PCP AUDIO

PROCEDIMIENTO DE DISEÑO Y REVISIÓN DE SCHEMATICS

Versión:

V1.0.2

Índice

Introducción....2
Guía de estilo....3
Plantilla....3
Diseño jerárquico y multi-channel....3
Nombrado de las nets....6
Uso de la cuadrícula....7
Uso de los componentes pasivos genéricos....7
Procedimiento de diseño 8
Diseño en dos fases....8
Estudiar diseños de referencia....8
Simulación....8
Validación por bloques....8
Diseño preventivo....9
Procedimiento previo a la captura de schematics....10
Acciones sobre el control de la potencia y calor:....10
Acciones para componentes pasivos:....10
Resistencias:....10
Condensadores:....10
Bobinas:....10
Acciones para componentes activos:....11
Acciones en los pines y nets....11
Acciones asociadas a los conectores:....12
Acciones sobre el BOM....13
CHECKLIST de revisión....14
Control de versiones....15

Introducción

Este documento define un método estandarizado de revisión de schematics con la finalidad de asegurar la calidad y la correcta operación de los diseños.

Describe unas reglas de estilo del diseño, una metodología de revisión y como evolucionar este documento para cubrir los nuevos errores recurrentes que se van detectando

Es sumamente importante cumplir la lista ya que cuando un error pasa a la siguiente fase se multiplica el coste de solucionarlo, en tiempo y en dinero.

Es obligatorio leer la sección "ventajas" en cada apartado,

Guía de estilo

Plantilla

Se debe partir de una plantilla de proyecto que incluye los elementos comunes a todos los proyectos.

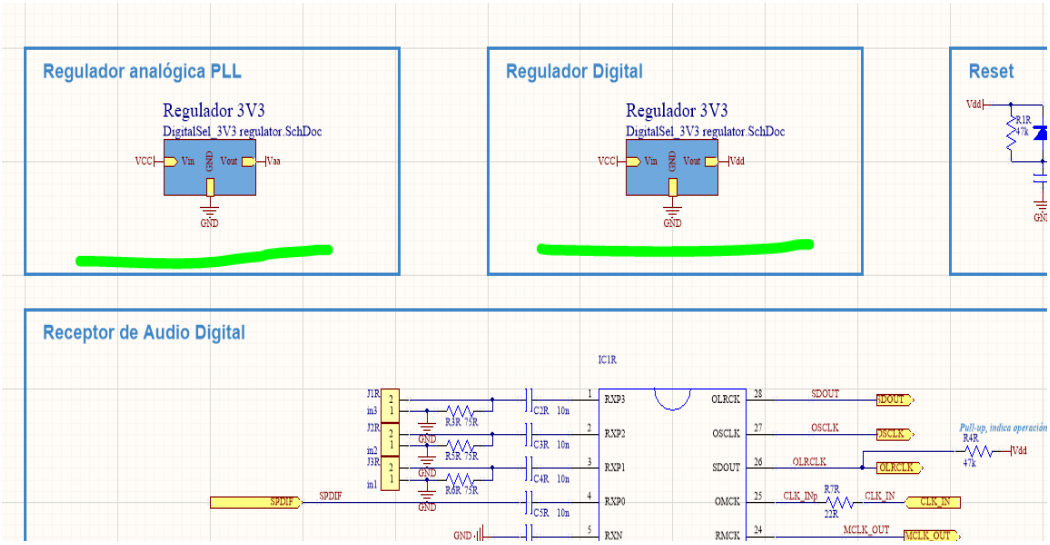
Ventajas:

- Ahorrar trabajo no repetiendo los mismos pasos cada vez.
- Estandarizar el entorno para que sea igual en todos los proyectos y facilite la búsqueda y el entedimiento.
- Crear un método en el que se puedan ir corrigiendo los errores ya conocidos y no repetirlos, y aplicando mejoras de forma estructurada para que no colisionen con prácticas anteriores.

