**ModLS.dll**

Manual Técnico



***Este documento pertenece a Symetry, S.A. de C.V.***

***Solo lea este documento si está autorizado a ello. Prohibida su reproducción total o parcial***

# Introducción

**ModLS.dll** es una biblioteca diseñada para utilizar los dispositivos LS100, LS40 comercializados por Symetry de una manera flexible y transparente. Se refiere a los [archivos](https://es.wikipedia.org/wiki/Archivo_inform%C3%A1tico) con [código ejecutable](https://es.wikipedia.org/wiki/C%C3%B3digo_ejecutable) que se cargan bajo demanda de un [programa](https://es.wikipedia.org/wiki/Programa_(inform%C3%A1tica)) por parte del [sistema operativo](https://es.wikipedia.org/wiki/Sistema_operativo), por lo cual, las aplicaciones que requieran hacer uso estos periféricos no tengan que programar su interacción a bajo nivel, con esto garantizamos el uso adecuado y la integración de todas sus funcionalidades.

Para su uso debe de referenciarse dentro del proyecto que se requiera y así poder llamar los procesos correspondientes.

# Uso

Una vez que se ha referenciado la biblioteca se invoca cualquiera de sus procesos de esta manera *ModLS.LSApi.Nombre del proceso,* los cuales se mencionan a continuación:

* **LSConnect**(Int32 hWnd, Int32 Hinst, short Peripheral, out short hConnect)

Parámetros:

* hWnd: Es un manejador el cual debe ser un 0.
* Hinst: Es un manejador el cual debe ser un 0.
* Peripheral: Representa el dispositivo con el que se va interactuar y el modo de conexión (USB, RS232, etc.). Para los casos con el LS100 y LS40 se utiliza la misma constante la cual es el siguiente: ModLS.LSApi.LS\_100\_USB.
* hConnect: Es un manejador el cual debe ser un 0 y en ultima instancia es donde se retorna el valor del estatus.

**LSConnect** se encarga de iniciar la comunicación con el dispositivo. Retornando un entero para determinar el estatus del mismo, el 0 representa el estatus óptimo de operación.

* **LSDisconnect**(short hConnect, Int32 hWnd)

Parámetros:

* hConnect: Es un manejador el cual debe ser un 0.
* hWnd: Es un manejador el cual debe ser un 0.

**LSDisconnect** se encarga de cortar la comunicación con el dispositivo.

**PARA LECTURA DE CHEQUES**

* **LSDocHandle**(short hConnect, Int32 hWnd, short Stamp

, short Validate, short CodeLine, char Side

, short ScanMode, short Feeder

, short Sorter, short WaitTimeout

, short Beep, ref Int32 NrDoc

, short Reserved1, Int32 Reserved2)

Parámetros:

* hConnect: Es un manejador el cual debe ser un 0.
* hWnd: Es un manejador el cual debe ser un 0.
* Stamp: Opción de lectura con o sin sello. Esta parte solo esta habilitada con la opción sin sello de esta manera: ModLS.LSApi.NO\_STAMP.
* Validate: Sirve para la validación de impresión y se puede tomar las siguientes opciones: ModLS.LSApi.NO\_PRINT\_VALIDATE o ModLS.LSApi. PRINT\_VALIDATE.
* CodeLine: Parámetro para la lectura del código del cheque y tiene solo la siguiente opción: ModLS.LSApi.READ\_CODELINE\_MICR.
* Side: Sirve para parametrizar los lados de lectura, los cuales pueden ser los siguientes: ModLS.LSApi.SIDE\_ALL\_IMAGE, ModLS.LSApi.SIDE\_NONE\_IMAGE, ModLS.LSApi.SIDE\_FRONT\_IMAGE, ModLS.LSApi.SIDE\_BACK\_IMAGE.
* ScanMode: Representa el modo de escaneo. Los cuales son los siguientes: ModLS.LSApi.SCAN\_MODE\_16GR200, ModLS.LSApi.SCAN\_MODE\_BW, ModLS.LSApi.SCAN\_MODE\_16GR100, ModLS.LSApi.SCAN\_MODE\_256GR100, ModLS.LSApi.SCAN\_MODE\_256GR200, ModLS.LSApi.SCAN\_MODE\_BW\_TIFF, ModLS.LSApi.SCAN\_MODE\_COLOR\_100, ModLS.LSApi.SCAN\_MODE\_COLOR\_200.
* Feeder: Tipo de alimentación y tiene una sola opción: ModLS.LSApi.AUTO\_FEED.
* Sorter: Opción de clasificador y solo hay una: ModLS.LSApi.SORTER\_BAY1.
* WaitTimeout: Parámetro de tiempo de espera para colocar el cheque. Pueden ser dos opciones: (short)ModLS.LSApi.WAIT\_NO, (short)ModLS.LSApi.WAIT\_YES.
* Beep: Activación de sonido por parte del dispositivo, pueden ser dos opciones: ModLS.LSApi.NO\_BEEP, ModLS.LSApi.BEEP.
* NrDoc: Es un manejador el cual debe ser un 0.
* Reserved1: Es un manejador el cual debe ser un 0.
* Reserved2: Es un manejador el cual debe ser un 0.

