Application Note

How to Interpret UCD90XXX CSV File

Yihe Hu

ABSTRACT

The UCD90xxx family devices are flexible and powerful enough to meet sequencing and monitoring needs. This application note addresses how to interpret UCD90xxx CSV File exported from TI Fusion Digital Power Designer™ GUI. This document does not apply to the UCD9080 and UCD9081 devices.

Table of Contents

List of Figures

List of Tables

Trademarks

Fusion Digital Power Designer™ is a trademark of System Management Interface Forum, Inc.
All trademarks are the property of their respective owners.

1 Introduction

The UCD90xxx family of digital sequencers, also known as system health monitors, are flexible and powerful enough to meet user sequencing, monitoring, margining and other needs. The entire families of devices are designed to have similar behaviors, but with a different number of rails or some other minor features. Users only need to learn how to use the device once, and can then seamlessly switch to other devices within the family that best fit their future designs. This document is to help applications to interpret data flash csv file exported from TI Fusion Digital Power Designer GUI. This document does not apply to the UCD9080 and UCD9081 devices. All commands listed in the document can be found in the publications listed in Section 5.

All byte values are represented in hexadecimal format. These are the codes to understand all I^2C communications that occur:

- [S] - I^2C Start bit.
- [Sr] - I^2C Restart bit. It is identical to the Start bit.
- [Sp] - I^2C Stop bit.
- [A] - I^2C Acknowledge bit.
- [N] - I^2C No Acknowledge bit or NACK.
- [Wr] - I^2C device address with the Write bit.
- [Slave Address] - I^2C device address
- [Rd] - I^2C device address with the Read bit.
- [W:x55] - Example of a write byte for value 55 hexadecimal.
- [Data_Byte^n] - Indicates that a byte is being write or read by the I^2C master. The n subscript is an ordered integer use to distinguish multiple bytes read back.

2 Options to Generate CSV File

Fusion Digital Power Designer GUI provides multiple options when generated the data flash script file as Figure 2-1. Application can choose the best programming steps to include in script such as: DEVICE_ID verification, read back.

PEC byte is optional and when it is selected, a PEC byte is added for reach individual command.

Enable configuration security is selected if application want to enable the security feature so that any authorized writes is NACKed by the device.

when Block length bye is selected, the block length byte is added automatically into each block write command.

