

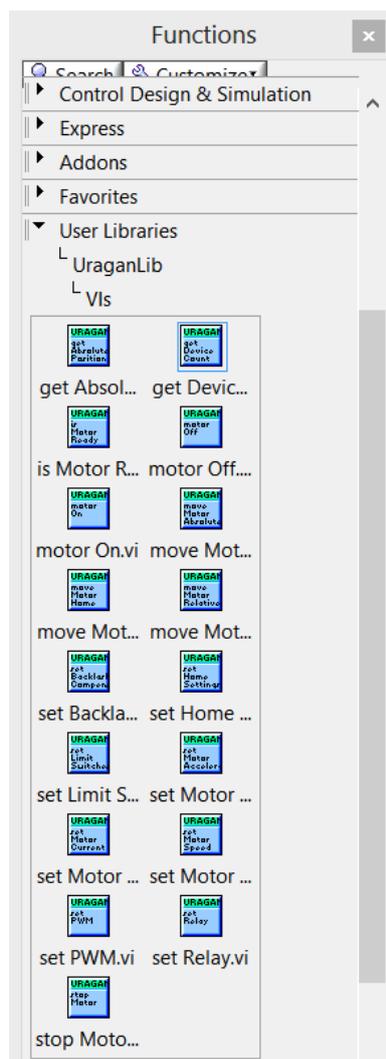
Uragan- μ for LabView

Installation

Follow these steps to install the LabView driver:

1. Download UraganLabView.zip
2. Unzip *UraganLabView.zip* in the *user.lib* directory of your LabView installation
3. Run LabView
4. Select *Tools* \rightarrow *Find VIs on Disk...*
5. Search the *user.lib* directory to find the Uragan LabView VIs

The VIs for interfacing with Uragan- μ devices should now be visible in LabView's *Functions* palette.



An example showing how to use the Uragan VIs can be found in the **/user.lib/UraganLib/example* directory.

Programming interface

VIs and error handling

The LabView driver and programming interface for Uragan- μ devices consists of a set of VIs. Every VI uses the standard LabView error handling technique and each VI has an *Error in* and *Error out* parameter. The error parameters are standard LabView error structures, which provide error feedback and flow control.

Error codes are listed in the table below:

Code	Description	Reason
0	ERROR_NONE	No error
1	ERROR_DEVICE_INDEX_OUT_OF_RANGE	Trying to index a device outside the device index range. For example: If two Uragan- μ devices are connected, only device indices 1 and 2 are valid. All other indices are not valid.
2	ERROR_DEVICE_ALREADY_IN_USE	Trying to connect to a device, which is already opened by another application. Close or disconnect the other application.
3	ERROR_DEVICE_OPEN_FAILURE	The device could not be opened. Check the USB cable connection and make sure the correct USB driver is installed.
4	ERROR_GET_MOTOR_CONFIG	Communication error while trying to read the motor configuration from the device. Check the USB cable connection.
5	ERROR_SET_MOTOR_CONFIG	Communication error while trying to set the motor configuration from the device. Check the USB cable connection.
6	ERROR_GET_MOTION_CONFIG	Communication error while trying to get the motion configuration from the device. Check the USB cable connection.
7	ERROR_SET_MOTION_CONFIG	Communication error while trying to set the motion configuration from the device. Check the USB cable connection.
8	ERROR_MOVE	Communication error while trying to send a move command to the device. Check the USB cable connection.
9	ERROR_MOTOR_ON	Unable to turn the motor driver on. Check the USB cable connection.
10	ERROR_MOTOR_OFF	Unable to turn the motor driver off. Check the USB cable connection.
11	ERROR_MOTOR_STOP	Unable to stop the motor. Check the USB cable connection.
12	ERROR_GET_MOTOR_INFO	Communication error while trying to get the motor status information from the device. Check the USB cable connection.
13	ERROR_SET_PWM	Communication error while trying to set the PWM value. Check the USB cable connection.
14	ERROR_SET_RELAY	Communication error while trying to set the solid state relay. Check the USB cable connection.
15	ERROR_MOTOR_UNINITIALISED	The motor driver is not initialised. This is an unexpected error. Try re-connecting the device.
16	ERROR_MOTOR_NOT_ON	The motor driver is not turned on. First turn on the motor driver using the <i>motor On</i> command.

