Ordinateurs, Structure et Applications

GIF-1001

Cours 17, Le port série

Etienne Tremblay

Université Laval, Hiver 2012

Définition et historique

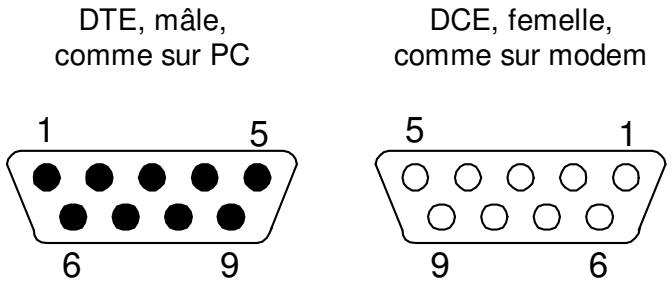
- Le port série est un très vieux port utilisé dans les tout premiers ordinateurs. Le protocole RS-232, définissant le format des données échangées sur ce port, date de 1962! Différentes versions du protocole ont été crées depuis. Principalement, on retrouvera RS-232C, créé en 1969 et RS-232D, en 1986.
- Au début, le port série était utilisé pour plusieurs périphériques du PC. De nos jours, il sert essentiellement à la communication avec des instruments de laboratoire ou appareils dédiés à des tâches spécifiques comme des lecteurs de code à bar, des caisses enregistreuses, etc. Le USB et le FireWire, beaucoup plus récents, ont remplacé progressivement le port série dans la plupart des applications.

Caractéristiques principales

- Le port série est un port point à point. Il relie deux appareils entre eux, branchés à chaque extrémité du fil. La communication entre les deux appareils est bidirectionnelle.
- Du fait de ses caractéristiques matérielles, le port série peut être utilisé sur de grandes distances. Les spécifications de base établissent la distance maximale à 50 pieds (environ 15 mètres), mais il est possible d'augmenter considérablement cette distance avec un fil de bonne qualité.
- Initialement, le port série avait des connecteurs 25 broches, mais un connecteur plus petit, 9 broches, est rapidement apparu dans le standard. Dans les faits, il est possible de communiquer par port série avec seulement 3 fils!
- Les signaux sur le port série vont de +15V à -15V avec des maximum à +-25V.
- La vitesse maximum du port série, selon la norme RS-232 est 20kbps (19200bps, plus exactement). Dans les faits, les utilisateurs du port poussent cette vitesse jusqu'à 115kbps (115200bps).

