VER LA ÚLTIMA VERSIÓN DEL DOCUMENTO EN PLANTIUM WIKI

|  |
| --- |
| <https://sboxhub.com.ar:8600/es/Procs/Desarrollo/Hardware/Bibliotecas-Altium>  NO AGREGAR INFORMACIÓN A ESTE DOCUMENTO. |
| Plantium_Logo_2017-Color.jpg |
| Uso de bibliotecas en Altium |
| Rev. 2 |
|  |
| 20/07/2022 |

Contenido

[Introducción 3](#_Toc109231574)

[Bibliotecas incluidas al proyecto. 3](#_Toc109231575)

[Bibliotecas ubicadas dentro del proyecto 4](#_Toc109231576)

[Bibliotecas ubicadas fuera del proyecto 4](#_Toc109231577)

[Repositorios (GIT) de proyectos con bibliotecas incluidas 4](#_Toc109231578)

[Ventajas y desventajas 4](#_Toc109231579)

[Bibliotecas instaladas 5](#_Toc109231580)

[Biliotecas de componentes (\*.SchLib) 5](#_Toc109231581)

[Bibliotecas de footprintes (\*.PcbLib) 5](#_Toc109231582)

[Ventajas y desventajas 5](#_Toc109231583)

[Bibliotecas Integradas en Altium (Integrated Libraries) 6](#_Toc109231584)

[Creación de un proyecto de biblioteca integrada 6](#_Toc109231585)

[Compilación de la biblioteca integrada 6](#_Toc109231586)

[Instalación de una biblioteca integrada 7](#_Toc109231587)

[Repositorio (GIT) HW-AltiumLibraries (Plantium) 8](#_Toc109231588)

[Migración de proyectos existentes 8](#_Toc109231589)

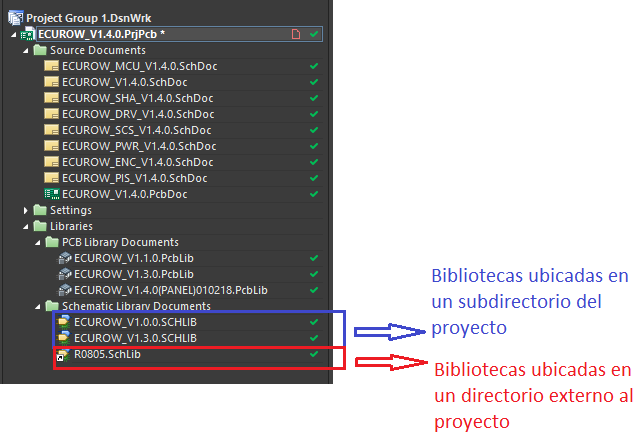
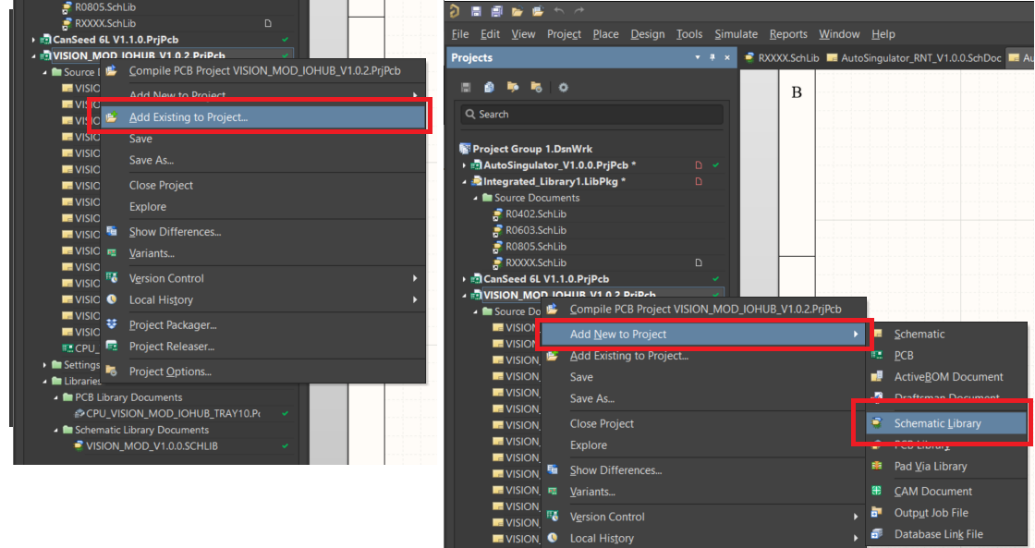
[Bibliografía 9](#_Toc109231590)

## Introducción

Altium permite manejar las bibliotecas de componentes (\*.SchLib, \*.PcbLib, \*.IntLib, etc.) de diversas maneras y este documento menciona varias de estas opciones destacando cuáles son sus ventajas y desventajas respecto del flujo de trabajo que pretende adoptar para el sector de Hardware dev de Plantium.

