

Mikrokontrolery z rdzeniem ARM, część 15

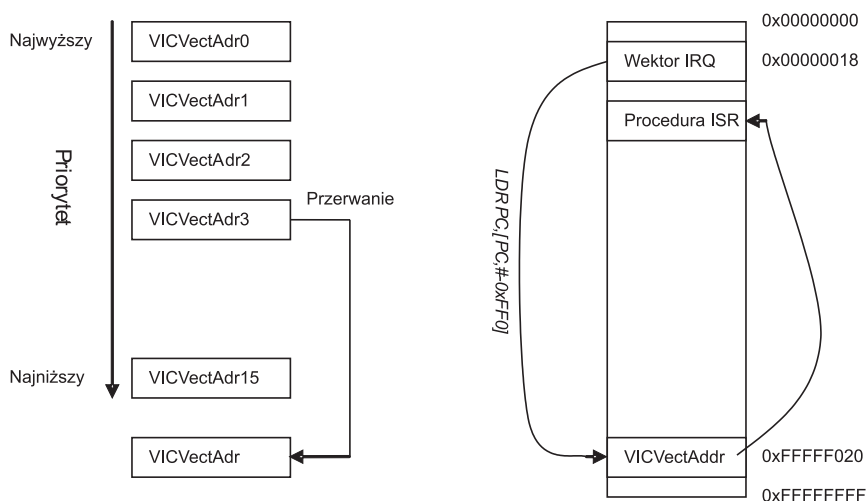
System przerwań c.d.

Bieżący stan układu peryferyjnego mikrokontrolera można określić poprzez odczytanie odpowiedniego rejestru SFR (tą metodą posługiwaliśmy się w poprzednio opisywanych przez nas przykładach). Taka metoda jest dobra tylko wtedy, gdy nie zależy nam na szybkiej reakcji na zdarzenie. W przypadku, gdy wymagana jest szybka odpowiedź, mikrokontroler musiałby cały czas cyklicznie badać stan rejestru SFR w celu wykrycia zdarzenia i nie mógłby w tym czasie robić nic innego.

Z tych właśnie względów wszystkie mikrokontrolery są wyposażone w mechanizm przerwań.

Wektoryzowane przerwania IRQ

Wcześniej pisaliśmy, że do linii FIQ generalnie powinno być podłączone tylko jedno przerwanie, stąd nasuwa się prosty wniosek, że pozostałe kanały muszą być podłączone do linii IRQ. Jak wiadomo programowe określenie źródła przerwania zajmuje pewien czas. Dlatego w kontrolerze VIC wprowadzono efektywny mechanizm przerwań wektorowych. Działanie jego polega na tym, że w momencie wystąpienia przerwania nie musimy sprawdzać znaczników przerwań w sposób programowy, ponieważ każde przerwanie ma swoją własną procedurę obsługi, wywoływaną w momencie jego zgłoszenia. Kontroler VIC posiada 16 slotów (wektorów), do których może być przypisany dowolny kanał przerwania. Każdy slot (0...15) posiada własny rejestr, do którego wpisywany jest adres procedury obsługi przerwania **VICVectAdr0...VICVectAdr15** oraz rejestr kon-



Rys. 32. Zasada działania przerwań wektorowych

trolny **VICVectCntl0...VICVectCntl15** umożliwiający przypisanie wybranego kanału do slotu. Slot 0 posiada najwyższy priorytet, natomiast slot o numerze 15 charakteryzuje się najniższym priorytetem. W tab. 20 przedstawiono znaczenie poszczególnych bitów dowolnego rejestru kontrolnego slotu **VICVectCntl0...15**.

Jak więc widzimy, do wybranego slotu możemy przypisać dowolny kanał (0...31). W dobrym zwyczaju jest, aby nie przypisywać tego samego kanału więcej niż do jednego slotu. Jeżeli jednak taka sytuacja będzie miała miejsce, wybrany zostanie slot o najniższym numerze. Do rejestru **VICVectAdr0...15** musimy wpisać adres procedury obsługi przerwania, która zostanie wywołana w momencie wystąpienia przerwania z danego kanału. Obsługa wektoryzowanego przerwania IRQ odbywa się następująco: Kontroler VIC określa numer kanału, od którego pochodzi przerwanie i do rejestru **VICVectAddr** (0xFFFFF030),

wpisuje adres procedury obsługi aktywnego przerwania o najwyższym priorytecie, co fizycznie odbywa się poprzez skopiowanie odpowiedniego rejestru **VICVectAdr0...VICVectAdr15**, natomiast rdzeń ARM7TD-MI-S wykonuje skok pod adres wektora wyjątku IRQ (0x00000018), gdzie znajduje się instrukcja: **LDR PC, [PC, #-0xFF0]**. Jej wykonanie powoduje odczytanie i skok pod adres wektora procedury obsługi przerwania umieszczonego w rejestrze **VICVectAddr**. Instrukcja ta wykonuje się w jednym cyklu maszynowym. Zasadę działania przerwań wektorowych zilustrowano na rys. 32.

