

Mikroelektronika w Technice i Medycynie
Podstawy projektowania systemów wbudowanych
Instrukcja do ćwiczeń laboratoryjnych
Akwizycja sygnałów - pomiar "On Interrupts"
Mirosław Żołędź 2017

Spis treści

Wprowadzenie	2
Przygotowanie projektu	2
Wykorzystanie zdarzeń	3
Wykorzystanie timera	3
Wykorzystanie przerwań	4

Wprowadzenie

Obecną wersja programu oparta jest na poolingu i działa w sposób następujący:

- a. oczekiwanie na znak z aplikacji nadrzędnej
- b. programowe wyzwolenie pojedynczej konwersji A/C
- c. programowe oczekiwanie na zakończenie konwersji
- d. programowa transmisja wyniku konwersji w postaci łańcucha znakowego do aplikacji nadrzędnej

W następnych punktach część programowa będzie zastępowana rozwiązaniami sprzętowymi

Przygotowanie projektu

1. Do poprzedniego rozwiązania (Solution) dodać projekt o nazwie "ADC_OnInterrupts" (Patrz instrukcja Akwizycja sygnałów - pomiar On Interrupts punkt 4.)
2. Dodać moduły ASF jak w projekcie "ADC_OnDemand"
3. Podmienić zawartość pliku main na zawartość pliku main z projektu "ADC_OnDemand"
4. Skompilować i sprawdzić działanie programu. Do tego i następnych testów w tym rozdziale użyć aplikacji użytkownika "OnDemand".
5. Ustawić ADC na konwersję singleended, wzmocnienie 1x, 8-bit, bez znaku (do odczytu wyniku konwersji użyć odpowiedniej funkcji)
6. Wrócić do wysyłania surowego wyniku konwersji (usunąć wszystkie nieużywane fragmenty kodu)
7. Podłączyć moduł z potencjometrem (podaje napięcie 0-3.3V na wejście ADC1)
8. Sprawdzić działanie programu podczas obracania gałką potencjometru w całym zakresie. (Poprawne działanie modułu z potencjometrem można sprawdzić moduł przy użyciu programu referencyjnego "main_Irq_1_OnDemand.hex")
9. Zmierzyć i zanotować wartość napięcia, przy której nasycy się przetwornik i sprawdzić jak się ma do wartości napięcia referencyjnego.
10. Ustawić napięcie odniesienia przetwornika A/C na zewnętrzne podawane z portu A (podłączono 3.3V)
11. Ponownie zmierzyć i zanotować wartości napięcia, przy której nasycy się przetwornik.
12. Sprawdzić dlaczego nie zgadza się z wartością napięcia podłączonego (patrz "Atmel AVR1300: Using the Atmel AVR XMEGA ADC")
- 13. Zapisać plik main pod nazwą "main_Irq_1_OnDemand".**

Wykorzystanie zdarzeń

Dotychczasowy scenariusz : PC->UART->soft->ADC->soft->UART->PC

Nowy scenariusz : PC->UART->soft->EVENT->ADC->soft->UART->PC

Aby zrealizować powyższy scenariusz należy w programie mikrokontrolera:

- włączyć moduł Eventów
- ustawić źródło zdarzeń dla kanału zerowego modułu Eventów (na razie na żadne bo Event ma być wyzwalany programowo)
- ustawić sposób wyzwalania konwersji A/C na Eventy
- zamiast funkcji do wyzwalania konwersji A/C użyć programowego wyzwalania eventów wpisując do rejestru EVSYS_STROBE wartość 0x01.

Testy przeprowadzić z użyciem potencjometru.

Pomocne informacje można znaleźć w dokumencie "Getting starting" modułu ADC ASF (drugi przykład).

Uwaga: dwa ostatnie argumenty funkcji "adc_set_conversion_trigger" powinny wynosić odpowiednio "1" i "0" (wyzwalany jest tylko jeden z kanałów przetwornika i jest to kanał 0).

Zapisać plik main pod nazwą "main_Irq_2_Event".

Wykorzystanie timera

Celem modyfikacji jest rezygnacja z handshakingu.

TIMER->EVENT->ADC->soft->UART->PC

Aby zrealizować powyższy scenariusz należy:

- w programie mikrokontrolera
 - włączyć i skonfigurować Timer
 - ustawić źródło zdarzeń dla kanału zerowego modułu Eventów na Timer
 - usunąć oczekiwanie na odbiór znaku
- w aplikacji nadrzędnej usunąć wysyłanie znaku.

Testy przeprowadzić z użyciem potencjometru.

Pomocne informacje można znaleźć w dokumencie "Getting starting" modułu ADC ASF (drugi przykład).

Uwaga: rozdzielczość Timera = 500000, częstotliwość generowanych zdarzeń = 10 Hz.