Diseño jerárquico y multi-channel

Se deben emplear las *sheets* para crear diseños jerárquicos, donde una *sheet* puede contener otras más dentro.

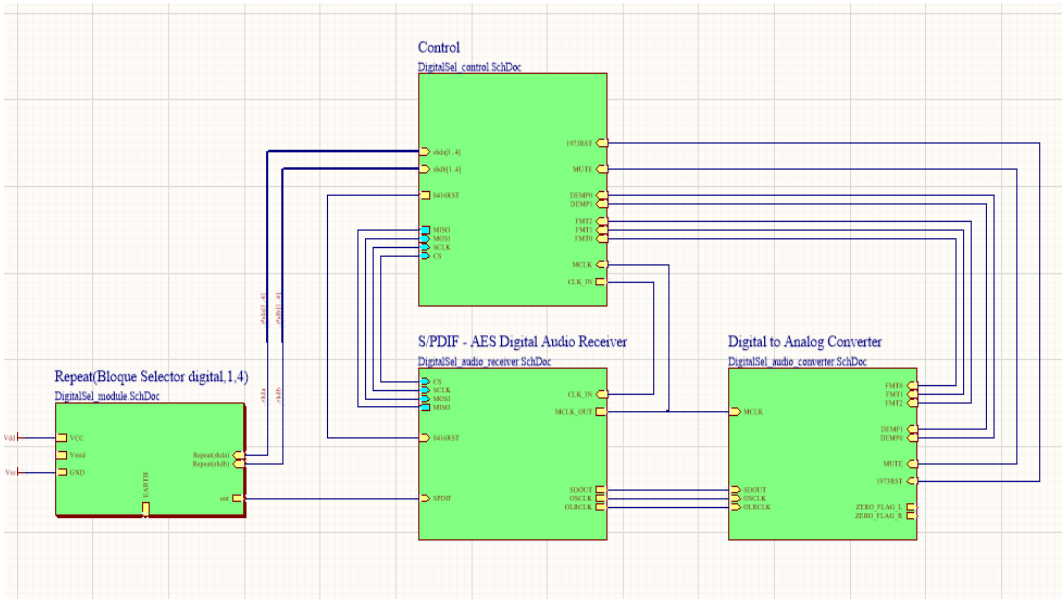
Cuando un bloque se repita, aunque sólo sea una vez, se debe crear una *sheet*.



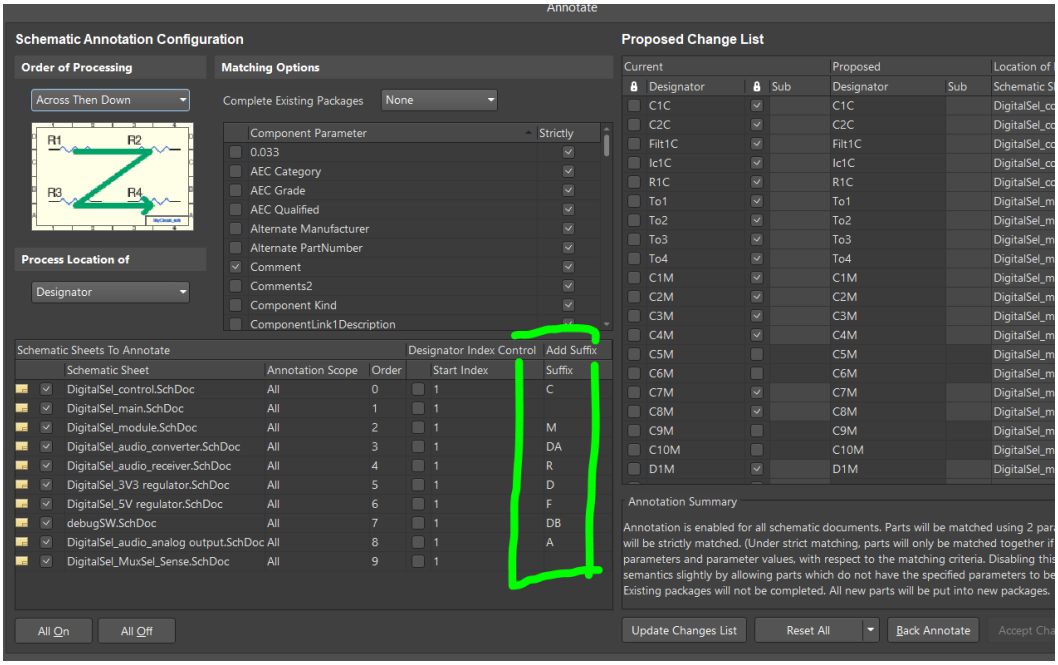
Las entradas de las **sheets** y los **puertos** deben estar **sincronizados** (d+p – *Design-> Synchronize Sheet Entries and Ports*). Esta herramienta es realmente útil y ahorra errores y trabajo. Se puede usar para añadir y quitar *ports* y *sheet entries*.

