

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	1/6

Tutorial n^o 06.01

Gửi đến: Đoàn Hiệp, www.picvietnam.com

Nội dung: Sử dụng các module ADC và MCPWM của dsPIC

MICROSOFT WORD

Tóm tắt:

Đây là tutorial 5 về lập trình với dsPIC. Tutorial này hướng dẫn cách sử dụng module ADC và MCPWM của dsPIC, là những module ngoại vi có sẵn trong các chip dsPIC. Các ví dụ được viết bằng hợp ngữ và ngôn ngữ C (dùng trình dịch C30 Compiler của Microchip), và trong đa số trường hợp hai phiên bản hợp ngữ và C30 là hoàn toàn tương đương.

1. Giới thiệu

Tutorial này giới thiệu các module ADC và MCPWM của dsPIC. Trước tiên, các thông tin chung về các module ADC và MCPWM sẽ được giới thiệu, cùng với một trình tự thiết lập chế độ làm việc cho mỗi module. Kế đến, các ví dụ sẽ minh họa cách sử dụng các module này trong thực tế. Ví dụ 5-1 sẽ giới thiệu cách điều chỉnh độ rộng xung theo một giá trị tương tự được đọc về thông qua một ngõ vào analog. Ví dụ 5-2 minh họa một chế độ làm việc khác của module ADC. Một số chú ý sẽ được nêu ra khi cần thiết.

2. Các quy ước trong tài liệu

Mô tả	Biểu thị	Ví dụ
Font Palatino Linotype:		
In nghiêng	Tài liệu tham khảo	<i>dsPIC30F/33F Programmer's Reference Manual</i>
Viết hoa chữ đầu	Một menu, hộp thoại	Chọn Project Wizard
Đặt trong dấu nháy kép	Tên riêng (của tập tin, project, ...)	"Vidu3"
Font Courier:		
Courier thường	Mã nguồn	mov #0x8010, W0
	Tên tập tin	Vidu3-1.s

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	2/6

3. Các ví dụ về module ADC và MCPWM

Bạn hãy chép các tập tin nguồn được đính kèm vào một thư mục nào đó mà bạn sẽ dùng để tạo ra project “Vidu5”.

3.1. Giới thiệu về các module ADC và MCPWM

Các chip dsPIC có module ADC thuộc một trong hai dạng: 10-bit với tốc độ lên đến 1 MSPS (triệu mẫu/giây) hay 12-bit với tốc độ lên đến 200 kSPS (nghìn mẫu/giây). Các module ADC được thiết kế nhằm phục vụ cho các mục đích khác nhau. Trong các dsPIC thuộc dòng điều khiển động cơ (mà chip 4012 đang được xét là một trong số đó), module ADC là loại 10-bit với tốc độ cao, nhằm đáp ứng yêu cầu về tốc độ lấy mẫu của các bộ điều khiển truyền động điện. Các bộ ADC trong các dsPIC sử dụng thuật toán chuyển đổi SAR, do đó có thời gian chuyển đổi đã được xác định trước.

Việc chuyển đổi tín hiệu từ dạng tương tự sang dạng số bao gồm hai giai đoạn: giai đoạn lấy mẫu tín hiệu (sampling) và giai đoạn chuyển đổi (conversion). Bạn có thể bắt đầu giai đoạn lấy mẫu bằng tay hay tự động. Thời điểm chấm dứt lấy mẫu (và bắt đầu chuyển đổi) có thể do người dùng xác định (bằng cách tắt bit SAMP) hay được một ngoại vi nào đó xác định (một timer dành riêng, module MCPWM, Timer 3, hay chân INT0). Như vậy thời gian lấy mẫu là khác nhau với các thiết lập khác nhau. Giai đoạn chuyển đổi dữ liệu cần có 12 xung clock cho module ADC, với một chu kỳ xung clock T_{AD} có thể được chọn là từ $T_{CY}/2$ đến $32T_{CY}$. T_{CY} chính là thời gian của một chu kỳ máy.

Module ADC của dsPIC có 4 bộ khuếch đại S/H (Sample and Hold), được đánh địa chỉ là kênh 0 đến kênh 3. Bạn có thể chỉ dùng kênh 0, hay dùng kênh 0 và 1, và cũng có thể dùng cả 4 kênh cho việc thu thập dữ liệu. Kênh 0 là kênh linh hoạt nhất trong việc chọn lựa các ngõ vào tương tự. Bộ đệm của module ADC có thể chứa được tối đa 16 kết quả.

