Data I/O

UFS Programming in LumenX

Getting Started Guide

© February 2020 | Data I/O Corporation

Part Number: 983-6000-001A

[Back-side page of Front Cover, intentionally blank]

Table of Contents

Chapter 1: Introduction	1
Scope, Audience, Safety	2
Conceptual Overview	3
Chapter 2: Configure the Hardware	4
Inserting the UFS Interface Board	4
Installing a UFS Socket Adapter	10
Validating the Hardware	11
Chapter 3: Configure the Software	12
Updating LumenX Data Management Software (DMS)	12
Updating the Programmer Firmware	15
Chapter 4: Run the Job	18
Configuring Job Settings	18
Configuring Device Settings	19
Configuring LUN Settings	22
Calculating LUN Size (dNumAllocUnits)	27
Starting the Job Run	28
Chapter 5: Verify Job Completion	30
Confirming Status	30
Checking Socket Adapter LEDs	31
Checking for Alerts	32
Appendices	33
Appendix A: Job Settings, Job Process Selection	34
Appendix B: Device Settings, UFS Configuration Descriptors	35
Appendix C: Device Settings, UFS Device Attributes	36
Appendix D: LUN Settings, UFS Unit Descriptors	37
Appendix E: Sample UFS Job with Default Settings	39
Appendix F: User Interface Legend	41
Index	42

Revision History

Comments	Date	Author
Initial 8-Socket Draft	October 15, 2019	Anthony Rosensprung
Added recommended boards table pg. 6	November 15, 2019	Ben Deagan

Chapter 1: Introduction

This document provides instructions for programming a Universal Flash Storage (UFS) device using Lumen[®]X technology, including:

- Hardware installation of UFS Interface board
- Hardware installation of UFS socket adapter
- Software configuration of UFS programming job
- Verification of job completion

Simplified Ease-of-Use

To maintain consistency and simplify ease-of-use, LumenX streamlines the process of programming a Universal Flash Storage (UFS) device to be the same as programming a standard e-MMC device:

- Select the target device to be programmed
- Select the programming algorithm
- Configure job settings (ex. Load the data/img files to write...)
- Run the job (ex. Program, Verify...)

Benefits

The UFS programming process in LumenX Data Management Software (DMS) involves no learning curve for existing LumenX users who want to create and run UFS jobs, negating the need for training and transition. It also introduces new UFS users to the proven LumenX platform.

The same LumenX programming hardware used today is capable of programming and outputting an entirely new class of flash memory (UFS) without sacrificing any existing capabilities.

Document Scope

This document does NOT cover how to configure the hardware and software for running LumenX programming jobs (see the LumenX Getting Started Guide); this document focuses on the UFS-specific steps.

Intended Audience

Readers need the LumenX Getting Started Guide and some familiarity with:

Configuring the LumenX programmer hardware and networking

Running jobs in LumenX Data Management Software (DMS)

It is important that readers have a basic understanding of UFS programming.

Safety Precautions

To prevent personal injury, lost time, and damage to equipment, please use extra caution when handling the powered programming equipment.



CAUTION: Electrostatic Discharge Hazard!

Electrostatic discharge (ESD) may damage equipment and integrated circuits. Always discharge static electricity to a common ground. Use ESD prevention devices that contain a 1 M-ohm to 10 M-ohm current-limiting resistor.



WARNING: Electric Shock Hazard!

Injury or death may result from contact to parts inside the programmer. Do not remove covers. There are no user-serviceable parts.



Excessive Socket Actuator Air Pressure

For automated systems (ex. PSV7000), decrease the socket actuator air pressure to prolong socket adapter(s) longevity. For pressure recommendations, see page 6.

Refer to Chapter 4 of the PSV7000 Owner's Manual for instructions: near the lower-left corner of the Power Panel, pull the collar on the black "Socket Opener Pressure Control" knob out, then rotate it counter-clockwise to decrease pressure. Ignore this note for Manual/Desktop programmers.

Conceptual Overview

To simplify programming and enable customers with UFS capability on their existing LumenX programmers, Data I/O built support for UFS by leveraging the existing LumenX programming model. This development extends the value of LumenX programmers because customers can use existing technology to program the newest class of flash memory devices.

The following diagram shows the key difference between standard LumenX and UFS programming models (note the **UFS Interface Board** highlighted in Red color).



The left side of the diagram above shows a standard LumenX programming job with a socket adapter/board that plugs into the programmer. The right side shows a LumenX programming job for UFS where an additional UFS Interface Board is introduced.

From bottom to top, the general process is the same from programmer to device. But with the UFS Interface Board, the resulting device is programmed with superior UFS capabilities.

* Note that UFS support in LumenX requires the following minimum software versions:

- LumenX Data Management Software (DMS) version 1.7.1+.
- Automated Handler (AH700) version 2.9.0+ for PSV7000 automated systems
- CH700 version 2.9+ for PSV5000 automated systems

Chapter 2: Configure the Hardware

This Chapter provides instructions on configuring the UFS hardware, specifically:

UFS Interface Board insertion UFS socket adapter installation Hardware validation

Installing UFS hardware is similar to installing LumenX hardware. First, follow the LumenX Getting Started Guide to ensure network connectivity between the programmer and Host PC. Then insert the UFS Interface Board into the programmer.

Inserting the UFS Interface Board

- 1. [PSV7000 Only] If it is running, close AH software on the Host PC.
- 2. Open the socket clamp on top of the programmer.



3. Unwrap the UFS Interface Board(s) from the packaging and handle carefully.





Electrostatic Discharge Hazard! Electrostatic discharge (ESD) may damage equipment and integrated circuits. Use ESD prevention devices that contain a 1 M-ohm to 10 M-ohm current-limiting resistor.

4. Rotate the orientation of the UFS Interface Board such that the notched/chamfered corners match the notched corners in the programmer.



CAUTION: Possible machine damage! Do not touch connector pins. Bent or damaged pins can cause programming malfunctions and/or reduced production yields.

5. Close the socket clamp after installing the UFS Interface Board.

Data j 🗸

Guidelines for UFS Socket Loading

Do NOT use 1 Interface Board alone because doing so introduces an imbalance in the mechanical loading of the socket actuator (resulting in compromised socket adapter longevity). Distribute the actuator load as evenly as possible across all inserted Interface Boards. To decrease socket opener pressure, refer to the bottom of page 2.

