EPIC RESEARCH DIAGNOSTICS™	# EPIC ClearView Firmware Design

ORIGINATOR:

Name	Job Title	Signature	Date
Brian Tompson	Engineering Consultant, Atego HighRely	Brian Tompson	5/21/2012

APPROVALS:

Name	Job Title	Signature	Date
Andrew Mason	Chief Technology Officer		5/21/12
Don Pegg	Engineering Consultant, PADT, Inc.		5/21/12
Andrea Miller	Quality Assurance Manager	Andrea Miller	5/21/12

This page intentionally left blank

EPIC RESEARCH DIAGNOSTICS	**EPIC ClearView Firmware Design**

Firmware Design Document

For

EPIC ClearView Device

This page intentionally left blank

Table of Contents

Table of Figures

Table of Tables

1.0 Purpose

The purpose of this document is to define the low level firmware design requirements for the EPIC device component of the ClearView solution. EPIC device is the name of the ClearView component used for the following purposes (not inclusive):

Respond to serial commands from the host
Control the PWM0 pulse duty cycle, frequency and duration
Control the Flood LED

2.0 Scope

The scope consists of the EPIC ClearView device firmware design, no other systems requirements are defined in this document. The initial release of this document addresses only the initial requirements for the firmware; this document will be augmented as new requirements are introduced.

3.0 Definitions

3.1 Terms and acronyms

<CR>	Carriage return character – ASCII 0x0D
<LF>	Linefeed character – ASCII 0x0A
ASCII	American Standard Code for Information Interchange
I/O	Input/Output
ISR	Interrupt service routine
MCU	Micro Controller Unit – Microchip PIC18F25J10-I/SS
PWM	Pulse Width Modulation
PWM0	The digital hardware signal coming from the MCU used to control the boost voltage power supply output pulse
SPI	Serial Peripheral Interface
USART	Universal Serial Asynchronous Receiver Transmitter
USB	Universal Serial Bus

Table 3-1 Terms and acronyms

3.2 Reference documents

Document	Description
EPIC Camera Firmware – Requirements.docx	Firmware requirements
39682E.pdf	PIC18F45J10 Family Data Sheet
ADC101S021.pdf	Single Channel, 50 to 200 ksps, 10-Bit A/D Converter
C18_User_Guide_51288j.pdf	MPLAB® C18 C COMPILER USER'S GUIDE
11195c.pdf	Single/Dual Digital Potentiometer with SPI™ Interface

Table 3-2 Reference documents

3.3 Document conventions

The use of the words **shall**, **should**, **may**, and **will** are used in this document with these specific meaning. The word **shall** denotes a mandatory requirement. The word **should** denotes a recommendation. Before disregarding a **should** statement, the full implementation impact must be considered. The word **may** denotes an optional or recommended implementation. The word **will** denotes a declaration of how another part of the system, outside the scope of this document, must operate. For example, "The MCU **will** be clocked at 8 MHz".

Informative statements are used to provide additional information or background, but do not denote requirements.

3.4 Traceability

Requirements text is numbered in a manner to support automated tracing the firmware design detail to the higher level firmware requirements document. Requirements begin with a uniquely numbered label "FDD_nnnn", and end with the label of the higher level requirement or requirements, enclosed in square brackets '[]'. In addition to the requirements that trace to higher level requirements, derived requirements are added based on best engineering practices for the design of the system.

4.0 System Overview

The EPIC ClearView measures electrical resistance of the skin by putting a fingertip in contact with a glass electrode. A series of electrical impulses are applied to the glass electrode generating a localized electromagnetic field around the finger. Under the influence of this field, and depending on the resistance of the skin of the fingertip, a high frequency current is created resulting in an ionization in the visible and ultraviolet light range. The ionization event is captured by a digital camera.

The ClearView device firmware runs on a Microchip PIC18F25J10 microcontroller (MCU). The firmware accepts serial commands from a host computer, and controls the electrical pulses applied to the glass electrode. The controlling pulse output by the MCU (PWM0) triggers the actual electrical pulse applied to the glass electrode. The duration that the PWM0 is active, is the exposure time for the camera.

The firmware design utilizes the hardware peripherals included in the MCU to control; system timing, communications, boost voltage, PWM0 pulse width frequency and duration, and to monitor the boost voltage level. Firmware execution is controlled by the use of timers and interrupts. The MCU Timer0 is used as the firmware clock and causes the main control loop to execute every 5 mS. The MCU Timer2 is used to control PWM0 pulse width and frequency. Serial communications with the host computer are interrupt driven for data reception and transmission.

4.1 Firmware Main Loop

The firmware main loop is shown in Figure 4-1. Upon power up, the hardware is initialized, and then the application operational parameters are set to their default values. After initialization, the main loop is entered. In the main loop any serial commands from the host are processed, and the exposure is controlled. At the end of the main loop the MCU then enters an idle state, waiting for the 5 mS firmware clock to wake it up.

