MODEL A6-CPS

Digital PLL

USER'S HANDBOOK

Quartzlock UK Ltd
Gothic,
Plymouth Road,
Totnes,
Devon
TQ9 5LH,
England.

Tel: +44 (0) 1803 862062
Fax: +44 (0) 1803 867962
Web: www.quartzlock.com
E-mail: quartzlock@quartzlock.com

Contents

1 Safety Considerations

1.1 General

This product and related documentation must be reviewed for familiarisation before operation. If the equipment is used in a manner not specified by the manufacturer, the protection provided by the instrument may be impaired.

1.1.1 Before Applying Power

Verify that the product is set to match the available line voltage and the correct fuse is installed.

1.1.2 Before Cleaning

Disconnect the product from operating power before cleaning.

WARNING

Bodily injury or death may result from failure to heed a warning. Do not proceed beyond a warning until the indicated conditions are fully understood and met.

CAUTION

Damage to equipment, or incorrect measurement data, may result from failure to heed a caution. Do not proceed beyond a caution until the indicated conditions are fully understood and met.

1.1.3 This equipment must be earthed

An uninterruptible safety earth ground must be maintained from the mains power source to the product's ground circuitry.

WARNING

When measuring power line signals, be extremely careful and use a step down isolation transformer whose output is compatible with the input measurement capabilities of this product. The product's front and rear panels are typically at earth ground. Thus, never try to measure AC power line signals without an isolation transformer.

WARNING

Instructions for adjustments when covers are removed and for servicing are for use by service-trained personnel only. To avoid dangerous electrical shock, do not perform such adjustments or servicing unless qualified to do so.

WARNING

Any interruption of the protective grounding conductor (inside or outside the instrument) or disconnecting of the protective earth terminal will cause a potential shock hazard that could result in personal injury. Grounding one conductor of a two conductor out-let is not sufficient protection.

Whenever it is likely that the protection has been impaired, the instrument must be made inoperative and be secured against any unintended operation.

If the instrument is to be energised via an autotransformer (for voltage reduction) make sure the common terminal is connected to the earthed pole terminal (neutral) of the power source.

Instructions for adjustments while the covers are removed and for servicing are for use by service-trained personnel only. To avoid dangerous electrical shock, do not perform such adjustments or servicing unless qualified to do so.

For continued protections against fire, replace the line fuse(s) with fuses of the same current rating and type (for example, normal blow time delay). Do not use repaired fuses of short-circuited fuse holders.

1.2 Voltage, Frequency and Power Characteristics

Voltage 14 – 30V DC (12 – 14V DC no on board OCXO)

Power characteristics 200mA Typical (50mA Typical no on board OCXO)

1.3 Environmental Conditions

1.3.1 Temperature

Operating (ambient)	-10°C to +55°C (-65 to +65 op)
Storage	-40°C to +85°C

1.4 Cleaning Instructions

To ensure long and trouble operation, keep the unit free from dust and use care with liquids around the unit.

Be careful not to spill liquids onto the unit. If the unit does get wet, turn the power off immediately and let the unit dry completely before turning it on again.

Never spray cleaner directly onto the unit or let liquid run into any part of it. Never use harsh or caustic products to clean the unit.

Digital PLL A6-CPS

2 Scope

The Quartzlock A6-CPS is a board level product designed for synchronising two stable oscillators. A typical example is the locking of a low phase noise OCXO to a rubidium frequency standard. The product uses a digital implementation of a standard phase lock loop.

3 Operating Procedure

3.1 Introduction

This module is designed to overcome the disadvantages of narrow bandwidth analogue phase lock loops used to lock relatively stable oscillators together, or to generate arbitrary frequencies from a 10MHz reference with good phase noise, freedom from non-harmonically related spurii, and good short term stability.

When locking a low noise OCXO to a rubidium reference, for example, the ideal PLL bandwidth will be very much less than 1Hz, probably in the region of 10 to 100mHz. An analogue loop will have a very long time constant integrator, leading to thermal drift, capacitor dielectric absorption, and operational amplifier offset drift. In addition, acquisition time of the loop will be very long, and if there is any frequency error, acquisition may not occur at all. There is also a problem of providing an effective in lock indicator to the user, or for use with associated equipment.

The digital loop overcomes all these problems. The long time constant integrator is replaced by a digital integrator that does not drift at all. A combination of an analogue phase detector for low noise and an extended range phase/frequency detector for certain acquisition can be used. The loop bandwidth can be set to maximum for acquisition, followed by glitch free reduction to the working bandwidth when the phase error becomes small. In addition performance measures related to the phase error in the loop, and the frequency error can easily be derived and used to indicate lock and bandwidth control. As an additional benefit a hold over mode that keeps the controlled oscillator tuning voltage constant if there should be a reference failure can be easily provided.

In order to generate arbitrary frequencies from a 10MHz reference, a DDS synthesiser is used. This has 36 bit resolution and is clocked at 10MHz from the reference. Output frequencies of 1.8MHz to 3.6MHz are available as the reference input to the digital PLL. This enables the controlled oscillator (OCXO) to have a frequency range of 1.8MHz to 28.8MHz. The resolution at 10MHz output will be 1.45x10^{-11}.

3.2 Technical details of design

The design uses switching mixer type phase detectors operating at frequencies between 1.8MHz and 10MHz. A dual phase detector is used with quadrature square wave inputs from the controlled oscillator. The main input , which is split between the quadrature phase detectors, is a sine wave input at a level between 0 and 13dBm, and is link selected to either come from the 10MHz reference input, the output of the DDS synthesiser, or the reference divided by 2.

The sine wave signal from the controlled oscillator is converted to a square wave using a fast comparator. It is then divided by 2, 4 or 8 using digital dividers. A link selects direct, 2, 4, or 8 divided signals.

The output from the dividers forms the "Q" reference signal to the Q phase detector. A quadrature "I" reference is generated by passing the Q signal through a programmable delay line, which may be set to delays from 10ns to 137ns, in steps of 0.5ns. This enables quadrature references to be generated for phase detector frequencies between 1.8MHz and 25MHz.

The outputs from the phase detectors are filtered and amplified by DC amplifiers with gain control using digital potentiometers. The gain is controlled by a software AGC system which tries to keep the input to the ADCs at optimum levels. The phase detector outputs are sampled by two channels of the 10bit A to D convertor internal to the PIC 16F689 microcontroller. All other functions of the PLL are carried out by software.

The control of the OCXO or other controlled oscillator uses a precision tuning voltage derived from D to A convertors. Two 16 bit DACs are used, with the output of the fine tune DAC divided by 256 and added to the output of the coarse tune DAC. This gives effectively 24 bit resolution with an overlap between the coarse and fine tune DACs. A software normalisation process ensures that the fine tune DAC is used for tuning most of the time. Only when the controlled oscillator has drifted out of range of the fine tune DAC would the coarse tune DAC need adjusting, with the chance of a very small glitch in the tuning voltage. A precision, low noise, voltage reference is used to supply the DACs.

The microcontroller is provided with an RS232 interface. A simple set of control codes enable monitoring and set up of the digital PLL parameters to accommodate a wide range of controlled oscillators. A Windows front end program will use the control codes to enable the operation of the PLL to be monitored with real time graphs of performance measures.

See Appendix B for Block Diagram.

3.3 Software design

The input to the software is the sampled I and Q signals from the phase detectors. These are sampled at a 1kHz rate. As the final bandwidth of the PLL will be less than 1Hz, this oversampling enables pre-filtering to be used which extends the resolution and reduces noise in the 10bit A to D convertor internal to the microcontroller. Single pole digital filters are used on both the I and Q channels. These are implemented as exponential filters which have a 3dB band width which is a function of the "order" of the filter. Filter orders between 0 (no filter) and 15 are provided. This gives bandwidths between 114Hz for order 1, and 4.8mHz for order 15. The filter order is varied as the user selected PLL bandwidth is varied.

After pre-filtering, the I and Q channels, now at 16 bit resolution, are sub-sampled at a rate between 15.625 s/s, and 1.953 s/s depending on the user bandwidth and lock state of the PLL. The "Q" sample is now divided by the "I" sample (after checking that I>Q) to give a binary fraction. This is used to look up the phase value in a TAN-1 look up table. The look up table is used to synthesise 2 types of phase detector:

a. A phase detector with 16 bit resolution between π/2 and -π/2.

b. A phase/frequency detector with 16 bit resolution between 2π and -2π. This phase detector is equivalent to the well known digital phase/frequency detector. This rolls over between 2π and 0 for positive cycle slips, and between -2π and 0 for negative cycle slips, and will always provide reliable lock if there is a initial frequency error.

The output of the selected phase detector now has digital gain applied, selectable between 1/256 and 128. After digital gain, the phase value is added into the integrator, which is 32 bits wide.

In order to make the loop stable, by providing a phase lead, the phase value has proportional term gain applied, also selectable between 1/256 and 128. This value is added to the upper 3 bytes of the integrator to give the tuning voltage (24 bits)

The tuning voltage is divided between the coarse and fine tune DACs as follows: When normalisation is performed, the fine tune DAC most significant 8 bits are set to mid point (80h). The least significant 8 bits of the fine tune DAC are set to the least significant 8 bits of the tuning word. The coarse tune DAC is then set to provide the final tuning voltage. During all subsequent tuning, only the fine tune DAC is used over its 16 bit range. If the range is exceeded, the normalisation procedure is repeated.