Si el bloque se repite de manera múltiple y relacionada (como los canales de una mesa de

mezclas, se debe usar diseño *multi-channel* (ver comando *REPEAT*).

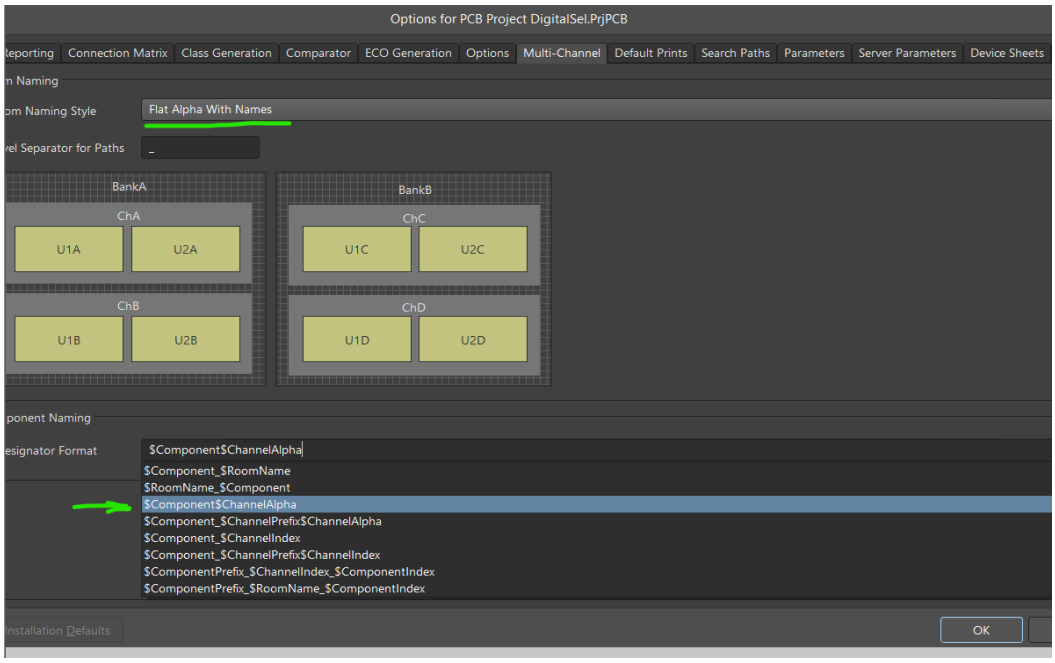


Se debe emplear la anotación automática (*Tools->Annotate Schematics*) con subfijos alfabéticos, nunca numéricos, y cada hoja debe llevar su subfijo distintivo.

