SE4120L-EK3 Evaluation Board User Guide

REVISION HISTORY

Revision	Description of Changes	Revision Date (mmm-dd-yyyy)
1.0	First Issue	May-23-2008

Table of Contents

1 Introduction

This document covers the specification, description and operation of the SE4120L-EK3 Evaluation Board, and applies to boards with PCB No. Z144-A.

The SE4120L-EK3 Evaluation Board is a GPS radio with high speed USB 2.0 USB interface which can be connected to a host PC.

Open-source driver software is provided which runs under Windows in a Linux shell. This allows the SE4120L to be programmed into all of its 21 real IF and baseband I/Q data output modes. There is an option to transfer 1 Mbyte data captures to the host PC, and process them off-line with Open-source Matlab scripts. Currently only acquisition only is supported, but the intention is support larger data captures to allow Matlab tracking and position fix in the future.

It is possible to run third party real-time Software GPS solutions on the SE4120L-EK3 evaluation board.

For product development, the limitations of the hardware implementation are discussed. Development of a real-time Software GPS solution is a highly complex exercise and there are issues with data loss across the USB interface. Trade-offs between data rate, implementation loss, host processor load, radio noise figure and antenna selection need to be considered.

2 Getting Started

2.1 Evaluation Kit Contents

The SE4120L-EK3 Evaluation Kit should contain the following items:

- SE4120L-EK3 USB dongle
- Software CD containing user guide, Open-source driver software, Open-source Matlab, example captures for all 16 valid modes with GPS and Galileo IF filter
- External active antenna (u-Blox ANN-MS-2-005-0)
- Mini-B USB cable

2.2 Operation

- On a PC with High Speed USB 2.0 interface download and install Cygwin (see Section 6.1.1)
- Copy the Open-source driver folder on the Software CD to the C: drive so that the contents are stored in `C:\SiGe_SE4120L_CU_USBdriver\`
- Connect the SE4120L-EK3 USB dongle to the PC using the USB cable supplied
- Install the driver for the SE4120L-EK3 which can be found in: `C:\ SiGe_SE4120L_CU_USBdriver\install` (see Section 6.2.1)
- In Cygwin, program the SE4120L-EK3 and run `ogusb-lite.exe` to get to the driver menus (see Section 6.2.2)
- Connect the external active antenna for a live GPS feed (the antenna needs a clear view of the sky) or connect to a GPS Simulator set to +10 dB relative to the nominal -130 dBm output.
- Copy the Matlab folder from the Software CD and store in a convenient directory on the C: drive.
- A version of Matlab with the Signal Processing Tool Box option is required for full graphical output
- The example data captures on the Software CD will demonstrate Matlab operation if a GPS feed is not available.

3 Hardware Specification

The table below gives the hardware specification for the SE4120L-EK3.

Parameter	Specification
Overall radio noise figure	2.5 dB (typical, includes 0.3 dB board and connector losses)
Input return loss	6 dB (typical)
Data output mode	21 programmable modes, 16 of which can be used for data capture on the host PC. Real IF or baseband I/Q Parallel or serial output Continuous or serial ‘bursted’
IF frequency selection	Real IF (4.092 MHz) or baseband I/Q (near-zero-IF)
IF filter BW selection	GPS or Galileo (2.2 MHz or 4.4 MHz)
Sample clock selection	16.368 MHz, 8.184 MHz, 5.456 MHz, 4.092 MHz
TCXO reference frequency	16.368 MHz Tolerance: ±1.5 ppm (Set tolerance and 2 reflows) Temperature (-40 to +85°C): ±0.5 ppm
Input supply voltage:	USB 5.0 V (3.6 V minimum for standalone tests)
Current consumption	37 mA (typical, excluding antenna bias)
Active antenna bias:	
Supply voltage:	3.3 V (or 5 V[1])
Short circuit current limit (3.3 V):	60 mA (typical), 300 mA max for 10 ohm load
Short circuit current limit (5.0 V):	Determined by USB supply characteristic, *2 A maximum*
RF input connector	MCX (jack)
PC interface connector	USB Mini-B
Host PC interface	USB 2.0 high speed (480 Mbps)

1 Requires 0 ohm link change

4 Circuit Description

The schematic is shown in Appendix A: Schematic. The circuit consists of a SE4120L [1] based GPS radio and a Cypress CY7C68013A high speed USB 2.0 controller interface [4].

4.1 SE4120L Radio

4.1.1 LNA

The LNA input matching has been optimized for minimum noise figure. In this implementation the low noise figure has been achieved with matching components L4 = 10nH Murata wire-wound inductor and C7 = 1p0. Normally C7 is set to 1.5pF for optimum noise match, but the value has been reduced to compensate for higher than usual PCB parasitics for this design. The input can be connected directly to a GPS simulator or signal generator for conducted test measurements. Note that a DC block may be required if the test equipment has a DC path to ground.

The LNA power supply is necessarily decoupled with a shunt 8.2pF capacitor (C5) to ‘tune out’ the parasitic inductance to the LNA and peak the gain (18.5 dB) at 1575.42 MHz. The purpose of choke L3 is to isolate the tuning effect of C5 from the general decoupling capacitor C4. Unless the decoupling effect of C4 is isolated, the gain response curve may exhibit a secondary peak, or at worst reduce the LNA gain by 2 dB, depending on the value for C4.

In some design cases where an external LNA is additionally used, the reduction in gain is sometimes beneficial to ensure that the overall front-end gain ahead of the mixer is not excessively high. This tends to have a ‘degeneration’ effect, where the noise figure is largely unchanged, but the input return loss is improved slightly. The downside is that the LNA is operating on a gain slope. To achieve this gain reduction C5 should be increased to 100pF.

4.1.2 External Active Antenna

An external active antenna can be connected to J1 for reception from live satellites or a radiated GPS simulator feed. A 3.3 V supply for the active antenna is present on the MCX connector J1. Most active antennas will operate with this supply voltage. Inductor L1 is a self-resonant bias choke to the 3.3 V supply, and C2 is a DC block to the LNA input bias on U1 pin 3. It is possible to change the antenna bias to 5 V by moving R2=0R to position R1, but beware that the short circuit current from the USB supply could be as high as 2 A.

It is recommended that an active antenna with low LNA gain (i.e. < 20 dB net gain) is used to avoid excessive front-end gain, e.g. Wi-Sys WS3914. Note that this antenna has an SMA connector fitted as standard. Other recommended antennas are u-Blox ANN-MS-2-005-0 (supplied with the kit) and Wi-Sys WS3910, but these have higher gains.

4.1.3 SAW Filter

There is a low loss Epcos B9000 0.5 dB SAW filter between the LNA output and mixer input, providing 15 to 20 dB of out of band rejection. This improves overall out of band blocking performance of the SE4120L so that it is limited by the SE4120L LNA. Use of higher attenuation SAWs (e.g. Epcos B7839 and B7829) will not necessarily improve the blocking performance further.

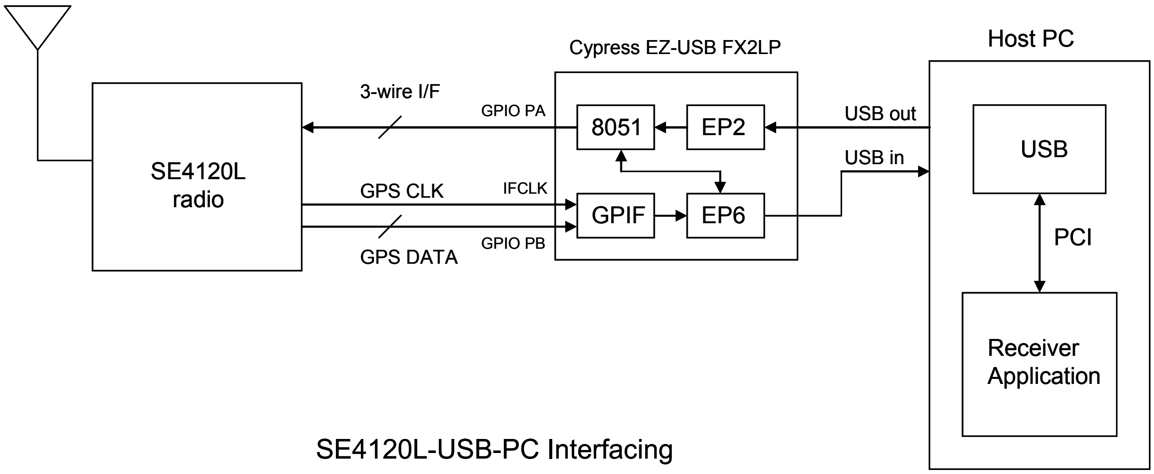
This inter-stage filter has minimal impact on the overall noise figure of the SE4120L radio. In designs using an additional external LNA stage or internal active antenna, this filter should be moved so that it is positioned at the external LNA output to give maximum blocker rejection with still minimal impact on overall noise figure.

4.2 USB Controller

The Cypress USB controller CY7C68013A is an 8x8mm, 56-QFN device [4]. It is an EZ-USB FX2LP USB Microcontroller device containing an 8051, 16 Kbytes of RAM and 4 Kbytes FIFO buffer.

4.2.1 Interfacing

The figure below shows how the USB controller interfaces with the host PC and SE4120L radio.



SE4120L-USB-PC Interfacing

The PC driver installs the target Open-source FX-2 driver in the USB controller volatile memory on connection of the USB interface via the Endpoint FIFO EP2. The EEPROM (U52) only stores the Open-source VID and PID which override the default Cypress VID and PID of the USB controller. The EEPROM interfaces to the USB controller via an I2C interface.

The SE4120L is programmed via PC control and the USB controller. There is a switch to transfer GPS data files from the SE4120L via the General Programmable Interface (GPIF) and Endpoint FIFO EP6 in the USB controller to the host PC.

The SE4120L CLK drives the IFCLK (Interface Clock) pin input to the FX-2, and IF samples are clocked into the FX-2 synchronously.

4.2.2 GPIO Interfaces

The GPS data output from the SE4120L is routed to GPIO port PB which is configured as an input port. The 3-wire software programming output for the SE4120L comes from GPIO port PA which is configured as an output port.

The SE4120L data (and also sync) is output on between one and four pins from SE4120L pins 10 to 13 to port PB. For 4-bit parallel data output modes, Fs_SEL0 / 3W_CLK (pin 12) and Fs_SEL1 / 3W_DATA (pin 13) of the SE4120L serve as SE4120L software programming inputs when 3W_ENB / FILT_BW (pin 14) is low, and as SE4120L data outputs when this pin is high. For other data output modes these two pins are undefined.

3W_ENB / FILT_BW (pin 14) is controlled by PA0. There is also a precautionary 47K pull-up resistor (R8) which by default puts the SE4120L in data output mode, but this is strictly not necessary as this pin is pulled high by PA0 when the SE4120L is not being programmed.

The GPIO pins PA1 (3W_DATA) and PA2 (3W_CLK) for software programming the SE4120L are tri-stated after the SE4120L is programmed in order not to conflict with 4-bit parallel output data from the SE4120L in data output mode. 4.7K resistors between output ports PA1 and PA2 for software programming and input ports PB3 and PB2 for GPS data ensure there is not a direct connection between the two pairs of GPIO interface pins, and the output pins of the SE4120L.

The GPS data inputs on the eight pins of the GPIO port PB interface are used to determine an 8-bit data word, where PB0 is the MSB. Unused input pins PB5 to PB7 are grounded. Pin PB4 is necessarily pulled high in data output mode since it is connected to 3W_ENB. The data value is determined by the state of the inputs PB0 to PB3. 47K pull-down resistors on the GPS data inputs PB2 and PB3 ensure that the these GPIO inputs are pulled low when the SE4120L is not in 4-bit parallel data output mode, since the SE4120L outputs on Fs_SEL0 / 3W_CLK (pin 12) and Fs_SEL1 / 3W_DATA are not defined. Removing these two resistors will result in incorrect data values for 2-bit modes. The exception to the pull-down requirement is input PB1, which is an undefined SE4120L output for 1-bit real IF modes, and is otherwise un-terminated.

One thing to be aware of with the configuration of 4.7K series and the 47K pull-up / pull-down resistors on 3W_CLK (pin 12), 3W_DATA (pin 13) and 3W_ENB (pin 14) is that there is a resistive pot-down on the programming voltage to the SE4120L such that the swing to one of the rails will only be 9/10th. The maximum voltage on 3W_CLK and 3W_DATA will be 3.0 V and the minimum voltage on 3W_ENB will be 0.3 V. If the 47K resistors were moved to the other side of the 4.7K series resistors there would be similar resistive pot-down of the SE4120L data outputs on 3W_CLK and 3W_DATA pins when the SE4120L is in data output mode.

4.2.3 Software Programming

GPIO port PA controls the SE4120L 3-wire bus and HW pin. PA3 pulls the SE4120L HW pin 20 low to put the SE4120L into software mode. It is also possible to pull the SE4120L HW pin low with R13=0R, but R16 must be removed to avoid shorting this output pin to ground, in other words move R16=0R to position R13.

The hardware also allows programming of the SE4120L with the GPIO port PB interface. This allows support for legacy software. Note that it is generally undesirable to share the same port for GPS data input and programming data output due to the conflicts with inputs and outputs.

Only modes with a continuous clock allow data to be transferred via the USB interface for data capture. ‘Bursted’ byte sync modes can be configured but data transfer via USB is not possible. For further information on the ‘bursted’ byte sync modes, and also pulse sync modes, refer to the SE4120L Datasheet [1].

For software mode programming two mission mode words are sent. Word D0 = 0x18 is a configuration word which essentially sets the SE4120L output drive to maximum to ensure that the slew rate limited clock can reliably drive the USB IFCLK at the maximum 16.368 MHz clock frequency. Word D1 sets the data output mode, see table below. The driver sends MSByte D1 first, then LSByte D0 next, with MSB bit first. Further details can be found in the SE4120L Programming Guide [3]. The words are sent twice for reliably reasons associated with the USB hardware.