Module ADC cần được thiết lập cấu hình theo những bước sau:

- Chọn các chân dùng làm ngõ vào analog bằng các bit ADPCFG<15:0>
- Chọn nguồn điện áp chuẩn bằng các bit ADCON2<15:13>
- Chọn tốc độ xung clock cho module ADC phù hợp với tốc độ dữ liệu và tốc độ của bộ xử lý bằng các bit ADCON3<5:0>
- Xác định bao nhiêu kênh S/H sẽ được dùng bằng các bit ADCON2<9:8> và ADPCFG<15:0>
- Xác định cách thức lấy mẫu bằng các bit ADCON1<3> và ADCSSL<15:0>
- Chọn thứ tự lấy mẫu/chuyển đổi thích hợp bằng các bit ADCON1<7:0> và ADCON3<12:8>
- Chọn cách biểu diễn kết quả chuyển đổi trong bộ đệm bằng các bit ADCON1<9:8>

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	3/6

- Chọn tốc độ tạo ngắt bằng các bit $ADCON2<5:9>$
- Bật module ADC bằng bit $ADCON1<15>$

Nếu có sử dụng ngắt, cần thiết lập thêm cấu hình cho ngắt A/D:

- Xóa bit $ADIF$
- Chọn độ ưu tiên cho ngắt A/D

Chú ý: Nguồn áp chuẩn ngoài cần được dùng cho các tốc độ chuyển đổi từ 500 kSPS trở lên.

Các bit $ADCS<5:0>$ ($ADCON3<5:0>$) được dùng để thiết lập tốc độ xung clock cho module ADC. Giá trị của $ADCS<5:0>$ được xác định từ công thức:

$$ADCS = \frac{2T_{AD}}{T_{CY}} - 1$$

Giá trị được chọn của T_{AD} không được phép nhỏ hơn 83.33 ns.

Nếu việc chuyển đổi A/D được thực hiện trong chế độ Sleep, cần phải dùng bộ dao động RC dành riêng cho module ADC bằng cách đặt bit $ADRC(ADCON3<7>)$. Khi đó giá trị của các bit $ADCS<5:0>$ không có ý nghĩa đối với hoạt động của module ADC.

Module MCPWM có nhiều ngõ ra được điều chế độ rộng, thích hợp cho các ứng dụng điều khiển động cơ, hay các nguồn xung. Một số dsPIC có nhiều module MCPWM, và một module có thể có 6 hay 8 ngõ ra. Đối với dsPIC30F4012 đang được xét ở đây, chip có một module với 6 ngõ ra, gồm 3 cặp bổ phụ (complementary).

Các dsPIC với module MCPWM có 6 ngõ ra sẽ có 3 bộ điều chế độ rộng độc lập, với độ phân giải là $T_{CY}/2$, tương ứng với 3 cặp ngõ ra bổ phụ. Các cặp ngõ ra này cũng có thể được sử dụng độc lập, với trạng thái tích cực được thiết lập bằng các bit cấu hình. Module MCPWM hỗ trợ nhiều kiểu ngõ ra: giống hàng theo cạnh hay theo điểm giữa, chế độ tạo xung duy nhất, hay giống hàng theo điểm giữa có cặp nhậm kép.

Module MCPWM có thể tạo ngắt đặc biệt để đồng bộ các phép chuyển đổi A/D. Nó cũng hỗ trợ tạo thời gian chết (dead time) cho chế độ dùng ngõ ra bổ phụ, và có một số chân phát hiện sự cố bằng phản cứng.

