

Mise en œuvre d'un système de contrôle réparti sur un réseau hétérogène

P. Andry, T.T. Dang Ngoc

TP Master Pro SIIC – octobre 2006

Abstract

Le but de ce TP est de mettre en commun les connaissances acquises en système embarqué et en réseaux dans le but de construire une application distribuée tirant parti d'un réseau hétérogène pour une application de surveillance à distance.

1 Introduction – présentation

Le Système à concevoir se rapproche de ce qui est décrit à la figure 1:

- Une caméra CCD est placée sur deux servomoteurs qui sont pilotés depuis un PC d'interface, au moyen d'une liaison série.
- Une photo-résistance détectant des variations du niveau de lumière est également reliée à un PC d'interface, par une liaison série.
- Un capteur, simulé par un bouton est relié à un troisième PC d'interface par une liaison série.
- Les PC d'interface et échangent des messages avec un PC de communication auxquels ils sont reliés par un réseau Ethernet. Les communications se font par sockets UDP.
- Le PC de communication relaie les messages entre les PC d'interface et un PC de contrôle qui fait serveur Web. La communication entre le serveur Web et le PC de communication se font en UDP.
- Enfin, un poste client Web peut contrôler l'ensemble grâce à une applet chargée depuis le serveur Web. Cette applet permet
 - de détecter des alarmes lumière et d'envoyer un accusé de réception lors de cette détection;
 - de détecter un contact électrique (bouton pressé). Là encore, un accusé de réception est renvoyé;
 - de déplacer la caméra en donnant les coordonnées angulaires souhaitées pour chacun des deux servo-moteurs (notamment lorsque le bouton est pressé, la caméra doit se replacer à une position prédéfinie correspondant à la direction du bouton).

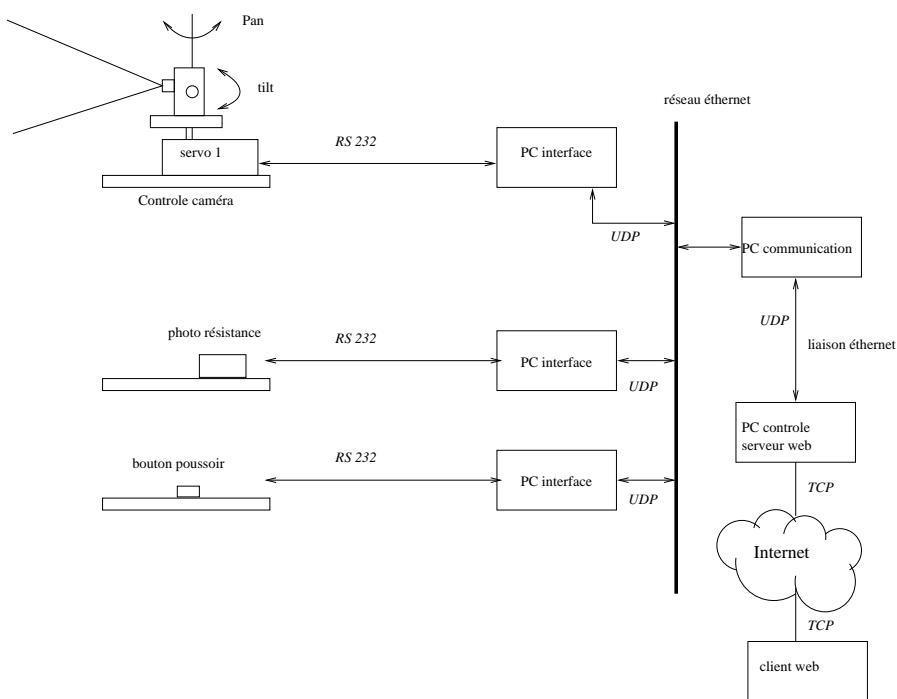


Figure 1: Schéma d'interconnexion des différents éléments du système global

2 Partie « Série – UDP »

2.1 Mise en œuvre de micro-contrôleurs

Afin de permettre le prototypage rapide d'une solution fonctionnelle, nous allons utiliser un Basic Stamp (BS2) de Parallax (voir fig. 2) et une carte de développement. Comme la plupart des systèmes à micro-processeurs servant au contrôle et à l'acquisition de données ne nécessitant pas une puissance de calcul importante (commande de relais ou de servo moteurs, lecture de capteurs de température, lumière, sons...), un micro-contrôleur est utilisé. Les micro-contrôleurs intègrent en plus d'un processeur classique, des périphériques d'entrées-sorties, de la mémoire (ROM, RAM, EPROM, EEPROM,... selon les cas) et des dispositifs de communications évolués (au minimum des liaisons séries mais souvent une liaison I2C, une liaison 1 fils, une liaison CAN, voire une pile TCP/IP pour des liaisons ethernet). Comme de nombreux autres systèmes, le Basic Stamp fait appel à un micro-contrôleur PIC.