Excessive Socket Actuator Air Pressure

For automated systems (ex. PSV7000), decrease the socket actuator air pressure to prolong socket adapter(s) longevity.

If 8 UFS Interface Boards are not available, then use at least 2 boards in the same actuator position/orientation: insert the 2 boards such that they occupy sockets 1 and 4 on the programmer (or sockets 5 and 8). Placing 2 UFS Interface Boards on the same side of the programmer provides equal mechanical loading of the socket actuator.

Boards	Support	Recommended Placement	Socket Opener Pressure
1	X	Not Supported	N/A
2	✓	Load 2 sockets in the same actuator row : Sockets 1 and 4 (or Sockets 2 and 3), OR Sockets 5 and 8 (or Sockets 6 and 7)	0.25 - 0.30 MPa (MegaPascals)
3 or 4	✓	Load any 3 or 4 sockets in the same row .	0.27 - 0.32 MPa
5	✓	Load any 3 sockets in the same actuator row, then load 2 sockets in other row (see recommendation for 2 boards above).	0.30 - 0.35 MPa
6	*	Load any 3 sockets in one actuator row, then load any 3 sockets in other row.	0.32 - 0.37 MPa
7	*	Load all 4 sockets in one actuator row, then load any 3 sockets in the other actuator row.	0.32 - 0.37 MPa
8	*	[Recommended] All sockets loaded.	0.35 - 0.40 MPa

The following table summarizes the board loading/socket placement recommendations.

The following pages show some examples of proper and improper UFS socket loading, based on the number of UFS Interface Boards available. After loading the boards properly, verify that all sockets open when actuated. Else increase socket opener air pressure (to the next interval shown in the chart above) until all sockets open fully when actuated.







With the UFS Interface Board installed, your LumenX hardware is almost ready for UFS programming. To program a UFS device, you install a device-specific UFS socket adapter on top of the UFS Interface Board (the same as installing a standard LumenX socket adapter).

Data j 🔊

Installing a UFS Socket Adapter

1. Remove the UFS socket adapter from the packaging and note the <u>PIN1</u> location.



Electrostatic Discharge Hazard!

Electrostatic discharge (ESD) may damage equipment and integrated circuits. Use ESD prevention devices that contain a 1 M-ohm to 10 M-ohm current-limiting resistor.

2. Rotate the orientation of the socket adapter such that the notched/chamfered corners match the notched corner shape printed on the UFS Interface Board.

NOTE: The <u>PIN1</u> writing on the socket adapter reads in the same direction as "Data I/O" on the UFS Interface Board.

3. Gently press down on the socket adapter until it fits into the UFS Interface Board.



CAUTION: Possible damage may occur if the socket adapter is not rotated to the proper <u>PIN1</u> orientation and/or excessive force is applied during insertion.

Validating the Hardware

Skip this section if using an automated system (such as a PSV7000). Otherwise, the UFS hardware configuration is complete. Before starting software configuration in the next Chapter, now is a good time to double-check that:

The LumenX programmer is connected by network cable to a LumenX Host PC The LumenX programmer is powered on The LumenX programmer is reachable from the Host PC by: *Ping* command to the IP address of the programmer Programmer status in LumenX DMS (see next Chapter)

Note: Programming file sizes greater than 64GB requires a LumenX programmer upgrade to 128GB or 256GB cache memory.

For PSV7000 systems, always close AH software before changing UFS adapters. Else, you may need to restart LumenX DMS and/or the LumenX programmer(s).

Chapter 3: Configure the Software

This Chapter provides instructions on configuring the software for UFS programming, specifically:

LumenX Data Management Software (DMS) update

LumenX programmer firmware update

Software configuration for UFS programming consists of updating the version of LumenX Data Management Software on the Host PC, and then updating the firmware on the LumenX programmer(s).

Updating LumenX Data Management Software (DMS)

- 1. Download and run the LumenX_DataManagementSoftware.exe file.
- 2. In the LumenX Setup Wizard, follow the prompts to complete the version update.



The latest version of LumenX Data Management Software (DMS) is always available for download at <u>http://www.dataio.com/Technology/</u> LumenX/LumenX-Release

3. Restart the Host PC.

To complete the installation, start LumenX DMS (click Start > Programs > Data IO > Data Management Software), and from the Tools menu at the top, click Algorithm Updater.

🗯 LumenX Data Management Software (DM	5 5 5		×	
File View Alerts Settings To	ools Help Algorithm Undater	He	llo, Loca	0
Jobs Job Descriptions User A Jobs Enter search text here (e.g.	Read from Master Device Retrieve all logs			<u>م</u>
Job Name Operato Pa	Update Programmer(s) Extensions		Package	

5. If prompted, click **Install** to install the AlgoUpdater.



6. In the lower-right corner of the algorithms dialog box, click **Update List**.

у́х	- 🗆 X
<u>F</u> ile View <u>W</u> indow <u>H</u> elp	
Local Algorithms	Remote Algorithms
Enter search text here (e.g. Intel BGA)	Enter search text here (e.g. Intel BGA)
Manufactur Part Nam Packag Versio Statu Keyword Strin Nam	Manufactur Part Nan Packag Versio Statu Keyword Str
(Update List
	Download Selected
Warning: there are no algorithms in the current algorithm dire	ctory, go to Settings -> Settings for more inform:

 After the algorithm list updates, under the **Remote Algorithms** column on the right, check the box in the upper-left corner (to select all algorithms), and then click **Download Selected**.