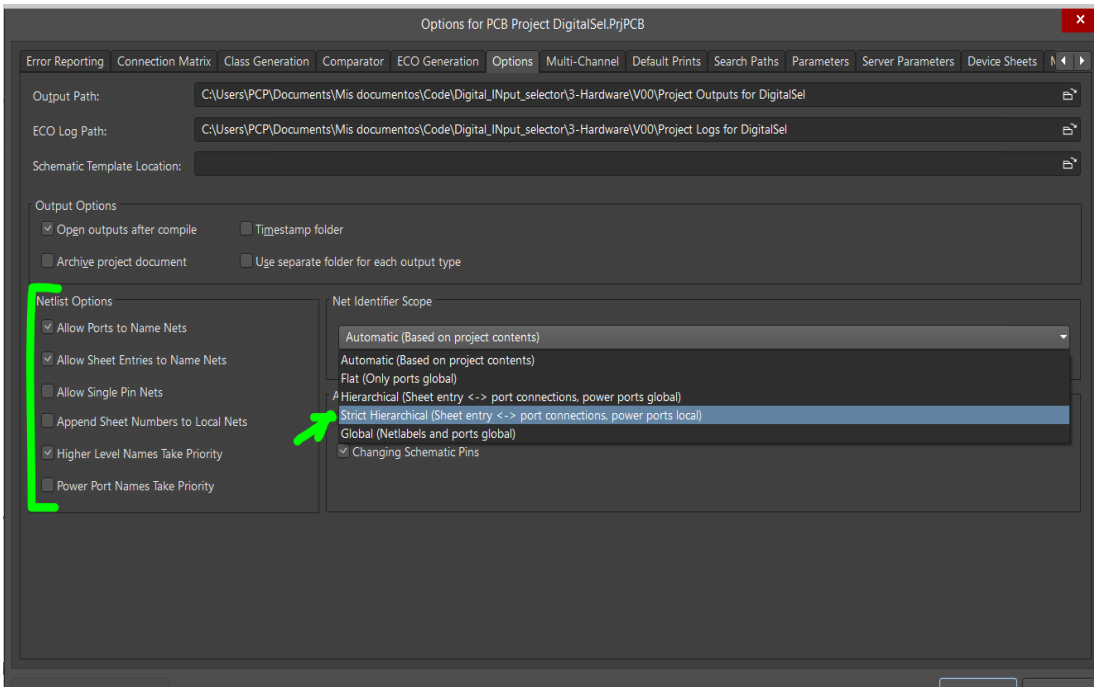


Se deben usar las opciones alfanuméricas en las opciones de diseño multichannel en

Project->Projecto Options->Multi-Channel.

