



par Bob Smith
<bob/at/linuxtoys.org>

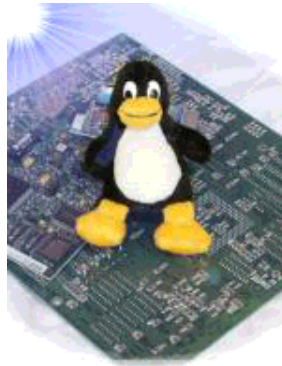
L'auteur:

Bob est un amateur en électronique et un programmeur sur Linux. Vous pouvez trouver son dernier projet sur www.linuxappliance-design.com et sa page d'accueil sur www.linuxtoys.org.

Traduit en Français par:

Iznogood
<icznogood/at/icznogood-factory.org>

Analyseur de donnée à 1 bit



Résumé:

L'analyseur de données à 1 bit capture des données de 1 bit à une vitesse de 46080 (ou 92160) échantillons par seconde et code les données vers un flux de caractères RS232 qui peut être lu à 57.6 (ou 115.2) kilobaud. L'analyseur de données est un projet assez simple qui utilise 6 circuits bon marché.

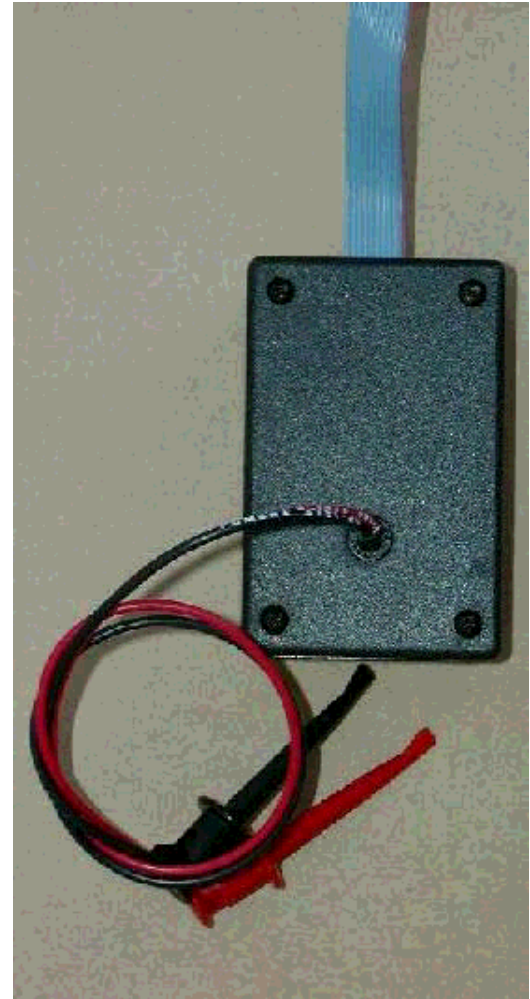
Introduction

Cet article décrit la conception, la construction et l'utilisation de l'analyseur. Un programme pour la capture et l'affichage du flux de données est fourni.

Conception et théorie du mode opératoire

Objectifs de la conception

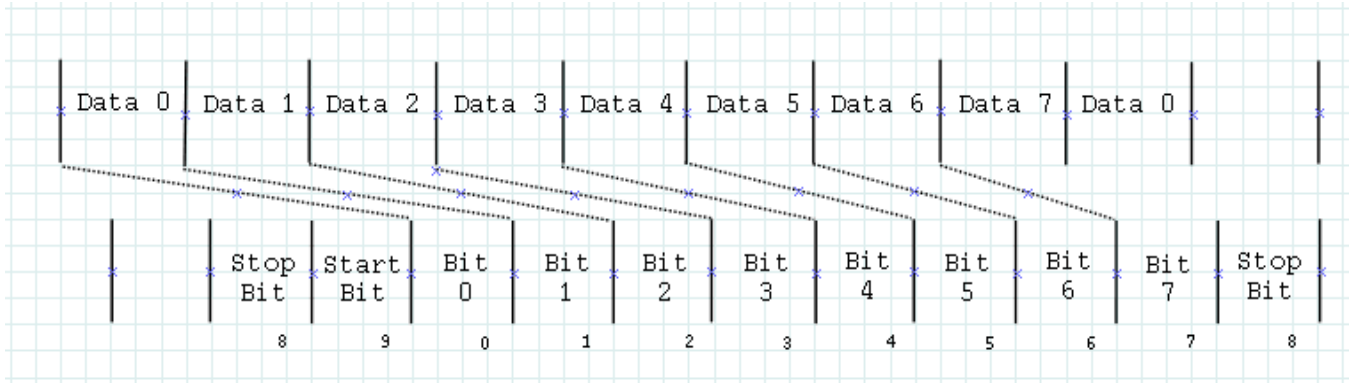
- un taux d'échantillonnage permettant de faire plus de 8000 échantillons par secondes
- une sortie des données sur un port série
- un échantillonnage continu sans qu'il manque un seul bit
- qu'il parasite son alimentation du port série



