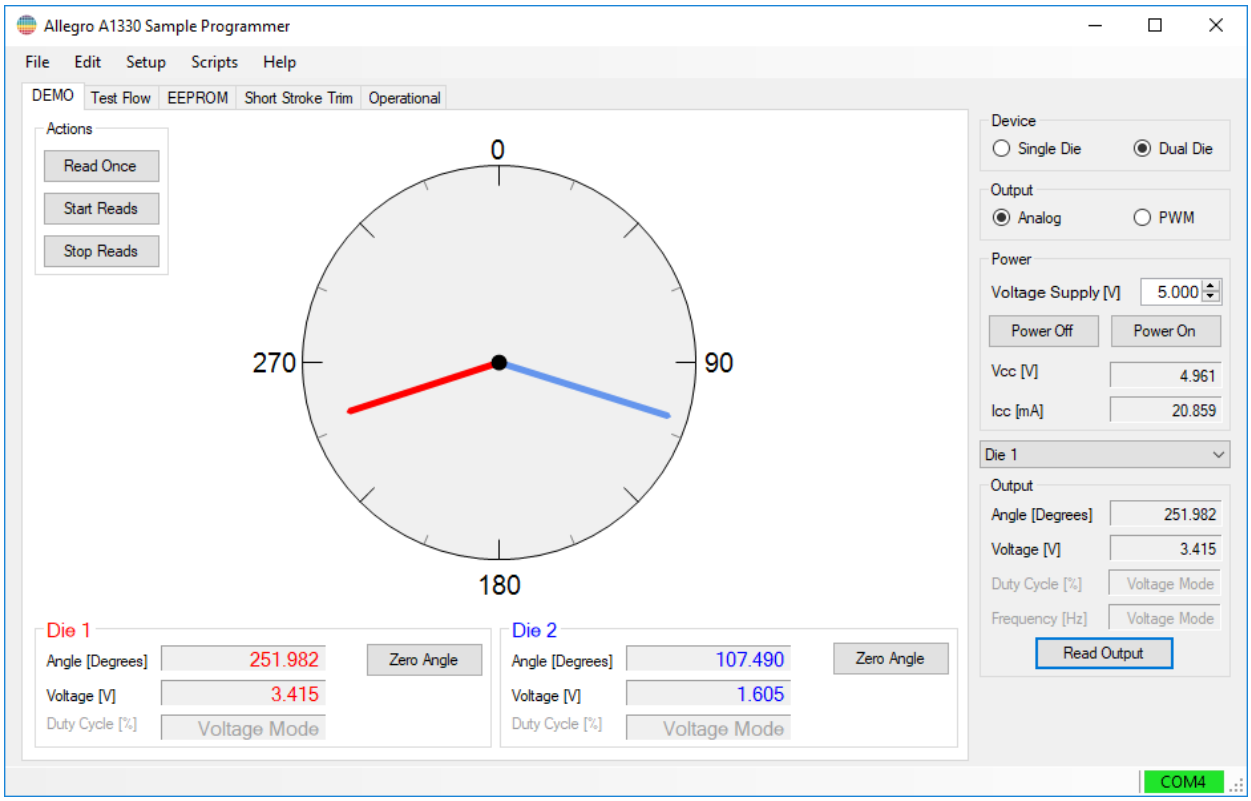


Figure 18: Die Selection

1. If using the ASEK-20 click **Single Die** for an A1330 single die part and **Dual Die** for a dual die part. This will enable the **Die** pop-up menu. If using an ASEK-21 and the **Map Die to Port** is not checked in **Device Setup:ASEK-21 Extra**, the **Single Die** and **Dual Die** radio buttons and the Die pop-up menu will be greyed out. Otherwise those controls will be enabled.
2. If **Dual Die** is checked, select **Die 1** from the **Die** pop-up menu to select the first die and **Die 2** to select the second die.

Status Bar

When the application is used with the ASEK-21, the status bar is extended with a port selection menu if **Map Die to Port** is not checked in **Device Setup:ASEK-21 Extra**.

Figure 19: Status Bar with ASEK-21

Port # (Only when ASEK-21 is used)

The Port popup displays the current port number. Clicking on it displays a popup menu which allows the user to select which port is used for Power, Ground and VOut.

Troubleshooting

Troubleshooting when using Manchester Protocol

1. If using a device setup to output PWM, is the pull-up resister settings for OUT enabled or is there a dedicated pull-up resister on OUT?
2. Turn power on to the device and verify that Vcc is within the operating range.
3. Turn off the device.
4. Turn on the device and verify using an oscilloscope that the access code was sent on the Vcc line during the power up.
5. Looking at the access code that was sent, were the Manchester serial high level and serial low level values within the spec for the device?
   5.1. If not, adjust the serial high level and serial low level values in the Device Setup dialog in the programming application then go back to step 2.
6. Perform a read and verify using an oscilloscope that there was a read response on the OUT line.
   6.1. If there was not a read response, slow down the communication speed and go back to step 2.
   6.2. If there still is not a read response, make sure the access code is correct.
   6.3. Make sure the address used is correct. For single die/device the address global can be used but for multiple die/devices, make sure the addresses are unique.
7. Once there is a read response, did the ASEK interpret the read response correctly?
   7.1. No. What was the error message?
      7.1.1. If the error was Timeout, using an oscilloscope, check the voltages of the response. Are the lower values below the threshold and the higher values above the threshold? If not, adjust the threshold so that is between the high and low values and go back to step 2.
      7.1.2. If the error was CRC error, retry the read. Try slowing down the communication speed.
8. Perform a write and verify using an oscilloscope that there was a write command.
9. Perform a read command to verify what was written previously.
   9.1. If the value read is not what was written?
      9.1.1. Verify that the part is not locked. If it is then try another part.
      9.1.2. Try another part to make sure the part is good.

Menu Bar

File

Load Memory File…
Restore the parameters set in a previous session. After selecting a file from the standard file browser, this command loads the parameters and selects them. The files do not have to contain all of the parameters.

Save Memory File…
Saves the currently selected parameters to a file. After naming the file using the standard file browser, this command saves the selected parameters into the file. The files do not have to contain all the parameters.

Exit
Exit the application.

Edit

Undo
Undo the last text box action, only enabled if the cursor is in a text box.

Cut
Copy the selected text onto the clipboard, only enabled if the cursor is in a text box.

Copy
Undo the last text box action, only enabled if the cursor is in a text box.

Paste
Paste from the clipboard into the currently highlighted text box, only enabled if the cursor is in a text box.

Select All
Select all of the rows in the memory table.

Select Highlighted
Select all of the rows in the memory table that are highlighted.

Deselect All
Deselect all of the rows in the memory table.

Deselect Highlighted
Deselect all of the rows in the memory table that are highlighted.

Restore Display Order

Sorts the memory table the same was as it was when the application was started.

Setup

Communication Setup…

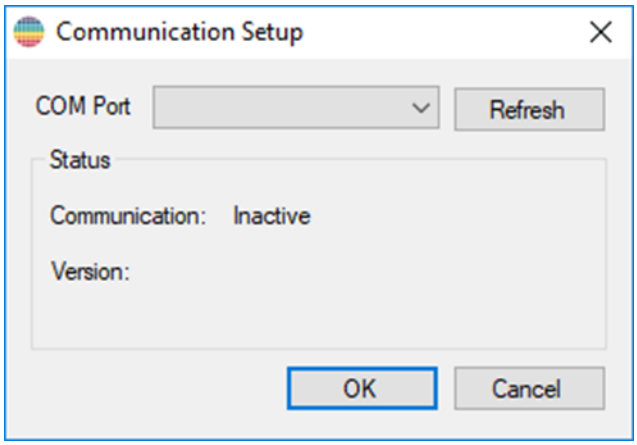


Figure 20: Communication Setup Dialog

COM Port Pull Down Menu

This popup menu displays the currently available COM ports. The COM port that is attached to an ASEK-20 or ASEK-21 needs to be selected.

Refresh

The refresh button will update the contents of the COM Port Pull Down Menu and then try to establish communication with the ASEK-20 if one is selected.

Status

The status will show the current status of communications with the ASEK-20.

Communication

If communication with the ASEK-20 has been established then this field will be “Active” otherwise it will be “Inactive”.

Version

If communication has been established with the ASEK-20 then this field will display the version of the firmware in the ASEK-20.

OK

Save the com port chosen and close the dialog.

Cancel
Discard any changes made and close the dialog.

