

MBARI SideARM MSP430 Environmental Processor (envp) Software

Functional and Interface Specification

Revision History

Version	Date	Comments
0.1	03/28/03	Initial Revision
0.2	04/07/03	First revision incorporating review comments
0.3	05/23/03	Minor revisions and added material related to RTC and countdown alarm functionality
1.6	02/15/04	Revised to include new SideARM v4 functionality including KVH C100 Compass and turns count. NOTE: changed version number to match CVS version.

This document describes the host interface and software commands used to interact with and obtain data and alarms from the SideARM Environmental Processor (envp).

Host Interface

The envp interfaces to the SideARM via a planar serial connection to serial port `/dev/ttySX15` on the SideARM. Because of the relatively modest amount of data that is exchanged between the SideARM and the envp, the connection data rate is 9600 baud (8 data bits, 1 stop bit, no parity). The interface utilized consists entirely of printable ASCII characters, so this interface could be used by a person typing at a terminal program as well as it's intended programmatic use. A command is provided to enable or disable the echoing of characters by the envp to facilitate this dual use. The returned data format has been made similar to NMEA strings. The envp will prompt the user or program to issue a command by sending a sequence of [CR] [LF] [>] [Space]. After this sequence is sent, the envp is ready to receive commands. These commands are CR-LF terminated strings as described in the Commands section

Sensor Sampling

The envp samples each of it's five sensors, temperature, pressure, humidity, and ground fault (highside, lowside) current at a uniform rate specified through the command interface. The current value of each sensor is updated at this time. Each time the sensors are sampled, their value is optionally checked against an upper and lower constraint. If desired, an alarm condition may be entered if these sensor values fall outside a constrained range specified by the user. This constrained range consists of a low and high limit value set by the user utilizing the "set limits" command. The envp considers the constraint range to be valid once the user has set limit values where (`lowLimit < highLimit`). After this

point, alarms may be enabled in software using the “set amask” command.

If enabled in software, sensor values will be checked against a valid constraint range and an alarm condition raised if the current value falls outside the constraint range. The alarm condition will cause the envp to assert the SideARM attention signal. The sensor value that caused the alarm condition will be latched (held) and all subsequent measurements will be discarded until the alarm condition is cleared for the particular sensor. The attention signal will remain asserted until there are no sensors with alarm conditions and alarms enabled.

Ground Fault Scan Algorithm

The envp firmware implements a ground fault scanning algorithm that is active when the application is started. In version 2.0 and greater firmware, this cyclical check can be halted for diagnostic purposes, and then restarted. This is accomplished with the 'gfscan off' and 'gfscan on' commands respectively. When the scan is running

1. assert CHECK_HIGH_SIDE
2. wait for a settling time (settable between 1 and 60 seconds in 1 sec increments)
3. read A/D channels for HIGH_SIDE_FAULT and LOW_SIDE_FAULT
4. de-assert CHECK_HIGH_SIDE
5. wait for settling time (same value as in Step 2)
6. read A/D channels for HIGH_SIDE_FAULT and LOW_SIDE_FAULT
7. subtract Low_Side value in Step 3, from Low_Side value in Step 6
8. Subtract High_Side value in Step 6, from High_Side value in Step 3
9. If either Low_Side or High_Side value is greater than the
10. alarm threshold, assert the Ground Fault Alarm
11. report Low_Side and High_Side values, and status of GF Alarm
12. repeat

Both the settling time mentioned above and the period of the operation can be changed at runtime using the 'set gfsettle' and 'set gfperiod' commands. By default, the settling time and period are set to 10 and 60 seconds respectively.

Ground Fault Diagnostic Commands

The SideARM v4 hardware includes a current injection circuit to test the operation of the ground fault circuitry on the board. Commands have been added to allow manual control of this fault injector, the highside selector signal, and the ground fault scanning algorithm.

In order to perform manual tests, a way to disable and re-enable the normal ground fault scanning algorithm was needed. Two commands, 'gfscan off' and 'gfscan on' provide a way to do this. During the time which the ground fault scan is not active, ground fault current values in the normal data stream are returned as -1.0. As soon as the scanning is re-enabled, the first measurement replaces this value.

When the ground fault scanning mechanism is disabled, two commands may be used to manually

maipulate the fault injector and highside selection signal. 'gftest on' and 'gftest off' activate and deactivate the fault injection circuitry. 'hiside on' and hiside off' select and de-select the highside current measurement. Lastly, the 'show gf' command may be used to obtain an instant reading of the ground fault current values when the normal ground fault scan mechanism is disabled.

