Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
Chipsmall
采购部

CEC1702 Efuse Generator Tool User's Guide

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BeaconThings, BitCloud, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KEELOQ, KEELOQ logo, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, RightTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, CryptoAuthentication, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, QMatrix, RightTouch logo, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

ISBN: 9781522415695

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Table of Contents

Preface

NOTICE TO CUSTOMERS

All documentation becomes dated, and this manual is no exception. Microchip tools and documentation are constantly evolving to meet customer needs, so some actual dialogs and/or tool descriptions may differ from those in this document. Please refer to our web site (www.microchip.com) to obtain the latest documentation available.

Documents are identified with a "DS" number. This number is located on the bottom of each page, in front of the page number. The numbering convention for the DS number is "DSXXXXXA", where "XXXXX" is the document number and "A" is the revision level of the document.

For the most up-to-date information on development tools, see the MPLAB® IDE online help. Select the Help menu, and then Topics to open a list of available online help files.

INTRODUCTION

This chapter contains general information that will be useful to know before using the CEC1702 Efuse Generator Tool User's Guide. Items discussed in this chapter include:

- Document Layout
- Conventions Used in this Guide
- The Microchip Web Site
- Development Systems Customer Change Notification Service
- Customer Support
- Document Revision History

DOCUMENT LAYOUT

This document describes how to use the CEC1702 Efuse Generator Tool User's Guide as a development tool for CEC1702 Efuse Programming. The manual layout is as follows:

- **Chapter 1. "Introduction"** – This chapter provides information on programming the Efuse using the CEC1702 Efuse Generator Tool.
- **Chapter 2. "CEC1702 Efuse Generator Tool Procedure"** – This chapter contains a step by step procedure.
- **Appendix A. "CEC1702 Clicker Board Schematic"** – This appendix contains schematic diagram of the CEC1702 Clicker board.
- **Appendix B. "Programming Steps For MikroE Tool"** – This section explains steps to program the Efuse using mikroProg Suite for ARM.
- **Appendix C. "References"** – This appendix includes helpful references.

CONVENTIONS USED IN THIS GUIDE

This manual uses the following documentation conventions:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic characters	Referenced books	*MPLAB® IDE User’s Guide*
	Emphasized text	...is the *only* compiler...
Initial caps	A window	the Output window
	A dialog	the Settings dialog
	A menu selection	select Enable Programmer
Quotes	A field name in a window or dialog	“Save project before build”
Underlined, italic text with right angle bracket	A menu path	<u>*File>Save*</u>
Bold characters	A dialog button	Click **OK**
	A tab	Click the **Power** tab
N‘Rnnnn	A number in verilog format, where N is the total number of digits, R is the radix and n is a digit.	4‘b0010, 2‘hF1
Text in angle brackets < >	A key on the keyboard	Press <Enter>, <F1>
Courier New font:		
Plain Courier New	Sample source code	`#define START`
	Filenames	`autoexec.bat`
	File paths	`c:\mcc18\h`
	Keywords	`_asm, _endasm, static`
	Command-line options	`-Opa+, -Opa-`
	Bit values	`0, 1`
	Constants	`0xFF, ‘A’`
Italic Courier New	A variable argument	*`file`*`.o`, where *`file`* can be any valid filename
Square brackets [ ]	Optional arguments	`mcc18 [options]` *`file`* `[options]`
Curly brackets and pipe character: { | }	Choice of mutually exclusive arguments; an OR selection	`errorlevel {0|1}`
Ellipses...	Replaces repeated text	`var_name [, var_name...]`

	Represents code supplied by user	`void main (void)`
		`{ ...`
		`}`

THE MICROCHIP WEB SITE

Microchip provides online support via our web site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

DEVELOPMENT SYSTEMS CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer Change Notification and follow the registration instructions.

The Development Systems product group categories are:

- **Compilers** – The latest information on Microchip C compilers, assemblers, linkers and other language tools. These include all MPLAB C compilers; all MPLAB assemblers (including MPASM assembler); all MPLAB linkers (including MPLINK object linker); and all MPLAB librarians (including MPLIB object librarian).
- **Emulators** – The latest information on Microchip in-circuit emulators.This includes the MPLAB REAL ICE and MPLAB ICE 2000 in-circuit emulators.
- **In-Circuit Debuggers** – The latest information on the Microchip in-circuit debuggers. This includes MPLAB ICD 3 in-circuit debuggers and PICkit 3 debug express.
- **MPLAB IDE** – The latest information on Microchip MPLAB IDE, the Windows Integrated Development Environment for development systems tools. This list is focused on the MPLAB IDE, MPLAB IDE Project Manager, MPLAB Editor and MPLAB SIM simulator, as well as general editing and debugging features.
- **Programmers** – The latest information on Microchip programmers. These include production programmers such as MPLAB REAL ICE in-circuit emulator, MPLAB ICD 3 in-circuit debugger and MPLAB PM3 device programmers. Also included are nonproduction development programmers such as PICSTART Plus and PIC-kit 2 and 3.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at:
http://www.microchip.com/support

DOCUMENT REVISION HISTORY

Revision	Section/Figure/Entry	Correction
DS50002598A (04-10-17)	Document Release	

Chapter 1. Introduction

1.1 CEC1702 EFUSE GENERATOR TOOL

1.1.1 Purpose

This document in intended for software engineer planning to generate public and private keys, signature for their firmware and custom Efuse data for CEC1702 for their system.

This document gives the introduction to the steps involved in CEC1702 Efuse programming as well as the hardware and software requirements for the custom Efuse programming once you get the blank part from Microchip.