L'utilisation d'un BASIC proche de l'assembleur permet de simplifier la prise en main de ces circuits. Les instructions BASIC sont traduites en pcode sur le PC servant à la programmation du BS2 avant d'être téléchargée sur le BS (voir fig. 3). Une fois téléchargé sur le BS2, le pcode est interprété par un programme mis dans la ROM du BS2 (voir le site www.parallax.com pour plus de détails). La contrepartie de cette facilité d'utilisation est une perte de vitesse relativement importante par rapport à l'assembleur ou un langage compilé (sans conséquence pour des applications simples ne posant pas de contraintes importantes de temps réel).

Comprendre le fonctionnement d'une liaison série est indispensable à la compréhension des dispositifs de communications modernes. Vous trouverez en annexe des informations complémentaires sur ce sujet. Le BS2 utilise une liaison série RS232 asynchrone sans contrôle de flux (fig. ??). Les pins d'E/S du BS2 peuvent être transformées à volonté en liaison séries.

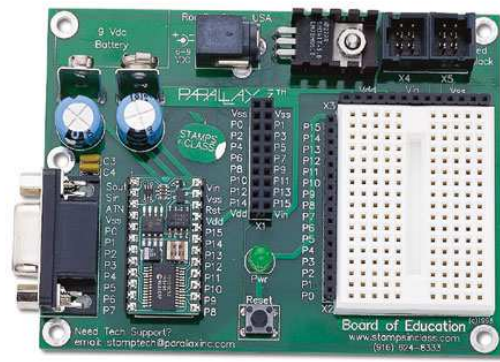


Figure 2: Carte de développement pour les basic stamps

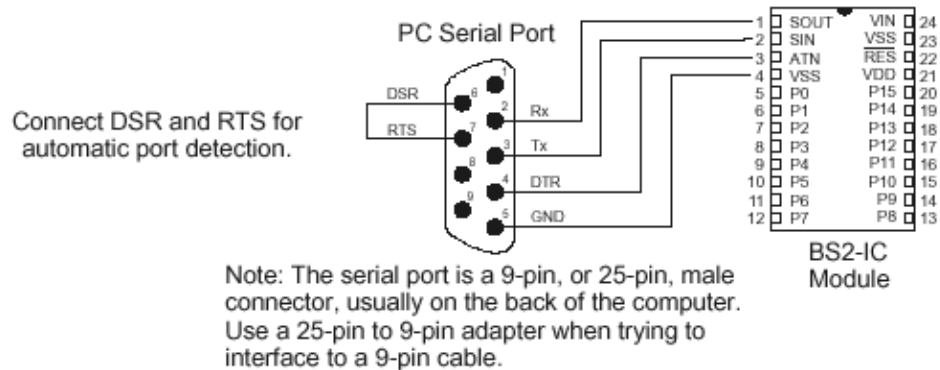


Figure 3: Schéma de connexion du BS2 à une liaison série pour le télé-chargement des programmes et leur debug.

2.2 Réalisation du programme de contrôle de la caméra CCD

Le dispositif de contrôle de la caméra CCD attend qu'on lui envoie des ordres pour positionner les 2 servomoteurs à un azimuth et site particulier. Pour cela nous utilisons une carte avec micro-contrôleur SSCII dont la documentation vous est fournie et qui décrit le protocole de communication. La communication avec la carte se fera *via* une liaison série RS232 par un programme écrit en C. Étant donné que cette carte ne renvoie pas d'informations, c'est le programme exécuté sur le PC d'interface qui se chargera de renvoyer les acquittements (nous assumons que toute commande envoyée au servo-moteur s'est bien déroulée).