The following is a demonstration of how to test the ground fault mechanism. The following description assumes that you have configured the serial port to the MSP430 by issuing the 'stty -F /dev/ttySX15 9600' command at the command prompt. This must be done as root. The rest of these commands can be issued from a user account (i.e. Running picocom), but being logged in as root is recommended.

```
sh-2.03 # picocom /dev/ttySX15
```

After you issue this command, press return again and the msp430 should respond with a prompt ('> '). We're going to manually look at the ground fault current, so you need to disable the default ground fault scanning routine. Enter the 'gfscan off' command:

```
> gfscan off
```

Ensure that you're measuring the lowside, by setting the switch. This is done with the 'hiside off' command:

```
> hiside off
```

You can now see the ground fault measurements with the 'show gf' command:

```
> show gf
```

```
High side analog: [0.424805] ADC[3]
Low side analog: [1.652310] ADC[10]
```

Next, turn the hiside switch on and measure the high side:

```
> hiside on
```

```
> show gf
```

```
High side analog: [0.424805] ADC[3]
Low side analog: [1.527210] ADC[9]
```

If needed, you can convince yourself of correct operation of the measurement circuit and commands by using the ground fault current injector. This is controlled with the 'gftest' command. Turn it on to induce a mild ground fault, then turn it off for testing:

```
> gftest on <-- turn on injector
```

```
> show gf
```

```
High side analog: [105.635000] ADC[746] <-- high side fault
Low side analog: [1.777410] ADC[11]
```

```
> hiside off <-- switch to low side
```

```
> show gf
```

```
High side analog: [0.424805] ADC[3]
```

```
Low side analog: [101.984000] ADC[812] <-- low side fault
```

```
> gftest off <-- turn fault injector off
```

To exit from picocom, press [Ctrl]-B, then X. To power down the system, issue the 'halt' command and wait for all the LEDs to be extinguished. When performing the tests, please record both the analog current value and the ADC count (ADC[x]) -- this will help us in characterizing the results.

KVH C100 Compass and Turns Counter

The KVH Compass and turns counter support includes 4 new commands, three of which are utilized for diagnostic and configuration purposes. The first new command is the 'set turns' command which, as its name implies, initializes the turns counter. This command expects one signed integer argument. If the number specified is negative, the number of counterclockwise turns will be initialized to the absolute value of this number and the clockwise turns initialized to zero. If the number is positive, the number of clockwise turns is initialized to this number and counterclockwise turns are initialized to zero.

'kvh get' and 'kvh put' commands send and retrieve a character string from the instrument. These commands can be useful for sending one-time commands, such as calibration commands, to the instrument which are not part of normal operation.

Command Descriptions

As noted in the first section, commands may be issued to the system when a prompt is observed. Commands should be terminated with the normal CR-LF sequence. Commands should be issued in lower case, as they are documented below, although case insensitivity will probably be added shortly if requested. Erroneous input is currently ignored by the command interpreter. Return of error codes will be added in a later release.

A summary of the commands is given in the table below.

Command	Subcommand	Description
set	amask	Sets the alarm enable mask
	levels	Sets the low and high constraints for each sensor, outside of which results in an alarm condition if alarms are enabled
	echo	Sets whether characters are echoed (for a person at a terminal) or not (programmatic interface)
	average*	Sets the number of samples over which a moving average filter is applied to the data
	period	Sets the interval (number of seconds) between samples

		(default = 10).
	date	Sets the date maintained by the RTC (depricated in v2.0 and above)
	time	Sets the time maintained by the RTC (depricated in v2.0 and above)
	countdown	Starts a countdown timer which can trigger an alarm if enabled
	error	Specifies whether error messages are output (for compatibility with early versions, error message output is disabled by default)
	gfperiod	Integral argument specifies the ground fault scanning period in seconds (default: 60)
	gfsettle	Integral argument specifies the delay after changing the SET_HIGH_SIDE signal in the ground fault scanning algorithm (default: 10)
	turns	Allows the initialization of the number of turns recorded by the KVH C100 compass/turns counter. Positive values are registered as clockwise turns, while negative values are registered as counterclockwise turns.
get	data	Returns a string with all data values
	levels	Returns all of the constraint values in a string
	amask	Returns the current alarm enable mask
	alarm	Returns the state of each alarm
	version	Returns the current firmware version
	date	Returns the date maintained by the RTC (depricated in V2.0)
	time	Returns the time maintained by the RTC (depricated in V2.0)
clear	alarm	Clear the alarm for the sensor(s) specified in the mask
poll		Causes an immediate sample of sensors in addition to normal periodic sampling
kvh	on	Turns on the FET power switch supplying +12VDC to the KVH C100 compass
	off	Turns off the FET power switch supplying +12VDC to the KVH C100 compass
	put	Sends a freeform ASCII string specified after the command to the KVH C100 via the serial port
	get	Returns the contents of the KVH serial receive buffer