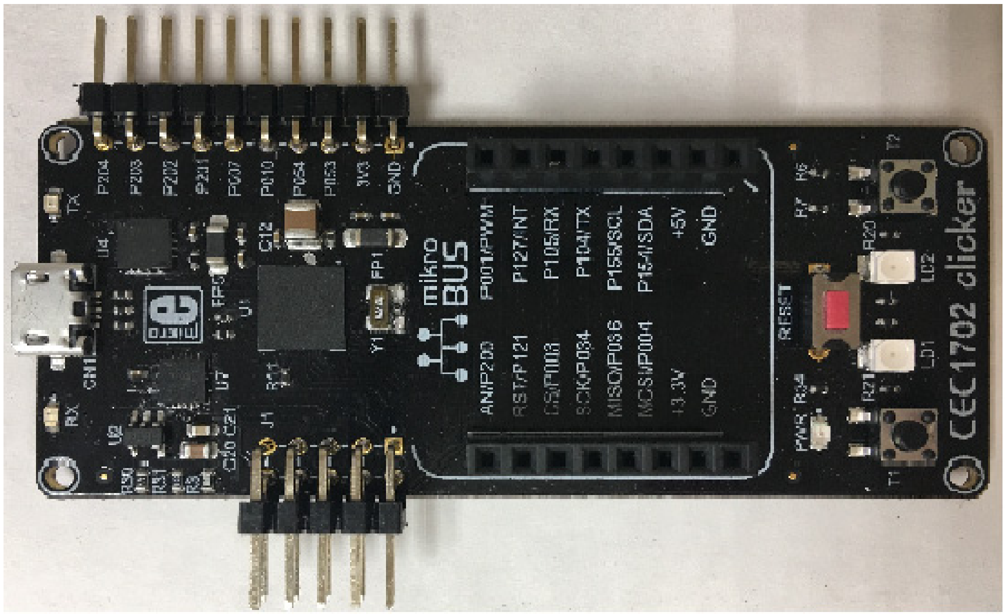
1.1.2 Introduction

This document gives step by step information on programming the Efuse using the CEC1702 Efuse Generator Tool. Following these steps will ensure that the CEC1702 part is programmed correctly each time without facing any difficulty. The CEC1702 Efuse Generator Tool is responsible for generating private and public key pairs, verifying private-public key pair validity, disabling JTAG and ATE modes and generating the Efuse programming values. The input from the user is GUI based and is simple to use. The output from the tool is a binary file for programming the CEC1702 part on the Clicker board using any JTAG programmer tool. The tool also writes the private and public key in an output file protected with the password set by the user while entering the parameters.

1.1.3 Introduction to Hardware

The picture of the hardware board used for programming is shown below in **Figure 1-1: "CEC1702 Clicker board"**.

FIGURE 1-1: CEC1702 CLICKER BOARD

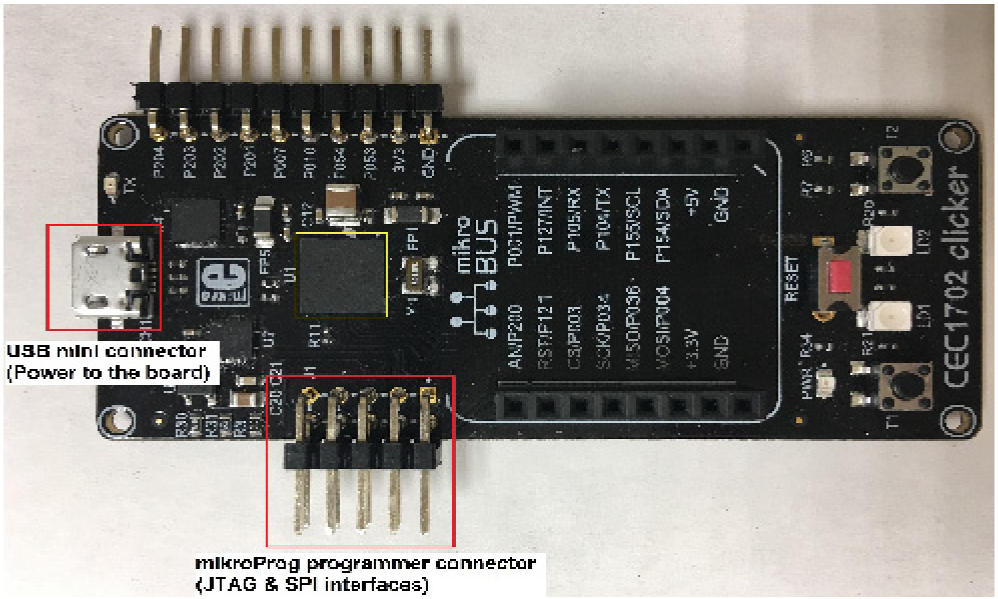


The Clicker board consist of following interfaces

1. USB mini Connector (Power to the board)
2. MikroeProg Programmer connector (JTAG and SPI)

Figure 1-2: "CEC1702 Clicker board Interface details" shows the location of various interfaces listed above on the Clicker board:

FIGURE 1-2: CEC1702 CLICKER BOARD INTERFACE DETAILS



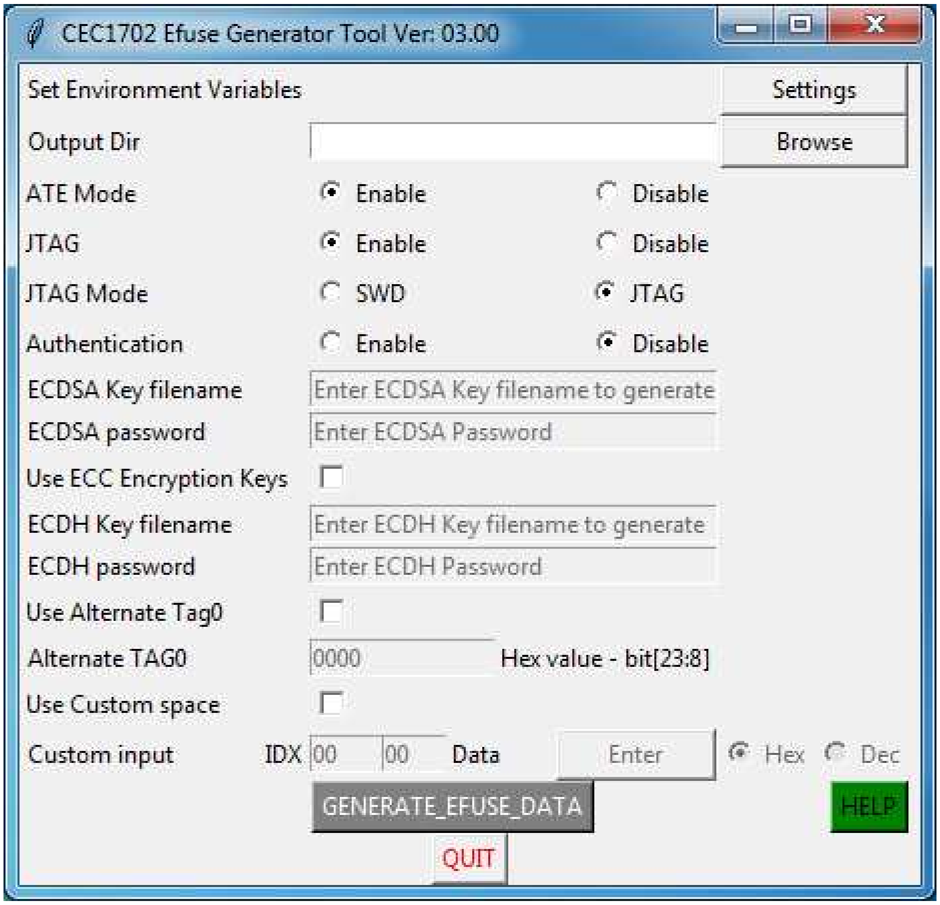
1.1.4 Introduction to the Programming Utility