Matériel et Connecteur

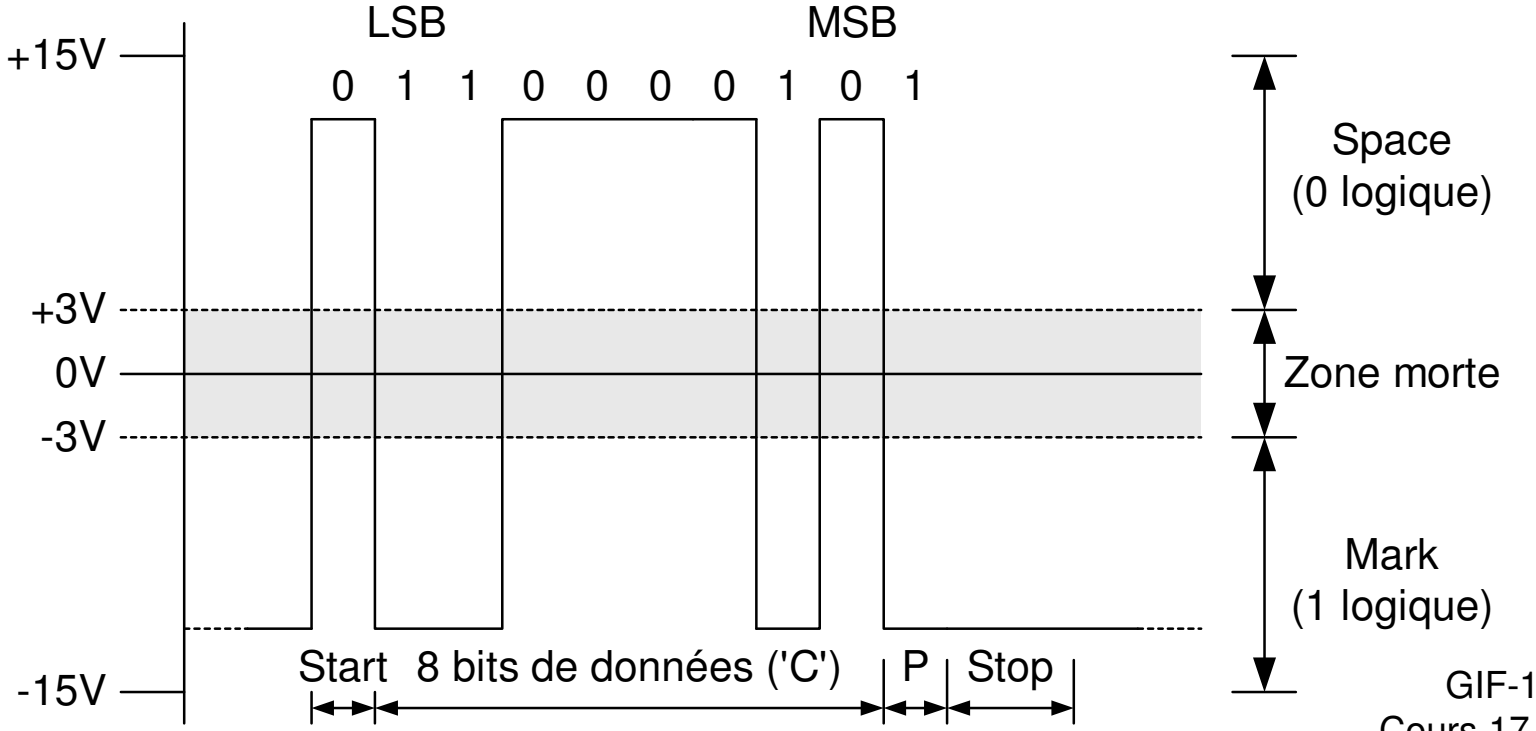
- Dans le protocole RS-232, il existe deux types d'appareils: les Data Terminal Equipment (DTE) qui sont équivalent aux PCs et les Data Communication Equipment (DCE) qui communiquent des données au PC.
- Deux connecteurs sont utilisés dans la norme RS-232: le DE-9 (appelé DB-9 par abus de langage) et le DB-25. Ces connecteurs ont respectivement 9 et 25 broches. La table ci-dessous montre les 9 broches du connecteur DE-9 avec les broches correspondantes pour le DB-25.
- Trois lignes, en vert, servent pour communiquer: RD, TD et la masse du signal. Les lignes RD et TD contiennent les signaux transmis du DTE au DCE et ceux du DCE au DTE, respectivement. Il est possible de faire de la communication par le port série avec ces trois lignes seulement!
- Les autres lignes servent au contrôle de flux de données entre le DTE et le DCE. Elles indiquent que le DTE ou le DCE est prêt à recevoir ou à émettre des données. Les lignes en bleu (DTR, DSR, RTS et CTS) sont couramment utilisées.



Nom	9-pin DTE	25-pin DTE	Contrôle
Carrier Detect (DCD)	1	8	DCE
Received Data (RD)	2	3	DCE
Transmitted Data (TD)	3	2	DTE
Data Terminal Ready (DTR)	4	20	DTE
Signal Ground	5	7	DCE
Data Set Ready (DSR)	6	6	DCE
Request To Send (RTS)	7	4	DTE
Clear To Send (CTS)	8	5	DCE
Ring Indicator (RI)	9	22	DCE

Signaux

- Le signal transmis sur les pins RD et TD va de +15V à -15V. S'il est entre +3V et +15V, il est interprété comme un 0 logique. S'il est entre -3V et 15V, il est interprété comme un 1 logique. Entre -3V et 3V, un signal est considéré invalide.
- La fréquence du signal est préétablie par l'opérateur du DTE ou du DCE. Elle peut aller de 300bps à 115kbps.
- Des bits de départs et de fins servent à délimiter les bits de données.
- Il peut y avoir un bit de parité servant à détecter les erreurs. Ce bit est décrit plus loin.