Survol de la conception

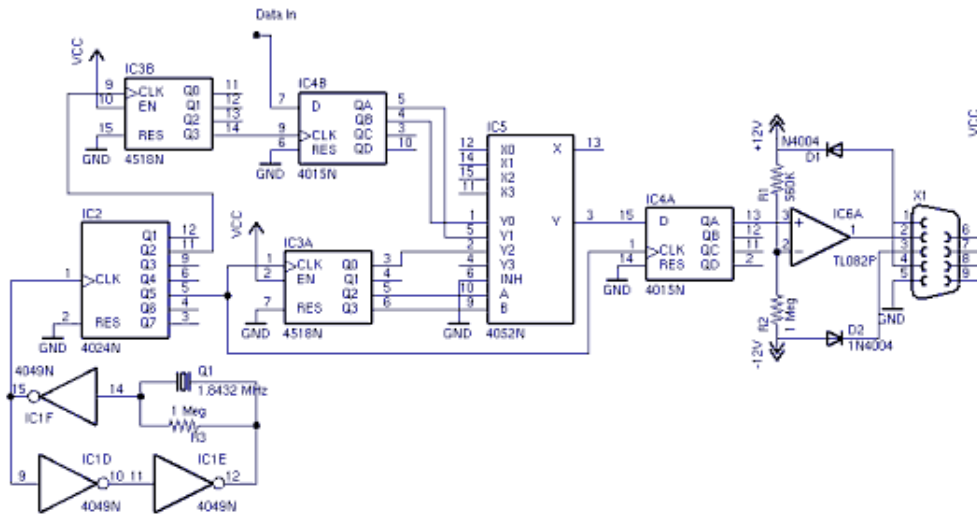
L'objectif est d'échantillonner une entrée à une fréquence stable et de sortir les données structurée en série avec un bit de démarrage, 8 bits de données et un bit de stop. La surcharge due aux bits de démarrage et de stop impliquent que le taux d'échantillonnage doit être de huit dixième le taux du port série. Pour un port série de 57.6 kbaud, le taux de donnée est de $(8 * 5.76 \text{ k})$ ou 46080 échantillons par seconde.

Le problème est que nous devons stocker les échantillons d'entrée lorsque nous devons envoyer les bits de démarrage et de stop. Le diagramme suivant illustre le problème.



Pendant les bits 0, 1, 2 et 3, nous souhaitons que les données soient décalées de deux coups d'horloge, nous sélectionnons donc Q(B) du registre à décalage 4015N. Pendant les bits 4, 5, 6 et 7, nous voulons que les données soient retardées d'un coup d'horloge, nous sélectionnons donc Q(A) du registre à décalage. Pendant les bits de démarrage et de stop (bits 8 et 9), nous voulons un zéro et un un, nous sélectionnons donc Q(0) du compteur de bit.

Shéma de l'analyseur de données 1 bit



La capture du shéma a été faite avec un programme très bien appelé « EAGLE » disponible sur Cadsoft. Le fichier de shéma est disponible sur 1bitla.sch.gz.

Le circuit horloge

Le circuit horloge possède deux sorties : une à la vitesse de transfert et une autre aux huit dixièmes de la vitesse de transmission. Le circuit d'horloge utilise trois inverseurs du 4049, le compteur binaire 7 étages du 4024 et un compteur de décade pour moitié du 4518. L'oscillateur fonctionne à 1.8432 MHz et Q(2) du 4024 sort des créneaux de 460.8 kHz. La sortie de Q(3) du diviseur par dix, 4518, est 46.08 kHz. La sortie Q(5) du compteur à retenue 7 étages a une sortie de créneaux à 57.6 kHz.