SE4120L programming and physical data outputs

Data Output format					D1 Word Program value	Physical Data Outputs			
Mode	Sync	Sample rate Msps	Data Format	CLK output MHz		DATA_OUT (pin 10)	SYNC (pin 11)	Fs_SEL0 (pin 12)	Fs_SEL1 (pin 13)
1	Serial pulse	8.184	2-bit I/Q	16.368	a4	Si, Sq, Si, Sq, ...	Pulse sync	-	-
2	Serial pulse	4.092	2-bit I/Q	8.184	ac	Si, Sq, Si, Sq, ...	Pulse sync	-	-
3	Serial pulse	4.092	4-bit I/Q	16.368	ec	Si, Mi, Sq, Mq, Si, Mi, Sq, Mq, ...	Pulse sync	-	-
4	Serial byte	5.456	1-bit real	8.184	38	S, S, S, S...	Byte sync	-	-
5	Serial byte	5.456	2-bit real	16.368	78	S, M, S, M...	Byte sync	-	-
6	Serial byte	5.456	2-bit I/Q	16.368	b8	Si, Sq, Si, Sq...	Byte sync	-	-
7	Serial byte	4.092	2-bit I/Q	16.368	bc	Si, Sq, Si, Sq...	Byte sync	-	-
8	Parallel	5.456	2-bit real	5.456	58	Sign	Mag	-	-
9	Parallel	16.368	1-bit real	16.368	10	Sign	-	-	-
10	Parallel	16.368	2-bit real	16.368	50	Sign	Mag	-	-
11	Unsampled	16.368	1-bit real	16.368	00, 04, 08, 0c	Unsampled Sign	-	-	-
12	Parallel	5.456	1-bit real	5.456	18	Sign	-	-	-
13	Bypassed	16.368	2-bit real	16.368	40, 44, 48, 4c, 80, 84, 88, 8c, c0, c4, c8, cc	Sign	Mag	-	-
14	Parallel	16.368	2-bit I/Q	16.368	90	Sign I	Sign Q	-	-
15	Parallel	8.184	2-bit I/Q	8.184	94	Sign I	Sign Q	-	-
16	Parallel	5.456	2-bit I/Q	5.456	98	Sign I	Sign Q	-	-
17	Parallel	4.092	2-bit I/Q	4.092	9c	Sign I	Sign Q	-	-
18	Parallel	16.368	4-bit I/Q	16.368	d0	Sign I	Sign Q	Mag I	Mag Q
19	Parallel	8.184	4-bit I/Q	8.184	d4	Sign I	Sign Q	Mag I	Mag Q
20	Parallel	5.456	4-bit I/Q	5.456	d8	Sign I	Sign Q	Mag I	Mag Q
21	Parallel	4.092	4-bit I/Q	4.092	dc	Sign I	Sign Q	Mag I	Mag Q

4.2.4 24 MHz Crystal

The 24 MHz crystal tank circuit follows the Cypress recommended circuit, namely 12pF shunt capacitors either side of the crystal. The additional capacitor C60 offers some fine frequency adjustment.

Resistor R53=470R is a precaution to limit the maximum drive from the oscillator circuit to the crystal. From the Cypress datasheet [4], the oscillator appears to deliver up to 500 uW. Typically, cheap metal can SMD49U crystals can withstand this power, but not the smaller surface mount crystal variety (typically 50 – 200 uW max). Excessive power can damage the crystal and permanently change the resonant frequency by an amount that exceeds the pulling range.

According to the Cypress datasheet, the total shunt capacitance to ground either side of the crystal is 12pF plus 3pF from the PCB. The effective capacitance across the crystal will be 15pF + 15pF = 7.5pF. Assuming an additional 2pF max each side due to the shunt loading of the Cypress pins from pad capacitance and internal circuit capacitance, the total load capacitance will be 8.5pF at most. The NDK 24 MHz crystal specified in the BOM has an 8pF specified load capacitance. Set tolerance is ±15 ppm and temperature stability is ±15 ppm (-10 ºC to +70 ºC).

The clock tolerance specification for high speed USB 2.0 is ±500 ppm.

4.3 Hardware Issues

There are a few hardware issues related to this design which the user should be aware of. Resolving these satisfactorily for a final product design may require additional circuitry. Some of the design features relevant to a final product have already been covered in previous sections.

4.3.1 Maximum Front-End Gain

The evaluation board is designed to demonstrate the optimized noise figure performance of the SE4120L at the RF connector. Adding an external active antenna is a convenience because it avoids the matching issues and cable losses associated with a purely passive antenna. Also active antennas are readily available. The user should be aware that the addition of an active antenna may result in excessive front-end gain which will tend to operate the SE4120L in 1-bit ‘limiter’ mode. This is undesirable for multi-bit data output modes where the low noise figure performance afforded by the active antenna is cancelled out by the increased implementation loss associated with single bit operation.

There is a datasheet specification [1] for the maximum front-end gain ahead of the mixer. The maximum signal load at MIX_IN (pin 21) for normal AGC operation (under typical conditions) is -137 dBm/Hz. Assuming the usual thermal noise limited situation, where the GPS input signal is below the noise floor, then the maximum excess noise permitted in front of the mixer is -137 +174 = 37 dB. Assuming 1 dB noise figure for the external LNA gives 36 dB maximum front-end gain. A more conservative target for worst-case conditions is 32 dB excess noise, i.e. 31 dB maximum front-end gain.

The SE4120L LNA has typically 18.5 dB gain, but this can be reduced to 16.5 dB with changes to the LNA decoupling previously described. With the reduced gain LNA, the additional gain that can be added is 19.5 dB typically and 15.5 dB for worst-case conditions.

Typically an external active antenna will have a net gain of 23 dB, taking into account 1.3 dB/m loss of the RG174 cable and 1-2 dB loss for the band-pass filter. However, some antennas (e.g. Wi-Sys WS3914) have a net gain of only 13 dB. External LNAs used in on-board designs tend to have gains of around 18 to 20 dB.

Normally, for designs for active antennas with > 20 dB net gain it is recommended that these are routed directly to the SE4120L mixer input (pin 21). However, this is not ideal for the low-gain active antenna case. A solution to satisfy the gain requirements for low- / high-gain active antennas and external on-board LNAs is to place a simple three resistor attenuator between the SE4120L LNA output and mixer input giving typically 5 dB attenuation. This will have little impact on the overall noise figure with the additional front-end gain stage. Assuming 16.5 dB gain for the SE4120L LNA and 5 dB attenuation, this will typically give overall front-end gains of 24.5 dB, 31.5 dB and 34.5 dB respectively for the three cases. (Note that the mixer input has DC bias on it, and a 22pF DC block should be fitted for isolation from any attenuator).

4.3.2 Antenna Short Circuit Protection

The 3.3 V radio regulator U102 has a short circuit fold-back current limit which restricts the maximum continuous antenna current to about 60 mA in the event of an inadvertent short. The Torex regulator has a guaranteed output current of 150 mA and but can source up to 300 mA. Instantaneous in-rush currents can cause damage to 39nH 0402 multi-layer inductors rated at 200 mA. For this reason a 39nH 0603 multi-layer inductor is fitted which is rated at 500 mA.

The antenna bias supply can be increased to 5 V by moving R2 to position R1. Note that the short circuit current of the 5 V raw USB supply can be as high as 2 A. There is a risk that the bias choke L1 or the 5 V supply filter inductor L101 could be fused open circuit by the excessive current. In some cases choke L1 may go high resistance and appear to supply close to 5 V until an antenna load is connected.

Beware also that a short circuit on the 3.3 V radio supply will cause the SE4120L to loose its software programmed state. A short circuit on the +5 V supply will cause the USB controller to loose its firmware.

4.3.3 USB Controller Reset Issues

Disconnecting and connecting the antenna feed can sometimes cause issues with the simple reset circuit for the USB controller. A watch-dog IC would make this more reliable but would add cost.

Sometimes the board can power up in the wrong state. This is indicated by a current consumption of 19 mA rather than 37 mA. This may be related to the reset issue.

4.3.4 Operating Temperature Range

The radio circuit including SE4120L and TCXO are specified to operate over -40 ºC to +85 ºC.

The Cypress USB controller part fitted is a CY7C68013A-56LFXC which is specified to operate over the commercial temperature range 0 to + 70 ºC. The –I suffix part should be used for a wider -40 ºC to +105 ºC industrial temperature range. The 24 MHz crystal should also be specified accordingly.

4.4 Software issues

4.4.1 Byte Sync Modes

With byte-sync data, the GPS CLK is not continuous. It has wait periods between bytes, which effectively causes the driver to stall and buffer overruns to occur.

The problem with adapting the Open-source FX-2 driver to transfer byte sync data is that it inherently uses synchronous data capture of samples using the FX-2 set to 'Master GPIF' in synchronous mode

One possible solution would be to change the driver to function in 'Slave Mode, Asynchronous', but this is quite a major change to an already complex driver. In this mode, data would be clocked using the SE4120L CLK as a data-strobe rather than a true clock. It would also require a physical link from CLK_OUT to SLRD. Also, in 'Master GPIF' mode (sync), the internal timing / control are defined using a separate 'GPIF-designer' GUI, which then generates pseudo-code for incorporation into the driver.

4.4.2 Buffer Overruns

Buffer overruns are best demonstrated by attempting a data capture in byte sync mode (see Section 6.2.3.1). The wait periods between bytes cause the driver to stall and buffer overruns to occur. For valid data capture modes, as the data capture length is increased the chances of buffer overruns increases. This is a challenge for developing a real-time, continuous driver.

5 PCB Layout

The PCB is 4-layer FR4, 1 mm thick, with tracking on layer 1 (component side) and layer 3, and ground fill on layer 2 and layer 4 (underside). The construction consists of a central double-sided core with pre-preg sections to the outer copper layers. The PCB lay-up is shown below. See Appendices for component placement and tracking layer plots.

1mm PCB build

PCB is 4 layer build

Total = 0.983 mm (excluding solder resist and silkscreen layers)

Overall thickness approx 1.0mm

LAYER 1 and LAYER 4 are 35um after plating

THE ORDER OF THE LAYERS IS VITAL

Electroless gold over Nickel

PCB dielectric is standard FR4, typical permittivity (Er) is 4.2 to 4.3

Note:
um = micron
Cu = copper
D/S = Double-sided

50 ohm track widths				
	Track layer	**GND layer(s)**	**mm**	**thou**
Microstrip	1	2	0.533	21
Stripline	3	2,4	0.203	8

6 Open-source Driver Software

6.1 Software Sources

6.1.1 Cygwin

A Cygwin shell is required for execution under Windows. Cygwin is a Linux-like environment for Windows and is required if the host source code is compiled for Linux OS. The USB driver libraries for different target OS are also included with the source code.

The software can be downloaded from: http://www.cygwin.com via a choice of mirror sites. Accept the default installation option to minimize data download and installation time. This operation will take several minutes.

Some of the batch file commands below can be run in standard Windows DOS shell, but since the Open-source software needs to be operated under Linux, it is easier to be consistent and use a Cygwin shell from the outset.

6.1.2 Open-source Software

The Open-source software is included in the Software CD. The folder contains the Open-source software and windows drivers required to operate the SE4120L-EK3:

`SiGe_SE4120L_CU_USBdriver`

This should be copied to the following directory:

`C:\SiGe_SE4120L_CU_USBdriver\`

6.2 Software Configuration of the SE4120L-EK3

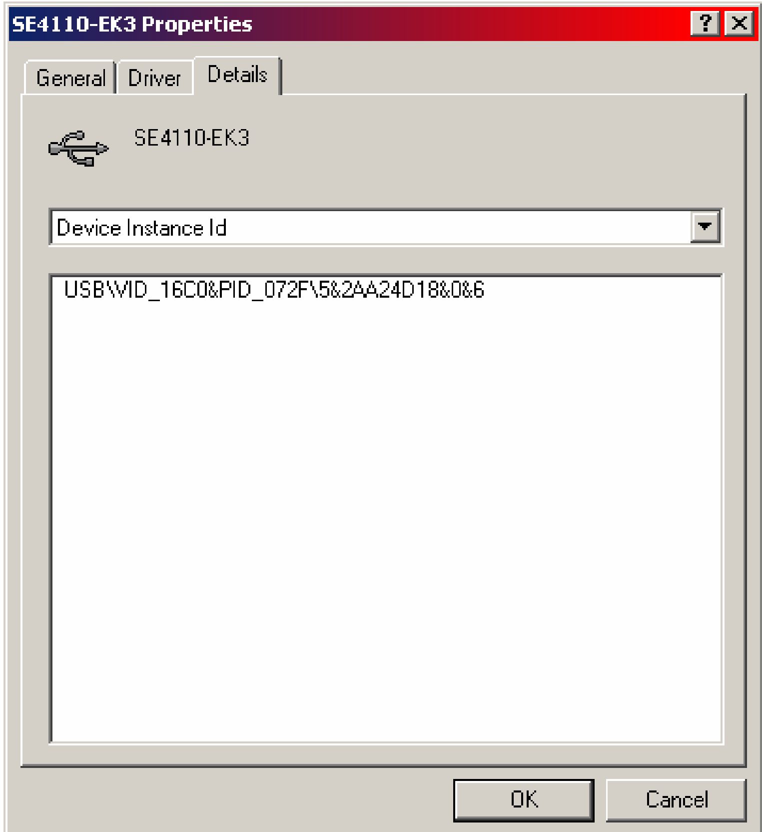
6.2.1 Windows Driver Installation

Plug the SE4120L-EK3 into a PC USB 2.0 interface with the appropriate USB lead. If this is the first time a SE4120L-EK3 has been plugged into the PC, windows will request a driver. Using the Windows 'Found New Hardware Wizard' navigate to where the driver is located:

`C:\ SiGe_SE4120L_CU_USBdriver\install`

Install, then unplug the board and re-insert.

Once the driver is successfully installed the Device Manager>SE4110 EK3 Properties>Details window should look like this:



6.2.1.1 Troubleshooting

There will be problems with installing the driver if the board has a VID and PID different than that shown above. For details on how to change the ID of an unprogrammed board refer to Section 6.3.

Problems may be encountered if the PC has already been used to run other Software GPS applications on similar Cypress USB controller hardware. It is recommended that these utilities are uninstalled or an alternative 'clean' PC is used.

To uninstall other Software GPS applications, ensure all existing cached drivers are removed using the following steps:

- Run the uninstall program for the Software GPS application.
- Ensure the hardware is removed from the USB port.
- Go to `C:\windows\system32\drivers` and remove any files with references to the Software GPS application. Typically these would be *.sys and *.hex files.
- Go to `C:\windows\inf` and remove any files with references to the Software GPS application, for example, oem*.inf files. The associated dates of the files will indicate whether they have been recently installed. To be absolutely certain unrelated files are not accidentally deleted, open the files in Notepad first to make sure.
- The manual delete process is necessary because normally Windows uninstall programs do not automatically remove these files.
- If the above process is unsuccessful, there may be additional references in the PC registry. In this case it may be easier to use an alternative 'clean' PC.

