



# MPOST .NET API Revision History

**MPOST .NET supports x86 and x64 architecture.**

## Version 3.50

- Added fast download feature at 38400bps into MPOST.
- Added a new STACKED event with the document type and value to avoid incorrect bill value report.
- Delayed the clearance of the document type and value from an accepting state to an escrow or rejecting state to avoid zero bill value report.
- Disabled the calibration button on the MPOST Demo to prevent the customer from clearing the calibration data by mistake.
- Corrected the two STACKED event report issue for the same note (rarely happens).
- Corrected the no value report for a stacked note after communication is cut and recovering.
- Resolved the hang up issue after multiple USB communication cut.
- Resolved the communication issue for lost messages by redesigning the data receiving method.
- Resolved disconnection issues with EBDS over USB harness and regular EBDS harness.

## Version 3.40

- Improved flash download support for older SC Advance firmware. Previously, the MPOST library was not able to perform upgrades on some devices due to a bug in the device firmware. (ex V2.40 of EBDS Retail for SC Advance).
- Corrected an indexing problem that caused incorrect note value to be reported for Series 2000 devices loaded with Brazil code V1.80 or later.
- Added new feature to support retrieving the unique serial number of the EASITRAX RFID Tag.

## Version 3.30

- <Version Not Released>

## Version 3.20

- Corrected critical exception situation when an AE device reports the 'stacked' event at the start of API communications. Previously, if the device reported stacked in the first response, MPOST would fail because it attempted to access a banknote table that was not yet initialized.

## Version 3.10

- Improved message timeout detection to prevent unnecessary timeouts on long operations (specifically the note audit command).
- Fixed bug that caused the disconnected event to be sent if the system time changed.
- Improved retry mechanism for synchronous commands. Addresses an issue where the download update event was not being raised for the very first download packet.

## Version 3.00

- Fixed bug introduced in last version that caused the Soft Reset command to get stuck in an infinite loop.

## Version 2.90



# MPOST .NET API Revision History

- Improved port management to better handle surprise USB disconnections. The library now ensures the OS has sufficient time to refresh hardware list before attempting a reconnection when using a Virtual Com Port (VCP) Driver.
- Corrected issue where the library could report credit for the value of the previous note after clearing a BNF jam.
- Fixed rare issue around potential response time out issues that could cause a stack request (or similar operation) to be ignored by the system because it was mistaken for a NAK command.

## Version 2.80

- Added support for changes to BNF Status command. New property "LastBNFError".

## Version 2.70

- Added support for new Audit commands. Host can retrieve the full audit data for all banknotes.

## Version 2.60

- Added support for *Clear Audit* command. SC Advance units only.
- Ensure specific denominations can be enabled/disabled for Series 2000 products.
- Demo applications can be built in Visual Studios 2010.

## Version 2.50

- Ensure bill data is displayed correctly.
- Ensure all documents are properly returned if they are inserted at the same time a disable command is sent.
- Capability for Flash download will now be properly set if the device powers up in the Flash Download state.
- Added support for Cashbox Cleanliness Reporting.

## Version 2.40

- Ensure Stacked event comes before Power Up Complete Event if communications are lost during power up.
- Ensure the device is disabled immediately after the connection is closed.
- Improved device capabilities detection on start up with Series 2000 devices.
- Prevent Blue Screen issue when disconnecting device that is using Virtual Comm Port Drivers.

## Version 2.30

- Ensure log file resources are properly released during the close function call.
- Events are now raised asynchronously to ensure thread safety between Host and API.
- Ensure DownloadRestart Event is raised correctly.
- Ensure message is properly built if received by serial port in multiple transactions.

## Version 2.20

- Updated the Open function to throw exceptions if the port is invalid for any reason.

## Version 2.10



# MPOST .NET API Revision History

- Completed support for COM interoperability. Be sure to review the “MPOST .NET COM Callable Wrapper Manual.pdf” for more information on how to properly reference the DLL for COM applications.
- There are no changes to the underlying API.

## Version 2.00

- In anticipation for interoperability support with COM, changes have been made to several methods and properties. These changes are as follows:
  - All properties of type `char` have been changed to type `string`.
  - All array properties that are both settable and gettable have been deprecated, generating a compiler warning in V1.60 and a compiler error in the next API version and will be removed completely in versions following that.
  - Settable array properties have been replaced by a `SetPropertyName` method and a `GetPropertyNames` method for symmetry sake.

These changes are as follows:

Signature prior to V1.60	Property Type	Method Signature in V2.00
<code>BillTypeEnables</code>	Get	<code>bool[] GetBillTypeEnables()</code>
<code>BillTypeEnables</code>	Set	<code>void SetBillTypeEnables(ref bool[])</code>
<code>BillValueEnables</code>	Get	<code>bool[] GetBillValueEnables()</code>
<code>BillValueEnables</code>	Set	<code>void SetBillValueEnables(ref bool[])</code>
<code>byte[] RawTransaction(byte[] command)</code>	N/A	<code>Byte[] RawTransaction(ref byte[] command)</code>

- Added several new events to the API.
  - `OnPowerUpComplete` – This event will fire after the event “OnPowerUp.” It signals that the device has completed all of its power up initialization and can now react normally to any user requests
  - `OnSendMessageFailure` – This event is raised if there was a problem issuing a synchronous command to the device. For example, if a command to stack an escrowed note fails, the user would be informed through this event. The event will carry the Message that failed to be sent.
  - `OnStackerFullCleared` – This event is designed to support older versions of Stackerless Code. It will fire when the Stacker Full condition was cleared.
  - `OnNoteRetrieved` – This event is raised when the user removes a note from the rejected/returned position. This event is only supported on a certain device software revision and will not be raised unless supported.
- Deprecated several uncommonly used functions
  - `SpecifyPupExtMode` – This function was deprecated. It is not supported by any current device.
  - `TransactionTimeout` Property – The API will now fully manage the timing for internal messages.
  - `DownloadTimeout` Property – The API will now fully manage the timing during flash download.
- New Properties/Methods
  - Overloaded the `Open` function. Now a device can be opened without having to specify a power up policy. The default will be the recommended `PowerUP_A`.



## MPOST .NET API Revision History

- New Property called DisconnectTimeout can be set (in milliseconds) to specify the amount of time before the unit is considered “disconnected.” The Disconnected Event will fire if the device does not respond to the API after the specified amount of time (Default is 30 seconds). It is recommended not to set this value too low because the host could receive incorrect Disconnected Events. (Recommended minimum 4 seconds (4000ms))
- New functions for Advanced Bookmark Feature. This feature is only supported by newer revisions of code. If the host calls “EnterAdvancedBookmarkMode()” the device will accept the next invalid document that is inserted (So long as it meets size restrictions 102mm > document > 160mm). This mode is exited as soon as a document is accepted or it can be canceled by calling “CancelAdvancedBookmarkMode()”. Note – Valid notes that are inserted while this mode is active will automatically be returned.
- General fixes and improvements
  - Reduced the overall CPU usage of the API
  - Improved control of the order of events during power up and connection.
  - Important events will now be re-raised after power is lost or the connection is lost. These events include: Failure, Jammed, StackerFull, Escrow, Cashbox Removed.
  - Increased reliability of message handling within the API and added more robust retry ability.
  - Improvements to Flash Download algorithm to improve reliability.

### Version 1.18

- The AutoStack property is now only settable once the connected event has been fired.
- After receiving a disconnect event, the Autostack property is now set to false.
- When the AutoStack property is set to true while a note is in escrow, the note is now stacked.