ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1994
Seria: ELEKTRYKA z.134 Nr kol.1225

Jerzy JAKUBIEC

PROGRAM DYSTRYBUCJI ZADAŃ W MIKROPROCESOROWYM PRZYRZĄDZIE POMIAROWYM

Streszczenie. W artykule opisano sposób budowy oprogramowania przyrządu mikroprocesorowego bazujący na zastosowaniu programu zarządzającego aktywizacją poszczególnych programów cząstkowych, czyli tzw. dystrybutora zadań. Scharakteryzowano wersje dystrybutora przeznaczone do zastosowań w przyrządach budowlanych w oparciu o mikroprocesor Z80 i INTEL 8051.

A MULTITASKING PROGRAM IN AN INTELLIGENT INSTRUMENT

Summary. A method of constructing a multitasking program (called the tusk distributor) designed for an intelligent measurement instrument is described. Two distributors assigned for using in the instruments based on microprocessors Z80 and INTEL 8051 are characterized.

ДИСТРИБУТОР ПРОГРАММНЫХ ЗАДАЧ В ИЗМЕРИТБЛЬНОМ ПРИБОРЕ

Резюме. В статье описан способ конструкции программы называемой дистрибутором задач, которая позволяет координировать параллельную работу подпрограмм реализируюших частные задачи измэритэльного прибора использующего микропроцэссор. Охарактэризованы дистрибуторы построены для двух типов микропроцессоров: 380 и ИНТЕЛ 8051.

1. WSTĘP

Konstrukcje współczesnych mikroprocesorowych przyrządów pomiarowych cechuje tendencja do realizacji możliwie dużej liczby funkcji programowych. Ponadto pojawiają się opracowania pozwalające na programowe podwyższanie dokładności pomiarów w wyniku

realizacji algorytmów korekcji błędów statycznych i dynamicznych przetworników; przykładem mogą tu być prace [2], [3]. Powyższe czynniki - obok ogólnych wymagań stawianych przyrządom mikroprocesorowym - są przyczyną, że oprogramowanie stanowi kompozycję wielu programów cząstkowych realizujących poszczególne zadania przyrządu. Duża liczba programów cząstkowych, ich wzajemne powiązania czasowe i funkcjonalne oraz potrzeba realizacji wielu zadań przyrządu powodują, że zbudowanie dobrego oprogramowania przyrządu jest procesem trudnym i czasochłonnym, szczególnie dla przyrządów przeznaczonych do pracy ciągłej. Zatem celowe wydaje się poszukiwanie ścieżek ułatwiających programowanie. Niniejszy artykuł stanowi próbę przedstawienia efektów prac autora nad poszukiwaniem struktur programowych, które zastosowane do budowy oprogramowania przyrządu mikroprocesorowego pracującego na bieżąco i w trybie ciągłym istotnie ułatwiają proces stworzenia tego oprogramowania. Opracowaną koncepcję zastosowano do budowy programu zarządzającego realizacją poszczególnych działań przyrządu nazwanego dystrybutorem zadań przyrządu mikroprocesorowego. Zrealizowlano dwie praktyczne wersje dystrybutora przeznaczone dla mikroprocesora Z80 i mikrokontrolera INTEL 8051.

Opracowana koncepcja budowy dystrybutora adresowana jest do określonej konstrukcji, którą można przyjąć jako reprezentowaną dla szerokiej klasy przyrządów mikroprocesorowych. Rozpatrując różne rozwiązania takich przyrządów, poczynając od elementarnych konstrukcji przetworników inteligentnych (ang. smart sensor [1]) a na konstrukcjach modularnych kończąc, można wskazać ich typowe układy funkcjonalne występujące praktycznie we wszystkich rozwiązanich. Do układów takich można zaliczyć:

- mikroprocesor z pamięcią programu typu ROM i pamięcią danych RAM,
- układ czasowo-licznikowy,
- układ współpracy z klawiaturą (zespołem przycisków) oraz wyświetlaczem,
- przetwornik analogowo-cyfrowy z komutatorem analogowym i wzmacniaczem pomiarowwym często o programowlanym wzmocnieniu,
- układy wejść i wyjść cyfrowych sterujących procesem pomiaru,
- układ transmisji szeregowej.