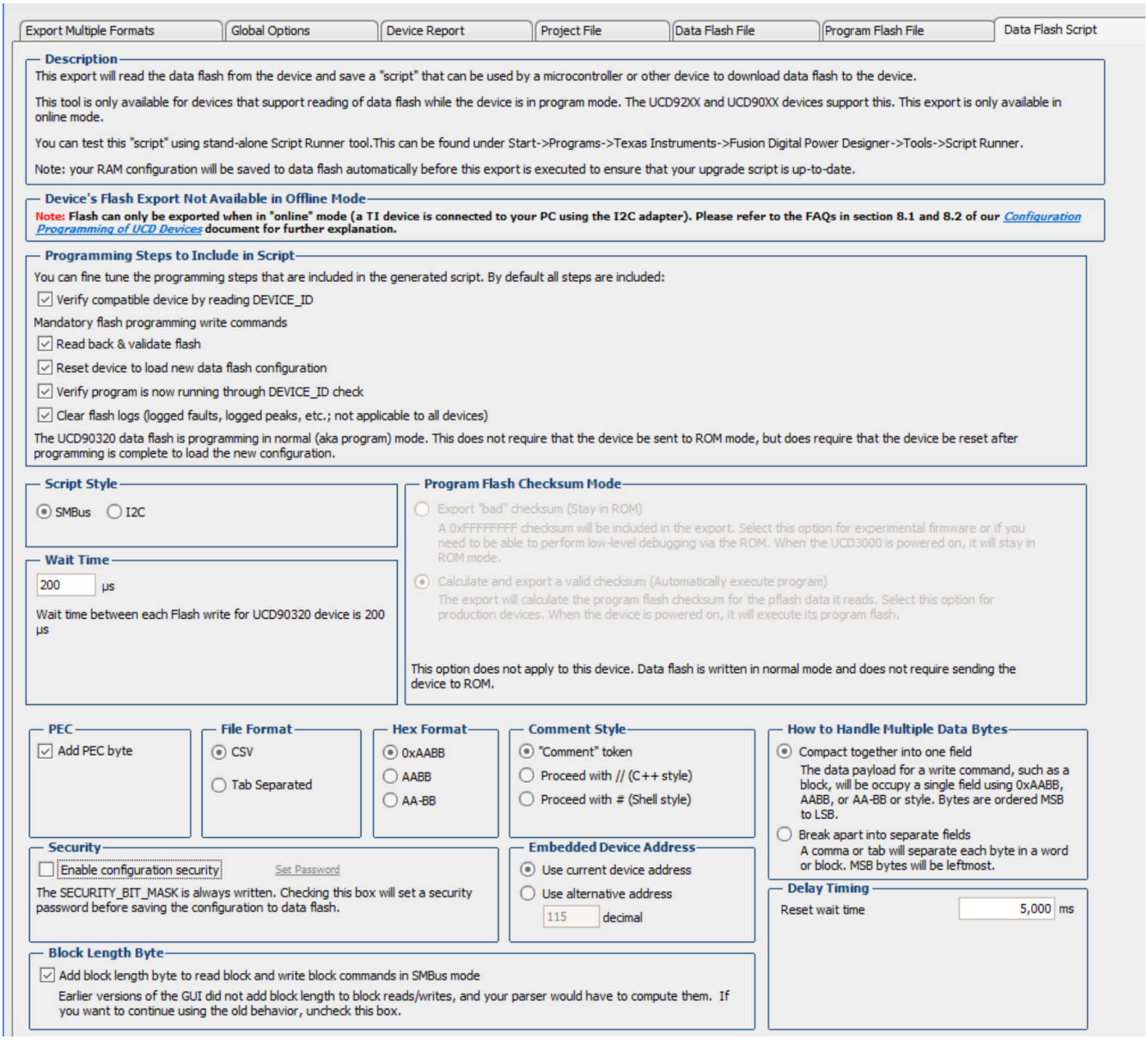

Figure 2-1. CVS File Options

3 Examples

This section demonstrates how to interpret the CSV file. The example in this dcouent is based on the below format

Script type = SMBus Format=CSV; Hex=CoderUpper; BreakOutBytes=False; PEC=True; IncludeBlockLength=True

In the csv file, all data is formatted as shown Table 3-1.

Table 3-1. CSV Example

Command Keyword	Device Address	Command Code	Data Byte(PEC is optional)
BlockWrite	0x13	0xE3	0x04F819000013

Command keyword is to tell which command protocol is used

Device Address is 7Bit(SMBus) or 8bit (I2C) Device Address.

The command code is covered in the Section 5

Date Byte is the payload data for the given command.

All lines in the csv file start with comments are comments and shall not be executed by the I2C host.

3.1 WriteByte

WriteByte is a SMBus protocol which sends the command mode followed by one byte data Figure 3-1.

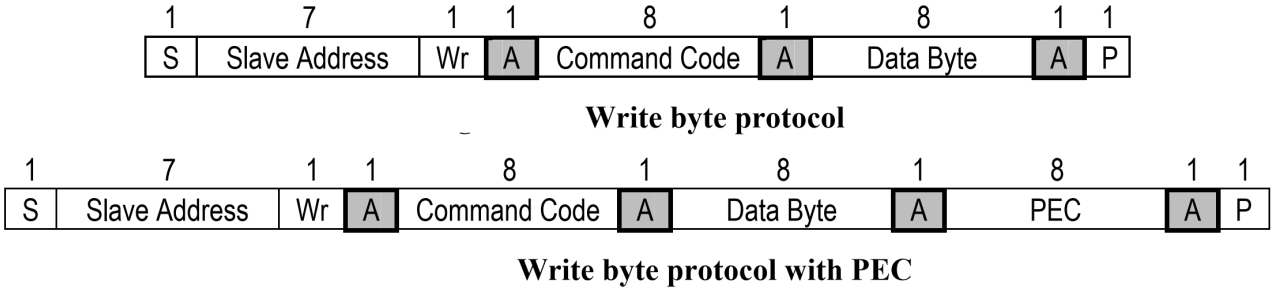

Figure 3-1. WriteByte

Table 3-2. WriteByte Example

Command Keyword	Device Address	Command Code	Data Byte(PEC is optional)
WriteByte	0x13	0x00	0xFFCD

Table 3-2 tells that the current row is a write byte protocol:

Device address is 0x13, command code is 0x00, Data is 0xFF and the last byte 0xCD is the PEC which is up to whether PEC is set or not when generating the CSV file.

3.2 BlockWrite

BlockWrite is a SMBus protocol which sends the command code followed by various length of data as shown in Figure 3-2.