<i>L</i> ∕X			-	σ×
<u>F</u> ile View <u>W</u> indow <u>H</u> elp				
Local Algorithms	Remo	te Algorithms		
Enter search text here (e.g. Intel BGA)	Em	ter search text	t here (e.g. Intel BGA)	
Manufactur Part Nam Packag Versio Statu Keyword Strin Nam		Manufactur	Part Name	Package 🚺
	☑	Micron	MTFC32GAKAEJP-AIT	VFBGA153
		Micron	MTFC64GAPALBH-AAT ES	TFBGA153
		Micron	MTFC128GAPALNS-AAT ES	TFBGA153
		Toshiba	THGBMHG9C8LBAW8	TFBGA153
	~	SanDisk	SDINBDA4-64G	TFBGA153
	~	SanDisk	SDINBDA4-32G	TFBGA153
		Micron	MTFC16GAKAEJP-AIT	VFBGA153
	✓	Micron	MTFC128GAJAECE-5M AIT	LFBGA169
		Micron	MTFC32GAKAEEF-AAT	TFBGA169
		_	Update List	
	C		Download Selected	
Updating finished				

8. After the update completes, verify that the **Local Algorithms** column on the left is now populated, and then close the algorithms dialog box. effect

JX							_		×
<u>F</u> ile View	<u>W</u> indow <u>H</u> elp								
Local Algorith	nms				Remo	te Algorithms			
Enter search text here (e.g. Intel BGA)						er search text	t here (e.g. Intel BGA)		
Manufactur	Part Name	Package	Versio	Sta 🔺	V	Manufactur	Part Name	Package	
Micron	MTFC32GAKAEJP-AIT	VFBGA153	9.3.0	Rele	\checkmark	Micron	MTFC32GAKAEJP-AIT	VFBGA1	53 🗍
Micron	MTFC64GAPALBH-AA	TFBGA153	9.3.0	Rele	\checkmark	Micron	MTFC64GAPALBH-AAT ES	TFBGA15	3
Micron	MTFC128GAPALNS-A	TFBGA153	9.3.0	Rele	$\overline{}$	Micron	MTFC128GAPALNS-AAT ES	TFBGA15	3
Toshiba	THGBMHG9C8LBAW8	TFBGA153	9.4.0	Rele	7	Toshiba	THGBMHG9C8I BAW8	TFBGA15	3
loshiba	THGBMHG9C8LBAU8	TFBGA153	9.4.0	Rele	Ĩ.,	SanDick		TERCAT	2
Ioshiba CDi-l-		TEBCA453	9.4.0	Rele	<u> </u>	Sandisk		TEDGAL	5
SanDisk	SDINBDA4-64G	TEBGA 153	9.3.0	Kele	~	SanDisk	SDINBDA4-32G	TFBGA15	ച
SanDisk	SDINBDA4-32G	TEBGA153	9.3.0	Rele	\checkmark	Micron	MTFC16GAKAEJP-AIT	VFBGA15	53
Micron	MIFC16GAKAEJP-AIT	VFBGA153	9.3.0	Rele		Micron	MTFC128GAJAECE-5M AIT	LFBGA16	9
Micron		LFBGA169	9.3.0	Rele	v	Micron	MTFC32GAKAEEF-AAT	TFBGA16	59 🔽
Milloron		TERISATIN	чкп						
							Update List		
						_	Download Selected		
Algorithm li	ist updated using direct	tory: C:\Lum	enX\Alg	orithms	:				

9. Close and restart LumenX Data Management Software (DMS) to recognize the algorithms.

Updating the Programmer Firmware

This example shows updating to version 1.5.1, but you should update to the latest version.

1. From the **Tools** menu at the top, select **Update Programmer(s)**.

🞉 LumenX Data Management Software (DMS)			- 🗆	×
File View Alerts Settings	Tools Help			Hello, Loca	0
Jobs User Authentication	Algorithm Updater				
	Read from Master Device				
Jobs Enter search text here (e.g	Retrieve all logs				Q
Lab Nama	Update Programmer(s)	Nister	Daut Manag	Manufashuran	
	Extensions >	Notes		Manufacturer	
Secure Job 123 for Infineon Optiga	hust-e Here are some optional no	otes for the Operator(s)	SLS32AIA020A	Infineon lechnol	og. l
TapeLabelTest0	Ensure that Host PC sees t	the Label Printer first	KLMAG1JETD-B041	Samsung	F
L					

2. In the left pane, check the box(es) for the programmer(s) you want to update.



3. To the right of the **Select from path** box, click the ellipsis (...) button to Browse for an update package to apply.



 Browse for and select the System Update package to apply (<u>System Update package</u> <u>must be applied BEFORE Firmware Update package</u>), and then click **Update Programmer(s)**.



- 5. In the left pane, verify that the **Status** of each selected programmer changes state from:
- Idle > Updating > Verifying > Rebooting > Idle
- 6. Repeat the previous steps to select and apply a Firmware Update package.



7. In the left pane under **Programmers**, verify that the **Firmware** version is updated (you can also hover/mouseover **Firmware** to reveal the System version).



Chapter 4: Run the Job

This Chapter provides instructions on running the UFS job, specifically:

- Job settings configuration
- Device settings configuration
- UFS Descriptors and Attributes configuration
- LUN settings configuration

With the software and firmware updated in the last Chapter, now configure UFS settings in LumenX Data Management Software (DMS) for the programming job.

Configuring Job Settings

- 1. On the Host PC, start LumenX Data Management Software (DMS).
- 2. Near the lower-right corner, click **New** to create a job.
- 3. On the **Devices** screen, search for "ufs", select your target UFS device, and click Next.
- 4. On the Algorithms screen, select the desired algorithm to use, and click Next.
- 5. On the **Create Job** screen, in the **Job Name** box, type a name for the job.
- 6. In the Job section, make the desired job process selections from the drop-down lists.

ا 🍂	LumenX Da	ta Manager	ment Software	(DMS)							-		\times
File	View	Alerts	Settings	Tools	Help						He	llo, Loca	al⊚
CRE	ATE JOB:	Samsung	g Generic UF	S 2.0 Alg	orithm TFBGA	153							
Jot	b Name		ner 26 UFS Jo	ob XYZ									
De	scription			z run 4 fo									
Ор	perator No	otes No n						ut befo					
							_		 	 _	_		
01		Proce	ess:										
	<u> </u>		Name		Value					 			
1	JOB	Job Pro	cess Selectio	on: Prog	gramming and	Verify '	·]		 	 			
		Erase S	election:	Full	chip Erase		_						