Przedstawione układy w przyrządach modularnych stanowią na ogół osobne bloki, a w prostych wersjach zabudowywane są łącznie z mikroprocesorem na jednej płycie.

W strukturach jednoukładowych duża ich część mieści się we wspólnej obudowie. W konstrukcjach przetworników inteligentnych wyraźna jest tendencja do umieszczania we wspólnej obudowie wszystkich elementów łącznie z czujnikami budowanymi na bazie krzemu [1].

Analizując ogólną strukturę przyrządu można wskazać szereg działań typowych dla przyrządu mikroprocesorowego niezależnie od stopnia jego złożoności. Uwzględniając przyjęte uprzednio założenie, że przyrząd działa w sposób ciągły, dane pomiarowe uzyskiwane są okresowo metodą próbkowania wielkości mierzonych. Na ogół wielkości tych jest kilka, gdyż często oprócz jednej lub kilku wielkości mierzonych próbkuje się aktualny poziom wielkości wpływających (jaką jest przykładowo temperatura), celem skorygowania błędów przez nie powodowanych. Aktualne wartości wielkości mierzonych są przetwarzane i wyprowadzane na wyświetlacz oraz na ogół umieszczane w pamięci w celu późniejszego zestawienia w protokole pomiarów. W takim przypadku każdy pomiar musi być opatrzony aktualnym czasem pomiaru, zachodzi więc konieczność wyznaczania czasu astronomicznego. Bloki zmierzonych danych są co jakiś czas transmitowane do komputera nadrzędnego, którym jest na ogół komputer klasy IBM PC, przy użyciu łącza szeregowego. Tą samą drogą mogą być wysyłane z komputera parametry określające aktualny tryb pracy przyrządu.

Przetworniki inteligentne pracujące na bieżąco każdy uzyskany wynik pomiaru przesyłają łączem szeregowym do nadrzędnego sterownika systemu pomiarowego lub automatyzacyjnego, którego są elementem. Wymaga to na ogół zapewnienia możliwości równoległej pracy programu pomiarów i programu transmisji danych.

Przedstawiona skrótowa charakterystyka zadań realizowanych przez przyrząd pozwala na zestawienie listy wymagań stawianych jego oprogramowaniu. Można wyróżnić dwie podstawowe grupy takich wymagań. Pierwsza grupa wymagań dotyczy własności funkcjonalnych oprogramowania. Podstawowe wymaganie jest dość oczywiste: w założonych warunkach przyrząd powinien poprawnie spełniać swoje funkcje. W praktyce oznacza to dostosowanie struktury programów do sytuacji skrajnych, w których wiele zadań cząstkowych musi być realizowanych równolegle w tych samych przedziałach czasu. Wzorowanie się na metodach pracy wieloprogramowej stosowanych w komputerach na ogół nie przynosi sukcesu z powodu braku odpowiednich mechanizmów sprzętowych w stosunkowo prostych mikroprocesorach stosowanych w przyrządach pomiarowych. Uwzględniając fakt, że spiętrzenia zadań mają charakter przypadkowy zależny od zdarzeń zachodzących w otoczeniu przyrządu, konieczne jest takie zaprojektowanie oprogramowania, które zapewniając ogólną

unifikację postępowania daje możliwość indywidualnego traktowania każdego problemu programowego w stopniu odpowiednim dla jego złożoności.

