Sciences et technologies
de l'Industrie et du développement durable

ET22 : Fonctionnements temporel et fréquentiel d'un système
ET228 : Transmission numérique

Document ressource

Transmission numérique entre un clavier et un ordinateur



Sommaire

Un peu d'histoire....2
Fonctionnement du clavier....4
Transmission de données entre le clavier et l'ordinateur....5

Un peu d'histoire

De la machine à écrire aux claviers actuels…

Pour plus d'information, consulter par exemple les sites suivants :

http://www.archivesnationales.culture.gouv.fr/camt/fr/se/fiche3/fiche3.html

http://dispourquoipapa.free.fr/histoire/hi0021.htm

http://fr.wikipedia.org/wiki/Clavier_d'ordinateur#Historique

http://www.commentcamarche.net/contents/pc/clavier.php3

« C'est un imprimeur, Latham Sholes, qui en 1873 aux Etats Unis conçoit la première « type-writer » dont il dépose le brevet. Il signe un contrat d'association avec le fabricant d'armes Remington pour une fabrication industrielle. La première machine sort des usines Remington en 1874. Très décorée, elle ressemble à une machine à coudre. »







Premières machines à écrire *Latham Sholes*

« Cette machine était constituée d'un clavier qui, lorsque l'on appuyait sur les touches, faisait se lever une tige munie d'une lettre en relief. Cette lettre tapait une bande imbibée d'encre placée devant la feuille de papier. La lettre correspondante s'imprimait donc sur la page par ce simple procédé mécanique. Il s'avérait que les utilisatrices des machines à écrire tapaient trop vite. Certaines tiges se levaient en même temps et bloquaient. Sholes, en 1868, eut alors l'idée de séparer, de part et d'autre du clavier, les lettres fréquemment utilisées en langue anglaise comme le Q, le R, E, le W, etc. Ainsi, les tiges correspondantes s'emmêlaient moins lorsque la frappe était rapide. Cela donna le clavier QWERTY. Une simple adaptation à la langue française, et le clavier

AZERTY que l'on connaît était né. En fait, la disposition que nous connaissons n'est purement due qu'à un problème mécanique au détriment de l'ergonomie. »

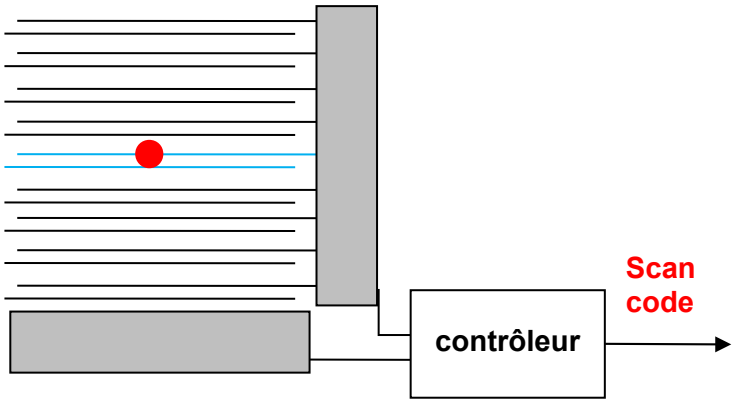
« Depuis, la configuration de la plupart des claviers informatiques est régie par la **norme ISO 9995** définie en 1985. Bien que le nombre de touches et leur disposition dépendent du pays ou de la langue utilisée, les claviers comportent toujours un peu plus de 100 touches (102, voire 105 avec l'adjonction des trois touches Windows). Les différents types de claviers se distinguent les uns des autres par des particularités nationales (caractères nationaux) et des particularités techniques (touches muettes). Il existe de nombreuses dispositions de touches différentes, qui sont principalement dues à la diversité des langues dans le monde. Chaque utilisateur peut ainsi travailler avec un clavier adapté à sa langue pour plus d'aisance et de rapidité. »