Device Setup…

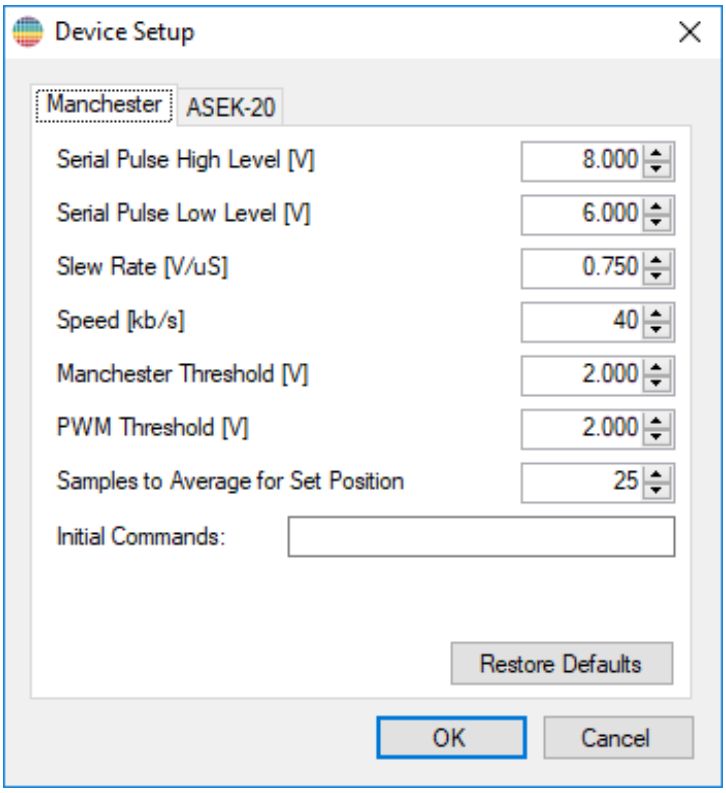


Figure 21: Device Setup

Serial Pulse High Level [V] Text Entry Box
This is the voltage that will be used for the high level of the Manchester signal.

Serial Pulse Low Level [V] Text Entry Box
This is the voltage that will be used for the low level of the Manchester signal.

Slew Rate [V/µS] Text Entry Box
Used to set the speed at which the Manchester signal will take to get from one voltage to another. The default is 0.75 Volts/µS.

Speed [kb/s] Text Entry Box
This is the bit rate that will be used when communicating with the A1330.

Manchester Threshold [V] Text Entry Box
This is the voltage that will be used to determine if the input is true or false when reading the Manchester signals.

PWM Threshold [V] Text Entry Box

Used to set the threshold for determining the difference between a 1 and a 0 when performing a PWM read.

Initial Command Text Entry Box

This is used for commands that need to be sent to the ASEK-20/21 when it is initialized.

Restore Defaults

When clicked, all of the default values for the Manchester settings are restored.

ASEK-20

When the programmer is connected to an ASEK-20, this tab is shown.

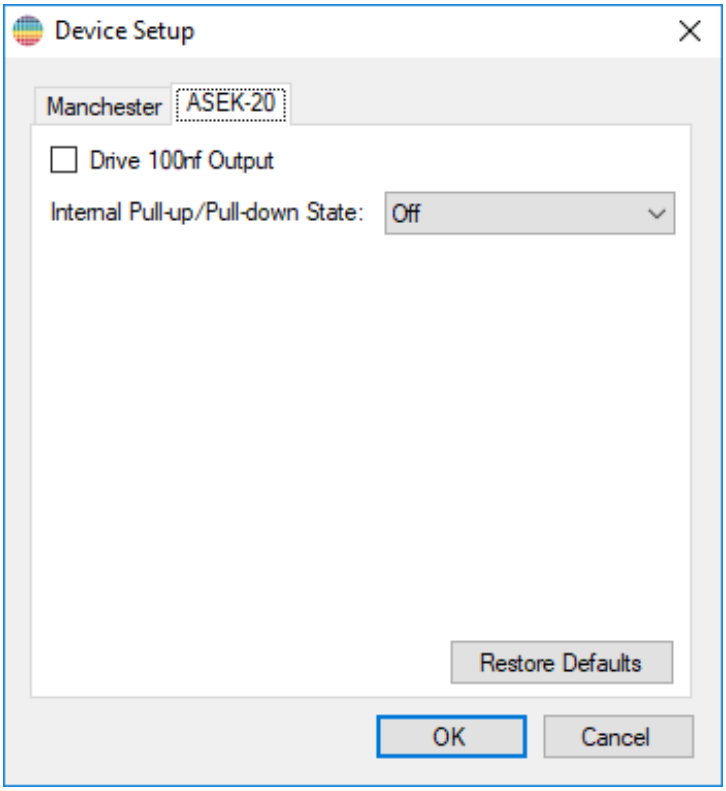


Figure 22: Device Setup with ASEK-20

Drive 100nf Output

If the Vcc line has about 100nF of capacitance then this option is used to control the overshoot.

Internal Pull-up/Pull-down State

The TED-002219 daughterboard has pull-up and pull-down resisters on the board and this option is used to control then.

- Off – Do not connect the pull-up or pull-down resisters to the output lines.
- Pull-up – Connect 4.75k ohm pull-up resisters from OUT to 5 volts.
- Pull-down – Connect 10k ohm pull-down resisters from OUT to ground.

Restore Defaults

When clicked, all of the default values for the ASEK-20 settings are restored.

ASEK-21

When the programmer is connected to an ASEK-21, this tab is shown.

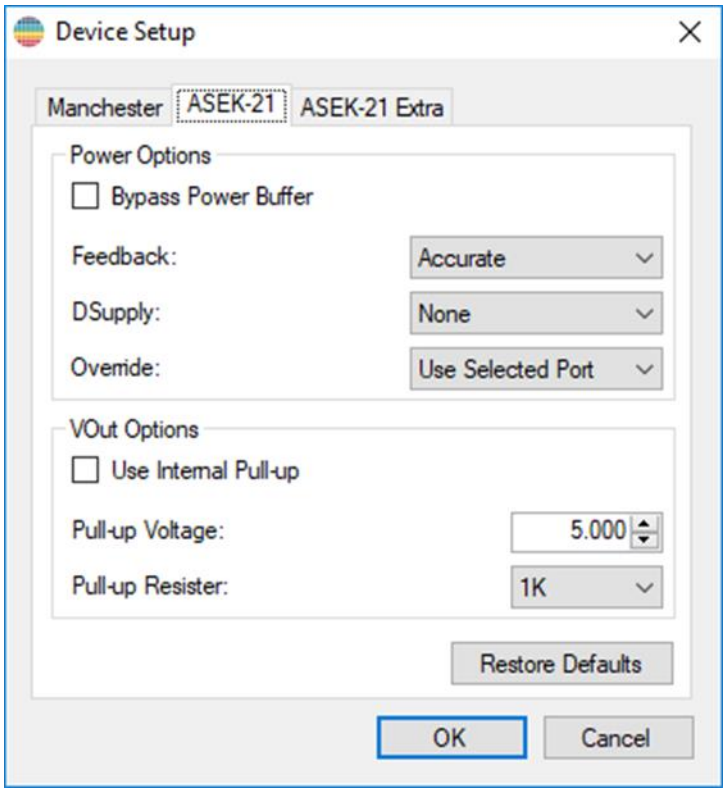


Figure 23: Device Setup with ASEK-21

Bypass Power Buffer Check Box

By default, the power supplied by the ASEK-20 is buffered internally by the ASEK-21. This allows higher current supply. This check box allows that buffer to be bypassed and the power to come directly from the ASEK-20. When checked, the power is limited to what the ASEK-20 can supply (about 40 mA).

Feedback Pull Down Menu

When the power is buffered by the ASEK-21, one of three feedback loops can be used. The first, **Stable** is the most stable but the voltage will decrease with the current draw (about 10mV per mA). The second is

Accurate which is the default. The third is **External** which uses the VCC_S from the module under test. This is the most accurate but the least stable.

DSupply Pull Down Menu

The DSupply pin can supply power separate from the VCC lines. It can only be one of four values and they are:

- No Voltage Supplied
- 5 Volts
- 3.3 Volts
- 2.7 Volts

Override Pull Down Menu

The Override is used to select which supply power and ground lines to use and can be separate from the VOut lines. It can only be one of five values and they are:

- Use Selected Port – Always use the power and ground lines specified by the port control.
- Use Port 1 – Always use the power and ground lines on port 1
- Use Port 2 – Always use the power and ground lines on port 2
- Use Port 3 – Always use the power and ground lines on port 3
- Use Port 4 – Always use the power and ground lines on port 4

Use Internal Pull-up Check Box

When checked, the VOut line is pulled up by the designated resister by the chosen voltage.

Pull-up Voltage Text Entry Box

Selects the voltage that VOut will be pulled up to.

Pull-up Resister Pull Down Menu

Selects either the 1K or 4.75K pull-up resister.

Restore Defaults

When clicked, all of the default values for the ASEK-21 settings are restored.

ASEK-21 Extra

When the programmer is connected to an ASEK-21, this tab is also shown.

