



Each edition of M.O.S.T. Toolz includes one of the four readers/scanners shown above.

M.O.S.T. Toolz™ is a development platform for multifunction and high security M.O.S.T. Card® smart card systems. Available in contact, dual mode, and biometric editions, the kits feature user-friendly configuration and demonstration software for rapid system development, powerful middleware for advanced reader functions, and includes up to 17 microprocessor smart cards for system and card file setup. The M.O.S.T. Card Configuration Utility™ helps you build an on-card file system with direct calls to the M.O.S.T. Card. High level commands can be sent through the Winplex® API to the card reader. Fully documented C# (.NET), C++ (.NET), Java, and Visual Basic programming examples complete the kit. Programmers can create a smart card-based ID or transaction system that updates and secures files while setting a variety of defensive measures to protect user ID, card access, and file information. CardLogix gives you the power to deliver multiple products and services on a single card, allowing for fast system design and easy updating without the need for card re-issuance.

M.O.S.T. Toolz™ Development Kits Include

- M.O.S.T. Card Configuration Utility™
- Winplex® API and Middleware
- M.O.S.T. Card Demo Software (includes demo cards and programming examples)
- 12 M.O.S.T. Cards (+5 M.O.S.T. Cards with Biometric Series)
- Your choice of smart card reader and/or fingerprint scanner
- PC/SC Reader Configuration Utility
- M.O.S.T. Card Bad Select Counter Utility to reset “Bad Select” counter
- Support for Windows 7 and 8
- A variety of idbiox® CFS templates that support the idbiox ID Credential Ecosystem
- M.O.S.T. Toolz™ User Manual
- Winplex® User Manual

M.O.S.T. Card Configuration Utility™

- User-friendly interface for configuring M.O.S.T. card file structures (CFSs)
- Key configuration function to randomize all keys with one click
- On-card data editor to load and edit user data instantly
- Card file list and label manager with memory resource manager
- User configurable NFC UID lengths

Winplex® API and Middleware

- Over 160 standard functions including card latching, LED control, and reading magnetic stripes (reader-specific)
- M.O.S.T. Biometric Series kits provide additional biometric functions for capture, matching, and compression
- PC/SC card reader support
- Programming examples in C# (.NET framework)

M.O.S.T. Card Demo Software (includes demo cards and programming examples)

- Hello World Demo: How to write a large binary file to the card
- Data Protection Demo: How to protect data with a Global Password and more
- Winplex Demo: Overview of M.O.S.T. Card functions
- Transaction Demo: How to initialize value with ePurse, linear EFs and cyclic EFs
- eSignature Demo: How to sign a data package with FIPS 198-1 signature
- Fingerprint Demo: How to capture, validate, and store ICAO fingerprints (Biometric Series only)



M.O.S.T. Toolz Kit Options

Part Number	Edition	Card Reader/Scanner
9 700 001	M.O.S.T. Toolz™	Contact smart card reader
9 700 009	M.O.S.T. Toolz™	Dual mode (contact and contactless) smart card reader
9 700 010	M.O.S.T. Toolz™ Biometric Series Fingerprint	Single-digit optical fingerprint scanner and contact smart card reader
9 700 015	M.O.S.T. Toolz™ Fingerprint Upgrade to Dual Mode Kit	Single-digit optical fingerprint scanner

M.O.S.T. Toolz Reader/Scanner Options



Contact Smart Card Reader



Contact/Contactless Smart Card Reader



Fingerprint Scanner + Contact Smart Card Reader



Fingerprint Scanner

Card File Directory
Employee_3036_XXXXX_C8.cfs

Filename	Type	3F Bytes	FS Bytes	Total
3F00	MF		27	27
2F01	ATR		36	36
3000	DF		32	32
3040	EF	30	18	48
3041	EF	10	18	28
3042	EF	8	18	26
3043	EF	2	18	20
3090	APP	38	38	76
3F80	CHV	34	34	68
3F90	APP	38	38	76
4000	DF		32	32
4043	EF	4000	18	4018
4044	EF	4000	18	4018
4048	EF	4000	18	4018
4049	EF	4000	18	4018
4080	CHV	34	34	68
4090	APP	38	38	76
4300	DF		32	32
4340	EF	30	18	48
4341	EF	40	18	58
4342	EF	30	18	48
4343	EF	30	18	48
5000	DF		32	32
5040	EF	20	18	38

File Properties
File Number: DB40 Encryption Status: None
File Name: CCC

Security Settings

Protection	File
Read: Always Accessible	-
Write: APP Protected	DB90
Update: APP Protected	DB90
Invalidate: APP Protected	DB90
Rehabilitate: APP Protected	DB90

File Size: 144

File Description
Card Capability Container includes: ISID, CARDiss, IF#, UNICODE Version #, LDS type, LDS version number, LDS file name, Data encoding method, RID AIDs

Key Status

Administrator	Transport
HMAC	DES AES
Hash ID	Global Password

File System Configuration - CLXSU624KC8/T=CLED
Memory Used: 42540 Available: 35460 Maximum: 78000

Edit EF File
EF File: DB40 CCC File Size: 144

File Security

Protection Type	File
Read: Always Accessible	-
Write: APP Protected	DB90
Update: APP Protected	DB90
Invalidate: APP Protected	DB90
Rehabilitate: APP Protected	DB90

Encrypted Session Required BAC

File Description
Card Capability Container includes: ISID, CARDiss, IF#, UNICODE Version #, LDS type, LDS version number, LDS file name, Data encoding method, RID AIDs

Edit AES Keys

AES Key Hex Values

Key 1: C2 F4 78 C3 43 CF RE D8 H1 D4 RE A5 F2 11 5D 7E DA 7A AA C2 A4 EA P5 74 F2 AF FC EE H02 14 9F

Key 2: 75 3F 23 C2 B2 53 H0C 43 A3 H0E D0 38 DA 3F H0D 5C 2A P5 P9 C1 H27 H0C H0C 7A H4 58 H0 H0 A1 C2 H0 F0

Key 3: H0C H7A H39 H32 H29 C3 H01 H00 H54 H04 H49 H0C H0D H3A H04 H0D H09 H19 H1C H48 H58 H44 H53 H16 H20 H03 H45 H57 H0C H34 H45

Key 4: H15 H42 H04 H11 H0C H0F H03 H49 H2C H1C H00 H09 H38 H2F H95 H28 H1A H81 H7 H0A H03 H08 H0A H03 H09 H81 H8E H98 H04 H57

Key 5: 70 H17 H66 H03 H88 H4A H60 H05 H8F H07 H75 H72 H8A H0E H4E H40 H4C H2D HCA H8E H80 H6A H00 H0E H88 H2A H0C H174 H0C H15 H30

Key 6: 70 H08 H02 H05 H4F H57 H25 H57 H5 H02 H44 H03 H05 H58 H44 H03 H70 H15 H79 H90 HAD H90 H08 H06 H3F H09 H26 H08 H05 H50 H7A H7A

Note - AES Keys are not written to the smart card using this tool. Use these fields to store your keys for use during production personalization by CardLogix. When using AES128, the first 16 bytes of the key are used. When using AES192, the first 24 bytes of the key are used. When using AES256, all 32 bytes of the key are used.