Code	Description	Reason
17	ERROR_MOTOR_NOT_OFF	The motor driver is not turned off. This is an unexpected error. Try re-connecting the device.
18	ERROR_MOTOR_INVALID_VOLTAGE	The voltage of the external power supply for Uragan- μ is too low. Make sure that the voltage is at least 22 V.
19	ERROR_MOTOR_ERROR	The motor driver encountered an error. Make sure that the stepper motor phases are connected correctly.
20	ERROR_MOTOR_BUSY	The motor is still busy moving and the next command cannot be executed until the current move operation is completed. Wait until the current move operation is completed.
21	ERROR_PARAM_OUT_OF_RANGE	The parameter passed into one of the Uragan VIs is out of range. Ensure that valid parameter values are used.
22	ERROR_SET_LIMITS	Communication error while trying to set the limit switch settings. Check the USB cable connection.
23	ERROR_SET_BACKLASH_COMPENSATION	Communication error while trying to set the backlash compensation settings. Check the USB cable connection.
24	ERROR_SET_HOME_SETTINGS	Communication error while trying to set the home position settings. Check the USB cable connection.
25	ERROR_ZERO_MOTOR_POSITION	Communication error while trying to reset the motor position to zero. Check the USB cable connection.
26	ERROR_INITIALSING	The device cannot be initialised. Check the USB cable connection.
27	ERROR_NOT_INITIALISED	The initialise VI was not called. First call the initialise VI.
28	ERROR_DEVICE_BUSY_OR_NOT_FOUND	The device being referenced via the device ID parameter is either busy or not available.
29	ERROR_RETRIEVING_DEVICE_SIGNATURE	The device signature could not be retrieved. Check the USB cable connection.
30	ERROR_DEVICE_SIGNATURE	There is an error in the device signature. Check the USB cable connection.
31	ERROR_UNSUPPORTED_FIRMWARE_VERSION	The device uses an older unsupported firmware version. Update the firmware using Uragan Studio. Alternatively, the LabView driver might be outdated. Get the latest LabView driver from the Synertronic Designs web page.
32	ERROR_UNSUPPORTED_FEATURE	The connected device does not support the feature which an VI tries to access.
33	ERROR_SET_PWM_MODE	There was an error setting the PWM output to indicate the motor steps. Check the USB cable connection.

In order to communicate with Uragan- μ devices, the following VIs are provided:

- initialise
- get Device Count
- get Device Index
- set Limit Switches
- set Home Settings
- set Backlash Compensation
- set Motor Current
- set Motor Speed
- set Motor Acceleration
- motor On
- motor Off
- move Motor Home
- move Motor Relative
- move Motor Absolute
- stop Motor
- is Motor Ready
- get Absolute Position
- zero Motor Position
- set PWM
- set PWM Step Mode
- set Relay

In the following sections, each of the VIs is described in more detail.

intialise

Execute this VI before any other commands are sent to the target device. It ensures that the LabView driver is reset and in a consistent state.

get Device Count

Returns the number of connected Uragan- μ instruments.

Parameter	Type	Description
pCount	Integer (out)	Number of connected devices.

The pCount value can be used to determine the range of device indices that can be used. For example: When three devices are connected, the individual devices can be accessed using device indices 1, 2 and 3.

get Device Index

Tries to find the device index of the Uragan- μ instrument with the given serial number.