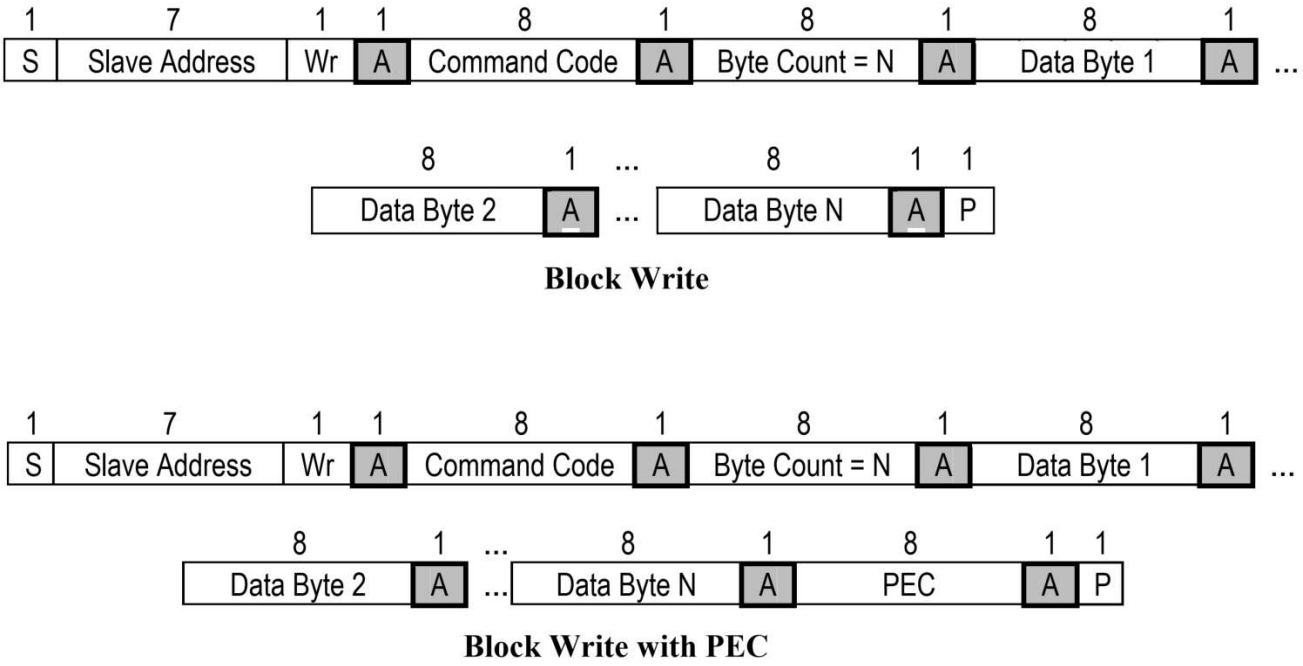

Figure 3-2. BlockWrite Protocol

Table 3-3. BlockWrite Example

Command Keyword	Device Address	Command Code	Data Byte(PEC is optional)
BlockWrite	0x13	0xE3	0x04F819000013

Table 3-3 tells that the current row is a block write protocol:

Device Address is 0x13. Command Code is 0xE3, Data Length is 4, Data is 0xF8190000 and the last byte 0x13 is PEC

Note: the data length byte is optional in the csv file which controlled by Figure 3-3

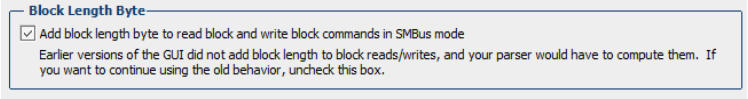

Figure 3-3. Block Length Byte Setting

3.3 BlockRead

Block Read is a SMBus protocol which read various length bytes from target device based on the command code Figure 3-4. The length of commands are listed in the document can be found in the publications listed in Section 5.