gfscan	off	Stops the default GFI scanning algorithm so that manual diagnostics may be performed.
	on	Restarts the GFI scanning algorithm if it has been stopped with the 'gfscan off' command.
hside	off	Clears the SET_HIGH_SIDE signal – to be used after the 'gfscan off' command has been issued to stop the automatic scanning.
	on	Sets the SET_HIGH_SIDE signal – to be used after the 'gfscan off' command has been issued to stop the automatic scanning.
gftest	on	Activates the ground fault injection circuitry to check the functioning of the ground fault sensing. This command should only be used after the automatic ground fault scan has been stopped with the 'gfscan off' command.
	Off	Deactivates the ground fault injection circuitry to check the functioning of the ground fault sensing. This command should only be used after the automatic ground fault scan has been stopped with the 'gfscan off' command.

* = not currently implemented

The following sections specify details specific to each command.

Set Commands

Set commands allow the user to modify the alarm enable mask that triggers an alarm condition if a sensor value falls outside of it's constraint values. These constraint values are also modified with a set level command.

Command	set amask
Arguments	Alarm mask specified as an 8-bit hex constant where the bit positions mean the following: 0x01 – Pressure 0x02 – Temperature 0x04 – Humidity 0x08 – Ground Fault (low) 0x10 – Ground Fault (high) 0x20 – Countdown timer 0x40 – Compass 0x80 – Compass turns counter The user can optionally prepend '0x' to the resulting hex value, similar to the ANSI C standard for hex constants.

	Note that only bits that correspond to actual sensor values will be reflected in subsequent 'get amask' commands. One's written to bit positions other than those specified above will be ignored.
Return String	(none)
Description	Set amask accepts an 8-bit hex constant and uses this as an enable mask for which sensors, whose values fall outside the constraint values, set with the 'set levels' command, should result in an alarm condition
Related Command	get amask

Command	set levels
Arguments	A composite string consisting of any combination of the low and high constraint values for each sensor expressed as a constraint list in tag/data format with each element of the list separated by commas. \$PALEVEL, <constraint list in tag/data format> The constraint list elements consist of the following: PL[presLow] PH[presHi] HL[humLo] HH[humHi] TL[tempLo] TH[tempHi] GFLL[gfLowsideLo] GFLH[gfLowsideHi] GFHL[gfHighsideLo] GFHH[gfHighSideHi] CL[compassLo] CH[compassHi] TCL[turnsCountLo] TCH[turnsCountHi] where the bracketed values are floating point constants representing the low and high constraints for the sensors. The constraint values have the following units: temperature: degrees celsius pressure: millibars humidity: relative percent ground fault current: microAmps compass: heading in degrees

	turns counter: integral number of turns
Return String	(none)
Description	Composite argument string contains the low and high constraints for each of the 5 sensor values. If an alarm is enabled, measured values outside this constraint range triggers an alarm condition
Related Command	get levels

Command	set echo
Arguments	“on” = echo all characters received “off” = echo no characters received, but still output prompt when ready to process a new command
Return String	(none)
Description	Sets whether characters sent to the envp are echoed back (for a person at a terminal querying the envp interactively) or not (programmatic interface)
Related Command	(none)

Command	set error
Arguments	“on” = output error messages for invalid commands or invalid command parameters “off” = output no error messages (compatibility mode)
Return String	(none)
Description	This command determines whether error messages are returned by the envp for invalid commands or invalid command parameters.
Related Command	(none)

Command	set period
Arguments	A single integer specifying the integral interval between samples in seconds.
Return String	(none)
Description	Specifies the sampling period by varying the interval between samples. The default value is 10. A value less than 2 is not

	recommended.
Related Command	(none)

Command	set countdown
Arguments	A single integer argument indicating the number of seconds to count down before issuing an alarm
Return String	(none)
Description	Sets a countdown timer if enabled to assert an alarm using the same mechanism as the environmental sensors. Thus to assert the SideARM attention signal, a countdown alarm must be enabled first with the set amask command.
Related Command	(none)

Get Commands

Get commands can be used to get the constraint levels used to evaluate the sensor values. The Get amask command is also employed to get the 8-bit alarm enable mask value.