Druga grupa wymagań związana jest z wymogami w fazie tworzenia oprogramowania. Koszt oprogramowania stanowi istotny składnik ogólnych kosztów przyrządu, zatem pożądane są wszelkie działania zmierzające do skrócenia czasu tworzenia programu, jego uruchamiania i testowania. Przekładając te zalecenia na język praktyki można stwierdzić, że należy dążyć do stosowania możliwie dużej liczby standardowych, gotowych programów, a konstrukcja oprogramowania nadrzędnego powinna zapewnić łatwość dołączania własnych programów. Ponadto oprogramowanie to powinno być wyposażone w programy serwisowe ułatwiające uruchamianie i testowanie programów użytkowych.

Wymienione cechy oprogramowania przyrządu skłaniają do wniosku, że należy dążyć do skonstruowania programu pełniącego w przyrządzie mikroprocesorowym funkcję systemu operacyjnego. W literaturze brak jest jednak opisów prób całościowego ujęcia tego problemu. W tej sytuacji podjęto próbę opracowania ścieżki konstruowania programu zarządzającego aktywizowaniem poszczególnych funkcji przyrządu, który nazwano dystrybutorem zadań. W dalszym ciągu opisano własności dystrybutorów zbudowanych dla mikroprocesora Z80 i mikrokontrolera INTEL 8051.

2. PODSTAWOWE WŁASNOŚCI DYSTRYBUTORA

Ogólnym założeniem, traktowanym jako podstawa konstrukcji oprogramowania, jest przyjęcie ciągłej gotowości przyrządu do nawiązania komunikacji z komputerem nadrzędnym. Oznacza to, że w tle programu głównego realizującego w sposób ciągły zadania pomiarowe, dla realizacji których przyrząd został zbudowany, "czuwa" program transmisji, który może być w dowolnym momencie zainicjowany przez komputer nadrzędny. Przykładowo taka sytuacja może mieć miejsce, gdy komputer ten inicjuje odbieranie danych pomiarowych zebranych uprzednio przez przyrząd. W praktyce oznacza to dwuprogramową pracę przyrządu. W zbudowanej wersji dystrybutora funkcjonują dwa rodzaje programów tła, aktywizowane w zależności od trybu pracy przyrządu. W trybie podstawowym programem tła jest program transmisji bloków danych między przyrządem a IBM PC. Drugim trybem pracy jest tryb serwisowy, kiedy programem tła jest program typu MONITOR, pozwalający na realizację

podstawowych działań związanych z uruchamianiem i testowaniem programu głównego. W tym trybie IBM PC służy jako konsola operatorska MONITORA, a także jako urządzenie do redagowania, tłumaczenia i przekazywania do przyrządu uruchamianych programów użytkowych. Należy przy tym pamiętać że w obu trybach pracy programem głównym jest program realizacji pomiarów i program tła nie może zakłócać funkcji programu głównego.

W opisywanej wersji oprogramowania rolę koordynatora wszystkich działań przyrządu spełnia dystrybutor zadań. Przyjmując, że poszczególne funkcje przyrządu realizują określone konstrukcje programowe, dystrybutor musi spełniać trzy podstawowe funkcje:

- ustalać kolejność wykonywania poszczególnych zadań,
- rozstrzygać o pierwszeństwie wykonywania zadań w przypadku wystąpienia konfliktu czasowego,
- określać sposoby wymiany informacji między zadaniami.

W dalszej części zostaną przedstawione niektóre bardziej istotne własności dystrybutora.

3. SPOSÓB DZIAŁANIA DYSTRYBUTORA

Dystrybutor zadań jest programem organizującym współdziałanie programów cząstkowych, realizujących poszczególne zadania przyrządu. Podstawowym elementem organizującym jest rejestr stanu dystrybutora przechowujący informacje o aktualnym statusie wszystkich programów cząstkowych. Status każdego z programów określany jest jednobitowo: 0 oznacza, że dany program nie wymaga realizacji, 1 - program oczekuje na wykonanie. Zatem rejestr stanu można określić jako fragment pamięci danych zorganizowany bitowo, zawierający ogólnie N bitów uszeregowanych wg określonej hierarchii:

bit 1, bit 2,...,bit n,...,bit N.