6.2.2 Opening a Cygwin DOS Box

Open up a Cygwin Shell DOS box (command prompt) by either selecting the Cygwin desktop shortcut or from:

`Start\Programs\cygwin\Cygwin Bash Shell`

In the Cygwin shell, type:

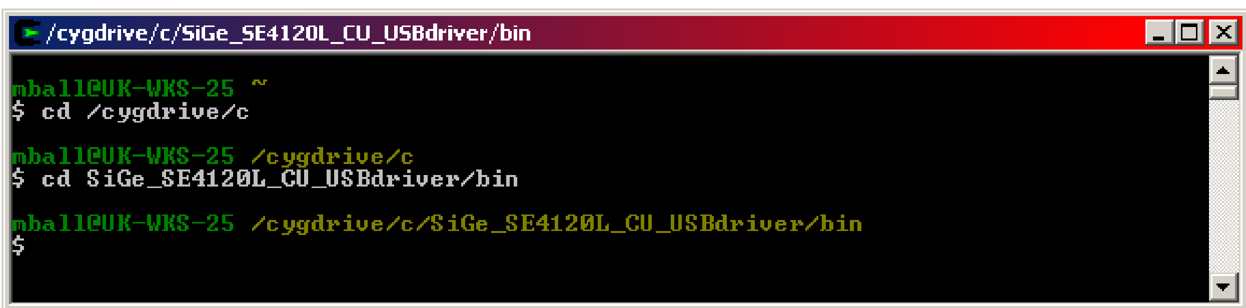
`cd /cygdrive/c`

and press return

Change directory to `C:/SiGe_SE4120L_CU_USBdriver/bin` by typing:

`cd SiGe_SE4120L_CU_USBdriver/bin`

and press Return



6.2.3 Running ogusb-lite.exe

In the Cygwin shell, type:

`./program.bat`

and press Return.

Note the importance of typing the ***"./"*** *prefix before all commands.*

The last part of Command Prompt dialog should show the following information:

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
Programming  14 bytes starting at 0x0c11, checksum ok
Programming   6 bytes starting at 0x0c1f, checksum ok
Programming   5 bytes starting at 0x0c25, checksum ok
Programming   5 bytes starting at 0x0c2a, checksum ok
Programming   3 bytes starting at 0x0c2f, checksum ok
Programming   9 bytes starting at 0x0c32, checksum ok
Programming   3 bytes starting at 0x0c44, checksum ok
Programming  10 bytes starting at 0x0c47, checksum ok
End of file
bus-0/\\.\libusb0-0001--0x16c0-0x072f 16C0/072F
Using device vendor id 0x16c0 product id 0x072f
Press any key to continue . . .

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

Note: `'program.bat'` needs to be run every time the board is powered up, as the memory in the Cypress USB controller is volatile. Note that only the VID and PID are stored, and this is held on a separate EEPROM device on the USB board.

Next type:

`./ogusb-lite.exe`

and press Return.

The window shows the menu of all data output modes available on the SE4120L. This is basically the help menu.

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
Press any key to continue . . .

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite.exe

SE4120 Program Option [file transfer]

 MENU  MSPS     BITS    DATA    SYNC
 ----  ----     ----    ----    ----
  1     8.184   2-bit   I/Q     pulse
  2     4.092   2-bit   I/Q     pulse
  3     4.092   4-bit   I/Q     pulse
  4     5.456   1-bit   real    byte
  5     5.456   2-bit   real    byte
  6     5.456   2-bit   I/Q     byte
  7     4.092   2-bit   I/Q     byte
  8     5.456   2-bit   real   parallel
  9     16.368  1-bit   real   parallel
 10     16.368  2-bit   real   parallel
 11     16.368  1-bit   real   unsampled **
 12     5.456   1-bit   real   parallel
 13     16.368  2-bit   real   bypass **
 14     16.368  2-bit   I/Q    parallel
 15     8.184   2-bit   I/Q    parallel
 16     5.456   2-bit   I/Q    parallel
 17     4.092   2-bit   I/Q    parallel
 18     16.368  4-bit   I/Q    parallel
 19     8.184   4-bit   I/Q    parallel
 20     5.456   4-bit   I/Q    parallel
 21     4.092   4-bit   I/Q    parallel

 For Galileo Filter add 100 to menu code
 For File transfer add option switch [1]

 e.g. 'ogusb-lite.exe 103 1' is:
 'Galileo, 4.092MSPS, 4-bit I/Q, pulse-sync + file transfer

  ** means Block Convertor bypass modes


mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

To select a GPS target mode just enter the same command but with an appended suffix number 1 to 21. For example: type

```
./ogusb-lite.exe 10
```

and press Return.

Note: *if you press the "up arrow" key repeatedly in the command box the last command(s) will be repeated and you can avoid unnecessary typing by editing a previous entry. This is useful if the board is powered down and requires programming again.*

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite.exe 10
Changing SE41120 Configuration
GPS, 16.368 Msps, 2b real, Parallel, 16.368MHz

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ _
```

To select a target mode and record a '`gnss.bin`' data file enter the previous command but with additional suffix "1".

For example, type:

```
./ogusb-lite.exe 10 1
```

and press Return

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite.exe 10 1
Changing SE41120 Configuration
GPS, 16.368 Msps, 2b real, Parallel, 16.368MHz
xfered 1 MB in 0.016 seconds.  62.5 MB/s.

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

Note: the file size transferred is currently fixed at 1 MByte. This can be changed by a source code modification. 1 MByte is adequate for simple analysis.

To select a target mode with Galileo filter set, enter the same command but add 100 to the value.

For example, for Mode 10 with Galileo filter and data capture, type**:**

`./ogusb-lite.exe 110 1`

and press Return.

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite.exe 110 1
Changing SE41120 Configuration
Galileo, 16.368 Msps, 2b real, Parallel, 16.368MHz
xfered 1 MB in 0.016 seconds.  62.5 MB/s.

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ _
```

Note: in each case, the time shown for data transfer is in 0.016 sec minimum steps and sometimes the displayed time is 0 seconds.

The captured data is stored in

`C:/SiGe_SE4120L_CU_USBdriver/bin/gnss.bin.`

With Open-source Matlab software it is possible to perform acquisition of satellites in the captured data. See Section 7 for further details.

Note that the data capture should not be performed with the un-sampled Mode 11 or the byte sync Modes 4 to 7. The byte sync modes have a non-continuous, 'bursted' clock which causes a problem with the USB controller during the clock idle periods.

6.2.3.1 Troubleshooting

Note that occasionally for data captures there may be a 'buffer overrun' error. If this occurs, try running the command again:

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite.exe 10 1
Changing SE4112O Configuration
GPS, 16.368 Msps, 2b real, Parallel, 16.368MHz
Buffer overrun...
xfered 0 MB in 0 seconds.  nan MB/s.

mball@UK-WKS-25 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ _
```

A 'buffer overrun' will also be displayed if attempts are made to capture data with one of the byte sync modes which have a non-continuous, 'bursted' clock, for example in 1-bit real IF byte sync mode at 5.456 MHz (Mode 4):

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin

mball@UK-WKS-04 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite 4 1
Changing SE4112O Configuration
GPS, 5.456 Msps, 1b real, Byte Sync, 8.184MHz
Buffer overrun...
xfered 0 MB in 0 seconds.  nan MB/s.

mball@UK-WKS-04 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

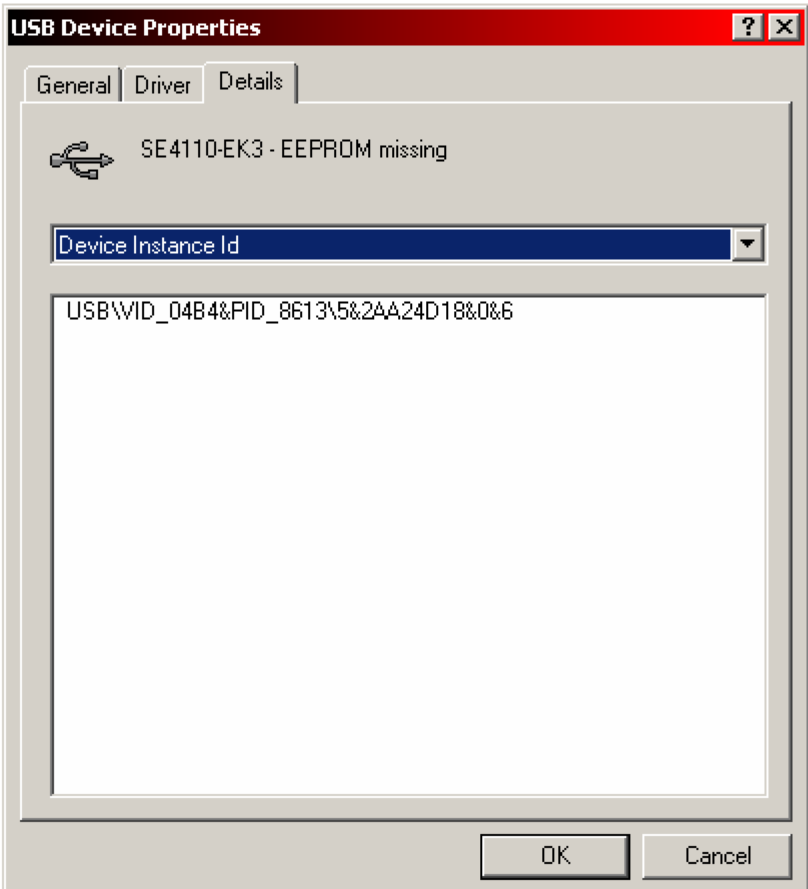
If the data capture file `'gnss.bin'` is open, for instance being viewed by a binary file viewer, the following error will be displayed:

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin

mball@UK-WKS-04 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./ogusb-lite 3 1
Changing SE4112O Configuration
GPS, 4.092 Msps, 4b I/Q, Pulse Sync, 16.368MHz
Could not open output file
fopen: Device or resource busy
xfered 1 MB in 0.016 seconds.  62.5 MB/s.
      6 [main] ogusb-lite 3248 _cygtls::handle_exceptions: Error while dumping s
tate (probably corrupted stack)
Segmentation fault (core dumped)

mball@UK-WKS-04 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ _
```

6.3 ID Configuration of an Un-programmed Board

The SE4120L-EK3 should be supplied with the Open-source ID, VID=16C0 and PID=072F. If it still has the default Cypress USB controller ID, VID=04B4 and PID=8613, the procedure described below should be used. It is possible to do this in a standard DOS shell. Note that if other GPS software is installed on the PC which is designed to work with the USB interface this may be difficult to implement.

In the Device Manager>SE4110 EK3 Properties>Details window, the ID of the board will look like this:



In a Cygwin shell in the C:`/SiGe_SE4120L_CU_USBdriver/bin` directory type:

`./loader.bat`

and press Return.

The last part of Command Prompt dialog should show the following information:

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
Programming  16 bytes starting at 0x09e7, checksum ok
Programming   7 bytes starting at 0x09f7, checksum ok
Programming   1 byte  starting at 0x09fe, checksum ok
Programming   3 bytes starting at 0x0000, checksum ok
Programming  12 bytes starting at 0x07f3, checksum ok
Programming  16 bytes starting at 0x0d03, checksum ok
Programming   1 byte  starting at 0x0d13, checksum ok
End of file
bus-0/\\.\libusb0-0001--0x04b4-0x8613 04B4/8613
Using device vendor id 0x04b4 product id 0x8613
Press any key to continue . . .

mball@UK-WKS-11 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

Next type:

./program_vendor.bat

and press Return.

The last part of Command Prompt dialog should show the Open-source VID and PID.

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
VID: 0x16c0, PID: 0x72f
Writing 8 bytes to eeprom 0x0
buf[0] = 0xc0
buf[1] = 0xc0
buf[2] = 0x16
buf[3] = 0x2f
buf[4] = 0x7
buf[5] = 0x1
buf[6] = 0x0
buf[7] = 0x1
Press any key to continue . . .

mball@UK-WKS-11 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ _
```

The ID of the board can be re-checked by typing:

./read_vendor.bat

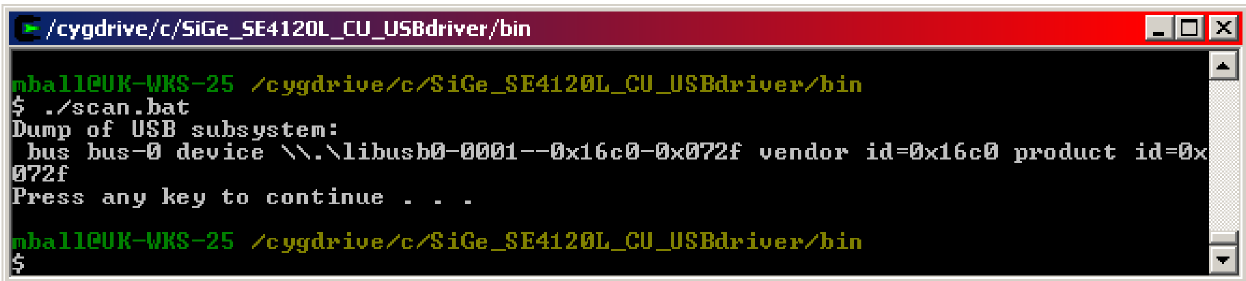
and press Return.

```
/cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
mball@UK-WKS-11 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$ ./read_vendor.bat
bus-0/\\.\libusb0-0001--0x04b4-0x8613 04B4/8613
Using device vendor id 0x04b4 product id 0x8613
Reading 8 bytes from eeprom 0x0
buf[0] = 0xc0
buf[1] = 0xc0
buf[2] = 0x16
buf[3] = 0x2f
buf[4] = 0x7
buf[5] = 0x1
buf[6] = 0x0
buf[7] = 0x1
Press any key to continue . . .

mball@UK-WKS-11 /cygdrive/c/SiGe_SE4120L_CU_USBdriver/bin
$
```

It is also possible to type:

./scan.bat

and press Return.



6.4 ID Configuration for Third Party Software

The SE4120L-EK3 is configured for Open-source VID=16C0 and PID=072F. For operation with non-Open-source third party software, either the `'*.inf'` file for the third party software driver needs to be manually edited using a text editor to accept the Open-source ID, or alternatively the ID can be reprogrammed by editing the VID and PID.

Reprogramming the ID is achieved by modifying the `'loader.bat'` and `'program_vendor.bat'` batch files, as follows. Open `'loader.bat'` with Notepad editor, replace the three instances of the default Cypress ID with the Open-source ID, and save under a new name:

```
@ECHO OFF
fx2-programmer 0x16c0 0x072f set 0xE600 1
fx2-programmer 0x16c0 0x072f program ../firmware/Vend_Ax.ihx
fx2-programmer 0x16c0 0x072f set 0xE600 0
PAUSE
```

Next, open `'program_vendor.bat'`, with Notepad editor, replace the first ID instance with the Open-source ID, and the second ID instance with the new, desired ID. The example below will change the Open-source ID back to the default Cypress ID:

```
@ECHO OFF
fx2-programmer 0x16C0 0x072F program_vendor 0x04b4 0x8613
PAUSE
```

In a Cygwin shell in the `C:/SiGe_SE4120L_CU_USBdriver/bin` directory type:

./loader.bat

and press Return.

Next type:

./program_vendor.bat

and press Return.

The last part of Command Prompt dialog should show the new VID and PID.

The new ID can be verified in Windows Control Panel at:

`System>Device Manager>SE4110 EK3 Properties>Details.`

6.4.1 Troubleshooting

In general, the programmed VID and PID can be changed by editing and running modified `'loader.bat'` and `'program_vendor.bat'` files. In the event that the VID and PID cannot be successfully changed, at the last resort it is possible replace the EEPROM with an un-programmed one. The board ID will then resort back to the default Cypress ID which can be reprogrammed as described in the previous section.

7 Matlab Analysis

This section describes the use of Open-source Matlab software to display time domain and frequency domain plots and perform a GPS acquisition on the data captured on the host PC by the Open-source driver.

7.1 Matlab Requirements

The Matlab processing was verified using Matlab version R2006a. The data plot outputs require the Signal Processing Toolbox option.

7.2 Matlab Scripts

The following Matlab scripts are used for the Matlab analysis. These are included in the Software CD.

Main Script:

- `findGPS1.m`

Sub-functions:

- `acquireGPS.m`
- `cacode.m`
- `fileConvert_2bit_IQ_pulse.m`
- `fileConvert_4bit_IQ_pulse.m`
- `fileConvert_2bit_IQ_parallel.m`
- `fileConvert_4bit_IQ_parallel.m`

These files should be stored in a convenient directory on the C: drive. The normal location is in the `'C:\Program Files\MATLAB\R2xxx\work'` folder. If you are not in the root directory within Matlab when running this main script, Matlab will ask if you wish to change the workspace. All Matlab scripts should be stored in the same folder.

`FindGPS1.m` is the primary script. The other scripts are sub-functions which are called by `findGPS1.m`, with the exception of `cacode.m` which is called by `acquireGPS.m`. `acquireGPS.m` does the acquisition processing and `cacode.m` generates the C/A code PRNs.

`FindGPS1.m` is a menu-driven script to ensure that the correct parameters are selected for each data output mode. It allows the analysis of data captures for the sixteen viable data output modes. The remaining five modes do not allow data captures because they are either unsampled or byte sync modes which cannot be captured by the USB controller in its current configuration.

7.3 Source Data

7.3.1 Data Files

`FindGPS1.m` expects the captured data samples to be in the format:

Real IF data: R0, R1, R2, R3…

Basband I/Q data: I0, Q0, I1, Q1, I2, Q2…

Where each sample is a signed 8-bit word which corresponds to each sample clock output of the SE4120L. For captured data from real modes, the samples contain successive single or multi-bit real values; for baseband I/Q modes the samples contain alternate single or multi-bit I or Q values.

Due to peculiarities of the Open-source driver implementation, only real modes are captured in a format which can be processed directly. For baseband I/Q modes, the data has to be converted into the correct format by the one of the four fileConvert sub-functions. This is carried out automatically by findGPS1 using a menu driven selection process.

The Open-source driver software stores the captured data in:

`C:/SiGe_SE4120L_CU_USBdriver/bin/gnss.bin.`

For baseband I/Q modes the fileConvert sub-routines reformat this data and store it in:

`C:/SiGe_SE4120L_CU_USBdriver/bin/gnss_out.bin.`

`findGPS1` is configured to take either `'gnss.bin'` or `'gnss_out.bin'` depending on whether the data is in real IF or baseband I/Q format.

7.3.2 Data Formatting

The table below summarizes the captured data file format, the converted data output file format (where applicable) and the various parameter settings for the Matlab software. It is also possible to verify these parameters by examining the various Matlab scripts.

The following sections give a detailed description of the data conversions to the Open-source driver data output. These Matlab conversion routines were developed in preference to rationalizing the Open-source driver source code.

7.3.2.1 Real IF Modes

In real modes the byte data transferred to the GPIOs from the SE4120L is in the form {b7, b6, b5, b4, b3, b2, b1, b0} where the values are either {x, x, x, x, x, x, M, S} for 2-bit mode or {x, x, x, x, x, x, x, S} for 1-bit mode. The Open-source driver takes 'S' as the SIGN bit and 'M' as the MAG bit and generates the correct signed value in the range {-3, -1, +1, +3} for 2-bit mode and {-1, +1} for 1-bit mode.

In findGPS1, the data value is re-inverted, because the Open-source driver assumes a negative value for SIGN=1. This is the convention for signed numbers, but in hardware terms SIGN=1 usually indicates a positive value. The driver inversion has been physically verified by tying GPIO PB0 high.

7.3.2.2 2-bit I/Q Pulse Mode

In 2-bit I/Q pulse mode the byte data transferred to the GPIOs from the SE4120L is in the form {b7, b6, b5, b4, b3, b2, b1, b0} where the values are alternately {x, x, x, x, x, x, sync=1, Si} and {x, x, x, x, x, x, sync=0, Sq}. The sync pulse identifies the Si sample.

The driver treats the signed values as it does the real mode values. Si samples are {-3, +3} and Sq samples are {-1, +1}. It is therefore only necessary to map {-3, +3} values to {-1, +1} values for the Si sample case. The identification of the Si sample is strictly not necessary, as alternate samples will have a 90 degree phase difference and in absolute terms it doesn't matter whether they are I or Q samples. However, the `fileConvert_2bit_IQ_pulse.m` does search for the first Si sample and then performs the conversion from there. For this reason the output file size is reduced to {input capture file size – 2} since the first Si sample could be at latest the second sample in the captured data file.

The driver inversion of the sign bit is reversed for the same reasons as explained for the real modes.

7.3.2.3 4-bit I/Q Pulse Mode

In 4-bit I/Q pulse mode the byte data transferred to the GPIOs from the SE4120L is in the form {b7, b6, b5, b4, b3, b2, b1, b0} where the values are the four repeating bytes {x, x, x, x, x, x, sync=1, Si}, {x, x, x, x, x, x, sync=0, Mi}, {x, x, x, x, x, x, sync=0, Sq}, {x, x, x, x, x, x, sync=0, Mq}. The sync pulse identifies the Si sample.

The driver treats the signed values as it does the real mode values. Si samples are {-3, +3}, and Mi, Sq and Mq samples are {-1, +1}. It is therefore necessary to map {-3, +3} values to {-1, +1} values for the Si sample. Here it is essential to identify the Si sample. `FileConvert_4bit_IQ_pulse.m` does a search for the first Si sample and then performs the conversion from there. For this reason the output file size is reduced to {input capture file size – 4} since the first Si sample could be at latest the third sample in the captured data file.

The driver inversion of the sign bit is reversed for the same reasons as explained for the real modes.

7.3.2.4 2-bit I/Q Parallel Mode

In 2-bit I/Q parallel mode the byte data transferred to the GPIOs from the SE4120L is in the form {b7, b6, b5, b4, b3, b2, b1, b0} where the values are determined by {x, x, x, x, x, x, Sq, Si}.

The driver treats the signed values as it does the real mode values. The sign of the value determines the value of the Si sample: {-1, -3} indicates Si is positive and {+1, +3} indicates Si is negative. This removes the driver inversion of the sign bit which is reversed for the same reasons as explained for the real modes. The magnitude of the value determines the value of the Sq sample: {-3, +3} indicates Sq is +1 and {-1, +1} indicates Sq is -1.

The output file size is twice the size of the capture file size in order to produce alternate I/Q samples for `findGPS1.m.`

7.3.2.5 2-bit I/Q Parallel Mode

In 4-bit I/Q parallel mode the byte data transferred to the GPIOs from the SE4120L is in the form {b7, b6, b5, b4, b3, b2, b1, b0} where the values are determined by {x, x, x, x, Mq, Mi, Sq, Si}.

The driver treats the signed values as it does the real mode values, however, it rearranges the output value to the form {x, x, x, x, Sq, Mi Mq Si}. The following 16 values are possible: {-15, -13, -11, -9, -7, -5, -3, -1, +1, +3, +5, +7, +9, +11, +13, +15}.

The four bit values are extracted by signed decimal arithmetic as follows:

- If data > 0 Si=-1 else 1

The value of Si is determined from the sign of the number. The driver inversion of the sign bit is reversed for the same reasons as explained for the real modes.

- Take absolute data value and subtract 8. If data>0, Mq =3, else 1

The value of Mq is determined by taking the absolute value and subtracting 8. The second MSB has a symmetrical bit pattern about zero, so only one side of zero needs to be considered. The original value needs to be greater than 8 for the second MSB to be set to 1.

- Move data 4 closer to zero. If data>0, Mi=3, else 1

The value of Mi is determined by shifting the data value from the previous calculation to create a decision threshold about zero, because the third MSB has a resulting asymmetric pattern about zero.

- Move data 2 closer to zero. If data>0, Sq=1, else -1

The value of Sq is determined by shifting the data value from the previous calculation to create a decision threshold about zero, because the fourth MSB has a resulting asymmetric pattern about zero.

- I=Si*Mi and Q=Sq*Mq

The 2-bit values of I and Q are calculated by multiplying the SIGN value by a MAG value of 1 or 3.

The output file size is twice the size of the capture file size in order to produce alternate I/Q samples for `findGPS1.m.`

Data Output Mode and Byte Format

Mode	Sync	Sampfreq (MHz)	Data Format	IF or lowest IF alias (MHz)	intfreq (MHz)	GPIO: PB7…PB0 b7, b6, b5, b4, b3, b2, b1, b0 Byte 1, byte 2…	gnss.bin format Byte 1, byte 2…	gnss_out.bin format Byte 1, byte 2…	File Size
1	Serial pulse	8.184	2-bit I/Q	0	0	x, x, x, x, x, x, sync=1, Si x, x, x, x, x, x, sync=0, Sq	Si: -3, +3 Sq: -1, +1	I: -1, +1 Q: -1, +1	input-2
2	Serial pulse	4.092	2-bit I/Q	0	0				input-2
3	Serial pulse	4.092	4-bit I/Q	0	0	x, x, x, x, x, x, sync=1, Si x, x, x, x, x, x, sync=0, Mi x, x, x, x, x, x, sync=0, Sq x, x, x, x, x, x, sync=0, Mq	Si: -3, +3 Mi: -1, +1 Sq: -1, +1 Mq: -1, +1	I: -3, -1, +1, +3 Q: -3, -1, +1, +3	input/2-4
8	Parallel	5.456	2-bit real	1.364	4.092	x, x, x, x, x, x, M, S	R: -3, -1, +1, +3	-	input
9	Parallel	16.368	1-bit real	4.092	4.092	x, x, x, x, x, x, x, S	R: -1, +1	-	input
10	Parallel	16.368	2-bit real	4.092	4.092	x, x, x, x, x, x, M, S	R: -3, -1, +1, +3	-	input
12	Parallel	5.456	1-bit real	1.364	4.092	x, x, x, x, x, x, x, S	R: -1, +1	-	input
13	Bypassed	16.368	2-bit real	4.092	4.092	x, x, x, x, x, x, M, S	R: -3, -1, +1, +3	-	input
14	Parallel	16.368	2-bit I/Q	0	0	x, x, x, x, x, x, Sq, Si	Si Sq: -3, -1, +1, +3	I: -1, +1 Q: -1, +1	input*2
15	Parallel	8.184	2-bit I/Q	0	0				input*2
16	Parallel	5.456	2-bit I/Q	0	0				input*2
17	Parallel	4.092	2-bit I/Q	0	0				input*2
18	Parallel	16.368	4-bit I/Q	0	0	x, x, x, x, Mq, Mi, Sq, Si	Si Mq Mi Sq: -15, -13, -11, -9, -7, -5, -3, -1, +1, +3, +5, +7, +9, +11, +13, +15	I: -3, -1, +1, +3 Q: -3, -1, +1, +3	input*2
19	Parallel	8.184	4-bit I/Q	0	0				input*2
20	Parallel	5.456	4-bit I/Q	0	0				input*2
21	Parallel	4.092	4-bit I/Q	0	0				input*2

7.3.3 Data File Size

The data capture file size is fixed at 1 MByte, (i.e. 1,048,576 bytes, which is equivalent to 2^{20}). The file size used for Matlab analysis varies from {input file size – 4} to {2*input file size} (i.e. 1,048,572 to 2,097,152). Refer to the previous table for further details. This is a convenient small size for storage and analysis, but the acquisition results are somewhat 'noisy'.

It is possible to specify a smaller input file size than 1,048,576 bytes, but not larger than the captured data size. Tracking and position fix calculations are not supported, but this requires a 38 sec data capture with resulting file sizes of up to 680 MByte.

The data capture file size is specified in the Open-source driver file `ogusb-lite.cc` on line 504. To change this, edit this line and re-build `ogusb-lite.exe`.

