**INSTALACIÓN DE DRIVERS DE LABJACK PARA GUNT EN MATLAB**

**✅ Requisitos previos**

1. **Sistema operativo Windows de 64 bits** (ej. Windows 7 o posterior).
2. **MATLAB instalado (versión de 64 bits)**.
3. **Simulink instalado** (si se usarán los ejemplos de Simulink).
4. Dispositivo **LabJack U12**.
5. **CD o archivos descargados** del paquete de instalación.

**1. Instalación del driver LabJack**

1. Ejecuta el archivo:  
   \drivers\U12SetupV121.exe  
   *(Este se encuentra en el CD o en la carpeta de archivos del software de GUNT o puede descargarse de:* [*labjack.com/support/u12/u12-setup*](http://labjack.com/support/u12/u12-setup)*)*

**2. Copiar drivers de MATLAB**

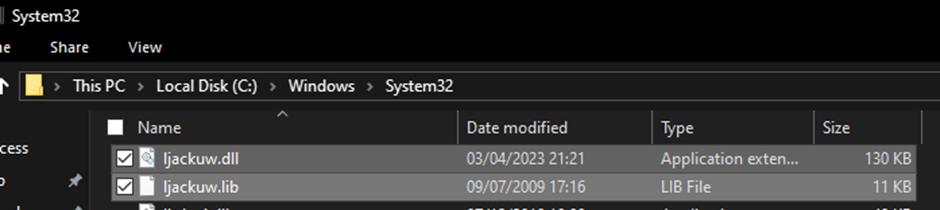
1. Copia el contenido del directorio:  
   \drivers\64bit  
   a una **carpeta local de tu PC** (por ejemplo: C:\LabJack\LJMATLAB).
2. **No uses** LJMATLAB\_W32 (es para sistemas de 32 bits).

⚠️ El path que se recomienda usar no debe llevar espacios, es decir, el nombre de ninguna de las carpetas que constituyen el path no deben tener ningun espacio.

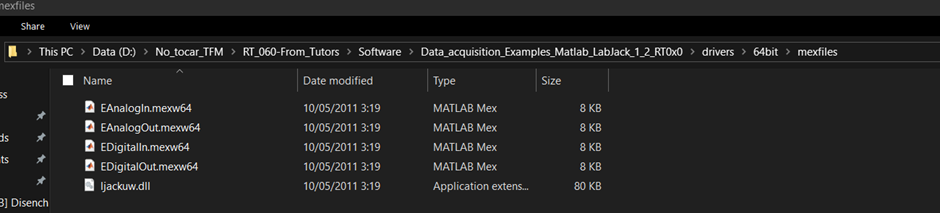
**3. Copia las funciones para MATLAB de 64 bits**

Esta parte es esencial, ya que los ejemplos fueron diseñados para MATLAB de 32 bits. Para que funcionen en 64 bits, se deben seguir pasos adicionales:

1. Descarga y/o descomprime el archivo:  
   [LabJackuw64bit.zip](http://files.labjack.com/uw/LabJackuw64bit.zip)
2. Copia los archivos **dentro del zip** a la carpeta del sistema (en la raíz, sin carpeta):  
   C:\Windows\System32\



1. Descarga y/o descomprime el archivo:  
   [mexfiles.zip](http://files.labjack.com/installers/mexfiles.zip/)
2. Crea una carpeta y mueve todas las funciones recompiladas de MATLAB para 64 bits. Estas incluyen:
   * EAnalogIn.mexw64
   * EAnalogOut.mexw64
   * EDigitalIn.mexw64
   * EDigitalOut.mexw64
   * ljackuw.dll