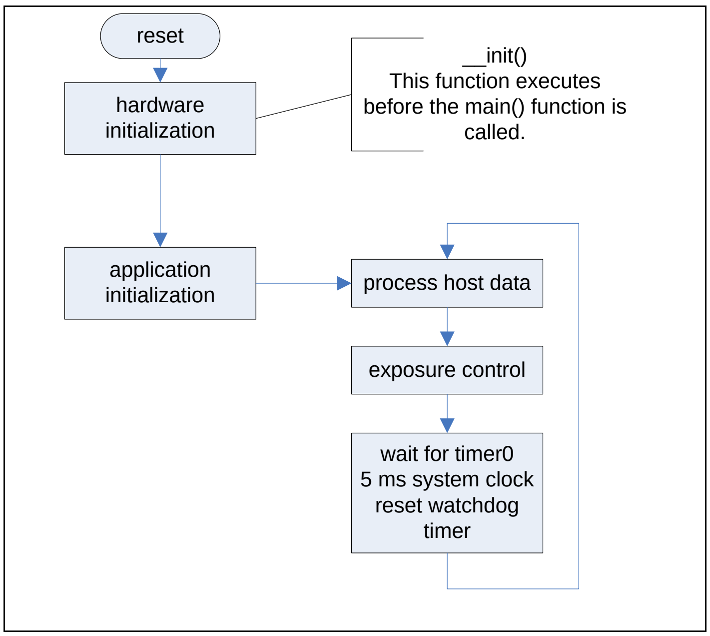

Figure 4-1 Firmware main loop

4.2 Host Communications

Host Communication Protocol Overview

- ASCII, human readable commands, terminated by a carriage return, linefeed combination
- All commands take the form of: <cmd>[=<value>]<CR><LF>
- Command parameters will use standard units; seconds, Hertz, Volts, …
- The device sends responses to the host only in response to commands from the host, except for the powerup information message
- Commands are acknowledged with a pass or fail response
- All responses are terminated with a <CR><LF> combination

The firmware checks for commands from the host and executes them during the execution of the main loop.

The format of host commands consist of printable ASCII characters, terminated with a carriage return and linefeed combination. The first character of the firmware response to the host command indicates success or failure status of the command. After the status character the response may include additional comma separated information.

The host communication routine is shown in **Error! Reference source not found.**. pon entry this routine checks if new data has arrived since the last time the routine was run.