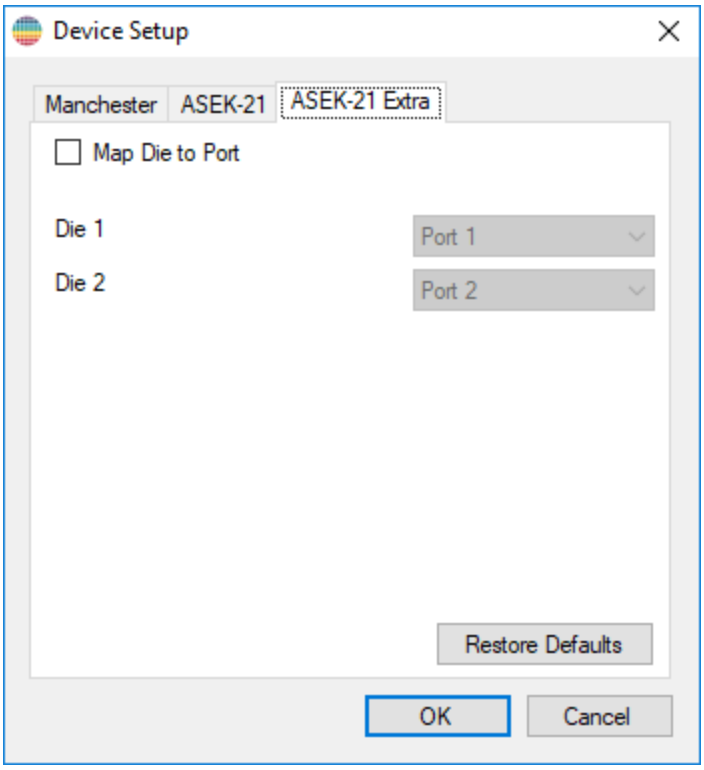


Figure 24: Device Setup with ASEK-21 Extras

Map Die to Port Check Box

By default, when using the ASEK-21 the **Die Selection** menu is disabled, and port selection is performed using the popup menu in the status bar at the bottom of the window. When this control is checked, the port selection menu is hidden, the Die Selection is enabled, and when the die is selected from the die selection menu the port is selected from the lower choices. The default is unchecked.

Die 1 Pull Down Menu

When **Map Die to Port** is checked and Die 1 is selected in the **Die Selection** menu, the port is set to one of the choices selected. It can only be one of four values and they are:

- Port 1 (default)
- Port 2
- Port 3
- Port 3

Die 2 Pull Down Menu

When **Map Die to Port** is checked and Die 2 is selected in the **Die Selection** menu, the port is set to one of the choices selected. It can only be one of four values and they are:

- Port 1

- Port 2 (default)
- Port 3
- Port 3

Restore Defaults

When clicked, the default values for the ASEK-21 extra settings are restored.

OK

Save the options chosen and close the dialog.

Cancel

Discard any changes made and close the dialog.

Board Calibration…

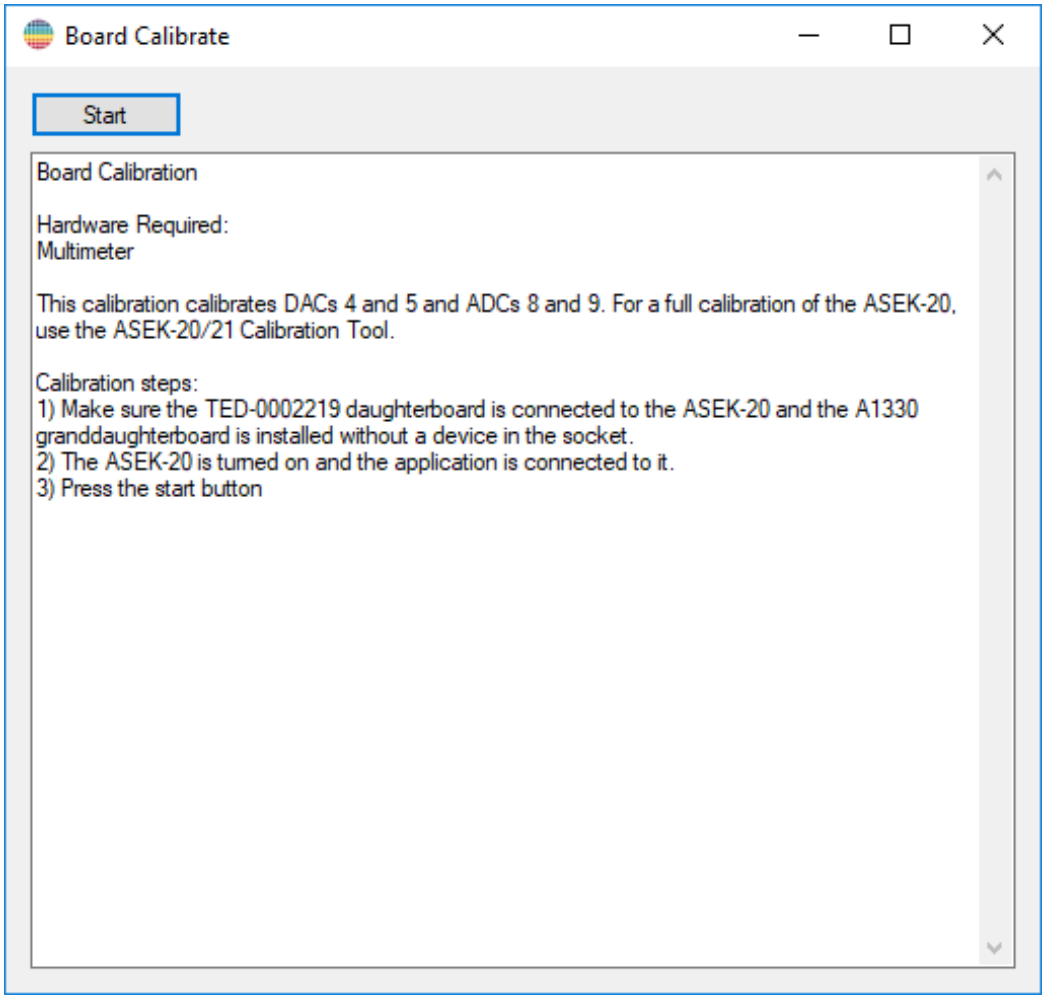


Figure 25: Board Calibration

This menu option is used to calibrate DACs 4 and 5 and ADCs 8 and 9. For a full calibration of the ASEK-20, use the ASEK-20/21 Calibration Tool. To perform the calibration, perform these steps:

1. Make sure the 85-0540-109 daughterboard is connected to the ASEK-20 and the A1330 grand-daughterboard is installed.
2. The ASEK-20 is turned on and the application is connected to it.
3. A multi-meter is set to measure the voltage at the VCC test-point.
4. Press the start button

Display as Decimals

Displays the code column in the memory grid and the raw angles in decimal notation. The values that are entered can be in either decimal or hexadecimal but the application will convert them and display them in decimal.

Display as Hexadecimals

Displays the code column in the memory grid and the raw angles in hexadecimal notation. The values that are entered can be in either decimal or hexadecimal but the application will convert them and display them in hexadecimal.

Scripts

Run Script…

Allows the user to specify and run scripts written in C# or Visual Basic.

Other scripts

Any scripts placed in "C:\ProgramData\Allegro MicroSystems\A1330 Samples Programmer\Scripts" or "C:\Users\{User Name}\AppData\Local\Allegro MicroSystems\A1330 Samples Programmer\Scripts" will be added to the menu and will be able to be run by selecting them from the menu.

Scripting

C#

To create a script using C# that runs in the Allegro A1330 Samples programmer, use this template:

```
using System;
using Allegro.Script;

namespace ScriptTemplate
{
    public class ScriptTemplate: IScriptSource
    {
       public int ScriptType()
       {
              return 0;
       }

       public string ScriptName()
       {
```

```
            return "ScriptTemplate";
        }

        public string RunScript(IScriptPerformer performer)
        {

            return "This script does nothing";
        }
    }
}
```

The return value of the script type is not used at the present time so leave it alone. The script name should match the name of the file without the extension and can include spaces. Run script is the method that will be executed when the script is run. The performer parameter is the scripts access to the application. The full C# language and all of the .net libraries are available for use in the scripting language.

IScriptPerformer Methods

SetScriptVariable

Set a string into the script variable dictionary with the given name.

```
void SetScriptVariable(string name, string value);
```

GetScriptVariable

Get a name string from the script variable dictionary. If the name is not in the dictionary a null is returned. The only predefined string is “scripting_path” which is how the script can set which file to be used instead of opening the file load or save dialogs and asking the user.

```
bool GetScriptVariable(string name, out string value);
```

SetScriptObject

Set an object into the script object dictionary with the given name.

```
void SetScriptObject(string name, object value);
```

GetScriptObject

Get a name object from the script object dictionary. If the name is not in the dictionary a null is returned. Currently the only predefined object is “ASEK” which is the ASEK programmer the application is currently using.

```
object GetScriptObject(string name);
```

Perform Action

This method performs the equivalent of a mouse click on an action keyword. There are action keywords for each of the buttons in the user interface.

```
bool PerformAction(string action);
```

PerformParameterAction

Perform parameter action will set or reset the select checkbox in the memory grid.

```
bool PerformParameterAction(string parameter, string action);
```

SetValue

Set the value of the given item.

```
bool SetValue(string target, string value);
```

SetParameterValue

Set the value of the given parameter.

```
bool SetParameterValue(string parameter, string parameterType, string value);
```

GetValue

Get the value of the given item.

```
string GetValue(string target);
```

GetDoubleValue

Get the value of the given item and try to convert it into a double.

```
double GetDoubleValue(string target);
```

GetIntegerValue

Get the value of the given item and try to convert it into an integer.

```
int GetIntegerValue(string target);
```

GetParameterValue

Gets the string value of the given parameter.

```
string GetParameterValue(string parameter, string parameterType);
```

GetDoubleParameterValue

Gets the double value of the given parameter.

```
double GetDoubleParameterValue(string parameter, string parameterType);
```

GetIntegerParameterValue

Gets the integer value of the given parameter.

```
int GetIntegerParameterValue(string parameter, string parameterType);
```