Fonctionnement du clavier

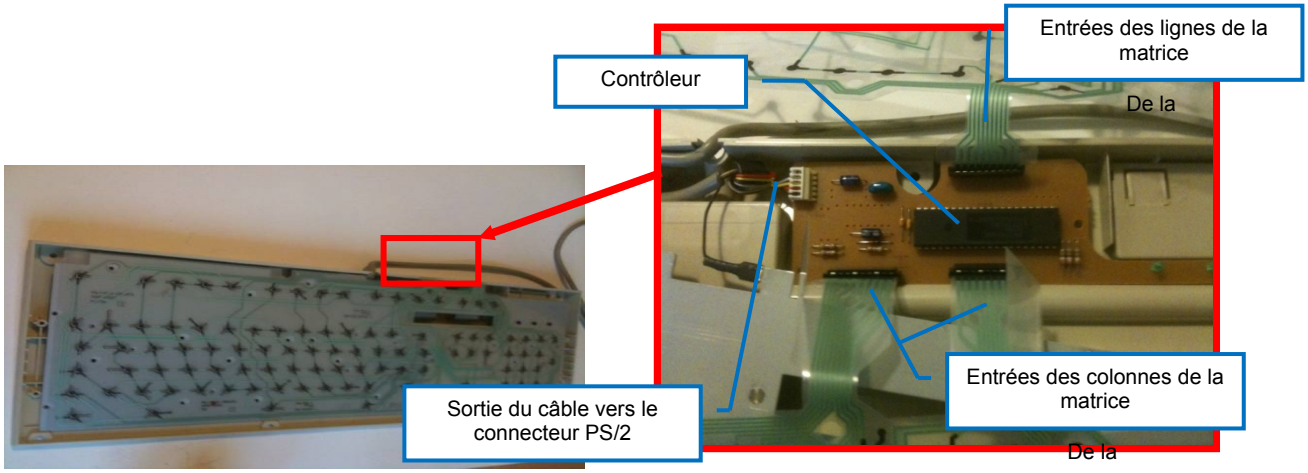
A chaque pression d'une touche du clavier, un signal spécifique est transmis à l'ordinateur. Le clavier utilise en effet un réseau matriciel permettant d'identifier chaque touche grâce à une ligne et une colonne. C'est uniquement la position de la touche sur le clavier qui est importante et non ce qui est écrit sur la touche.

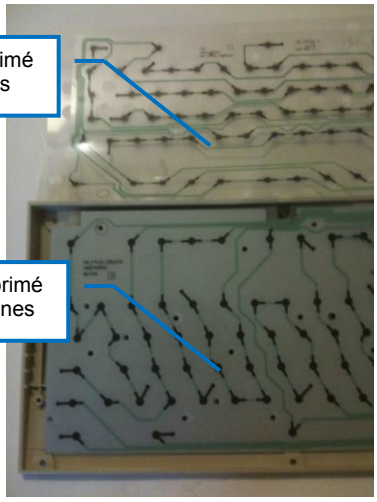
Lorsqu'une touche du clavier est pressée, un contact électrique s'établit entre la ligne et la colonne, comme le montre le schéma ci-dessous.

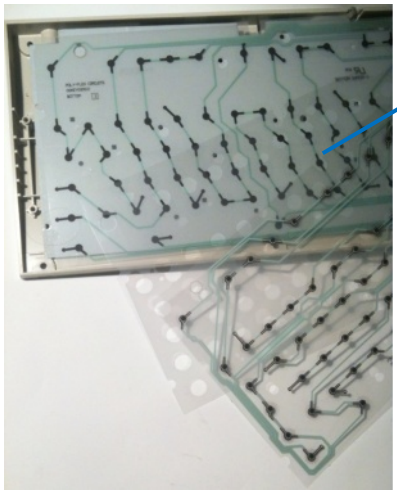
Le contrôleur qui se trouve dans le clavier surveille ainsi une matrice appelée matrice scan. Il fait correspondre à chaque touche un code appelée scan code qui est placé dans une mémoire tampon puis envoyé au clavier.



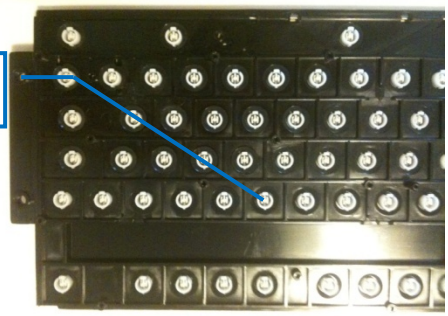
Des photos d'un clavier démonté, données ci-dessous, font bien apparaitre ce principe.