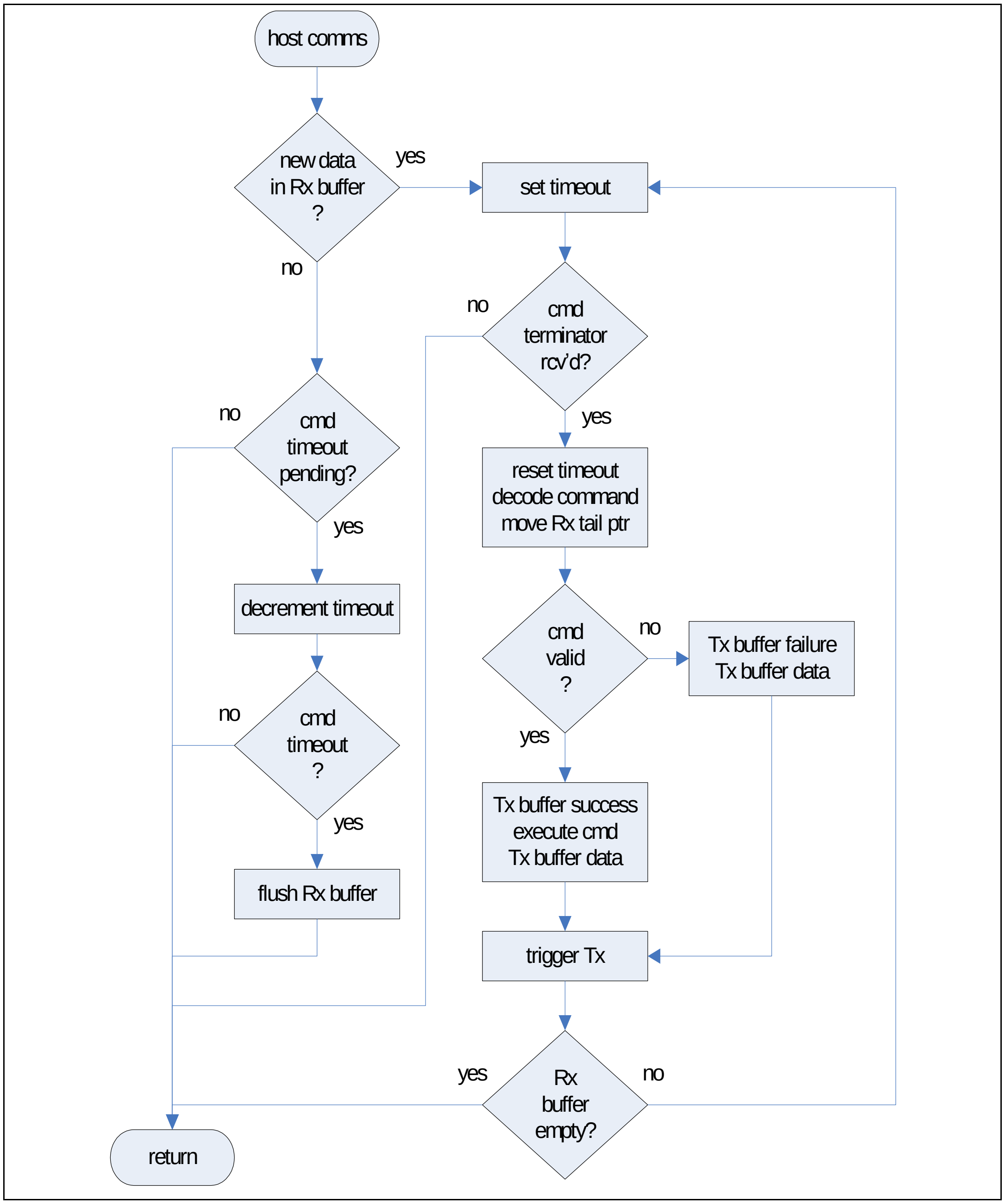

Figure 4-2 Host communication routine

4.2.1 Digital Outputs

The device firmware controls the outputs listed in Table 4-1.

Output	Description
PWM0	Digital pulse output (RC2)
TP1	Heartbeat debugging aid (RA0)
PON	Power on, enable boost voltage (RB2)
LED	Flood LED (RB4)

Table 4-1 Digital outputs

4.3 PWM0 Timing

The actual timing and desired timing may not be equal due to timing constraints of the MCU. Some of the desired and achievable timing values are listed in Table 4-2.

Desired Frequency	PR2 Value	Actual Frequency	Frequency Error
750 Hz	0xA7	748.5 Hz	-0.20%
800 Hz	0x9C	801.3 Hz	0.16%
850 Hz	0x93	850.3 Hz	0.04%
900 Hz	0x8B	899.3 Hz	-0.08%
950 Hz	0x84	947.0 Hz	-0.32%
1000 Hz	0x7D	1000.0 Hz	0.00%
1050 Hz	0x77	1050.4 Hz	0.04%
1100 Hz	0x72	1096.5 Hz	-0.32%
1150 Hz	0x6D	1146.8 Hz	-0.28%
1200 Hz	0x68	1201.9 Hz	0.16%

Table 4-2 PWM0 frequencies

4.4 Exposure Control

This routine checks if the boost voltage is outside the acceptable range and shuts it off if it is. This routine decrements a 5 mS exposure duration counter. The exposure control routine terminates the exposure when the exposure duration is reached.