Configuring Device Settings

1. In the **Device** section, from the **Adapter Id** drop-down list, select the specific adapter.

<u>J</u> X	LumenX Da	ta M	anagement Software (DN	1S)					_		×
File	View	Al	erts Settings T	ools Help					Hel	lo, Loca	0
CRE	ATE JOB:	Sa	msung Generic UFS 2	.0 Algorithm TFBGA15	3						
ol	b Name		Partner 26 UFS Job	KYZ							
De	escription		This is ufs job xyz ru	n 4 for Partner 26							
o	perator No	otes	No need to re-teach	n pkg; Test tray feeder (and tape	-out before star	ting run				╟
01	01010		Process:						_		Ĭ.
·	JOB	5	Name	Value							
			D Process Selection.	Programming and v	erny 🔻						
		E	rase Selection:	Full chip Erase	•						
		evico Igori dapt) Nc	e: Samsung Gener thm: LUN 0 to 7 Vers er Id: 110103 ▼ otes: No notes.	ic UFS 2.0 Algorithm T ion: 0.0.0	FBGA15:	3					
	SUMM (go ba To crea	IAR i ck t ate a	XY: Edit job, then S to job list to run th nd test a job:	<i>Save</i> job to validate e new or changed jo	it and ob).	save it to disk	•	Back	Sa	핟 ve as Jo	b

2. To expand the device-specific options, click Settings.



Note: The settings and their default values may vary slightly depending on the specific device and algorithm selected.

3. To expand the device-level UFS Configuration Descriptors, click **UFS Descriptor**.



4. Configure the UFS Descriptors as desired.



NOTE: LumenX Data Management Software (DMS) supports the standard set of descriptors defined by the JEDEC specification for UFS Version 2.1 (JESD220C, March 2016). For more information, see **Appendix B: Device Settings, UFS Configuration Descriptors**.

5. To set the UFS Device Attributes, expand the desired attribute and set appropriately.



LumenX supports a standard set of device attributes. For more information about these attributes, see **Appendix C: Device Settings, UFS Device Attributes**.

Configuring LUN Settings

1. In the **Memory Region** section, click **Add Static Data** for the specific Logical Unit Number (LUN) to program (ex. LUN 0). The number of LUNs is dependent on the specific device.

🎉 Lumen)	🛱 LumenX Data Management Software (DMS) - 🗆 X									
File Vie	w Alerts Settings To	ools Help			Hello, L	ocal 🛛				
CREATE JO	DB: Samsung Generic UFS 2.	0 Algorithm TFBGA153								
DEVICE	Device: Samsung Generi Algorithm: LUN 0 to 7 Version Adapter Id: 110103 ♥ Notes: No notes. ♥ Settings: 1 settings, 0 e	c UFS 2.0 Algorithm TFBGA on: 0.0.0 rrors	153							
	LUN0 Memory Region (29.8	GB - 0x773C00000)	Add St	tatic Data Add Se	erialized D	ata				
MEMORY 1010 1010	Data	Checksum (hex)	Start-End (he	Process	Edit	Del				
REGION	Serialized Data Source	Start With	Start-End (he	Process	Edit	Del				
	Settings: 2 settings , 0 e	rrors								

2. In the **LumenX Edit Memory Region Information** dialog box, click **Browse** to load your data file.

L	t Lume	enX Edit Memo	ry Range Information				1		×
٢		File						Brov	vse
			File Size						
ŀ	001011		Load Start Address (hex)		Load Range (hex)				
ľ	DATA	Checksum	Method	32 Bit CRC	₹				
			Result (hex)					Calc	ulate
		Device	Start Offset (hex)						
HELP	SUMMARY: If in edit mode, enter or select your image file, check CRC to confirm image, select the part of the file to load, and select the location in the device to place the image. Device Regions and Ranges A region on a device is a main area to be programmed. It may be the whole device or it may be a layer or it may be a sub-area partitioned by the device (and its controlling algorithm) or it may be a special area with configuration data or a OTP area. Within this region the job creator can place image data in some or all of it.							k Car) ncel

3. (Optional) To program only a specific segment of the loaded data/image file into the selected LUN, specify the starting address and data range.

🔏 Lur	nenX Edit Memo	ry Range Information				-	
ſ	File	C:\Users\Desktop\IMG\Image123.img					Browse
		File Size	26.2 KB	Warning: file will b boundary!			
100101		Load Start Address (hex)	0	Load Range (hex)	68A8		
DATA	Checksum	Method	32 Bit CRC	¥			
		Result (hex)					Calculate
	Device	Start Offset (hex)	0				
HELP	SUMMARY: I select the par Device Region A region on a be a layer or i may be a spec can place ima Set or Change To add or cha	f in edit mode, enter or a rt of the file to load, and as and Ranges device is a main area to l t may be a sub-area partit cial area with configuration ge data in some or all of it a Image File Path for the Ra ange the nath to the image	select your image select the locatio pe programmed. I ioned by the device a data or a OTP are ange	file, check CRC to confirm n in the device to place the t may be the whole device o e (and its controlling algorith a. Within this region the job	image, image. or it may nm) or it o creator	П ОК	9 Cancel
Image	file C:\Users\	Desktop\IMG\Image123.in	ng added.				

By default, LumenX writes the entire loaded data/image file starting at address 0x00 (and automatically calculates/populates the **Load Range** based on file size).

4. (Optional) To validate programmed data at the file level (checksum compare), select the desired **Checksum Method**, and then click **Calculate**.

🛵 Li	umenX Edit Memo	ry Range Information						×
Г	File	C:\Users\Desktop\IMG\Image123.img					Brow	se
		File Size	26.2 KB					
10010	11	Load Start Address (hex)	0	Load Range (hex)	7000			
DAT	A Checksum	Method	SHA256	7				
	Result (hex)		D133E9289B9B8963	6FC4EB69C27A5BC33A3FF	7F0CAC485	CF73572D9852	Calcu	late
	Device	Start Offset (hex)	0					
HELP	SUMMARY: I select the part Device Region A region on a be a layer or i may be a spec can place ima Set or Change To add or cha	f in edit mode, enter or a rt of the file to load, and as and Ranges device is a main area to b t may be a sub-area partit cial area with configuration ge data in some or all of it a mage File Path for the Ra	select your image fi select the location i be programmed. It n ioned by the device (n data or a OTP area. - ange a you want to import	le, check CRC to confirm in the device to place the nay be the whole device o and its controlling algorith Within this region the job	image, image. r it may im) or it creator	нок	S Can) cel
CRC	calculation finis	hed						

5. (Optional) To program data at a specific starting address on the UFS device, complete the **Device Offset** field. (By default, LumenX writes to the lowest available address on device.)