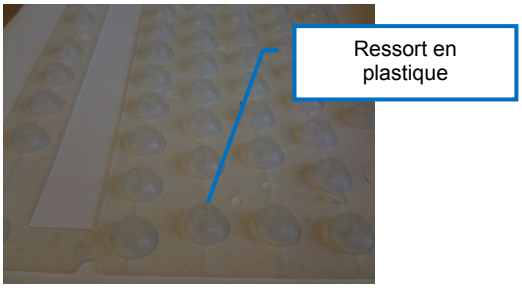












Transmission de données entre le clavier et l'ordinateur

1.1. Du connecteur DIN au Bluetooth

Les claviers des premiers PC utilisaient des connecteurs de type DIN (ce mode de branchement a complètement disparu).

En 1987, le connecteur PS/2, plus petit, le remplaça en conservant toutefois la compatibilité électrique avec le connecteur DIN.



En 1998, l'arrivée de l'USB tend à faire disparaitre le connecteur PS/2.



Depuis 2004, les technologies de transmission sans fil apparaissent. Elles utilisent un émetteur/récepteur qui est tributaire de l'un des modes de raccordement standard d'un clavier. Le sans-fil élimine donc la nécessité du fil, mais pas du connecteur (sauf dans certains cas pour les claviers Bluetooth si un récepteur de ce type est intégré à l'ordinateur).

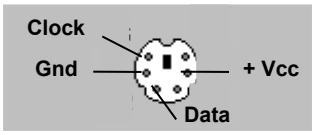


1.2. Transmission PS2

1.2.1. Nature des signaux

Il n'y a que 4 fils dans le câble reliant le clavier au connecteur PS/2. La nature des signaux est donnée sur le schéma ci-contre. Suivant les claviers, les couleurs des fils changent !

- **+Vcc** : alimentation du clavier (+ 5 V) ;
- **Gnd** : masse d'alimentation et des signaux ;
- **Data** : transmission bidirectionnelle synchrone des données ;
- **Clock** : horloge de synchronisation des données.

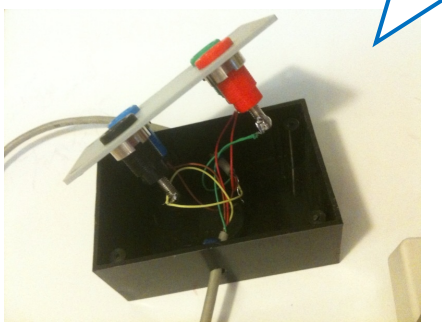


Prise mâle (vue en bout de cordon)

1.2.2. Principe des mesures

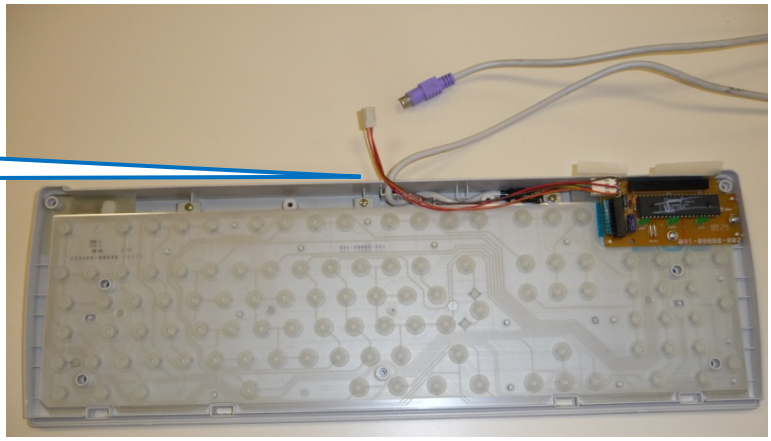
Plusieurs solutions peuvent être utilisées pour la visualisation des signaux :

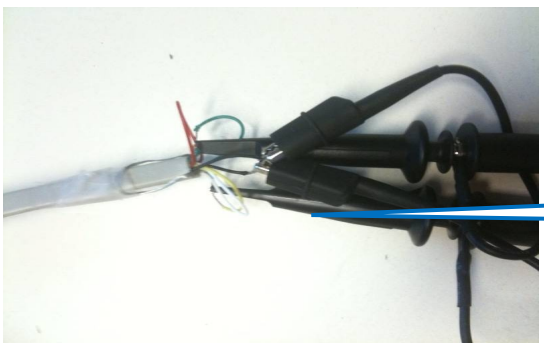
Réaliser un boitier avec 4 bornes en coupant les fils