**LSDocHandle** es el proceso para preparar la lectura de un cheque, determinando si existe un problema con el dispositivo, el 0 representa el estatus óptimo de operación.

* **LSReadCodelineMod**(short hConnect, Int32 hWnd, out String Codeline, out string Barcode, out string Optic)

Parámetros:

* hConnect: Es un manejador el cual debe ser un 0.
* hWnd: Es un manejador el cual debe ser un 0.
* Codeline: Variable string con información del código impreso en el cheque.
* Barcode: Variable string con información del código impreso en el cheque.
* Optic: Variable string con información del código impreso en el cheque.

**LSReadCodelineMod** Proceso para validar la lectura correcta del código de los cheques, el 0 representa el estatus óptimo de operación.

* **LSReadImage**(short hConnect, Int32 hWnd, short ClearBlack, char Side, short ReadMode, Int32 NrDoc, out Int32 FrontImage, out Int32 BackImage, ref Int32 Reserved1, ref Int32 Reserved2)

Parámetros:

* hConnect: Es un manejador el cual debe ser un 0.
* hWnd: Es un manejador el cual debe ser un 0.
* ClearBlack: Opciones de la salida de las imágenes, pueden ser dos: ModLS.LSApi.CLEAR\_ALL\_BLACK o ModLS.LSApi.NO\_CLEAR\_BLACK.
* Side: Opciones de la dimensión de la imagen a capturar, pueden ser las siguientes opciones: ModLS.LSApi.SIDE\_ALL\_IMAGE, ModLS.LSApi.SIDE\_NONE\_IMAGE, ModLS.LSApi.SIDE\_FRONT\_IMAGE, ModLS.LSApi.SIDE\_BACK\_IMAGE.
* ReadMode: Modo de lectura y pueden ser las siguientes: ModLS.LSApi.READMODE\_ALL, ModLS.LSApi.READMODE\_BRUTTO, ModLS.LSApi.READMODE\_NETTO.
* NrDoc: Es un manejador el cual debe ser un 0.
* FrontImage: Es un manejador el cual debe ser un 0.
* BackImage: Es un manejador el cual debe ser un 0.
* Reserved1: Es un manejador el cual debe ser un 0.
* Reserved2: Es un manejador el cual debe ser un 0.

**LSReadImage** obtiene el resultado del escaneo realizado por el dispositivo.

* **LSSaveJPEG**(Int32 hWnd, Int32 Face, Int32 quality, string filename)

Parámetros:

* hWnd: Es un manejador el cual debe ser un 0.
* Face: Es un manejador el cual debe ser un 0.
* quality: Entero que determina la calidad de imagen de salida.
* filename: String con la ruta en donde se guardará la imagen procesada.

**LSSaveJPEG** Se encarga de guardar en disco la imagen del cheque procesado.

* **LSReset**(short hConnect, Int32 hWnd, char ResetType)

Parámetros:

* hConnect: Es un manejador el cual debe ser un 0.
* hWnd: Es un manejador el cual debe ser un 0.
* ResetType: Define el tipo de Reset a realizar en el dispositivo, los cuales pueden ser los siguientes: ModLS.LSApi.RESET\_FREE\_PATH, ModLS.LSApi.RESET\_ERROR.

**LSReset** es el proceso que se encarga del reset del dispositivo en caso de un atascamiento del cheque.

**PARA LECTURA DE TARJETAS**

* **LSReadBadgeWithTimeout**(short hConnect, Int32 hWnd, char Format, short MaxLength

, [MarshalAs(UnmanagedType.LPArray)] byte[] String

, [MarshalAs(UnmanagedType.LPArray)] Int16[] length

, Int32 Timeout)

Parámetros:

* HConnect: Es un manejador el cual debe ser un 0.
* HWnd: Es un manejador el cual debe ser un 0.
* Format: Definición de formato, los cuales pueden ser: ModLS.LSApi.FORMAT\_IATA\_ABA, ModLS.LSApi.FORMAT\_IATA, ModLS.LSApi.FORMAT\_ABA, ModLS.LSApi.FORMAT\_MINTS, ModLS.LSApi.FORMAT\_ABA\_MINTS, ModLS.LSApi.FORMAT\_IATA\_ABA\_MINTS
* MaxLength: Tamaño máximo.
* Arreglo de bytes: Es un manejador el cual debe ser [256].
* Arreglo de shorts: Es un manejador el cual debe ser {256}.
* Timeout: Entero para determinar el tiempo de espera para la lectura. Debe ser expresado en milisegundos.

**LSReadBadgeWithTimeout** es el proceso para la lectura de la banda magnética de las tarjetas y así obtener la información de la misma.