Pour faire tourner le circuit à 115.2 kHz, nous utilisons les sorties Q(1) et Q(4) du 4024 au lieu des Q(2) et Q(5). D'autres vitesses sont possibles en utilisant des valeurs différentes pour le quartz ou en le remplaçant par un oscillateur RC.

Il est possible qu'un quartz consomme moins qu'un réseau RC à la même fréquence. Pour être sûr de limiter la consommation, les entrées inutilisées du 4049 doivent être mises à la masse.

Les buffers d'entrée

Il existe deux manières faciles d'ajouter des entrées de buffers au circuit ci-dessus. Vous pouvez connecter l'entrée des données sur un des inverseur inutilisé du 4049. Cela ne pose pas de problème car l'entrée ne représente qu'une charge CMOS et garde tout de même un niveau de déclenchement (qui représente 3 volts dans ce circuit).

Un meilleur buffer d'entrée peut être utilisé en utilisant le second TL082. Vous pouvez utiliser un pont de résistance sur une des entrées de l'amplificateur opérationnel et l'autre pour l'entrée des données. Ceci vous permet de contrôler le seuil de déclenchement d'une manière précise et présente une très haute impédance pour le système en cours de test.

CMOS vers convertisseur RS-232

Une moitié de l'amplificateur TL082 fournit une conversion CMOS vers RS-232. Le diviseur à résistances R1, R2 décale l'entrée négative de 3 volts. Comme la sortie du registre 4015 va en-dessous et au-dessus de 3 volts, la sortie de l'AOP va aux rails d'alimentation. Notez que les données échantillonnées sont inversées sur la ligne RS-232 et que la lecture par logiciel doit prendre ce fait en compte.

Alimentation

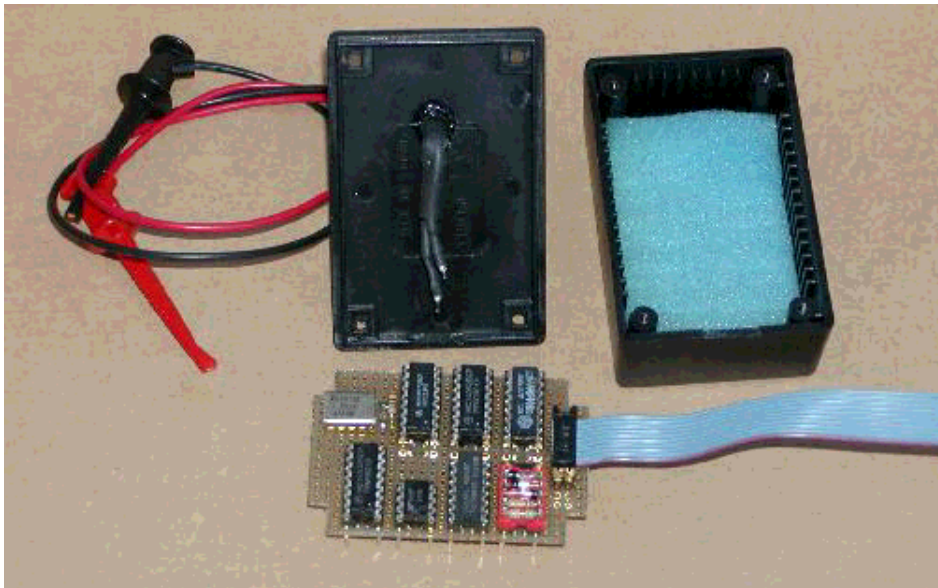
L'alimentation pour le circuit est pris sur la ligne RS-232 de l'ordinateur. Le Tx de l'ordinateur fournit la tension négative pour le convertisseur RS-232 (le TL082P) et DTR fournit la tension positive. Dans le prototype, les diodes D1 et D2 ne sont pas nécessaires. Vous pouvez essayer les circuits sans eux.

