

LABORATORIUM nr 7

Temat: ARM7 - obsługa przetwornika A/C

1. PRZETWORNIK A/C W ZESTAWIE URUCHOMIENIOWYM MCB2300

Mikrokontroler LPC2378 jest w stanie przetworzyć sygnał w postaci cyfrowej. W zestawie uruchomieniowym MCB2300 znajduje się jedno źródło sygnału analogowego, w postaci potencjometru, oraz jeden odbiornik sygnału analogowego, w postaci głośnika.

Chcąc sterować za pomocą potencjometru dowolnym innym elementem zestawu uruchomieniowego, należy przetworzyć sygnał analogowy (ciągły) z potencjometru na odpowiadającą jemu wersję cyfrową (dyskretną). Służy do tego **przetwornik analogowo/cyfrowy**.

Zastosowany w zestawie uruchomieniowym przetwornik jest urządzeniem 10 bitowym. Obsługiwany jest przez piny **AD0[7:0]**, które stanowią wejścia analogowe konwertera, na których mierzone jest napięcie wejściowe. Na pin **VREF** podawane jest napięcie referencyjne, wynoszące 3,3V. Piny **V_{DDA}** i **V_{SSA}** odpowiadają faktycznemu napięciu zasilania przetwornika, oraz jego uziemieniu.

Konwerter A/C zawiera rejestry pokazane w tabeli 1.

Tabela. 1: Rejestry przetwornika A/C

Nazwa	Opis	Dostęp	Wartość po restarcie
AD0CR	Rejestr kontrolny przetwornika. Przed rozpoczęciem przetwarzania, określamy w nim tryb pracy.	R/W	0x0000 0001
AD0GDR	Ogólny rejestr danych przetwornika. Przechowuje wynik konwersji.	R/W	Brak
AD0STAT	Rejestr stanu przetwornika. Zawiera flagi DONE oraz OVERRUN dla wszystkich kanałów przetwornika, oraz flagi przerwań.	RO	0
AD0INTEN	Rejestr włączenia przerwań. Zawiera bity włączenia przerwania, co pozwala fładze DONE dla każdego kanału A/C być wystawioną bądź nie w przypadku pojawienia się przerwania.	R/W	0x0000 0100
AD0DR[0:7]	Rejestr kanału [0:7] przetwornika. Zawiera wynik konwersji A/C dla kanału [0:7].	R/W	brak

Rejestr kontrolny AD0CR zawiera ustawienia bitów wybierających kanał A/C do konwersji, ustawienia taktowania, tryb pracy i przełącznik start/stop. Funkcje poszczególnych bitów rejestru omówiono w tabeli 2.

Rejestr AD0GDR zawiera rezultat konwersji A/C. Zawiera dane, oraz znaczniki DONE, Overrun oraz numer kanału przetwornika, z którego pochodzą przekonwertowane dane. Funkcje obsługiwane przez ten rejestr zostały omówione w tabeli 3.

Rejestr stanu AD0STAT pozwala na kontrolę stanu wszystkich kanałów przetwornika jednocześnie. Flagi DONE i OVERRUN są odzwierciedleniem wartości z rejestru ADDRn. Bity [7:0] zawierają flagi DONE dla poszczególnych kanałów, bity [15:8] zawierają flagi OVERRUN, bit 16 zawiera flagę przerwania ADINT, a bity [31:17] są nieużywane.

Rejestr AD0INTEN służy do kontroli, który z kanałów przetwornika zakończył konwersję, i w związku z tym faktem generuje przerwanie. Bity [7:0] ADINTEN pozwalają kontrolować, który z kanałów wygenerował przerwanie po zakończeniu konwersji. Bit 8 ADGINTEN, kiedy zawiera 1, ustawie ogólną flagę DONE w ADDR aby wygenerować przerwanie. Jeżeli zawiera 0, informację o wygenerowanym przerwaniu zawiera ADINTEN. Bity [31:9] zawsze zawierają 0 i są nieużywane.