The CEC1702 Efuse Generator Tool is an executable file with the name "CEC1702_efuseGEN.exe". This tool is targeted to program the following bits in the OTP:

1. ATE mode disable bit - Efuse byte 35 bit[7]
2. TAG enable bit on Boot ROM Exit - Efuse byte 34 bit[1] - Bit[4]
3. Authentication Bit for the ECDSA key generation - Efuse byte 483 bit[0]
4. Authentication key region - Efuse bytes 128 - 191
5. ECC private key region - Efuse byte 0 - 31
6. Custom User space region - Efuse byte 192 - 479
7. Custom TAG update region - Efuse bytes 508 - 509

Once you run the utility, you will see the GUI as shown in **Figure 1-3: "CEC1702 Efuse Programming tool input form"**. The user is required to fill in the following information in the tool to generate the binary and hex file required to program the Efuse in CEC1702. Details about using this tool are described in **Chapter 2. "CEC1702 Efuse Generator Tool Procedure"**.

FIGURE 1-3: CEC1702 EFUSE PROGRAMMING TOOL INPUT FORM



CEC1702 Efuse Generator Tool Ver: 03.00

Set Environment Variables

Output Dir

ATE Mode ● Enable ○ Disable

JTAG ● Enable ○ Disable

JTAG Mode ○ SWD ● JTAG

Authentication ○ Enable ● Disable

ECDSA Key filename: Enter ECDSA Key filename to generate

ECDSA password: Enter ECDSA Password

Use ECC Encryption Keys □

ECDH Key filename: Enter ECDH Key filename to generate

ECDH password: Enter ECDH Password

Use Alternate Tag0 □

Alternate TAG0: 0000 Hex value - bit[23:8]

Use Custom space □

Custom input IDX 00 00 Data ● Hex ○ Dec

1.1.5 Prerequisites

Following are the prerequisites for generating the header, hex and binary file for programming described in **Chapter 2. “CEC1702 Efuse Generator Tool Procedure”**:

1. Clicker board with blank CEC1702 should be available.
2. The hardware board must have proper input circuitry for providing 1.59V Efuse programming voltage.
3. PC should have Windows7/8/10.
4. The user is expected to have downloaded and installed openssl version 1.0.1e. Please refer to the following •
5. The user is expected to have downloaded the CEC1702 Efuse Generator Tool on your PC. If the user is using mikroProg tool for programming the Clicker board, this utility is part of the tool package.
6. The user is expected to have any JTAG programmer tool installed on their PC for programming the Efuse hex / binary file in CEC1702 on the Clicker board.
7. The user is expected to have read the CEC1702 data sheet and know all the encryption modes planned to be used in their system.
8. The user is expected to know which keys need to be generated for their system and application.

Note:

- CEC1702 Efuse Generator Tool is only supported for Windows at this time.
- One may download openssl from https://www.openssl.org/source/old/1.0.1/. Version number is openssl-1.0.1e.tar.gz or later.
- Please refer to CEC1702 Clicker board documentation for other requirements for executing step **2.**

Chapter 2. CEC1702 Efuse Generator Tool Procedure

2.1 CEC1702 EFUSE PROGRAMMING PROCEDURE

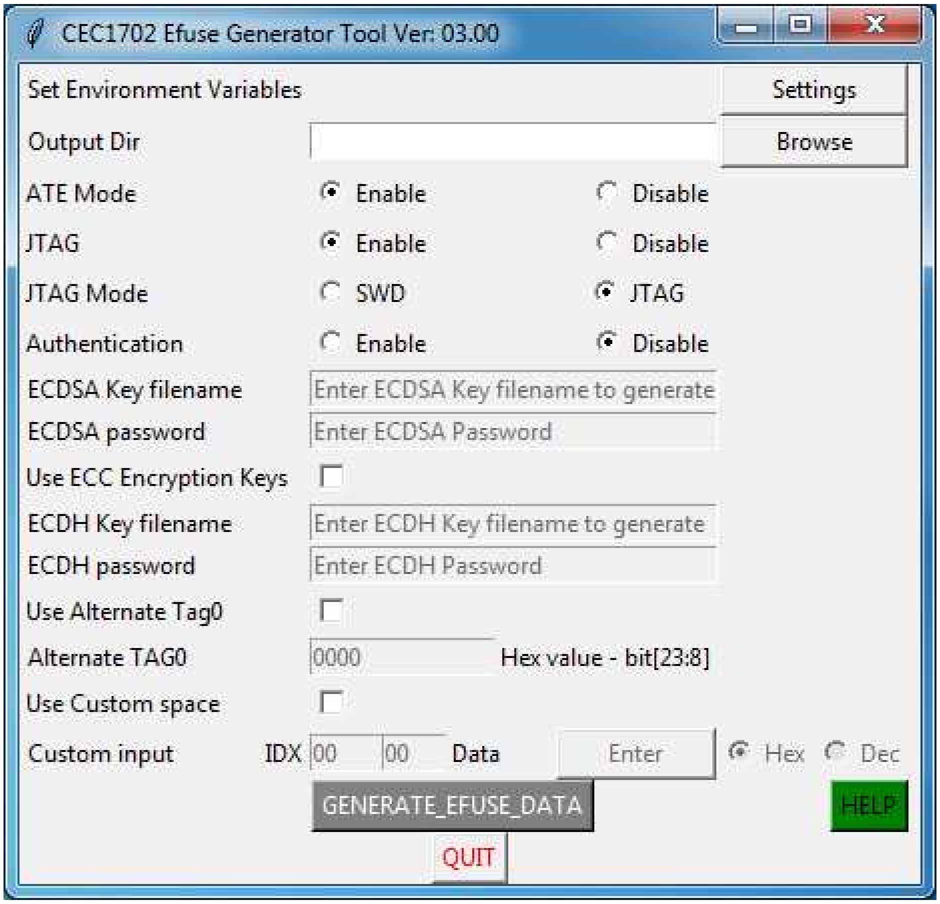
2.1.1 Directory Structure

Once you download and uncompress the "CEC1702 Efuse Generator Tool" utility, the directory structure appears as shown below:

```
<efuse_generator>
    |  CEC1702_efuseGEN.exe          -> CEC1702 efuse generator
    |  config.ini                    -> Environment file details
    +---efuse
    |  +---original_binary           -> Build output Binaries
    |  |      otp_prog_original.bin  -> Binary format
    |  |      otp_prog_original.hex  -> Intel Hex format
    |  |
    |  \---test_keys                 -> Test Key files
    |         ecprivkey001.pem       -> Sample Private key
    |         ecprivkey001_crt.pem   -> Sample Private Certificate
    |         ecprivkey002.pem       -> Sample Private key
    |         keys.txt
    \---tools                        -> Support tools
        CEC1702_key_extractor.exe    -> Tool to extract Key content
        openssl.exe                  -> Tool to generate Keys
        srec_cat.exe                 -> Tool for bin to hex conversion
```

You may install the openssl tool at any other directory path also. In such a case you will have to provide the openssl tool path in Set Environment Variable "Settings" button. This is shown in **Figure 2-1: "CEC1702 Efuse Generator Tool Main Form"**.