W efekcie tych czynności każda linia (kanał) obsługi przerwania posiada swoją własną procedurę obsługi, wywoływaną w momencie wystąpienia przerwania, przez co odpada konieczność programowego sprawdzania źródła przerwania. Tak samo jak w przypadku przerwania FIQ, aby przerwanie było zgłoszone, oprócz przypisania i włączenia

Tab. 20. Znaczenie bitów dowolnego rejestru kontrolnego slotu **VICVectCntl0...15**

Bit	Nazwa	Opis	Wartość pocz.
[4:0]	INT_REQ	Numer kanału, który będzie przypisany do tego slotu IRQ	0
[5]	IRQ_EN	0 – wybrany slot IRQ jest nieaktywny 1 – wybrany slot IRQ jest aktywny	0
[31:6]	–	Zarezerwowane	–

określonego slotu musi być ono odblokowane w rejestrze *VICIntEnable*. Zakończenie obsługi przerwania powinno się odbyć poprzez wyzerowanie flagi zgłoszenia przerwania w wybranym urządzeniu peryferyjnym, a ponadto wpisaniu do rejestru *VICVectAddr* dowolnej wartości, co jest sygnałem dla kontrolera przerwania VIC, że procedura obsługi przerwania dobiegła końca.

Przerwania niewektoryzowane

W przypadku, gdy istnieje konieczność użycia większej liczby przerwania (IRQ=16+FIQ=1 > 17) pozostałe przerwania mogą być zgłoszone jako przerwania niewektoryzowane. Są one obsługiwane przez jedną wspólną procedurę obsługi przerwania, której adres powinien być umieszczony w rejestrze *VICDefVectAddr* (0xFFFF034). W momencie, gdy zostanie odblokowany kanał przerwania i nie jest on przypisany do żadnego slotu, wówczas do rejestru *VICVectAddr* zostanie załadowana zawartość rejestru *VICDefVectAddr*, w wyniku czego procesor skoczy do procedury obsługi przerwania niewektoryzowanego. Procedura na początku musi sprawdzić zawartość rejestru *VICIRQStatus* (0xFFFF000), w celu określenia, który kanał zgłosił przerwanie. Jeżeli bit o danym numerze jest ustawiony, oznacza to, że źródłem przerwania był kanał o tym numerze. Przerwanie niewektoryzowane są najwolniejszym rodzajem przerwania, ponieważ procedura ich obsługi musi określić na podstawie zawartości rejestru statusu, źródło przerwania. Jest to jednak jedyny sposób na obsługę większej liczby przerwania. W praktyce bardzo rzadko będziemy potrzebowali więcej niż 17 przerwania, dlatego ten sposób obsługi nie będzie zbyt często wykorzystywany. Zakończenie obsługi przerwania niewektoryzowanego odbywa się w taki sam sposób jak przerwania wektoryzowanego, czyli oprócz wyzerowania flagi zgłaszającej przerwanie w urządzeniu peryferyjnym musimy zapisać jakąś wartość do rejestru *VICVectAddr*.

Blokowanie oraz odblokowywanie przerwania w jednostce centralnej

Przerwania IRQ oraz FIQ są zgłaszane tylko wtedy, jeżeli flagi I oraz F w słowie stanu procesora (CPSR) są wyzerowane, natomiast w przeciwnym przypadku przerwania są zablokowane. Podczas zmiany zawartości reje-