Questions :

- *Ecrire le programme sur le basic stamp correspondant à cet esclave*
- *Ecrire le programme en C permettant de contrôler la carte depuis un PC*

La figure 4 montre un exemple de connexion de 2 LED (voyants) sur les pins 9 et 10 du basicstamp. Faites attention au sens de branchement des LEDs sous peine de détruire les composants... Le metplat ou la patte la plus courte de la LED doivent être connectées au (-), l'autre coté au (+) ou Vdd.

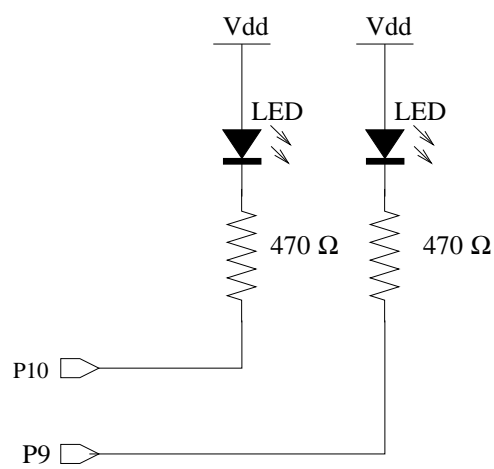


Figure 4: Schéma électrique pour la connexion de 2 leds en sortie d'un BS2.

Remarque: La commande des servomoteurs utilise l'instruction `PULSOUT`. Il ne s'agit pas d'une vraie commande de servomoteur. En effet, après l'exécution de la commande la sortie commandée ne reste pas activée. Le servo n'est donc plus sous contrôle et peut donc bouger librement. De la même manière, si la consigne est loin de la position courante du servo, il y a toutes les chances qu'en exécutant une seule fois la commande le servo n'ait pas eu le temps d'atteindre la position désirée. Il faudra donc relancer plusieurs fois cette commande. On peut à cette occasion voir les effets négatifs du régulateur en position utilisé sur le servo. Le servo peut un peu osciller (le programme de contrôle devra en tenir compte le cas échéant).

2.3 Réalisation de programmes interfaçant les capteurs (déclenchement des alertes)

On réalise deux cartes permettant de réaliser des télémessures et de générer des alarmes lorsque certains signaux dépassent un seuil donné.

- détection d'une variation du niveau de lumière (à partir d'un seuil de variation)
Si une variation de lumière est détectée, un message du type **P:xxx,yyy** est émis vers le PC interface correspondant (xxx représentant le numéro du capteur considéré et yyy la valeur obtenue).
- détection sur contact électrique (bouton poussoir). Un premier message est émis lorsque l'on appuie sur le contact (**I:xxx,1**) et un deuxième au moment où l'on relâche le bouton (**I:xxx,0**) avec xxx le numéro du contacteur considéré.

Le branchement des capteurs de lumière sur le BS2 est décrit à la figure 5.

La passerelle vers le réseau doit être en attente de messages car il n'est pas possible de savoir quand ils arriveront. Le PC devra donc être en permanence en attente de message sur la liaison série.

Remarque : Avec une version de **LINUX** classique, il n'est pas possible d'assurer que les messages seront traités dans les microsecondes suivant leur réception (il faut soit patcher le noyau de Linux pour lui rajouter des capacités temps réel soit passer à un OS temps réel dur - voir le cours de temps réel).

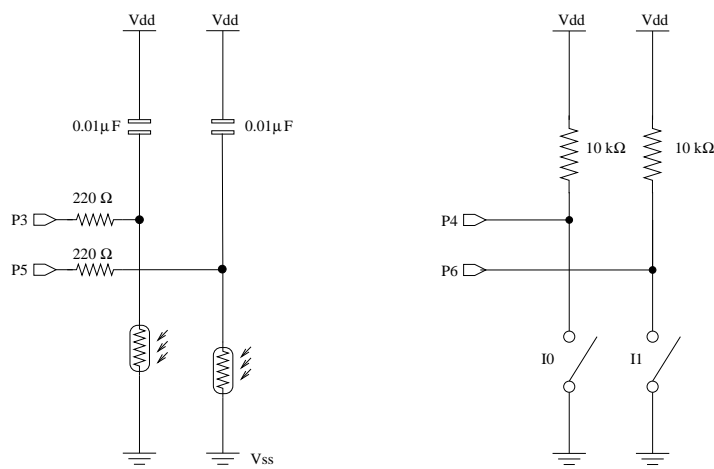


Figure 5: à gauche: Connexion des photo-résistances sur le BS2 en utilisant un circuit RC. Le BS2 va mesurer le temps de charge de la capacité pour en déduire la valeur de la résistance sensible à la lumière. A droite: Connexion à des entrées binaires de 2 interrupteurs I0 et I1.