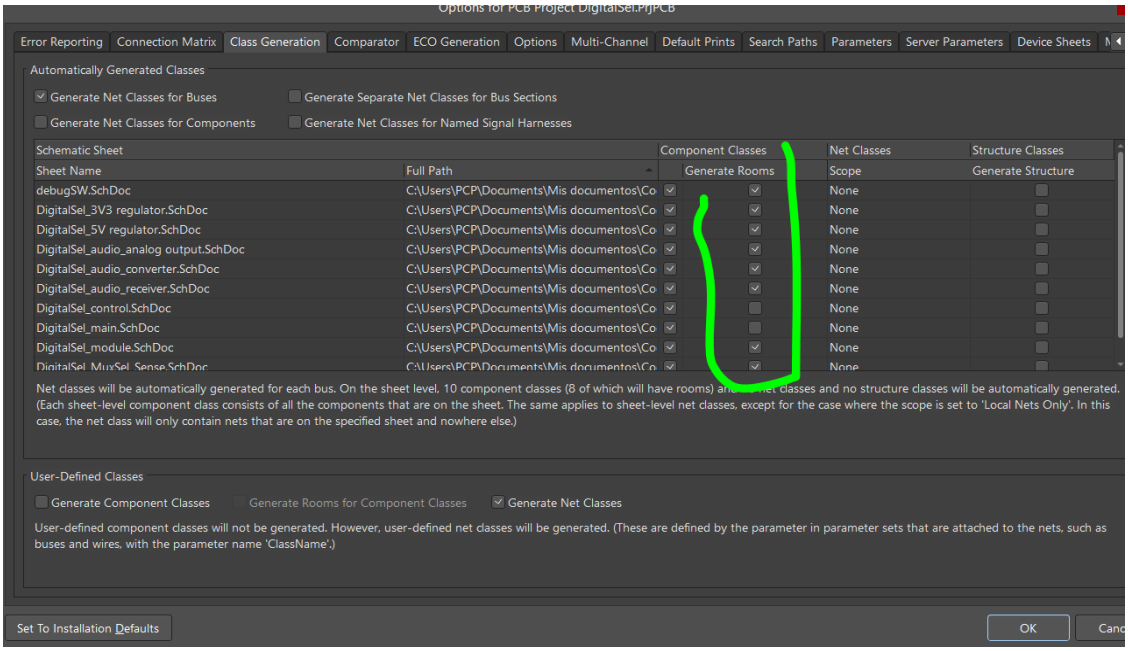


Se deben usar las opciones de nombrado jerárquico en *Project->Projecto Options->Multi-Channel.*



Se deben crear rooms asociadas a cada diseño muti-channel. Luego esto nos permitirá

replicar el diseño de PCB de cada bloque y ahorrar mucho trabajo.



Ventajas:

- El diseño multisheet permite un diseño modular por bloques lógicos, pudiendo profundizar en el diseño cuando es necesario.
- Permite reutilizar bloques como una entrada USB, un bloque de conector para debug, ... (que ya han sido probados) de forma cómoda, segura y eficiente.
- En el caso del diseño multichannel, donde se repiten las estructuras, es sumamente útil poder hacer cambios en sólo un diseño y que este cambio se replique a todos los demás. También ahorra errores potenciales.
- En el caso del rutado para componentes multichannel las rooms permiten copiar el placement y el routado de un bloque a otro, con las mimas ventajas que el punto anterior trasladadas al PCB.

Nombrado de las nets

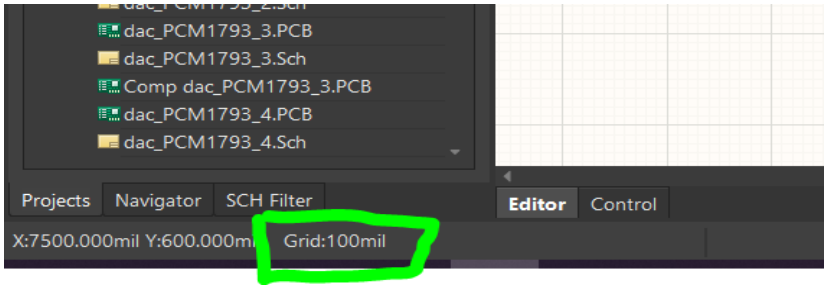
Todas las *nets* que impliquen un trazado con una relevancia no nula deben tener asignado un nombre.

Los pares diferenciales deben nombrarse (y deben usarse los subfijos _N y _P para que Altium los reconozca). En el caso de X-signals, se pueden usar subfijos _1, _2,.. _n para indicar los distintos tramos.

Para esto se debe usar el comando **p+n** -*Place->Net Label*.

Uso de la cuadrícula

Se debe usar la cuadrícula ajustada a 100 mils. En casos especiales se puede usar ajustada a 50 mils.



Nunca se debe prescindir de ella porque puede generar errores en las conexiones. Puede parecer que una net está conectada y no estarlo.

Uso de los componentes pasivos genéricos

Los componentes comunes, resistencias en sistemas digitales, condensadores de desacoplo, ferritas, tienen un coste inferior al 1% dentro del diseño total pero garantizan la correcta operación y que el diseño cumplirá con los criterios de seguridad e inmunidad EMC requeridos. Jamás se debe escatimar en su uso.

Particularmente, se deben incluir resistencias de terminación en las líneas de alta velocidad.