A state machine provides control of locking. After reset the last value of the integrator, which has been stored in EEPROM on a regular basis, is restored. This will retune the controlled oscillator to very nearly the correct frequency. The loop is then opened and the software waits for the following all to occur (state 0):

a. Rubidium reference warm up input to go high.

b. OCXO supply current to drop below a threshold showing the OCXO has warmed up

c. A measure |I|+|Q| which is an approximate measure of the signal level at the phase detector to rise above a threshold.

When these conditions are fulfilled, the software attempts to lock the loop (state 1) by selecting the phase frequency detector, maximum bandwidth, and maximum subsample rate. It then closes the loop and waits for another measure, which is |phaseresult| (delta phi in the block diagram), to drop below a threshold. The measure |phaseresult| is the modulus of each phase calculation filtered in an 8th order exponential filter, the bandwidth of which, for the 15.625 s/s subsample rate, equals 9.7mHz.

Once the lock threshold for |phaseresult| is reached, the lock state (state 2) is entered. The bandwidth is switched to the users selected bandwidth, which has been maintained in EEPROM, and the phase detector is switched over to the narrow band phase detector (π/2 to -π/2). All the time during normal operation, |phaseresult| is being compared to a lower threshold than the lock threshold. If it exceeds this threshold, state 3 is entered which provides a brief flash of the lock LED to warn the user that the selected bandwidth may be too narrow for the PLL to track the drift of the controlled oscillator fast enough. This low threshold is currently set at 480ps maximum phase error (at 10MHz phase detector frequency).

In extreme cases the lock threshold (4.8ns phase error) may be exceeded, in which case the software assumes lock is lost and re-enters state 1.

A further performance measure is calculated, which is available over the interface. This is the first difference of the phase error, filtered in an 8th order exponential filter. It is corrected for subsample rate, and has a constant sensitivity of 5.8x10-15 per bit (at 10MHz phase detector frequency).

This performance measure gives the mean fractional frequency difference between the controlled oscillator and the reference, and is useful for setting up the optimum bandwidth of the PLL.

The band width and damping of the PLL is controlled by 4 parameters, integrator digital gain, proportional digital gain, pre-filter order, and subsample rate. These are preset for 8 values of user selected bandwidth, and can only be changed by modifying the software. It is possible to temporarily adjust the four individual parameters as part of a test procedure carried out over the RS232 interface. The selection of the 4 parameters has been optimised using a mathematical model of the PLL modelled as a MATHCAD spreadsheet. This could be made available to customers who wished to readjust the PLL parameters.

3.4 Interface

The RS232 interface enables setup and monitoring of the DPLL.

The basic control codes are defined in appendix A. The purpose and operation of the control codes are considered in this section.

3.4.1 Basic

All control codes operate on fixed string lengths, there is no terminator character, i.e. "enter" is not required.

All control codes start with two letters (upper case only) which defines the control group. In each group the code may be followed by a "?", a “+” or an additional letter (upper case) with a variable length hexadecimal string. In a few cases only the initial two letters are required, in which case the microcontroller will acknowledge with a single carriage return character (0Dh) after the initial two characters.

The "?" as the third character (query command) will return a hex string representing the current setting of the parameters appropriate to the control group, followed by a carriage return.

The "+" as the third character will enter the query command into the internal repeat stack. The command will then be repeatedly processed at a preset rate. The operation of the command repeat facility is described later.

The control groups which accept an additional letter are intended to update internal parameters, and are described fully under each control group. After updating a parameter, the controller will reply with a carriage return and an automatic repeat of the query form of the control group.

If an invalid code or string is sent that cannot be parsed, the controller will reply with "!" "carriage return", and will clear the input buffer.

3.4.2 Detailed use of interface commands.

WARNING

By incorrect use of the interface, it is possible to get the controller into a state where the PLL will not work. The key command to avoid is "EU" (EPROM update) which writes any changed parameters into the EPROM memory. Provided this is not done, a power on reset will always restore the previous values. An alternative to the power on reset is the command "SR" (software reset).

3.4.3 Command list

UA (user adjust)

This command is intended for the user to change the PLL control bandwidth. The command "UA?" returns a string of 1byte, followed by 2 bytes.

The first byte is the PLL bandwidth, range 0 to 7, with 0 the minimum bandwidth. bits 3 to 7 of the byte are not used.

The second pair of bytes is the power on time of the microcontroller since the firmware was programmed. This is in units of 2.33 hours. This parameter cannot be changed using the interface. (but see "ER" command).

To change the bandwidth, send "UABxx" where xx is the new bandwidth byte.

After changing the bandwidth, the command "EU" must be used to update the EPROM with the new bandwidth.

The actual physical control bandwidth of the PLL will depend upon the tuning sensitivity of the OCXO or other controlled oscillator, also the division ratio between the controlled oscillator and the phase detector. In a later section hints will be given for configuring the board for a new application, which will include aspects of the PLL control parameters.

OS (overall status)

This command monitors and controls the state machine which controls the locking and monitoring of the PLL. It also controls various test modes which are more concerned with product development than user adjustment. However there are two functions which need to be set up using this command for a new application. These are the delay line setting, and the tuning voltage span.

"OS?" returns 8 hex strings of different lengths. The control code definition section has a full specification of these.

Test status byte

The test status byte switches on and off various functions for test purposes. To change a particular bit, a new hex string must be calculated. One of the most important bit switches is bit 7, which when set disables the automatic lock control state machine, and enables the PLL to be manually controlled. In order to open the control loop, set bits 3 and 4.

Bits 0, 1, 2 control the special use of the fine tune DAC. This can be set to output any of the listed internal parameters at a sample rate of 2ks/s. The most useful is the phase result. Display of this parameter on an oscilloscope will show the loop pulling in and locking. When using the fine tune DAC for test purposes, link JP1 should be opened to disconnect the fine tune DAC from the tuning voltage, as otherwise strange results may be observed. The PLL will work quite well with just the coarse tune DAC used for tuning.

IMPORTANT The test status byte is maintained in EPROM. If the command EU (EPROM update) is used when the test status byte has been manually altered, the unit may not work after the next power on reset. If in doubt, the test status byte should be set to 00hex, and the EU command executed.

Lock status byte

The lock status byte monitors the operation of the state machine. Bits 0, 1, 2 indicate the current internal state. Bits 4, 5 are read only bits which indicate the state of the OCXO and PLL. Bits 6, 7 are automatically controlled by the state machine, but may be set and reset by the user for test purposes. In order to avoid overwrite by the state machine; bit 7 of test status should be set.

Bit 6 switches between the narrow range high sensitivity phase detector (range ±π/2, resolution 16 bits) and the wide range low sensitivity phase/frequency detector (range ± 2π, resolution 16bits)

Bit 7 prevents the use of the user preset PLL bandwidth. Every time the PLL algorithm is run, which occurs at the subsample rate, the user bandwidth parameter (0 to 7) is used to look up a table which contains the 4 parameters that control the dynamic behaviour of the loop. These are subsample rate, exponential filter order, integrator gain, and proportional gain. When the loop is unlocked, the state machine sets this bit to inhibit the update. It then loads special parameters which are only used during locking. This gives the loop maximum bandwidth and an under damped response which provides the fastest possible frequency and phase pull in.

PLL control (2 bytes)

These bytes contain the current PLL parameters. These will be refreshed continually from the look up table unless bit 7 of lock status is set. These parameters may be varied by the user; however they are not stored in EPROM memory. The look up table which relates the user bandwidth parameter to the PLL control parameters is stored as a lookup table in program memory, and can only be changed by reloading the firmware.

Quadrature delay line setting (1 byte)

This user adjustable parameter controls the delay setting of the digital delay line that provides the "I" input to the phase detectors. The delay required depends upon the comparison frequency at the phase detectors. For 10MHz comparison, the required delay is 25ns. The delay line has a fixed delay of about 10ns, and a variable delay of about 0.5ns per bit. Thus to obtain 25ns total delay, the setting is 30 decimal or 1Eh. The approximate setting can be calculated from the comparison frequency using the above example. Later in this manual a procedure for accurate setting of the delay will be given

The delay parameter should be stored in EPROM after change, using the "EU" command.

Tune voltage span (1 byte)

The tune voltage span can be set to suit the OCXO. A value of 00h will give a span of 0 to 10V, and a setting of FFh a span of 0 to 5.8V.

After setting the value should be stored in EPROM using the "EU" command.

Q amp AGC setting.

The AGC system tries to optimise the variable gain amplifiers that follow the phase detectors so the input ADCs operate over their full range. As the normal lock point is with the Q channel near the midpoint of the ADC range (2.5V), the I channel actually controls the AGC level when the loop is locked. When unlocked there is a sine wave at both the I and Q channels so both ADCs will control the AGC.

There is little point in the user controlling the amplifier gains manually, although this is possible by writing to these bytes. Bit 5 of test status must be set first.

I amp AGC setting

See previous description

OCXO current (2 bytes).