List. 4. Przykładowe procedury umożliwiające włączanie oraz wyłączenie przerwania FIQ oraz IRQ

```
#include "armint.h"

#define IRQ_MASK 0x00000080
#define FIQ_MASK 0x00000040
#define INT_MASK (IRQ_MASK|FIQ_MASK)

//Odczytuje słowo stanu procesora
static inline cpu_t get_cpsr(void)
{
    cpu_t val;
    asm volatile ("mrs %[val], cpsr\n":[val]="r"(val):);
    return val;
}

//Zapisuje słowo stanu procesora
static inline void set_cpsr(cpu_t val)
{
    asm volatile ("msr cpsr, %[val]\n" ::[val]"r"(val) );
}

//Wyłącza przerwania IRQ
cpu_t disable_irq(void)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr(cpsr | IRQ_MASK);
    return cpsr;
}

//Włącza przerwania IRQ
cpu_t enable_irq(void)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr(cpsr & ~IRQ_MASK);
    return cpsr;
}

//Odtwarza stan przerwania IRQ
cpu_t restore_irq(cpu_t old_cpsr)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr((cpsr & ~IRQ_MASK) | (old_cpsr & IRQ_MASK) );
    return cpsr;
}

//Wyłącza przerwania FIQ
cpu_t disable_fiq(void)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr(cpsr | FIQ_MASK);
    return cpsr;
}

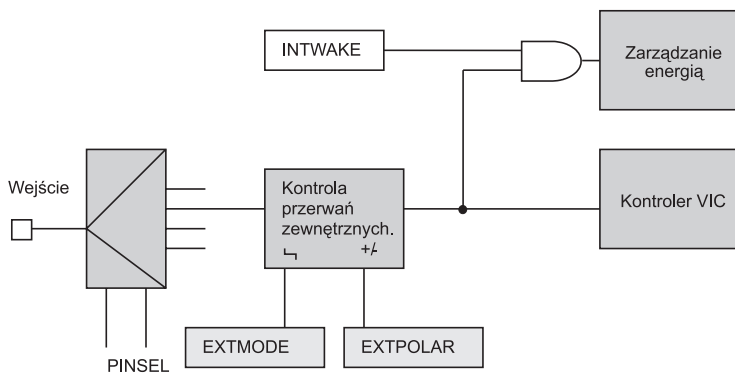
//Włącza przerwania FIQ
cpu_t enable_fiq(void)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr(cpsr & ~FIQ_MASK);
    return cpsr;
}

//Odtwarza stan przerwania FIQ
cpu_t restore_fiq(cpu_t old_cpsr)
{
    cpu_t cpsr;
    cpsr = get_cpsr();
    set_cpsr((cpsr & ~FIQ_MASK) | (old_cpsr & FIQ_MASK) );
    return cpsr;
}
```

strów kontrolera przerwania VIC dobrą praktyką jest chwilowe zablokowanie przerwania, dlatego napiszemy bardzo proste procedury umożliwiające włączanie oraz wyłączenie przerwania FIQ oraz IRQ, które przedstawiono na list. 4.

Funkcje *get_cpsr* oraz *set_cpsr* zadeklarowane jako *inline*, zawierają bardzo krótką wstawkę assemblerową, która zapisuje lub odczytuje zawartość rejestru słowa stanu procesora CPSR. Funkcje *disable_irq*, *disable_fiq* wyłą-

czają przerwania IRQ oraz FIQ. Ich działanie polega na odczytaniu słowa stanu procesora (CPSR), ustawieniu odpowiednio bitu I lub F oraz ponownym zapisaniu rejestru CPSR. Funkcje *enable_irq* oraz *enable_fiq* dokonują natomiast poprzez wyzerowanie bitu I lub F – odblokowania zgłaszania przerwania IRQ lub FIQ. Wszystkie te funkcje zwracają stan rejestru CPSR przed modyfikacją, umożliwiając jego ponowne otwarcie za pomocą funkcji *restore_irq* oraz *restore_fiq*, których



Rys. 33. Budowa systemu przerw zewnętrznych w mikrokontrolerze LPC213x

działanie polega na odczytaniu rejestru CPSR, zamaskowaniu flagi przerwania oraz ponownym jej ustawieniu zgodnie ze stanem zapamiętanym poprzednio. Nazwa zmiennej `cpu_t` jest po prostu synonimem zmiennej `unsigned int`, która jest równa długości słowa stanu maszyny. Należy tutaj podkreślić, że funkcje operujące na flagach przerw działają tylko w trybach uprzywilejowanych. Aby skorzystać z zestawu tych funkcji procesor wykonując program powinien pracować w trybie *System*. Jeśli chcemy, aby procesor działał w najbardziej bezpiecznym trybie *User*; zestaw funkcji obsługujących blokowanie i odblokowywanie przerw powinien być napisany z wykorzystaniem interfejsu przerw programowych (SWI).

Przerwania zewnętrzne EINT0...EINT3

Mikrokontrolery LPC213x/214x posiadają 4 zewnętrzne wejścia przerywające EINT0...EINT3, przy czym są one wprowadzone jako funkcje alternatywne kilku portów. Na przykład przerwianie zewnętrzne EINT3 dla mikrokontrolera LPC2138 jest dostępne na linii P0.9, P0.20 oraz P0.30. Akty-

wacji wybranej linii dokonujemy poprzez włączenie odpowiedniej funkcji alternatywnej portu, modyfikując rejestry PINSEL0 oraz PINSEL1, co było omówione w poprzedniej części kursu. Przerwania zewnętrzne mogą być zgłaszane zboczem narastającym lub opadającym oraz poziomem niskim lub wysokim. W przypadku, gdy dane przerwianie jest wybrane jako funkcja alternatywna przez więcej niż jedną linię portu oraz jest ono skonfigurowane jako zgłaszane poziomem, wówczas wykonywana jest operacja sumy logicznej, w efekcie, czego odpowiedni poziom na dowolnym z wejść przerywających powoduje zgłoszenie danego przerwania. Nie jest natomiast dopuszczalne ustawienie więcej niż jednego wejścia przerywającego, gdy wybrana linia jest skonfigurowana jako wyzwalana zboczem. Istnieje również możliwość ustawienia mikrokontrolera tak, aby stan aktywny na odpowiednim wejściu przerywającym powodował wyjście mikrokontrolera ze stanu uśpienia.