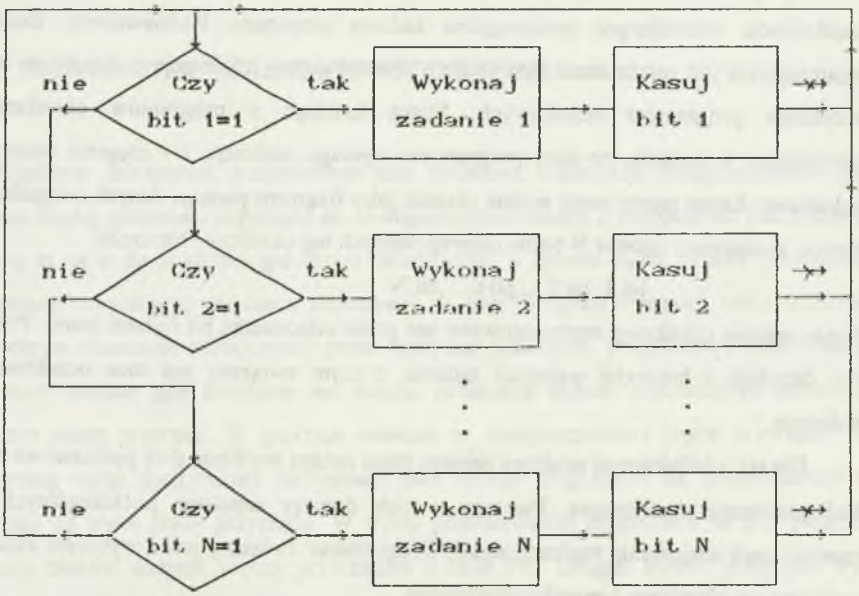
Każde zadanie cząstkowe reprezentowane jest przez odpowiedni bit rejestru stanu. Położenie bitu decyduje o hierarchii ważności zadania, z czym związany jest czas oczekiwania na realizację.

Dla tak zdefiniowanej struktury rejestru stanu można wyróżnić dwa podstawowe obszary funkcjonowania dystrybutora. Pierwszy z nich dotyczy zapalania poszczególnych bitów rejestru, czyli inicjowania realizacji zadań. Drugi obszar związany jest z wyborem zadań i ich realizacją wg określonej hierarchii priorytetów.

Wskazywanie zadań do realizacji może dokonywać się dwoma sposobami. Jeden polega na wykorzystaniu przerwań: pojawienie się zdarzeń zewnętrznych (np. naciśnięcie przycisku przyrządu) uaktywnia sygnał przerwania. Identyfikacja przerwania i wykonanie jego elementarnej obsługi sygnalizowane są zapaleniem odpowiedniego bitu rejestru stanu (przykładowow bitu wskazującego naciśnięcie przycisku WYKONAJ POMIAR). Drugi sposób inicjowania zadań polega na zapalaniu ich bitów stanu w wyniku realizacji innych zadań. Przykładem może tu być program wyznaczania czasu astronomicznego: przepełnienie licznika minut zapala bit stanu inicjujący zadanie dodania godziny do licznika godzin.

Wyboru zadań do realizacji dokonuje fragment programu nazywany główną pętlą dystrybutora. Zasadę działania tej pętli w jednej z najprostszych postaci pokazuje rysunek 1. Można powiedzieć, że z punktu widzenia struktury dystrybutora pętla główna stanowi program główny, natomiast pozostałe stuktury programowe są podprogramami realizującymi określone zadania.