FIGURE 2-1: CEC1702 EFUSE GENERATOR TOOL MAIN FORM



CEC1702 Efuse Generator Tool Ver: 03.00

Set Environment Variables [Settings]

Output Dir [Browse]

ATE Mode ● Enable ○ Disable

JTAG ● Enable ○ Disable

JTAG Mode ○ SWD ● JTAG

Authentication ○ Enable ● Disable

ECDSA Key filename: Enter ECDSA Key filename to generate

ECDSA password: Enter ECDSA Password

Use ECC Encryption Keys □

ECDH Key filename: Enter ECDH Key filename to generate

ECDH password: Enter ECDH Password

Use Alternate Tag0 □

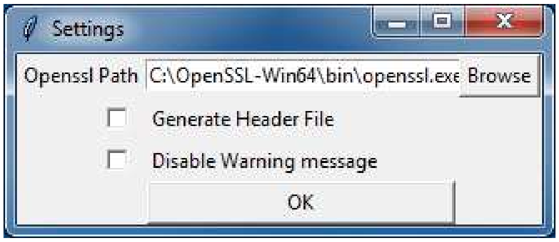
Alternate TAG0: 0000 Hex value - bit[23:8]

Use Custom space □

Custom input IDX 00 00 Data [Enter] ● Hex ○ Dec

When you click the Settings button, form in **Figure 2-2: "Openssl Tool Path Setting"** opens up. Please set the path either by typing the full path or using the Browse option. Click "OK" button after entering the path and making the other choices in this window. The explanation of other options is mentioned in steps **3.** and **4.** of **2.1.3 "Key Generation Steps"**.

FIGURE 2-2: OPENSSL TOOL PATH SETTING



Settings

Openssl Path: C:\OpenSSL-Win64\bin\openssl.exe [Browse]

□ Generate Header File

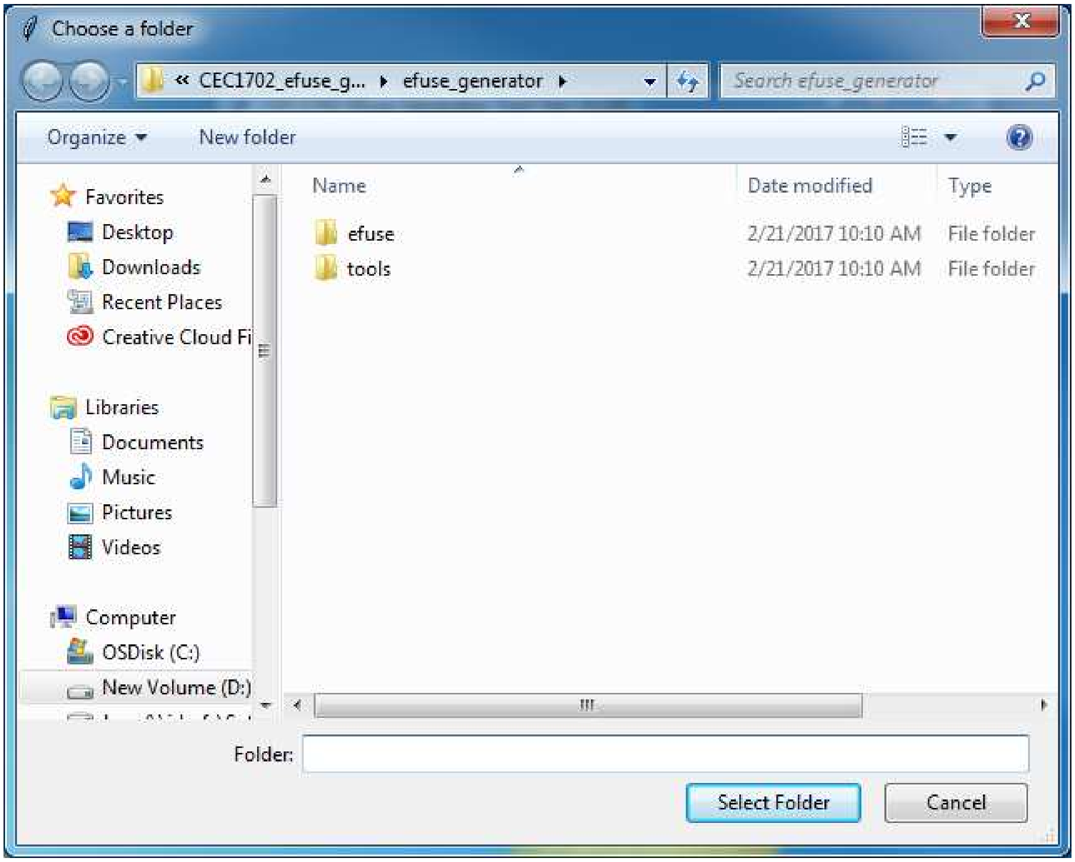
□ Disable Warning message

The setting of openssl tool path needs to be done only for the first time when you launch the program.

2.1.2 Output Files

You may define your own output directory for the output files generated from the tool. You may type the output directory path directly or select the output directory by clicking the Browse button shown in **Figure 2-1: “CEC1702 Efuse Generator Tool Main Form”**. The window in **Figure 2-3: “Select Output Directory Form”** will open up. If the “Output Dir” field is left blank, the tool will be generating the runtime output directory under <root>\efuse\ as efuse_<YYYYMMDD>_<WHHMMSS> where YYYYMMDD represents Year, Month and Date respectively and WHHMMSS represents the Week of the month, Hour, Minute and Second respectively.