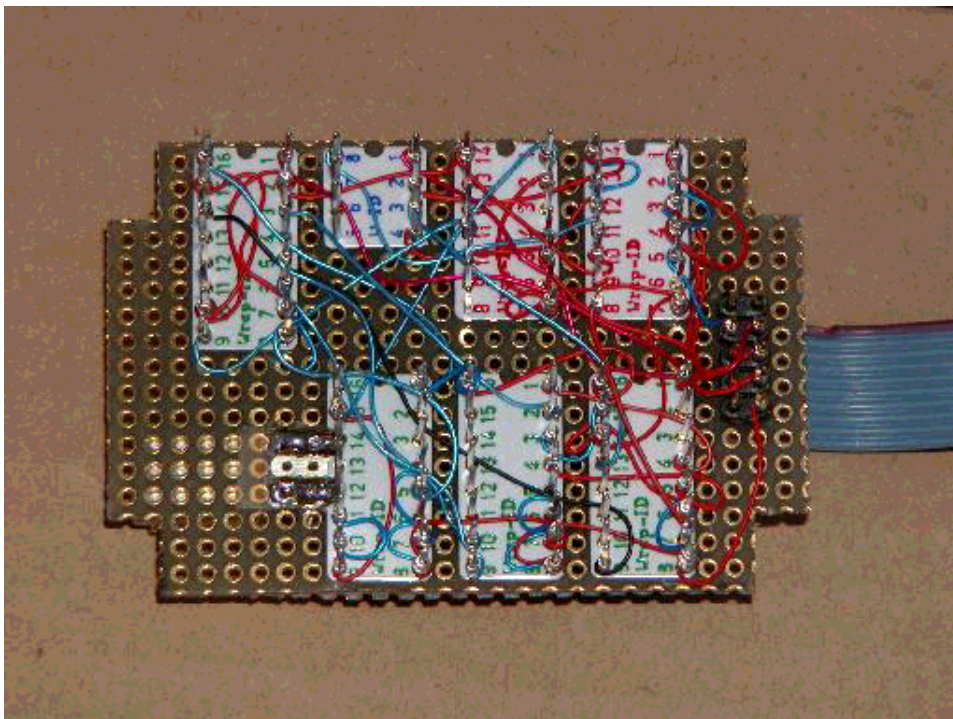
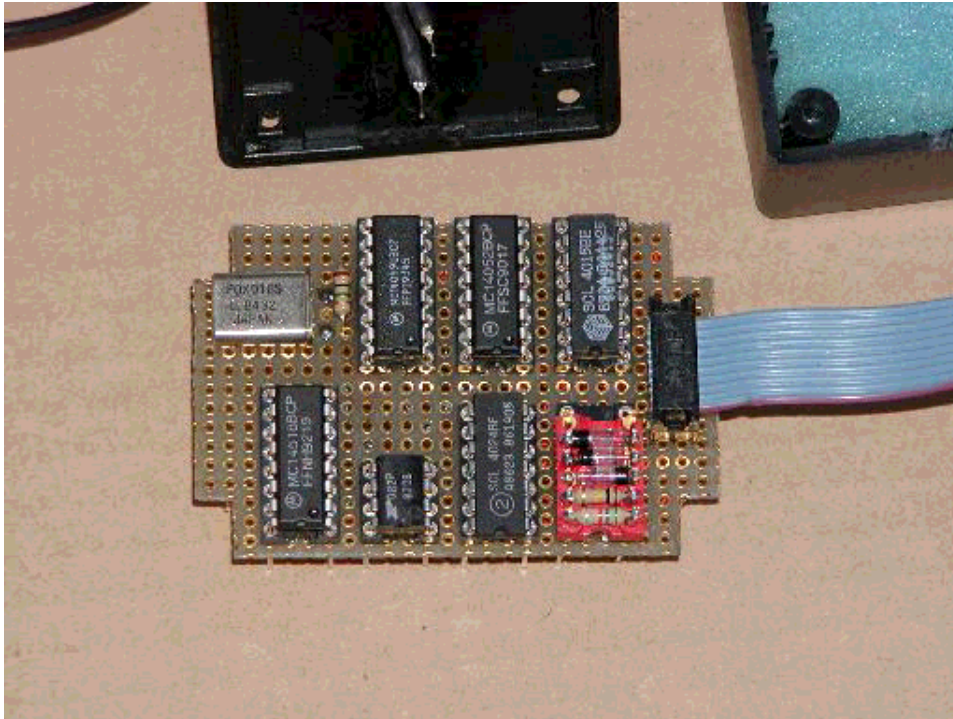
RTS fournit l'alimentation (Vcc) pour le reste du circuit. Dans le prototype, Vcc avait à peu près 7 volts. Votre voltage pour Vcc peut varier en fonction de votre interface RS-232. Le logiciel pilotant l'analyseur doit mettre en haut DTR et RTS.