Command	get levels
Arguments	(none)
Return String	A composite string consisting of all combinations of the low and high constraint values separated by commas in the following format: \$PALEVEL, PL[presLow], PH[presHi], HL[humLo], HH[humHi], TL[tempLo], TH[tempHi], GFLH[gfLowsideLo], GFLH[gfLowsideHi], GFHL[gfHighsideLo], GFHH[gfHighsideHi], CL[headingLo], CH[headingHi], TCL[turnsCountLo], TCH[turnsCountHi] *[checksum][CR][LF] where the bracketed values (except checksum, CR, LF) are floating point constants representing the low and high constraints for the sensors. A checksum is computed on all values excluding the '*' character, the checksum itself, and the terminating CR-LF sequence. The units for each of the measurements are the following: temperature: degrees celsius pressure: millibars humidity: relative percent ground fault current: microAmps Compass heading: degrees Compass turns counter: integral number of turns
Description	Returns a Composite string containing the low and high constraints for each of the 5 sensor values. If an alarm is enabled, measured values outside this constraint range triggers an alarm condition
Related Command	set levels

Command	get amask
Arguments	(none)
Return String	Returns a composite string in the following format \$PAMASK, [alarmMask], *[checksum] where [alarmMask] is an 8-bit hex constant where the bit positions mean the following: 01 – Pressure 02 – Temperature

	04 – Humidity 08 – Ground Fault (low) 10 – Ground Fault (high) 20 – Countdown timer 40 – Compass 80 – Compass turns counter A checksum is computed on all values excluding the '*' character, the checksum itself, and the terminating CR-LF sequence.
Description	get amask returns the currently programmed 8-bit hex constant used as an alarm enable mask. Note that the mask value is zero upon system startup.
Related Command	set amask

Command	get alarm
Arguments	(none)
Return String	Returns a composite string in the following format “\$PALARM, [alarms], *[checksum]” where [alarms] is an 8-bit hex constant where the bit positions mean the following: 01 – Pressure 02 – Temperature 04 – Humidity 08 – Ground Fault (low) 10 – Ground Fault (high) 20 – Countdown timer expired 40 – Compass 80 – Compass turns counter Each of the bit positions is set or clear based on whether the alarm is triggered or not. A checksum is computed on all values excluding the '*' character, the checksum itself, and the terminating CR-LF sequence.
Description	get amask returns the current status of each alarm as an 8-bit hex constant.
Related Command	clear alarm

Command	get data
Arguments	(none)
Return String	A composite string containing all of the measured data values and of the format: “\$PEDATA P[pressure], H[humidity], T[temperature], GFL [gfLowsideCurrent], GFH[gfHighsideCurrent], C[compass], TC [turns count] *[checksum][CR][LF]” where the bracketed values (except checksum, CR, LF) are floating point constants representing the last measurement value. In the case of values where an alarm has been enabled, the first value that falls outside the constraint range specified by the set levels command is latched and triggers an alarm condition for that sensor. The checksum includes all characters up to the asterisk in the string. The units for the values are the following: temperature: degrees celsius pressure: millibars humidity: relative percent ground fault current: microAmps Compass heading: degrees Compass turns counter: integral number of turns
Description	Returns the last measurement value of each sampled sensor.
Related Command	(none)

Command	get version
Arguments	(none)
Return String	A string of the format “\$PSID, V[version][toolchain] *[checksum][CR][LF]” indicating the version of the envp firmware
Description	Returns the firmware version and the toolchain used to build the application image. Toolchains include IAR Systems ('I') and Quadravox ('Q')
Related Command	(none)

Command	get time
Arguments	(none)
Return String	A string of the format “\$PTIME12, hh:mm:ss[a/p] *[checksum]

	[CR][LF]" indicating the hour, minute, and seconds of the current time maintained by the RTC. Note that the time is presently returned in 12-hour format with an "a" or "p" appended for AM and PM times respectively.
Description	Returns the current date maintained by the RTC (depricated in v2.0)
Related Command	set time

Clear Command

There is only one clear command and it's purpose is to acknowledge any alarm condition that has occurred. The attention SideARM signal will remain asserted when an alarm condition has been triggered until all alarm conditions have been cleared.

Command	clear alarm
Arguments	Alarm mask specified as an 8-bit hex constant where the bit positions mean the following: 01 – Pressure 02 – Temperature 04 – Humidity 08 – Ground Fault (low) 10 – Ground Fault (high) 20 – Countdown timer
Return String	(none)
Description	Clears the alarms indicated by the 8-bit hex argument. The sensor value constraints will be evaulated on the next sample and an alarm condition entered if the value falls outside of the constraint values if specified and if the alarm is enabled.
Related Command	set amask, clear amask

Other Commands

Command	poll
Arguments	(none)
Return String	(none)
Description	Causes an immediate sample of sensors in addition to normal periodic sampling. After issuing a poll command the user or controlling program may issue a 'get data' command to retrieve the updated data.

