

Proposition de correction du TP 4

Informatique Embarquée

Ce document propose des réponses pour certaines questions du TP. Les réponses aux autres questions sont fournies dans des fichiers à part.

Partie 1

Question 1

La suite de bits à analyser est la suivante :

```
1110010000101011110110100111011010010101101011110110101010111010010011101111
```

Il s'agit de l'état logique d'une ligne de transmission au cours du temps. Chaque bit représente l'état logique forcé sur la ligne durant un cycle d'horloge.

D'après l'énoncé, les paramètres suivant sont utilisés :

- 8 bits de données, en commençant par le bit de poids faible
- pas de bit de parité
- 1 bit de stop

Lorsqu'aucune données n'est transmise, la ligne est maintenue à l'état logique haut (1). Le bit de **start** est positionné à l'état bas (0). Les 8 bits suivant sont les bits de données, en commençant par le bit de poids faible. Le bit de **stop** est positionné à l'état haut (1). La trame suivante peut débuter, ou la ligne peut rester à l'état logique haut en attendant une nouvelle transmission.

En notant - les bits **idle** (sans transmission), B les bits de start, E les bits de stop, et [0-7] les bits de données (où 0 représente le bit de poids faible et 7 le bit de poids fort), on obtient donc la représentation suivante :

```
1110010000101011110110100111011010010101101011110110101010111010010011101111
---B01234567EB01234567EB01234567EB01234567EB01234567EB01234567E---
```

On a donc transmis les données :

01000010	B
01101111	o
01101111	n
01101011	j
01101111	o
01110101	u
01110011	r

Question 2

Voir le fichier `decode.py`, contenant l'automate et une proposition d'implémentation en Python pour la question 3.

Partie 2

Question 1

D'après la documentation de l'ATmega328P, la broche PD1 porte la fonction Tx et la broche PD0 porte la fonction Rx (tableau 14-9). Il est également possible de repérer ces fonctions sur le schéma 1-1.

Question 2

Les broches PD0 (Rx) et PD1 (Tx) sont respectivement connectées aux broches PD3 (Tx) et PD2 (Rx) du composant ATmega8U2-MU. Ce composant est l'interface entre un ordinateur (branché via USB) et l'ATmega328P que nous programmons. Ainsi, les données lues par le périphérique UART de l'ATmega328P (sur RX) peuvent avoir été envoyées via la liaison USB, et inversement les données écrites via le périphérique UART de l'ATmega328P (sur TX) sont transmises à travers la liaison USB.