

REDES DE COMUNICAÇÃO EM AMBIENTES INDUSTRIAIS

Trabalhos Práticos para 2001

1. Upgrade do Loader para placas CANivete

Na actual estrutura de Loader para placas CANivete apenas se pode enviar, através de um *gateway* ligado a RS232, um programa de cada vez para cada placa ligada ao barramento CAN. O processo é o seguinte:

- a) Um programa no PC entra em contacto com a placa *gateway* pela porta série (RS232).
- b) Se o programa a carregar (ficheiro .hex gerado por um compilador cruzado, por exemplo o da Keil) se destinar à placa *gateway* (esta pode ser o alvo), esta vai recebendo os dados e guarda-os na sua memória que há-de vir a ser de código.
- c) Se o programa se destina a um nó da rede, a *gateway* vai recebendo o programa aos poucos e vai enviando-o para o processador do nó via CAN, recebendo confirmações sucessivas.

Alterações a efectuar:

- a) Introduzir um mecanismo de *Multicast*: o *gateway* envia as partes comuns do programa para um endereço global aceite por todos ou por parte dos nós sem receber confirmações. Em seguida envia individualmente as partes específicas (em geral apenas bytes de endereço) e verifica a correcta recepção do programa.
- b) Introduzir comando de RUN: depois do envio do programa tem de se dar início à sua execução no nó.

2. Gateway para porta paralela

O *gateway* do sistema actual recebe dados (o programa a enviar) por RS232. Integra um controlador de barramento que faz parte do microprocessador. Neste trabalho pretende-se substituir o *gateway* baseado em processador por um *gateway* não inteligente o qual deve integrar um controlador CAN *stand-alone* (não integrado no CPU). A geração dos sinais de interface com o controlador é feita por manipulação de baixo nível da porta paralela do PC.

Além do hardware associado à porta, é necessário alterar o software do PC que lê o ficheiro do programa a enviar (.hex) para que este passe a aceder directamente ao controlador.

Nota: O controlador CAN trata das operações de baixo nível das comunicações tais como acesso ao barramento, detecção de erros, codificação, ...

3. Sistema árbitro com co-processador

Alguns nós de um sistema distribuído podem ter tarefas especiais mais complexas e requerendo maior capacidade de processamento. Tal pode acontecer para execução de algoritmos de controlo, para determinação do tráfego do barramento, etc. Como os processadores habitualmente utilizados (os quais integram controladores CAN) são de baixo poder de cálculo, a introdução de um co-processador facilita a implementação daquelas tarefas.

O co-processador deverá neste caso ser um microprocessador (por exemplo 188 ou outro) o qual é controlado pelo CPU do nó. Este pode receber programas e coloca-os na memória de código do co-processador. Pode também colocar dados na memória de dados e retirar de lá outros dados (p.ex. resultantes de cálculos). Pode ainda comandar a execução das tarefas e o reset do co-processador.

4. Nodos CAN para diferentes processadores

Neste trabalho pretende-se construir nodos CAN para outros processadores. Uma das opções é a do Kit188 ao qual poderá ser ligado um controlador CAN, por exemplo o SJA1000 o qual pode funcionar num modo compatível com o controlador embedded no 80592 (CPU utilizado na placa CANivete). Os alunos poderão optar por outro processador à sua escolha, desde que existam ferramentas de desenvolvimento mínimas.