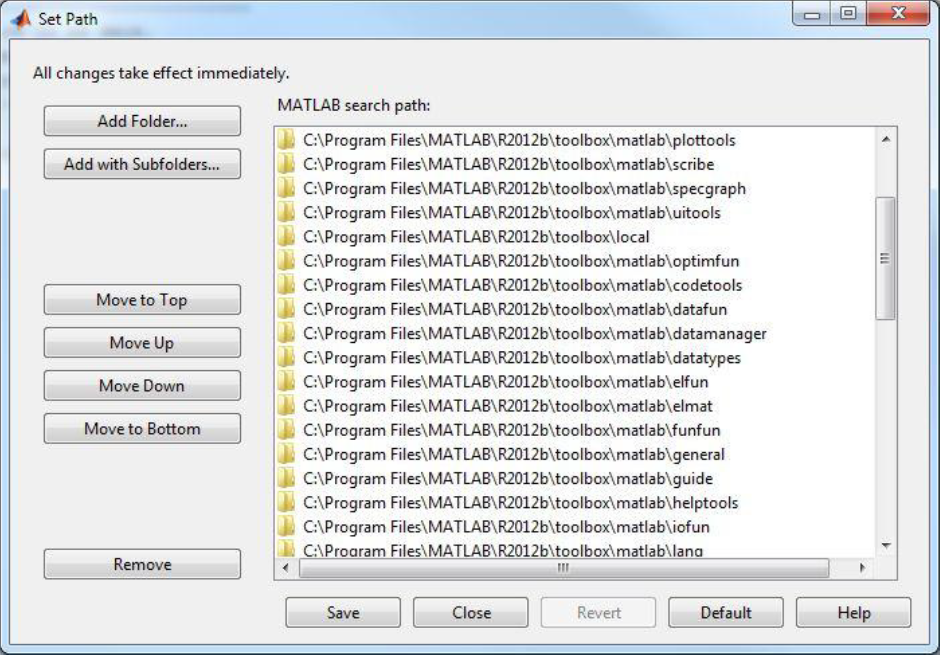


**4. Añadir carpeta al path de Matlab**

1. Abre MATLAB.
2. Haz clic en **“Set Path”**.



1. Elige **“Add with Subfolders…”** y selecciona la carpeta del apartado 2.1, en el ejemplo se usó C:\LabJack\LJMATLAB). Con esto se cargarán las funciones. También debes “**Add Folder”** y elegir en específico la carpeta .



1. Haz clic en **“Save”** y luego en **“Close”**.

⚠️ Se recomienda que los archivos de Matlab que se vayan a crear se añadan a una carpeta de trabajo local, y esta se añada también al path.

**5. Usar los ejemplos (en MATLAB)**

1. Ubica los archivos de \examples\MATLAB en la carpeta de instalación del apartado 1 que estarán dentro \Data\_acquisition\_Examples\_Matlab\_LabJack\_1\_2\_RT0x0\examples\MATLAB .
2. Copia a tu carpeta de trabajo local.
3. En MATLAB, puedes abrirlos directamente o bien navega a esa carpeta con el comando:

cd C:\ruta\_previa\Data\_acquisition\_Examples\_Matlab\_LabJack\_1\_2\_RT0x0\examples\MATLAB

(ruta no valida, deben añadir donde está en su ordenador)

1. Ejecuta los scripts escribiendo su nombre en la consola.Por ejemplo:

digital\_IO

⚠️ Algunos ejemplos requieren una señal de entrada digital alta (por ejemplo, en IO0) para detenerse correctamente.

**6. Usar los ejemplos (en Simulink)**

1. Ubica los archivos de \examples\SIMULINK en la carpeta de instalación del apartado 1 que estarán dentro \Data\_acquisition\_Examples\_Simulink\_LabJack\_1\_2\_RT0x0\examples\SIMULINK.
2. Copia a tu carpeta de trabajo local.
3. En MATLAB, puedes abrirlos directamente o bien navega a esa carpeta.
4. Abre y ejecuta los archivos .slx en Simulink.
5. Asegúrate de estar en la carpeta correcta o Simulink no podrá encontrar los bloques personalizados.
6. Los bloques están definidos en las librerías LJ\_lib.slx y RT0x0\_lib.slx.

⚠️ En algunos casos deberás usar **"Terminator" blocks** para evitar advertencias si una salida no se conecta.