These bytes are read only and give the filtered value of the OCXO supply current. The range is 0 to 500mA. The digital filter has a 3dB bandwidth of about 5mHz, so the value responds quite slowly to a change in OCXO current. The value is compared to a threshold to detect OCXO warm up. When the current falls below the threshold, the state machine will move to state 1 and try and lock the loop. The OCXO threshold is not adjustable without reloading the firmware.

PL (phase lock loop)

This command monitors the current state of the PLL. It also enables user adjustment of some of the parameters.

Last value of I sample (filtered), 2s complement, 16 bit (2 bytes)

This is the output of the pre-filter (exponential filter) on the I phase detector channel. It is a signed integer in 2s complement format.

Last value of Q sample (filtered), 2s complement 16 bit. (2 bytes)

ditto Q channel

Last value of PLL integrator, 32 bit integer (4 bytes).

Current value of the integrator. This may be overwritten with the PLI command. This is useful for transient testing of the PLL response as it represents a step in the tune voltage.

Coarse tune DAC, 16 bit integer

Fine tune DAC, 16bit integer

Current value of the coarse and fine tune DACs. In normal operation the tuning DACs are updated at the subsample rate. If bits 3 and 4 of the test status byte are set, i.e. there is no integrator update or proportional term, and then the update of the tuning DACs is inhibited. In this situation the DACs may be written to with the PLC and PLF commands.

PD (phase detector)

This command outputs various measures associated with the phase calculation.

Last phase result, 16bit 2s complement, 2 bytes

This useful measure shows the current phase error after the phase calculation. This is after the pre-filtering of the I and Q samples. The phase calculation occurs at the subsample rate.

Last mod(I) + mod(Q) 16bit integer

This is the filtered value of mod(I) +mod(Q). This is the approximate magnitude of the signal after the I and Q ADCs

2.5V reference

This is the filtered result of converting the reference channel of the ADC

mod(phase result) (filtered)

This important measure is the filtered magnitude of the phase error. It is the main measure that the software uses to determine if the loop is in lock. If this measure is greater than a threshold, the lock LED will flash. If the measure increases above a second threshold, the lock state machine will assume that lock has been lost, and will switch to state 0 to try and re-establish lock.

mod(freq offset) (filtered)

This is the rate of change of phase or instantaneous frequency offset, measured each subsample interval. one bit is a fractional frequency offset of 5.82E-15.

DD (DDS synthesiser) 5 bytes

This is the 40 bit tuning word for the DDS synthesiser. Only the least significant 36 bits are used. The tuning word for the physical DDS chip is only 28 bits, so the least significant byte of the tuning word controls a dither algorithm which modulates the physical tuning word by +2 / +1 / 0 / - 2 bits in order to interpolate the frequency and give an effective 36 bit resolution.

The 36 bit tuning word may be selected to any value with the following exceptions:

The DDS chip is serially loaded as two 14 bit words. The dither algorithm only reloads the least significant 14 bits. Therefore if there is any overflow required to the ms 14 bits, then the dither will not work. The value of the least significant 14 bits may be modified by the addition of -1, 0, +1 or +2. This gives forbidden values of the DDS tuning word as: (in binary)

```
0000xxxx  xxxxxxxx  xx000000  00000000  xxxxxxxx
0000xxxx  xxxxxxxx  xx111111  11111111  xxxxxxxx
0000xxxx  xxxxxxxx  xx111111  11111110  xxxxxxxx
```

With a 10MHz reference the DDS tuning word is calculated as follows:

phase detector frequency = $N/2^{36} *10^{7}$

N = (phase detector frequency $*2^{36})/10^{7}$

Example: - For a phase detector frequency of 2.048MHz,

N = 14073748836 (decimal) or 346DC5D64 (hex)

As the ms 4 bits are always zero, the hex string to be loaded with the DDS command is 0346DC5D64

For test purposes, if the ms bit of the tuning word is set to 1, then the dither algorithm is inhibited

The DDS tuning word is maintained in EEPROM and must be stored using the EU command.

EU (EEPROM update)

This command writes all the parameters that are currently in volatile memory to EEPROM memory.

These include:

bandwidth control byte
clock registers
test status byte
quadrature delay setting
tune voltage span
last value of PLL integrator
DDS tuning word

The PLL integrator and the clock registers are automatically written to EEPROM every 2.33 hours providing the PLL is locked

SR (software reset)

This causes a reset equivalent to a power on reset without the need to cycle the power

ER (EEPROM read)

This is a read from any part of the EEPROM, either as ASCII characters or as hex numbers

The command "ERC0000" will dump the entire EEPROM memory to the interface.

This command is usually used by a Windows application which can be used to read and write to the scratchpad part of the EEPROM, where serial number and other production data is stored.

EW (EEPROM write)

This command is usually used by a Windows application for updating the scratchpad part of the EEPROM

RI (repeat interval)

This command is usually used by a Windows application to send continuous data. Any command with the permitted form XX+ may be written to a repeat stack which repeats the command XX? at a rate determined by the RI command. Commands may be mixed in the repeat stack.

The command RID clears all commands in the repeat stack

The command RI0aa where aa is one byte sets the repeat interval in units of 50ms.

4 Configuration of board

In order to configure the A6-CPS board for a new controlled oscillator (whether mounted on or off the board), the following must be considered:

4.1 Basic operating mode (DDS or no DDS)

If the frequency of the controlled oscillator is related to the reference frequency by the following equation, no DDS is required:

Fosc = k/m Freference

where k = 1,2 ,4 or 8 and m = 1 or 2

k is set on LINK2 according to the board ident
m is set to either "10MHz" (m = 1)   or "REF/2" (m = 2) on LINK1

If the DDS is required, LINK1 is set to "DDS"

In this case the controlled oscillator is related to the DDS frequency by the following equation:

Fosc = Fdds * k

Fdds and k should be chosen so the DDS frequency is between 1.8MHz and 3.6MHz

k should be set on LINK2

The DDS tuning work should be programmed as described in section 3.4

The final hardware setup is to select the processor clock source either from the controlled oscillator or from the reference input. The processor clock must be 10MHz. LINK3 sets the clock source (Rb is the reference input CONN1).

The links should be in place on JP1 and JP3.

4.2 Setting the quadrature delay

The quadrature delay is set using software. Power up the board and connect the controlled oscillator, if mounted off the board. Connect a PC with RS232 to CONN4. Only pins 6 (TXD) pin4 (RXD) and pin 2 (ground) need to be connected. Start a general terminal program, set the interface to 9600baud, 1start bit, no parity, and check that the microcontroller responds to commands by sending "OS?"

Open the loop by sending OST98

The controlled oscillator may now be manually tuned by sending PLCdddd where dddd is 2 bytes in order to write to the coarse tune DAC. Set the tuning so the sine wave on TP4 and TP5 is about 1Hz

Connect a dual channel oscilloscope to TP4 and TP5. A slow sine wave should be observed.

The signals on the two channels should have a 90 degrees phase difference

By sending the command OSDdd trim the quadrature delay until the phase difference is 90 degrees

Send OST00 to restore normal operation

Finally send EU to write the new quadrature delay to EEPROM

4.3 Setting the tuning span

This sets the upper limit of the controlled oscillator tuning voltage.

Power up the board and connect the controlled oscillator, if mounted off the board. Connect a PC with RS232 to CONN4. Only pins 6 (TXD) pin4 (RXD) and pin 2 (ground) need to be connected. Start a general terminal program, set the interface to 9600baud, 1start bit, no parity, and check that the microcontroller responds to commands by sending "OS?"

send OST98 to inhibit state control and open the loop
send PLCFFFF
send PLFFFFF

The tuning voltage is now set to a maximum.

Measure the voltage at TP3 and send commands OSSxx until the maximum tuning voltage is equal to the maximum rated tuning voltage of the controlled oscillator

OSSFF will give a voltage of about 5.8 volts and OSS00 a voltage of 10V. The maximum recommended setting is 8V

Finally send OST00 to restore to normal operation and EU to update the EEPROM.

4.4 Changing the PLL control parameters.

The bandwidth and damping of the PLL is controlled using 4 parameters. These are the integrator digital gain, the proportional term digital gain, the subsample rate, and the order (cut off frequency) of the pre-filter.

The control parameters are selected from a look up table in program memory using the user bandwidth control (command UABxx) as the address. The actual control parameters can only be changed by reprogramming the microcontroller.

The parameters have been selected by modelling the PLL using MathCad, and are based on average values of OCXO tuning rates. These can vary between 2 and 15 r/Vs or 0.3 to 3 Hz/V

It is possible to modify the PLL control parameters on a temporary basis.

To do this send: -

OST80	inhibit state control
OSL80	inhibit auto load of PLL control parameters

These commands should be sent when the loop is locked

The new parameters are then sent using the OSG command (2 bytes)

The parameters in use can be queried using OS?

The parameters will affect the PLL performance as follows:

bits 0,1,2,3 subsample rate values 1, 2 4 8 only (1 bit set) give a subsample rate of:

value	subsample rate
1	15.625Hz
2	7.8 Hz
4	3.9Hz
8	1.95Hz

The subsample rate affects the integrator gain

bits 4,5,6,7 exponential filter order values 0 to 15

a value of 0 bypasses the filter

The filters are single pole low pass filters with a 3bD cut off frequency that depends upon the order of the filter, i.e. order 8 is a cut off of 0.6Hz. Order 15 is a cut off of 6mHz.

