

Slocum Glider G3 SM32 New Processor Guide

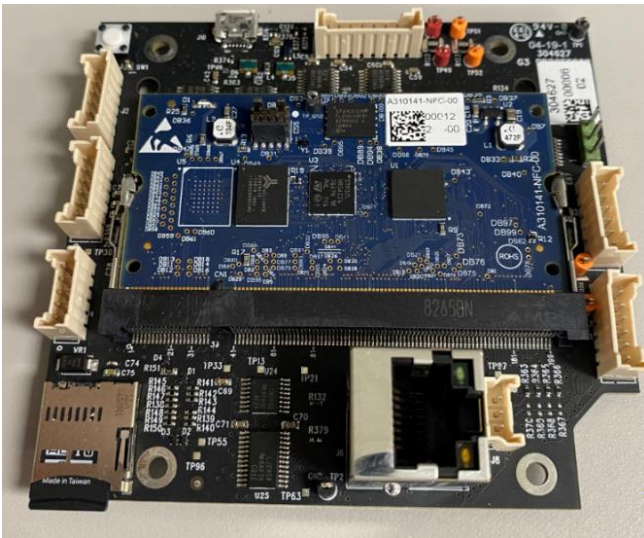
March 2020 DRAFT A

In the Spring of 2020, Teledyne Webb Research began shipping Slocum Gliders with a new processor. This processor replaced the original Persistor processor as the onboard, low-power controller of the Slocum Glider. The goal of the project was to replace the processor and provide the user with the existing control and interface structure while consuming the same amount of energy. The user must be aware of critical differences. This guide will highlight the differences.

It is highly recommended that new users attend a Slocum Glider training session. Contact glidersupport@teledyne.com to inquire about upcoming training sessions or with any questions about this document or the new processor.

Hardware

The new processor is a STM32L4 CPU utilizing OpenRTOS running up to 120 MHz, 8 Mbytes RAM, 32 Mbytes flash for missions and configuration files supported by Swissbit SDHC cards up to 32 Gbytes for logging data. The use of consumer SD cards is not recommended.



Software

Software support will continue in the same manner. New releases of Flight and Science code starting with version RELEASE_10_0 can be found at the Slocum Glider user forum. As always, Teledyne Webb Research (TWR) highly recommends reading the 'readme.txt' before installing new software on a vehicle.

<https://datahost.webbresearch.com/>

<https://datahost.webbresearch.com/files.php?cwd=/glider/production/readme.txt>

Main Differences

There are two primary differences in the software.

1. The operating system is now called 'GliderShell' instead of 'PicoDos.' All related commands have consistently changed (i.e. 'exit pico' is now 'exit shell').
2. There are now two different drives, C and D, on both the Flight and Science processors. The C drive contains all configuration-related files; config, bin, missions, mafiles, and state directories. The D drive contains all data-related files; logs and sentlogs directories. The current directory includes the drive, so the commands "cd d:/logs" or "cd c:/config" will move between those directories as expected. The command 'dir' will list the current directory.

The table below shows the differences between common commands and features.

Persistor	STM32
exit pico	exit shell
boot pico	boot shell
app	app
boot app	boot app (Note: Will not actually boot the app, just set the vehicle to boot app)
consci	consci
exit	exit
send/s/zs *.*	send/s/zs *.* (Note: * will also work now)
burnapp	flash-flight or flash-science
u4stalk	uart (Note: uart now uses the actual comms port ID from the science bay motherboard, i.e. J0, J1, J2)
talk	talk (Note: Some device names have changed, 'talk ?' will list all available device names)
date	Date (time command not supported)
adtest	adtest (Note: 'adtest all' shows all channels as before. 'adtest <channel>' will show a single channel. 'adtest ?' will list all channels.)

Known Issues

Insitu Compass Cal

This Persistor feature is not presently supported. This feature is scheduled to be implemented. Check future releases for “Insitu Compass Cal” support.

Legacy Scripts

G3 SM32 processors are not supported by all legacy scripts. Existing scripts that support the new processor can be found in the forum at the following post:

<https://datahost.webbresearch.com/viewtopic.php?f=11&t=427>

Low Power Mode

The maximum value for low power cycle time is presently 15 seconds. This feature is scheduled to be implemented. Check future releases for “Low Power Mode fix”.

```
sensor: u_low_power_cycle_time(sec) -1.0      # 15 second max for G3 SM32 processor
                                              # in, num of secs/cycle on glider processor
                                              # during low power mode (dive/climbs)
                                              # <=0 disables low power mode
```

Science Sensor Limitations

The new hardware presently supports a maximum of 4 sensors.

