

I2S et DSPi_y

16/05/2015

L'I2S est un bus dédié à l'audio au format série synchrone avec horloges.

Il s'utilise pour communiquer entre ICs d'un même PCB ou différents PCB dans un même appareil.

Pas pour communiquer entre différents appareils.

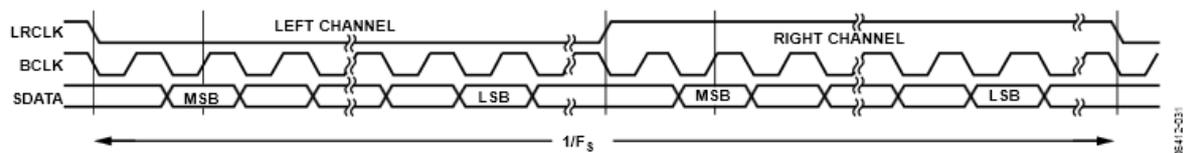
Bus à 3 fils :

Sdata : la valeur de chaque sample sous forme sérielle. (un bit après l'autre)

BCLK : bit clock qui est l'horloge pour la ligne data pour donner la cadence.

LRCLK : sélecteur de word qui permet de déterminer si c'est le canal droit ou gauche qui est en cours d'envoi.

Voici ce à quoi cela ressemble : (extrait du datasheet de l'ADAU1701)



In/Out, Master/Slave :

Ce bus est omnidirectionnel, on communique donc d'un Out vers un (ou plusieurs) In.

Là où tout se complique, c'est que cette règle n'est appliquée qu'à la ligne Sdata.

Le flux de donnée pouvant être synchronisé soit par le Out soit par le In.

C'est là qu'intervient la notion de Maître/ Esclave.

Le maître demande et l'esclave répond.

Prenons le cas d'un DSP, il va recevoir un signal en In, le traiter et le retransmettre en Out.

Les différentes combinaisons possibles :

Il est Master In :

Dans ce cas, il va générer les signaux d'horloges BCLK et LRCLK, qui seront donc des sorties, à sa fréquence d'échantillonnage, la source connectée enverra les données sur Sdata synchronisées sur ces horloges.

Il est Slave In :

Dans ce cas, les 3 lignes du bus sont des entrées. La source va envoyer les données sur Sdata et les horloges selon sa propre fréquence de sampling.

Il est Master Out :

Dans ce cas, les 3 lignes du bus sont des sorties, le DSP va envoyer (généralement à un DAC) les signaux d'horloges et le Sdata à sa fréquence de sampling.

Il est Slave Out :

Dans ce cas, c'est le DSP qui va recevoir les signaux d'horloges et transmettre les données sur Sdata à la fréquence de sampling demandée.

C'est pas vraiment compliqué, mais il y a des pièges.

Il faut toujours un seul Master, logique sinon on aura 2 ICs qui enverront leurs horloges sur les mêmes lignes avec le risque détériorer définitivement ces ICs.

Il y a dans certains cas le problème que les ICs doivent être synchro entre eux via le MasterClock, on y reviendra.

L'I2S en config standard du DSPi :

Sans l'ADAU1452 on a le chemin suivant.

Le CS8422 dont le rôle est de décoder les entrées SPDif ou I2S in venant d'une carte externe du style Amanero.

Si l'on a sélectionné USB/I2S comme source, le CS8422 sera Slave In et Slave Out.

Amanéro (ou autre) enverra donc Sdata et horloges au CS8422. Via son ASRC il va re-sampler le signal et le renvoyer sur son Sdata-Out vers le DSP en synchronisation avec ce dernier.

Si l'on sélectionne SPDif comme source, ce SPDif sera décodé et renvoyé de la même manière de manière synchrone avec les horloges du DSP.

Le DSP, ADAU1701 serait donc Master In, mais aussi Master Out.

Raté, il est Slave In et Master Out.

Une particularité qui est hardware, les signaux d'horloges BCLK-Out et LRCLK-Out du DSP sont connectées à ses propres lignes BCLK-In et LRCLK-In.