The exponential filter is selected to have only a small effect on the loop damping.

bits 8,9,10,11 Integrator gain

This parameter, along with the subsample rate, sets the closed loop bandwidth

bits 12,13,14,15 Proportional gain

This parameter controls the loop damping

The dynamic behaviour of the loop can best be verified by inserting a small frequency step while the loop is locked. This can be done by writing a new integrator value "on the fly" while observing the phase with an oscilloscope. The state control should be disabled using the command OST80 to prevent the state control deciding that the loop has unlocked. The new value of the PLL integrator, which should be only slightly different from the current value, can be written using the command PLIxxxxxxxx (4 bytes). The frequency of the controlled oscillator will then shift to a new value, generating a phase ramp which the loop will track out. The phase overshoot will give an indication of the damping factor. At the moment there is no means of simulating a phase step. This could probably be done by adding a small voltage offset to the output of the quadrature phase detector.

5 Specification

REFERENCE INPUT		
FREQUENCY:	10MHz	(DDS is used)
	1MHz to 10MHz	(No DDS)
LEVEL:	100mV PP to 5VPP	(DDS is used, REF/2 is used)
	1VPP to 5VPP	(No DDS)
INPUT IMPEDANCE:	1000 OHMs	
CONTROLLED OSCILLATOR		
FREQUENCY:	1MHz to 40MHz	(DDS not used)
	1.8MHz to 28.8MHz	(DDS is used)
LEVEL (external oscillator)	100mVPP to 5VPP	
INPUT IMPEDANCE	500 Ohms	
EXTERNAL TUNE VOLTAGE	0 to SPAN, where SPAN is software adjustable between 5.8V and 10V	

Notes:

a. If DDS is not used, controlled oscillator must be k/m times higher frequency than reference, where k is link adjusted to 1,2,4,8., and m is link adjusted to 1 or 2. m is reference divider.

b. Either reference or controlled oscillator must be 10MHz to provide microcontroller clock

POWER SUPPLY:	14 to 30V	(on board OCXO is used)
	12 to 30V	(no on board OCXO)
CURRENT CONSUMPTION:	150mA typical	(on board OCXO)
	50mA typical	(no on board OCXO)
PLL BANDWIDTHS	4mHz to 500mHz typical in 8 binary increments	
FREQUENCY PULL IN	up to 7Hz initial frequency error	
LOCK INDICATOR	on	not locked
	off	locked, low phase error
	short flash every second:	locked, high phase error
	long flash, short flash	no processor clock
INTERFACE	9.6kbaud, RS232, PC compatible	
INTERFACE CODES	See Appendix A	

6 Appendix A – RS232 Control Codes

RS232 control codes (all values following command or returned from the microcontroller are hexadecimal)

* = backed up in EEPROM

UA **User adjust**

	UA?	returns user parameters		
		aa bbbb		
	aa	*is bandwidth control: bits set:*	*bit 0,1, 2:*	*bandwidth (0 to 7)*
			bit 3:	*set to inhibit auto load of PLL gain parameters*
*	*bbbb*	*is clock registers 3 and 4 (elapsed time)*		
	UABaa	write new bandwidth control byte		

OS **Overall Status**

	OS?	returns overall status bytes:			
		aa bb cccc dd ee ff gg hhhh			
*	*aa*	*is test status byte:*	*bits set::*	*bit0,1,2:*	*bits 0 to 2 DAC output select*
				bit3:	*no integrator update*
				bit4:	*no proportional term*
				bit5:	*AGC off*
				bit6:	
				bit7:	
		bits 2,1,0:	*000*	*no test output, fine tune DAC used for tuning*	
			001	*sub sampled I*	
			010	*sub sampled Q*	
			011	*PLL Integrator upper 16 bits*	
			100	*Phase result*	
			101	*I sample (filtered)*	
			110	*Q sample (filtered)*	
			111	*reference CH6 (filtered)*	
	bb	*is lock status byte:*	*bits set*	*bit0 to 2:*	*State control, states 0 to 7*
				bit3:	*set to normalise tuning DACs (cleared automatically)*
				bit4:	*OCXO warmed up*
				bit5:	*Loop locked*
				bit6:	*narrow range phase detector in use*
				bit7:	*set to inhibit state control*
	cccc	*is PLL control:*	*bits set*	*bit0,1,2,3*	*subsample rate*
				bit4, 5, 6, 7	*exp filter order*
				bit8, 9, 10, 11	*integrator gain*
				bit12, 13, 14, 15	*proportional gain*
*	*dd*	*is quadrature delay line setting*			
*	*ee*	*is tune voltage span (FFh min,00h max) 0 to 5.8V (FFh), and 0 to 10V (00h):*			
	ff	*is Q amp AGC setting*			
	gg	*is I amp AGC setting*			
	hhhh	*is OCXO current*			
	OSTaa	write new test status byte			
	OSLbb	write new lock status byte			
	OSGcccc	Write new PLL control			
	OSDdd	Write new quadrature setting			
	OSSee	Write new tuning span			
	OSQff	Write new Q amp AGC byte			

OSIgg Write new I amp AGC byte

PL Phase lock loop

PL? returns current status of PLL

aaaa bbbb cccccccc dddd eeee

	aaaa	*last value of I sample(filtered) , 2s complement, 16 bit*
	bbbb	*last value of Q sample(filtered), 2s complement 16 bit*
*	*cccccccc*	*last value of PLL integrator (32 bit integer)*
	dddd	*Coarse tune DAC 16 bit integer*
	eeee	*Fine tune DAC 16 bit integer*

PLIcccccccc write new PLL integrator

PLCdddd write new coarse tune DAC

PLFeeee write new fine tune DAC

PL+ enter command PL? into repeat stack

PD Phase detector

PD? returns phase detector parameters

aaaa bbbb cccc dddd eeee

aaaa	*Last phase result, 2s complement*
bbbb	*Last mod[I] +mod[Q]*
cccc	*2.5V reference (filtered)*
dddd	*mod (phase result) (filtered) lsb=0.763ps*
eeee	*mod(freq offset) (filtered) lsb = 5.82E-15*

PD+ write PD? to command repeat stack

DD DDS synthesizer

DD? returns DDS tuning word

* *aaaaaaaaaa*

DDSaaaaaaaaaa write new DDS tuning word

EU EEPROM update (backed up values)

SR Software Reset

ER EEPROM read

ERCaabb returns bb bytes from starting address aa as ASCII characters

ERNaabb returns bb bytes from starting address aa as hexadecimal numbers (character pairs)

EW EEPROM write

EWCaabbccccc------c

writes bb characters to starting address aa. Correct number of characters must be included in string

EWNaabbcccc------c

Writes bb bytes to starting address aa. Character pairs cc etc are interpreted as hexadecimal numbers.