🍂 Lum	enX Edit Memo	ory Range Information			-		×
\square	File	C:\Users\Desktop\IMG\Ir	nage123.img		Brow	vse	
		File Size	26.2 KB				
1001011		Load Start Address (hex)	0	Load Range (hex) 7000			
DATA	Checksum	Method	SHA256	· · · · · · · · · · · · · · · · · · ·		-	
		Result (hex)	D133E9289B9B89636FC4EB69C27A5BC33A3FF7F0CAC485CF73572D98		85CF73572D9852	Calc	ulate
	Device	Start Offset (hex)	0				
HELP	 The offse The file a The start to have t The load file size. The targe device. 	ets, lengths and checksums and device data locations w t address in the file will de this range start at a differe range or range of the file Enter a different size if de et location on the device w Enter a different starting ad	s on this dialog are vill default as the in fault to 0 or the sta nt location in this f to be programme sired. vill default to 0 or ddress to start at in	in hexidecimal. mage file is loaded. Int of the file. Enter a different offset ile. d in the device will default to the full the start of the current region of the this region if desired.	H OK	Car	2 ncel
CRC ca	Iculation finis	shed					

- 6. To include additional data in the LUN, repeat this procedure starting from Step 1.
- 7. To include dynamic/serialized data in the programming job, click **Add Serialized Data**.

🥼 Lumen	X Data Management Software (DN	VIS)			-		×
<u>File V</u> ie	w <u>A</u> lerts <u>S</u> ettings <u>T</u>	íools <u>H</u> elp			Hel	o, Loc	al 🛛
CREATE JO	OB: Samsung Generic UFS 2	.0 Algorithm TFBGA153					
	Device: Samsung Gener Algorithm: UFS 2.0 Algorith	ric UFS 2.0 Algorithm TFBGA153 nm Version: 0.1.1					٦*
DEVICE	Adapter Id: 110103 🔻						
	Notes: No notes.						
	Settings: 18 settings, 0) errors					
	LUN0 Memory Region (29.	8 GB - 0x773C00000)		Add Static Data	Add Serialize	d Dat	
HEHODY	Data	Checksum (hex)	Start-End (hex	Process	E	dit D	el
1010 1010 1011 1011	Image123.img		0 - 6FFF	👽 Program 🔍 V	erify		×
REGION	Serialized Data Source	Start With	Start-End (hex	Process	E	dit D	el
	Settings: 7 settings , 0	errors					
	LUN1 Memory Region (29.	8 GB - 0x773C00000)		Add Static Data	Add Serialize	d Dat	a)
MEMORY	Data	Checksum (hex)	Start-End (hex	Process	E	dit D	el
REGION	Serialized Data Source	Start With	Start-End (hex	Process	E	dit D	el

8. In the **LumenX Edit Serial Data Information** dialog box, complete the desired fields, and then click **OK**.

<u>L</u>	LumenX E	dit Serial Data Informat	tion				-		×
ſ		Serialized Data	Template						
	1001011		Source	Incrementing counte	r			•	
s	ERIALIZED		Start Offset (hex)	0	Data Size (bytes) 8				
	DATA	Serial Data File						Brov	vse
L			Edit Starting Number						
HELP	SUMM Serial • F • T Increr • ¢ e	MARY: Add serializ ized Data Range p lange must be eithe wo or more serializ menting Counter S A self-incrementing endian format.	ed data range to dev placement restrictions er entirely within a stat red data ranges cannot Source: 9 64-bit unsigned inte	ice region ic data range or entire : overlap each other. ger is used. The data	ly outside. a is written in little-] (в	Cat	2 ncel

9. To configure LUN-specific Unit Descriptors, click **Settings** to expand the available options.

🦛 Lumen	X Data Management Software (DMS)					x c
File ⊻ie	w <u>A</u> lerts <u>S</u> ettings <u>T</u> ool	s <u>H</u> elp			Hello, I	.ocal ⊚
CREATE J	OB: Samsung Generic UFS 2.0 A	Algorithm TFBGA153				
TITITIT -	Device: Samsung Generic U Algorithm: UFS 2.0 Algorithm	JFS 2.0 Algorithm TFBGA153 Version: 0.1.1				
DEVICE	Adapter Id: 110103 🔻					
	Notes: No notes.					
	 Settings: 18 settings, 0 en 	rors				
	LUN0 Memory Region (29.8 G	B - 0x773C00000)		Add Static Data Add S	Serialized [Data
	Data	Checksum (hex)	Start-End (hex	Process	Edit	Del
MEMORY 1010 1010	Image123.img		0 - 6FFF	🗸 Program 🔍 Verify		×
REGION	Serialized Data Source	Start With	Start-End (hex	Process	Edit	Del
	Incrementing counter		0 - 7	🗸 Program 🕑 Verify		×
	🕑 Settings: 7 settings , 0 erro	Irs				
	LUN1 Memory Region (29.8 G	B - 0x773C00000)		Add Static Data Add S	Serialized [Data
MEMORY	Data	Checksum (hex)	Start-End (hex	Process	Edit	Del
					L matter	

10. Configure the Unit Descriptors as desired.

🧯 Lumen	X Data Management Software (DMS)			- 🗆 X
<u>F</u> ile <u>V</u> ie	w <u>A</u> lerts <u>S</u> ettings <u>T</u> ools <u>H</u> elp			Hello, Local ⊚
CREATE J	OB: Samsung Generic UFS 2.0 Algorithm TFBGA153			
	Settings:			1
	Name	Value	Туре	Erro
MEMORY	bBootLunID - Boot LUN Identifier	0x00: Not a Boot Logical Unit (defau	ult) 🔻 🔪	
1010-1010 1011-1011	bLUWriteProtect - Logical Unit Write Protect	0x00: Not Write-Protected (default)		
REGION	bMemoryType - Logical Unit Memory Type	0x00: Normal Memory (default)	▼	
	bDataReliability - Enable the Data Reliability feature			
	bLogicalBlockSize - Size of addressable logical blocks	C	Hex	
	dNumAllocUnits - Size of LUN in allocation units	0	Hex	
	bProvisioningType - Logical Unit Provisioning Type	0x00: Thin Provisioning disabled (de	efault) 🔻 🌙	
	Help Text for: bBootLunID - Boot LUN Identifier Selection for bBootLunID - Marks whether this LUN is bo	ootable, and if so, with which ID		
MEMORY	LUN1 Memory Region (29.8 GB - 0x773C00000)	Add St.	atic Data Add	Serialized Data
MENIORY	Data Checksum (hex) Start-End (hex)	Process	Fdit Del
SU list To	MMARY: Edit job, the <i>n Save</i> job to validate it and t to run the new or changed job). create and test a job:	l save it to disk (go back to job	Back	Save as Job

LumenX supports the standard set of LUN descriptors defined by the JEDEC specification for UFS Version 2.1 (JESD220C, March 2016). For more information about these descriptors, see **Appendix D: LUN Settings, UFS Unit Descriptors**.