FIGURE 2-3: SELECT OUTPUT DIRECTORY FORM



Name	Date modified	Type
efuse	2/21/2017 10:10 AM	File folder
tools	2/21/2017 10:10 AM	File folder

On program execution the tool generates two folders in the output directory “Output Dir”. These are “keys” and “out_binaries”. As the name suggests, the files for programming the CEC1702 are stored in the “out_binaries” directory. The ECDH private key (AES Encrypted), self signed certificate, ECDSA private key (AES Encrypted), ECDSA Self Signed Certificate and their corresponding certificate requests are stored in the “keys” directory. The pictorial representation of the same is given below:

```
<efuse_generator>
 |
 +---efuse
     +--efuse_<YYYYMMDD>_<WHHMMSS>   -> Output Dir self generated
       +---keys                    -> Contains all the keys
       |      <ECDH>.pem           -> ECDH Private key AES Encrypted
       |      <ECDH>_crt.pem       -> ECDH Self Signed Certificate
       |      <ECDH>_csr.pem       -> ECDH Certificate Request
       |      <ECDSA>.pem          -> ECDSA Private key AES Encrypted
       |      <ECDSA>_crt.pem-> ECDSA Self Signed Certificate
       |      <ECDSA>_csr.pem -> ECDSA Certificate Request
       |      keys_info.txt        -> Generated Key details
       |      key_file.bin         -> key_file.bin extracted key output
       |
       \---out_binaries
       |      otp_efuse.bin        -> updated Binary with Efuse details
       |      otp_efuse.hex        -> Equivalent hex file for download
       |
       \---efuse_log.txt           -> Efuse Log file
```

2.1.3 Key Generation Steps

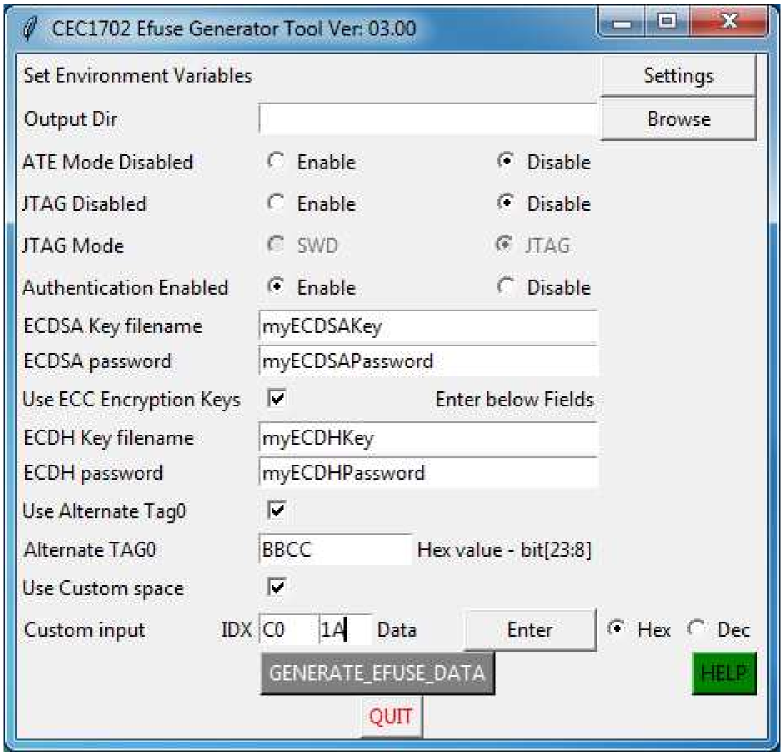
Following are the steps to generate the Efuse binary and hex file for programming the CEC1702 blank chip:

1. Click on the “CEC1702_efuseGEN.exe” program shortcut to start the application and the GUI will be displayed as in **Figure 2-1: “CEC1702 Efuse Generator Tool Main Form”**.
2. Set the openssl path as described in section **2.1.1 “Directory Structure”**.
3. If “Generate Header File” is selected in **Figure 2-2: “Openssl Tool Path Setting”**, the utility will generate a header file equivalent to the Efuse binary file. This file may then be used to add to any project environment, if required. If “Generate Header File” option is not selected, utility will not generate the header file. “Genrate Header File” option must be used if the Efuse programmer tool requires header file.

Note: Please refer to JTAG programmer documentation for deciding which file format (.hex, .bin or .h) it requires for programming the Efuse of CEC1702.

4. The CEC1702 Efuse Generator Tool generates a warning message shown in **Figure 2-8: "Warning Message Before Generating The Keys"** and described in step **26.** This message could be disabled using the "Disable Warning message" select option provided in settings window in **Figure 2-2: "Openssl Tool Path Setting"**.
5. Set the output directory path as described in section **2.1.2 "Output Files"**.
6. Select Disable ATE radio button to bring the chip out of the ATE mode after Efuse / OTP programming. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
7. If there are more programming steps, you may choose to leave ATE Mode enabled.
8. Select JTAG Disable radio button to disable the JTAG interface after Efuse programming. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
9. Our recommendation is to disable JTAG mode after all the programming is completed in production.
10. JTAG mode allows you to choose between SWD (2-Wire) or JTAG(4-wire) mode. This option is only available when JTAG is Enabled. One may use this on engineering parts for run time debug.
11. In case you would like to have authentication enabled, select the "Authentication" Enable radio button. This will in turn enable the firmware Header and Image signing.
12. Only when authentication is enabled, one can generate the ECDSA Key. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
13. If authentication of the firmware binary is enabled, in the field marked "ECDSA Key filename", enter the file name for the ECDSA Key. One just needs to enter the key file name without an extension. For example: myECDSAkey. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
14. The "ECDSA Password" field will be used to protect the ECDSA Key using AES-256-CBC encryption. The generated Keys and Signatures will be encrypted using AES-256-CBC encryption mode and stored in the output directory. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
15. If ECC encryption Keys are required, select the "Use ECC Encryption Keys" select button. This will enable entry into the "ECDH Key filename" field. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
16. If "Use ECC Encryption Keys" is selected, enter the file name for the ECDH Key. One just needs to enter the key file name without an extension. For example: myECDHkey. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
17. The "ECDH Password" field will be used to protect the ECDH Key using AES-256-CBC encryption. The generated Keys and Signatures will be encrypted using AES-256-CBC encryption mode and stored in the output directory.
18. If the user wishes to use alternate Tag0, select "Alternate Tag0".
19. If "Alternate Tag0" is enabled, alternate TAG fields will be added to the Efuse data. The user needs to provide bits [23:8] for the Alternate TAG address. The field must be written with hexadecimal values. For example: If the address in SPI Flash is 0xAABBCC00, the user needs to write alternate tag field with 0xBBCC value. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.
20. If the user wishes to program the custom space in Efuse / OTP, one must select the "Use Custom space" field. See **Figure 2-4: "Sample CEC1702 Efuse Generator Tool Entry"** for reference.