## Bibliotecas incluidas al proyecto.

Altium permite incluir bibliotecas en los proyectos, ya sean bibliotecas existentes o nuevas bibliotecas vacías. Las bibliotecas incluidas pasan a ser parte constitutiva de los proyectos y son mostradas dentro de la estructura de archivos del proyecto (en la solapa projects). Incluir una nueva biblioteca o borrar una biblioteca ya incluida implica realizar cambios al proyecto y se reflejan como cambios en el archivo de proyecto MyProject.PcbPrj.



### Bibliotecas ubicadas dentro del proyecto

Cuando las bibliotecas incluidas se encuentran dentro del directorio del proyecto el programa las referencia mediante rutas relativas y siempre intenta encontrarlas dentro del directorio del proyecto o en sus subdirectorios (aun cuando el proyecto movido de ubicación o copiado a otra estación de trabajo).

### Bibliotecas ubicadas fuera del proyecto

Cuando las bibliotecas incluidas se encuentran fuera del directorio del proyecto Altium las referencia por rutas absolutas y las trata como “accesos directos”. Notar la diferencia de los íconos en la imagen más arriba**. NO es una buena práctica incluir bibliotecas que estén en directorios externos ya que cuando se copia el proyecto a otra estación de trabajo (vía copia de archivos o vía repositorio) es altamente probable que esas bibliotecas no sean encontradas y el proyecto pierda integridad.**

### Repositorios (GIT) de proyectos con bibliotecas incluidas

**Cuando se trabaja con bibliotecas incluidas estas deben ser incorporadas al repositorio del proyecto** para mantener la integridad del mismo cuando es clonado en otro puesto de trabajo. Caso contrario, cuando el proyecto es abierto el programa arroja mensajes de error por no encontrar los archivos y genera cambios en el proyecto eliminando las dependencias no encontradas.

**Notar la imposibilidad de mantener de forma consistente repositorios de proyectos que incluyan bibliotecas ubicadas en directorios externos al proyecto**

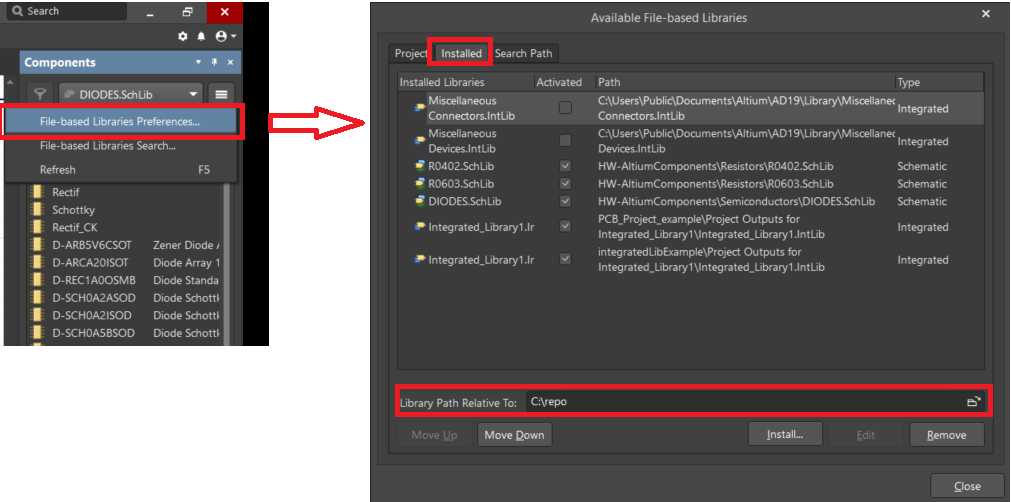
### Ventajas y desventajas

La ventaja de esta forma de trabajo es que el proyecto contiene todas sus dependencias y no es necesario contar con ninguna otra biblioteca adicional instalada.

La principal desventaja es que no permite el mantenimiento y actualización de una única biblioteca común a todos los proyectos como se pretende hacer dentro del sector Hardware de Plantium**. Es por esto que NO se utilizará este flujo de trabajo.**

## Bibliotecas instaladas

Otra alternativa es usar bibliotecas instaladas las cuales se pueden referenciar y usar desde todos los proyectos.