Example

```
namespace VerifyOperational
{
    public class VerifyOperational : IScriptSource
```

```
    {
        public int ScriptType()
        {
            return 0;
        }
        public string ScriptName()
        {
            return "VerifyOperational";
        }
        public string RunScript(IScriptPerformer performer)
        {
            ASEK asek = (ASEK)performer.GetScriptObject("ASEK");
            if (asek == null)
            {
                return "Communication not initialized.";
            }
            // Check to see if the ASEK is available to program with.
            if (!asek.IsActive())
            {
                return "Unable to verify the ASEK is connected and powered on Make sure
it is connected and powered on.";
            }
            // Make sure the test starts in a known point, power off.
            if (!performer.PerformAction("PowerOff"))
            {
                return "Unable to perform the power off action. The ASEK is not
responding to commands. Could be a bad ASEK.";
            }

            // Make sure the voltage supply is really off
            if (performer.GetDoubleValue("Vcc") > 0.1)
            {
                return "Unable to verify that the supply voltage is off. Could be a bad
ASEK.";
            }
            // Power up the device.
            if (!performer.PerformAction("PowerOn"))
            {
                return "Unable to perform the power on action. Could be a bad ASEK.";
            }
            // Make sure the supply is close to the desired voltage.
            double desiredVoltage = performer.GetDoubleValue("VoltageSupply");
            double voltageReading = performer.GetDoubleValue("Vcc");
            if ((voltageReading < (desiredVoltage - 0.1)) || (voltageReading >
(desiredVoltage + 0.1)))
            {
                return string.Format("The supply voltage is not close to the desired
voltage. Could be a bad ASEK. desired={0}, supply={1}", desiredVoltage, voltageReading);
            }
            // Write to a e_f_custwrd_c register to check to see if the device is present
and operational
            int readResults;
            Random randObj = new Random();
            int data = randObj.Next(0x0FFFFFF);
```

```
            try
            {
                ((IRegisterAccess)asek).WriteRegister(MemoryAccessType.primary,
                0x3F, (uint)data);
            }
            catch
            {
                return "Error while writing to the A1330, verify that there is an A1330
in the socket.\r\nIf there is one then it could be a bad A1330 or a different device.";
            }

            // Read the e_f_custwrd_c register
            try
            {
                readResults =
(int)((IRegisterAccess)asek).ReadRegister(MemoryAccessType.primary, 0x3F);
            }
            catch
            {
                return "Error while reading from the device, could be a bad A1330 or a
different device.";
            }
            // Check to make sure the number that was written is the same as what was
read.
            if (data != readResults)
            {
                return string.Format("Data read from the A1330 does not match what was
written, bad device. written={0}, read={1}", data, readResults);
            }
            return "The set-up is operational.";
        }
    }
}
```

Visual Basic

To create a script using Visual Basic that runs in the Allegro A1330 Samples programmer, use this template:

```
Imports System
Imports Allegro.Script

Public Class ScriptTemplate
       Implements IScriptSource

       Public Function ScriptType() As Integer Implements IScriptSource.ScriptType
              Return 0
       End Function

       Public Function ScriptName() As String Implements IScriptSource.ScriptName
              Return " ScriptTemplate "
       End Function

       Public Function RunScript(performer As IScriptPerformer ) As String Implements
IScriptSource.RunScript
              Return "The script is done."
```

```
    End Function
End Class
```

The return value of the script type is not used at the present time so leave it alone. The script name should match the name of the file without the extension and can include spaces. Run script is the method that will be executed when the script is run. The performer parameter is the scripts access to the application. The full visual basic language and all of the .net libraries are available for use in the scripting language.

IScriptPerformer Methods

SetScriptVariable

Set a string into the script variable dictionary with the given name.

```
void SetScriptVariable(string name, string value);
```

GetScriptVariable

Get a name string from the script variable dictionary. If the name is not in the dictionary a null is returned. The only predefined string is “scripting_path” which is how the script can set which file to be used instead of opening the file load or save dialogs and asking the user.

```
bool GetScriptVariable(string name, out string value);
```

SetScriptObject

Set an object into the script object dictionary with the given name.

```
void SetScriptObject(string name, object value);
```

GetScriptObject

Get a name object from the script object dictionary. If the name is not in the dictionary a null is returned. Currently the only predefined object is “ASEK” which is the ASEK programmer the application is currently using.

```
object GetScriptObject(string name);
```

Perform Action

This method performs the equivalent of a mouse click on an action keyword. There are action keywords for each of the buttons in the user interface.

```
bool PerformAction(string action);
```

PerformParameterAction

Perform parameter action will set or reset the select checkbox in the memory grid.

```
bool PerformParameterAction(string parameter, string action);
```

SetValue

Set the value of the given item.

```
bool SetValue(string target, string value);
```

SetParameterValue

Set the value of the given parameter.

```
bool SetParameterValue(string parameter, string parameterType, string value);
```

GetValue

Get the value of the given item.

```
string GetValue(string target);
```

GetDoubleValue

Get the value of the given item and try to convert it into a double.

```
double GetDoubleValue(string target);
```

GetIntegerValue

Get the value of the given item and try to convert it into an integer.

```
int GetIntegerValue(string target);
```

GetParameterValue

Gets the string value of the given parameter.

```
string GetParameterValue(string parameter, string parameterType);
```

GetDoubleParameterValue

Gets the double value of the given parameter.

```
double GetDoubleParameterValue(string parameter, string parameterType);
```

GetIntegerParameterValue

Gets the integer value of the given parameter.

```
int GetIntegerParameterValue(string parameter, string parameterType);
```

Example

```
Imports System
Imports Allegro.ASEK
Imports Allegro.Script
Public Class VerifyOperational
    Implements IScriptSource
    Public Function ScriptType() As Integer Implements IScriptSource.ScriptType
        Return 0
    End Function
    Public Function ScriptName() As String Implements IScriptSource.ScriptName
        Return "VerifyOperational"
    End Function
```

```
    Public Function RunScript(performer As IScriptPerformer) As String Implements
IScriptSource.
    RunScript
        Dim asek As ASEK = DirectCast(performer.GetScriptObject("ASEK"), ASEK)
        Dim reg As IRegisterAccess = DirectCast(performer.GetScriptObject("ASEK"),
IRegisterAccess)
        If asek = null Then
            Return "Communication not intialized."
        End If
        ' Check to see if the ASEK is available to program with.
        If asek.IsActive() = False Then
            Return "Unable to verify the ASEK is connected and powered on. Make sure it
is on the local network, connected it and powered on."
        End If
        ' Make sure the test starts in a known point, power off.
        If performer.PerformAction("PowerOff") = False Then
            Return "Unable to perform the power off action. The ASEK is not responding to
commands. Could be a bad ASEK."
        End If
        ' Make sure the voltage supply is really off
        If performer.GetDoubleValue("voltage_supply_reading") > 0.1 Then
            Return "Unable to verify that the supply voltage is off. Could be a bad
ASEK."
        End If
        ' Power up the device.
        If performer.PerformAction("PowerOn") = False Then
            Return "Unable to perform the power on action. Could be a bad ASEK."
        End If
        ' Make sure the supply is close to the desired voltage.
        Dim desiredVoltage As Double = performer.GetDoubleValue("VoltageSupply")
        Dim voltageReading As Double = performer.GetDoubleValue("Vcc")
        If ((voltageReading < (desiredVoltage - 0.1)) Or (voltageReading >
(desiredVoltage + 0.1))) Then
            Return String.Format("The supply voltage is not close to the desired voltage.
Could be a bad ASEK. desired={0}, supply={1}", desiredVoltage, voltageReading)
        End If
        ' Write to the e_f_custwrd_c register to check to see if the device is present
and operational
        Dim readResults As UInteger
        Dim address As UInteger = &H1D
        Dim randObj As Random = New Random()
        Dim data As UInteger = randObj.Next(&HFFF)
        Try
            reg.WriteRegister(MemoryAccessType.primary, &H3F, data)
        Catch ex As Exception
            Return "Error while writing to the A1330, verify that there is an A1330 in
the socket.\r\nIf there is one then it could be a bad A1330 or a different device."
        End Try
        ' Read the e_f_custwrd_c register
        Try
            readResults = reg.ReadRegister(MemoryAccessType.primary, &H3F)
        Catch ex As Exception
            Return "Error while reading from the device, could be a bad A1330 or a
different device."
```

```
        End Try
        ' Check to make sure the number that was written is the same as what was read.
        If data <> readResults Then
            Return String.Format("Data read from the A1330 does not match what was
written, bad device. written={0}, read={1}", data, readResults)
        End If
        Return "The set-up is operational."
    End Function
End Class
```

Keywords

Action Keywords

LoadMemoryFile

Loads the memory grid from a file. If the script variable scripting_path is set, it is used instead of opening up a file browser and asking the user where the file is.

SaveMemoryFile

Saves the selected parameters in the memory grid to a file. If the script variable scripting_path is set, it is used instead of opening up a file browser and asking the user where the file is.

PowerOn

Sets the voltage supplied to the device to the value contained in VoltageSupply then turns the supply on.

PowerOff

Turns the power off for the device.

UpdateAngles

Updates the Vcc, Icc, Angle, Voltage, Duty Cycle and Frequency.

Die1ZeroAngle

Takes an angle reading on die 1 then writes a value into the ANGLE_OFFSET field so that the currently read angle is zero.