Budowę systemu przerw zewnętrznych przedstawiono na rys. 33.

Za konfigurację systemu przerw zewnętrznych odpowiedzialne

są cztery rejestry, przy czym każdy bit w każdym z tych rejestrów odpowiada za odpowiednie przerwianie (bit 0 – odpowiada za przerwianie EINT1, bit 1 – odpowiada za przerwianie EINT2 itd.)

EXTMODE (0xE01FC148) – umożliwia określenie sposobu zgłaszania przerwania. Jeżeli bit odpowiedzialny za wybrane przerwianie przyjmie wartość 0, wówczas jest ono zgłaszane poziomem, natomiast gdy przyjmie wartość 1, przerwianie aktywowane jest zboczem.

EXTPOLAR (0xE01FC14C) – umożliwia określenie polaryzacji zbocza lub poziomu aktywującego przerwianie. Skasowanie wybranego bitu spowoduje, że przerwianie będzie aktywowane zboczem opadającym lub stanem niskim, natomiast jego ustawienie spowoduje, że przerwianie będzie aktywowane zboczem narastającym lub stanem wysokim.

INTWAKE (0xE01FC144) – umożliwia aktywowanie budzenia mikrokontrolera przerwaniem zewnętrznym ze stanu uśpienia.

EXTINT (0xE01FC140) – zawiera flagi zgłoszeń przerw zewnętrznych.

Ustawienie danego bitu oznacza, że odpowiadające mu przerwianie zostało aktywowane. Flagę tę musimy skasować przed wyjściem z procedury obsługi przerwania poprzez zapis 1 na odpowiadającym bicie. W przypadku, gdy przerwianie było ustawione jako wyzwalane poziomem a pin wejściowy znajduje się w stanie aktywnym, wówczas skasowanie bitu zgłoszenia odpowiadającego przerwania nie będzie możliwe.

Lucjan Bryndza, EP
lucjan.bryndza@ep.com.pl



evatronix

Firma z branży elektronicznej poszukuje pracownika obsługi techniczno-handlowej do swojego biura w Gliwicach

Wymagania:

- Wykształcenie wyższe elektroniczne lub pokrewne
- Znajomość podstaw techniki cyfrowej
- Znajomość kontrolerów 8051, 68000
- Biegła znajomość języka angielskiego w mowie i w piśmie (poziom FCE lub wyższy, przewidziany test kwalifikacyjny)
- Mile widziana znajomość języków opisu sprzętu VHDL lub Verilog
- Mile widziane doświadczenie w obsłudze klientów na rynku B2B

Zakres obowiązków:

- Dostarczanie pomocy technicznej Klientom, dystrybutorom i handlowcom firmy
- Bieżąca koordynacja obsługi i wsparcia dostarczanego przez inżynierów
- Współpraca z kierownikami projektów

Zainteresowanych prosimy o składanie CV wraz z listem motywacyjnym w oddziale firmy w Gliwicach, ul. Dubois 16, 44-100 Gliwice, tel. 032 231 11 71, lub przesyłanie ich pocztą elektroniczną na adres praca@evatronix.pl. Termin zgłaszania podań mija 16 lutego.

Do przesłanej oferty prosimy dołączyć oświadczenie: „Wyrażam zgodę na przetwarzanie moich danych osobowych dla potrzeb niezbędnych dla realizacji procesu rekrutacji (zgodnie z ustawą z dn. 29.08.97 roku o ochronie danych osobowych Dz.U Nr 133 poz. 883)“.

Jednocześnie zastrzegamy sobie prawo kontaktu wyłącznie z wybranymi kandydatami.

PRACA • PRACA • PRACA



Renomowany producent drukarek INK-JET oferuje wysokiej klasy

Aktywny detektor podczerwieni do zastosowań w układach automatyki i zabezpieczeń

małe wymiary budowy (M18x1)

duża odporność na zakłócenia

wbudowany wskaźnik zadziałania

wyjście odporne na zwarcie

wykonania PNP, NPN



EBS Ink- Jet Systems Poland Sp. z o.o.

ul. Tarnogajska 13, 50-512 Wrocław
tel. (071) 367 04 11, fax (071) 373 32 69