### Biliotecas de componentes (\*.SchLib)

Las bibliotecas de esquemáticos \*.SchLib se instalan desde “Components->File-based-libraries preferences->Installed” y se pueden instalar haciendo referencia a una ruta absoluta o a una ruta relativa. La ventaja de usar una ruta relativa es que se podría reubicar o cambiar la carpeta que contiene todas las bibliotecas y solo sería necesario editar el campo “Library Path Reative To” sin necesidad de volver a instalar todas las bibliotecas una a una.

### Bibliotecas de footprintes (\*.PcbLib)

Las bibliotecas de footprints \*.PcbLib se pueden instalar de igual manera que las \*.SchLibs o se pueden referenciar mediantes las rutas de búsqueda (search paths).

Las \*.PcbLib´s instaladas se pueden acceder desde el panel “Components” y permiten agregar footprints sueltos al PCB.

Por su parte, los “search paths” indican donde buscar los footprints asociados a cada componente del esquemático. Los “search path” se definen desde “Options -> Search Pthas” y se pueden definir como rutas absolutas o como rutas relativas al proyecto.

Notar que las bibliotecas instaladas son una configuración de global de Altium mientras que las rutas de búsqueda son propias de cada proyecto.

### Ventajas y desventajas

La ventaja de usar este flujo de trabajo es que se puede usar un único set de bibliotecas comunes a todos los proyectos siendo más fácil en mantenimiento y actualización de las mismas. **Esto se adecua más a la filosofía que se pretende implementar en el sector de Hardware de Plantium.**

La desventaja es que se debe asegurar una correcta configuración de cada puesto de trabajo (instalación de las bibliotecas correspondiente) antes de trabajar con los proyectos.

## Bibliotecas Integradas en Altium (Integrated Libraries)

El uso de bibliotecas integradas se puede considerar como una variante del caso anterior con la diferencia que se instala una única biblioteca (un único archivo) que contiene todos los símbolos de esquemáticos y sus footprint asociados.

Las bibliotecas integradas compilan en un único archivo \*.IntLib todos los símbolos de componentes de todas las bibliotecas \*.ShcLib y \*.PcbLib junto con cualquier otra información asociada a cada símbolo. De esta forma una única biblioteca contiene todos los componentes utilizados para todos los proyecto y solo esa biblioteca debe ser instalada.

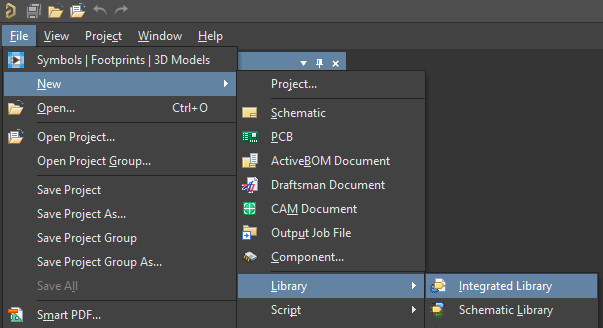
Ventajas de utilizar bibliotecas integradas

* Toda la información está atomizada en único archivo.
* No son editables, lo que evita introducir cambios a la misma por error. Para editarla se deben editar las bibliotecas fuentes y volverlas a compilar.
* Durante el proceso de compilación se detectan errores de diseño como ser discrepancia en cantidad o número de pines entre los símbolos y los footprints.

**Se propone como metodología de trabajo para el equipo de hardware de Plantium utilizar una biblioteca integrada que contenga todas las bibliotecas creadas por el área y dar mantenimiento solo a esta biblioteca.**

### Creación de un proyecto de biblioteca integrada

Los paquetes de bibliotecas integradas (Integrated Library Package \*.LibPkg) son un tipo particular de proyecto de Altium. Un nuevo paquete integrado se crea desde File->New->Library->Integrated Library.

