

L'USB en bref

Donner un sens au standard USB

Introduction

Commencer de nouveau avec l'USB peut être intimidant. Avec la spécification de l'USB 2.0 de 650 pages on pourrait facilement être découragé par la dimension absolue du standard. C'est seulement le commencement d'une longue liste de standards associés pour l'USB. Il y a des standards de Classes USB tel que la Spécification Classe HID qui détaille l'opération commune des appareils (claviers, souris, etc.) tombant sous la Classe HID (Interface Homme Machine) - Seulement 97 autres pages. Si vous concevez un Hôte (Host) USB, alors vous avez trois standards d'Interfaces Contrôleurs d'hôte à choisir. Aucun de ceux-ci n'est détaillé dans la spécification de l'USB 2.0.

La bonne nouvelle est que vous n'avez pas besoin de vous ennuyer à lire le standard USB en entier. Quelques chapitres ont été pondus par le marketing, d'autres parlent de couches bas niveau pris en charge par le circuit intégré contrôleur d'USB et une partie concerne les développeurs de Host et de HUB. Parcourons les différents chapitres de la spécification 2.0 et introduisons les points clés.

Chapitre	Nom	Description	Pages
1	Introduction	Inclut la motivation et le domaine pour l'USB. Le renseignement le plus important de ce chapitre est de faire référence aux Spécifications de la Classe USB. Aucun intérêt.	2
2	Termes et Abréviations	Ce chapitre s'explique de lui-même et est un mal nécessaire pour tout standard	8
3	Origine	Explique les objectifs de l'USB qui est Plug And Play et la simplification pour l'utilisateur final (pas pour le développeur). Il introduit les gammes de Vitesse Basses, Pleines et Hautes avec une liste de caractéristique venant tout droit du Marketing. Nul besoin de lire aussi ce chapitre.	4
4	Vue d'ensemble, architecture	C'est à partir d'ici que vous pouvez commencer à lire. Ce chapitre fournit une vue d'ensemble de base d'un système USB y compris sa topologie, le débit de données, les types de flux de données, les spécifications électriques de bases etc.	10
5	Modèle de flux de données de l'USB	Ce chapitre commence à parler de la manière dont les données transitent sur l'USB. Il introduit des termes tel que Terminaisons (endpoints) et "pipes" (tuyaux) puis passe la plupart du chapitre sur chacun des types de flux de données (Contrôle, Interruption, Isochrone et en Bloc). Tandis qu'il est important de connaître chaque type de transfert et ses propriétés, c'est un peu lourd pour une première lecture.	60

6	La mécanique	Ce chapitre détaille les deux connecteurs standards de l'USB. L'information importante ici est que le connecteur de type A est orienté en aval et que le connecteur de type B en amont. Par conséquent il devrait être impossible de brancher un câble dans deux ports amont. Tous Les câbles détachables doivent être Pleines et Hautes vitesses, tandis que tout câble basse vitesse doit être connecté durablement à l'appareil. A part un rapide coup d'œil aux connecteurs, vous pouvez sauter ce chapitre à moins que vous ne projetiez de fabriquer des connecteurs et/ou câbles USB. Les dessinateurs PCB peuvent trouver des empreintes de pas standards dans ce chapitre.	33
7	L'Electrique	Ce chapitre regarde la couche bas niveau des signaux électriques y compris l'impédance de la ligne, les temps de montée et descente, les spécifications de l'émetteur/récepteur et les niveaux de codage des bits, rembourrage de bits, etc. Les parties plus importantes de ce chapitre sont l'identification de la vitesse de l'appareil en utilisant une résistance pour polariser aussi bien la ligne de données et le Bus d'alimentation de l'appareil contre des appareils auto-alimentés. A moins que vous ne conceviez des émetteur-récepteurs USB au niveau silicium, vous pouvez feuilleter ce chapitre. Les bonnes documentations techniques des circuits USB détailleront quelles valeurs de résistances de terminaison vous aurez besoin pour une impédance de Bus adéquate.	75
8	La Couche protocole	Maintenant nous commençons à entrer dans les couches du protocole. Ce chapitre décrit les paquets USB au niveau de l'octet y compris la synchronisation, pid, l'adressage, l'endpoint, les champs CRC. Une fois que cela a été saisi, on passe à la couche suivante du protocole, les paquets USB. La plupart des développeurs ne voient pas ces couches inférieures du protocole car leur circuit USB s'en occupe. Cependant une compréhension des indications de situation et des échanges (Handshaking) en vaut la peine.	45
9	Cadre de travail du circuit USB	C'est le chapitre le plus fréquemment utilisé dans la spécification entière et le seul que je ne me suis pas lassé d'imprimer et de relire. Ce chapitre détaille l'énumération du Bus et les codes de demande (mise en place d'une adresse, obtention d'un descripteur, etc.) qui composent la couche du protocole la plus commune que les programmeurs et concepteurs USB devront voir. Ce chapitre doit être lu en détail.	36