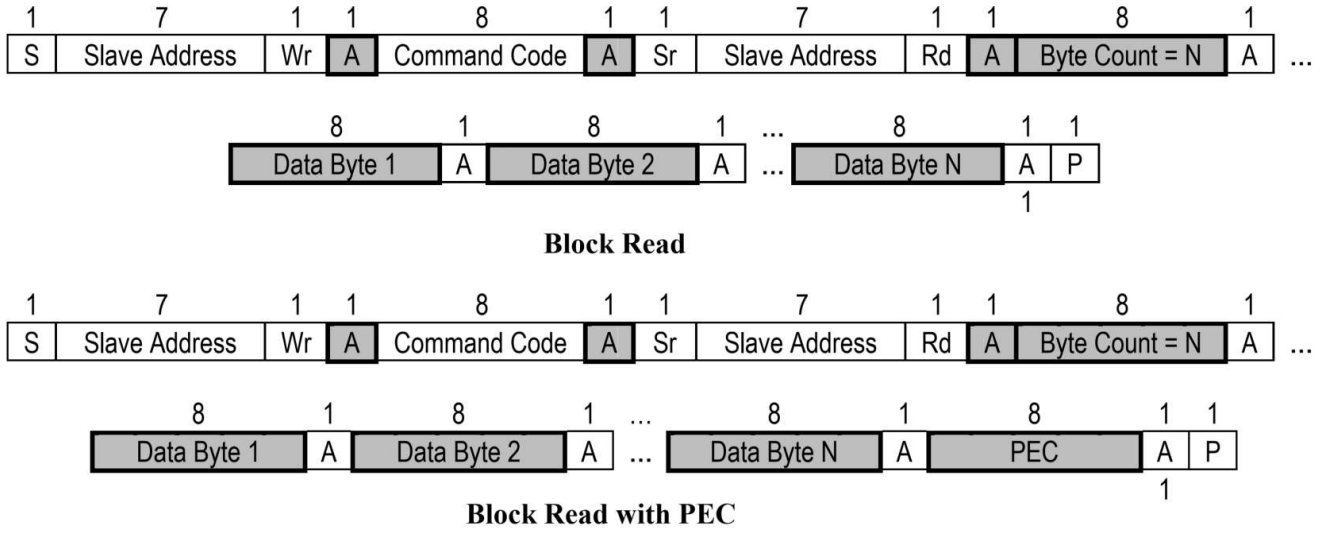

Figure 3-4. BlockRead Example

Table 3-4. BlockRead Example

Command Keyword	Device Address	Command Code	Data Byte(PEC is optional)
BlockRead	0x13	0xE3	0x10E58C0100A90B0030F8190000064D4652

Table 3-4 tells that the current row is a block read protocol

Device Address is 0x13, command code is 0xE3, Data Length is 0x10. Data is 0xE5 0x8C 0x01 0x00 0xA9 0x0B 0x00 0x30 0xF8 0x19 0x00 0x00 0x06 0x4D 0x46 and 0x52

3.4 SendByte

Sendbyte is a SMBus protocol which sends one byte only Figure 3-5.

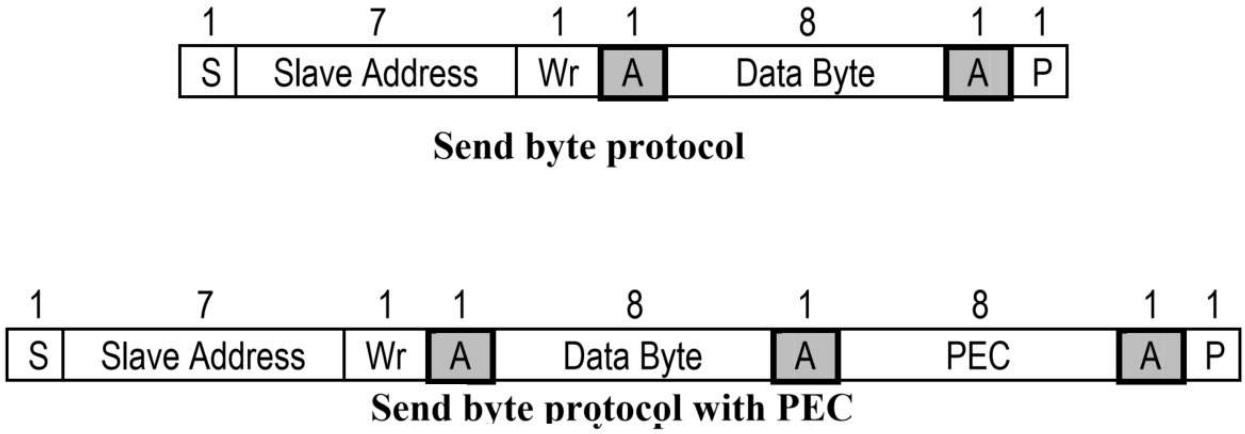

Figure 3-5. SendByte Protocol

Table 3-5. SendByte Example

Command Keyword	Device Address	Command Code	Data Byte(PEC is optional)
SendByte	0x13	0xDB	

Table 3-5 tells that the current row is a sendbyte protocol:

Device address is 0x13, Command code is 0xDB and there is no data byte.

3.5 Pause

The pause keyword is in milliseconds to pause before next command. The application is responsible to add this delay into the program

Figure 3-6 shows that 0.2ms pause is required before executed the next blockwrite command

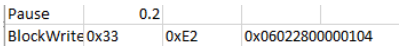

Figure 3-6. Pause Example

4 Summary

This application note outlines how to parse the data flash csv file generated from TI Digital Fusion GUI designer for UCD90xxx sequencer family device. The application engineer can follow this guide to develop software to program the UCD90xxx Sequencer with any third-party programming hardware.

5 References

- Texas Instruments, *UCD90xxx Sequencer and System Health Controller PMBus Command Reference User's Guide*.
- Texas Instruments, *UCD90320 Sequencer and System Health Controller PMBus Command Reference User's Guide*.
- Texas Instruments, *UCD90320U Sequencer and System Health Controller PMBus Command Reference User's Guide*.
- System Management Bus(SMBus) Specification 2.0.
- *The PMBus Power System Management Protocol Specification Part II - Command Language, Revision 1.1*.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265