Construction

Les fréquences dans l'analyseur de données sont relativement basses et toute méthode de fabrication devrait fonctionner. Le wrapping utilisé dans le prototype a donné de bons résultats. Vous installez en premier l'alimentation. Le circuit a son alimentation sur la broche 14/16 exception faite de l'inverseur 4049 qui a son alimentation sur la broche 1.

Ici, vous avez quelques photos pour vous montrer l'approche de la construction.





Programme simple de capture de données

Le programme simple ici (1bitla.c) capture les données séries et peut sortir les données en trois formats. Le premier format est la donnée brute, affichée d'un octet par ligne avec un afficheur à deux lignes. Notez que le programme inverse les données avant l'affichage et que les données sont envoyées avec le bit au poids le plus fort en premier. La sortie brute est invoquée avec la commande -r. Le second format a une valeur (0 or 1) suivi par le compte du nombre d'échantillons que possède la valeur. La sortie de la valeur/compte est invoquée avec l'option -c. Le troisième format de sortie est une valeur (0 or 1) suivi par le nombre de secondes pour lequel l'entrée garde la même valeur. La résolution à 46080 kHz est de 21.7 microsecondes. La sortie de la valeur/compte est invoquée avec l'option -t.

Le programme est invoqué avec
 1bitla [option] port_série

où les options peuvent prendre -r, -c ou -t. La valeur par défaut est (-c).

La conception du programme est assez simple. Nous traitons les options de la ligne de commande, nous ouvrons un port série et faisons une boucle infinie qui lit et affiche les octets. Comme nous ne faisons pas de processus en tâche de fond, la lecture est bloquante.

Vous pouvez vous satisfaire de l'affichage des échantillons sur la console ou de les rediriger dans un fichier pour un traitement ultérieur ou vous pouvez vouloir effectuer une vraie « analyse logique » sur les données en construisant une machine à états pour traiter les échantillons. Les exemples de machines à état incluent le décodage des commandes infrarouge ou le décodage de la position d'une impulsion radio contrôlée avec une sortie modulée.

Sortie d'échantillonnage		
-r	-c	-t
03	0, 0	0, 0.000000
e3	1, 5	1, 0.000065
ff	0, 3	0, 0.000065
03	1, 10	1, 0.000217
e3	0, 6	0, 0.000130
ff	1, 5	1, 0.000109
03	0, 3	0, 0.000065
e3	1, 10	1, 0.000217
ff	0, 6	0, 0.000130
03	1, 5	1, 0.000109
e3	0, 3	0, 0.000065
ff	1, 10	1, 0.000217
03	0, 6	0, 0.000130
e3	1, 5	1, 0.000109
ff	0, 3	0, 0.000065
03	1, 10	1, 0.000217
e3	0, 6	0, 0.000130
ff	1, 5	1, 0.000109
03	0, 3	0, 0.000065
e3	1, 10	1, 0.000217
ff	0, 6	0, 0.000130
03	1, 5	1, 0.000109
e3	0, 3	0, 0.000065
ff	1, 10	1, 0.000217
03	0, 6	0, 0.000130
e3	1, 5	1, 0.000109
ff	0, 3	0, 0.000065
03	1, 10	1, 0.000217
e3		
ff		
03		
e3		
ff		
03		
e3		

Références

- Page de téléchargement pour cet article

- <http://www.linuxtoys.org/>, celui-ci et d'autres projets matériels pour Linux

Site Web maintenu par l'équipe d'édition LinuxFocus © Bob Smith "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Translation information: en --> -- : Bob Smith <bob/at/linuxtoys.org> en --> fr: Iznogood <iznogood/at/iznogood-factory.org>
--	--