```
504   unsigned int loops = 1 * 64;  // 1 MB (x * 64 * 16 kB) only need 1MB.
```

7.3.4 Source Code Editors

If you wish to edit the source code, you need an editor. It is possible to use an Open-source Windows editor called Notepad++ which is suitable for many types of text / code C, C++ etc:

http://notepad-plus.sourceforge.net/uk/site.htm

7.3.5 Source Code Compilers

The makefiles in the Host / Target source code are written for specific Open-source compilers, which are gcc and sdcc respectively.

These are only required if you want to change the source code and rebuild the project.

Host:

Open-source compiler for Windows C++ host (`ogusb-lite.exe`)

http://gcc.gnu.org/

Target:

Open-source compiler for 8051C target (gn3s_firmware.ihx)

http://sdcc.sourceforge.net/

7.3.6 Viewing Data Files

The Matlab produces time domain plots of the first 50 samples of the single or multi-bit real IF and baseband I/Q data. It is possible to determine the data sequence from these plots.

For more detailed analysis an application is required to view the binary files. One possible binary viewer is FlexHEX from Inv Softworks LLC who offer a free trial version available from:

http://www.flexhex.com/download/

Alternatively, try the Open-source viewer called 'hexplorer' available from:

http://hexplorer.sourceforge.net/

7.4 Running the Matlab

To run the Matlab from an arbitrary folder on the C: drive, navigate to `findGPS1.m`, open it and select 'run'. This may require the Matlab Current Directory to be changed. Alternatively, with the appropriate Current Directory, type 'findGPS1' in the Command Window.

7.4.1 Matlab Input

The Matlab presents a series of menus:

- **Enter data output mode of captured data (1-21):**

The mode number entered should correspond to the mode used for the last data capture. FindGPS1 will use the file `‘gnss.bin’` in `‘C:\SiGe_SE4120L_CU_USBdriver\bin’`. Only the 16 relevant modes for Matlab processing can be selected. If the mode selected is not compatible with the data captured then the subsequent acquisition will fail to acquire any satellites.

- **Input data file size in bytes: "1": 1048576 (1MB) or "2": other ? 1**

The current capture size is fixed at 1 MByte. It is possibly to enter a smaller data file size for processing and the processing will be faster, but the acquisition result noisier. Entering a larger file size will cause the Matlab to crash unless the data capture file is correspondingly larger than 1 MByte.

- **Enter TCXO freq in Hz: "1": 16.368e6 or "2": other ? 1**

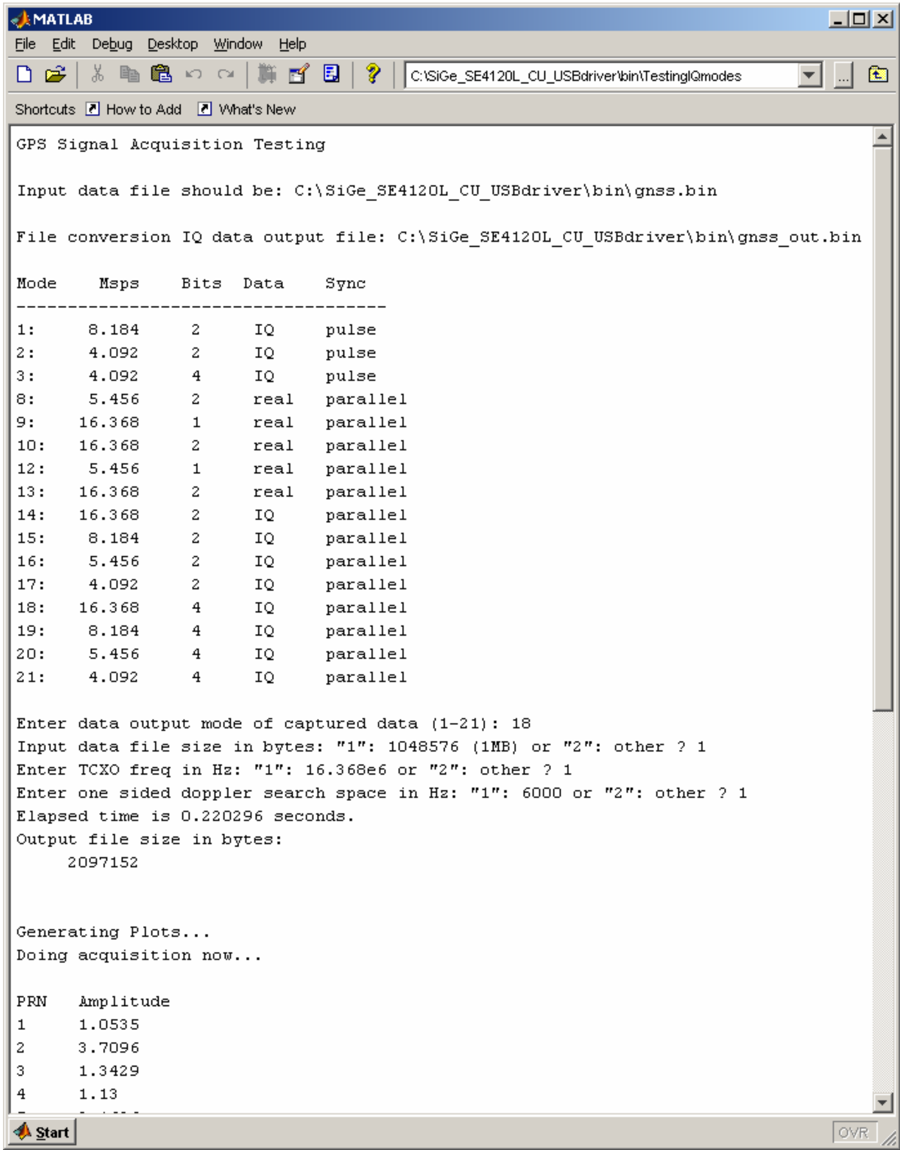
The TCXO fitted to the board has a frequency of 16.368 MHz. It is possible to replace the TCXO with another part with a slight frequency offset on 16.368 MHz. If a 16.3676 MHz TCXO is fitted, for example, the new TCXO frequency should be entered, otherwise the acquisition will fail to acquire any satellites.

Note also that the baseband I/Q data will no longer be centred at 0 Hz, and the resulting offset is currently not factored into `findGPS1.m`.

- **Enter one sided doppler search space in Hz: "1": 6000 or "2": other ? 1**

It is left to the user to experiment with different search spaces.

The image below shows the Command Window for menu selection and subsequent acquisition for the 4-bit I/Q parallel data output mode 18 example:

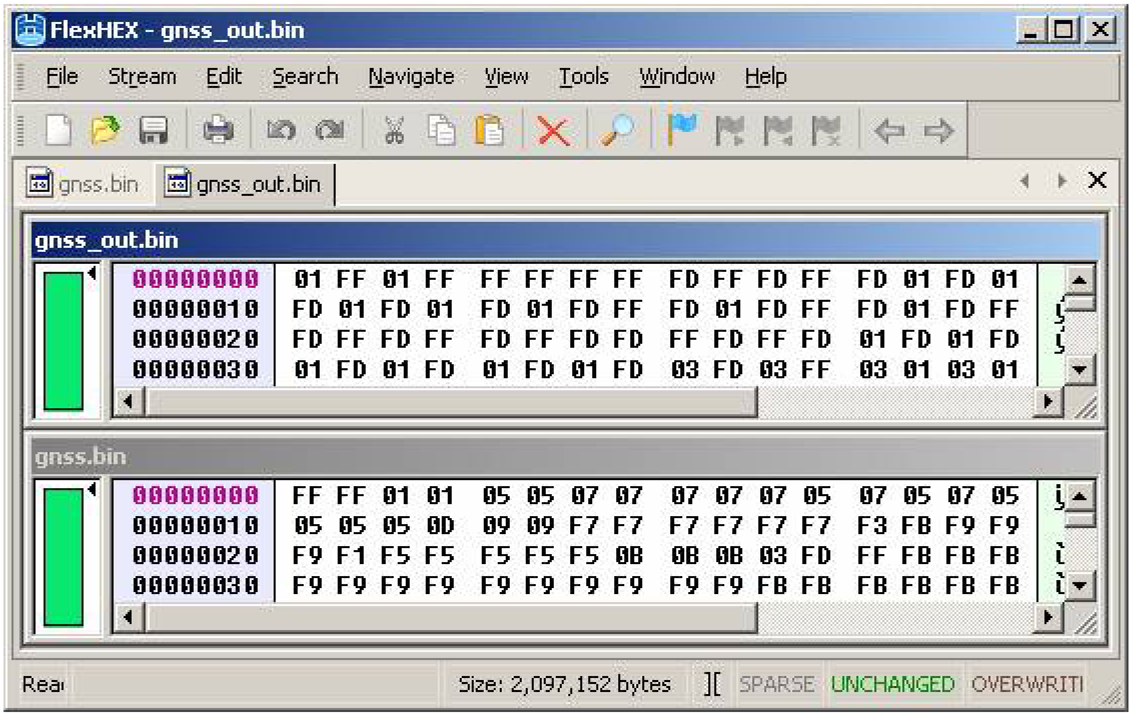



```
GPS Signal Acquisition Testing

Input data file should be: C:\SiGe_SE4120L_CU_USBdriver\bin\gnss.bin

File conversion IQ data output file: C:\SiGe_SE4120L_CU_USBdriver\bin\gnss_out.bin

Mode    Msps    Bits  Data    Sync
------------------------------------
1:      8.184     2     IQ      pulse
2:      4.092     2     IQ      pulse
3:      4.092     4     IQ      pulse
8:      5.456     2     real    parallel
9:     16.368     1     real    parallel
10:    16.368     2     real    parallel
12:     5.456     1     real    parallel
13:    16.368     2     real    parallel
14:    16.368     2     IQ      parallel
15:     8.184     2     IQ      parallel
16:     5.456     2     IQ      parallel
17:     4.092     2     IQ      parallel
18:    16.368     4     IQ      parallel
19:     8.184     4     IQ      parallel
20:     5.456     4     IQ      parallel
21:     4.092     4     IQ      parallel

Enter data output mode of captured data (1-21): 18
Input data file size in bytes: "1": 1048576 (1MB) or "2": other ? 1
Enter TCXO freq in Hz: "1": 16.368e6 or "2": other ? 1