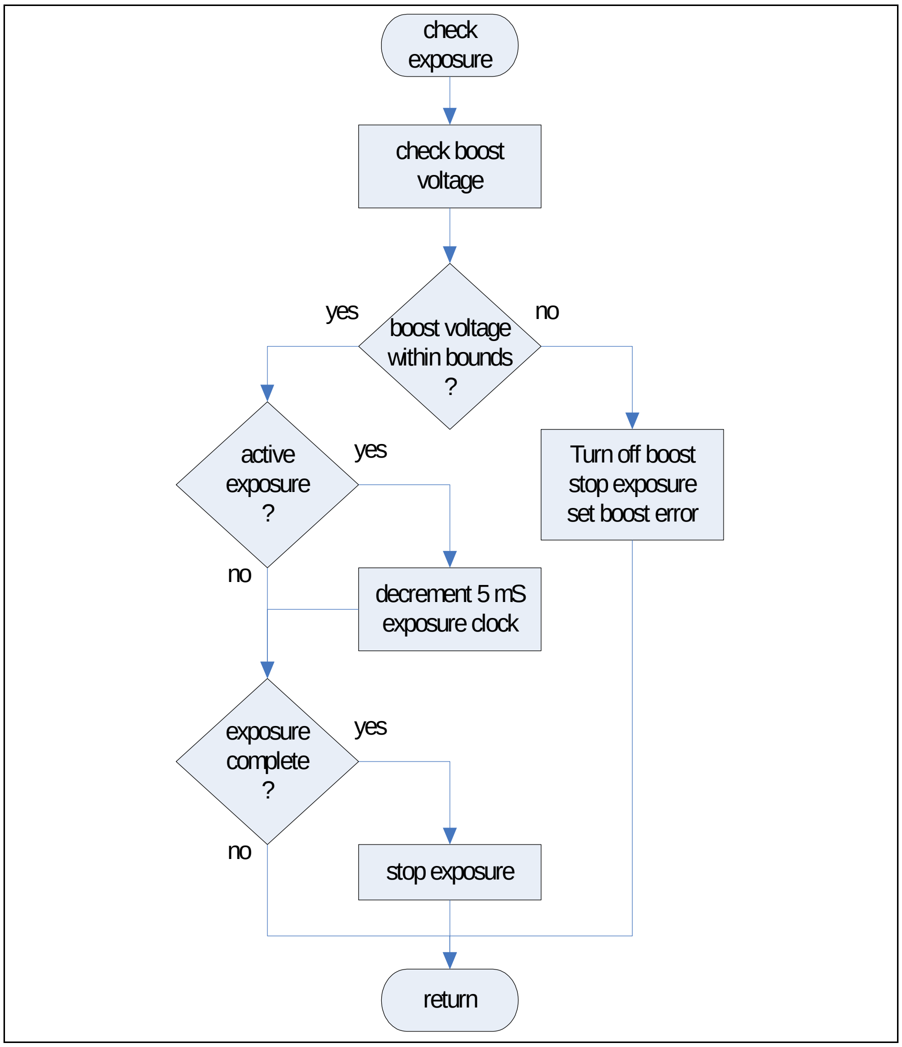

Figure 4-3 Exposure control routine

4.5 SPI Communications

The ClearView device hardware includes 2 SPI modules onboard that communicate using the SPI interface. These modules are a digital potentiometer, and an analog to digital converter. These modules set and read the boost voltage.

4.6 Interrupt Service Routines

The firmware uses 4 Interrupt Service Routines (ISR) to handle; host communications (USART), SPI communications, and the firmware clock. The MCU allows for 2 interrupt priority levels. The host communications receiver ISR and SPI ISR use the high priority interrupt. The other ISR interrupts are low priority. This provides a more deterministic latency between the reception of the exposure start command and the actual application of the PWM0 signal.

The communication interrupt service routines are intended to be light weight and fast. Their purpose is to store received data into a buffer, or transmit data from a buffer. The contents of the buffer are to be handled in the main loop.

4.6.1 **Host Rx ISR**

The host communication receiver ISR places data received from the host in the Rx circular buffer, and increments the circular buffer head pointer. If the buffer is full the new data is discarded and communication receiver overflow error is set. This repeats for every byte received until space is available in the buffer.

4.6.2 **Host Tx ISR**

The host communication transmitter ISR is invoked upon the completion of transmission of a data byte to the host. The host communication transmitter ISR takes the next byte to be transmitted, if any, from the transmit circular buffer and transmits it to the host.

4.6.3 SPI ISR

The SPI communication ISR is invoked when a byte is clocked into or out of the SPI interface. The SPI modules built into the device use 16-bit data. The SPI ISR sends and receives the 2nd byte of data. The SPI communications operate either the digital pot sending SPI data, or the ADC receiving SPI data.

4.6.4 Timer0 ISR

The Timer0 firmware clock interrupt sets a flag causing the main loop to run. That flag is reset at the end of the main loop.

5.0 Firmware Design Requirements

The ClearView device firmware requirements are described in this section.

5.1 Software language

FDD_1001. The code for the device firmware shall be written in the 'C' computer language. [FRD_0001]

The device firmware may include third party software to provide ancillary functionality, such as a program bootloader. Some of this software may be written in assembly language.

5.2 Fault Recovery

FDD_1002. The device firmware shall utilize the watchdog timer feature of the MCU to reset the device firmware if an unrecoverable fault occurs. [FRD_0013]

The watchdog timer will reset the MCU if the timer is not reset within 32 mS. The 32 mS watchdog timeout was selected to ensure that the power up initialization would complete before the watchdog needed to be reset.

5.3 Resource Usage

The firmware should utilize less than or equal to 70% of the available program memory (22.9 KBytes) and less than or equal to 70% of RAM (716 Bytes). The longest execution of the main loop should be less than or equal to 70% of the 5 mS firmware clock (3.5 mS).

5.4 Deleted

5.5 System Constants

Symbol	Value	Notes
FOSC	8000000	MCU oscillator frequency
FIRMWARE_TICK	5.0E-3	Firmware clock tick
FIRMWARE_PRESCALAR	2	TMR0 firmware clock Prescalar
FIRMWARE_CLOCK	0xECEC	0xFFFF – (FIRMWARE_TICK\ /(FIRMWARE_PRESCALAR\ *FOSC*4))) This value includes TMR0 overhead of 117 uS

Table 5-1 System constants

5.6 Hardware Initialization

The initialization of the hardware is described here.