Related Command	get data
------------------------	----------

KVH C100 Compass and Turns Counter commands

The 'set turns' command allows the user to initialize the number of turns if desired. The other commands allow manual operations to be performed on the KVH C100 compass.

Command	Kvh on
Arguments	(none)
Return String	(none)
Description	Turns on the FET power switch supplying +12VDC to the KVH C100 compass
Related Command	Kvh off

Command	Kvh off
Arguments	(none)
Return String	(none)
Description	Turns off the FET power switch supplying +12VDC to the KVH C100 compass
Related Command	Kvh on

Command	Kvh put
Arguments	A freeform ASCII string
Return String	(none)
Description	The freeform ASCII string following the command is sent to the KVH C100 Compass via the serial port. This command can be useful for entering one-time commands like starting a calibration or resetting the unit.
Related Command	Kvh get

Command	Kvh get
Arguments	(none)
Return String	A freeform ASCII string of up to the serial buffer size received from the KVH compass via the serial port.
Description	This command can be useful for debugging sent commands or checking the status of commands issued with the 'kvh put' command.
Related Command	Kvh put

Command	Set turns
Arguments	Integral number of clockwise turns
Return String	(none)
Description	This command allows the number of turns to be preset. A signed integral argument is expected indicating the number of clockwise turns. If the user wishes to specify counterclockwise turns, a negative number should be specified.
Related Command	

Ground Fault and GF Diagnostics commands

The first two commands ('set gfsettle' and 'set gfperiod') can be used to modify the behavior of the automatic ground fault scanning algorithm. The 'gfscan off' command allows the user to temporarily stop the automatic ground fault scan so that manual diagnostic commands can be issued. The 'hiside' and 'gftest' commands may then be used to manually manipulate the state of the CHECK_HIGH_SIDE and GFI injection signals respectively.

Command	set gfsettle
Arguments	A single integer specifying the integral settling time in the ground fault scan algorithm. .
Return String	(none)
Description	Specifies the settling time in seconds observed after changing SET_HIGH_SIDE in the ground fault scan algorithm. This time should always be on the order of 1/5 the gfperiod value. The default value is 10 seconds.
Related Command	Set gfperiod

Command	set gfperiod
Arguments	A single integer specifying the integral interval between ground fault scan cycles in seconds.
Return String	(none)
Description	Specifies the ground fault scanning sampling period in seconds. The default value is 60.
Related Command	(none)

Command	gfscan off
Arguments	(none)
Return String	(none)
Description	This command stops the default GFI scanning algorithm. After this command has been issued, ground fault current values reported by the 'get data' command will be reported as '-1.0' until the algorithm is restarted. The purpose of this command is to allow manipulation of the diagnostic commands independent of the ground fault scanning algorithm.
Related Command	gfscan on

Command	gfscan on
Arguments	(none)
Return String	(none)
Description	This command restarts the default GFI scanning algorithm if it has been stopped with the 'gfscan off' command. GFI current values will be reported after the first scan has completed.
Related Command	gfscan off

Command	hiscide off
Arguments	(none)
Return String	(none)

Description	This command, along with 'hiside on' allow manual manipulation of the CHECK_HIGH_SIDE signal for diagnostic purposes. This command will deassert CHECK_HIGH_SIDE and should only be used after the normal ground fault scan has been stopped via the 'gfscan off' command.
Related Commands	hiside on gftest on, gftest off gfscan off

Command	hiside on
Arguments	(none)
Return String	(none)
Description	This command, along with 'hiside off' allow manual manipulation of the CHECK_HIGH_SIDE signal for diagnostic purposes. This command will assert CHECK_HIGH_SIDE, and should only be used after the normal ground fault scan has been stopped via the 'gfscan off' command.
Related Commands	hiside off gftest on, gftest off gfscan off

Command	gftest on
Arguments	(none)
Return String	(none)
Description	This command activates the ground fault injection circuitry which is present on the SideARM v4 for checking the functioning of the ground fault sensing. This command should only be used after the normal ground fault scan has been stopped via the 'gfscan off' command.
Related Commands	gftest off hiside on, hiside off gfscan off

Command	gftest off
Arguments	(none)
Return String	(none)
Description	This command de-activates the ground fault injection circuitry which is present on the SideARM v4 for checking the functioning of the ground fault sensing. This command should only be used after the normal ground fault scan has been stopped via the 'gfscan off' command.
Related Command	gftest on hiside on, hiside off gfscan off

CVS Version \$Id: envp.html,v 1.7 2004/07/29 19:08:47 timm Exp \$
 Author: Tim Meese (timmm@mbari.org)