Parameter	Type	Description
serialNumber	String (in)	Serial number of the target Uragan- μ .
deviceIndex	Integer(out)	The device index of the Uragan- μ instrument with the given serial number

If the device with the given serial number cannot be found, an error code is returned and deviceIndex will be equal to -1.

set Limit Switches

Specifies the limit switch settings.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
startChannel	Integer (in)	Specifies the channel number, to which the start limit switch is connected. Range: 1 .. 4
startChannelMode	Integer (in)	Specifies the input configuration for the start limit channel. 0 No limit switch 1 High-Z, triggered on rising edge 2 High-Z, triggered on falling edge 3 Pull-down, triggered on rising edge 4 Pull-down, triggered on falling edge 5 Pull-up, triggered on rising edge 6 Pull-up, triggered on falling edge
endChannel	Integer (in)	Specifies the channel number, to which the end limit switch is connected. Range: 1 .. 4
endChannelMode	in	Specifies the input configuration for the end limit channel. 0 No limit switch 1 High-Z, triggered on rising edge 2 High-Z, triggered on falling edge 3 Pull-down, triggered on rising edge 4 Pull-down, triggered on falling edge 5 Pull-up, triggered on rising edge 6 Pull-up, triggered on falling edge

set Home Settings

Specifies the home position settings. It is possible to specify one of the limit switches as a home position. For this feature to work, at least one limit switch must be configured.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
position	Integer (in)	Specifies which limit switch is used as the home position. 0 Start limit switch 1 End limit switch
homelsAbsoluteZero	Bool (in)	When set to TRUE the absolute position counter of Uragan- μ will be set to zero when the home position is reached. When set to FALSE the absolute position will be left unchanged.
accurateHomeSearch	Bool (in)	When set to TRUE an accurate home search is performed. When the <i>move Motor Home</i> command is executed, Uragan-m will move the motor with the <i>fastSpeed</i> to the target limit switch. When the target switch is reached the motor will be moved with the <i>slowSpeed</i> until the limit switch opens. When set to FALSE the <i>slowSpeed</i> home search will be omitted.
fastSpeed	Integer (in)	Specifies the input fast motor speed to reach the home position. Note: This speed is specified in <i>full-steps/second</i> .
slowSpeed	Integer (in)	Specifies the input slow motor speed for the accurate home search. Note: This speed is specified in <i>full-steps/second</i> .

set Backlash Compensation

Specifies the backlash compensation. In order to improve the accuracy of positioning systems, the backlash compensation can be specified. For this, a motor direction must be specified as the primary or non-backlash direction. When the motor is moved into the opposite direction, it will first move past the target position and move back in the non-backlash direction. The amount of overshoot, by which the motor moves past the target position before returning, is specified by the backlash step count.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
enable	Bool (in)	When set to TRUE, the backlash compensation is enabled.
direction	Integer (in)	The primary or non-backlash direction. 0 Positive direction (forward) 1 Negative direction (backward)
backlashStepCount	Integer (in)	The backlash overshoot. Note: The step count is specified in <i>micro-steps</i> .

set Motor Current

Specifies the motor current in [mA]. To reduce power consumption and motor heating, the holding current (i.e. when the motor is not moving) can be reduced.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
mA	Integer (in)	Motor current in [mA]. Range: 0 .. 5000 mA
enableReducedHoldCurrent	Bool (in)	When set to TRUE, the motor holding current will be 50% of the normal motor current specified above. When set to FALSE the holding and normal motor current will be the same.

set Motor Speed

Specifies the motor speed and micro-stepping value.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
fullStepsPerSecond	Integer (in)	Specifies the motor speed. Note: The speed is specified in <i>full-steps/second</i> . Range: 1 .. $\frac{20,000}{\text{micro-stepping}}$
microStepping	Integer (in)	Specifies the amount of micro-stepping. 0 Full-step (x1 micro-stepping) 1 Half-step (x2 micro-stepping) 2 x4 micro-stepping 3 x8 micro-stepping 4 x16 micro-stepping

set Motor Acceleration

Specifies the motor acceleration.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
fullStepsPerSecondSquared	Integer (in)	Specifies the motor acceleration. Note: The speed is specified in <i>full-steps/second²</i> . If zero is specified, no acceleration will be used and the motor will start moving at full speed. Max. acceleration: limited to 500,000 Min. acceleration: limited to a ramp-up time of 5 seconds.

