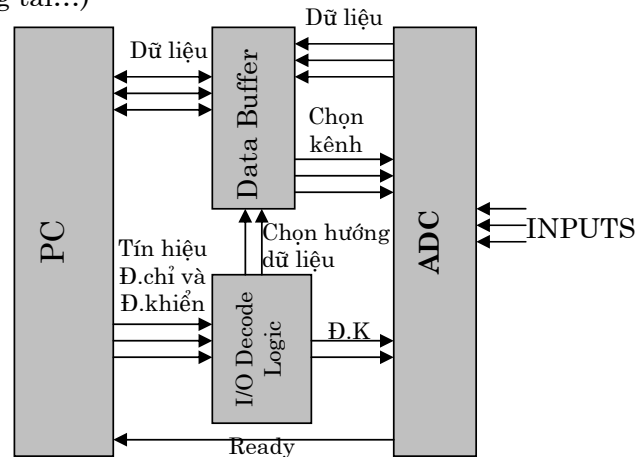


THỂ MẠNH SỬ DỤNG VI XỬ LÝ THIẾT KẾ MẠCH TRONG ĐO LƯỜNG VÀ ĐIỀU KHIỂN

Trong hầu hết các hệ máy hiện đại hiện nay, máy vi tính- với vai trò bộ não sẽ thu thập , xử lý dữ liệu và điều khiển các thiết bị ngoại vi. Kỹ thuật số ngày càng phát triển, mức tích hợp ngày càng cao, vi xử lý ra đời cùng với sự phát triển phần mềm buộc các nhà thiết kế phần cứng phải có cách nhìn mới. Trong đo lường và điều khiển ,vi xử lý giúp cho việc thu thập dữ liệu và điều khiển ngoại vi trở nên mềm dẻo thân thiện , việc thiết kế mạch điện tử cũng đơn giản và gọn nhẹ hơn nhiều.

Trong thu thập dữ liệu các nhà thiết kế truyền thống thường thiết kế CARD ghép nối ADC đa kênh để chuyển tải thông tin từ các thiết bị ngoại vi về máy tính (như thông tin về nhiệt độ trong lò luyện gang, thời gian và trọng lượng trong cân băng tải...)

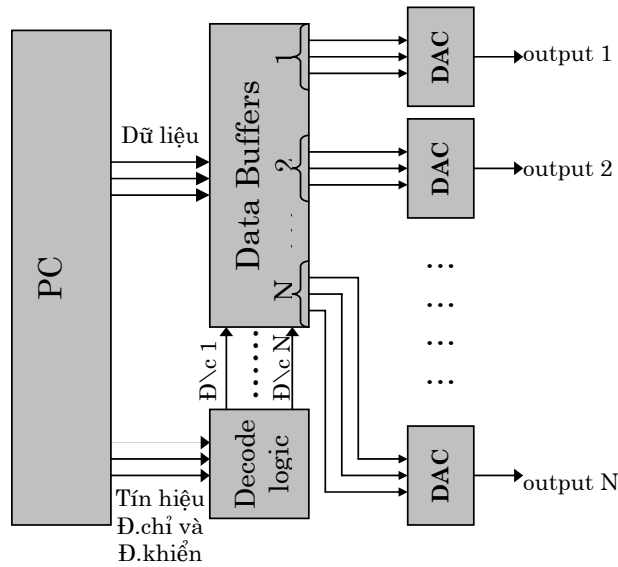


CARD GHÉP NỐI ADC

Quá trình giao tiếp: Khi cần thu thập các thông tin từ ngoại vi, máy tính sẽ xuất ra tín hiệu địa chỉ, dữ liệu và IOW (in/out write) sau đó đưa ra tín hiệu AEN (address enable). Lập tức khối I/O Decode Logic (giải mã lô gíc vào/ra) điều khiển Data Buffer (đệm số liệu) chuyển dữ liệu đến ADC để chọn ngoại vi cần truy cập. Tín hiệu tương tự theo kênh được chọn chuyển đổi thành tín hiệu số nhờ ADC. Máy tính sẽ đưa ra tín hiệu địa chỉ và tín hiệu IOR (in/out read). Khối I/O Decode Logic điều khiển Data Buffer chuyển dữ liệu từ ADC tới các thanh ghi máy tính và điều khiển ADC làm việc. Khi dữ liệu số hoá ổn định, ADC gửi tín hiệu Ready thông báo cho máy tính có thể đọc được dữ liệu từ ngoại vi. Lúc này dữ liệu từ thanh ghi được chuyển vào bộ nhớ.