Có thể khởi tạo hoạt động của module MCPWM theo trình tự sau:

- Xác định thời gian cho một chu kỳ xung bằng cách đặt một giá trị 15-bit vào thanh ghi ngưỡng đếm chu kỳ xung $PTPER$, theo công thức

$$PTPER = \frac{f_{CY}}{f_{PWM} \times PTMR \text{ Prescaler}} - 1, \text{ cho giống hàng theo cạnh}$$

$$PTPER = \frac{f_{CY}}{f_{PWM} \times PTMR \text{ Prescaler} \times 2} - 1, \text{ cho giống hàng theo điểm giữa}$$

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	4/6

- Thiết lập thanh ghi SEVTCMP cho việc tạo ngắt sự kiện đặc biệt (nếu cần)
- Thiết lập chế độ dùng các chân PWM bằng thanh ghi PWMCON1
- Thiết lập việc dùng overdrive trong thanh ghi OVDCON
- Đặt giá trị chu kỳ nhiệm vụ vào các thanh ghi PDCx tương ứng. Chú ý là độ phân giải của chu kỳ nhiệm vụ là $T_{cy}/2$, còn độ phân giải của chu kỳ xung là T_{cy} . Do đó, chẳng hạn $PTPER = 0x7F$ thì $PDCx = 0xFF$ mới là chu kỳ nhiệm vụ = 100%.
- Thiết lập postscaler cho việc tạo ngắt sự kiện đặc biệt, cách thức cập nhật thanh ghi chu kỳ nhiệm vụ và việc đồng bộ overdrive bằng thanh ghi PWMCON2
- Thiết lập prescaler và postscaler cho thanh ghi đếm chu kỳ xung, chế độ đếm chu kỳ xung và kích hoạt việc đếm chu kỳ xung bằng thanh ghi PTCON

Khi các ngõ ra PWM được sử dụng ở chế độ bổ phụ, bộ tạo thời gian chết được tự động cho phép để tránh cho các linh kiện công suất bị trùng dẫn. Mỗi cặp ngõ ra bổ phụ được dành riêng một thanh ghi đếm thời gian chết 6-bit, với một prescaler giúp cho việc tạo thời gian chết được linh hoạt hơn.

Một (hay nhiều) chân phát hiện sự cố cũng có thể được dùng để đưa các ngõ ra PWM về một trạng thái xác định khi xảy ra sự cố, bằng tác động của phần cứng.

Tín hiệu sự kiện đặc biệt luôn luôn được module MCPWM tạo ra, và module ADC có thể dùng nó để đồng bộ việc chuyển đổi với việc tạo xung PWM. Tín hiệu sự kiện đặc biệt này có thể dùng với một postscaler từ 1:1 đến 1:16, và điều này sẽ có ích nếu việc chuyển đổi A/D không cần phải được thực hiện ở mỗi chu kỳ PWM.

Thông tin chi tiết về module ADC và MCPWM có thể được tìm thấy ở các chương 15 (module MCPWM) và 17 hay 18 (module ADC) của tài liệu *dsPIC30F Family Reference Manual* (DS70046-phiên bản hiện tại là DS70046E).

3.2. Ví dụ 5-1

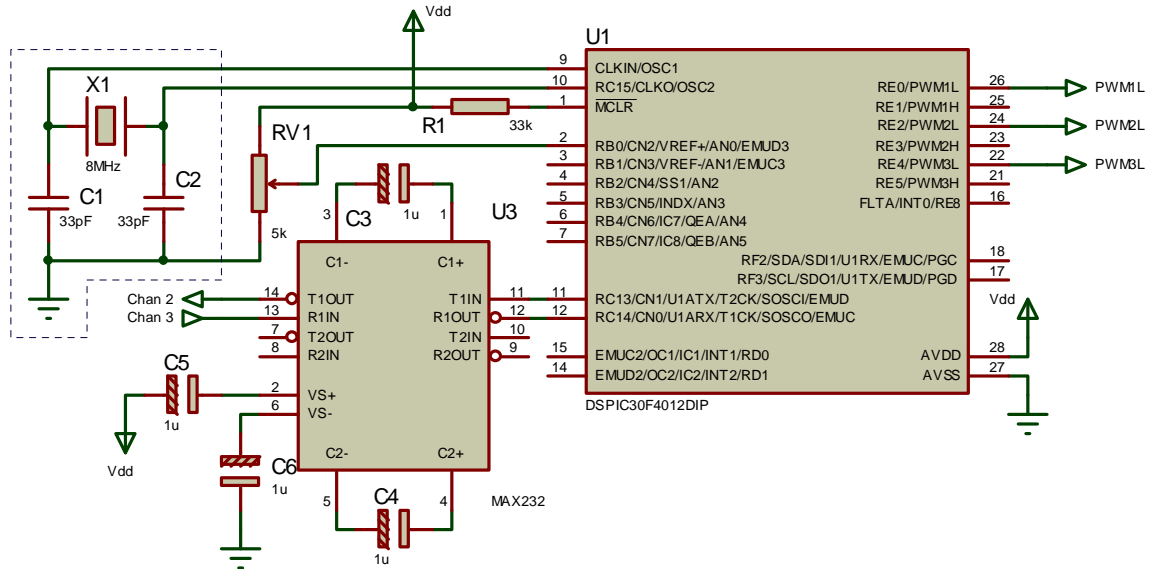
Làm theo các bước như trong tutorial 1 để tạo ra một project có tên là “Vidu5”, sau đó bạn thêm vào project tập tin nguồn “Vidu5-1.s” và kịch bản liên kết “p30f4012.gld” cho trường hợp dùng hợp ngữ. Đối với trường hợp dùng C30, chúng ta cũng tạo một project theo cách tương tự, chỉ khác là chúng ta sẽ chọn “Microchip C30 Toolsuite” ở bước 2, và tập tin nguồn được sử dụng là “Vidu5-1.c”.