After coming out of reset all MCU I/O ports are configured as inputs, and all interrupts and timers are disabled.

FDD_2002_Derived. The MCU shall use the configuration values in Table 5-2.

Parameter	Default Value	Description
DEBUG	OFF	In-circuit debug
XINST	OFF	Instruction set extension
STVREN	ON	Reset on stack overflow
WDTEN	ON	Watchdog timer enable
CP0	OFF	Code protection
FCMEN	OFF	Fail-Safe clock monitor
IESO	OFF	Two-speed start-up
FOSC2	ON	Default system clock
FOSC	HS	HS oscillator
WDTPS	8	Watchdog timer 32 ms
CCP2MX	DEFAULT	CCP2 MUX

Table 5-2 MCU configuration values

FDD_2003_Derived. After reset and before control is transferred to the main function, the firmware shall initialize the MCU hardware.

FDD_2004_Derived. The device firmware shall configure unused MCU I/O pins as outputs.

FDD_2005_Derived. The device firmware shall operate a single SPI interface on the MCU.

Register	Value	Description
TRISA	0x00	all outputs
TRISB	0x00	all outputs
TRISC	0x90	Rx & SDI inputs, remainder outputs

Table 5-3 Data direction control registers

FDD_2006_Derived. The device firmware shall drive unused outputs to a logic low state during hardware initialization.

5.7 Application Initialization

When control passes from the 'C' startup code to the main function, data used by the application is initialized.

5.7.1 Reset status

During initialization the firmware records the conditions that caused the last reset, and confirm that the boost voltage is off.

FDD_1003. The device software shall record the condition flags that indicate the cause of the last reset. [FRD_0016]

FDD_2007_Derived. The device firmware shall clear all the reset condition flags during application initialization.

FDD_2012_Derived. After reset the firmware shall read the SPI ADC to confirm that the boost voltage is turned off.

5.7.2 Host communications

FDD_1004. The device firmware shall communicate with the host using the parameters in Table 5-4. [FRD_0014]

Value	Parameter
Baud rate	9600 bps
Data bits	8
Parity	none
Stop bits	1

Table 5-4 Communications parameters

5.7.3 SPI communications

FDD_1005. The device firmware shall operate the SPI interface at a baud rate of 2 Mbps. [FRD_0005, FRD_0006]

5.7.4 Firmware clock

The main firmware loop operates every 5 mS. At the end of the loop, the MCU is placed in the idle state, halting code execution, but allowing the MCU hardware peripherals and timers to continue to run.

FDD_1006. The device firmware shall use the MCU Timer0 to generate a firmware clock interrupt at 5 mS intervals. [FRD_0003, FRD_0012]

FDD_1007. The firmware clock interrupt shall set a flag causing the main program loop to run. [FRD_0003, FRD_0012]

Register	Value	Description
T0CON	0b00000010	TMR0 on, 8-bit, internal clk, prescalar 2
TMR0L	FIRMWARE_CLOCK	TMR0 value for firmware clock
TMR0IE	1	TMR0 interrupt enable
TMR0IP	0	TMR0 interrupt low priority

Table 5-5 Timer0 register values

5.7.5 PWM0 timing

FDD_1008. The default PWM0 timing parameters shall be set as shown in Table 5-6. [FRD_0004]

Parameter	Default Value
PWM0 pulse width	14 uS
PWM0 frequency	1,100 Hz
PWM0 duration (exposure duration)	0.5 seconds

Table 5-6 PWM0 default values

5.7.6 Application variables

FDD_1009. The firmware shall initialize application parameters to the values shown in Table 5-7. [FRD_0005]

Parameter	Default Value
TP1_OUTPUT	on
SPI_ADC_CS	disable
SPI_POT_CS	disable
clearView_state	STARTUP_STATE
IDLEN	1 (idle sleep mode)
TMR0IE	enable
TMR0ON	1
PWM0 Frequency	1100 Hz
PWM0 Width	14 uS
clearView_exposure	0.5 seconds
Boost voltage	100%

Table 5-7 Application default values

5.7.7 Enable Interrupts

After the application initializations are complete, the MCU interrupts are enabled allowing system operation.

- enable interrupt priorities
- enable low priority interrupts
- enable high priority interrupts

5.7.8 Initialize MCU PWM

Run the MCU PWM during initialization before the PWM0 pin is configured as an output. To ensure the PWM pulses are consistent, the MCU PWM peripheral is started during initialization. The normal exposure control routine turns off PWM0 when it runs.

FDD_2019_Derived. The firmware shall start the MCU PWM with the PWM0 pin configured as an input to initialize the MCU PWM.

5.8 Program Control

This section describes the overall flow of the program control.

After initialization the firmware enters the main program loop. In the main loop host communications are processed, and the exposure is controlled. Communications from and to the host are implemented through the use of interrupts, and are not synchronous with the main program loop. Once a properly formatted command is received from the host, the command is executed during the next pass of the main program loop.