En estos componentes se debe especificar:

- Tipo de componente (R,L,C,F,...)
- Valor
- Material*
- Voltaje/corriente*
- Tolerancia

* Si procede, como es el caso de los condensadores o inductancias.

TAREA: meter todas estas reglas en la plantilla de proyecto de Altium

cosa 2

...

Procedimiento de diseño

Diseño en dos fases

No es realista pensar que un diseño va a funcionar a la primera.

Se debe considerar un diseño en dos fases, en la primera se diseña un prototipo, y sobre ese prototipo se tendrán que realizar correcciones artesanales para buscar solución a los posibles errores que tenga. Esos errores se deben anotar y documentar para crear la versión definitiva.

Si el error es recurrente, se debe incluir en este documento, bien en la sección de "Procedimiento previo a la captura de schematics" o al "checklist de revisión".

Estudiar diseños de referencia

Es muy recomendable estudiar los diseños de referencia que puedan ofrecer los fabricantes de chips.

Se deben evaluar los diseños principalmente en los datasheets, y si el circuito es más avanzado, en las hojas de aplicación que se puedan encontrar.

Ventajas: Hay más probabilidades de éxito usando un circuito ya probado.

Simulación

Es recomendable simular el diseño antes de realizar ninguna otra acción. Es el método más rápido y económico de validar un diseño y prever posibles errores.

Ventajas: Aunque no se vea su utilidad en un principio, si hay problemas, simular será una de las acciones necesarias y habrá ahorrado tiempo y estrés.

Validación por bloques

Si el circuito es especialmente complejo o es radicalmente nuevo, es conveniente validarlo por partes.

Las opciones son:

- Montar un bloque en placa de prototipos
- Comprar un módulo o placa de evaluación
- Encargar una PCB de 2 caras económica.

Se debe emplear la que menos tiempo de trabajo (no necesariamente total) consuma.

Diseño preventivo.

En conjunto con el diseño de la PCB, se deben realizar estas acciones preventivas por si fueran necesarias para solucionar problemas futuros o responder a necesidades nuevas.

- Dejar *pads* de acceso a pines libres de los microcontroladores.
- Además de usar testpoints, dejar puntos de conexión intermedia en las señales importantes.
- Usar puentes de estaño, o ferritas, o jumpers, para cortar la alimentación tras una fuente/ regulador, y poder probar por partes el circuito por separado y sin riesgo.
- Dejar puntos de inserción de alimentación externa.
- Prever varias topologías posibles. La correcta operación siempre es prioritaria.
- Dejar footprints para elementos de protección ESD aunque no se monten.
- Dejar footprints para LEDs indicadores de alimentación, aunque en fase de producción no se monten.
- En la fase de primer prototipo se deben extender estas acciones lo máximo posible.

Procedimiento previo a la captura de schematics

Se deben examinar los siguientes puntos antes de dibujar el esquema en Altium (captura).

Acciones sobre el control de la potencia y calor:

Se debe examinar el consumo de **corriente** de cada IC (ver datasheet o simular), sumar todas las corrientes que salen de un mismo regulador y comprobar que puede entregar la corriente necesaria.

Si es un **regulador lineal,** comprobar también su disipación de **potencia** y el **calentamiento** que tendrá, teniendo en cuenta la temperatura ambiente máxima y la resistencia térmica del encapsulado.

En circuitos lineales, se deben calcular todas las corrientes, tensiones, potencias y calentamientos. Es recomendable simular. Se puede usar (i*zq_click*+*alt*) sobre un componente en LTSpice para ver su consumo.

Acciones para componentes pasivos:

Resistencias:

Resistencia crítica: Por debajo de cierto valor (R=V^2/P), puede haber problemas de potencia. Por ejemplo, alimentamos a 5V, con resistencias 0603, que disipan 0,1W, esa resistencia crítica sería de 250 Ohm. Por debajo de ese valor, se debe examinar el consumo de todas las resistencias, en el peor caso posible.