RI Repeat Interval

RI? returns command repeat interval

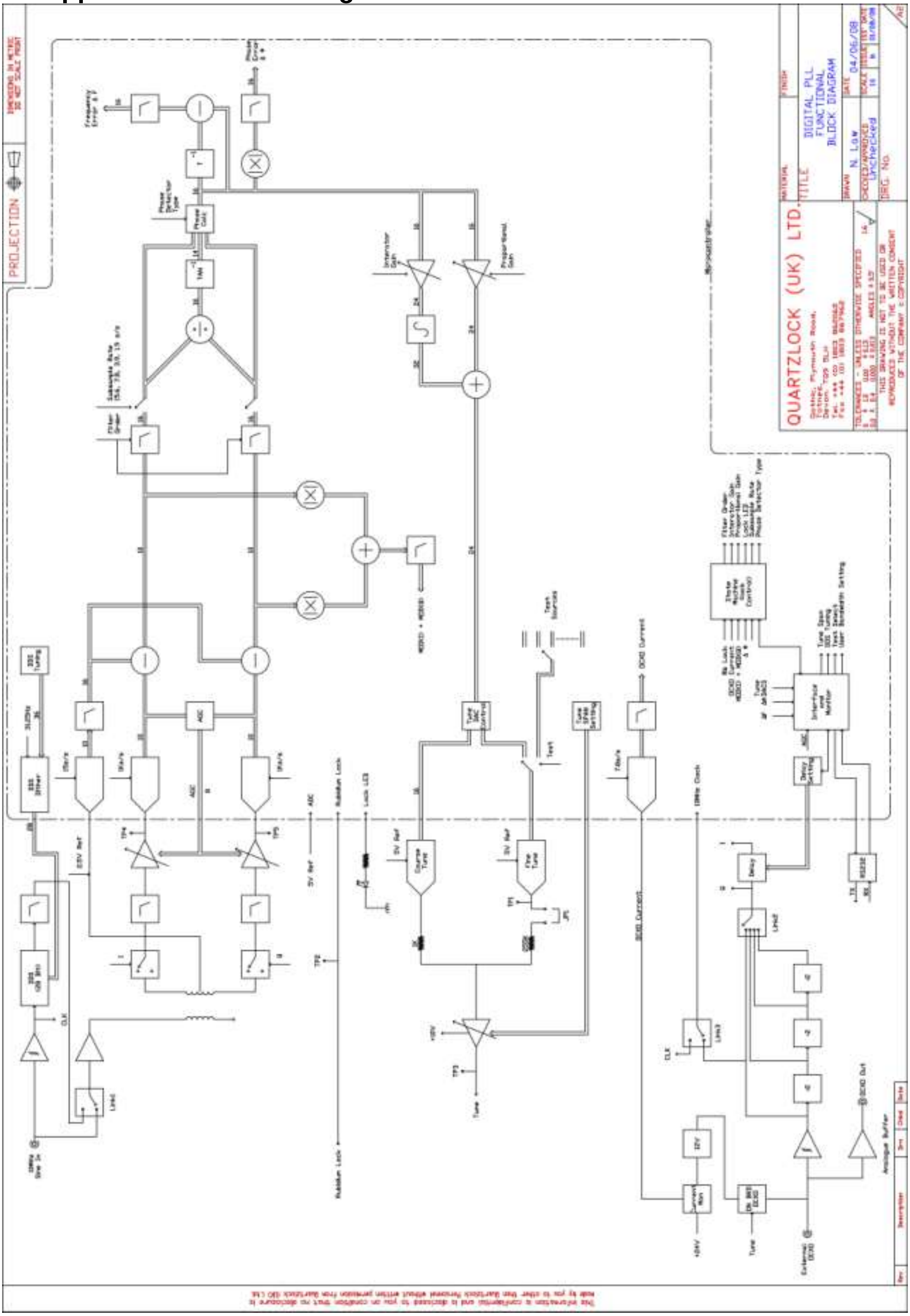
aa

aa *8 bit command repeat interval multiplier. Range 1 to 255. Command repeat interval is 50ms x aa*

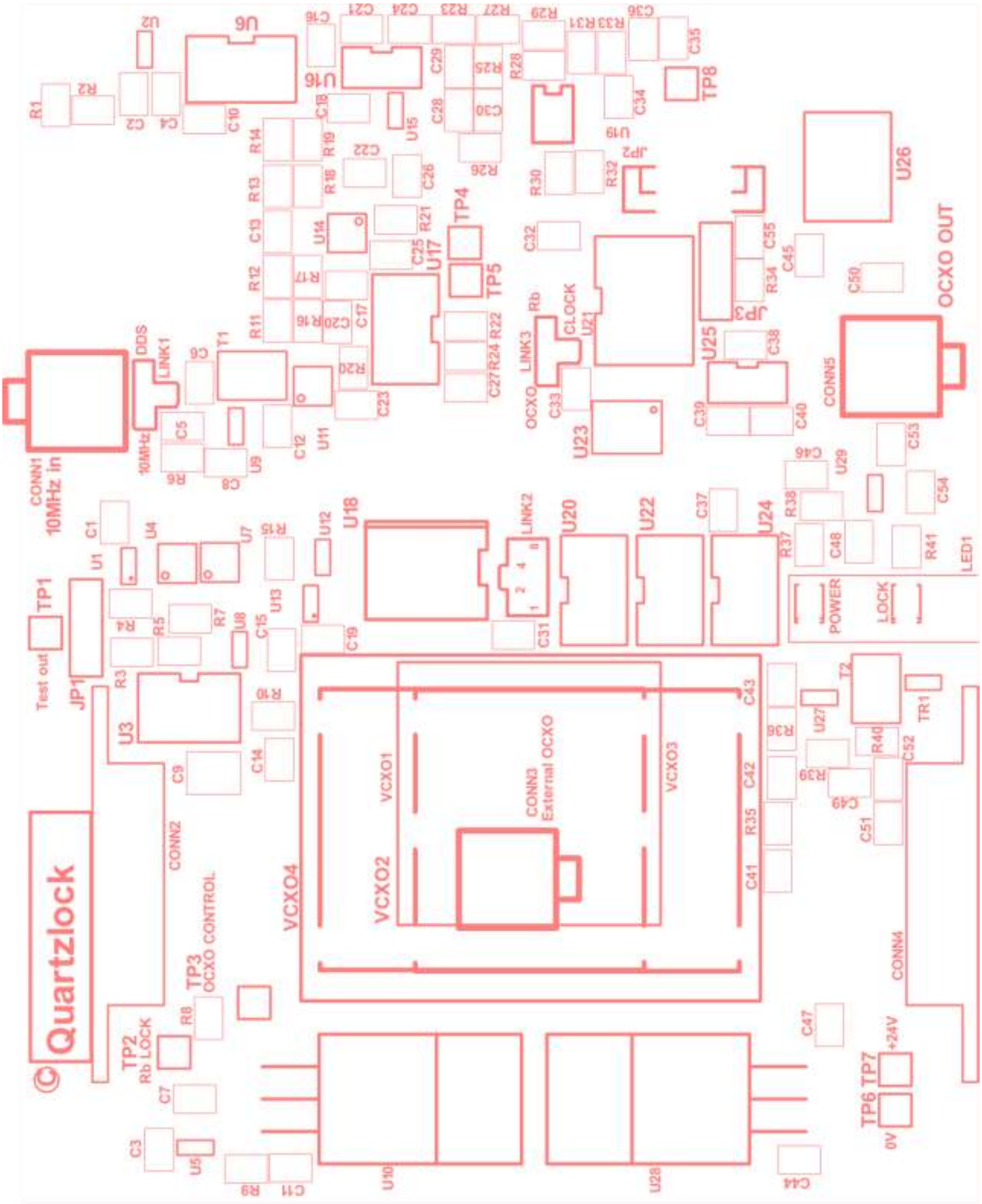
RI0aa write new command repeat interval

RID cancel command repeat and clear command repeat stack

7 Appendix B – Block Diagram



8 Appendix C – Component Ident



9 Appendix D – Connector, Jumper and Link References

Conn 1

10MHz External RF Reference Input

Conn 2

Pin 1 – Reference Oscillator Lock	Pin 6 – Not Connected
Pin 2 – Ground	Pin 7 – Conn 4 Pin 7 (External Reference +24Vdc Supply)
Pin 3 – Ground	Pin 8 – Conn 4 Pin 8 (External Reference VCXO Monitor)
Pin 4 – Not Connected	Pin 9 – Conn 4 Pin 9 (External Reference Lamp Monitor)
Pin 5 – Conn 4 Pin 5 (External Reference Control)	

Conn 3

External OCXO RF Input

Conn 4

Pin 1 – Lock	Pin 6 – RS232 TX
Pin 2 – Ground	Pin 7 – Supply +24Vdc
Pin 3 – Ground	Pin 8 – Conn 2 Pin 8 (External Reference VCXO Monitor)
Pin 4 – RS232 RX	Pin 9 – Conn 2 Pin 9 (External Reference Lamp Monitor)
Pin 5 – Conn 2 Pin 5 (External Reference Control)	

Conn 5

OCXO RF Output

JP1

Normally closed (Opened to disconnect the fine tune DAC from the tuning voltage) see section 3.4.3

JP2

Eeprom programming connection

JP3

Normally closed (Opened to program eeprom)

Link 1

Select either 10MHz Reference Input or DDS see section 4.1

Link 2

Select divider 1, 2, 4 or 8 see section 4.1

Link 3

Select reference clock OCXO or External Reference see section 4.1

TP1

Test Out

TP2

External Reference Oscillator Lock Input (High +5Vdc = Not Locked. Low 0Vdc = Locked)