Tabela. 2: Funkcje rejestru AD0CR

Bit	Symbol	Opis	Wartość po restarcie
7:0	SEL	Wskazuje, które z pinów AD0.7:0 mają być próbkowane oraz konwertowane. Bit 0 wskazuje pin AD0.0.	0x01
15:8	CLKDIV	Zegar magistrali APB (PCLK) dzielony jest przez tę wartość zwiększoną o 1 i określa taktowanie przetwornika A/C, które powinno być mniejsze bądź równe 4,5 MHz.	0
16	BURST	W przypadku gdy jest ustawiony na 0 konwersja jest kontrolowana programowo i wymaga 11 taktów. W przypadku gdy jest ustawiony na 1 konwerter A/C dokonuje konwersji z częstotliwością wynikającą z ustawienia pola CLKS. Konwersja jest przerywana poprzez wyczyszczenie tego bitu, ale zadanie konwersji jest kończone. Ważnym jest, aby bity START były ustawione na 000, dopiero wtedy rozpocznie się konwersja dla BURST =1.	0
19:17	CLKS	Ustawia ilość taktów zegara stosowany do konwersji w trybie BURST, oraz ilość bitów dokładności wyniku konwersji. Przyjmuje następujące wartości: 000 – 11 taktów/10 bitów; 001 – 10 taktów/9 bitów; 010 – 9 taktów/8 bitów; 011 – 8 taktów/7 bitów; 100 – 7 taktów/6 bitów; 101 – 6 taktów/5 bitów; 110 – 5 taktów/4 bity; 111 – 4 takty/3 bity.	0
20		Zarezerwowany	Brak
21	PDN	Dla 1 konwerter pracuje, dla 0 jest wyłączony (power down).	0
23:22		Zarezerwowany	Brak
26:24	START	Kiedy flaga BURST ustawiona jest na 0, bity te kontrolują kiedy ma się rozpocząć konwersja A/C: 000 – zatrzymanie konwersji (gdy PDN ustawiamy na 0); 001 – rozpoczyna konwersję; 010 – rozpoczyna konwersję gdy na pinie P2.0/EINT0 wybranym poprzez EDGE pojawi się zbocze; 011 – rozpoczyna konwersję gdy na pinie P1.27/CAP0.1 wybranym poprzez EDGE pojawi się zbocze; 100 – rozpoczyna konwersję gdy na pinie MAT0.1 wybranym poprzez EDGE pojawi się zbocze; 101 – rozpoczyna konwersję gdy na pinie MAT0.3 wybranym poprzez EDGE pojawi się zbocze; 110 – rozpoczyna konwersję gdy na pinie MAT1.0 wybranym poprzez EDGE pojawi się zbocze; 111 – rozpoczyna konwersję gdy na pinie MAT1.1 wybranym poprzez EDGE pojawi się zbocze;	0
27	EDGE	Ma znaczenie tylko w sytuacji, gdy w START ustawiono wartość inną niż 000. Dla 1 rozpoczyna konwersję przy opadającym zboczu, dla 0 rozpoczyna konwersję przy narastającym zboczu.	0
31:28		Zarezerwowany	Brak

Tabela. 3: Funkcje rejestru AD0GDR

Bit	Symbol	Opis	Wartość po restarcie
5:0	Nie używany	Zawsze zawierają 0	0
15:6	V/V _{REF}	Kiedy flaga DONE ustawiona jest na 1, zawiera bitową reprezentację napięcia na pinie Ain określonym polem SEL, podzielony przez napięcie V _{DDA} . Zero oznacza że napięcie na Ain jest mniejsze niż, równe lub bliskie do V _{SSA} , zaś dla 0x3FF napięcie na Ain jest większe niż, równe lub bliskie do V _{REF} .	X
23:16	Nie używany	Zawsze zawierają 0	0
26:24	CHN	Numer kanału z którego przekonwertowane zostały bity LS.	X
29-27	Nie używany	Zawsze zawierają 0.	0
30	OVERRUN	Przy trybie burst zawiera 1, jeśli jedna lub więcej konwersji zostały utracone i nadpisane przed konwersją.	0
31	DONE	Po zakończeniu konwersji zawiera 1. Jest ustawiany na zero po operacji odczytu tego rejestru i kiedy zapisywany jest ADCR.	0

Rejestry danych A/C (AD0DR0 do AD0DR7) zawierają wynik konwersji A/C w postaci wartości napięcia. Funkcje rejestru omówiono w tabeli 4.