Resistencias de terminación: Usar resistencias de terminación en los buses digitales, junto al *load*. Calcular el valor para no limitar demasiado los tiempos de rise y fall.

Condensadores:

Se deben comprobar que nunca, ni en el peor caso posibile, se supere la **tensión máxima** admitida. En el caso de los electrolíticos, esa tensión debe tener un márgen de seguridad para no comprometer su longevidad.

En los electrolíticos y los cerámicos de clase 2, la **alinealidad del dieléctrico** puede causar distorsiones, desviaciones en constante RC o pérdidas de capacidad. Comprobar que no afectan al funcioamiento del circuito.

Bobinas:

Se debe comprobar que nunca se supera la **corriente de saturación** de las bobinas, incluyendo la corriente RMS + corriente de pico.

Si no son apantalladas, se debe comprobar cómo se produce su **radiación magnética** y si puede afectar a algún circuito alrededor. Si tienen "punto", colocarlo en la tensión más constante (definen el exterior del bobinado)

Acciones para componentes activos:

• Buscar los **datasheet** de todos los componentes activos, guardar una copia en la carpeta del proyecto, y leer **la primera página** para conocer:

- Función
- Características
- Rangos de tensión, corriente y potencia
- Encapsulado + Símbolo + pinout
  - *NOTA: No todos los encapsulados, ni todos los modos de operación tienen el mismo pinout*

• Cuanto más complejo sea un componente activo, más hay que leer. Si es un **circuito integrado**, leer también las secciones:

- *Electrical Characteristics*
- *Recommended Operating Conditions*
  - VCC, VDD (Minimum and Maximum operating voltage),
  - Vin High / Low (Low and High input levels),
  - Operating temperature,
  - Maximum required current
- *Functional overview* – Atención especial a:
  - Esquemas de referencia
  - Tablas
  - Ecuaciones

• Crear el símbolo siguiendo el documento "Procedimiento de diseño de *symbols* y *footprints*".

Acciones en los pines y nets

- En los pines y nets de alimentación se debe comprobar:
  - El margen de tensión que admite y que nunca se superará la tensión máxima.

  - Que tiene los condensadores de desacoplo, filtros o ferritas que necesita.
- Si el pin es usado durante el proceso de grabación, asegurarse que tras el reset se configure según las necesidades.
- Nombrar las nets importantes para ayudar al proceso de ruteado. Es recomendable que el nombre del pin sea parecido/igual al nombre de la net.
- Pull-Up's y pull-Down's
  - Asegurarse de que hay uno, y sólo uno, *pull-down* o *pull-up* en cada net que lo requiera.
  - Si es un pin de configuración, leer la tabla de configuración y asegurarse de que tiene el valor correcto.
  - Asegurar que los pull-up están conectados a la tensión correcta.
  - A veces un pull up/down interno se habilita por defecto. Asegurarse de:
    - No hay dos *pull-up* haciendo lo mismo.
    - Que ese *pull-up* interno no lucha contra un *pull-down* externo, lo que daría una tensión intermedia.
  - Si el pin es un *open drain* asegurarse que tiene su *pull-up*.
  - Durante los arranques, los GPIOs pueden no estar inicializados hasta que arranca el software. Esas tensiones indeterminadas pueden generar problemas y se debe usar un *pull-up* o *pull-down*.
- En los buses *UART/USART*, asegurarse que el TX del *master* está conectado al RX del *slave* y viceversa. Se recomienda usar flechas guión-mayor "->".
- Asegurarse de que todos los pines de una *net* funcionan al mismo voltaje: Niveles CMOS/TTL, etc...
- En los pines *Enable, R/~W,* comprobar que se emplea el nivel lógico (H/L) adecuado para habilitar/leer/escrbir.
- Definir *netclasses* para poder crear reglas como
  - Ancho de pistas de potencia.
  - *Clearance* de alto voltaje.
  - *Matched lenght* en buses.
  - Z nominal.