10	USB Host (Hôte) Matériel et Logiciel	Ce chapitre couvre des questions concernant l'Host. Cela inclut la structure et la génération de microstructure, exigences du contrôleur de l'Host, mécanismes logiciels et le modèle de driver de l'USB. À moins que vous conceviez des Host, vous pouvez sauter ce chapitre.	23
11	La Spécification du HUB (moyeu)	Ce chapitre détaille les mécanismes des hubs USB y compris leur configuration, partage de transactions, descripteurs standards pour la classe du hub, etc. A moins que vous ne conceviez des hubs, vous pouvez sauter ce chapitre.	143

Donc maintenant nous pouvons commencer à lire les chapitres relatifs à nos besoins. Si vous développez des pilotes (Logiciels) pour les périphériques USB alors vous pouvez avoir besoin de lire seulement les chapitres,

- o 4 - Vue d'ensemble, architecture.
- o 5 - Modèle de flux de données de l'USB.
- o 9 - Cadre de travail du circuit USB.
- o 10 - USB Host Matériel et Logiciel.

Les concepteurs de périphériques électroniques peuvent seulement avoir besoin de lire les chapitres:

- o 4 - Vue d'ensemble, architecture.
- o 5 - Modèle de flux de données de l'USB.
- o 6 - La mécanique, et
- o 7 - L'électrique.

L'USB en bref pour les concepteurs de périphériques

Maintenant voyons les choses en face, (1) la plupart d'entre nous sommes ici pour développer des périphériques USB et (2) il est courant de lire la documentation d'un standard sans avoir une idée de la manière dont il faut se servir de l'appareil. Aussi dans les 7 prochains chapitres, nous nous concentrons sur les parties pertinentes nécessaires au développement d'un appareil USB. Ceci vous permet de comprendre l'USB et ces problèmes et vous guidera dans une recherche approfondie des problèmes spécifiques relatifs à votre application.

Le standard USB 1.1 était suffisamment complexe avant que la Haute vitesse soit introduite dans l'USB 2.0. Afin d'aider à la compréhension des principes fondamentaux de l'USB, nous omettons volontairement beaucoup de domaines spécifiques aux appareils Hautes vitesses.

Présentation du Bus Série Universel

L'USB version 1.1 comprenait 2 vitesses, un mode vitesse rapide de 12Mbits/s et un mode vitesse lente de 1,5Mbits/s. Le mode de 1,5Mbits/s est plus lent et moins sujet aux perturbations Electromagnétiques (EMI) réduisant ainsi le coût de perles de ferrites et des composants de qualité. Par exemple les Quartz peuvent être remplacés par des résonateurs meilleur marché. L'USB 2.0 qui est encore à la veille de voir le jour sur les ordinateurs grand public a fait monter les enchères jusqu'à 480Mbits/s. Le 480Mbits/s est connu sous le nom de mode Haute vitesse et a été créé pour entrer en compétition avec le Bus Série Firewire.

Les vitesses USB

- Vitesse Haute - 480Mbits/s.....**High Speed**
- Vitesse Pleine - 12Mbits/s.....**Full Speed**
- Vitesse Basse - 1,5Mbits/s.....**Low Speed**

Le Bus Série Universel est contrôlé par l'Hôte. Il ne peut y avoir qu'un Hôte par Bus. La spécification en elle-même ne comporte aucune forme d'arrangement multi-maître. Cependant la spécification "au pied levé" (On-The-Go specification) qui a été rajouté à l'USB 2.0 a introduit un protocole de négociation de l'Hôte qui permet à 2 appareils de négocier pour le rôle d'Hôte. Ceci est uniquement réservé à des connexions simples point par point tel qu'un téléphone mobile ou un organisateur personnel et non à un Hub multiple ou à des configurations d'appareils de bureau multiples. L'Hôte USB a la charge de mener à bien toutes les transactions et de programmer la bande passante. Les données peuvent être envoyées par différentes méthodes de transactions en utilisant un protocole basé sur un système de jetons (Token).

A mon avis la topologie du Bus de l'USB limite quelque peu. L'une des intentions original de l'USB était de réduire la quantité de câble à l'arrière de votre PC. Les gens d'Apple diront que l'idée est venue du Bus pour appareil de bureau de chez Apple, où le clavier, la souris et d'autres périphériques pouvaient être connectés ensemble (daisy chained ou connexions en guirlandes) en utilisant le même câble.

Cependant USB utilise une topologie en étoile à étages, qui ressemble à celle d'Ethernet base10. Ceci impose l'utilisation d'un Hub quelque part, ce qui sous-entend une plus grande dépense, d'avantages de boîtes sur votre bureau et d'avantages de câbles. Cependant la situation n'est pas aussi mauvaise qu'on le croit. Beaucoup d'appareils comprennent des Hubs USB. Par exemple votre clavier peut contenir un Hub qui est connecté à votre ordinateur. Votre souris et d'autres appareils tel qu'un caméscope numérique peuvent être branchés facilement au dos de votre clavier. Les moniteurs ne sont que d'autres périphériques sur une longue liste d'appareils qui comportent communément des Hubs intégrés.