TP3

OCXO Tuning Control Voltage Output

TP4

Quadrature Oscillator Tuning

TP5

Quadrature Oscillator Tuning

TP6

0Vdc

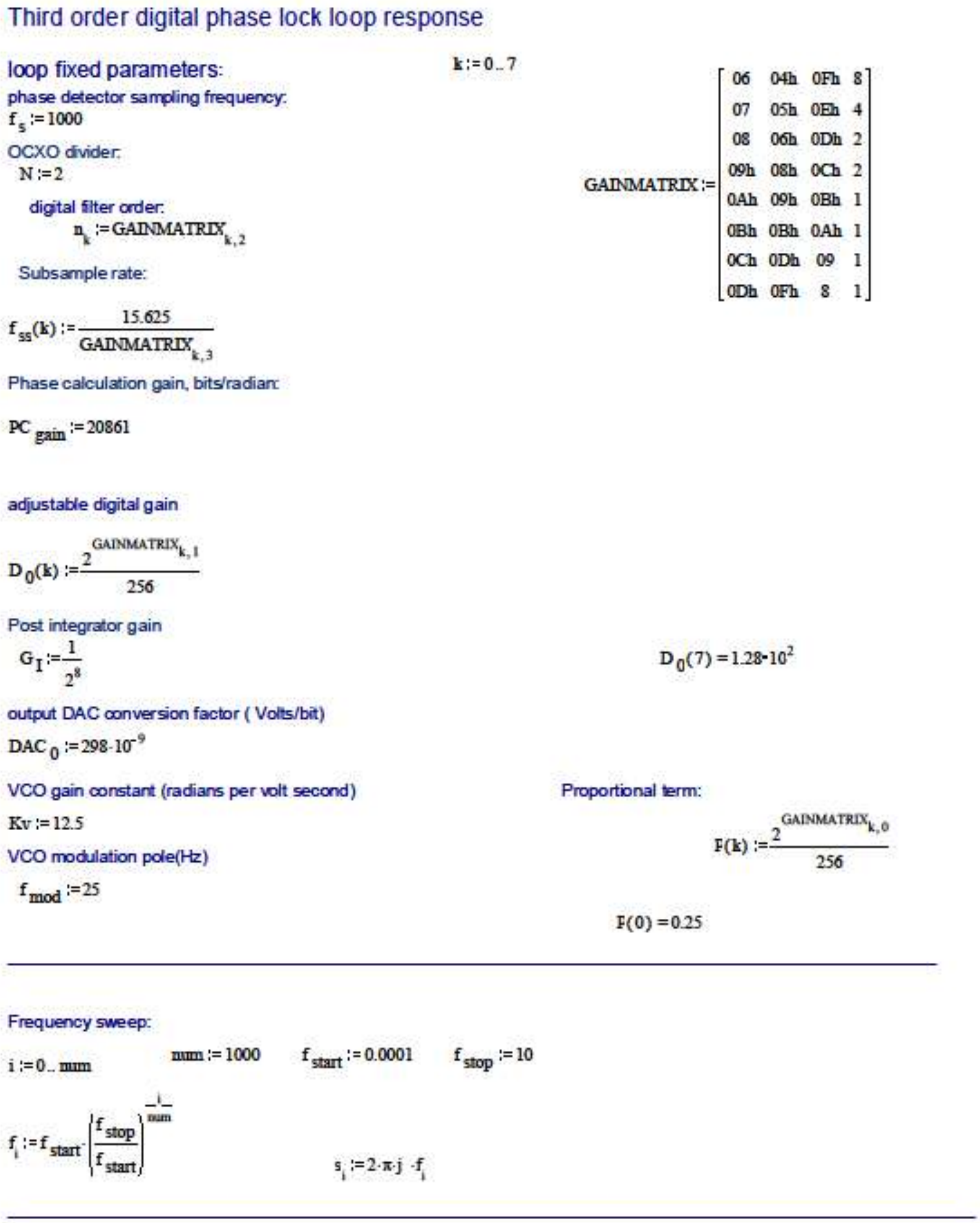
TP7

+24Vdc

TP8

Factory use only

10 Appendix E – Bandwidth Analysis



Third order digital phase lock loop response

loop fixed parameters:

phase detector sampling frequency:

$f_s := 1000$

OCXO divider:

$N := 2$

digital filter order:

$n_k := \text{GAINMATRIX}_{k,2}$

Subsample rate:

$f_{ss}(k) := \dfrac{15.625}{\text{GAINMATRIX}_{k,3}}$

$k := 0..7$

$$\text{GAINMATRIX} := \begin{bmatrix} 06 & 04h & 0Fh & 8 \\ 07 & 05h & 0Eh & 4 \\ 08 & 06h & 0Dh & 2 \\ 09h & 08h & 0Ch & 2 \\ 0Ah & 09h & 0Bh & 1 \\ 0Bh & 0Bh & 0Ah & 1 \\ 0Ch & 0Dh & 09 & 1 \\ 0Dh & 0Fh & 8 & 1 \end{bmatrix}$$

Phase calculation gain, bits/radian:

$PC_{gain} := 20861$

adjustable digital gain

$D_0(k) := \dfrac{2^{\text{GAINMATRIX}_{k,1}}}{256}$

Post integrator gain

$G_I := \dfrac{1}{2^8}$

$D_0(7) = 1.28 \cdot 10^2$

output DAC conversion factor (Volts/bit)

$DAC_0 := 298 \cdot 10^{-9}$

VCO gain constant (radians per volt second)

$Kv := 12.5$

VCO modulation pole(Hz)

$f_{mod} := 25$

Proportional term:

$F(k) := \dfrac{2^{\text{GAINMATRIX}_{k,0}}}{256}$

$F(0) = 0.25$

Frequency sweep:

$i := 0..\text{num}$ $\quad \text{num} := 1000 \quad f_{start} := 0.0001 \quad f_{stop} := 10$

$f_i := f_{start} \cdot \left(\dfrac{f_{stop}}{f_{start}}\right)^{\frac{i}{\text{num}}}$ $\qquad s_i := 2 \cdot \pi \cdot j \cdot f_i$

calculated values

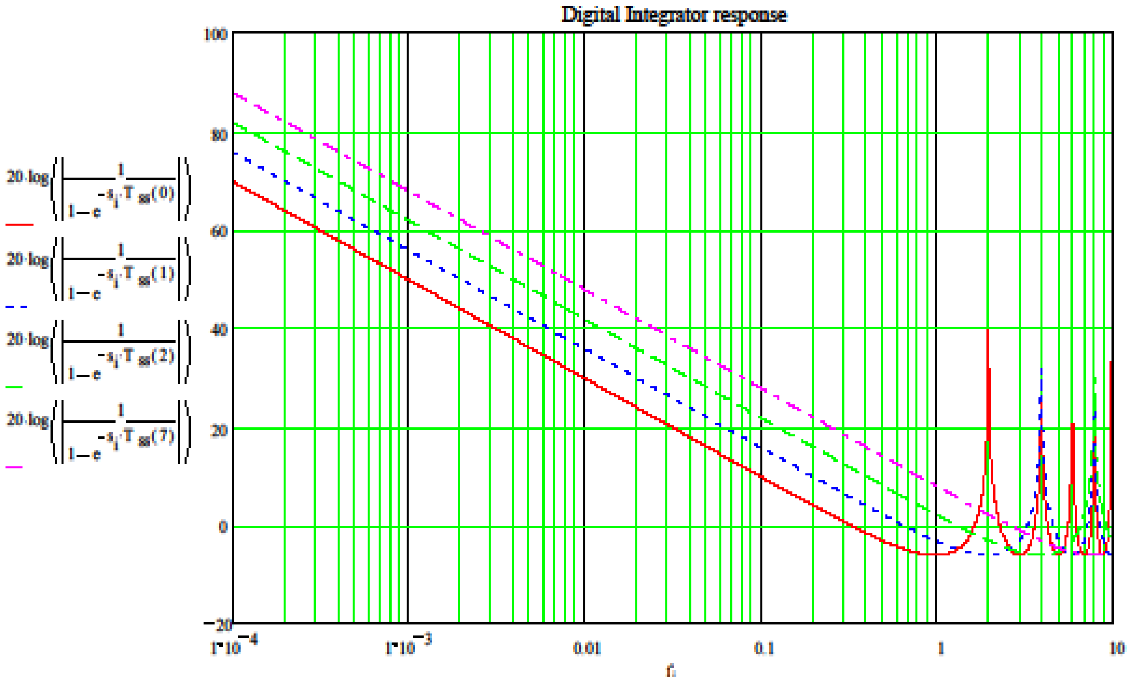
$$T_s := \frac{1}{f_s} \qquad t_{mod} := \frac{1}{2\cdot\pi\cdot f_{mod}} \qquad T_{ss}(k) := \frac{1}{f_{ss}(k)}$$

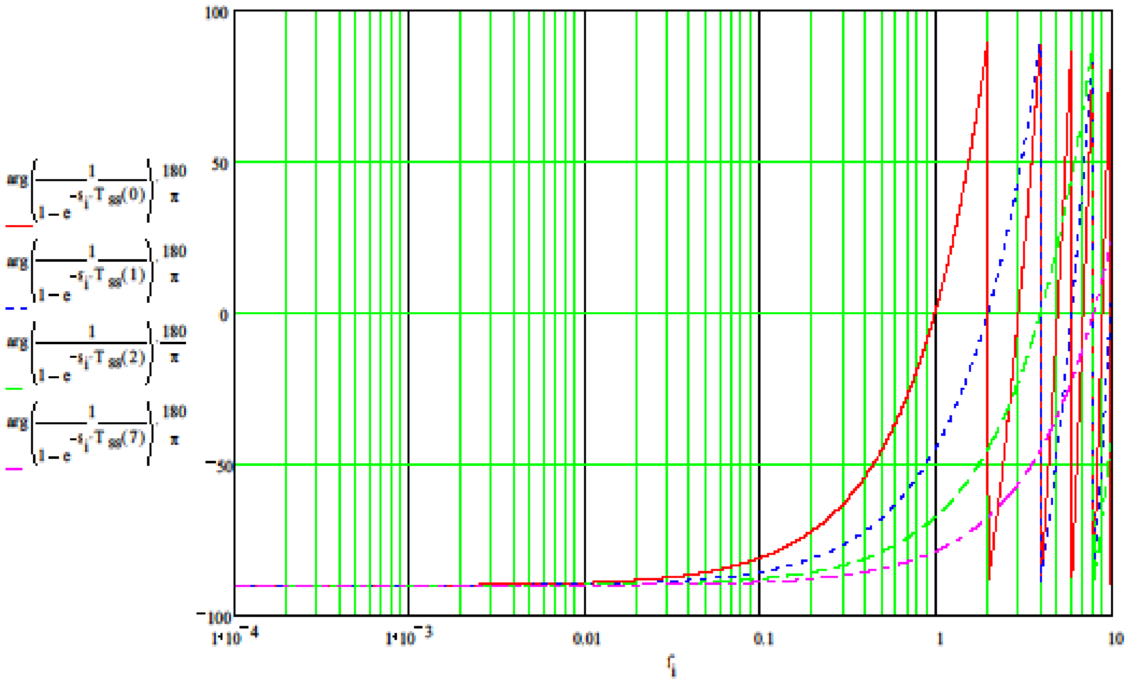
$$F(s,k) := \left[\frac{\frac{1}{2^{n_k}}}{1-e^{-s\cdot T_s}\cdot\left(1-\frac{1}{2^{n_k}}\right)}\right]\cdot\left(D_0(k)\cdot\frac{1}{1-e^{-s\cdot T_{ss}(k)}}\cdot G_I + F(k)\right)\cdot PC_{gain}\cdot DAC_0\cdot\frac{1}{1+s\cdot t_{mod}}$$