Récupérer les 4 signaux dans le clavier en soudant 4 fils





Dénuder simplement les 4 fils ! Attention au court-circuit…

1.2.3. Protocole de communication

• **Type de protocole**

C'est un **protocole de type série synchrone bidirectionnel** : le clavier peut envoyer (les codes des touches) et recevoir des données (commandes). Dans tous les cas, la réception de données est prioritaire : si le clavier est en cours de transfert d'un code touche ou d'une réponse à une commande, et qu'au même moment il reçoit une commande, le clavier abandonne le contrôle des lignes Data et Clock. Après transmission de la commande par l'équipement et réponse du clavier, ce dernier tente de retransmettre la donnée dont la transmission a été précédemment interrompue.

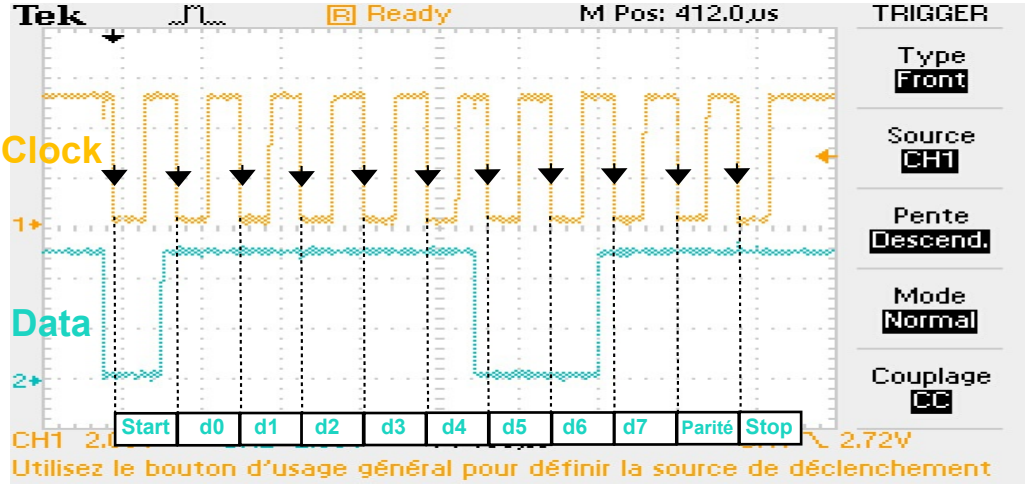
La transmission des données est une trame de 11 bits, le format des données est le suivant :

- 1 bit de start ;
- 8 bits de données (LSB en premier => d0 à d7) ;
- 1 bit de parité (parité : impaire) ;
- 1 bit de stop.

En réception comme en émission, c'est toujours le clavier qui génère l'horloge de synchronisation (Clock). La fréquence d'horloge moyenne varie entre 10 kHz et 30 kHz, suivant les modèles de clavier et les fabricants.
Chaque bit de données sera lu par le PC sur le front descendant de l'horloge.

• **Exemple de trame émise**

L'écran suivant a été relevé lorsqu'une touche a été appuyée :



On retrouve sur la première voie de l'oscilloscope CH1 l'horloge Clock, sur la seconde CH2 les données Data.
La ligne de données au repos est à 1. On retrouve ensuite la trame de 11 bits :

- le premier bit est le bit de Start à 0 ;
- on retrouve ensuite les 8 bits de données, la donnée envoyée est ici 1001 1111 soit 9F en hexadécimal ;

d7	d6	d5	d4	d3	d2	d1	d0
1	0	0	1	1	1	1	1

- il y a 6 bits de données à 1, soit un nombre pair, le bit de parité est à 1 pour qu'il y ait un nombre impair de 1 ;
- le dernier bit est le bit de Stop à 1.

Le « décodage » se fait ensuite en utilisant les tables de codes scan.

1.2.4. Codes scan

Il existe trois versions de code scan :

- le Set 1 utilisé par les ordinateurs compatible PC/XT et PS2-30 ;
- le Set 2 utilisé par les ordinateurs compatibles AT et PS/2 ;
- le Set 3 est prévu pour les stations de travail et l'émulation de terminaux sur PC.