Note: At this point (before starting the job run), double-check that the air pressure for socket actuators is decreased (see page 6). Else, excessive socket actuator air pressure can compromise socket adapter(s) longevity.



Calculating LUN Size (dNumAllocUnits)

The **dNumAllocUnits** setting (as highlighted by arrow in the preceding screenshot) is unique in that it is <u>REQUIRED</u> (you cannot save UFS jobs if **dNumAllocUnits** is blank or otherwise invalid).

dNumAllocUnits simply specifies the desired size of the LUN (similar to partitioning any storage drive), so the size you specify must be equal to or greater than the sum of all the data/image files that you intend to program into the specific LUN/memory region. Also include additional storage as a buffer if your application involves logging (allocate extra space for the log files).

The JEDEC formula for calculating **dNumAllocUnits** is:

 $dNumAllocUnits (hex) = \frac{\text{Desired LUN Size (bytes, decimal) x CapacityAdjFactor}}{bAllocationUnitSize (bytes, decimal) x dSegmentSize x 512}$

For the numerator, you set the **desired LUN size** while the **CapacityAdjFactor** is always 1 for Normal memory type. For Enhanced memory types, see the JEDEC specification on how to calculate the **CapacityAdjFactor**.

For the denominator, both of the variables are fixed per device specifications. So calculating **dNumAllocUnits** is relatively easy because 3 of the 4 variables are fixed and you set the 4th. For example, <u>if 3 GB is the desired LUN size</u> and your device has the following specifications:

- CapacityAdjFactor = 1
- bAllocationUnitSize = 0x01 (Hex)
- dSegmentSize = 0x00002000 (Hex)

Then calculate **dNumAllocUnits** as follows:

Step	Calculation
 Determine the desired LUN size, then convert this value to <u>bytes</u>. 	3 GB = <u>3,000,000,000</u>
 In the specifications for your device, calculate the CapacityAdjFactor for your device's memory type. (The value is 1 for Normal memory type.) 	Hex 0x01 = Decimal <u>1</u>
3. In your device specifications, lookup the bAllocationUnitSize and dSegmentSize values, then convert them from hexadecimal to decimal.	bAllocationUnitSize= <u>1</u> dSegmentSize= <u>8192</u>
4. Using the formula above for calculating dNumAllocUnits, plug the values from the first 3 steps into the formula.	<u>3,000,000,000 x 1</u> 1 x 8192 x 512
5. Convert the quotient in Step 4 from decimal to <u>hexadecimal</u> , and enter this hex value in the dNumAllocUnits box.	Decimal 715 = Hex 2CC

Starting the Job Run

- 1. Gently insert and place a blank UFS device into its socket adapter (note the <u>PIN1</u> location indicated on the socket adapter board).
- 2. Ensure the system is in the appropriate presenter mode: in the **Settings** group, from the **Presenter Mode** drop-down list, select **Desktop Mode** (vs PSV7000 or PSV5000 for automated systems).

Settings				×
Programmers		Directories		
10.0.0.2	×Ì	Jobs	C:\LumenX\Jobs	Browse Update
Status: Idle		Algorithms	C:\LumenX\Algorithms	Browse Update
Filliwale. 1.3.1			No Algorithms? Algorithm Updater	
		Updates	C:\LumenX\Updates	Browse Update
		Settings		
		Presenter N	lode Desktop Mode 🔻	
Discover				

3. In the left pane, check the box for the desired programmer(s), and then click **Run**.

🍂 LumenX Data M	🖌 LumenX Data Management Software (DMS)						
<u>File V</u> iew <u>/</u>	<u>A</u> lerts <u>S</u> ettings <u>T</u> ools <u>H</u> elp He	ello, Local 🛛					
Programmers	Programmers O Loaded Job (open for details): Partner 26 UFS Job XYZ						
Status: Ic Firmware	Jle 2: 1.5.1 2						
SUMMA INSERT To run a Change Prog	ARY: Select programmer(s), run job. DEVICES BEFORE RUNNING. iob: rammer selections if desired, load devices and click Run Job button to run this joł	Run					

- 4. View the **Job Progress** pane for the current status:
- **DUT** = Device Under Test = Socket
- **IDLE** = Job is downloading/no activity in slots
- **BUSY** = Job is programming
- PASS = Job has successfully completed
- **FAIL** = Error occurred



When you run a specific algorithm for the first time, there may be an apparent delay between the **Downloading** and **Programming** states because LumenX may need to update the bitstream version of the UFS Interface Board. If needed, the programmer takes approximately one minute (the yellow LED remains lit) to complete the bitstream update.

Chapter 5: Verify Job Completion

This Chapter provides instructions to verify that the job completed, specifically:

Programmer and socket status confirmation

Socket adapter LEDs check

Error alert notification

After configuring and running the programming job in the last Chapter, now verify in LumenX that the job completed properly. Ultimately, verifying that the part/device was programmed properly involves reading from it (not addressed in this document).

Confirming Status

1. In the left pane under **Programmers**, verify that the **Status** entry shows **Done**.



2. To confirm socket status, verify that the **Job Progress** pane shows **PASS** for each socket and programmer used.



Checking Socket Adapter LEDs

All LEDs will light simultaneously at startup and go off when the startup process is complete.