Trong ví dụ này, một giá trị điện áp từ 0V đến VDD được đặt vào ngõ vào AN0/RB0 (sử dụng một biến trở với điểm giữa nối vào AN0/RB0), chương trình sẽ đọc giá trị điện áp thông qua module ADC và điều chỉnh độ rộng của xung tại các ngõ ra PWM1L/RE0, PWM2L/RE2, và PWM3L/RE4 một cách tương ứng. Nếu bạn có dao động ký thì bạn có thể theo dõi sự thay đổi độ rộng của xung (tần số xung được giữ cố định ở 40 kHz), nếu không có dao động ký thì bạn có thể dùng một VOM để đo giá trị trung bình của dạng

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	5/6

sóng điện áp ra tại PWM1L/RE0, PWM2L/RE2, và PWM3L/RE4. Chương trình khá trực quan và đã được chú thích đầy đủ, do đó không cần giải thích thêm ở đây.

Mạch nguyên lý chung cho ví dụ này và ví dụ 5-2 được thể hiện trong hình 3.1.



Hình 3.1: Mạch nguyên lý của phần cứng dùng cho các ví dụ 5-1 và 5-2

Những điểm cần chú ý trong ví dụ này là module MCPWM được dùng để kích hoạt việc chuyển đổi A/D, với module ADC làm việc ở chế độ lấy mẫu tự động. Hơn nữa, module MCPWM chỉ sử dụng các chân PWM1L, PWM2L, và PWM3L một cách độc lập.

3.3. Ví dụ 5-2

Thực hiện các thao tác tương tự như trong tutorial 2, bạn hãy loại bỏ tập tin “Vidu5-1.s” khỏi project, và thêm tập tin “Vidu5-2.s” vào project (với project viết bằng C30 thì bạn sẽ thay “Vidu5-1.c” bằng “Vidu5-2.c”).

Ví dụ này cũng thực hiện một việc khá đơn giản là đọc giá trị điện áp tại ngõ vào AN0/RB0 sau mỗi giây, sau đó gửi kết quả đã đọc ra cổng RS-232 (ở dạng số hex). Mã nguồn chương trình đã chú thích khá chi tiết, do đó không cần giải thích thêm ở đây.

Điểm cần chú ý ở ví dụ này là module ADC sẽ dùng bộ định thời dành riêng để tự chấm dứt lấy mẫu và kích hoạt chuyển đổi A/D. Ngoài ra, module ADC làm việc ở chế độ tự động lấy mẫu (lấy mẫu ngay sau khi module được kích hoạt).

4. Tóm tắt

Qua tutorial này, các module ADC và MCPWM của dsPIC đã được giới thiệu. Những thông tin khái quát về các module ADC và MCPWM của dsPIC đã được đề cập. Một trình tự khởi tạo cho các module ADC và MCPWM cũng đã được đề nghị. Thông

Người báo cáo:	Nguyễn Quang Nam	Tài liệu:	TUT06.01
Ngày:	10/3/2006	Trang:	6/6

qua 2 ví dụ, một số chế độ làm việc của module ADC cũng như module MCPWM đã được minh họa. Hy vọng thông qua tutorial này, bạn đã tiến thêm một bước trong việc làm chủ dsPIC.

Chúc bạn thành công!