Il est nécessaire d'identifier si une touche a été enfoncée ou relâchée ou tenue appuyée, le clavier enverra alors un scan code différent à l'ordinateur :

- les "Make Codes" (MC) quand une touche est appuyée ou maintenue appuyée ;
- les "Break Codes" (BC) quand une touche est relâchée.

Il n'y a pas de relation définie entre un Make Code et un code ASCII. C'est à l'ordinateur de transcrire les Scan codes en caractères.

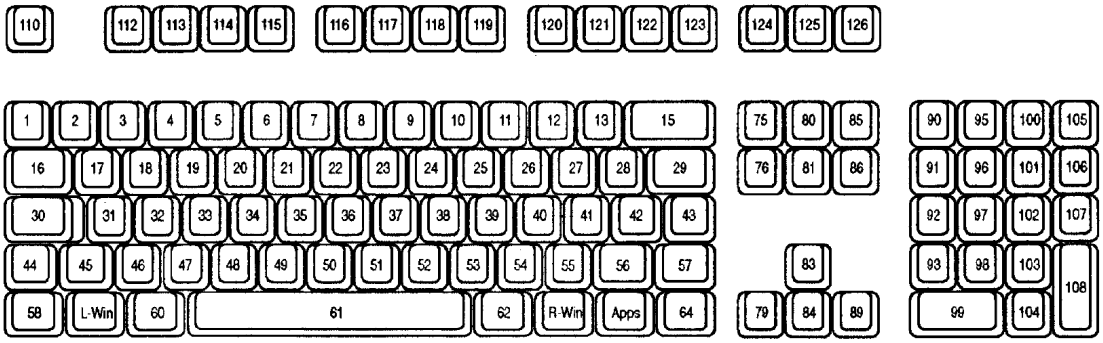
Extrait du tableau Scan Code (pour obtenir l'ensemble des codes se référer au document)

Microsoft

Windows Platform Design Notes

Designing Hardware for the Microsoft® Windows® Family of Operating Systems

Keyboard Scan Code Specification



key location	101/102 Enhanced Keyboard	scan 1 make	scan 1 break	scan 2 make	scan 2 brake
		E0_0D	E0_8D	E0_55	E0_F0 55
15	Backspace	0E	8E	66	F0 66
		E0_0E	E0_8E	E0_66	E0_F0 66
16	Tab	0F	8F	0D	F0 0D
		E0_0F	E0_8F	E0_0D	E0_F0 0D
17	Q	10	90	15	F0 15
		E0_10	E0_90	E0_15	E0_F0 15
18	W	11	91	1D	F0 1D
		E0_11	E0_91	E0_1D	E0_F0 1D
19	E	12	92	24	F0 24
		E0_12	E0_92	E0_24	E0_F0 24
43	Enter	1C	9C	5A	F0 5A
44	L SHIFT	2A	AA	12	F0 12
		E0_2A	E0_AA	E0_12	E0_F0 12
45**		56	D6	61	F0 61
		E0_56	E0_D6	E0_61	E0_F0 61

Note 4 Key 124 for Scan Codes 1 and 2:
This Key is Typematic **(Print Screen on US keyboards)**

Scan Code	Make	Break	LCtrl or RCtrl + LSHIFT or RSHIFT + Make	LCtrl or RCtrl + LSHIFT or RSHIFT + Break	LAlt or RAlt + Make	LAlt or RAlt + Break
1	E0 2A E0 37	E0 B7 E0 AA	E0 37	E0 B7	54	D4
2	E0 12 E0 7C	E0 F0 7C E0 F0 12	E0 7C	E0 F0 7C	84	F0 84

1.3. Transmission USB

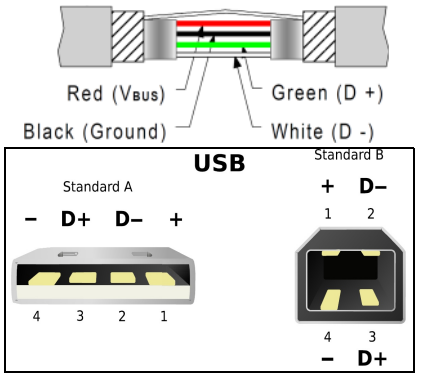
Cette présentation est limitée au strict minimum à connaitre sur les signaux USB « Low Speed » pour la mesure et l'analyse des informations échangées entre le PC et le clavier.