Électronique associée au port série

- Il existe une multitude de composantes électroniques reliées au port série. Certaines composantes permettent d'adapter un câble 9 fils à un câble 25 fils, d'adapter un port série à un autre type de port, d'alimenter un appareil à partir du port série, etc.
- Un câble NULL modem est un câble qui permet de connecter deux PCs entre eux. Comme les PCs sont tous deux des DTE, ils transmettent tous deux sur la pin 3 de leur connecteur DB9 et ils reçoivent tous deux sur la pin 2. Un câble NULL modem est essentiellement un câble dans lequel les pins 2 et 3 sont inversées. Ainsi, chacun des PCs transmet sur la ligne de réception de l'autre PC. D'autres pins du câble sont inversées (DTR-DSR, RTS-CTS) afin de ne pas avoir deux DTE imposant des tensions différentes sur la même ligne.
- Le port série n'est pas isolé électriquement. Autrement dit, la référence du signal (Signal Ground) est la référence de votre PC. Un appareil relié au port série d'un PC doit avoir la même référence. Dans certains cas, un appareil alimenté en 220Vac par exemple, cette référence n'est pas les mêmes et connecter deux appareils au même port série peut créer un court circuit (et, potentiellement, griller un PC). Pour cette raison, des circuits d'isolation à base d'optocoupleurs sont fréquents sur le marché.

Protocole de communication

- Le principal protocole de communication utilisé sur le port série est le RS-232. Cette spécification détermine:
  - Les caractéristiques des signaux électriques transmis (voltages, vitesse, transitions, longueurs de fils, etc.).
  - Le connecteur utilisé.
  - Les fonctions de chaque partie du port.
  - Quelques applications typiques.

Paramètres du port série

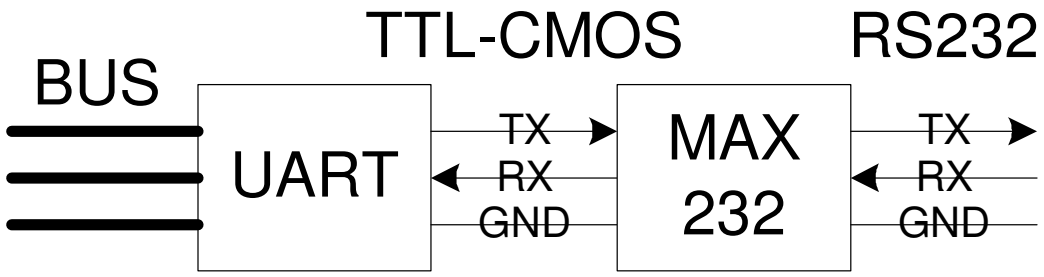
- Lorsqu'on utilise un port série, il faut toujours déterminer les paramètres de communications sur le port. Ces paramètres (Settings) sont décrits ci-dessous:
  - *Baud Rate*: Il s'agit de la fréquence des bits transmis sur le port série. Les fréquences disponibles sont pré-établies: 300bps, 600bos, 1200bps, …19200bps, 38400bps, etc. Défaut = 9600bps
  - *Parité*: Le bit de parité sert à vérifier s'il y a eu des erreurs dans le byte transmis. Le nombre de 1 dans le byte transmis est comptabilisé et le bit de parité est ajusté en fonction de ce dernier. En réception, on compte le nombre de 1, puis on vérifie si le bit de parité est bon. Il y 3 valeurs possibles à ce paramètre: paire, impaire et pas de parité. Défaut: pas de parité.
  - *Nombre de bits de stop*: Nombre de bit de stop (1) qui suivent le byte transmis. Défaut = 1.
  - *Nombre de bits par byte*: Nombre de bit transmis par byte. Peut être 5,6,7, 8 et 9 (très rare!). Défaut = 8.

Le port série et le x8086

- Le port série est plus ancien que le 8086 et il était possible de l'utiliser avec les premières versions de ce microprocesseur.
- La première façon d'accéder au port série, avec un 8086, consiste à écrire directement aux ports reliés au port série. Ces ports permettent de contrôler tous les paramètres de la communication. COM1 est habituellement situé aux ports 3f8h à 3ffh alors que COM2 est habituellement situé aux ports 2f8h à 2ffh. C'est le BIOS, lors de la mise sous tension de l'ordinateur (lors du POST), qui alloue des ports aux interfaces détectées.
- Il y a moyen de combiner l'utilisation de ces ports avec l'utilisation des interruptions du port série (interruptions matérielles IRQ3 et IRQ4).
- La deuxième façon d'accéder au port série, avec un 8086, consiste à utiliser l'interruption 14h du BIOS. L'interruption 14h permet de le configurer, d'émettre ou de recevoir des bytes et de tester l'état de la communication. Notez toutefois que cette méthode est dépassée et qu'il est préférable de communiquer directement avec le port série.