FIGURE 2-4: SAMPLE CEC1702 EFUSE GENERATOR TOOL ENTRY



CEC1702 Efuse Generator Tool Ver: 03.00

Set Environment Variables — Settings

Output Dir — Browse

ATE Mode Disabled ○ Enable ● Disable

JTAG Disabled ○ Enable ● Disable

JTAG Mode ○ SWD ● JTAG

Authentication Enabled ● Enable ○ Disable

ECDSA Key filename: myECDSAKey

ECDSA password: myECDSAPassword

Use ECC Encryption Keys ☑ Enter below Fields

ECDH Key filename: myECDHKey

ECDH password: myECDHPassword

Use Alternate Tag0 ☑

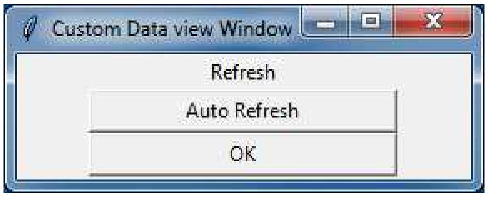
Alternate TAG0: BBCC Hex value - bit[23:8]

Use Custom space ☑

Custom input IDX C0 1A Data Enter ● Hex ○ Dec

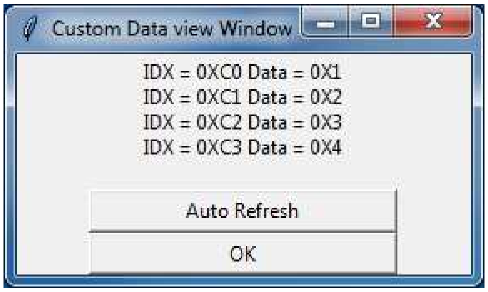
21. If “Use Custom space” field is enabled, the Custom Efuse data space from 192-479 can be used for storing user data. Please refer to Chapter 37 “eFUSE Block” of CEC1702 Data Sheet for more details.
22. The “Custom input” will be active if 'Use Custom Space' is selected. “IDX” refers to the custom space offset from 192-479 (decimal value, the equivalent hexadecimal value is 0xC0 to 0x1DF). “Data” refers to the data value to be written to the offset address. This tool allows the IDX and Data values to be entered in Hexadecimal (Hex) or Decimal (Dec) format with the radio button selection option. The default mode is hexadecimal. The user will need to add one data at a given address at a time and will need to use the “Enter” button on the form to enter each data value in the location. IDX will be checked against valid entry for the custom space and return an error message, if the address offset value is outside of the range. **Figure 2-7: “Index Out Of Range Indication”** shows the error message window that would be displayed, if the wrong entry is entered. If the entry is correct the window shown in **Figure 2-5: “Custom Data View Window”** will pop up.
23. Clicking on the “Auto Refresh” button of **Figure 2-5: “Custom Data View Window”** refreshes the custom data and address as it is entered. This is shown in **Figure 2-6: “Example Of Custom input Data display”**. Clicking the “OK” button will close the window and no further information on “Custom input” data will be displayed.

FIGURE 2-5: CUSTOM DATA VIEW WINDOW



24. **Figure 2-6: “Example Of Custom input Data display”** shows an example of how the Custom input data would appear in the Custom Data View Window. The values in **Figure 2-6: “Example Of Custom input Data display”** are just an example. Customers must program their required values in the custom data input field.

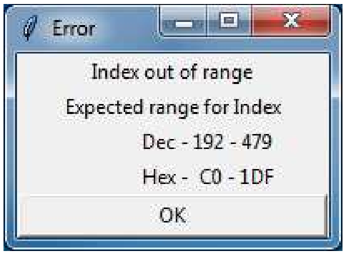
FIGURE 2-6: EXAMPLE OF CUSTOM INPUT DATA DISPLAY



```
IDX = 0XC0 Data = 0X1
IDX = 0XC1 Data = 0X2
IDX = 0XC2 Data = 0X3
IDX = 0XC3 Data = 0X4
```

25. Click the “Generate Efuse” button to generate the Efuse key.
26. If “Disable Warning message” is not selected, the window shown in **Figure 2-8: “Warning Message Before Generating The Keys”** will pop up and will display the configuration options selected by the user along with the list of custom inputs entered by the user along with the address where these are being programmed.
27. If there are changes to be made, the user may click the “Quit” button in the form shown in **Figure 2-8: “Warning Message Before Generating The Keys”** and re-enter his configuration. 'Quit' will exit out without generating any output files with default menu option. This means that the option selected earlier will be discarded and a fresh selection will have to be done.

FIGURE 2-7: INDEX OUT OF RANGE INDICATION



Index out of range

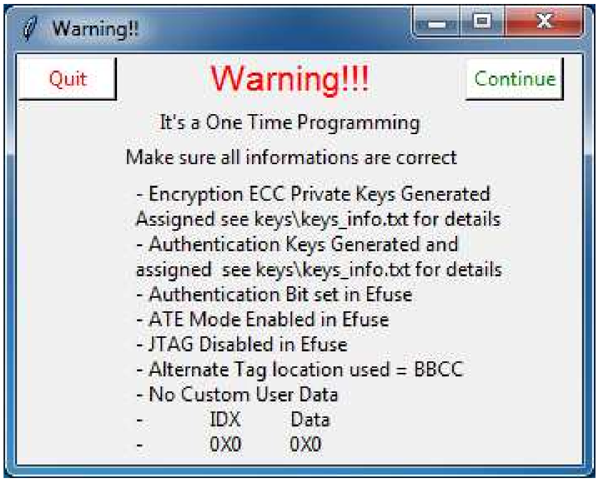
Expected range for Index

Dec - 192 - 479

Hex - C0 - 1DF

28. If the configuration is correct, the user may click the “Continue” button in the form shown in **Figure 2-8: “Warning Message Before Generating The Keys”**and generate the files.

FIGURE 2-8: WARNING MESSAGE BEFORE GENERATING THE KEYS



Warning!!!

It's a One Time Programming

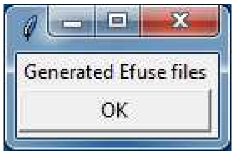
Make sure all informations are correct

- Encryption ECC Private Keys Generated Assigned see keys\keys_info.txt for details
- Authentication Keys Generated and assigned see keys\keys_info.txt for details
- Authentication Bit set in Efuse
- ATE Mode Enabled in Efuse
- JTAG Disabled in Efuse
- Alternate Tag location used = BBCC
- No Custom User Data

```
-        IDX      Data
-        0X0      0X0
```

29. Once the generation is complete, you will get a pop up window as shown in **Figure 2-9: “Efuse File Generation Completion Message”**.
30. The output binary and hex file will be generated and stored in the out_binaries directory.
31. The output file names are fixed. These are otp_efuse.bin, efuse_data.h and otp_efuse.hex.
32. If the programmer requires a header file to compile with the environment, the efuse_data.h header file may be used as the input. Please refer to step 3.
33. If any JTAG programmer is used to program the board, then otp_efuse.hex or otp_efuse.bin file will be used as the input, depending on the tool requirements. The start address of JTAG download code should be 0xE0000. Once your Efuse download is completed, write 0xE0349 to address 0x40002738 to the reset handler address itself, to launch the downloaded code.