Zapisać plik main pod nazwą "main_Irq_3_Timer".

Zapisać aplikację nadrzędną (nowy vi) pod nazwą "OnInerrupts"

Wykorzystanie przerw

Celem modyfikacji jest rezygnacja wykorzystanie przerwań do częściowego usunięcia poolingu
TIMER->EVENT->ADC->IRQ[soft->UART->PC]

1. Transmisja danych w postaci łańcucha znaków

Aby zrealizować powyższy scenariusz należy:

- skonfigurować ADC do pracy z przerwaniami (w tym podłączyć do przerwań funkcję, tzw. callback)
- przenieść wysyłanie wyniku konwersji do callback'a
- usunąć oczekiwanie na koniec konwersji (pusta pętla główna)

Pomocne informacje można znaleźć w dokumencie "Getting started" modułu ADC ASF (pierwszy przykład).

Testy przeprowadzić z użyciem potencjometru.

Zapisać plik main pod nazwą "main_Irq_4_Interrupts".

2. Transmisja danych w postaci bajtów

- a) Zmienić wysyłanie wyniku w postaci sformatowanego łańcucha na wysyłanie (młodszy) bajtu wyniku konwersji.

W tym celu:

- w programie mikrokontrolera do wysyłania znaku użyć funkcji modułu USART ASF (jest już dołączony w ramach modułu "Standard Serial I/O")
- w aplikacji nadrzędnej:
 - wyłączyć odbiór "na terminator"
 - ustawić liczbę odbieranych znaków na 1
 - zamienić funkcje do konwersji odpowiedzi na funkcję zamieniającą łańcuch na tablicę bajtów

Testy przeprowadzić z użyciem potencjometru.

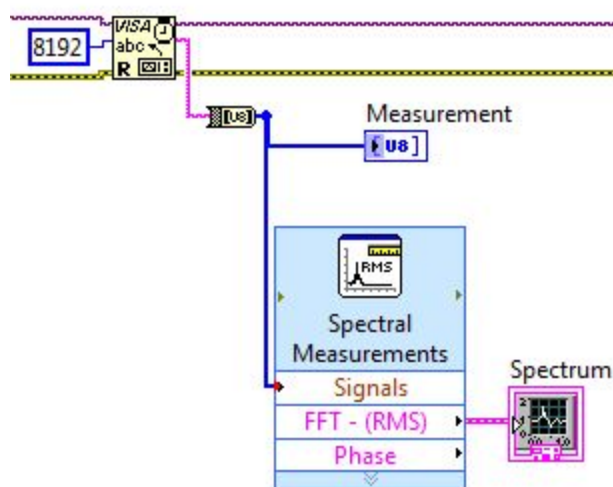
Zapisać plik main pod nazwą "main_Irq_5_Interrupts_Raw_Slow".

Zapisać aplikację nadrzędną (nowy vi) pod nazwą "OnInterrupts_Raw_Slow"

b) Zwiększyć częstotliwość akwizycji do 50 kHz

W tym celu zmienić:

- w programie mikrokontrolera ustawić:
 - baudrate UARTa na 921600 i aplikacji nadrzędnej na
 - częstotliwość wyzwalania konwersji na 50 kHz
- w aplikacji nadrzędnej
 - ustawić baudrate na 921600
 - ustawić liczbę bajtów odebranych podczas pojedynczego odczytu na 8192
 - zmienić kontrolkę “waveform chart” na “weveform graph”, wyłączyć automatyczne skalowanie osi y (ustawić na 0-255)
 - wstawić do pętli głównej kod realizujący FFT (Kontrolka spectrum to Waveform Graph) a następnie wyłączyć skalowanie osi X (0-40) i Y (0-0.05).



Testy przeprowadzić z użyciem

1. Podpiąć modułu mikrofonu (Poprawne działanie modułu można sprawdzić moduł przy użyciu programu referencyjnego “main_Irq_5_Interrupt_Raw_Fas.hex” oraz aplikacji nadrzędnej “3_IRQ_50kps.exe”)
2. Jako źródła dźwięku użyć telefonu komórkowego z prostym generatorem dźwięku (np. dmitsoft - tone generator)

Zatrzymać aplikację podczas rejestracji dźwięku a następnie sprawdzić kształt przebiegu a w szczególności ilość próbek przypadającą na jeden okres (policzyć czy się zgadza).

3. Jako źródła dźwięku użyć telefonu komórkowego z generatorem dźwięku w standardzie DTFM (patrz wikipedia).

Zapisać plik main pod nazwą “main_Irq_6_Interrupts_Raw_Fast”.

Zapisać aplikację nadrzędną (nowy vi) pod nazwą “Onlnerrupts_Raw_Fast”