UART et RS232

- UART signifie Universal Asynchronous Receiver Transmetter
- Un UART est un module d'E/S qui convertit les signaux parallèles d'un bus en signaux en série.
- Le signal série sortant d'un UART est comme le signal RS232 (Start bit, Octet de données, Parité, Stop Bit), mais avec des niveaux de tension TTL ou CMOS.
- Le UART utilise un registre à décalage pour convertir les signaux parallèles en signaux série tel qu'illustré en annexe A.



Programmation avec un PC

- Pour réaliser un programme utilisant un port série, il faut d'abord avoir une librairie de fonctions permettant d'accéder au port. Bien qu'il soit possible de créer soi-même les fonctions pour accéder au port série (voir l'acétate précédente), de telles fonctions sont disponibles dans presque tous les langages de programmation.
- Une librairie de fonctions permettant de communiquer par RS-232 comprends habituellement les fonctions suivantes:
  - Ouverture du port: Cette fonction vérifie que le port série existe et qu'il est disponible. Ensuite, elle réserve cette ressource pour l'application. Finalement, cette fonction détermine les paramètres de configuration du port (baud rate, parité, stop bits et nombre de bits). Lors de l'ouverture du port série, la fonction réserve également de la mémoire tampon pour les bytes à transmettre et ceux reçus.
  - Écriture de bytes: Envoyer des bytes dans un tampon de transmission. Le tampon est vidé progressivement lorsque les bytes sont transmis.
  - Réception de bytes: Lire dans un tampon si des bytes sont reçus. Habituellement, les bytes reçus sont mis dans un tampon et les fonctions permettent d'accéder au tampon. Ce dernier est vider lors de la lecture des bytes.
  - Fermeture du port: Libération du port pour qu'il puisse être utilisé par d'autres applications.

Conception d'un périphérique RS-232

- Un des problèmes qui survient lors de l'utilisation du port série est les niveaux de tensions du signal. Les microprocesseurs sont habituellement alimentés en 5Vdc, en 3.3Vdc ou en 2.5Vdc. Or, le signal RS-232 avoisine habituellement +12Vdc et -12Vdc. Il est impossible d'y connecter directement une sortie de microprocesseur. Heureusement, il existe une multitude de circuits intégrés permettant de communiquer par RS-232 et solutionnant ce problème. Le plus connu de ces circuits est le MAX232. Tout périphérique communiquant par RS-232 devrait inclure un tel circuit intégré ou l'équivalent. Vous retrouverez facilement la datasheet du MAX232 à l'adresse http://www.maxim-ic.com/ ou en cherchant très peu sur internet.
- Un autre problème est la gestion des bytes en série. Il faut implémenter un registre à décalage pour recevoir tous les bits reçus un à un ou, encore, posséder un microprocesseur qui gère cet aspect de la communication (très commun!).
- Il est possible d'alimenter un périphérique directement à partir du câble série. Toutefois, cette méthode est peu commune et le périphérique doit consommer très peu de courant. Habituellement, un périphérique sur le port série a sa propre alimentation.
- Avant de brancher votre appareil sur le port série, assurez-vous de ne pas griller votre PC en reliant sa masse à celle de l'appareil branché sur le port.

RS-422 et RS-485

- Le RS-422 est une extension du RS-232. Afin de communiquer sur de plus grandes distances, la couche physique du RS-232 a été transformée, créant le RS-422. Le RS-422 est un protocole de communication série, point à point (ou avec 1 seul transmetteur et plusieurs receveurs), basé sur le RS-232. Les bits en RS-422 sont transmis en mode différentiel: la différence de tension entre deux lignes détermine s'il s'agit d'un « 1 » ou d'un « 0 ». Ce mode de transmission élimine le bruit en mode commun (Signal ligne A + Bruit sur ligne A – Signal ligne B + Bruit sur ligne B ~ Signal ligne A - Signal ligne B = Bit transmis) et le signal peut être transmis sur de plus longues distances. Le protocole RS-422 supporte la transmission half-duplex ou full-duplex.
- Le RS-485 est une extension du RS-422. Des buffers tri-state en transmission et une résistance aux collisions sur la ligne permettent la communication multipoint en mode différentiel.
- Plusieurs circuits intégrés et appareils électroniques permettent la conversion de RS-232 à RS-422 ou RS-485 et vice versa.

Références et exercices

- Références
  - http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html
  - http://www.arcelect.com/rs232.htm
  - http://www.aurel32.net/elec/port_serie.php

Annexe A, Exemple de UART

- Pour plus d'explications sur la figure ci-dessous, demandez à l'enseignant!