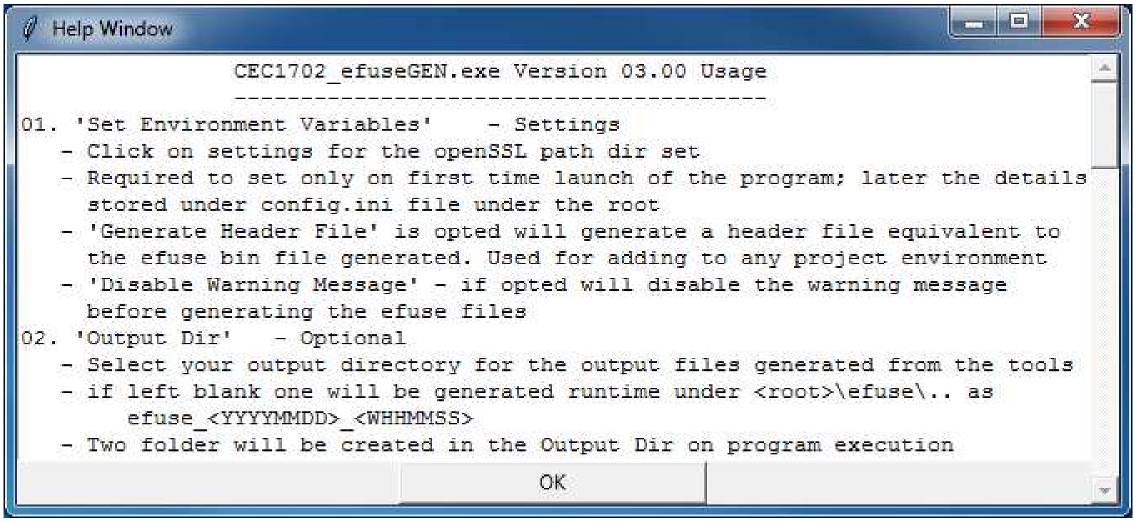
FIGURE 2-9: EFUSE FILE GENERATION COMPLETION MESSAGE



Generated Efuse files

34. The “HELP” button shown in **Figure 2-1: “CEC1702 Efuse Generator Tool Main Form”** is a good source of quick reference. The form shown in **Figure 2-10: “Help Window”** opens up on clicking the “HELP” button.

FIGURE 2-10: HELP WINDOW



```
                CEC1702_efuseGEN.exe Version 03.00 Usage
                ----------------------------------------
01. 'Set Environment Variables'    - Settings
   - Click on settings for the openSSL path dir set
   - Required to set only on first time launch of the program; later the details
     stored under config.ini file under the root
   - 'Generate Header File' is opted will generate a header file equivalent to
     the efuse bin file generated. Used for adding to any project environment
   - 'Disable Warning Message' - if opted will disable the warning message
     before generating the efuse files
02. 'Output Dir'   - Optional
   - Select your output directory for the output files generated from the tools
   - if left blank one will be generated runtime under <root>\efuse\.. as
        efuse_<YYYYMMDD>_<WHHMMSS>
   - Two folder will be created in the Output Dir on program execution
```

Note:

- Keys will not be visible in text format, unless requested to extract the portion of the keys.
- It is the end user's responsibility to make sure that their keys and passwords are stored in a secure location as there is no way to recover the keys after the device is programmed and the password for the encrypted keys is lost.

Appendix A. CEC1702 Clicker Board Schematic

The schematic of the CEC1702 Clicker board is shown in **Figure A-1: "Schematic Diagram of CEC1702 Clicker board"**.

FIGURE A-1: SCHEMATIC DIAGRAM OF CEC1702 CLICKER BOARD

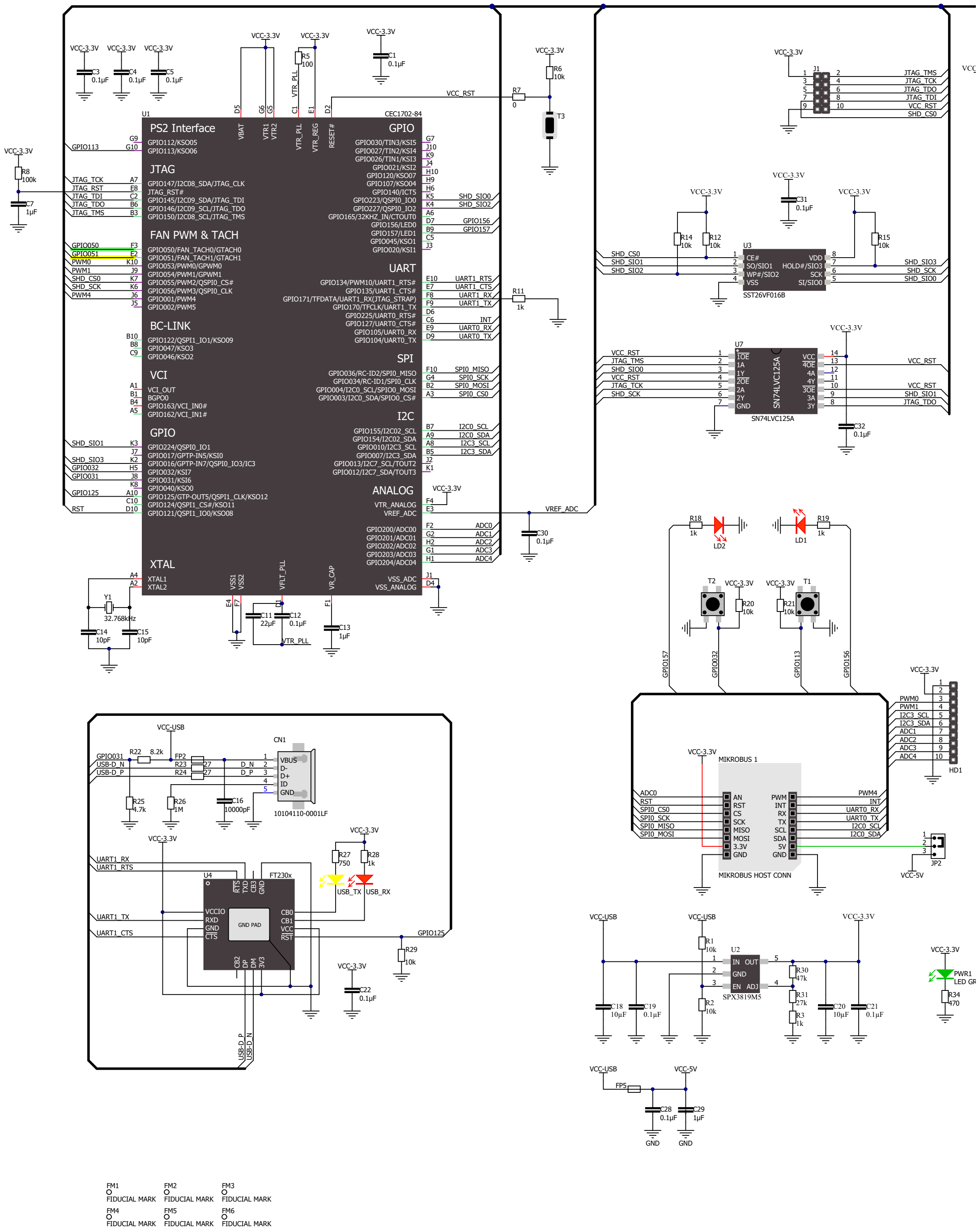
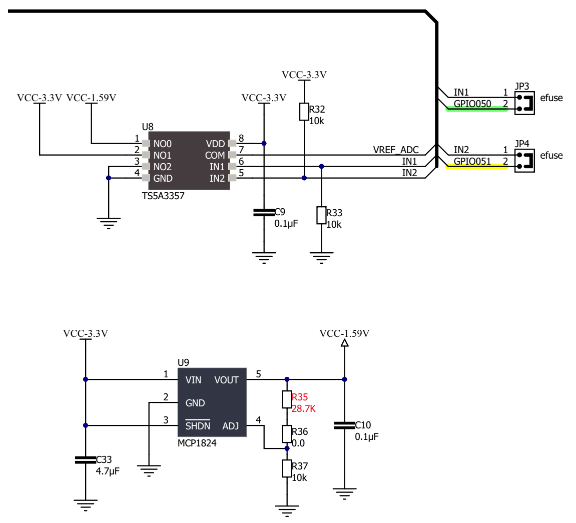