Les spécifications officielles sont détaillées dans *USB11.pdf*.

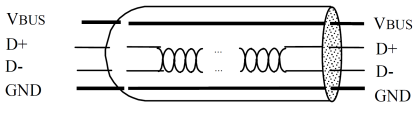
1.3.1. Nature des signaux

Il n'y a que 4 fils dans un câble USB. La nature des signaux est donnée ci-dessous.

Fonction	Couleur fils	Numéro de broche
Alimentation +5V (VBUS)	Rouge	1
Données (D-)	Blanc	2
Données (D+)	Vert	3
Masse (GND)	Noir	4



Les deux signaux de données D+ et D- sont complémentaires lors des transmissions d'informations, ils constituent la paire différentielle, pas nécessairement torsadée pour l'USB « Low Speed » du clavier.

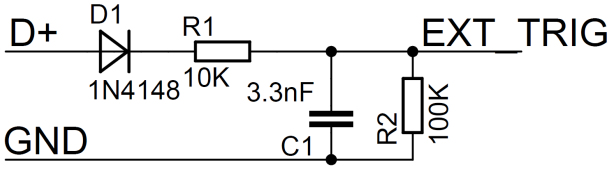


1.3.2. Principe des mesures

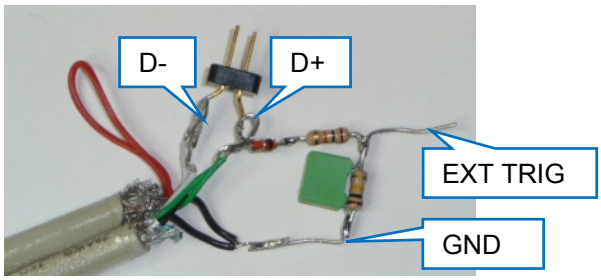
La solution la plus simple pour accéder aux signaux est de dénuder un câble prolongateur USB (Connecteur A femelle /connecteur A mâle) et de l'insérer entre le PC et le clavier. L'opération peut être réalisée directement sur le câble du clavier.

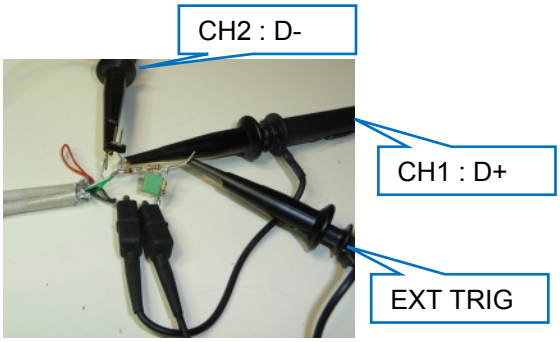
La transaction de données liées à l'appui ou relâchement d'une touche est noyée dans les trames cycliques d'interrogation du PC. Le PC transmet au clavier deux demandes « nouvelle information ? » espacées d'une milliseconde toutes les huit millisecondes.

Pour pouvoir synchroniser facilement l'oscilloscope sur les transactions avec transfert de données, il faut ajouter le montage ci-contre : les transactions avec données sont plus longues, C1 se charge à une tension de 2 volts contre 1 volt pour les transactions sans données.



Réalisation en fils volants : attention au court-circuit.





Pour un montage plus robuste, il est conseillé de réaliser ce montage dans un boitier du type présenté pour la mesure PS2. Les quatre bornes présenteront alors les signaux : GND, D+, D-, EXT TRIG.

1.3.3. Protocole et codage

- **Type de protocole**

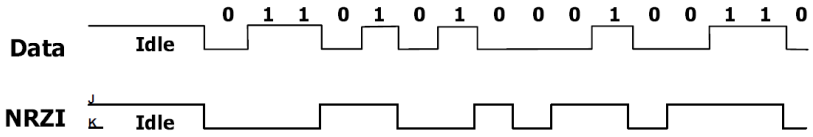
C'est un protocole de type série bidirectionnel half-duplex : le clavier peut envoyer et recevoir des informations. C'est la configuration « Low Speed » de la norme USB qui est utilisée pour le clavier soit un débit de 1,5 Mbit/s.