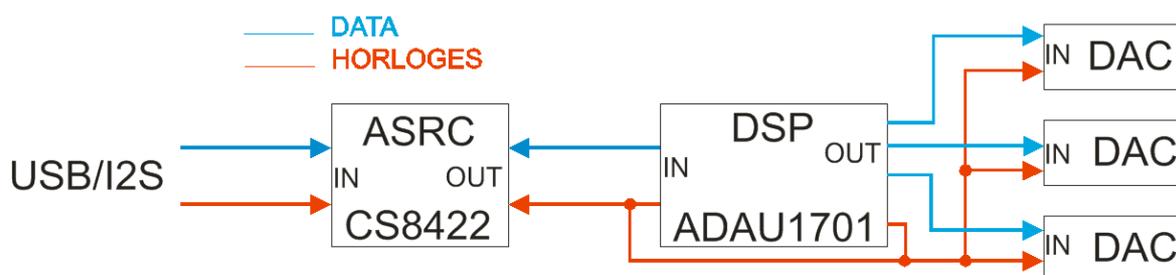
Ce qui fait que tout en étant configuré en Slave In, il va recevoir, tout comme le CS8422, ses signaux d'horloges Out et le Sdata du CS8422.

Cela permet d'avoir In et Out synchrones et de contourner le fait que le petit ADAU1701 ne dispose pas de toutes les options I2S tels qu'un ADAU1452 par exemple.

On peut donc dire qu'il se comporte comme une Master In et Master Out.

Après, c'est plus simple, le DSP envoie en tant que Maître Out ses horloges aux 3 DACs et les datas via 3 bus I2S (dont les horloges sont communes) aux DAC via Sdata1, Sdata2 et Sdata3.

Voici sur un schéma :



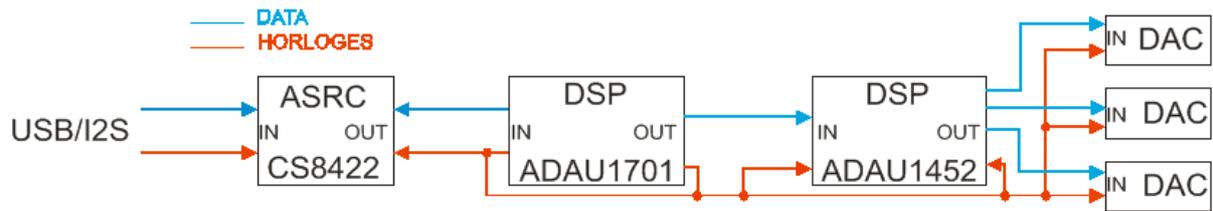
L'I2S avec l'ADAU1452 en crossover :

L'ADAU1452 est plus évolué que le 1701, il intègre en outre des ASRC et plus de fonctions de configuration et de routage interne. Il peut ainsi être indépendant du Master Clock, mais on parle plus tard.

Dans cette config, les datas I2S OUT de l'ADAU1701 sont mis en haute impédance, donc similaire à non connecté.

Un quatrième port I2S Out de cet ADAU1701 va envoyer le signal en Master vers l'ADA1452 en Slave In qui lui-même va envoyer en Slave Out les datas aux DAC.

Les horloges étant toujours celles de l'ADAU1701. C'est donc un schéma qui sort légèrement des sentiers battus que voici :



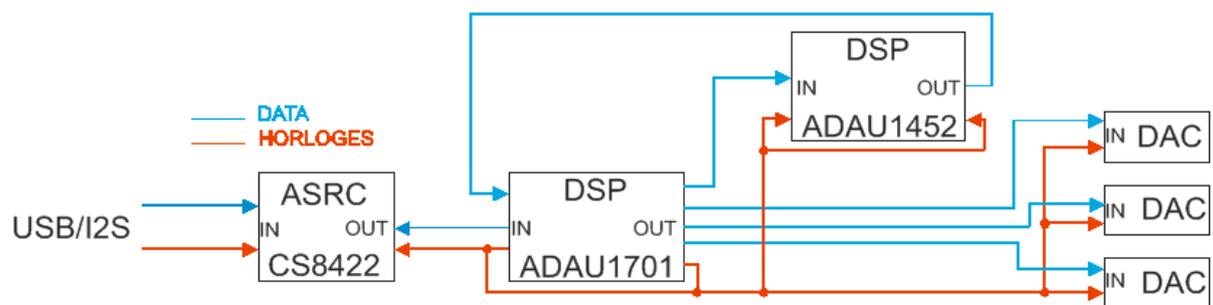
L'I2S avec l'ADAU1452 en Global :

Sans aucune modification hardware, il est possible de créer un routage différent.