Die2ZeroAngle

Takes an angle reading on die 2 then writes a value into the ANGLE_OFFSET field so that the currently read angle is zero.

SelectAllEEPROM

Selects all of the “EEPROM”. Note: It will select both the memory locations and the fields.

DeselectAllEEPROM

Deselects all of the “EEPROM”.

ZeroSelectedEEPROM
Replaces the values associated with the selected “EEPROM” with zeros.

ClearSelectedEEPROM
Removes the values associated with the selected “EEPROM”.

RestoreDefaultSelectedEEPROM
Replaces the values associated with the selected “EEPROM” with the factory defaults.

ReadSelectedEEPROM
Reads all of the selected “EEPROM” to the A1330.

WriteSelectedEEPROM
Writes all of the selected “EEPROM” to the A1330. They can be a mix of memory locations and fields.

VerifySelectedEEPROM
Reads the selected EEPROM fields from the A1330 and verifies that it is the same as what is in the EEPROM table.

LoadEEPROMFile
Load the EEPROM fields and/or memory locations from a text or csv file and select the fields and memory locations that were loaded. If the scripting variable “scripting_path” is set, this path will be used instead of opening a file browser and asking the user for the file.

SaveEEPROMFile
Save the selected EEPROM fields and/or memory locations to a text or csv. If the scripting variable “scripting_path” is set, this path will be used instead of opening a file browser and asking the user for the file.

Value Keywords

VoltageSupply
This is the desired voltage supply. It will be used when the power is turned on.

Vcc
Vcc is the voltage that is currently being supplied to the device. (Read Only)

Icc
Icc is the current that is currently being supplied to the device. (Read Only)

Die1Angle
The value in degrees of the last angle reading of die 1 of the device. (Read Only)

Die2Angle
The value in degrees of the last angle reading from die 2 of the device. (Read Only)

Angle
The value in degrees of the last angle reading. (Read Only)

Voltage
The value in volts of the last voltage reading. (Read Only)

DutyCycle
The value in percent of the last duty cycle reading. (Read Only)

Frequency
The value in hertz of the last duty cycle reading. (Read Only)

SingleDieSupport
Switches the application to only support single dies.

MultiDieSupport
Switches the application to only support multiple dies.

MemoryTabControl
Controls what will be displayed in the main window. Possible values are “DEMO”, “EEPROM”, “Short Stroke Trim” and “Operational”.

Selected
Returns the number of selected parameters in the memory grid for the tab displayed in the main window.

ShowEEPROMViewAs
Selects which fields or memory locations will be displayed by the EEPROM tab. Possible values are “All Memory Locations”, “All Fields” or “Short Stroke Fields”.

Port1 (ASEK-21 only)
Switch the power, ground and VOut lines to port 1.

Port2 (ASEK-21 only)
Switch the power, ground and VOut lines to port 1.

Port3 (ASEK-21 only)
Switch the power, ground and VOut lines to port 1.

Port4 (ASEK-21 only)
Switch the power, ground and VOut lines to port 1.

SelectDie (ASEK-20, Multiple Die only)
Selects which die is to operated on. Possible values are “Die 1” or “Die 2”.

Parameters

The parameters are identified by the parameter name, a @ then the group name. For example, to specify the customer field in the eeprom the string would be “Customer word@eeprom”

eeprom

e_dac_f	ROE	e_e_errmsk_c	MISS_MAG_THRSH
PREGAIN_OFFSET	CE	MMF	ANG_AVE
e_b_shrtstrk0_c	e_shrtstrk2_c	MINA	PWM_FREQ
GAIN	LOW_CLAMP	MAXA	HYS
SS	HIGH_CLAMP	EED	HYS_O
e_shrtstrk1_c	POSTGAIN_OFFSET	OVLO	EELO
MIN_INPUT	PO	TOR	e_f_custwrd_c
MAX_INPUT	ABW	INTER	Customer word

Help

View Help

Opens the help viewer and displays the help file for Allegro A1330 Samples Programmer.

A1330 Registers Diagram

Displays a window with a representation of the registers of the A1330.

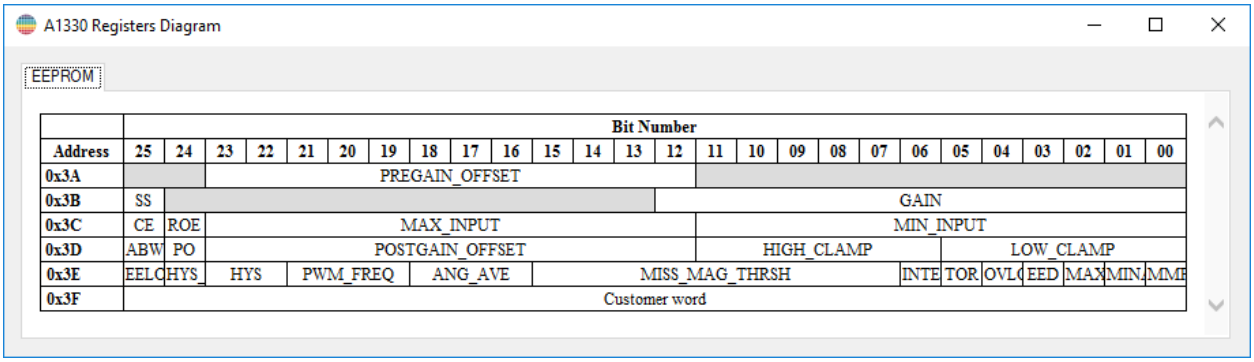


Figure 26: A1330 Registers Diagram

Allegro Software Web Site

Opens the web browser main page on the Allegro Software Web site.

Allegro Web Site

Opens the web browser main page on the Allegro Web site.

About Allegro A1330 Samples Programmer

Figure 27: About Allegro A1330 Samples Programmer Dialog Box

Displays the version of the application and the Allegro specific DLLs.

DEMO

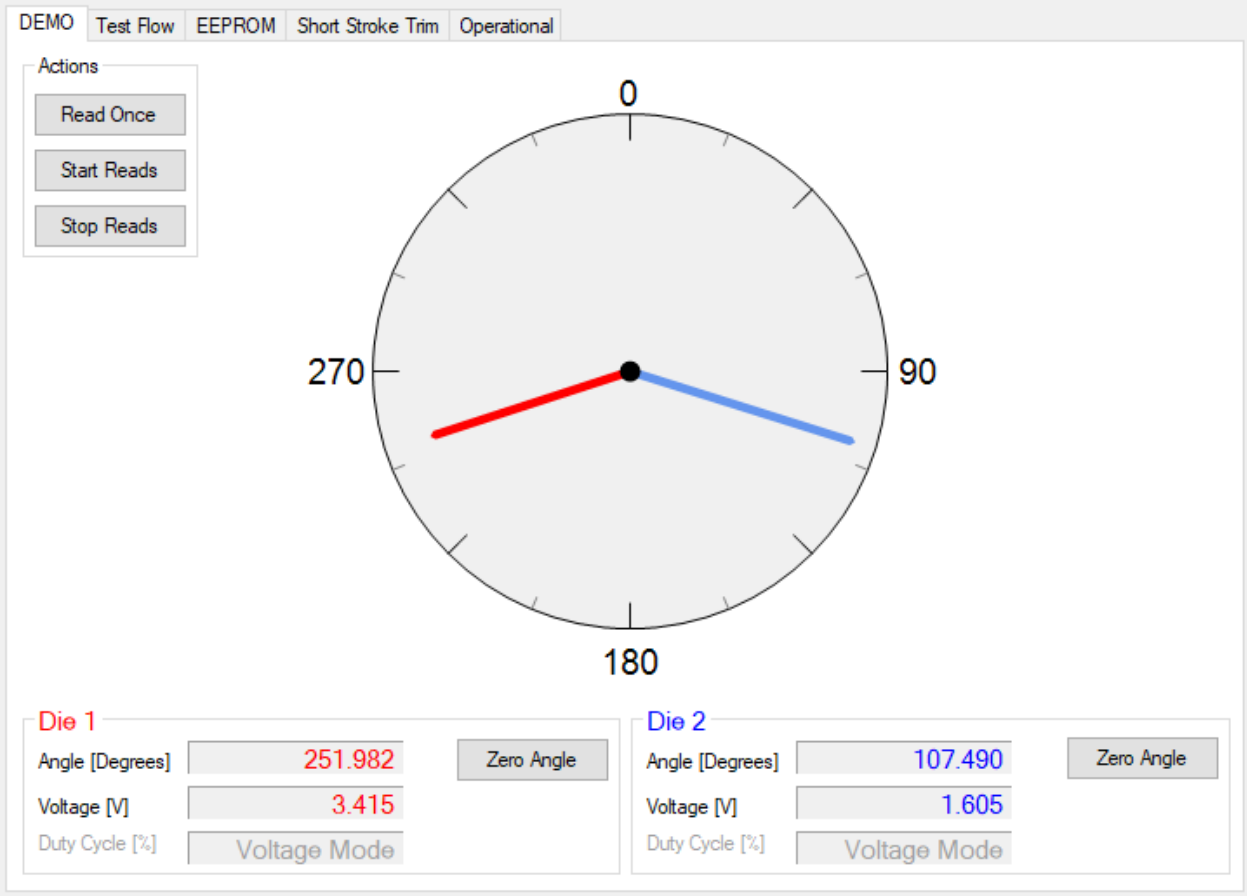


Figure 28: DEMO

Reading From the Device

1. Click **Read Output**. All of the angle, Voltage and Duty Cycle fields will be updated.
2. Clicking **Read Once** will also update all of the angle, voltage and duty cycle fields.
3. Clicking **Start Read** will continuously update all of the angle, voltage and duty cycle fields until **Stop Reads** is clicked.