3 Partie « Réseaux »

3.1 Le PC de communication

La principale difficulté dans la gestion du PC de communication est le multiplexage: il reçoit des messages de trois ou quatre sources distinctes (le(s) PC d'interface et le serveur Web) suivant le protocole UDP.

3.2 Le serveur Web

Vous utiliserez le serveur Web situé sur la machine 10.4.106.6, sur laquelle le compte «*etudiant*» (mdp «*etudiant*») a été créé. Ce compte est propriétaire d'un sous-répertoire de `/var/www/html`, où vous placerez vos pages et votre applet. On prendra évidemment soin de créer un sous-répertoire par groupe de TP afin de ne pas interférer avec les applets des autres groupes!

3.3 Protocoles de communication

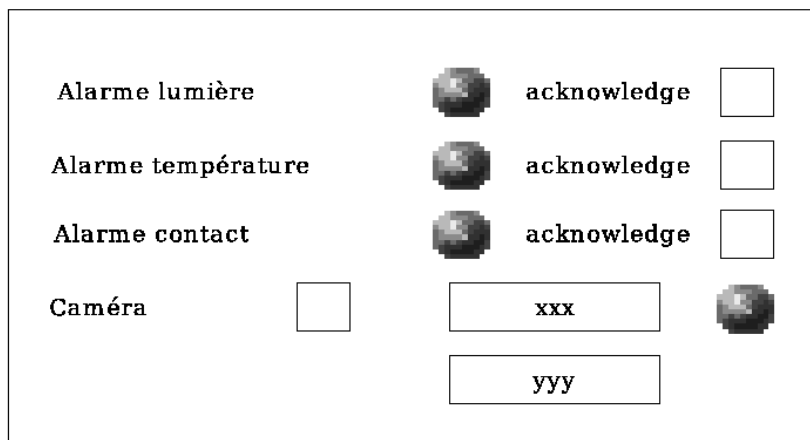
Le tableau ci-dessous résume le protocole à mettre en œuvre pour gérer les communications entre le serveur Web et le PC de communication. On rappelle que celles-ci se font après établissement d'une connexion TCP entre les deux PC. La première colonne représente suivant les cas des événements provenant à l'origine de l'un des PC d'interface.

Événement extérieur	Emission PC comm.	Emission serveur Web
appui bouton	B	
ack. appui bouton		b / b,e
variation lumière	L	
ack. variation lumière		l / l,e
déplacement caméra		C,n°,angle
ack. déplacement caméra	c / c,e	

N.B.: les *acknowledge* sont suivis d'un «,e» en cas d'erreur sur le traitement de la commande.

3.4 Interface graphique de contrôle

L'IHM exposée par l'applet peut avoir l'allure indiquée par la figure suivante:



Les cercles grisés indiquent des composants de type «lumière» (en Java, des `JButton` par exemple), pouvant être remplis de vert (état normal) ou de rouge (message reçu, ou erreur de déplacement de la caméra). Les carrés représentent des boutons (`Button` ou `JButton`) qui doivent être appuyés par l'utilisateur pour déclencher l'acknowledge ou la demande de déplacement de la caméra. Enfin, les rectangles sont des champs texte (`TextField`) permettant d'entrer les angles souhaités pour les servomoteurs de la caméra.

L'interface proposée n'est bien sûr qu'une indication, l'essentiel étant que toutes les informations soient disponibles.

4 Pour aller plus loin : un système d'alarme à distance

Il est intéressant de récupérer les images transmises par la caméra vidéo lorsque les capteurs détectent une alarme. Pour cela, il va falloir envoyer des images depuis la caméra jusqu'au client web pour que l'utilisateur distant puisse visualiser l'intrusion. Vous devrez adapter vos programmes pour que les images soient envoyées sur le réseau.

Vous définirez vous-même le mode de fonctionnement de cette nouvelle fonctionnalité (décision du client de voir ou non les images suivant les signaux reçus et/ou décision affichage automatisée)