Enter one sided doppler search space in Hz: "1": 6000 or "2": other ? 1
Elapsed time is 0.220296 seconds.
Output file size in bytes:
     2097152


Generating Plots...
Doing acquisition now...

PRN   Amplitude
1     1.0535
2     3.7096
3     1.3429
4     1.13
```

7.4.2 Matlab Output

The following uses the 4-bit I/Q parallel data mode as an example case. Files for this and other modes are included in the Software CD. The corresponding Command Window is shown above. The output data file size is reported as '2097152' bytes, which is 2*1,048,576 bytes.

The image below shows the captured data file `'gnss.bin'` and the converted file for processing, `'gnss_out.bin'`. Note that the bytes in `'gnss.bin'` have 16 possible values and the bytes in `'gnss_out.bin'` have four possible values.



'Figure1' is generated first. The time domain plots show the first 50 I and Q samples. These signed decimal values correspond to the signed hex values in `'gnss_out.bin'` above.

The histograms show the distribution of the 4 data values {-3, -1, +1, +3}. As expected there is an equal distribution of positive and negative samples, indicating an equal mark space ratio for the SIGN bit. There is a greater distribution of '1's than '3's because the MAG bit nominally has a 33 % duty cycle. This allows some verification of a data capture file using a 'clean' conducted GPS input signal. Data captures with a radiated set-up may exhibit distortion of the histogram distribution due to interference and 'feed-round' effects.

The FFT plot shows the baseband I/Q spectrum for IF filter response with the GPS signal superimposed. The first alias is at 16.368 MHz.

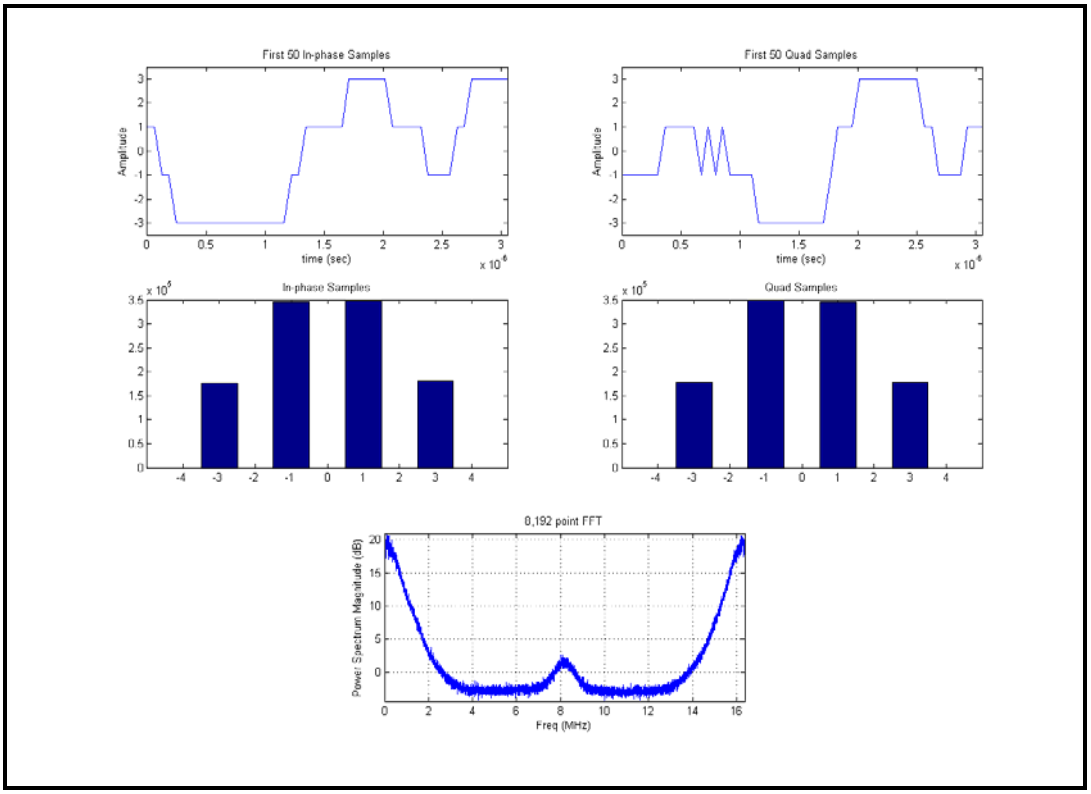
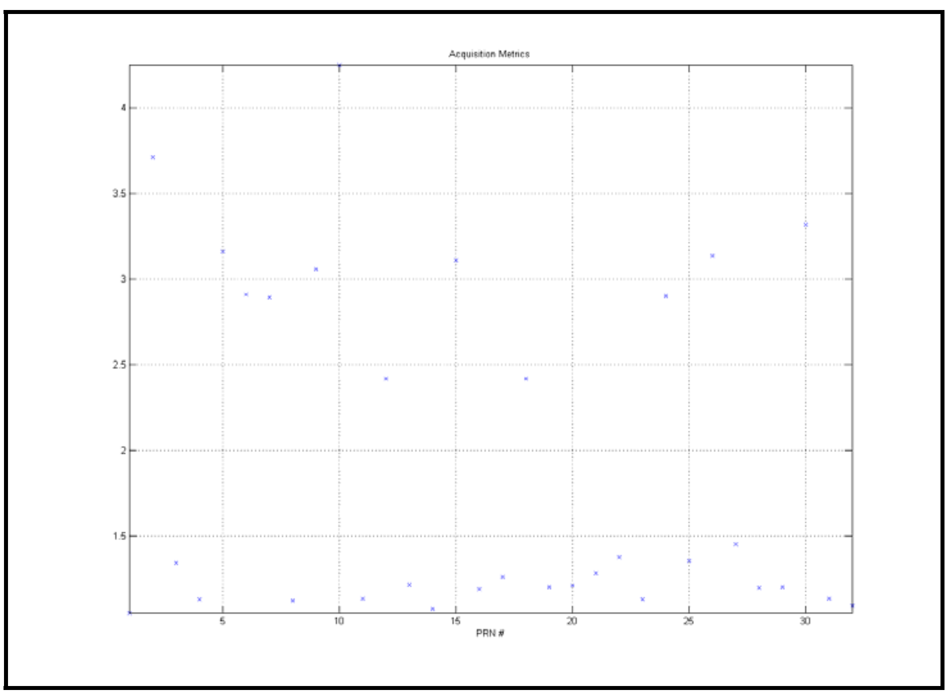


Figure 2 is generated at the end of the acquisition, see below:



The Matlab searches all possible 32 PRN codes, and all possible code phases. The decision threshold between noise and a valid satellite PRN is typically 1.75. Note that the acquisition is noisy and will vary for subsequent new captures. To determine CNRs requires some estimation of the noise to be made. This can be more accurately calculated in tracking mode and with knowledge of the AGC voltage.

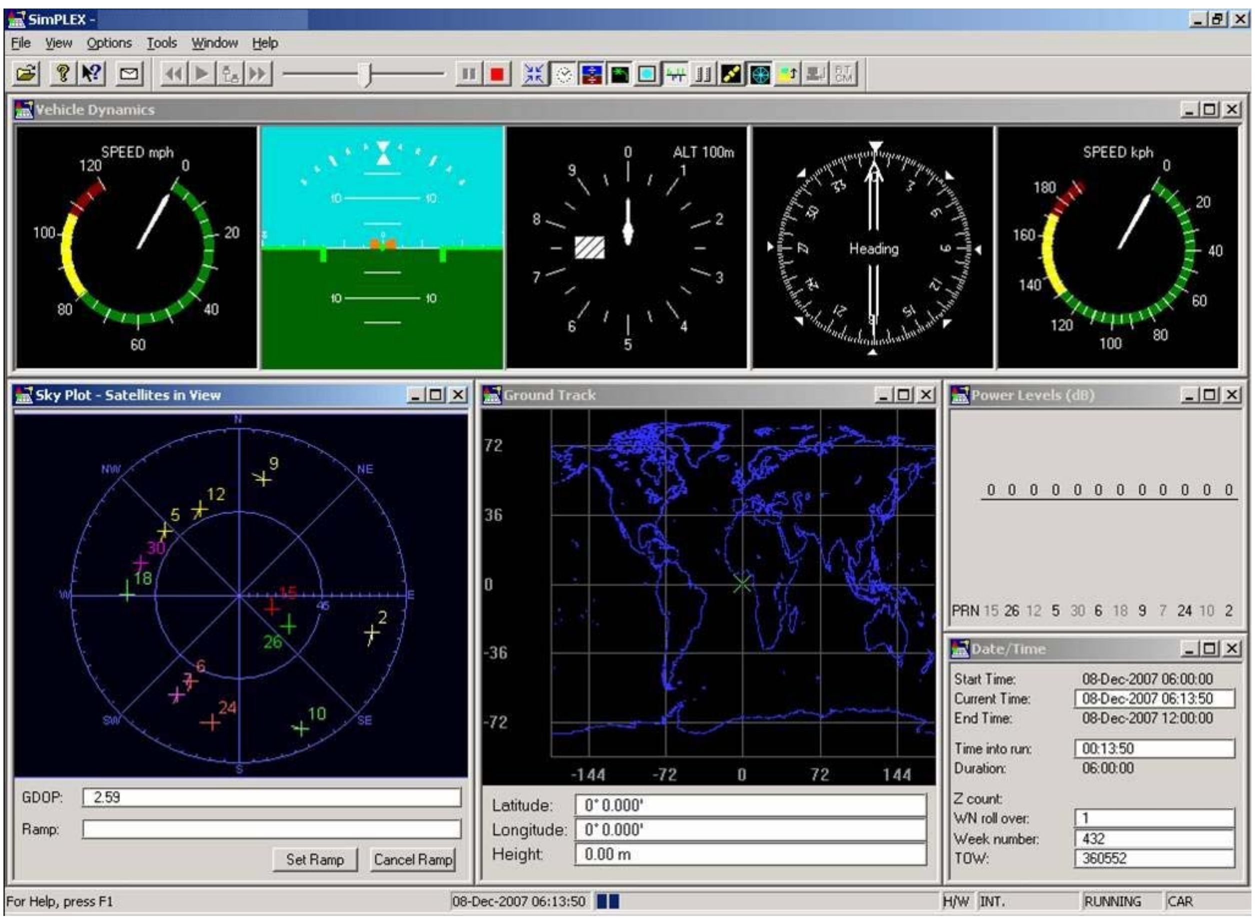
The acquisition data can be saved by copying the data in the Command Window and pasting into a text editor.

7.4.3 Example Data Captures

A set of example data captures and accompanying plots is supplied in the Software CD. There are captures for all 16 valid modes for both GPS and Galileo filter selections. Where applicable the `'gnss_out.bin'` data conversion file is included. This provides useful reference data in case problems are encountered with the Matlab.

These data captures were made using a Spirent STR4500 set to +10 dB on the nominal -130 dBm output. This elevated power level was used to ensure reliable acquisition of the 12 satellites above the noise floor. For this reason, the GPS signal can be seen above the filter response noise floor.

The image below shows the STR4500 application window.



8 Test Measurements

This section describes a few measurements on the SE4120L-EK3.

8.1 Test Equipment

The following test equipment is recommended for measurements:

- PC with USB 2.0 interface
- PSU capable of delivering 100 mA, +5 V.
- 1.6 GHz RF CW signal generator.
- Spectrum analyzer (4.192 MHz).
- Digital Multi-Meter (DMM)

8.2 Supply Current

The supply current can be measured by modifying a USB cable by breaking into the red +5V lead and inserting a series ammeter.

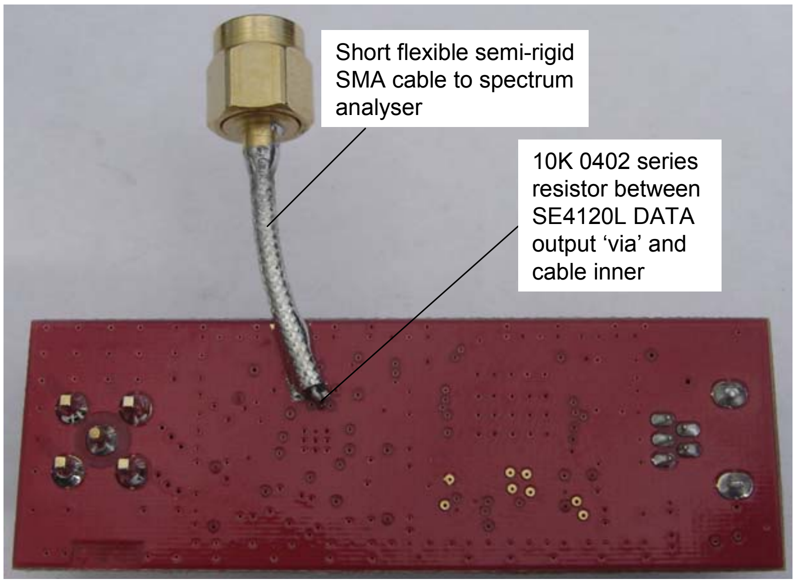
The nominal current consumption of the SE4120L-EK3 is 37 mA. On occasions, the board can be incorrectly configured on power up. In this case, the current consumption may only be 19 mA, and the board should be power cycled.

8.3 3 dB down Noise Figure Measurement

8.3.1 Board Setup

Solder a short SMA semi-rigid 'pig-tail' to the underside of the board. Remove an area of solder resist to allow the cable to be soldered to the ground plane. Position the cable so the short inner conductor at the open end is directly above the data via near the SE4120L (see Appendix G: Layer 4 Tracking (underside view)). Mount a 10K 0402 resistor between the cable inner and the data via.

SMA 'pig-tail' arrangement for '3 dB down' noise figure measurement:



Set the board to operate in Mode 13 (2-bit SIGN/MAG real, parallel at 16.368 MHz, block converter bypassed):