Dans ce cas l'ADAU1452 vient « s'insérer » dans le 1701.

Il est en Slave IN pour recevoir le signal du 1701 et en Slave Out pour le lui renvoyer.

Ce qui donne :



Les extensions I2S de l'ADAU1452 :

Sur le PCB on trouve 2 connecteurs I2S pouvant être configurés en Slave ou Master et pouvant être dirigés vers des ASRC pour gérer des fréquences de sampling différentes ou non synchrones avec celle du DSPi.

De la même manière on dispose d'un connecteur I2S Out.

Le Master Clock :

La majorité des ICs que l'on connecte en I2S vont réaliser un traitement digital.

Il en va de même pour un DAC pour faire la conversion digital-> analogique.

Ces traitements se font à fréquence plus élevée que le BCLK, c'est là qu'intervient le Master Clock qui est donc l'horloge qui fera « tourner » ces composants.

C'est pas toujours vrai, un DSP va certainement tourner plus vite mais alors sur un multiple du Master Clock et donc synchrone avec lui.

On trouve 2 groupes de fréquences possibles pour un Master Clock, soit un multiple de 44.1kHz : 11.2896MHz ou pour 48kHz 12.288 MHz.

Le DSPi fonctionne sous 24.576Mhz qui est le double de 12.288MHz.

Il est donc prévu pour utiliser des fréquences de sampling de 48,96 et 192kHz.

Transporter de telles fréquences devient très délicat, même sur un PCB.

Pour simplifier la conception, certains composants n'ont pas besoin d'un Master Clock synchrone avec le bus I2S, d'autres oui.

Dans le cas du DSPi :

Le CS8422 ne demande rien de particulier, en DSPi Version 1 avec la carte D-In, il dispose de son propre oscillateur. En DSPi V2 il est commun avec celui du DSP et donc synchrone.

Il en est de même pour les DAC ES9023.

L'ADAU1701, lui demande un Master Clock synchrone avec ses bus I2S.

Cela explique qu'il reste toujours le centre du système, l'ADAU1452 étant lui bien plus souple.

Via quelques résistances de pontage, il est possible de modifier la config du Master Clock entre ADAU1452 et ADAU1701. Mais sort du cadre de cette doc.

Bit perfect ?

Question qui devient à la mode mais rarement justifiée, quoique ...

On peut se dire :

Ma source sort du 48KHz, le DSPi ne devrait-il pas fonctionner à la même fréquence ?

Si la source est Master, elle va envoyer son 48KHz qui ne sera pas synchrone avec celui du DSP ni de toute manière exactement à la même fréquence que ce dernier.

Donc il faut de toute manière passer par un ASRC.

Si la source est Slave, oui, c'est possible vu que le Slave se synchronise avec le Master.

Mais dans ce cas il se peut tout de même que la source fasse du re-sampling à un autre niveau.

Conclusions :

L'I2S permet de nombreuses possibilités, mais il faut toujours bien voir avec les fiches techniques des composants utilisés pour éviter tout risques de non fonctionnement ou d'avoir des coupures, plocs...

Il existe aussi des alternatives, le Left-Justified, Right-Justified qui ressemblent à l'I2S mais n'en sont pas par de petites différences.

Enfin le mode TDM qui permet de faire passer plus que 2 canaux dans un bus, mais dans ce cas les fréquences des horloges deviennent plus élevées.

L'I2S a été développé par Philips, malheureusement les noms tels de BCLK ne sont pas standardisés, il faut donc parfois chercher l'équivalent.

Par exemple :

BCLK = SCK

LRCLK=WS

SDATA = DS

Ce document peut contenir des erreurs, l'auteur ne peut être tenu de dommages qui pourraient en découler.

L'usage de ce document est réservé à un usage non commercial.