FIGURE A-1: SCHEMATIC DIAGRAM OF CEC1702 CLICKER BOARD (CONTINUED)



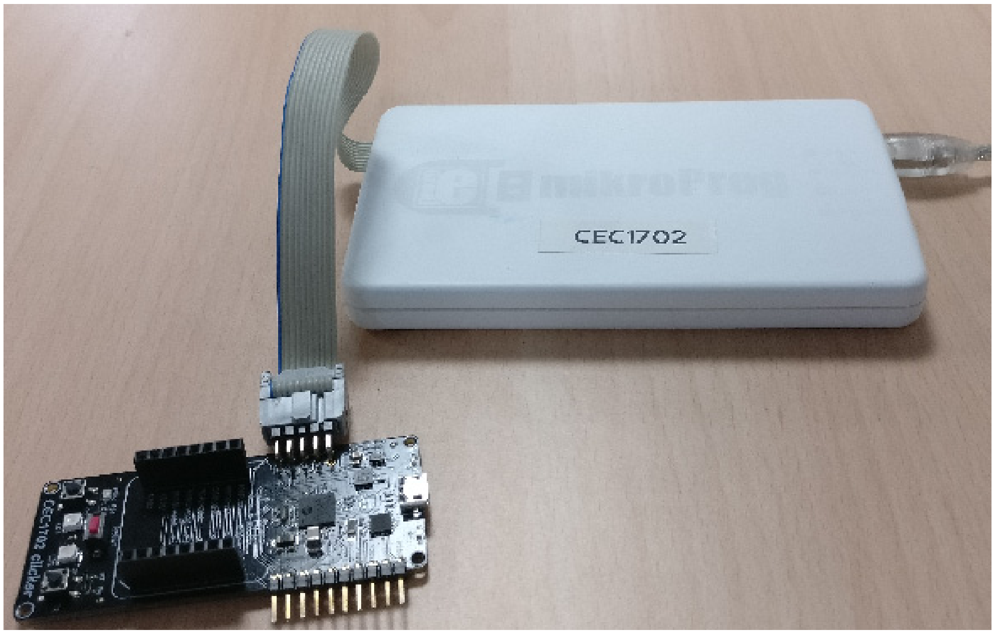
Appendix B. Programming Steps For MikroE Tool

B.1 STEPS TO PROGRAM THE EFUSE USING MIKROPROG SUITE FOR ARM

If the user plans to use the mikroProg Suite for ARM for programming, the following are the steps for programming the Efuse

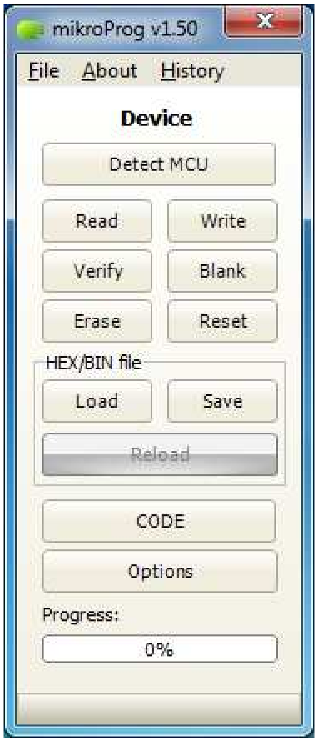
1. Power up the Clicker board.
2. Connect the microProg to the Clicker board and the computer as shown in **Figure B-1: "CEC1702 Clicker board Setup"**.

FIGURE B-1: CEC1702 CLICKER BOARD SETUP



3. Double click on the mikroProg suite for ARM tool program shortcut and open the mikroProg tool. **Figure B-2: "mikroProg Suite for ARM Screen"** would open up.

FIGURE B-2: MIKROPROG SUITE FOR ARM SCREEN



4. Click on the “Detect MCU” button in **Figure B-2: “mikroProg Suite for ARM Screen”** to detect the device connected.
5. Click on “Options” button. **Figure B-3: “Options Menu in mikroProg”** would open up.
6. Under Interface “Hardware” section (top portion in **Figure B-3: “Options Menu in mikroProg”**), click on the drop down menu and select CEC.
7. Click the “Program EFUSE” button in **Figure B-3: “Options Menu in mikroProg”**, “Open” window would pop up as shown in **Figure B-4: “Open Menu”**. Please select the otp_efuse.hex file generated earlier using instructions in section **2.1.3 “Key Generation Steps”** and click the open button.
8. The “Program EFUSE” button in **Figure B-3: “Options Menu in mikroProg”** turning green indicates a successful Efuse programming of CEC1702 Clicker board.
9. LD1 LED on the board will be ON. This LED indicates that programming of Efuse is in progress. Once the programming is complete, LD1 LED on the board will be OFF.
10. If error occurs during Efuse programming, LD1 and LD2 LED’s on the CEC1702 Clicker board start blinking. In this case the Efuse programming operation is terminated.
11. Once the programming is over, the color of “Program EFUSE” button will turn Green.