$$G_{i,k} := \frac{Kv\cdot F\left(s_i,k\right)}{N\cdot s_i} \qquad H := 1$$

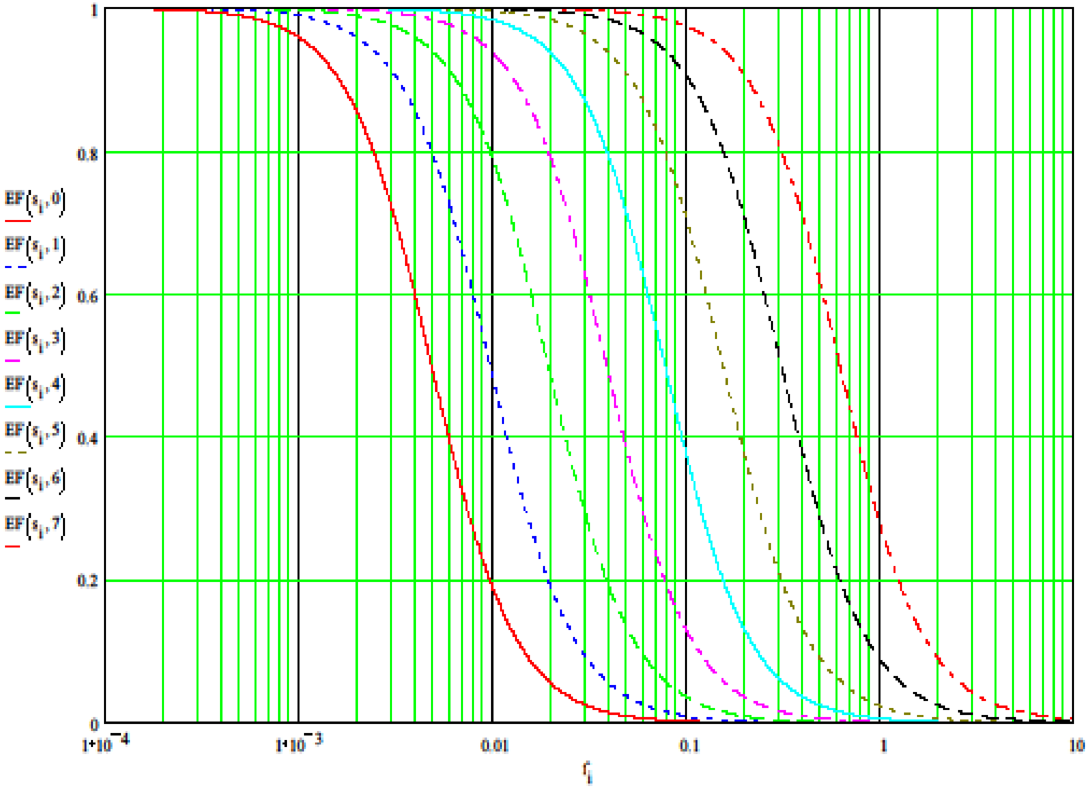
$$A_{i,k} := G_{i,k}\cdot H \qquad B_{i,k} := \frac{G_{i,k}}{1+G_{i,k}\cdot H} \qquad EF(s,k) := \frac{\frac{1}{2^{n_k}}}{1-e^{-s\cdot T_s}\cdot\left(1-\frac{1}{2^{n_k}}\right)}$$