The main loop operates in one of five states; startup, powering up, ready, exposure active, and fatal error. In the startup state, the device checks that the boost voltage is off before turning it on (below 53 VDC). In the powering up state, the device checks for the boost voltage to be within operational limits (between 53 VDC and 200 VDC). In the ready state the device is prepared for any command from the host. In the exposure active state, the device only accepts an abort command from the host. In the fatal error state, all commands from the host are rejected. Only a device reset or a special diagnostic command can cause the device to exit the fatal error state.

FDD_1010. The device firmware shall execute the instructions in the main program loop at the rate of the firmware clock. [FRD_0003, FRD_0012]

FDD_1011. The device firmware shall initiate the MCU idle mode at the end of the main loop. [FRD_0003, FRD_0012]

FDD_1012. The device firmware shall return to the MCU idle mode when awakened until the firmware clock indicates it is time to run the main program loop. [FRD_0003, FRD_0012]

5.9 Host Communications

The EPIC ClearView device is controlled via USB commands sent by the ClearView host application. The ClearView device hardware will convert the host USB communications to the serial UART communications used by the MCU.

FDD_1013. The device firmware communications with the host shall be full duplex. [FRD_0014]

FDD_1014. In normal operation mode, the device firmware shall silently discard host messages as incomplete if time to receive the complete message exceeds 500 mS. [FRD_0015]

FDD_1015. In normal operation mode the device firmware shall only transmit to the host in response to a command from the host, except for the powerup information message. [FRD_0024]

FDD_1016. In normal operation mode the device firmware shall not echo characters received from the host back to the host. [FRD_0015, FRD_0024]

5.10 Host Commands

The ClearView device host computer will transmit commands to control the device operation.

FDD_1017. The device firmware shall accept and decode the host commands listed in Table 5-8. [FRD_0005, FRD_0007, FRD_0008, FRD_0009, FRD_0010, FRD_0011, FRD_0016, FRD_0025]

FDD_2008_Derived. The device firmware shall recognize either a <CR> (0x0D) or <LF> (0x0A) character as a host command terminator.

FDD_1018. The device firmware shall consider a host command reception complete when a command terminator character is received. [FRD_0015]

FDD_1019. The device firmware shall silently discard additional command terminator characters until the first non-terminator character is received. [FRD_0015, FRD_0024]

Command	Syntax	Values	Example
Abort	a<CR><LF>	n/a	Abort exposure a<CR><LF> 0x61,0x0D,0x0A
Diagnostic	D=1<CR><LF>	n/a	force a MCU watchdog reset D=1<CR><LF> 0x44,0x3D,0x31,0x0D,0x0A NOTE: no response is sent to the host
Exposure duration	e=nn.n<CR><LF>	00.5 second exposure 01.0 second exposure 02.0 second exposure 30.0 second exposure	set next exposure time to 0.5 seconds e=00.5<CR><LF> 0x65,0x3D,0x30,0x30,0x2E,0x35,0x0D,0x0A
Frequency PWM0	f=nnnn<CR><LF>	0750 Hz, 0800 Hz, 0850 Hz, 0900 Hz, 0950 Hz, 1000 Hz, 1050 Hz, 1100 Hz, 1150 Hz, 1200 Hz	set PWM0 frequency to 1,100 Hz f=1100<CR><LF> 0x66,0x3D,0x31,0x31,0x30,0x30,0x0D,0x0A
Go	g<CR><LF>	n/a	start exposure g<CR><LF> 0x67,0x0D,0x0A
Information	i<CR><LF>	n/a	Request status information i<CR><LF> 0x69,0x0D,0x0A
LED	l=n<CR><LF>	0 - off 1 - on	turn Flood LED off l=0<CR><LF> 0x6c,0x3d,0x30,0x0D,0x0A
Power	P=n<CR><LF>	0 - off 1 - on	turn boost power off P=0<CR><LF> 0x50,0x3D,0x30,0x0D,0x0A
Reset	R=1<CR><LF>	n/a	force a MCU software reset R=1<CR><LF> 0x52,0x3D,0x31,0x0D,0x0A NOTE: no response is sent to the host
Terminal	T=n	0 - off 1 - on	set terminal on – echo back host commands T=1<CR><LF> 0x54,0x3D,0x31,0x0D,0x0A

Command	Syntax	Values	Example
Voltage (digital pot)	v=nnn<CR><LF>	050 %, 060 %, 070 %, 080 %, 090 %, 100 %, 110 %, 120 %,	set voltage to 110 % of normal v=110<CR><LF> 0x76,0x3d,0x31,0x31,0x30,0x0D,0x0A
Voltage RAW pot value	V=nn<CR><LF>	0xFF – 0x00	set digital pot value to 0x28 V=28<CR><LF> 0x56,0x3D,0x32,0x38,0x0D,0x0A

