Timing Commander Configuration EEPROM File Checksum for ClockMatrix Devices

This application note outlines how to verify that a EEPROM configuration file generated using the Timing Commander has been properly burned onto an EEPROM. Two cyclic redundancy check calculations can be checked to verify the configuration block header and payload.

For more information on Renesas' Clocking products, visit ClockMatrix Timing Solutions.

1. Introduction

The Timing Commander graphical user interface (GUI) can generate an EEPROM file of the configuration in either a .bin or .hex format. An EEPROM configuration file generated by the GUI contains two cyclic redundancy check (CRC) calculations that can be used to validate the contents of a burned EEPROM. The payload CRC and the header CRC registers can be read from the EEPROM and their contents can be compared against the generated EEPROM configuration file. If their CRC registers match, then the contents of the EEPROM have been successfully burned without error.

2. Verifying an EEPROM Configuration Block

The EEPROM configuration file generated by Timing Commander uses two cyclic redundancy check (CRC) calculations to validate the contents of the EEPROM header and payload. These CRC calculations can be used to verify that a configuration file has been properly burned onto an EEPROM.

All data in the EEPROM file stored in a larger than byte-by-byte format is in little-endian format (the least significant byte is at the lowest address). Text-based fields assume ASCII encoding of characters in a byte-by-byte format. Text strings must end with a NULL character (00).

Figure 1 shows the header and CRC bytes within the EEPROM configuration file. The header is 12 bytes long, extends from offset 0x0 to 0xB, and consists of a marker, the payload length, the payload type, the payload CRC, and the header CRC. The marker is an 8-byte string that is used to indicate the existence of a configuration block on the EEPROM. The 8 bytes of the marker will have an ASCII equivalent of "CMX-CFG".

Following the marker, there are 2 bytes for the payload length and 2 bytes for the payload type. The payload length indicates the size of the payload section in bytes. Only one payload type is currently defined (type 0), therefore, these bytes will always be 0.

The payload CRC is a 4-byte CRC calculated on the payload section. This CRC can be used to validate the payload contents on the EEPROM. Similarly, the header CRC is a 4-byte CRC that can be used to validate the contents of the header. The payload CRC and the header CRC registers can be read from the EEPROM and their contents can be compared against the generated EEPROM configuration file. If their CRC registers match, then the contents of the EEPROM have been successfully burned without error.

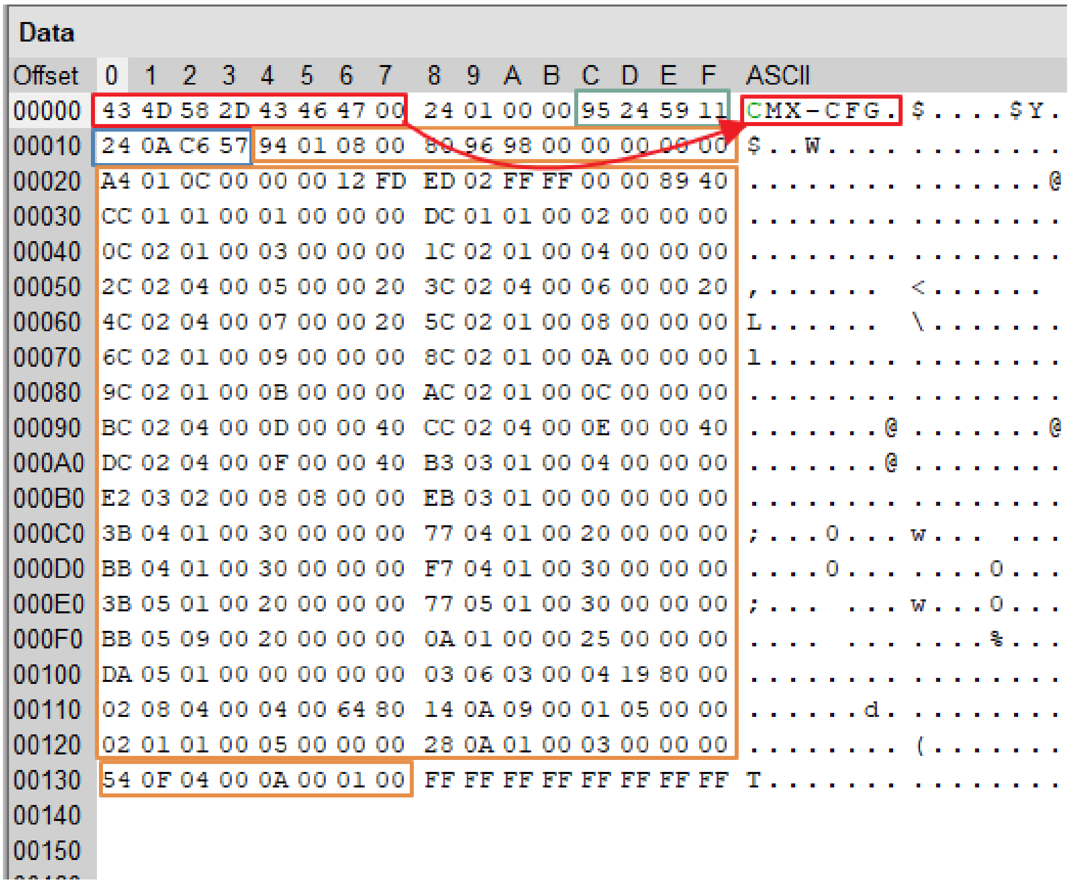
Following the header CRC, all remaining data in the configuration file comprises the payload. The payload contains the configuration settings required to program the device.