Zeroing the Angle

1. Click **Zero Angle** to read the angle and then set pregain_offset so that the angle read will be zero.

Reference

Read Once Button

Update the angle, voltage and duty cycle in the window. If dual die is selected, updates both die.

Start Reads Button

Start a periodic update for angle, voltage and duty cycle on the page. If dual die is selected, updates both die.

Stop Reads Button

Stop the periodic update for angle, voltage and duty cycle on the page. If dual die is selected, updates both die.

Die # Field

The information specific to the die are displayed in these fields. If single die is selected, then Die 2 is grayed out.

Angle [Degrees] Text Box

This is the angle value read from the die and displayed in degrees.

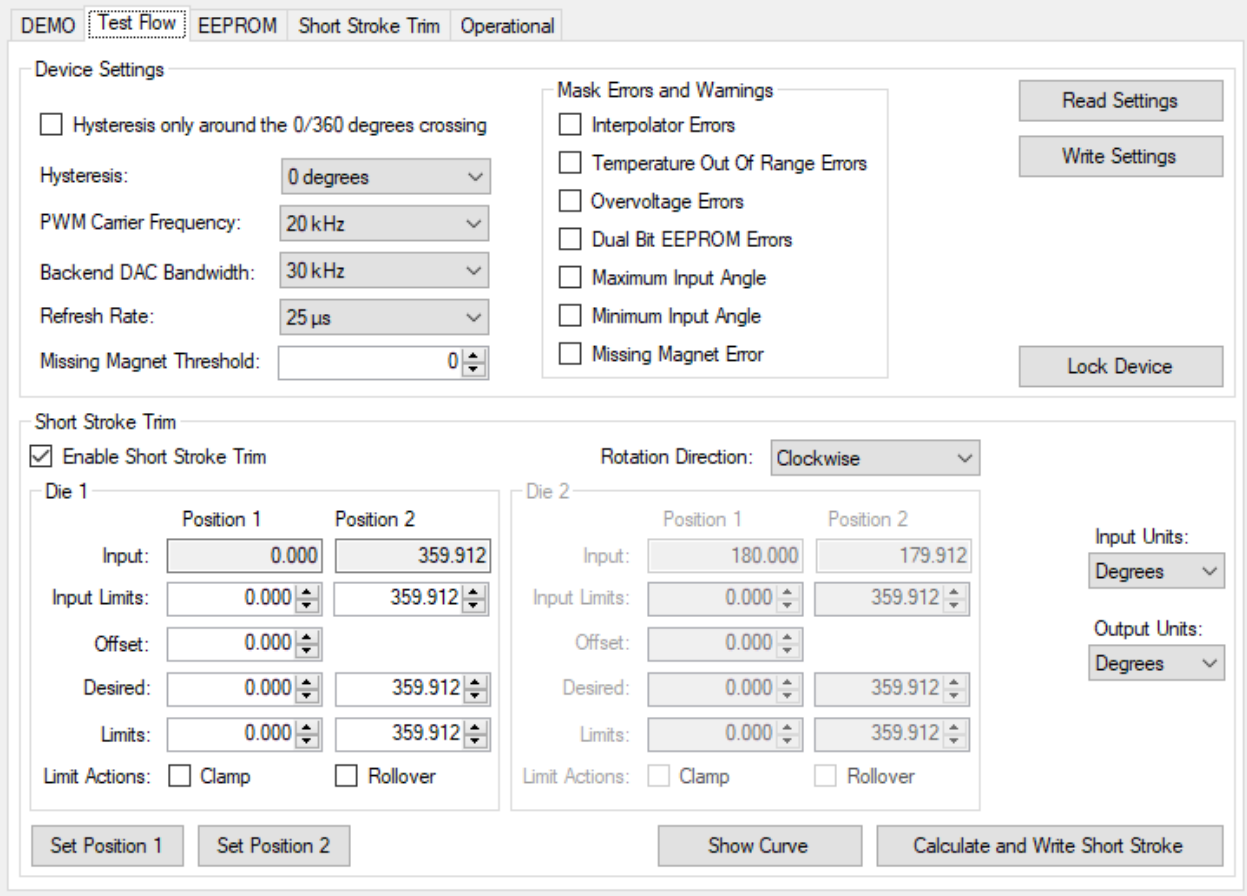
Voltage [Volts] Text Box

This is the value read from the die and displayed in volts.

Duty Cycle [%] Text Box

This is the pwm duty cycle value read from the die and displayed in percentage.

Test Flow



Device Settings

Hysteresis only around the 0/360 degrees crossing Checkbox

When checked hysteresis is only applied within ±11.16° of the 0/360 crossover point.

Hysteresis Popup Menu

Allows selection of when the angle will not update unless a change larger than the hysteresis range is observed. Possible values are:

- 0 degrees
- 0.352 degrees
- 0.703 degrees
- 1.406 degrees

PWM Carrier Frequency Popup Menu

Sets the PWM carrier frequency. The possible values are:

- 20 kHz

- 10 kHz
- 5 kHz
- 2.5 kHz
- 1.25 kHz
- 625 Hz
- 312.5 Hz
- 156.25 Hz

Backend DAC Bandwidth Popup Menu

Allows selection of the band width of the analog filter. The possible values are:

- 30 kHz
- 15 kHz

Refresh Rate Popup Menu

Allows selection of the approximate refresh rate. The possible values are:

- 25 µs
- 50 µs
- 100 µs
- 200 µs
- 400 µs
- 800 µs
- 1600 µs
- 3200 µs

Missing Magnec Threshold Text Box

Allows selection of the threshold below which the missing magnet flag will assert. This is programmed for a default of 100 G.

If a setting other than 100 G is desired, simply scale the existing value by d_field / 100 where “d_field” is the desired trip point in gauss.

Example: If the desired trip point is 300 G, and the default factory EEPROM value is 5, then the final value is 300 / 100 × 5 = 15.

Mask Errors and Warnings

When any of the following mask bits are set, the error or warning they describe will not tri-state the output.

Interpolator Error Checkbox

When set, prevents an interpolator error from tri-stating the output.

Temperature Out Of Range Error Checkbox
When set, prevents an interpolator error from tri-stating the output.

Overvoltage Error Checkbox
When set, prevents an overvoltage error from tri-stating the output.

Dual Bit EEPROM Error Checkbox
When set, prevents a dual bit EEPROM error from tri-stating the output

Maximum Input Angle Checkbox
When set, the output will not tri-state when the input angle is above the MAX_INPUT value.

Minimum Input Angle Checkbox
When set, the output will not tri-state when the input angle is below the MIN_INPUT value.

Missing Magnet Flag Checkbox
When set, output will not tri-state if the measured magnetic amplitude is below the MIS_MAG_THRSH.

Read Settings
Reads the selected device and sets the values to the controls.

Write Settings
Takes the values from the controls and writes them into the selected device.

Lock Device
Presents the Lock Memory dialog, and if the user presses **Yes**, sets the field “EELO” to 1 and writes them into the device(s).

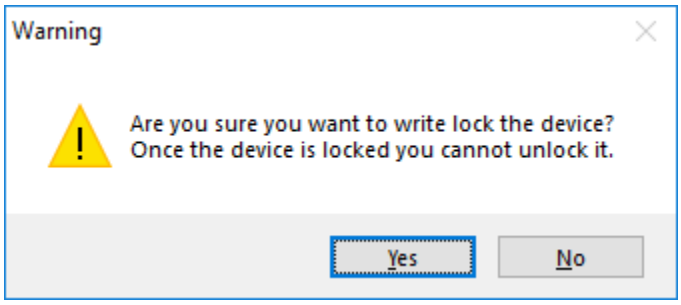


Figure 29: Lock Memory Dialog

Two Point Programming

Enable Short Stroke Trim Checkbox
When set, short stroke trim controls will be enabled. In single die mode, only Die 1 will be enabled and in Dual Die mode, both die are enables and will be read and written to in the same operation.

Rotation Direction Popup Menu

Choose the direction to match the direction of rotation of the magnetic field. The possible values are:

- Clockwise – pol will be initialized to 0
- Counter Clockwise – pol will be initialized to 1

Input Units Pop-up Menu

This pop-up menu selects which units will be used in the input limits and offset text boxes. The options are Degrees and Percentage.

Output Units Pop-up Menu

This pop-up menu selects which units will be used in the desired and limit text boxes and displayed in the Output Angle controls. When Analog is selected as the device output, the options are Degrees and Volts. When PWM is selected as the device output, the options are Degrees and Duty Cycle.

Input Position 1 Text Box

The is the measured value when the target is in position 1.

Input Position 2 Text Box

The is the measured value when the target is in position 2.

Input Limits Position 1 Text Box

The is the input value pre-gain below which the output will tri-state unless the Minimum Input Angle Checkbox is set. This value is written into MIN_INPUT.