Table 5-8 Host commands

FDD_1020. The device firmware shall begin command responses with a '>' character to indicate the host command was processed successfully. [FRD_0015]

FDD_1021. The device firmware shall begin responses with a '?' character to indicate an error occurred. [FRD_0015]

FDD_1022. The device firmware shall terminate responses with a <CR><LF> combination. [FRD_0015]

FDD_1023. While an exposure is in progress, the device shall reject any host commands except the abort command. [FRD_0015]

5.10.1 **Abort command**

When an exposure is active, the abort command terminates the exposure. If there is no exposure active, the abort command is ignored.

FDD_1024. The device firmware shall terminate an active exposure when an abort command is received from the host. [FRD_0010]

5.10.2 **Diagnostic command**

This diagnostic command forces the MCU to perform a watchdog reset, by entering an infinite loop. This command is not intended to be used during normal operation.

FDD_2013_Derived. The device firmware shall provide a command to force a MCU watchdog reset.

5.10.3 **Exposure Duration command**

The exposure duration command sets the exposure duration for subsequent exposures. The new exposure duration remains in effect until the device is reset, or another exposure command is received.

FDD_1026. The device firmware shall set the exposure duration to the value received. [FRD_0007]

FDD_2009_Derived. The device firmware shall reject exposure durations of less than 0.1 seconds or greater than 90 seconds.

5.10.4 **Frequency command**

The frequency command sets the PWM0 frequency for subsequent exposures. The new PWM0 frequency remains in effect until the device is reset, or another frequency command is received.

FDD_1027. The device firmware shall set the PWM0 frequency to a frequency as close to the requested frequency as allowed by the hardware. [FRD_0009]

5.10.5 **Go command**

The go command is only allowed when the firmware is in the ready state.

FDD_1028. The device firmware shall initiate an exposure when the go command is received from the host if the firmware is in the ready state. [FRD_0011]

The exposure is initiated by activating the PWM0 signal. The PWM0 signal is activated by configuring the PWM0 pin as an output. The PWM timer was started during initialization and runs continuously. This prevents a condition where the PWM timer might be turned off during the active duty cycle portion of the PWM signal, causing the duty cycle to be too long.

FDD_2021_Derived. The device firmware shall initiate an exposure by configuring the PWM0 pin as an output.

5.10.6 Information command

The responses to the information command include the information in Table 5-9. The data appears in the response data section. The data items are comma separated. Section 5.11 has additional information on the response format.

FDD_1029. The device firmware shall respond to the information command with the information listed in Table 5-9. [FRD_0016]

Data	Format and units
Flood LED status	- 1 byte ('0' or '1')
Exposure duration	- 4 bytes Seconds (ss.s)
Boost voltage setting	- 3 bytes %
PWM0 frequency	- 4 bytes Hertz
PON state	- 1 byte ('0' or '1')
Boost voltage reading	- 5 bytes Volts (vvv.v)
PWM0 width	- 2 bytes uS
Reset condition	- 2 bytes hex
Firmware version	- 9 bytes (Rnn.nn.nn)

Table 5-9 Information command data

5.10.7 LED command

The LED command turns on and off the flood LED. The flood LED remains in the commanded state until the device is reset or a new LED command is received to change the LED state.

FDD_1030. The device firmware shall operate the flood LED based on the parameter value of the LED command. [FRD_0008]

5.10.8 Voltage command

The voltage command sets the boost voltage by setting the value of the digital potentiometer. The new voltage setting remains in effect until the device is reset or another voltage command is received.

FDD_1031. The device firmware shall set the boost voltage to a value as close to the requested voltage as allowed by the hardware. [FRD_0005]

5.10.9 Power command

The power command turns the Boost Power off or on. This command is not intended to be used during normal operation.

FDD_2014_Derived. The device firmware shall provide a command to turn the Boost Voltage on or off.

5.10.10 Reset command

The reset command forces the MCU reset to perform a software reset. This command is not intended to be used during normal operation.

FDD_2015_Derived. The device firmware shall provide a command to force a software reset.

5.10.11 Terminal command

The terminal command causes commands sent from the host to be echoed back to the host. Additionally the 500 millisecond command timeout is not enforced. This command is not intended to be used during normal operation.

FDD_2016_Derived. The device firmware shall provide a command to turn terminal mode on and off.

5.10.12 Voltage RAW command

The voltage RAW command allows setting an explicit value to the Boost Voltage digital potentiometer. This command is not intended to be used during normal operation.

FDD_2017_Derived. The device firmware shall provide a command to set explicit values of the digital potentiometer.

5.10.13 Deleted

FDD_2018_Derived. Deleted.