$$T_s = 1\cdot 10^{-3}$$

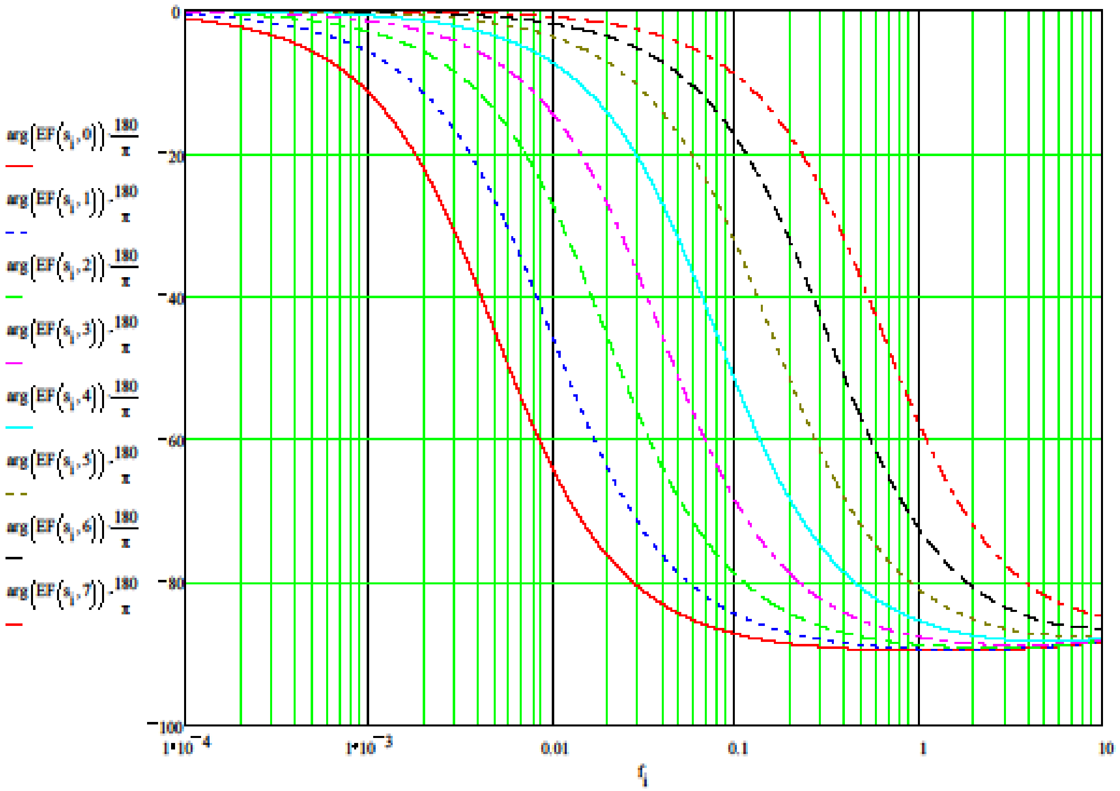




gain of exponential filter

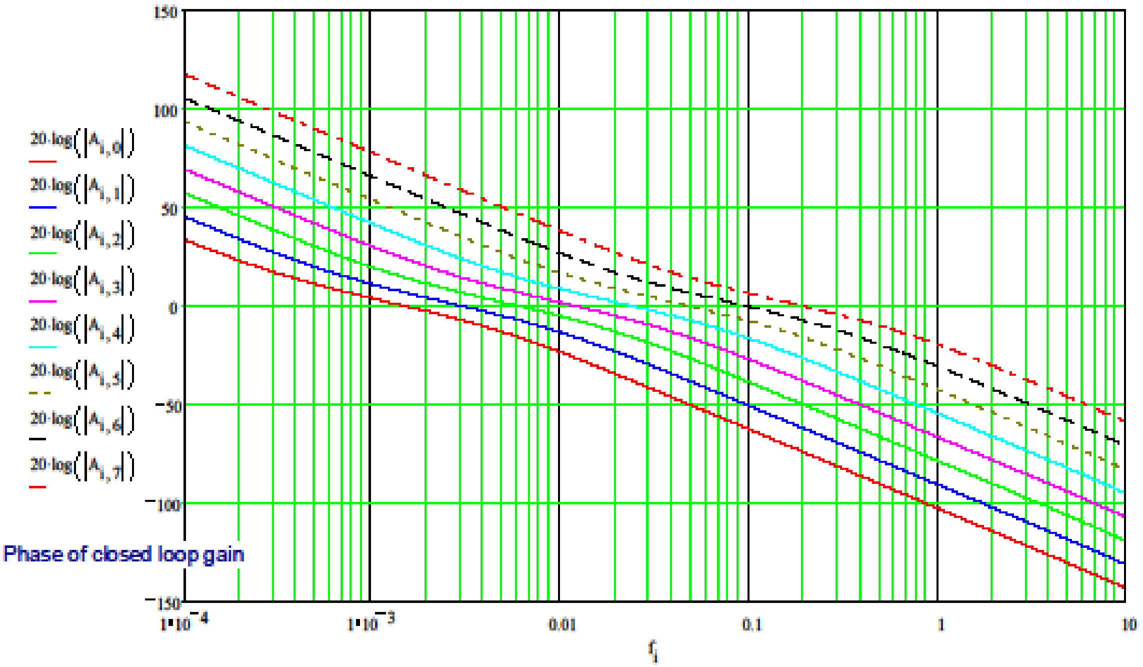


Phase of exponential filter

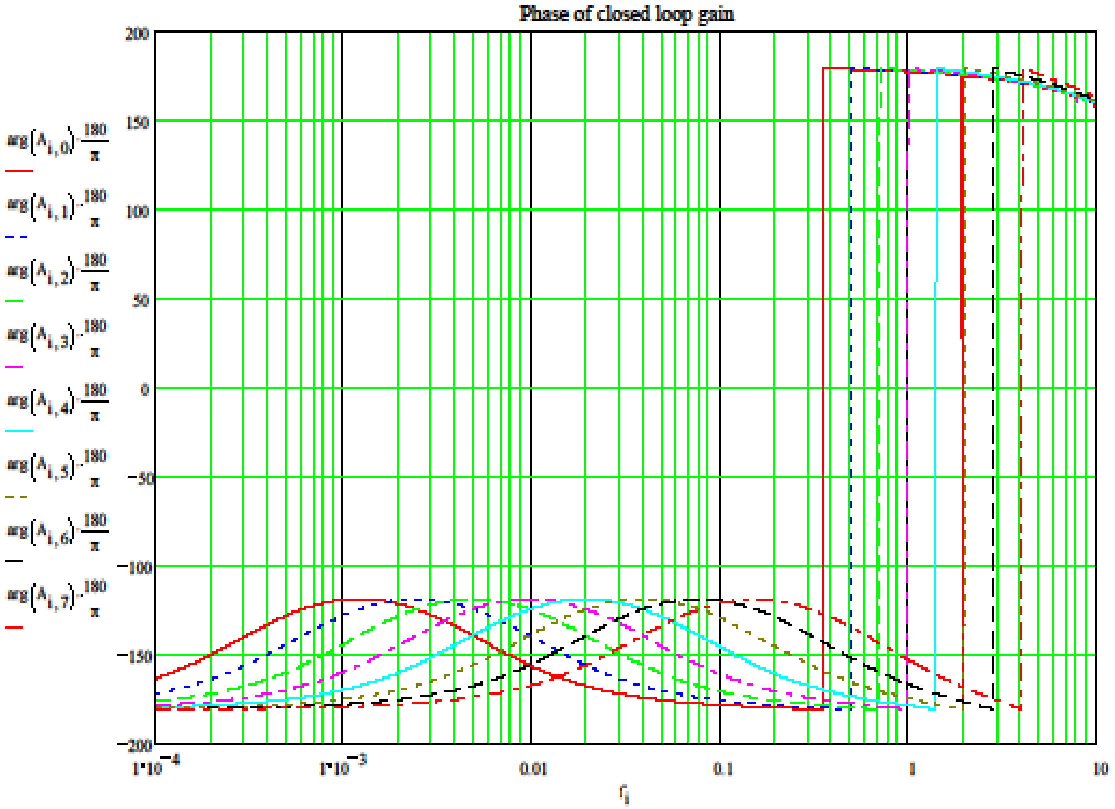


Performance graphs:

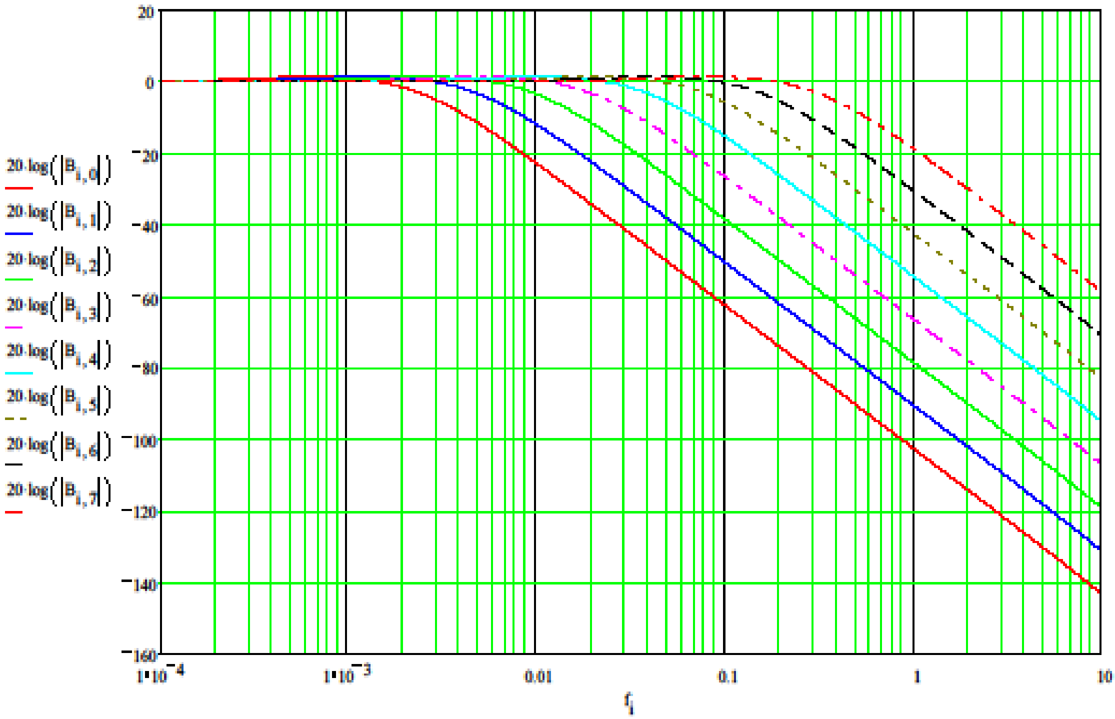
Modulus of closed loop gain



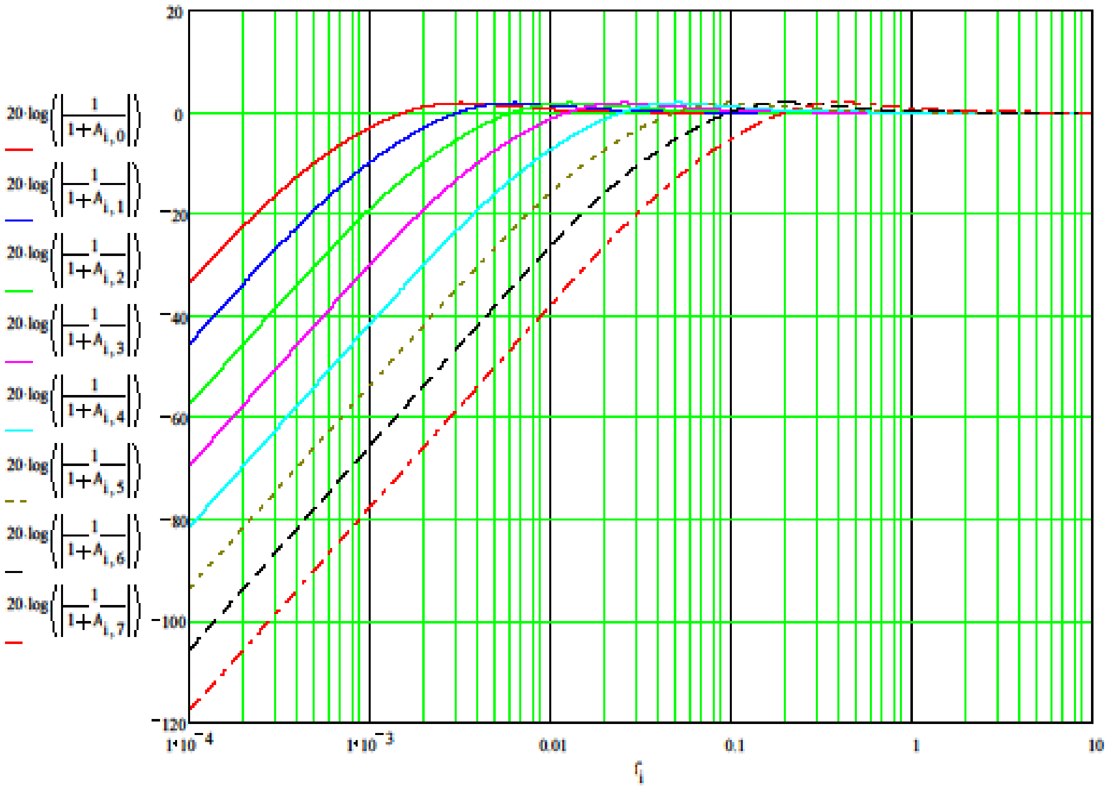
Phase of closed loop gain



Transfer function(output phase to reference phase)



VCO phase noise reduction by loop


$20\cdot\log\left(\left|\frac{1}{1+A_{i,0}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,1}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,2}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,3}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,4}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,5}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,6}}\right|\right)$
$20\cdot\log\left(\left|\frac{1}{1+A_{i,7}}\right|\right)$
20
0
−20
−40
−60
−80
−100
−120
$1\cdot10^{-4}$
$1\cdot10^{-3}$
0.01
0.1
1
10
f_i