Input Limits Position 2 Text Box

The is the input value pre-gain above which the output will tri-state unless the Maximum Input Angle Checkbox is set. This value is written into MAX_INPUT.

Offset Text Box

The offset will shift the start of the output by this amount divided by the gain.

Desired Position 1 Text Box

After programming, when the input is at the first position, this is the value that will be output by the A1330.

Desired Position 2 Text Box

After programming, when the input is at second position, this is the value that will be output by the A1330.

Limits Position 1 Text Box

After programming, when the output value is less than this value, this is the value that will be output by the A1330.

Limits Position 2 Text Box

After programming, when the output value is greater than this value, this is the value that will be output by the A31314. Available for both IC No. 1 and IC No. 2.

Limit Actions: Clamp Checkbox

When set, prevents an interpolator error from tri-stating the output.

Limit Actions: Rollover Checkbox

When set, prevents an interpolator error from tri-stating the output.

Set Position 1 Button

After moving the target to position 1, this button will read the values from the device(s) and display them in the Measured Position 1 Text Boxes.

Set Position 2 Button

After moving the target to position 2, this button will read the values from the device(s) and display them in the Measured Position 2 Text Boxes.

Show Curve Button

This button displays a window which contains the curve that the device will output based on the current short stroke inputs.

Calculate and Write Short Stroke

Takes the measured point values and calculates the gain and offset needed to get the first and second positions. And writes the values into the device(s).

EEPROM

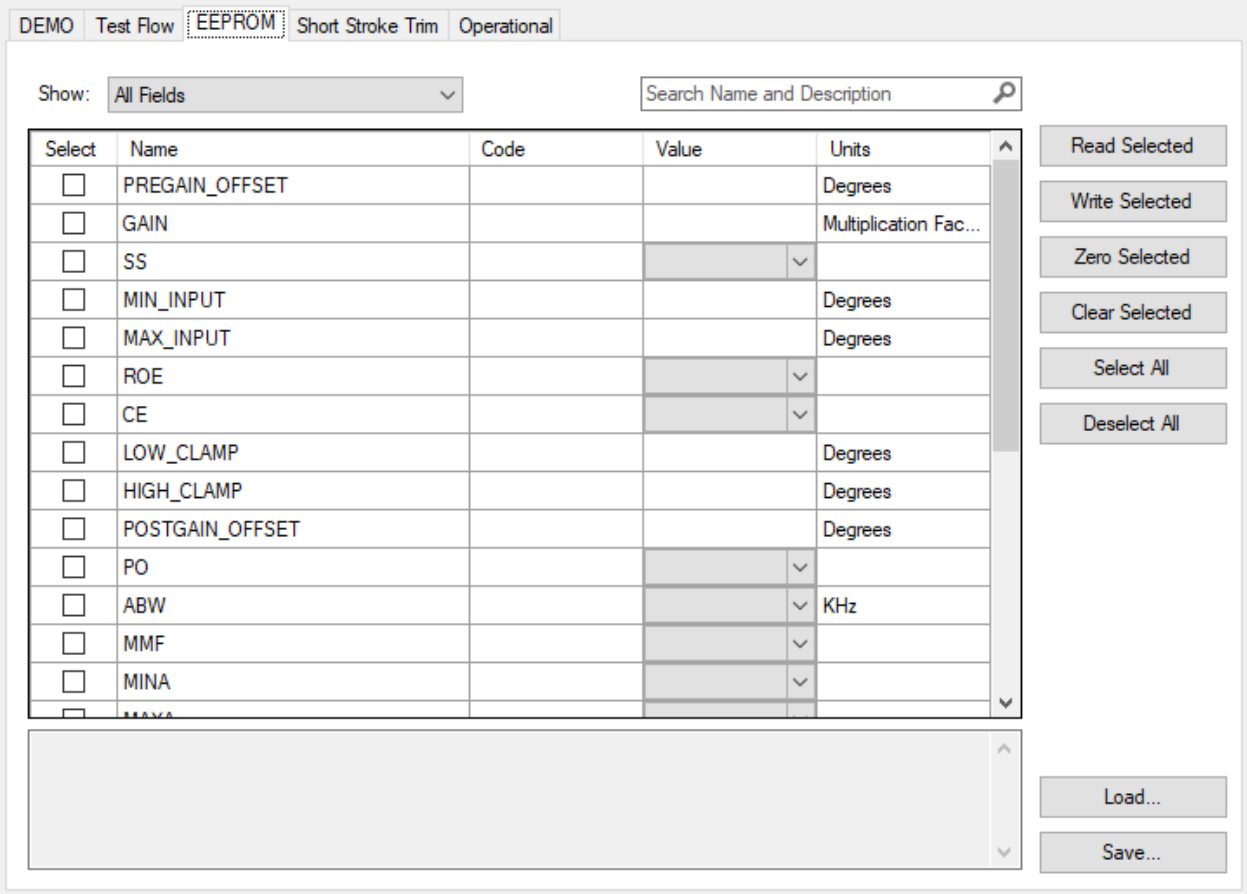


Figure 30: EEPROM

Reference

Show: Pull down Menu

The Show pull down menu is used to select what is displayed in the EEPROM table.

- All Memory Locations
- All Fields
- Short Stroke Fields – Fields used by the short stroke trim

Search Name and Description Test Box

Text entered into this text box will be searched for in the names and descriptions of the fields and memory locations and only those fields and locations that contain it will be displayed.

Table Field

The table displays the EEPROM fields and memory locations in a spreadsheet.

Select Column
This column is used to select the rows on which the buttons are to operate on.

Name Column
The Name column displays the name of the field or memory location. If the mouse is hovered over the cell, a longer version of the name is displayed in a pop-up window.

Code Column
Displays the decimal or hexadecimal version of the data in the field or memory location depending on which menu item is selected, **Display as Decimals** or **Display as Hexadecimals** in the Setup menu. If the field or memory location is writable, double clicking on the cell will allow the user to change the value. The new value is not written to the device until **Write Selected** is clicked. If the mouse is hovered over the cell the range of values is display in a pop-up window if one exists.

Value Column
Displays a version of the data that has been interpreted in the units that are defined for the field or location. For example, the ANGLE field is displayed in degrees. If the field or memory location is writable, double clicking on the cell will allow the user to change the value. The new value is not written to the device until **Write Selected** is clicked. . If the mouse is hovered over the cell the range of values is display in a pop-up window if one exists. This column is only displayed for fields.

Units Column
If the value column has units, then the units are displayed in this column. This column is only displayed for fields.

Description Field
If a selected row has a detailed description, it is displayed in this box.

Read Selected Button
Read all of the selected rows from the device.

Write Selected Button
Write all of the data from the selected rows to the device. This button is enabled when the device EEPROM is writable.

Zero Selected Button
Put a zero into the code column of the selected rows.

Default Selected Button
Put the factory default value into the code column of the selected rows.

Clear Selected Button
Clear the code and value columns of the selected rows.

Select All Button

Select all of the rows in the table.

Deselect All Button

Deselect all of the rows in the table.

Load… Button

Load the EEPROM values from a file.

Save… Button

Save the selected values to a file.