Tabela. 4: Funkcje rejestru AD0DR[0-7]

Bit	Symbol	Opis	Wartość po restarcie
5:0	Nie używany	Zawsze zawierają 0	0
15:6	V/V _{REF}	Kiedy flaga DONE ustawiona jest na 1, zawiera bitową reprezentację napięcia na pinie Ain określonym polem SEL, podzielony przez napięcie V _{DDA} . Zero oznacza że napięcie na Ain jest mniejsze niż, równe lub bliskie do V _{SSA} , zaś dla 0x3FF napięcie na Ain jest większe niż, równe lub bliskie do V _{REF} .	X
29:16	Nie używany	Zawsze zawierają 0	0
30	OVERRUN	Przy trybie burst zawiera 1, jeśli jedna lub więcej konwersji zostały utracone i nadpisane przed konwersją.	0
31	DONE	Po zakończeniu konwersji zawiera 1. Jest ustawiany na zero po operacji odczytu tego rejestru.	0

2. OBSŁUGA PRZERWAŃ POPRZECZ MIKROKONTROLER LPC2378

Przerwanie jest zgłaszane do wektorowego kontrolera przerwań VIC kiedy bit ADINT w rejestrze ADSTAT ustawiony jest na 1. Procesor ARM obsługuje dwa wejścia przerwań: standardowe **IRQ** i szybkie **FIQ**. Kontroler przerwań VIC obsługuje 32 wejścia przerwań, i na poziomie oprogramowania przypisuje je dynamicznie jako IRQ lub FIQ. Przerwania FIQ mają wyższy priorytet. W sytuacji idealnej, pojawia się jedno przerwanie FIQ, które jest obsługiwane jako pierwsze. W przypadku pojawienia się kilku przerwań FIQ, wykonywana jest operacja OR i w ten sposób wyłaniane jest przerwanie, które będzie przetwarzane w pierwszej kolejności. W przypadku pojawienia się kilku przerwań standardowych IRQ, w pierwszej kolejności obsługiwane jest to, które jest podłączone do kanału o niższym numerze. Źródła przerwań mają przypisane kanały w sposób pokazany w tabeli 5.

Następujące rejestry przechowują informacje zgodnie ze wskazaną tabelą (bit ustawiony na 1 wskazuje źródło przerwania):

- VICSoftInt – używany do generowania przerwań programowych;
- VICSoftIntClear – rejestr tylko do odczytu. Pozwala oprogramowaniu na ustawienie na 0 jednego lub kilku bitów w VICSoftInt;
- VICRawIntr – rejestr tylko do odczytu, zawiera informację z 32 sprzętowych źródeł przerwań lub programowych;

- VICIntEnable – kontroluje które z 32 sprzętowych lub programowych przerwań jest wystawione;
- VICIntEnClear – rejestr tylko do zapisu, pozwala oprogramowaniu ustawić na 0 bit wystawienia przerwania w VICIntEnable;
- VICIntSelect – klasyfikuje każde z 32 przerwań jako IRQ (bit na 0) lub FIQ (bit na 1);
- VICIRQStatus – rejestr tylko do odczytu, zawiera informację o stanie przerwania, które zaklasyfikowane zostało jako IRQ (ustawienie na 1);
- VICFIQStatus – rejestr tylko do odczytu, zawiera informację o stanie przerwania, które zaklasyfikowane zostało jako FIQ (ustawienie na 1);

Tabela. 5: Źródła przerwań obsługiwane przez mikrokontroler LPC2378

Bit	31	30	29	28	27	26	25	24
Symbol	I2S	I2C2	UART3	UART2	TIMER3	TIMER2	GPDMA	SD/MMC
Bit	23	22	21	20	19	18	17	16
Symbol	CAN1&2	USB	Ethernet	BOD	I2C1	AD0	EINT3	EINT2
Bit	15	14	13	12	11	10	9	8
Symbol	EINT1	EINT0	RTC	PLL	SSP1	SPI/SSP0	I2C0	PWM0
BIT	7	6	5	4	3	2	1	0
Symbol	UART1	UART0	TIMER1	TIMER0	ARMCore1	ARMCore0	-	WDT

Pozostałe rejestry stosowane do obsługi przerwań to:

- VICVectAddr[0-31] – przechowuje adresy 32 procedur obsługi przerwań (ISR – Interrupt Service Routines) dla przerwań IRQ;
- VICVectPriority[0-31] – przechowują priorytety obsługi każdego z przerwań IRQ (niezależnie). Priorytety przyjmują wartości od 0 (najwyższy) do 15 (najniższy). Po restarcie wszystkie przerwania otrzymują priorytet najniższy. Wykorzystywane są tylko bity [3-0];
- VICAddress – zawiera adres procedury obsługi przerwania IRQ, która ma być aktualnie wykonana. Adres jest dostarczany z jednego z rejestrów VICVectAddr;
- VICSWPriorityMask – zawiera maskę nakładaną na 16 priorytetów przerwań. Maskę opisana jest na bitach [15:0]. Bit ustawiony na 0 oznacza, że priorytet przerwania jest maskowany, zaś na 1 że priorytet przerwania nie jest maskowany;
- VICProtection – ten jednobitowy rejestr (bit 0) kontroluje dostęp programów uruchomionych w trybie użytkownika do rejestrów VIC. Jeżeli zawiera 0 rejestry VIC są dostępne w trybie użytkownika oraz uprzywilejowanym, zaś jeżeli zawiera 1, rejestry VIC są dostępne tylko w trybie uprzywilejowanym.

3. OBSŁUGA UKŁADÓW TIMER[0-3]

Układ czasomierza/licznika służy do zliczania taktów zegara sprzętowego (PCLK), lub zegara zewnętrznego. Mikrokontroler LPC2378 posiada cztery takie układy. Do rozwiązania postawionego w niniejszym laboratorium zadania zastosujemy układ czasomierza o numerze 0 (**Timer0**). Każdy z układów czasomierza/licznika sterowany jest za pomocą rejestrów, które dokładnie zostały omówione w [1], tabela 472 i dalsze. Z punktu widzenia realizacji postawionych zadań ważne są następujące rejestry:

- TCR [T0TCR-T3TCR] (Timer Control Register) – kontroluje funkcje układu czasomierza, 8 bitowy. Licznik układu może być wyłączony lub wyzerowany poprzez ten rejestr. Jeśli bit 0 ma wartość 1, licznik jest włączony. Jeśli bit 1 jest ustawiony na 1, licznik jest zerowany (reset). Bity [7:2] są zarezerwowane.
- MR0 [T0MR0-T3MR0] (Match Register 0) – rejestr zgodności o numerze 0 (łącznie istnieją cztery takie rejestry), uruchamiany i obsługiwany jest z poziomu rejestru MCR i stosowany do generowania przerwania, zerowania i zatrzymania układu czasomierza/licznika. Wartość tego rejestru jest cały czas porównywana z wartością TC (Timer Counter);
- MCR [T0MCR-T3MCR] (Match Control Register) – 16 bitowy. Kontroluje pojawienie się przerwania dla układu czasomierza/licznika oraz wykonanie zerowania licznika. Każdy z układów kontrolowany jest przez 3 bity. Dla układu Timer0 są to odpowiednio: **bit 0** [MR0I] – jeśli ustawiony na 1, generowane jest przerwanie w momencie gdy rejestr MR0 osiągnie wartość określoną w TC, gdy jest ustawiony na 0 obsługa przerwania jest wyłączona; **bit 1** [MR0R] – jeśli jest ustawiony na 1, w przypadku zgodności z TC następuje zerowanie (reset); **bit 2** [MR0S] – jeśli jest ustawiony na 1, TC zostanie zatrzymany, a

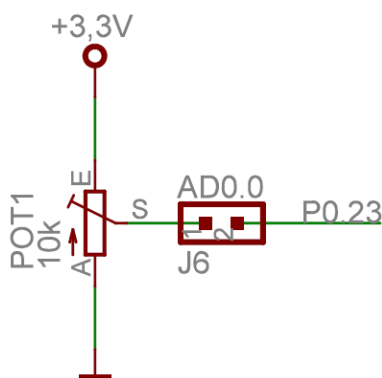
TCR[0] zostanie ustawiony na 0 w przypadku zgodności pomiędzy TC i MR0. W tym rejestrze bity [15:12] są nieużywane.