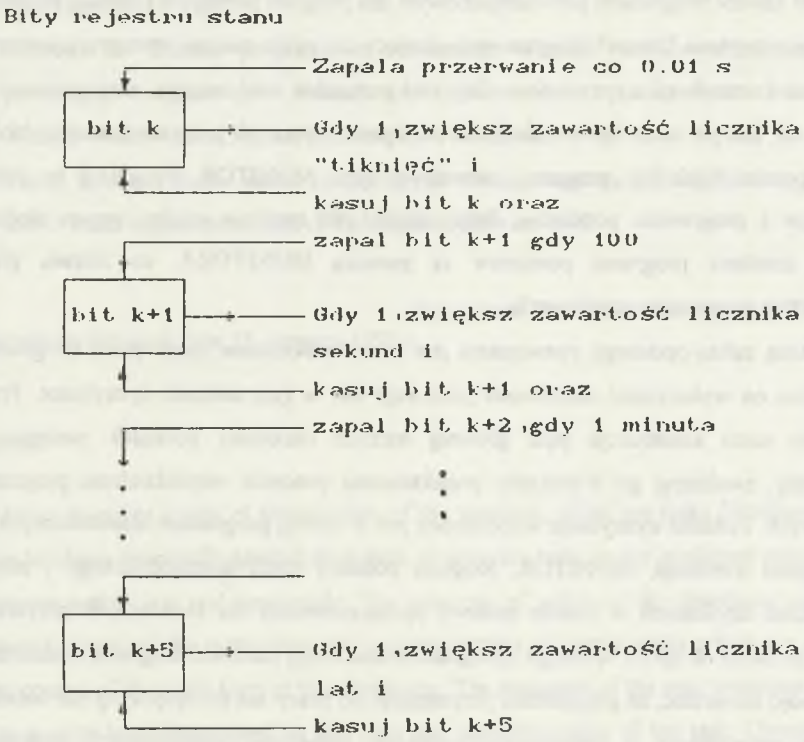
Sieć działań pętli głównej pokazana na rysunku 1 narzuca procedurę obsługi zadań wg następującej zasady: im niższy numer bitu zadania, tym wyższy priorytet. Oznacza to, że z dwóch zadań czekających na wykonanie realizowane jest najpierw to o wyższym priorytecie. Ponadto bit rejestru stanu o wyższym priorytecie jest przeglądany częściej niż bit zadania o niższym priorytecie.



Rys. 1. Sieć działań głównej pętli dystrybutora
Fig. 1. Block-diagram of the main loop of the distributor

PRZYKŁAD

Zostanie tu pokazany sposób, jaki zastosowano w celu wyznaczania czasu astronomicznego. Sposób ten dobrze ilustruje typowe elementy postępowania w procesie budowy programów cząstkowych. Ogólny problem wyznaczania czasu astronomicznego został podzielony na szereg zadań o różnej skali ważności. Najczęściej muszą być obsługiwane zadania realizowane w krótkich odcinkach czasu; niższy priorytet mają zadania pojawiające się w dłuższych odstępach czasu. Zasada wyznaczania czasu polega na zliczaniu "tiknięć" pojawiających się na skutek występowania przerwań od układu czasowo - licznikowego co 0.01s (przerwania te są wykorzystywane także do innych zadań, np. sterowania wyświetlaczem ciekłokrystalicznym). Strukturę fragmentu pętli głównej używanego do wyznaczania czasu astronomicznego ilustruje rysunek 2.



Rys. 2. Wyznaczanie czasu astronomicznego
Fig. 2. The principle of calculation of the astronomical time

Przedstawiony przykład wskazuje również na istotne zalecenie, które powinno być brane pod uwagę w trakcie ustalania struktury zadań cząstkowych i ich priorytetów. Mianowicie każde z zadań powinno być dzielone na możliwie dużą liczbę podzadań o różnych poziomach ważności: od pilnych do takich, które mogą być wykonywane z relatywnie dużymi opóźnieniami czasowymi. Odpowiednio wysokie usytuowanie pilnych fragmentów zadań umożliwi bezkolizyjne, praktycznie równoległe wykonuwanie wielu funkcji przyrządu.