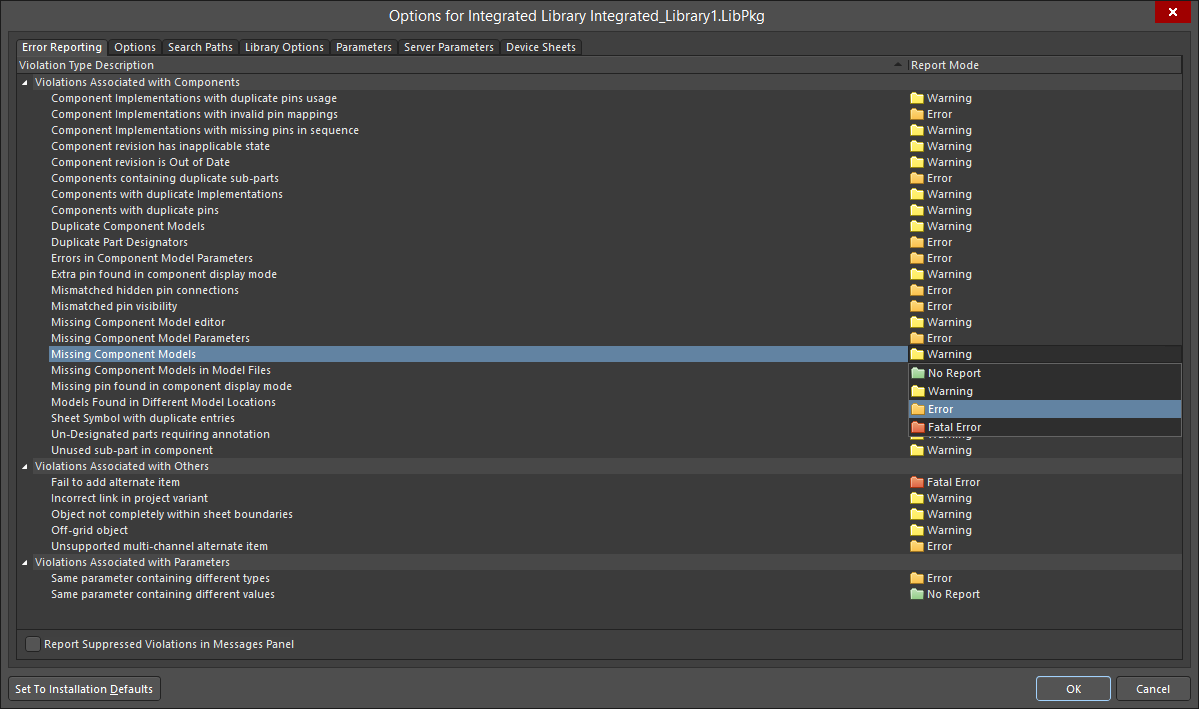


El proyecto creado es un proyecto vacío al que se debe incluir las bibliotecas \*.SchLib y \*.PcbLib que se quieran compilar.

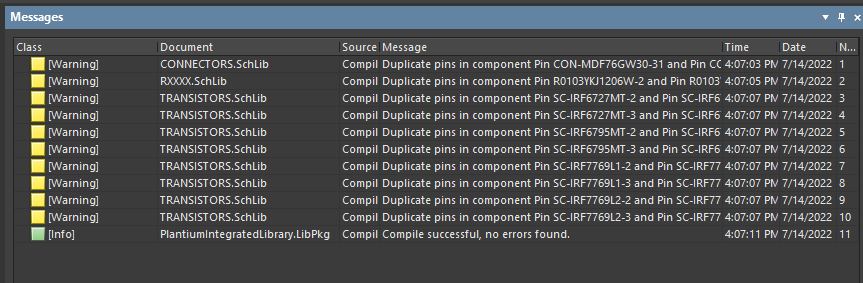
En un proyecto \*.LibPkg las bibliotecas se incluyen de igual manera que un proyecto \*.PcbPrj, tal como se explicó en el apartado “Bibliotecas incluida al proyecto”. También son válidos los comentarios que se hicieron respecto de la inclusión de bibliotecas que se encuentren dentro o fuera del directorio del proyecto.

### Compilación de la biblioteca integrada

Previo a realizar la compilación de la biblioteca es recomendable revisar la configuración de reportes desde “Options-> Error Reporting->Violations Associated with components” para asegurarnos una correcta configuración.



La biblioteca integrada es generada solo si durante el proceso de compilación no producen errores y es por esto que una correcta configuración del Error Reporting es esencial. Aun cuando la compilación es exitosa es altamente recomendable revisar la solapa Messages en busca de cualquier anomalía que pueda existir

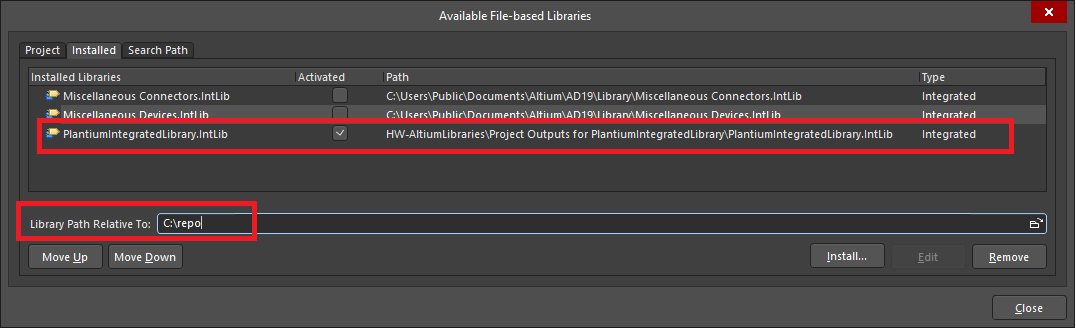


El proceso de compilación crea automáticamente el directorio “Project Outputs for MyProjectName” y dentro crea el archivo resultante de la compilación “MyProjectName.IntLib” que es la biblioteca integrada propiamente dicho.