- **Green** = PASS
- Yellow = BUSY
- **Red** = FAIL
- White = Continuity Error

Checking for Alerts

If an error occurs (red-color LED of socket stays lit or flashes on-and-off repeatedly), check for alert notifications at the top and bottom of LumenX Data Management Software (DMS).



Appendices

The following Appendices provide additional details about UFS configuration:

Appendix A: Job Settings, Job Process Selection Appendix B: Device Settings, UFS Configuration Descriptors Appendix C: Device Settings, UFS Device Attributes Appendix D: LUN Settings, UFS Unit Descriptors Appendix E: Sample UFS Job with Default Settings Appendix F: User Interface Legend

To avoid disrupting the workflow of running the programming job, many of the UFS settings are detailed here rather than inline with the steps earlier.

Note: The settings and default values shown here are for sample purposes only; they may vary slightly from those of your specific device and algorithm.

Appendix A: Job Settings, Job Process Selection

Make the desired job process selection(s) from the drop-down lists in the **Job** section.

💢 LumenX Data	Management Software (DMS)	_		×
<u>F</u> ile <u>V</u> iew	<u>A</u> lerts <u>S</u> ettings <u>T</u> ools <u>H</u> elp	He	llo, Loca	l⊗
CREATE JOB: 5	Samsung Generic UFS 2.0 Algorithm TFBGA153			
Job Name	Partner 8 UFS Job XYZ			
Description	This is UFS job XYZ, Run #4 for Partner 8			
Operator Not	No need to re-teach pkg; test tray feeder before starting job run			
Job ID	75618fe1-ddfc-4da0-9525-190122411fc0			
Job Checksum	CD5302FD			
ото то	Process: Name Value Job Process Selection: Programming and Verify Erase Selection: Full chip Erase			
Device Ada	rice: 32GB Automotive UFS 2.1 Algorithm TFBGA153 orithm: UFS 2.1 Algorithm Version: 0.1.1 pter Id: 110103 ▼			ן

Setting	Available Options	Description
Job Process Selection	 Program and Verify (default) 	Specifies the programming operation(s) to perform on the device.
	Verify Only	
Erase Selection	 Disabled Full Chip Erase (default) 	Specifies if Full Chip Erase is enabled.

Appendix B: Device Settings, UFS Configuration Descriptors

Configure the desired descriptors in the **Device** section.

🎉 l	LumenX Dat	a Managen	nent Software	(DMS)					—		×
<u>F</u> ile	<u>V</u> iew	<u>A</u> lerts	<u>S</u> ettings	Tools	<u>H</u> elp				He	llo, Local	0
CRE	ATE JOB:	Samsung	Generic UF	5 2.0 Algo	orithm TFB	GA153					
	•	UFS Des	criptor - Pre-	Program	ming 9 ite	m(s)					
		bBootEr	nable - Enab	le the Bo	ot feature						
		bInitPov	werMode - Iı	nitial Pow	er Mode			0x01: Active	Mode (def	ault)	
		bHighP	riorityLUN -	High Prio	ority Logica	l Unit		7F			
		bDescr/	AccessEn - Ei	nable acc	ess to the l	Device Descript	or after	•			
		bSecure	eRemovalTyp	e - Secur	re Removal	Туре		0x00: Erase	of physical	memory	
		blnitAct	tivelCCLevel	- Initial A	ctive ICC L	evel		0			
		wPeriod	licRTCUpdat	e[8:6] - TI	IME_UNIT			0			
		wPeriod	licRTCUpdat	e[9] - TIM	1E_BASELIN	IE		0			
		wPeriod	licRTCUpdat	e[5:0] - TI	IME_PERIO	D		0			41_
		hDooti	nEn Attribut	a 1 itami	-)						

Setting	Available Options	Description
bBootEnable	 00h Disabled (default) 	Specifies if the device is bootable.
blnitPowerMode	00h UFS-Sleep Mode01h Active Mode (default)	Specifies the initial power mode.
bHighPriorityLUN	 7Fh (default) 0 to <i>n</i>, where <i>n</i> = # of LUNs specified by bMaxNumberLU 	Specifies which LUN has the command queue with the highest priority. Default value 7F means equal priority across all LUNs.
bDescrAccessEn	 00h Disabled (default) 01h Enable 	Specifies if Device Descriptors are accessible after initialization.
bSecureRemovalType	 00h Erase (default) 01h Overwrite once, then erase 02h Overwrite thrice, then erase 	Specifies the method by which information is removed.

	 03h Remove by vendor method 	
blnitActivelCCLevel	0 (default)00h to 0Fh	Specifies the bActivelCCLevel after power on or reset.
wPeriodicRTCUpdate[8:6]	 Ob Undefined (default) 	Specifies the TIME_UNIT of real- time clock updates.
wPeriodicRTCUpdate[9]	 0b Time from last update 	Specifies the TIME_BASELINE of real-time clock updates.
wPeriodicRTCUpdate[5:0]	• 0b	Specifies the TIME_PERIOD of real-time clock updates.

Appendix C: Device Settings, UFS Device Attributes

Configure the desired device attributes in the **Device** section.

🎎 LumenX Data Management Software (DMS)							×
File	ile <u>V</u> iew <u>A</u> lerts <u>S</u> ettings <u>T</u> ools <u>H</u> elp				Hello, Local 🛛		0
CRE	ATE JOB: Samsung G	ieneric UFS 2.0	Algorithm TFBGA153				
$ \cap $	🔊 Settings: 1	8 settings, 0 e	rrors				
			Name	Value		Тур	e
	😔 UFS Descri	iptor - Pre-Pro	ramming 9 item(s)				
	📀 bBootLunEn Attribute 1 item(s)						ll"
	Soutoford	derDataEn Attr	bute 1 item(s)				ш.
	📀 bConfigDe	scrLock Attrib	te 1 item(s)				ш.
	📀 bMaxData	InSize Attribute	1 item(s)				ш.
	📀 bMaxData	OutSize Attribu	te 1 item(s)				ш.
	📀 bMaxNum	OfRTT Attribut	1 item(s)				ш.
	📀 bRefClkFre	q Attribute 1 i	em(s)				ш.
	📀 fPermanen	tWPEn Flag 1	em(s)				ш.
	V fPermanen	itlyDisableFwU	odate Flag 1 item(s)				