```
./ogusb-lite.exe 13
```

8.3.2 Measurement Procedure

This measurement determines the overall noise figure from RF input to data output using the '3 dB down' method.

1) Connect a signal generator at 1575.52 MHz (CW) set to deliver -75 dBm at J1. An MCX (M) to SMA (F) adaptor is required. Ensure there is a DC block at the signal generator output, or else disable the antenna bias by removing resistor R2.
2) Connect a spectrum analyzer via a cable to the pig-tail output. Ensure that the spectrum analyzer is rated for the DC level on the SE4120L data SMA output or else fit a DC block.
3) Set the spectrum analyzer to 4.192 MHz centre freq, 10 kHz span, 3 kHz RBW, 30 Hz VBW, ref level –52 dBm, scale 1 dB/div. If necessary set the tone peak to centre frequency and adjust the reference level. Ensure the analyzer is not overloaded prior to reducing the VBW from the auto setting.
4) Measure the tone amplitude using the peak marker.
5) Select marker delta, or use another equivalent marker function. Reduce the signal generator output power until the SIGN amplitude drops by exactly 3 dB. Record the signal generator level, RFin.
6) The radio noise figure at J1 is given by:

   NF_{RX} = RFin + 110.6 dB (assuming all cable losses have been accounted for)

The expected noise figure is 2.5 dB at the RF connector, J1.

The measurement can be calibrated for signal generator inaccuracy by making this measurement with and without an external LNA with known noise figure.

Refer to Section 5.12 of the SE4120L-AK1 Applications Board User Guide [2] for full details of this measurement and how to apply correction.

9 Product Development

9.1 Hardware Considerations

Customers wishing to develop this SE4120L USB radio evaluation board into a product should be aware of the hardware limitations of the current implementation and the performance trade-offs described below. Many of the hardware issues have been covered in earlier sections.

One of the major advantages of Software GPS solutions is that the cost of a hardware baseband processor is eliminated. However, high speed USB 2.0 controllers can have significant cost also.

The customer should be aware of the high speed 480 Mbps nature of the physical interface between the hardware and the host PC. Note that the full payload data rate (53 Mbytes/sec or 424 Mbps) would not be utilized for any of the data output modes. In practice the host processor loading will limit the payload data rate.

9.1.1 Antenna Switching

For designs requiring internal and external antenna capability (e.g. for PND applications in cars with high attenuation, athermic windscreens) it is recommended that a switched MMCX connector or equivalent is used. The mechanical switch solves both RF and antenna / LNA bias switching issues, and does not require detection circuitry. An equivalent circuit implementation for external antenna detection and switching will require a not insignificant number of external components.

9.1.2 USB Controller

The Cypress CY7C68013A USB controller has several features that make it suitable for Software GPS applications. These features should be borne in mind when investigating lower cost USB alternatives. Firstly, it has a microcontroller which is required to software program the SE4120L into the desired data output mode, unless one of the four hardware programmable I/Q modes are suitable. Secondly, it has a 4 Kbyte FIFO buffer which will helps reduce the frequency of buffer overruns. Thirdly, it allows detection and mitigation of buffer overflows.

9.2 Software Considerations

It is possible to adapt the SE4120L-EK3 Open-source driver to run in real-time mode and continuously transfer data to the host PC. However, there are issues with buffer overruns which make this a non-trivial exercise. The buffer overruns may cause the application software to hang, and loss of position fix and the necessary re-acquisition of satellites during operation. This behavior may not be considered acceptable in a product. Ideally the driver software needs to be rewritten at the earliest opportunity.

There are a number of third party providers of such real-time Software GPS solutions who have solved the continuous data transfer issues, and have real-time Software GPS solutions that run on the host PC.

9.3 Performance Trade-Offs

9.3.1 Data Rate

There are a number of trade-offs between GPS data rate, processor load and USB date rate which need to be considered. A reduced data rate is required to operate the host processor with a sensible loading. Reducing the data rate will increase the implementation loss of the processing. The data rate should be selected with consideration to the radio noise figure and antenna selection. These factors will determine the overall acquisition and tracking performance of the solution. Typically the acquisition sensitivity threshold will be 15 dB higher than the tracking sensitivity threshold, and it is this parameter which will determine how well the solution performs in practice.

As a rough bench-mark the 2-bit real IF parallel data output mode at 5.456 MHz has the potential of giving acquisition and tracking sensitivity performance comparable with that of SiRFstarIII. The level of processor loading should be such that it should run on a host PC with a 1 GHz minimum clock processor. Running 2-bits at the maximum 16.368 MHz rate, however, would require a processor clock in excess of 2 GHz, which would restrict the range of host PCs that support it, and is therefore less attractive.

Reducing the data rate to 1-bit at 5.456 MHz will in theory increase the implementation loss by 1.2 dB (see table in Section 9.3.4). This assumes that an ideal correlator is used to de-spread the GPS signal. In practice, the implementation loss may increase by up to 2 dB. For practical purposes this is the lowest data rate which will give acceptable GPS performance. For this reduced data rate, the performance of the antenna and the radio noise figure are more critical.

An advantage of the 1-bit 5.456 MHz solution is that is within the maximum 8 Mbps (1 MByte) pay-load data rate of USB 1.1 or USB 2.0 full-speed 12 Mbps interfaces. This allows lower cost USB interface devices to be used, typically half the price of a USB 2.0 high speed (480 Mbps) controller. The cost saving has to be offset by the cost of additional hardware required to pack the single 1-bit data samples into 8-bit words. This can be typically implemented with a serial-to-parallel converter logic IC or a programmable logic device (e.g. CPLD), or by the USB interface device itself if it contains a suitable microcontroller.

Another advantage of the full-speed USB 12 Mbps solution is that it is supported by a larger number of host PCs and legacy devices.

9.3.2 Real IF vs Baseband I/Q Data

The baseband I/Q modes may be preferred over real IF modes for software architectural reasons. The multi-bit nature of these modes also lends itself to the parallel format of the GPIOs, with less redundancy in the 8-bit word.

There may be advantages in reduced processor loading if the I/Q modes are used, as it removes the mix-to-baseband processing step from the software. The software needs to do a frequency adjustment on each channel anyway to remove Doppler shift, but with an I/Q input, the frequency offset is low enough that this can be deferred to a much later stage of the processing where the signal has been de-spread and integrated a bit so that the processing rate is much lower. This may ultimately reduce processor load if the software is structured correctly

The sample frequency can be reduced to a lower value than for real IF modes, e.g. 4.092 MHz or even 2.046 MHz with external decimation if there is a limit on processor sample rate. However, there may be issues with the exact sample frequency (e.g. 2.046 MHz sample rate rather than a preferred 2.048 MHz sample rate) which make this impractical. The sample rate for real IF modes is limited by aliasing issues (e.g. real IF mode at 8.184 MHz is no possible since an alias falls on the 4.092 MHz IF) and the proximity of the 0 Hz frequency boundary.

In general, doubling the number of bits to give separate I and Q samples does not given an implementation loss advantage over the real IF case for the same bit rate. Taking the 2-bit real IF mode at 5.456 MHz (10.912 Mbps) as a benchmark, the 4-bit and 2-bit baseband I/Q modes at 4.092 MHz (at 16.368 Mbps and 8.184 Mbps respectively) may be worth considering. Referring to the theoretical implementation loss table in Section 9.3.4, the 2-bit I/Q mode gives implementation loss performance between that for 1-bit and 2-bit real at 5.456 MHz for an intermediate bit rate. In a practical case the relative performances may vary. Certainly for a given I/Q mode bit rate the implementation loss decreases as sample rate is traded for bits. A 4-bit baseband I/Q mode at 2.046 MHz is also possible, but would require external decimation by two.

9.3.3 Radio Performance

Assuming other radio implementation loss contributors to be negligible (e.g. synthesizer phase noise, IF group delay ripple etc), then:

Overall implementation loss (dB) = radio noise figure (dB) + SW processing implementation loss (dB)