Acciones asociadas a los conectores:

- Con conectores estándar, buscar la especificación del *pinout* del conector y

comprobar que el esquemático es correcto.

- Intercalar alimentación y GND entre las pistas de señal. Si no es posible, poner alimentación y GND en los extremos opuestos.
- Emplear preferiblemente conectores que sólo entran en una posición (*poka-yokes*) Si no es posible, se debe plantear una protección o una distribución de pines que permita mantener la integridad del circuito cuando se conecte mal. Se debe dar por hecho que alguien la conectará mal.

Acciones sobre el BOM

Valores prioritarios. Cuando hay valores de resistencias o condensadores donde hay mucho márgen para elegir, se debe elegir un valor usado con mucha frecuencia.

Optimización del BOM: Cuando se haya terminado el diseño, se debe generar un BOM, para analizarlo buscando los componentes de los que sólo hay una o unas pocas unidades.

Puede haber componentes heredados de otros diseños, en los que no tiene mucha importancia usar un valor u otro (pines de configuración, pulldowns). En ese caso es preferible unificar el valor de los componentes y eliminar una línea del BOM.

También para condensadores de desacoplo, cuando se unen varios esquemáticos, que tengan distinta referencia, tensión nominal, o encapsulado, pero sea perfectamente posible unificarlos, se debe hacer.

El funcionamiento del circuto siempre es prioritario, pero si es posible unificar valores, se debe hacer para reducir la cuenta total de componentes y usar la máxima cantidad posible del mínimo número de de componentes distintos.

CHECKLIST de revisión

Tras la captura del *schematic*, comprobar el siguiente checklist.

Se deberá registrar este checklist como documento aparte, junto a los documentos de validación de diseño, en la carpeta de "1-Doc Pruebas"

- ☐Se han **leído** los **datasheets.**
- ☐Se han hecho los **cálculos** de **corriente** y potencia.
- ☐El diseño se **compila** y no da **ningún error** (c+c – *Project-> Compile PCB proyect ...*)
- ☐Optimizacíón del BOM:
  - ☐Generar un BOM, examinar todos los componentes que tengan una cantidad baja, y ver si se pueden unificar.
  - ☐Componentes con la misma función: ver si se pueden unificar
- ☐Mover cada componente para comprobar que **las nets están conectadas** correctamente a él (i*zq_click+mover ratón; salir con ESC*)
- ☐Comprobar que cada net esta trazada correctamente, señalar una por una en "*proyecto*.PrjPcb" -> nets.
- ☐Todos los **componentes** que se usan ya han sido **verificados**. Si no han sido verificados se verifica la correspondencia entre pines del símbolo y de su footprint.
- ☐Todos los IC's están conectados a la **alimentación correcta**.
- **Pull-up**'s y **pull-down**'s –
  - ☐Todas las net que los necesitan tienen su PU/PD correspondiente y sólo uno.
  - ☐El valor de la resistencia es el correcto.
  - ☐No existen entradas digitales al aire.
- ☐En los buses UART/USART, asegurarse que el TX del maestro está conectado al RX del esclavo y viceversa.
- ☐Existen **testpoints** en las nets importantes.
- ☐Se han **definido** correctamente las **netclass** para generar en PCB *rules* de potencia, alto voltaje, Z nominal..

Control de versiones

versión	Cambios	fecha
'0.0	Versión inicial del documento-("Plantilla de documento.odt")	01/09/19
'1.0	Creación de PROCEDIMIENTO DE REVISIÓN DE SCHEMATICS	11/09/19
'1,0.1	Se añade CHECKLIST de revisión	01/12/19
'1,0,2	Se completan textos y se añade "Procedimiento de diseño "	29/12/19
'1.0.3	Se añaden las casillas a marcar al checklist de revisión. Se genera documento dependiente "Checklist de revisión de SCH"	18/01/20
'1,0,4	Se añade el punto de optimización del BOM al checklist	19/01/20