Setting	Available Options	Description
bBootLunEn	 00h Boot disabled (default) 	Specifies if a particular LUN is active during boot.
bOutOfOrderDateEn	00h Disabled (default)01h Enabled	Specifies if out of order sequencing is enabled.
bConfigDescrLock	 Oh Disabled (default) 1h Enabled 	Specifies if device configuration (Configuration Descriptor) is locked.

bMaxDataInSize	 0 to n, where n = bMaxInBufferSize 	Specifies the maximum size of data in (number of 512-byte units).
bMaxDataOutSize	 0 to n, where n = bMaxOutBufferSize 	Specifies the maximum size of data out (number of 512-byte units).
bMaxNumOfRTT	 0 to n, where n = bDeviceRTTCap 	Specifies the maximum number of outstanding RTTs allowed.
bRefClkFreq	 0x00h - 19.2 MHz 0x01h - 26 MHz (default) 0x02h - 38.4 MHz 0x03h - 52 MHz Others: Reserved 	Specifies the reference clock frequency.
fPermanentWPEn	 00h Disabled (default) 01h Enabled 	Specifies if permanent write protection is enabled.
fPermanentlyDisableFwUpdate	 Ob Disabled (default) 1b Enabled 	Specifies if firmware updates are permanently disallowed.

Appendix D: LUN Settings, UFS Unit Descriptors

Configure the desired LUN descriptors in the **Memory Region** section.

	LUN0 Memory Region (29.8 Gl	Add Static Data Add Serialized Data				
	Data	Checksum (hex)	Start-End (hex)	Process	Edit	Del
	Serialized Data Source	Start With	Start-End (hex)	Process	Edit	Del
	Settings:					
	Name		١	/alue	Туре	
	bBootLunID - Boot LUN Ident	ifier	0x00: Not a Boot Lo	gical Unit (default) 🛛 🔻		
MEMORY	bLUWriteProtect - Logical Un	it Write Protect	0x00: Not Write-Pro	otected (default)		
1010 1010 1011 1011	bMemoryType - Logical Unit	Memory Type	0x00: Normal Memo	ory (default) 🔹 🔻		
REGION	bDataReliability - Enable the	Data Reliability feature				
	bLogicalBlockSize - Size of ad	dressable logical blocks	С		Hex	
	dNumAllocUnits - Size of LUN	I in allocation units	0		Hex	
	bProvisioningType - Logical L	Init Provisioning Type	0x00: Thin Provision	ing disabled (default) 🔻		

Setting	Available Options	Description
• bBootLunID	 00h Not Bootable (default) 01h Boot LUN A 02h Boot LUN B 	Specifies if a particular LUN is bootable.
bLUWriteProtect	 00h Not Write- Protected 	Specifies if the LUN is write- protected.
 bMemoryType 	 00h Normal Memory 	Specifies a particular memory type (as supported by wSupportedMemoryTypes).
 bDataReliability 	 00h Disabled (default) 01h Enable 	Specifies device behavior when a power failure occurs while writing to the LUN.
bLogicalBlockSize	• C (default)	Specifies the logical block size.
dNumAllocUnits	• 0 (default)	Specifies the size of the LUN.
• bProvisioningType	 00h Disabled (default) 01h Enabled and TPRZ=0 02h Enabled and TPRZ=1 	Specifies if thin provisioning is enabled.

Appendix E: Sample UFS Job with Default Settings

The following table shows the default UFS settings applied to a UFS job/device in LumenX Data Management Software (DMS). At a minimum, you only need to 1) load the intended data/image file(s) to program into the device and 2) set the LUN size/capacity with **dNumAllocUnits**.

Level	Setting	Options
Job	Job Process Selection	Program and Verify
	Erase Selection	Full Chip Erase
Device	Descriptor: bBootEnable	00h Disabled
	Descriptor: blnitPowerMode	01h Active Mode
	Descriptor: bHighPriorityLUN	• 7F (all LUNs equally prioritized)
	Descriptor: bDescrAccessEn	00h Disabled
	Descriptor: bSecureRemovalType	O0h Erase
	Descriptor: blnitActivelCCLevel	• 0
	Descriptor: bPeriodicRTCUpdate[8:6] TIME_UNIT	Ob Undefined
	Descriptor: bPeriodicRTCUpdate[9] TIME_BASELINE	Ob Undefined
	Descriptor: bPeriodicRTCUpdate[5:0] TIME_PERIOD	Ob Undefined
	Attribute: bBootLunEn	00h Boot disabled
	Attribute: bOutOfOrderDataEn	00h Disabled
	Attribute: bConfigDescrLock	Oh Disabled
	Attribute: bMaxDataInSize	• 8
	Attribute: bMaxDataOutSize	• 8
	Attribute: bMaxNumOfRTT	• 2
	Attribute: bRefClkFreq	• 0x01h - 26 MHz
	Attribute: bPermanentWPEn	00h Disabled
	Attribute: bPermanentlyDisableFwUpdate	Ob Disabled
LUN	bBootLunID	00h Not Bootable
	bLUWriteProtect	O0h Not Write- Protected
	bMemoryType	00h Normal Memory
	bDataReliability	00h Disabled

bLogicalBlockSize	• C
dNumAllocUnits	• 0
bProvisioningType	• 00h Disabled

Appendix F: User Interface Legend



Index

Α

air pressure · 2 socket opener · 7 alert notifications · 33 algorithm · 1, 14, 15, 19

В

bitstream version · 30

С

checksum · 24 Configuration Descriptors · 21

D

Device Attributes · 22 dNumAllocUnits · 28

Ε

Electrostatic discharge (ESD) \cdot 2

F

firmware · 13, 19, 38 Firmware Update package · 18

Η

Hazards $\cdot 2$

L

LUN Settings · 23

Ν

network connectivity · 12

S

Safety Precautions · 2 socket actuator · 2, 7 adapter · 1, 3, 5, 10, 11, 29 clamp · 5, 6 status · 31 Socket Loading · 7 socket opener air pressure · 7 System Update package · 17

U

Unit Descriptors · 26 User Interface Legend · 42

V

version Automated Handler (AH) · 3 LumenX Data Management Software (DMS) · 3 programmer firmware · 18 programmer system · 17

W

WARNINGS · 2