motor On

Turns the motor driver on and enters the motor hold state. If the motor is already on, the command is ignored.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

motor Off

Turns the motor driver off. If the motor is already off, the command is ignored.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

move Motor Home

Moves the motor to the home position. The home settings must be correctly specified for this command to execute correctly. This command will fail if the motor is not turned on or if the motor is busy moving.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

move Motor Relative

Moves the motor relative to the current position by the given number of steps. This command will fail if the motor is not turned on or if the motor is busy moving.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
steps	Integer (in)	The relative number of steps. Note: The steps are specified in <i>micro-steps</i> .

move Motor Absolute

Moves the motor to the given absolute position. This command will fail if the motor is not turned on or if the motor is busy moving. This feature works best in conjunction with a valid home position. It is advisable to enable the *homeIsAbsoluteZero* function and to use the home position as the absolute zero position.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
steps	Integer (in)	The absolute position. Note: The steps are specified in <i>micro-steps</i> .

stop Motor

Stops the motor and enters the motor hold state. If the motor is already stopped, the command is ignored.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

is Motor Ready

Checks if the motor is ready to receive the next move command.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
isReady	Bool (out)	Return TRUE if the motor is ready to receive the next move command. Return FALSE if the motor is busy moving.

This VI can be used in a while-loop to wait until the motor has stopped moving. **Note:** It is very important to also add a short wait period (between 50 and 100 ms) in the while-loop. Without a wait period, the USB driver will be flooded with requests. This will slow down the PC and can cause the application to crash. See the example provided together with the LabView driver.

get Absolute Position

Returns the absolute position as tracked by the target Uragan- μ device..

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
pPos	Integer (out)	The absolute position. Note: The position is specified in <i>micro-steps</i> .

zero Motor Position

Sets the motor position to zero.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

set PWM

Sets the PWM duty cycle.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
percent	Integer (in)	The duty cycle in [%].

set PWM Step Mode

Sets the PWM output to generate a positive pulse for each motor step.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount

set Relay

Turns the solid-state relay on or off.

Parameter	Type	Description
deviceIndex	Integer (in)	The target device index. Range: 1 .. pCount
on	Bool (in)	When set to TRUE, the solid-state relay is turned on. Otherwise, the relay is turned off.

Tips

Speed and acceleration parameters are all specified in full-steps. Position and step counts are all specified in micro-steps. In order to keep step and position values consistent, it is advisable to configure the micro-step setting at the beginning of your LabView program and to keep it unchanged. When the micro-step setting is changed during the LabView program, all step and position values will be interpreted using the new micro-step setting. This can cause problems, when working with absolute positions.

When using absolute positions, it is highly recommended to use the home position as the absolute zero position. For this, at least one limit switch must be configured and the for the home settings the *homeIsAbsoluteZero* Boolean flag must be enabled. Alternatively, the motor position can be manually set to zero using the *zero Motor Position VI*.

IMPORTANT NOTICE

Synertronic Designs reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to Synertronic Designs' terms and conditions of sale supplied at the time of order acknowledgment.

Synertronic Designs assumes no liability for applications assistance or customer product design. Customers are responsible for their applications using Synertronic Designs products. To minimize the risks associated with customer applications, customers should provide adequate operating safeguards.

Reproduction of information in Synertronic Designs data sheets, summary notes and brochures is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. Synertronic Designs is not responsible or liable for such altered documentation.

Synertronic Designs on the web: www.synertronic.co.za

E-mail: info@synertronic.co.za

Postal address: Kaneel Cr 34
Stellenbosch
7600
South Africa