Do not connect a MicroRider to J1. Use ports J0, J2, or J3 for the MicroRider.

The current default configuration reserves J1 for an optode.

Unusual Output – Unexpected GliderDos Prompt

Occasionally, the glider will print the GliderDos prompt while a mission is running. A user might confuse the prompt with the vehicle being out of the mission. If in doubt, typing ‘dir’ will reveal if the glider is in a mission or not.

File Transfer

The full path may be necessary when sending files. For example ‘zs [c:/mafiles/surfac*](#)’ to send surface files.

Exit Core Dump

Each time the system shuts down or exit resets, a “Core Dump” is printed to the screen. If you believe that the processor is not performing as expected, this log may be of assistance to the developers. In addition, the ‘core’ command will display this log at any time and “core clear” will delete the contents.

Time Handling

Persistor users are familiar with a slowly drifting clock and the option to “snyc_time.” This is no longer required. System time is now updated to GPS time at each surfacing.

Log Files

Log files from the new system have a slightly different format. dbd2asc version 2.5 or newer is required to read the log data. All released versions of Slocum Fleet Mission Controller (SFMC) have the newer version. Older dock-servers or customer-integrated shoreside systems may require an upgrade.

New Feature Procedures

Addressing “Smart Batteries” (Rechargeable Smart Batteries)

- From SFMC dock-server, communicate over serial perspective
- From ‘GliderShell’
 - Enable battery communication, “gpio gpio_111 on”
 - Run the talk program “talk gliderbus”
 - Query the smart data for each pack “\$R,0”

Syntax Examples

```
!type mafiles/yol4.ma
!zs config/sbdlist.dat
!szs config/*.dat
```

Moving from one drive to another drive.

Note: The glider and users do not need to be aware of which drive they are in. All functionality, (sending, receiving, testing, and missions) can be run from any directory.

```
GliderLAB N -1 >dir
Directory of c:/

config/
mafiles/
missions/
state/
bin/
sensors/
6 files

GliderLAB N -1 >cd d:
d:/

GliderLAB N -1 >dir
Directory of d:/

logs/
sentlogs/
2 files

GliderLAB N -1 >
```

Help Menu

GliderDos N -1 >help

?	ballast	ballvalve	boot
callback	cat	CD	chdir
CLRDEVERRS	compass_cal	consci	COPY
core	CP	date	DEL
DELLOG	DEVICES?	DF	digifin
DIR	drift_table	echo	EXECUTE
exit	format	freport	GET
HARDWARE?	HEAP	HELP	hs
LAB_MODE	LIST	loadmission	logging
LONGTERM	LONGTERM_PUT	lpstop	LS
MBD	MKDIR	MV	PRUNEDISK
PURGELOGS	PUT	RENAME	REPORT
RM	RMDIR	run	SBD
SEND	sequence	SETDEVLIMIT	SETNUMWARN
SIMUL?	SRF_DISPLAY	strobe	sync_time
SZR	SZS	tvalve	TYPE
USE	VER	WHERE	whoru
WHY?	wiggle	ZERO_OCEAN_PRESSUREZR	
ZS			

help list all commands in alphabetical order
 help -full list all commands with their help msgs
 help <cmd> .. <cmd> prints the help msgs for listed commands
 command names in lower case are NOT executable in mission via !

Burning Software

The following procedure is to be used if users intend to update a glider's firmware and plan to only update the apps without opening the glider.

1. Boot glider. If in boot shell, proceed to step 2.
 - a. If in boot app, get control of glider by typing "ctrl-c" and "callback 30".
 - b. Exit to shell by typing "exit shell".
 - c. Once glider has restarted in GliderShell, type "boot shell". This will force the glider to boot into shell instead of the glider application.
2. Switch the console to the science processor by typing "consci".
3. If the science processor is set to boot shell, proceed to step 4.
 - a. When science begins booting the app, at some point it will say "science booting in standalone mode, do you want to continue booting". At that prompt, enter "n".
 - b. Several seconds later, it will then ask you if you want to restart or boot to shell, enter "s" for boot to shell here.
4. At first SciShell prompt, put science processor into boot shell by typing "boot shell". This will prevent the science processor from automatically booting the app.
5. From the TWR code release, drag and drop the flash-science.gex file from the target-science directory to SFMC's to-glider folder.
6. Type "dockzr *" to send this to the science processor. This file transfer should take a few minutes.
7. Once transferred, type "flash-science" to flash (update) the science application. The user will then be asked if they want to proceed with flashing the application, at that prompt type "y". Flashing the application should take less than a minute, approximately.
8. When the application has been flashed, type "app". The app will boot just as in Step 3, where it will soon say "science booting in standalone mode, do you want to continue booting". At this prompt, type "n", then type "s" to boot to shell.