Sau khi cập nhập các thông tin về ngoại vi, để điều khiển chúng vận hành theo chu trình định sẵn (ví dụ nếu thông tin từ ngoại vi gửi về cho biết :nhiệt độ trong lò luyện gang giảm so với nhiệt độ chuẩn, máy tính sẽ điều khiển

ngoại vi tác động các yếu tố để giảm nhiệt độ ...), các nhà thiết kế thường thiết kế CARD ghép nối DAC kinh điển sau.

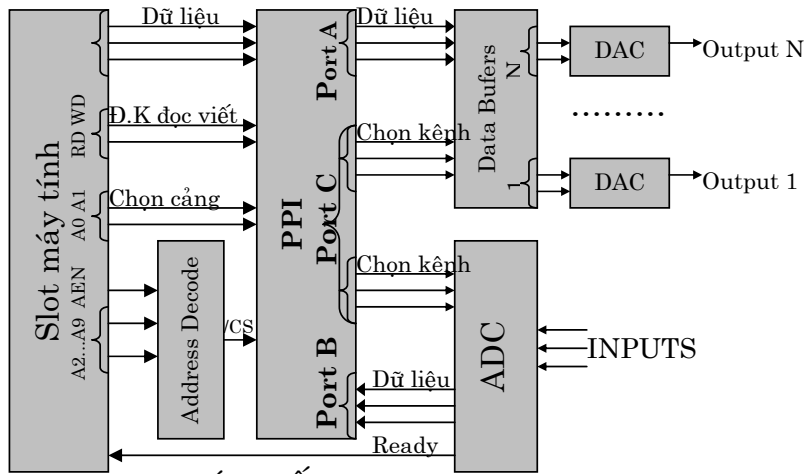


CARD GHEP NOI DAC N KENH

Quá trình giao tiếp: Điều khiển một trong N kênh, máy tính ra các tín hiệu địa chỉ tương ứng, các tín hiệu dữ liệu và điều khiển. Decode Logic (giải mã logic) nhận tín hiệu địa chỉ và điều khiển sẽ cho dữ liệu qua một trong các bộ đệm số liệu tới DAC điều khiển ngoại vi.

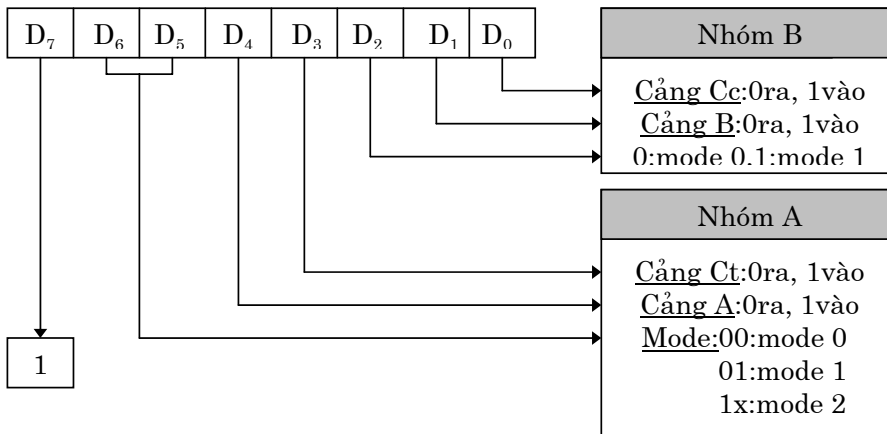
Với các phương pháp ghép nối kinh điển trên mỗi cổng vào/ra cần một khối giải mã và khối đệm số liệu nên khối giao tiếp khá cồng kềnh và chiều dữ liệu do phần cứng quy định. Hơn nữa, khi thiết kế CARD ghép nối AD, DA đa kênh, do nhiều linh kiện nên họ thường phải thiết kế CARD AD và DA riêng rẽ nhau. Trong rất nhiều trường hợp, PPI-8255 (Programmable Peripheral interface) vi mạch ghép nối song song điều khiển bằng chương trình làm cho công việc thiết kế ghép nối CPU với các thiết bị ngoại vi trở nên đơn giản hơn nhiều. Sử dụng PPI trong thiết kế CARD ghép nối AD-DA có ưu điểm sau:

- Tiết kiệm không gian thiết kế mạch
 - Chỉ cần một bộ giải mã địa chỉ
 - Có ba cổng vào/ra có thể thay đổi chiều tùy ý bằng các câu lệnh đơn giản
 - Bản thân các cổng là các bộ đệm dữ liệu.
 - Cổng C gồm hai nhóm có thể vào/ra độc lập nên tối ưu trong một số mạch cụ thể.
 - Sử dụng các mode hoạt động sẽ tạo ra sự trao đổi dữ liệu chặt chẽ.
 - Rất an toàn trong ghép nối máy tính với ngoại vi...
- Sau đây là một phương án thiết kế CARD ghép nối AD-DA có sử dụng PPI



CARD GHÉP NỐI AD-DA

Lập trình cho PPI: Thực hiện bằng việc viết từ điều khiển vào thanh ghi từ điều khiển (Control Word Register) để xác lập chế độ hoạt động. Sau đây là định dạng thanh ghi từ điều khiển:



Giả sử địa chỉ cơ sở là: adr , lúc đó địa chỉ cổng A là: adr , cổng B là: $adr+1$, cổng C là: $adr+2$ và từ điều khiển sẽ là: $adr+3$. Trường hợp cụ thể trên ta chọn mode 0, cổng A, C thấp (C0..C3) và C cao (C3..C7) ra; cổng B vào nên ta viết vào từ điều khiển bằng dòng lệnh ví dụ bằng ngôn ngữ Pascal

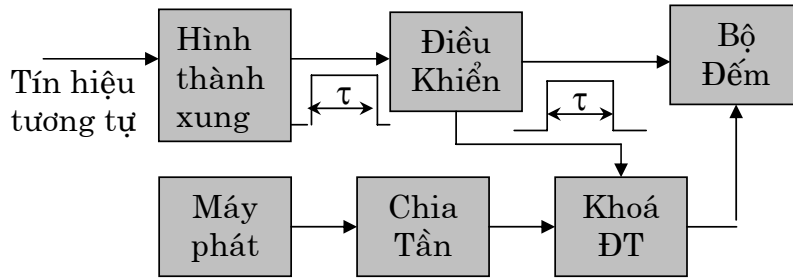
`Port[adr+3]:= $86 {1000 0100} {Mode 0; Cảng A,Ct,Cc ra; Cảng B vào}`

Sau dòng lệnh này quá trình giao tiếp giữa máy tính và ngoại vi diễn ra tương tự như cách truyền thống trình bày ở trên.

Vậy chỉ thêm một dòng lệnh thiết kế phần cứng sẽ đơn giản hơn nhiều nhất là trong thiết kế giải mã và đệm dữ liệu.

Một ví dụ phổ biến nữa trong đo lường và điều khiển là đo thời khoảng thời gian τ chính xác (ví dụ: đo chu kỳ tín hiệu, đo thời gian giữa hai sự kiện,

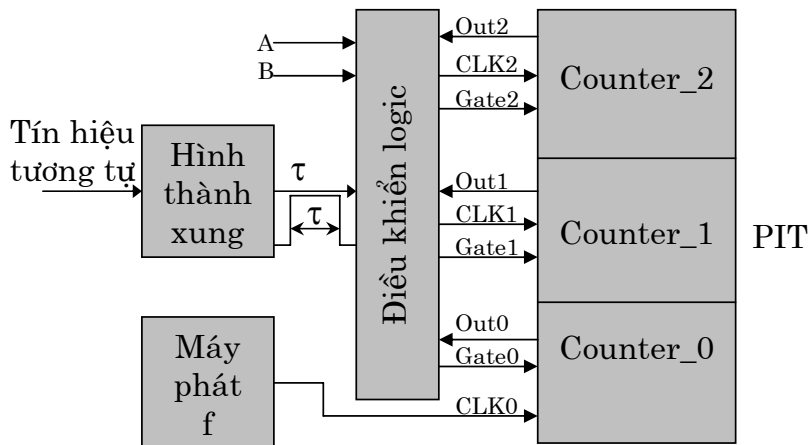
gián tiếp đo vận tốc của cân bằng tải...) .Theo cách truyền thống sơ đồ khối được trình bày như sau.