As the software implementation loss is increased with greater data decimation and with fewer bits per sample, there is a greater importance attached to maximizing the radio performance. This means minimizing the radio noise figure by using a low noise figure external LNA. Typically the overall performance will be dictated by the noise figure of the external LNA with only a tenth of a dB or so additional second stage contribution from the subsequent stages. For example, a 0.9 dB noise figure LNA ahead of the SE4120L should give an overall noise figure of about 1 dB, which is a 1.2 dB improvement on using the SE4120L alone. The designer should resist the temptation to add excessive front-end gain, as it will tend to operate the SE4120L in 1-bit mode and tend to cancel any implementation loss advantage of any multi-bit implementation.

The antenna can make a significant difference to overall system performance. For optimum antenna performance, and especially with 1-bit implementations at 5.456 MHz, a square patch antenna should be considered. This is inherently suited to the right hand circular polarization (RHCP) of the GPS signals, and tends to have a hemispherical radiation pattern which is ideally suited to open sky reception when pointed upwards. A 25 mm square patch antenna will give the best performance. Smaller patch antennas of 18 mm and 13 mm will give successively poorer performance due to lower antenna gain. In addition, they will tend to be more narrow-band, which makes them more difficult to match for best performance. Even a 13 mm patch antenna should be considered in preference to a linear, omni-directional antenna due to its RHCP and hemispherical radiation pattern properties. Patch antennas tend to exhibit more omni-directional properties when the reference ground plane is reduced in size.

In practical terms, the overall implementation loss is given by:

Overall implementation loss (dB) = GPS signal level (e.g. -130) (dBm) + 174 (dBm/Hz) – CNR reading (dB/Hz)

This assumes correct calculation of the CNR or C/No.

9.3.4 Theoretical Software Implementation Loss

The table below summarises the available output modes for the SE4120L and also provides the simulated implementation loss figures for each of the modes, assuming an ideal correlator is used to de-spread the GPS/Galileo signal.

Note these ideal figures should be used as a guideline, as in practice the absolute implementation loss and relative implementation losses will increase.

	Sample rate at fref = 16.368 MHz			
OUTMODE format	16.368 MHz	8.184 MHz	5.456 MHz	4.092 MHz
1-bit real	Parallel Parallel un-sampled GPS loss 1.26 dB Galileo loss 1.76 dB		Parallel Serial byte GPS loss 2.60 dB Galileo loss >4 dB	
2-bit real	Parallel Parallel (Block Convertor bypassed) GPS loss 0.66 dB Galileo loss 1.06 dB		Parallel Serial byte GPS loss 1.40 dB Galileo loss >4 dB	
2-bit I/Q	Parallel GPS loss 1.32 dB Galileo loss 2.02 dB	Parallel Serial pulse GPS loss 1.49 dB Galileo loss 2.62 dB	Parallel Serial byte GPS loss 1.60 dB Galileo loss 3.06 dB	Parallel Serial pulse Serial byte GPS loss 1.86 dB Galileo loss 3.78 dB
4-bit I/Q	Parallel GPS loss 0.70 dB Galileo loss 1.24 dB	Parallel GPS loss 0.80 dB Galileo loss 1.56 dB	Parallel GPS loss 0.87 dB Galileo loss 1.95 dB	Parallel Serial pulse GPS loss 1.02 dB Galileo loss 2.44 dB

9.4 SiGe Radio IC Selection

The SE4120L will support both real IF and baseband I/Q modes with 16.368 MHz frequency plan. Since the USB device interface is typically a parallel one, it is possible to replace the SE4120L with SiGe Semiconductor's pin-for-pin compatible SE4110L GPS receiver IC for real IF, parallel modes. The RF performance is similar, and the choice may be one of convenience.

The SE4110L can either be software programmed to 5.456 MHz decimated mode, or else the decimation of the data (i.e. discard two out of three data samples) can be carried out by the USB controller. The essential difference between the data decimation in the SE4120L and the SE4110L is that the decimation in the SE4120L is performed by the block-converter, whereas for the SE4110L the ADC sample clock is divided down.

The advantage of the SE4110L is that it will additionally support a 13 MHz frequency plan, and software programmed decimated data rates of 6.5 MHz and 4.33 MHz.

When using the SE4120L in non-decimated, 2-bit, real parallel full data rate mode at 16.368 MHz there is a choice of two functionally equivalent modes, either using the block converter or by-passing it. In terms of performance, there is a slight increase in radio noise figure (typically a tenth of a dB or so) due to the feed-round issues associated with running the additional block converter circuitry. In practice, this mode is unlikely to be used with the SE4120L, but the reader should be aware of this preference.

10 References

[1] 184-DST-01 SE4120L PointCharger™ GNSS Receiver IC, Rev 3.3, May-06-2008

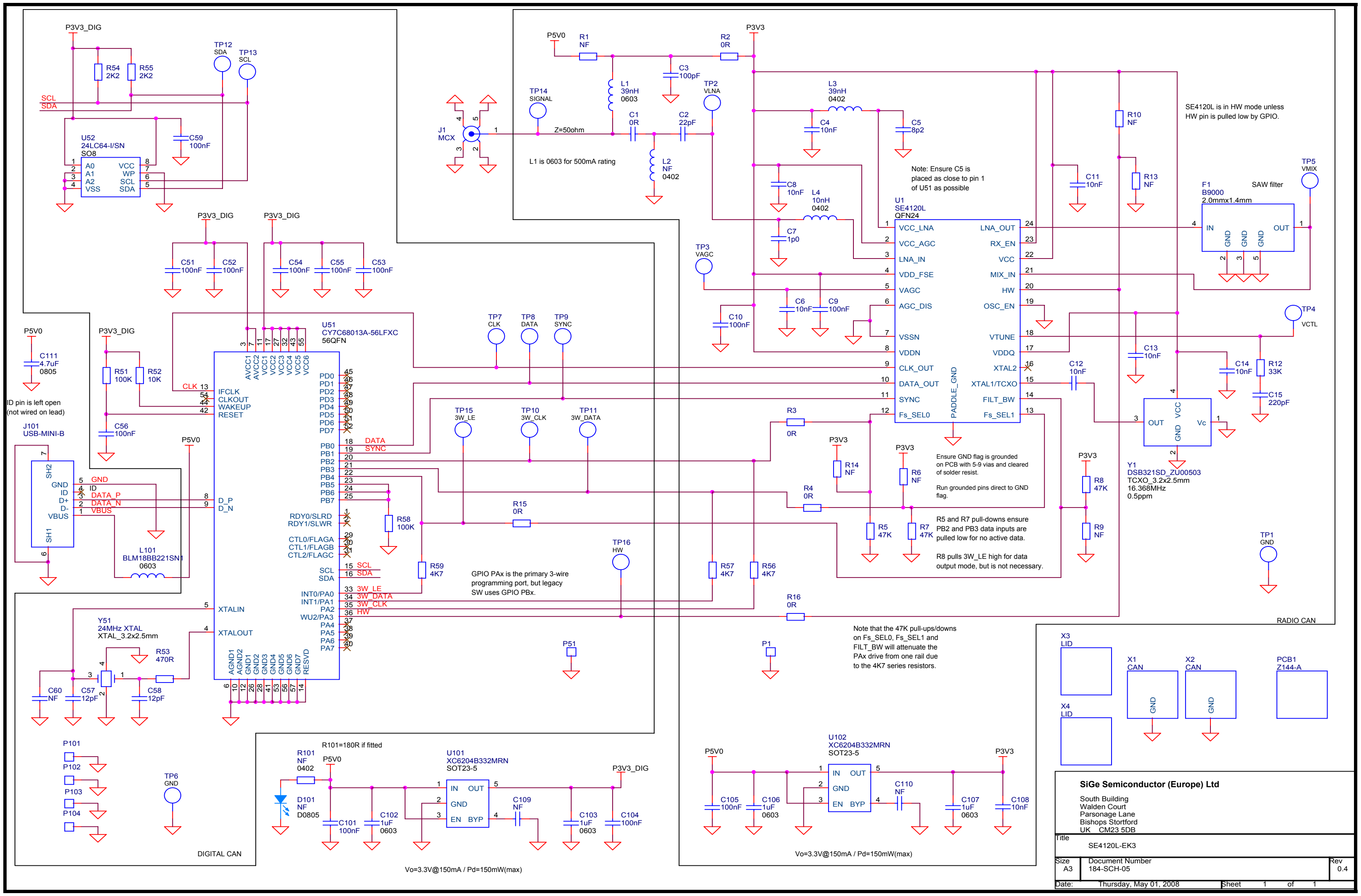
[2] DOC-00052 SE4120L-AK1 Applications Board User Guide, Rev 1.1, Feb-22-2008

[3] 184-DOC-08 SE4120L Programming Guide, Rev 2.0, Mar-27-2007

[4] Cypress CY7C68013A Datasheet, Document #: 38-08032 Rev. *L, February 8, 2008

[5] 'A Software-Defined GPS and Galileo Receiver: A Single Frequency Approach', Kai Borre, Dennis M. Akos, Nicolaj Bertelsen, Peter Rinder, Soren Holdt Jensen, 2007

Appendix A: Schematic



Appendix B: Bill of Materials

SE4120L-EK3 Revised: Thursday, May 01, 2008
184-SCH-05 Revision: 0.4

SiGe Semiconductor (Europe) Ltd
South Building
Walden Court
Parsonage Lane
Bishops Stortford

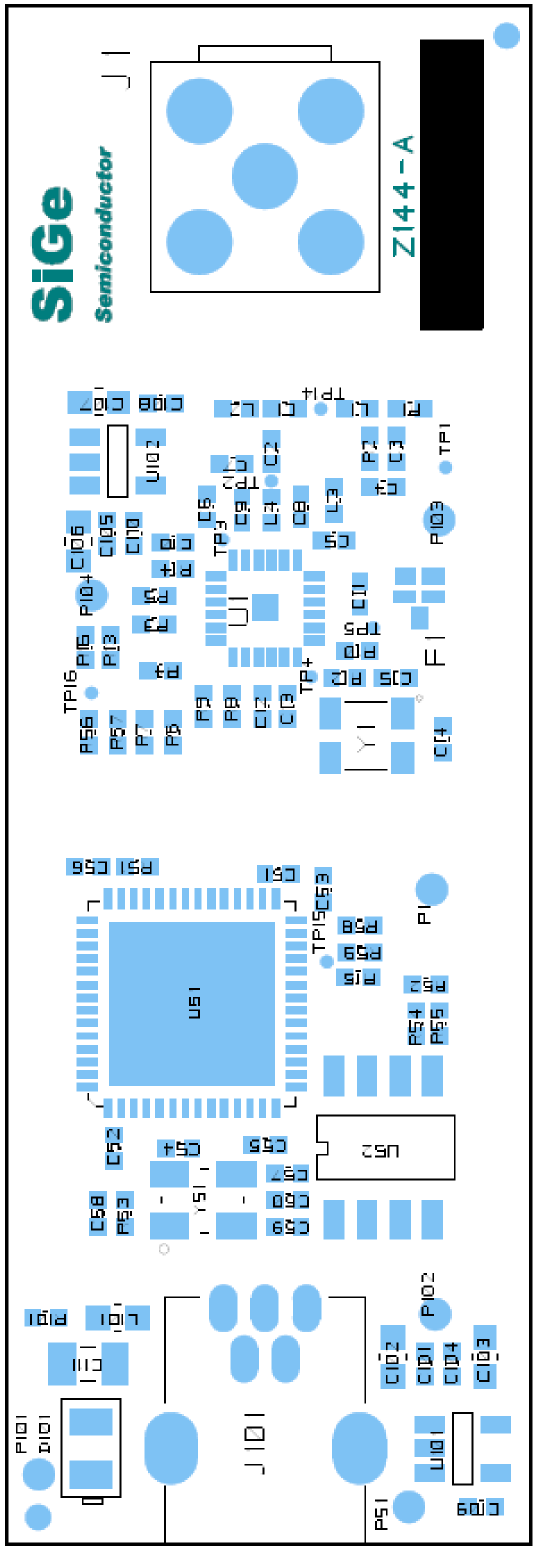
Bill Of I Page1

Item	Quantity	Reference	Value	PCB Footprint	Manufacturer	Part No.	Package	Tolerance	Rating	Material	Distributor	Order Ref
1	6	C1,R2,R3,R4,R15,R16	0R	R0402	Phycomp	232270591001	0402	5%	0.063W	Thick film	Digikey	311-0.0JRCT-ND
2	1	C2	22pF	C0402	Murata	GRM1555C1H220JZ01D	0402	5%	50V	C0G	Digikey	490-1283-1-ND
3	1	C3	100pF	C0402	Murata	GRM1555C1H101JD01D	0402	5%	50V	C0G	Digikey	490-1291-1-ND
4	8	C4,C6,C8,C11,C12,C13,C14, C108	10nF	C0402	Murata	GRM155R71E103KA01D	0402	10%	16V	X7R	Digikey	490-1312-1-ND
5	1	C5	8p2	C0402	Murata	GRM1555C1H8R2DZ01D	0402	+/-0p5	50V	C0G	Digikey	490-3213-1-ND
6	1	C7	1p0	C0402	Murata	GRM1555C1H1R0CZ01D	0402	+/-0p25	50V	C0G	Digikey	490-3199-1-ND
7	12	C9,C10,C51,C52,C53,C54, C55,C56,C59,C101,C104, C105	100nF	C0402	Murata	GRM155R71C104KA88D	0402	10%	16V	X7R	Digikey	490-3261-1-ND
8	1	C15	220pF	C0402	Murata	GRM1555C1H221JA01D	0402	5%	16V	C0G	Digikey	490-1293-1-ND
9	2	C57,C58	12pF	C0402	Murata	GRM1555C1H120JZ01D	0402	5%	50V	C0G	Digikey	490-1279-1-ND
10	3	C60,C109,C110	NF	C0402			0402					
11	4	C102,C103,C106,C107	1uF	C0603	Murata	GRM185R61A105KE36D	0603	10%	10V	X5R	Digikey	490-3893-1-ND
12	1	C111	4.7uF	C0805	Murata	GRM21BR61C475KA88L	0805	10%	16V	X5R	Digikey	490-3338-2-ND
13	1	D101	NF	D0805	OSRAM	LSR976	0805			LED	Newark	05M0542
14	1	F1	B9000	SAW_2x1.4mm	Epcos	B39162-B9000-C710	2.0mmx1.4mm		SAW			
15	1	J1	MCX	MCX_thro_hole	Amphenol	MCX6252B1-3GT30G-50	MCX_thro_hole				Newark	06M5034
16	1	J101	USB-MINI-B	USB-MINI-B	Molex	548190519	USB-MINI-B				Mouser	538-54819-0519
17	1	L1	39nH	L0402	Toko	LL1608-FSL39NJ	0603	5%		Multilayer		
18	1	L2	NF	L0402			0402					
19	1	L3	39nH	L0402	Toko	LL1005-FHL39NJ	0402	5%		Multilayer		
20	1	L4	10nH	L0402	Murata	LQW15AN10NH00	0402	5%		Wirewound	Digikey	490-1146-1-ND
21	1	L101	BLM18BB221SN1	L0603	Murata	BLM18BB221SN1	0603				Mouser	81-BLM18BB221SN1D
22	1	PCB1	Z144-A									
23	6	P1,P51,P101,P102,P103, P104	NF	PIN			PIN					
24	7	R1,R6,R9,R10,R13,R14, R101	NF	R0402			0402					
25	3	R5,R7,R8	47K	R0402	Phycomp	232270570473	0402	5%	0.063W	Thick film	Digikey	311-47KJRCT-ND
26	1	R12	33K	R0402	Phycomp	232270570333	0402	5%	0.063W	Thick film	Digikey	311-33KJRCT-ND
27	2	R51,R58	100K	R0402	Phycomp	232270570104	0402	5%	0.063W	Thick film	Digikey	311-100KJRCT-ND
28	1	R52	10K	R0402	Phycomp	232270570103	0402	5%	0.063W	Thick film	Digikey	311-10KJRCT-ND
29	1	R53	470R	R0402	Phycomp	232270570471	0402	5%	0.063W	Thick film	Digikey	311-470JRCT-ND
30	2	R54,R55	2K2	R0402	Phycomp	232270570222	0402	5%	0.063W	Thick film	Digikey	311-2.2KJRCT-ND
31	3	R56,R57,R59	4K7	R0402	Phycomp	232270570472	0402	5%	0.063W	Thick film	Digikey	311-4.7KJRCT-ND
32	16	TP1,TP2,TP3,TP4,TP5,TP6, TP7,TP8,TP9,TP10,TP11, TP12,TP13,TP14,TP15,TP16	NF	TP_SM			TP_SM					
33	1	U1	SE4120L	QFN24	SiGe	SE4120L	QFN24					
34	1	U51	CY7C68013A-56LFXC	56QFN	Cypress	CY7C68013A-56LFXC	56QFN				Digikey	428-1669-ND
35	1	U52	24LC64-I/SN	SO8	Microchip	24LC64-I/SN	SO8				Farnell	9758070
36	2	U101,U102	XC6204B332MRN	SOT23-5	Torex	XC6204B332MRN	SOT23-5		150mA		Farnell	1106658
37	2	X1,X2	CAN	BMIS-202-F	LairdTech	BMIS-202-F	BMIS-202-F				Mouser	739-BMIS-202-F
38	2	X3,X4	LID		LairdTech	BMIS-202-C					Mouser	739-BMIS-202-C
39	1	Y1	DSB321SD_ZU00503	TCXO_3.2x2.5mm	KDS	DSB321SD_ZU00503	TCXO_3.2x2.5mm	0.5ppm	16.368MHz			
40	1	Y51	24MHz XTAL	XTAL_3.2x2.5mm	NDK	NX3225SA-24.000000MHZ	XTAL_3.2x2.5mm	15ppm	24MHz		Digikey	644-1052-1-ND

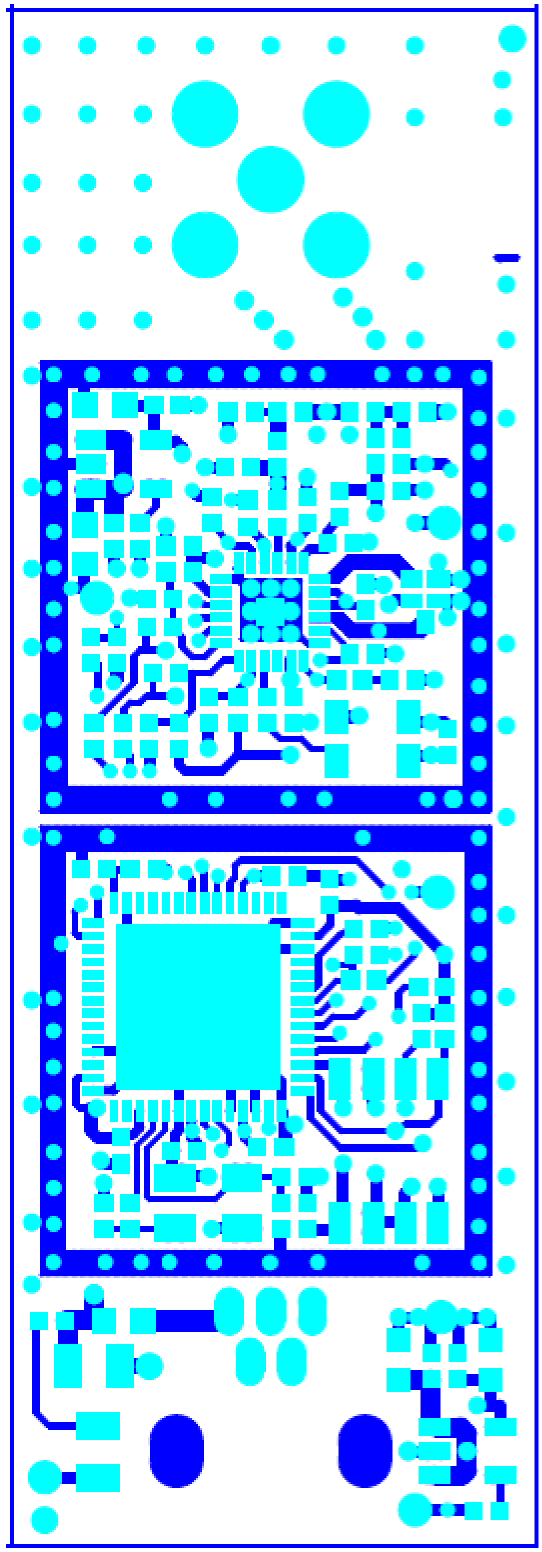
Appendix C: Board Modifications

No	Component reference	Description	Comment
1	L1	39 nH 0603 multilayer inductor fitted in place of original 0402 part	The 0603 inductor is rated at 500 mA and should survive instantaneous maximum currents from the Torex regulator under short-circuit conditions

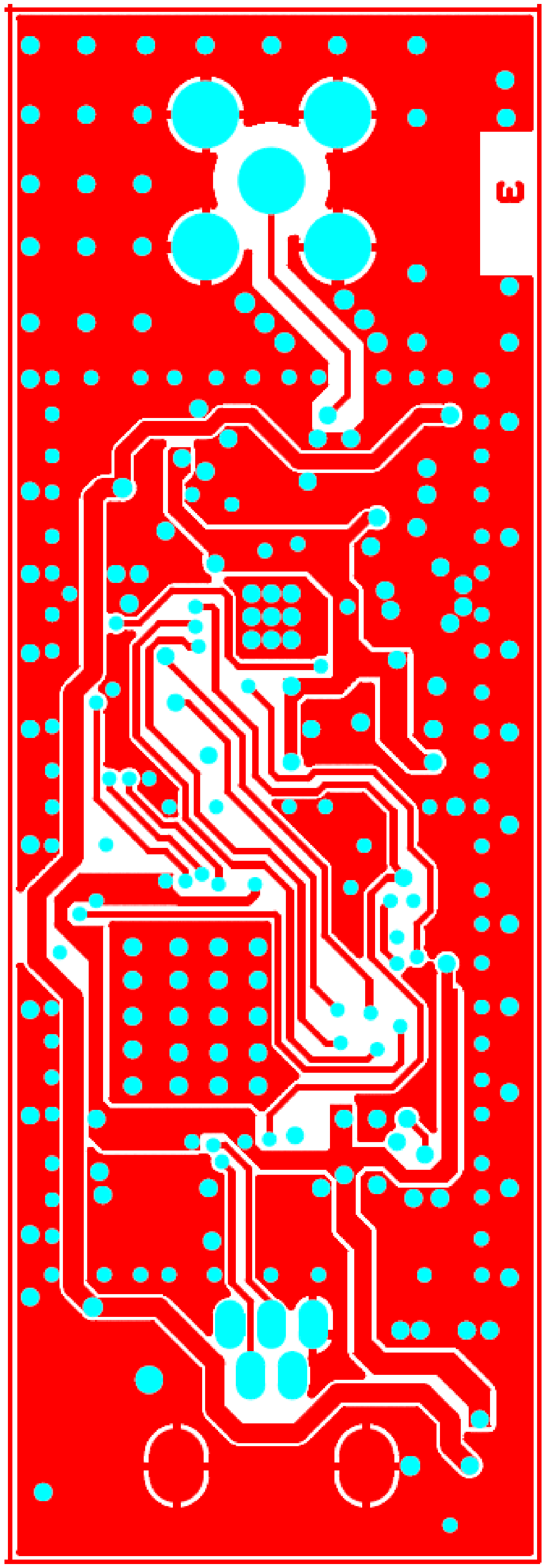
Appendix D: Assembly Drawing (Top)



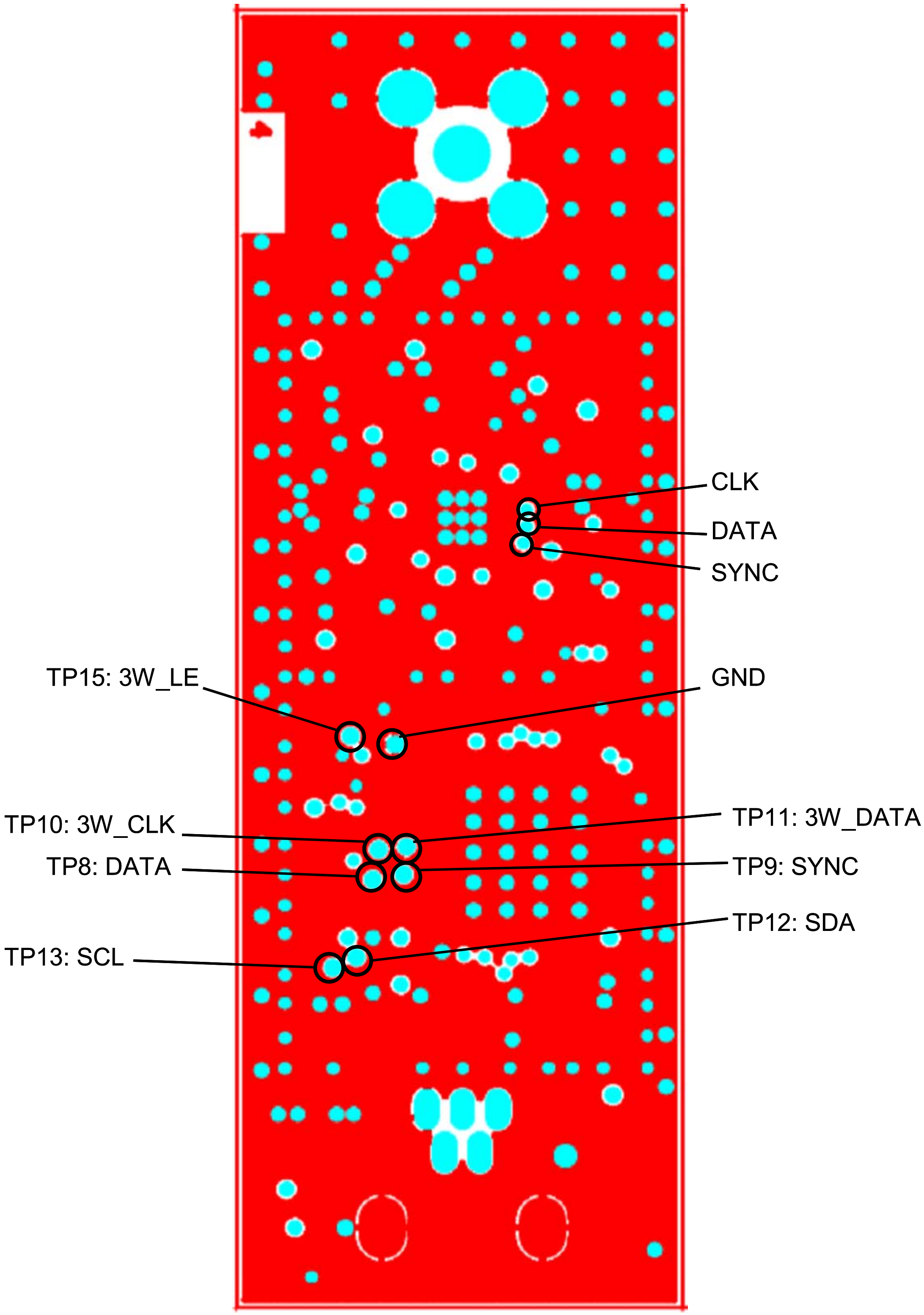
Appendix E: Component Side Tracking



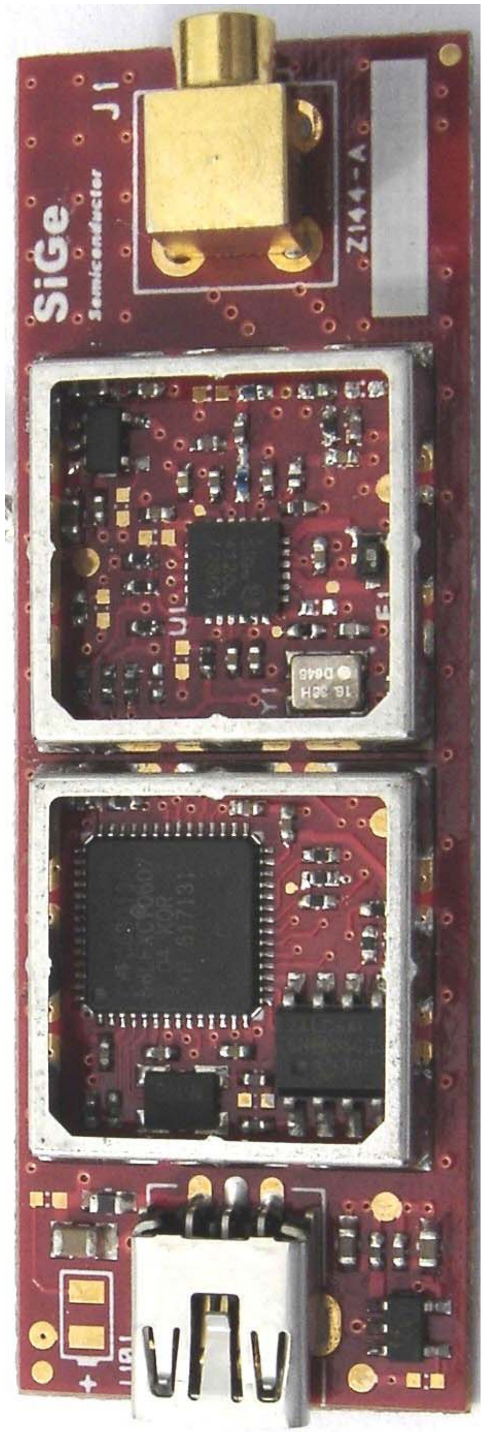
Appendix F: Layer 3 Tracking



Appendix G: Layer 4 Tracking (underside view)



Appendix H: SE4120L-EK3 (lids removed)



END OF DOCUMENT