Short Stroke

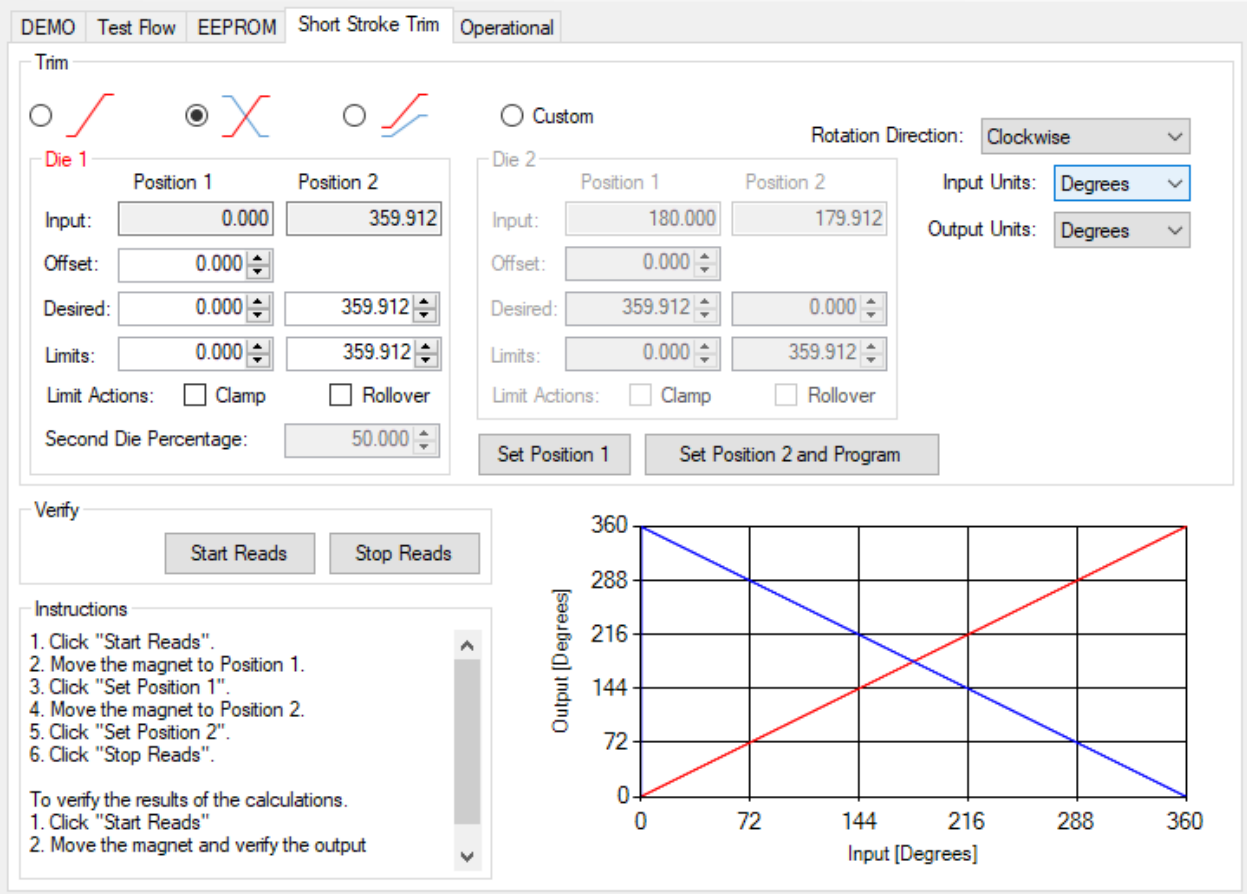


Figure 31: Short Stroke

The short stroke trim tab is used when a full rotation of the target is not possible or desirable.

Program Short Stroke

1. Determine the type of short stroke:
    a. If a single die is to be calculated then choose the single curve. 
    b. If using a dual die part and the output of the dies are to go in opposite directions then choose the 2 curves that cross. 
    c. If using a dual die and the output of the second die is to be a percentage of the first die then choose the 2 curves do not cross. 
    d. If using a dual die and more flexibility is needed in setting the inputs then choose Custom.
2. Enter the desired values for position 1 and position 2.
3. If clamps are wanted, fill in the limit values for position 1 and position 2.
4. Click **Start Reads**.
5. Move the magnet to Position 1.
6. Click **Set Position 1**.
7. Move the magnet to Position 2.
8. Click **Set Position 2**.
9. Click **Stop Reads**.

Verify Short Stroke

1. Click **Start Reads**
2. Move the magnet and verify the output
3. Click **Stop Reads**.

Reference

Rotation Direction

This pop-up menu selects in which direction the magnetic field rotates to cause an increase in angle. The choices are “clockwise” or “counter-clockwise”.

Input Units Pop-up Menu

This pop-up menu selects which units will be used in the input text boxes and displayed in the Input Angle controls. The options are Degrees and Percentage.

Output Units Pop-up Menu

This pop-up menu selects which units will be used in the desired and limit text boxes and displayed in the Output Angle controls. When Analog is selected as the device output, the options are Degrees and Volts. When PWM is selected as the device output, the options are Degrees and Duty Cycle.

Start Reads Button

This button starts a periodic reading from the device and displays the raw values in output graph.

Stop Reads Button
This button stops the periodic reads.

Input Position 1 Text Box
The smallest input value that is to be used.

Input Position 2 Text Box
The largest input value that is to be used.

Offset Text Box
The offset will shift the start of the output by this amount divided by the gain.

Desired Position 1 Text Box
After programming, when the input is at position 1, this is the value that will be output by the A1330.

Desired Position 2 Text Box
After programming, when the input is at position 2, this is the value that will be output by the A1330.

Limits Position 1 Text Box
After programming, when the output value is less than this value, this is the value that will be output by the A1330.

Limits Position 2 Text Box
After programming, when the output value is greater than this value, this is the value that will be output by the A1330.

Limits Actions: Clamp Check Box
After programming, when the output is outside the limits, the output will be the limit.

Limits Actions: Rollover Check Box
After programming, when the output is outside the limits, the output is rolled over at the limits.

Second Die Percentage Text Box
This is only enabled if  is selected and make the second curve this percentage of the first curve.

Set Position 1 Button
Sets position 1. If dual die than reads position 1 for both die 1 and die 2.

Set Position 2 and Program Button
Sets position 2 then calculate the parameters and write them to the device. If dual die than reads position 2 and calculates the desired outputs for both die 1 and die 2.

Output Graph

Shows the output of the A1330. Shows the expected output curve(s) with the settings that have been entered.

Operational

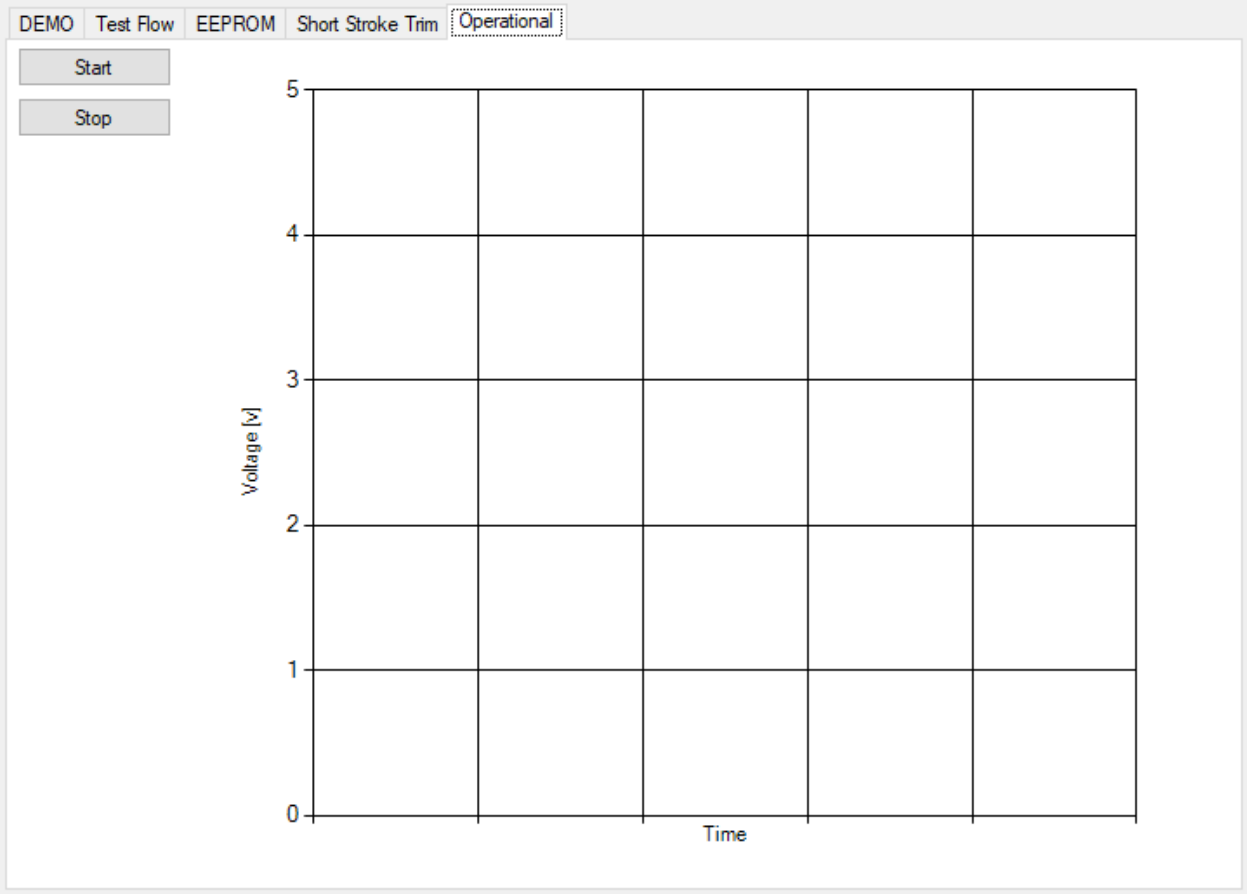


Figure 32: Operation

The Operational tab is used when the full operational aspects of the device including the failure modes.

Reading From the Device

Clicking **Start** will continuously read and insert into the graph the voltage or duty cycle (depending on what output is selected) until **Stop** is clicked.

Reference

Start Button

Start a periodic update for voltage or duty cycle on the page.

Stop Button

Stop the periodic update for voltage or duty cycle on the page.

Input and Output File Formats

Memory

CSV or Comma Separated Value files use the file extension ".csv". Lines can be blank or if they start with # then they are comments.

Each set of parameters will start with the group name. The group name is "EEPROM. The parameters are one per line with the name of the parameter, a comma, and then the value of the parameter.

Text files use the file extension ".txt" and are the same as csv files except instead of commas the parameter name and value are separated by an equal sign.

Example of CSV file

```
# this is a comment line
eeprom
shrtstrk_e,1,true
pregain_offset,512,45.000
gain,512,3
min_input,0,0.000
max_input,1024,90.000
low_clamp,0,0.000
high_clamp,0,360.000
postgain_offset,0,0.000
```

Example of Text file

```
# this is a comment line
eeprom
shrtstrk_e=1,true
pregain_offset=512,45.000
gain=512,3
min_input=0,0.000
max_input=1024,90.000
low_clamp=0,0.000
high_clamp=0,360.000
postgain_offset=0,0.000
```