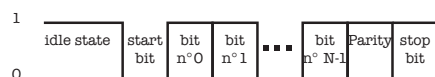


Liaison série asynchrone

Définie à l'origine pour les échanges via un modem, la liaison série asynchrone est très utilisée dans les microcontrôleurs pour communiquer avec les PC via leur port série (COM) suivant la norme RS232. La plupart des microcontrôleurs sont pourvus d'une ou plusieurs UART (Universal Asynchronous Receiver Transmitter) qui rendent très simple la mise en oeuvre de ce protocole. La liaison série permet de transmettre et recevoir des données en série sans transmettre d'horloge. Deux fils sont utilisés pour émettre et recevoir simultanément, l'un pour l'émission (Tx) et l'autre pour la réception (Rx). Un troisième fil est utilisé pour la masse. Les autres signaux, principalement destinés à la communication avec un modem, sont souvent ignorés ou absents.

Trame

Au repos, la ligne de transmission est à l'état logique 1. Un front descendant signale qu'une donnée va être émise. Pour cela, le premier bit transmis, appelé start-bit, est un niveau logique 0 et la transmission se termine par un niveau logique 1 (stop-bit) de durée configurable. La transmission des données elles-mêmes s'effectuent en série, c'est-à-dire bit après bit en commençant par le poids faible. Le nombre de bits de données transmis est aussi configurable. Enfin, un bit de parité peut être ajouté. Dans ce cas, l'état de ce bit est calculé par l'UART pour que le nombre total

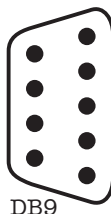


de bits transmis (données+parité) soit pair (parité paire) ou impair (parité impaire). Si le récepteur est également configuré pour vérifier cette parité, il est ainsi possible de détecter une erreur.

BaudRate

Comme l'horloge n'est pas transmise, émetteur et récepteur doivent être configurés pour utiliser la même vitesse de transmission. On parle du BaudRate qui s'exprime en Baud, inverse du temps d'un bit. En toute rigueur, un Baud est une vitesse de transmission

- 6 : Data Set Ready (DSR)
- 7 : Request To Send (RTS)
- 8 : Clear To Send (CTS)
- 9 : Ring Indicator (RI)



- 1 : Data Carrier Detect (CD)
- 2 : Receive Data (RD or RX or RXD)
- 3 : Transmitted Data (TD or TX or TXD)
- 4 : Data Terminal Ready (DTR)
- 5 : Signal Ground (GND)

de symboles par seconde et non de bit, mais, dans une liaison série les symboles étant binaires, les deux définitions sont équivalentes. Une transmission à 300 Baud signifie que la durée d'un bit sera de 1/300 s soit 3,33 ms. Les vitesses de transmission les plus utilisées sont 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 et 115200 Baud. Sur les microcontrôleurs, ces vitesses sont obtenues par division de la vitesse d'horloge. Les diviseurs de fréquence utilisés étant des compteurs, le facteur de division est un nombre entier, ce qui peut conduire à un écart entre la vitesse voulue et la vitesse réelle. Lorsque cet écart dépasse 5%, l'erreur accumulée sur l'ensemble de la trame peut induire un décalage qui peut amener le récepteur à échantillonner la donnée au mauvais moment, provoquant une défaillance de la communication.

Niveaux logiques

Pour communiquer à relativement longue distance — et pour communiquer avec un PC — on utilise un transceiver qui convertit les niveaux TTL en niveaux RS232, typiquement un circuit MAX232 ou MAX 233. D'après cette norme, le 1 logique, appelé MARK est représentée par une tension typique de -12 V alors que le zéro logique (appelé SPACE) est lui représenté par une tension typique de +12 V. Une donnée émise est alors doublement retournée visuellement lorsqu'on l'observe à l'oscilloscope après un transceiver RS232. D'une part, les bits de poids forts sont à droite, et d'autre part le niveau haut est en bas. Ceci est illustré par la figure en bas de cette fiche qui représente l'émission des trois caractères «IUT» avec une trame de 8 bits de données, un bit de parité paire et un bit de stop.

Mise en oeuvre

Le HC12 que nous utilisons sur la carte STAR12 (MC9S12DP256B) comprend deux UART nommées SCIO et SCI1 par Motorola (Serial Communication Interface). Sur la carte Star12, SCIO est connectée via un transceiver MAX232 à un connecteur DB9. Le câblage de celui-ci est prévu pour communiquer avec un PC, les lignes Tx et Rx sont donc croisées sur la carte. L'UART SCI1 est reliée directement (sans transceiver) sur un connecteur HE10 10 broches.

Pour configurer l'UART, il faut, sur le HC12, configurer le BaudRate et autoriser émetteur et récepteur. Les registres 8 bits SCiBDH et SCiBDL représentent les 8 bits de poids forts et les 8 bits de poids faibles du diviseur qui génère le BaudRate de l'UART SCi, où i vaut 0 ou 1. La fréquence du bus du microcontrôleur (généralement moitié de celle de l'oscillateur externe, sauf si la PLL interne est activée) est divisée par 256 x SCiBDH + SCiBDL puis à nouveau divisée par 16 pour obtenir le BaudRate. Les deux bits TE_i et RE_i autorisent l'émission et la réception de l'UART SCi. Par défaut, c'est-à-dire sans programmer d'autres registres, la communication se fera avec 8 bits de données, sans parité et avec un bit de stop. C'est justement la configuration la plus couramment utilisée.

Avant d'émettre un caractère, il faut attendre que le buffer d'émission soit vide, cet événement est signalé par le passage à 1 du bit TDRE_i. L'émission à proprement parlé se fait en écrivant dans le registre SCiDRL. En réception, il faut attendre qu'une donnée reçue soit arrivée, événement signalé par le bit RDRF_i. La donnée reçue peut alors être lue dans le registre de réception qui occupe la même adresse que celui d'émission : SCiDRL.

Interruptions

Les deux événements, buffer d'émission vide et buffer de réception plein, peuvent déclencher une interruption si elle est autorisée. Au passage, il est bon de savoir que le MC9S12DP256 souffre d'une erreur de conception : si les deux événements surviennent simultanément, aucune interruption ne se déclenche (il y a dans le circuit une porte OU EXCLUSIF à la place d'une porte OU). Mieux vaut être prévenu.