Cette topologie en étoile à étages, comparé à des appareils à simple connexion en guirlandes (daisy chaining) comporte tout de même des avantages. D'abord l'alimentation de chaque appareil peut être contrôlée et même coupée si des conditions de surintensité se produisent sans perturber d'autres appareils USB. Les appareils à Haute vitesse ainsi que ceux à vitesse Pleine et Basse peuvent être maintenus alors que le Hub filtre les transactions de vitesse Haute et Pleine de façon à ce que les appareils à vitesse lente ne les reçoivent pas.

On peut connecter jusqu'à 127 appareils à un Bus USB à un temps donné. Vous avez besoin d'avantages d'appareils? Ajouter simplement un autre port ou un autre Hôte. Alors que la plupart des Hôtes USB précédents n'avaient que 2 ports, la plupart des constructeurs ont considéré ceci comme limitatif et commencent à introduire 4 et 5 ports Hôtes sur carte avec un port interne pour les disques durs etc. Les Hôtes précédents n'avaient qu'un contrôleur USB et ainsi les 2 ports partageaient la même bande passante USB disponible. Comme les exigences des bandes passantes ont augmenté, nous commençons à voir des cartes multi-ports avec 2 contrôleurs ou plus qui permettent des canaux individuels.

Les contrôleurs d'Hôte USB ont leurs propres caractéristiques. L'USB 1.1 comprenait 2 spécifications de contrôleur d'interface Hôte, UHCI (Contrôleur d'Interface d'Hôte Universel) développé par Intel qui met d'avantage le poids sur le logiciel Microsoft par exemple et qui permet d'avoir des appareils meilleur marché et le OHCI (Contrôleur d'Interface d'Hôte Ouvert) développé par Compaq, Microsoft et National Semiconductor qui met d'avantage la charge de travail sur l'appareil (Intel) et se contente de logiciel plus simple. Ce sont les relations typiques entre les ingénieurs qui fabriquent le matériel et ceux qui font les logiciels...

Avec l'apparition de l'USB 2.0 on a eu besoin d'une nouvelle caractéristique de contrôleur d'interface d'Hôte pour décrire les détails du niveau d'enregistrement spécifique à l'USB 2.0.

Le EHCI (Contrôleur d'Interface d'Hôte Accrue) était né. Les collaborateurs les plus importants comprennent Intel, Compaq, Nec, Lucent et Microsoft, il semble donc comme nous l'espérons qu'ils se seraient regroupés pour nous fournir une interface standard et ainsi un seul nouveau driver (pilote) à exécuter dans notre O.S. (Operating System ou système d'exploitation). Il était temps.

L'USB, comme son nom l'indique est un Bus Série. Il utilise 4 fils isolés dont 2 sont l'alimentation (+5V et GND). Les 2 restants forment une paire torsadée qui véhiculent les signaux de données différentiels. Il utilise un schéma d'encodage NRZI (Pas de retour à Zéro inversé) pour envoyer des données avec un champ *sync* de manière à synchroniser les horloges de l'Hôte et du récepteur.

L'USB supporte le système " plug'n play " branchement à chaud avec des drivers qui sont directement chargeable et déchargeable. L'utilisateur branche simplement l'appareil sur le Bus. L'Hôte détectera cet ajout, interrogera l'appareil nouvellement inséré et chargera le driver approprié pendant le temps qu'il faut au sablier pour clignoter sur votre écran assurant qu'un driver est installé pour votre appareil. L'utilisateur final n'a pas besoin de se soucier des terminaisons, de termes tel que IRQs et adresses de ports, ou de la réinitialisation de l'ordinateur. Une fois que l'utilisateur a terminé, on peut simplement retirer le câble, l'Hôte détectera cette absence et déchargera automatiquement le driver.

Le chargement du driver approprié sera réalisé en utilisant une combinaison PID / VID (Interface Produit Machine / Vendeur Machine). On peut se procurer le VID au forum des fournisseurs USB en payant, ce qui est considéré comme un autre point de blocage par USB. On peut trouver le catalogue des tarifs réactualisés sur le site web des fournisseurs USB.

D'autres organisations fournissent un VID supplémentaire pour des activités non commerciales tel que l'enseignement, la recherche ou le bricolage. Le forum des fournisseurs USB ne peut pas encore vous procurer ce service. Dans ce cas vous voudrez peut-être utiliser un appareil distribué chez le constructeur de votre système de développement. Par exemple la plupart des constructeurs de puces auront une combinaison VID/ PID que vous pouvez utiliser pour vos puces et qui n'est pas connu comme appareil commercialisé. D'autres constructeurs de puces peuvent même vous vendre un PID à utiliser avec leur VID pour votre appareil commercial.

Une autre caractéristique intéressante de l'USB réside dans ces modes de transferts. L'USB soutient des transferts de contrôles, d'interruptions, en Bloc et Isochrone. Lorsque nous examinerons les autres modes de transferts ultérieurement, nous nous rendrons compte que l'Isochrone permet à un appareil de réserver une approximation définie de la bande passante avec un temps d'attente garanti. Ce système se révèle idéal dans les applications Audio et Vidéo où l'encombrement peut susciter une perte de données ou une chute de trames. Chaque mode de transfert fourni au concepteur des compromis dans les domaines de la détection d'erreur et de la reprise, du temps d'attente garanti et de la bande passante.