La trame de transmission appelée transaction est constituée de 2 ou 3 paquets :

- Token Packet : paquet jeton, en-tête indiquant l'opération demandée (PC → Clavier) ;
- Data Packet : paquet de données (optionnel) qui contient ici le scan code (PC ← Clavier) ;
- Handshake Packet : paquet de poignée de main pour valider l'opération demandée (PC → Clavier ou PC ← Clavier).

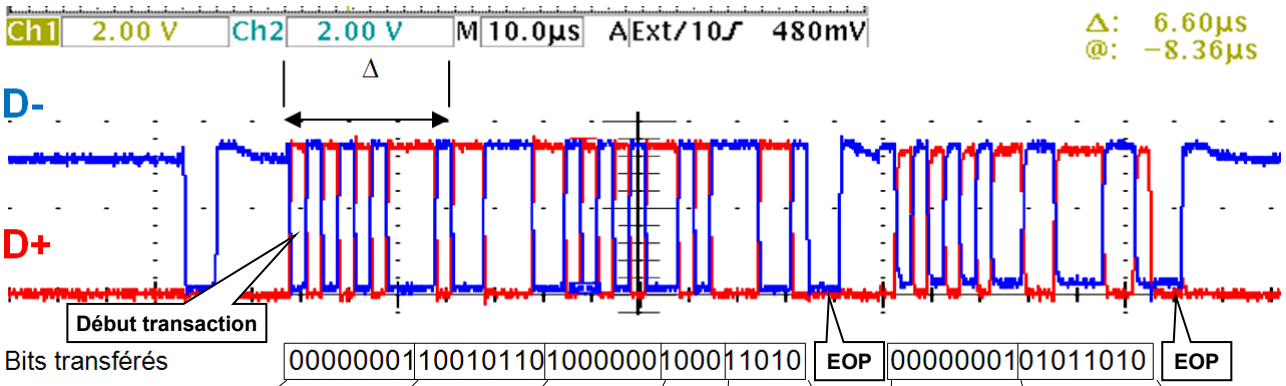
La couche physique, niveau 1 du modèle OSI, utilise un encodage NRZI sur une paire différentielle en 0/+3,3volts.

L'encodage NRZI (*Non Return to Zero Inverted) est une* variante du NRZ, son principe de codage est le suivant : changement d'état pour 0 et pas de changement pour 1.



- **Exemple de transaction**

La transaction suivante a été relevée lorsqu'aucune touche n'est appuyée : c'est la demande « nouvelle information ? » du PC au clavier avec une réponse négative du clavier.



Synchro	PID	Adresse	EndP	CRC5	Synchro	PID
8 bits	4+4 bits	7 bits	4 bits	5 bits	8 bits	4+4 bits

Token Packet	Handshake Packet

Au repos D+ = 0 et D- = 1

Le début de la transaction est annoncée par D+ = D- = 0 pendant environ 2 bits et devient effective au premier 0 codé par les transitions simultanées de D+ et D-.

- **Premier paquet transmis : Paquet Jeton**, c'est la demande du PC.

Ce paquet est constitué de :

- Synchro : 8 bits pour synchroniser l'horloge du récepteur : le clavier ;
- PID : Packet ID, pour identifier le type de paquet, 4 bits répétés à l'inverse pour le contrôle d'erreur. Le PID comme tous les mots transmis en USB est présenté avec le poids faible en tête ;

PID0	PID1	PID2	PID3	/PID0	/PID1	/PID2	/PID3
1	0	0	1	0	1	1	0

Tableau des 16 valeurs du PID et de leur signification :

Groupe	Valeur PID	Identificateur Paquet
Token	0001	OUT Token
	1001	IN Token
	0101	SOF Token
	1101	SETUP Token
Data	0011	DATA0
	1011	DATA1
	0111	DATA2
	1111	MDATA
Handshake	0010	ACK Handshake
	1010	NAK Handshake
	1110	STALL Handshake
	0110	NYET (No response Yet)
Special	0000	PREambule
	1100	ERR
	1000	Split
	0100	Ping

PID = 1001, IN Token, pour une demande de lecture du clavier ;

- Adresse = 000 0001, l'adresse du clavier est 1, seul périphérique connecté lors du relevé ;
- EndP = 0001, adresse interne au clavier demandée ;
- CRC5 = Contrôle d'erreur, sa valeur dépend des bits transmis, son calcul n'est pas abordé dans ce module ;
- EOP : End Of Packet. Fin du paquet représenté par D+ = D- = 0 pendant environ 2 bits.