### Instalación de una biblioteca integrada

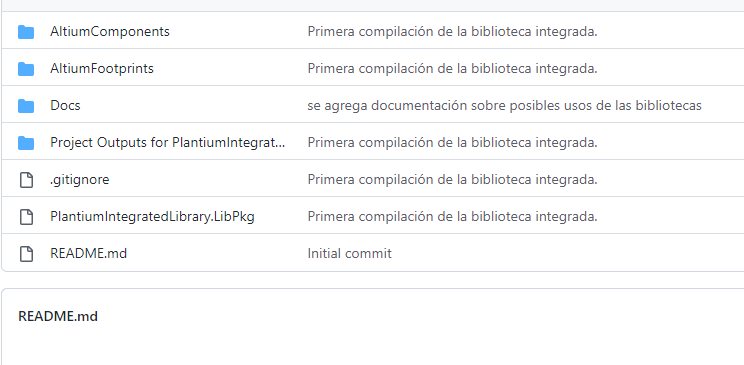
Las bibliotecas integradas \*.IntLib se instalan de igual manera que las \*.SchLib como ya explicó previamente. También es válido lo explicado respecto de la instalación con rutas absolutas o relativas.

**Se recomienda instalar la biblioteca integrada usando una ruta relativa al repositorio que la contiene.**

****

### Repositorio (GIT) HW-AltiumLibraries (Plantium)

En nuestro caso se creó un repositorio que contiene el proyecto de Biblioteca integrada “PlantiumIntegratedLibrary.LibPkg” junto con todas sus dependencias para asegurar la integridad del proyecto cada vez que este se clone en otra estación de trabajo. Con esta filosofía el repositorio creado centralizará el mantenimiento de todas las bibliotecas utilizadas por el sector.



**Como se mencionó previamente, no es posible mantener de forma consistente el repositorio de un proyecto \*.LibPkg que incluya bibliotecas externa al directorio del proyecto.**

### Migración de proyectos existentes

Para los proyectos existentes que posean la estructura de librerías anterior (basadas en los repositorios separados de componentes y footprints) se puede hacer una migración directa de todos los componentes. La clave de esta migración está en que el *Design Item ID* no cambia entre los dos modos de manejar las librerías.

En principio se deben seleccionar todos los componentes del circuito (se puede seleccionar todo y filtrar sólo los componentes), y luego seleccionar “PlantiumIntegratedLibrary.IntLib” en el campo *Source* de las propiedades. Nos aparecerá un cuadro de diálogo donde se consulta el tipo de operación deseada:

* Clean: Elimina el ID para luego poder reemplazarlo manualmente. Es la opción recomendada por el programa, pero lleva más trabajo y hay que hacerlo de manera individual.
* Do not change: Mantiene el ID. Resulta útil en nuestro caso porque los nombres se mantienen, pero puede llevar a errores si no existe el ID o está asociado a un componente distinto.
* Select: Permite seleccionar los componentes de manera manual en cuadros de diálogo (arrojará un cuadro de diálogo por cada componente).

La mejor opción para los proyectos que ya se estaban trabajando con librerías unificadas es *Do not change*, que permite cambiar todo junto. Si no, conviene elegir *Clean* y luego asignar manualmente los componentes más adelante, basándose en el *Manufacturer Part Number*.

Posteriormente se deben guardar los cambios en el esquemático y en el proyecto.

Por último, desde el archivo de PCB se debe seleccionar “*Design -> Import Changes from .PrjPcb*” y en el cuadro de diálogo validar los cambios que sean necesarios (en librerías, parámetros de componentes, etc) y ejecutarlos. Si no aparecen cambios deseados, de todas maneras, al cerrar el cuadro de diálogo se actualizarán los links de los componentes del PCB.

## Bibliografía

Este instructivo está basado en la documentación oficial de Altium

* <https://www.altium.com/documentation/altium-designer/working-with-integrated-libraries?version=19.0>
* <https://www.altium.com/documentation/altium-designer/understanding-file-based-models-components-libraries?version=19.0>
* <https://www.altium.com/documentation/altium-designer/project-options-search-paths?version=21>