9. If this is successful, type “boot app”. This will cause the science application to boot when science is powered on.
10. Unplug the Freewave for approximately 10 seconds and plug it back in. The glider should then return to GliderShell.
11. From the TWR code release, drag and drop the flash-flight.gex file from the target-flight directory to SFMC’s to-glider folder.
12. Type “dockzr *” to send this to the flight processor. This file transfer should take a few minutes.
13. Once transferred, type “flash-flight” to flash (update) the flight application. The user will then be asked if they want to proceed with flashing the application, at that prompt type “y”. Flashing the application should take less than a minute, approximately.
14. When the application has been flashed, type “app -lab”. The glider should boot into GliderLab. At the first GliderLab prompt, type “callback 30”, followed by “exit shell”.
15. One back at a GliderShell prompt, type “boot app”. Note that this command will *not* actually boot the app as it used to do with persistors. The glider will now boot the app on start up.

High-Speed Communications (hs)

- High Speed (hs) allows for communications to the glider through a direct serial connection at a higher baud rate. The standard baud rate is 115200; the high-speed baud rate is 921600.

	hs off	hs on
Baud Rate	115200	921600
Connection	Direct Serial Freewave	Direct Serial

- Note: not all laptop serial ports and USB-to-serial adapters are capable of the high-speed rate.
- Note, high speed is not supported by the dockserver.
- The high-speed rate (921600) only works over the direct (hard-wired) serial connection.
- Important, turn off high speed (921600) before switching to Freewave (115200) communications, otherwise communications will be lost.
- The communications baud rate will automatically return to 115200 when the glider is power cycled, or when the CPU's for flight and science are reset, i.e. (reset button), 'exit reset', 'reboot', etc.
- High speed is currently available in GliderShell, SciShell, and GliderDos/LAB.
- Help from the command line (GliderShell, SciShell, and GliderDos/LAB):

```
SciShell> hs
```

```
Usage: hs on|off      - Go to High Speed RS232 Mode
```

- Recommended for uploading new Flight and Science executables in GliderShell and SciShell.

	Operator Terminal	Vehicle Shell
--	-------------------	---------------

[illegible]

	- Change baud rate of terminal program from 921600 to 115200	
--	--	--

- High speed mode works in GliderDos and not in SciDos. It is possible to turn on high speed in GliderDos and use the 'send' command. To receive files from science, the terminal programs baud rate would need to be switched to 115200. Once science is done sending, then the terminal programs baud rate would need to be switched back to 921600.
- The 'zs' command can be used to transfer flight and science log files from GliderShell and SciShell in high speed mode. One would have to use the 'mv' command to move them to d:/sentlogs when done.

	Operator Terminal	Vehicle Shell
flight	<p>- Switch high speed on</p> <p>- Change baud rate of terminal program from 115200 to 921600</p> <p>- Send log files</p> <p>- Move log files to sentlogs</p> <p>- Switch high speed off</p> <p>- Change baud rate of terminal program from 921600 to 115200</p>	<p>GliderShell></p> <p>??w?erShell> hs on</p> <p>GliderShell></p> <p>GliderShell> zs d:/logs/0055*</p> <p>About to send 4 files</p> <p>Prechecking 4 files: hit control-C to abort</p> <p>total_size of prechecked files = 15472 bytes</p> <p>START</p> <p>**B00</p> <p>Done!</p> <p>GliderShell> mv d:/logs/0055* d:/sentlogs</p> <p>moved 4 files</p> <p>GliderShell></p> <p>GliderShell> hs off</p> <p>????????????????????????????????????</p> <p>GliderShell></p>
science	- consci to science	<p>GliderShell> consci</p> <p>Switching to science</p>

Edits

Initials	Date	Change
JG	1/8/2020	Initial
JG	3/23/2020	Added procedure for updating application
RC	3/23/2020	Added High Speed communications documentation.
BA	3/25/2020	Reorganized for an initial guide
DP	3/27/2020	Added log files section, reformatted
MJB	3/27/2020	Formatting and readability
BA	3/31/2020	Draft A - minor edits for release into the wild