- **Second paquet transmis : Paquet Poignée de main**, c'est la réponse du clavier

Ce paquet est constitué de :

- Synchro : 8 bits pour synchroniser l'horloge ;
- PID : Packet ID, pour identifier le type de paquet, 4 bits répétés à l'inverse pour le contrôle d'erreur ;

PID0	PID1	PID2	PID3	/PID0	/PID1	/PID2	/PID3
0	1	0	1	1	0	1	0

- PID = 1010, NAK : le clavier indique qu'il n'a pas de nouvelle information à transmettre ;
- EOP : End Of Packet. Fin du paquet représenté par D+ = D- = 0 pendant environ 2 bits.

- **Transaction suite à l'appui d'une touche**

Lors de l'appui sur une touche, la trame échangée entre le PC et le clavier est augmentée d'un paquet de données de 8 octets de données :

Token Packet	Data Packet	Handshake Packet

Avec :

- Token Packet : le même que précédemment ;
- Data Packet : Synchro, PID, 8 octets de données, CRC16 ;
- Handshake Packet : Synchro, PID

1.3.4. Code scan

Lors de l'appui sur une touche, sur les 8 octets de données transmis par le clavier, seul un octet est utile, les autres sont à 00, sa valeur est définie par le code scan USB présenté en « Appendix C » du document Scan Code Microsoft, page 28.

Appendix C: USB Keyboard/Keypad Page (0x07)

This section is the **Usage Page** for key codes to be used in implementing a USB keyboard. A **Boot Keyboard** (84-, 101- or 104-key) should at a minimum support all associated usage codes as indicated in the "Boot" column below.

Note: Due to the variation of keyboards from language to language, it is not feasible to specify exact key mappings for every language. Where this list is not specific for a key function in a language, the closest equivalent key position should be used, so that a keyboard may be modified for a different language by simply printing different keycaps. One example is the Y key on a North American keyboard. In Germany, this is typically Z. Rather than changing the keyboard firmware to put the Z Usage into that place in the descriptor list, the vendor should use the Y Usage on both the North American and German keyboards. This continues to be the existing practice in the industry, in order to minimize the number of changes to the electronics to accommodate other languages.

Usage index (dec)	Usage Index (hex)	Usage	Ref:typical AT-101 position	PC-AT	Mac-intosh	UNIX	Boot
0	00	Reserved (no event indicated)[9]	N/A	√	√	√	84/101/104
1	01	Keyboard ErrorRollOver[9]	N/A	√	√	√	84/101/104
2	02	Keyboard POSTFail[9]	N/A	√	√	√	84/101/104
3	03	Keyboard ErrorUndefined[9]	N/A	√	√	√	84/101/104
4	04	Keyboard a and A[4]	31	√	√	√	84/101/104
5	05	Keyboard b and B	50	√	√	√	84/101/104
6	06	Keyboard c and C[4]	48	√	√	√	84/101/104
7	07	Keyboard d and D	33	√	√	√	84/101/104
8	08	Keyboard e and E	19	√	√	√	84/101/104
9	09	Keyboard f and F	34	√	√	√	84/101/104
10	0A	Keyboard g and G	35	√	√	√	84/101/104
11	0B	Keyboard h and H	36	√	√	√	84/101/104
12	0C	Keyboard i and I	24	√	√	√	84/101/104
13	0D	Keyboard j and J	37	√	√	√	84/101/104
14	0E	Keyboard k and K	38	√	√	√	84/101/104
15	0F	Keyboard l and L	39	√	√	√	84/101/104
16	10	Keyboard m and M[4]	52	√	√	√	84/101/104
17	11	Keyboard n and N	51	√	√	√	84/101/104
18	12	Keyboard o and O[4]	25	√	√	√	84/101/104
19	13	Keyboard p and P[4]	26	√	√	√	84/101/104
20	14	Keyboard q and Q[4]	17	√	√	√	84/101/104
21	15	Keyboard r and R	20	√	√	√	84/101/104

…