5.11 Host Responses

The format of the responses to commands received from the host is described here. The responses are in printable ASCII characters.

{status},{error code},{command }[,{ data}]<CR><LF>

Field	Description
status	command status – '>' indicates success, '?' indicates failure
error code	system error code – a 1 byte value expressed as 2 ASCII hex digits
command	echo of the command sent by the host
data	optional – some commands may include additional data in this section
<CR><LF>	response terminator

Table 5-10 Host response formats

5.12 Exposure Control Routine

The exposure control routine monitors the boost voltage and times the exposure.

FDD_1033. The device firmware shall shut off the boost voltage if the voltage measured is less than 53V or greater than 200V. [FRD_0006]

FDD_1034. The device firmware shall terminate an exposure if the voltage measured is less than 53V or greater than 200V. [FRD_0006]

Shutting off the boost voltage is a fatal hardware error. After this event all host command replies will indicate failure. The device must be reset to recover from a fatal error.

The device firmware tracks the exposure duration by decrementing a 5 mS counter each time the main program loop is executed.

FDD_1035. The device firmware shall terminate an exposure when the exposure duration has been met. [FRD_0007]

The exposure control routine configures the PWM0 pin as an input at to terminate an exposure. This allows the PWM timer to continue to run, but prevents the PWM signal from being output on the PWM0 pin.

FDD_2020_Derived. The device firmware shall configure the PWM0 pin as an input to end an exposure.

5.13 SPI Communications

The ClearView device hardware includes two SPI components, a digital potentiometer to control the boost voltage, and an analog to digital converter to read the boost voltage.

FDD_1036. The device firmware shall transmit SPI commands to the digital potentiometer to control the component. [FRD_0005]

FDD_1037. The device firmware shall receive SPI data from the analog to digital converter component. [FRD_0006]

5.14 Interrupt Service Routines

The communication interrupt service routines are light weight and intended buffer the communications data. Actual processing of the communications occurs in the main program loop.

FDD_1038. The device firmware shall use an interrupt service routine to collect the characters received from the host. [FRD_0012, FRD_0014]

FDD_1039. The device firmware shall use an interrupt service routine to transmit responses to the host. [FRD_0012, FRD_0014]

FDD_1040. The device firmware shall use a timer interrupt to initiate execution of the main program loop at 5 mS intervals. [FRD_0003, FRD_0012]

FDD_2010_Derived. The device firmware shall use an interrupt service routine to send and receive multi-byte SPI data.

5.15 Error Codes

Error codes are 1 byte long and are represented as 2 ASCII hexadecimal digits. The ‘0x’ format identifier is omitted when outputting error codes. The error codes are binary bit values.

Error code	Description
0x00	No error
0x80	General error
0x81	Bad command
0x82	TxBuf full
0x84	RxBuf full
0x88	Boost voltage error
0x90	SPI error1
0xA0	SPI error2
0x40	Boost startup delay (information only)

Table 5-11 Error codes

5.16 Deleted

Document Revision History

Cause for Change:	Version Number:	Description of Change:	Date:	Updated by:
Development	0.1	Initial document creation	23-Nov-10	B. Tompson
Development	0.2	1st draft	03-Dec-10	B. Tompson
Development	0.3	1st draft without out-of-band 'Go' command	12-Dec-10	B. Tompson
Development	0.4	Applied design review comments EGF-003	13-Dec-10	B. Tompson
Development	0.5	Correct values of CR & LF Modified information command data Removed voltage settings 130, 140 Changed host baud rate Changed watchdog timeout to 32 mS Updated Error codes in Table 5-11	01-Feb-11	B. Tompson
Development	1.0	Added diagnostic commands to mainline code. Added PWM initialization code to prevent long first PWM0 pulse after power up. Cleaned up paragraph numbering and TOC.	18-Dec-11	B. Tompson
PWM Problem report	1.1	Changed the control of the PWM0 signal to gate the output of the signal instead of turning of the PWM timer. Removed the PWM pulse "Width" command from the recognized host commands to prevent changing the PWM0 pulse width Added detail to the PWM0 Frequencies table Minor cleanup of wording and formatting	03-Feb-12	B. Tompson

Cause for Change:	Version Number:	Description of Change:	Date:	Updated by:
Design review	1.2	Added 'reset watchdog timer' to Figure 4-1 Added power up information message to section 4.2 Added watchdog timeout details Corrected firmware clock values in Table 5-1, and Table 5-5 Added additional default values to Table 5-7 Added boost voltage limits to section 5.8, FDD_1033, and FDD_1034 Repaired MS-Word table and figure captions and cross references Minor grammar and spelling corrections	24-Feb-12	B. Tompson
Hardware review	1.3	Removed Boost Voltage value from default settings	15-Mar-12	B. Tompson
Final Approval	000	Final design specification from initial redevelopment.		B. Tompson