<table>
<tr><th rowspan="2">Row Address within EEPROM (Hex)</th><th colspan="16">Byte Offset within Row</th></tr>
<tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>XX00</td><td colspan="8">Marker</td><td colspan="2">Payload Length</td><td colspan="2">Payload Type</td><td colspan="4">Payload CRC</td></tr>
<tr><td>XX10</td><td colspan="4">Header CRC</td><td colspan="12">Payload</td></tr>
</table>

Figure 1. EEPROM Configuration Block Header Field Locations

Figure 2 shows an example of an EEPROM configuration file that has been opened in the Total Phase Flash Center software, so that the ASCII equivalent of each byte can be seen. This is done to easily indicate the marker. The CRC bytes and the different sections of the header have been outlined and colour coded.

- Offsets 0 to 7: This is the “CMX-CFG.” header preamble that the chip searches for within the EEPROM image. This is how it determines where to find the config. This is the red outlined data below.
- Offset 0x8 and 0x9 are a 16-bit integer indicating how big the data is (0x0124 bytes in this case).
- Offset 0xA and 0xB indicate are a type field (0x0000). Leave as 00.
- Offsets 0xC to 0xF: This is the CRC for the data.
- Offsets 0x10 to 0x13: This is the CRC for the header, so it the CRC for bytes 0 to 0x13
- The data follows the header. In the excerpt below, the data is offsets 0x14 to 0x137.



Data

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00000	43	4D	58	2D	43	46	47	00	24	01	00	00	95	24	59	11	CMX-CFG.$....$Y.
00010	24	0A	C6	57	94	01	08	00	80	96	98	00	00	00	00	00	$..W............
00020	A4	01	0C	00	00	00	12	FD	ED	02	FF	FF	00	00	89	40	...............@
00030	CC	01	01	00	01	00	00	00	DC	01	01	00	02	00	00	00	....
00040	0C	02	01	00	03	00	00	00	1C	02	01	00	04	00	00	00	....
00050	2C	02	04	00	05	00	00	20	3C	02	04	00	06	00	00	20	,...... <......
00060	4C	02	04	00	07	00	00	20	5C	02	01	00	08	00	00	00	L...... \.......
00070	6C	02	01	00	09	00	00	00	8C	02	01	00	0A	00	00	00	l...............
00080	9C	02	01	00	0B	00	00	00	AC	02	01	00	0C	00	00	00	....
00090	BC	02	04	00	0D	00	00	40	CC	02	04	00	0E	00	00	40	.......@.......@
000A0	DC	02	04	00	0F	00	00	40	B3	03	01	00	04	00	00	00	.......@........
000B0	E2	03	02	00	08	08	00	00	EB	03	01	00	00	00	00	00	....
000C0	3B	04	01	00	30	00	00	00	77	04	01	00	20	00	00	00	;...0...w... ...
000D0	BB	04	01	00	30	00	00	00	F7	04	01	00	30	00	00	00	....0.......0...
000E0	3B	05	01	00	20	00	00	00	77	05	01	00	30	00	00	00	;... ...w...0...
000F0	BB	05	09	00	20	00	00	00	0A	01	00	00	25	00	00	00	....%...
00100	DA	05	01	00	00	00	00	00	03	06	03	00	04	19	80	00	....
00110	02	08	04	00	04	00	64	80	14	0A	09	00	01	05	00	00	......d.........
00120	02	01	01	00	05	00	00	00	28	0A	01	00	03	00	00	00	........(.......
00130	54	0F	04	00	0A	00	01	00	FF	FF	FF	FF	FF	FF	FF	FF	T...............
00140																	
00150																	

Figure 2. Example of an EEPROM Configuration Block

3. Revision History

Revision	Date	Description
1.00	Nov 1, 2022	Initial release.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.