Układ czasomierza/licznika posłuży do konwertowania sygnału analogowego z potencjometru (napięcia) co określony interwał czasu. Omówionym rejestrom należy nadać odpowiednie wartości w celu zainicjowania układu czasomierza/licznika. Poniżej pokazano przykład takiej procedury. Procedura ta wymaga samodzielnego **uzupełnienia** niektórych wartości.

```
01  TOMR0      = 11999;          /* 1msec = 12000-1 przy 12.0 MHz */
02  TOMCR      = ?;              /* przerwanie i zerowanie w MR0 */
03  TOTCR      = ?;              /* Timer0 włączony */
04  VICVectAddr4 = (unsigned long)T0_IRQHandler; /* ustaw VIC */
05  VICVectCntl4 = 15;           /*ustawienie priorytetu*/
06  VICIntEnable = (1 << ?);    /*włączenie przerwania dla Timer0 */
```

Podobną procedurę należy określić dla przetwornika A/C. W procedurze takiej należy włączyć przetwornik (zasiłić go), wybrać odpowiednią funkcję dla pinu mikrokontrolera obsługującego AD0.0 (P0.23, rysunek 1), wybrać kanał konwersji (w naszym przypadku kanał 0), ustawić tryb pracy konwertera, oraz obsługę przerwań. Poniżej pokazano przykład takiej procedury. Procedura ta wymaga samodzielnego **uzupełnienia** niektórych wartości.

```
01  PCONP      = (1 << ?);      /* włączenie zasilania do bloku A/C */
02  PINSEL1     = ?;              /*wybranie na pinie funkcji AD0.0*/
03  AD0INTEN    = (1 << ?);      /* CH0 włącza przerwanie */
04  AD0CR       = 0x?;           /*zasilanie, PCLK/4, wybór AD0.0 */
05  VICVectAddr18 = (unsigned long)ADC_IRQHandler; /* ustawienie VIC */
06  VICVectCntl18 = 14;          /* ustawienie priorytetu przerwania */
07  VICIntEnable = (1 << ?);    /* włączenie przerwania dla ADC */
```



Rys. 1. Sposób podłączenia potencjometru w zestawie uruchomieniowym MCB2300

W kolejnym kroku należy zdefiniować procedury obsługi obu wskazanych w powyższych przykładach przerwań. Procedurę taką definiuje się w ten sposób, że przed jej nazwą musi pojawić się klauzula `__irq`, np. `__irq void T0_IRQHandler (void)`.

Procedura obsługi przerwania czasomierza/licznika powinna kontrolować zwiększanie zmiennej reprezentującej licznik. W przypadku, gdy licznik odliczy 1s (wartość 1000), licznik powinien zostać wyzerowany. Następnie powinno nastąpić opcjonalne przeskalowanie ostatnio skonwertowanej wartości (konwersja następuje co 1 ms). W ostatniej części procedury należy wykonać konwersję (AD0CR), wyzerować flagę przerwania (TOIR), oraz potwierdzić przerwanie (VICVectAddr).

Druga procedura obsługi przerwania dotyczącego konwertera A/C, powinna pobrać wynik konwersji (AD0DR0) i przypisać zmiennej, a następnie potwierdzić przerwanie (VICVectAddr).

W części głównej programu należy przeskalować wartość po konwersji i wyświetlić wymagane informacje na wyświetlaczu LCD.

4. ZADANIA

1. Napisz program, w którym za pomocą potencjometru zwiększał i zmniejszał będziesz wartość wyświetlaną na ekranie LCD. Dokonaj takiego skalowania odczytanych wartości, aby na wyświetlaczu LCD wyświetlić 0 dla lewego skrajnego położenia potencjometru, oraz 79 dla skrajnego prawego.
2. Napisz program, który dokona takiego skalowania odczytanych wartości, aby na wyświetlaczu LCD wyświetlić -20 dla lewego skrajnego położenia potencjometru, oraz 20 dla skrajnego prawego. Dla środkowego położenia powinna pojawić się na wyświetlaczu wartość 0. Dodatkowo, w drugiej linii wyświetlacza powinny pojawiać się znaki * dla wartości różnych od 0, oraz + dla 0, np. ****+**.

5. LITERATURA

- [1] NXP: Dokumentacja techniczna mikrokontrolerów z rodziny LPC23xx, <http://ics.nxp.com/support/documents/microcontrollers/pdf/user.manual.lpc23xx.pdf> (dostęp: kwiecień 2011).
- [2] NXP, Serwis poświęcony mikrokontrolerom z rodziny LPC23xx, zawierający przykłady wykorzystania i programowania, <http://ics.nxp.com/support/documents/microcontrollers/?scope=LPC2300> (dostęp: kwiecień 2011).
- [3] Keil, Dokumentacja zestawu uruchomieniowego MCB2300 wraz z przykładami, <http://www.keil.com/support/man/docs/mcb2300> (dostęp: kwiecień 2011).