Nguyên lí làm việc: Giả sử τ là thời gian kéo dài hai sự kiện, qua khối hình thành xung ,tín hiệu tương tự được chuẩn hoá mức TTL và tạo được xung độ rộng τ giây. Sườn lên của xung τ khởi phát bộ đếm đồng thời mở khoá điện tử cho các xung của máy phát thạch anh được chia tần đi qua. Bộ đếm đếm số xung gửi qua khoá điện tử. Quá trình đếm kết thúc khi sườn xuống xung τ . Giả sử f là tần số máy phát xung vuông góc, qua khối chia tần (gồm M trigô nôm tiếp nhau) thì tín hiệu lới ra có tần số $F=f/2^M$. Nếu N là số xung trong τ giây thì $\tau=N \cdot F$ và sai số $=1/N$.

Phương pháp này có nhược điểm : dải đo hẹp, sai số thay đổi tỷ lệ thuận với τ , tràn số nếu τ lớn, F lớn khối chia tần rất cồng kềnh,cứng nhắc...Chip PIT-8253/54 (Programmable interval Timer)_chíp định thời gian điều khiển bằng chương trình khác phục được nhược điểm trên.

Vi mạch 8253 hoặc 8254 gồm ba bộ đếm lùi 16 bit độc lập nhau và có thể chương trình hoá.Mỗi bộ đếm có thể xem là bộ chia tần có hệ số có thể điều khiển được (hệ số chia tần tối đa $2^{16}-1$). Các bộ đếm có các cửa Gate0 đến Gate2 để cho phép đếm (Gate=0: không cho phép đếm, Gate=1: cho phép đếm). Bạn đọc có thể tìm hiểu kĩ về quá trình lập trình của vi mạch PIT trong nhiều tài liệu. Sau đây là một phương án thiết kế sử dụng PIT dải đo rộng:



Khối Điều khiển logic sẽ lựa chọn các phương án làm việc theo các bit điều khiển A,B

Gate0= $A/B\tau + A/B$;

Gate1= $AB\tau + A/B$; CLK1= $AB \cdot Out0$;

Gate2= $A/B\tau$; CLK2= $A/B \cdot Out1$.

AB: 00 Gate0= τ (Chỉ sử dụng bộ đếm 0)

AB: 01 Gate0=1, Gate1= τ , CLK1=Out0. (Nối tiếp bộ đếm 0 và 1)

AB: 10 Gate0=1, Gate1=1, Gate2= τ , CLK1=Out0, CLK2=Out1. (Nối tiếp ba bộ đếm)

Comment [H1]:

Ta xác lập chế độ hoạt động (chọn bộ đếm, nạp hệ số chia tần và mode làm việc cho các bộ đếm), đọc bộ đếm, xử lý kết quả sẽ được thời gian τ . Với thiết kế này, hệ có các ưu điểm sau:

-Cố định được sai số bằng cách nạp các bộ đếm hệ số chia tần phù hợp với tín hiệu cần đo tuần hoàn.

-Khắc phục hiện tượng tràn bằng xử lý tình tế với bộ đếm hay bằng phương pháp dùng biến lưu trữ số lần tràn và Reset bộ đếm.

-Cấp chính xác cao.

-Có thể tạo được tín hiệu có tần số từ f tới $f/(2^{16}-1)^3$ (nối tiếp 3 bộ đếm)

-Dải đo rộng.

-Phối hợp các mode đếm tạo được những được xung theo ý muốn.

Khó mà nói hết những thế mạnh sử dụng VXL trong đo lường và điều khiển. Trên chỉ là hai ví dụ cụ thể minh chứng cho ưu điểm thiết kế mạch sử dụng chúng so với các phương pháp kinh điển.

Tuy nhiên phương pháp truyền thống vẫn tỏ ra phù hợp với các yêu cầu cụ thể.

Do khuôn khổ tạp chí có hạn, các bạn muốn tìm hiểu kỹ nên tìm đọc: Cấu trúc máy tính (Trần Quang Vinh), Đo lường và điều khiển bằng máy tính (Ngô Diên Tập), Kỹ thuật vi xử lý (Văn Thế Minh), PC programmer's Bible (Microsoft Press-A Division of Microsoft Corporation, 1993), Principles of Computer Hardware (PWS-KENT Publishing Company, Boston, 1992)...