4. UWAGI KOŃCOWE

Opisany sposób konstruowania oprogramowania zastosowano w praktyce do budowy dystrybutorów zadań przeznaczonych do zastosowań w przyrządach z mikroprocesorem Z80 i mikrokontrolerem INTEL 8051 [4]. Istotną cechą tych rozwiązań jest dwuprogramowa praca, przy czym zawsze programem pierwszoplanowym jest program pomiarów i obsługi przyrządu. W tle tego programu "czuwa" program sprawdzający czy miała miejsce zewnętrzna inicjatywa nawiązania komunikacji z przyrządem. Gdy taki przypadek miał miejsce, aktywizowany jest program tła, którym może być w zależności od wyboru operatora program transmisji bloków danych pomiarowych lub program serwisowy typu MONITOR. Programy te działają równolegle z programem pomiarów, dzięki czemu jest możliwe między innymi śledzenie efektów działania programu pomiarów za pomocą MONITORA, co ułatwia proces uruchamiania programów użytkowych.

Ważną zaletą opisanego rozwiązania jest znaczne skrócenie czasu pracy programisty, gdy zechce on wykorzystać możliwości jakie daje mu w tym zakresie dystrybutor. Przede wszystkim sama konstrukcja pętli głównej narzuca określony porządek postępowania programisty, zwalniając go z potrzeby projektowania procedur współdziałania programów cząstkowych. Ponadto dystrybutor wyposażony jest w szereg programów standardowych, jak np. program transmisji, MONITOR, program pomiaru czasu astronomicznego i inne. Z doświadczeń uzyskanych w trakcie budowy oprogramowania dla konkretnych przyrządów wynika, że około 60 do 70 % całego oprogramowania mogą stanowić programy standardowe. Można więc stwierdzić, że programista przystępując do pracy ma do dyspozycji już określoną strukturę programową o narzuconym porządku dołączania specyficznych programów pomiarowych i wyposażoną w narzędzia uruchamiania tych programów.

W chwili obecnej zbierane są doświadczenia eksploatacyjne z pracy przyrządów wyposażonych w oprogramowanie zbudowane w oparciu o dystrybutor zadań. Planuje się ponadto eksperymenty z nowymi strukturami pętli głównej, pozwalającymi na adaptację działania przyrządu do skrajnych sytuacji przy spiętrzeniach zadań. Możliwe są tu konstrukcje z maskowaniem bitów rejestru stanu czy wręcz dynamicznej zmiany struktury pętli dystrybutora.

LITERATURA

[1] Brignell J.E.: Smart Sensors, in Grandtke T. and W.H.Ko (eds), Sensors; A Comprehensive Survey, Vol.1, Fundamentals and General Aspects, VCH, Weinheim, 1989, Ch.12

[2] Brignell J.E.: Software Techniques for Sensor Compensation. Sensors and Actuators A, 25-27, pp.29-35, 1991

[3] Jakubiec J.: Bieżące programowe odtwarzanie wartości chwilowych dynamicznych przebiegów wejściowych nieliniowych przetworników pomiarowych. Monografia. Zeszyty Naukowe Pol. Śl., seria Elektryka .111, Gliwice 1988

[4] Peroń Z.: Dystrybutor zadań pomiarowych mikrokomputera jednoukładowego INTEL 8051. Praca dyplomowa. IMiAE Pol.Śląskiej, Gliwice 1992

Recenzent: Prof.dr hab.inż. Zygmunt Kuśmierek

Wpłynęło do Redakcji dnia 15 czerwca 1993 r.

Abstract

The paper describes a way of construction of the program, called the tasks distributor, which allows to obtain practically parallel execution of program tasks in the intelligent measurement instrument acting in a real time mode. The principle of action of the distributor consists in dividing all tasks of the instrument into a great number of partial tasks which are activated under control of the main loop of the distributor. The sequence of the task activation depends on its priority level determined by the rank and execution time of the task. Owning to the build-in standard programs one obtains possibility of substantial reduction of time neccesary to create the instrument's program.