

Poznajemy mikrokontrolery ST7Lite, część 3

Podstawą działań wielu programów są wszelkiego rodzaju operacje arytmetyczne wykonywane czy to na zmiennych wejściowych, czy to na zmiennych wewnętrznych. Podstawą wielu interfejsów użytkownika jest bardzo często wyświetlacz LED czy LCD i konwersja liczb binarnych na dziesiętne. Trudno jest bowiem zrozumieć i właściwie zinterpretować wynik pomiaru wyświetlany na przykład w kodzie U2.

W artykule zaprezentuję podstawowe działania arytmetyczne w języku assembler mikrokontrolera ST7. Opiszę dodawanie, odejmowanie, mnożenie i dzielenie liczb. Na podstawie tych działań zaprezentuję również funkcje porównywania liczb oraz konwersję liczb binarnych na dziesiętne.

W artykule wykorzystano algorytm i kody źródłowe zawarte w nocy aplikacyjnej firmy ST Microelectronics *ST7 Math Utility Routines*.

Dzielenie całkowite dwóch liczb 1-bajtowych bez znaku

Na rys. 5 przedstawiono schemat podprogramu dzielenia dwóch

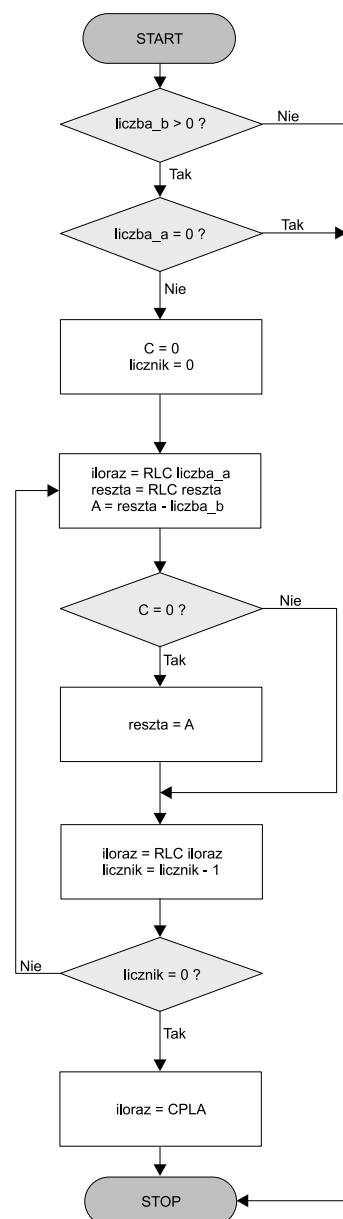
liczb 1-bajtowych zamieszczonego na list. 4. Obie liczby muszą być większe od 0. Funkcja nie sygnalizuje faktu błędnej operacji dzielenia (dzielenia przez 0). W przypadku, gdy dzielna lub dzielnik nie są właściwe, wynikiem działania jest po prostu iloraz równy 0 a funkcja kończy pracę. Wykonywana jest operacja dzielenia liczb całkowitych $liczba_a/liczba_b$.

List. 4. Podprogram dzielenia dwóch liczb 1-bajtowych

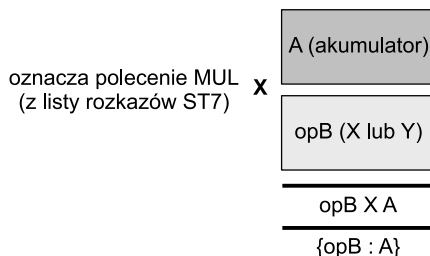
```

;*****
;dzielenie liczba_a / liczba_b
;argumenty:
; - liczba_a = dzielna
; - liczba_b = dzielnik
; - liczba_a i liczba_b ≠ 0
; - cnt (jako licznik działań)
;rezultaty:
; - iloraz zawiera część całkowitą z ilorazu liczb
; - reszta zawiera resztę z dzielenia
;modyfikowane:
; - zmienna licznikowa cnt
; - flagi Z, C
;
;PRZYKŁAD UŻYCIA
;ld A,#SE7
;ld  liczba_a,A
;ld  A,#$10
;ld  liczba_b,A
;call div_ab
;*****
.div_ab
;inicjacja zmiennych
push A ;zapamiętanie akumulatora
clr iloraz ;część całkowita = 0
clr reszta ;reszta z dzielenia = 0
;sprawdzenie, czy liczba_a i liczba_b ≠ 0
ld A,liczba_b
jrugt dziel0 ;jeśli liczba_b ≠ 0, to sprawdzenie liczba_a
end dziel
pop A ;koniec pracy podprogramu, odtworzenie akumulatora
ret
dziel0
ld A,number_a ;jeśli A=0, to wynik również = 0
jreq end_div ;jeśli A≠0, to wykonaj działania
;podprogram wykonujący dzielenie
rcf ;flaga C=0
ld A,#$08 ;inicjacja licznika przesunąć liczby
ld cnt,A
ld A,liczba_a
rlc A ;przesunięcie w lewo i zapamiętanie rezultatu
ld iloraz,A
dziel1
ld A,reszta
rlc A ;resztę z dzielenia w lewo, aby uwzględnić flagę C
ld reszta,A
sub A,liczba_b ;kontynuacja pracy?
jrc dziel2
ld reszta,A
dziel2
ld A,iloraz
rlc A ;przesunięcie w lewo ilorazu
ld iloraz,A
dec cnt
jreq dziel3 ;jeśli licznik przesunąć=0, to koniec pracy
jra dziel1
dziel3
cpl A ;negowanie wyniku (ilorazu)
ld iloraz,A ;zapamiętanie ilorazu
jra end_dziel ;koniec pracy funkcji

```



Rys. 5. Schemat działania podprogramu dzielenia liczb 1-bajtowych



Rys. 6. Uproszczony schemat funkcjonowania polecenia MUL

Rezultatem działania jest wynik zapamiętany w zmiennych *iloraz* i *reszta*.

Mnożenie całkowitych liczb 1-bajtowych bez znaku

Prezentowany na list. 5 podprogram mnoży przez siebie dwie liczby 1-bajtowe (powszechnie używane jest określenie typu zmiennych: *byte*). ST7 zawiera na swojej liście rozkazów polecenie mnożenia dwóch liczb, jednak trzeba pamiętać o tym, że wynik mnożenia (iloczyn) może być liczbą 2-bajtową, inaczej niż czynniki. W zwią-

List. 5. Podprogram mnożenia dwóch liczb 8-bitowych

```

;*****
;mnozenie liczba_a x liczba_b
;argumenty:
; - liczba_a, liczba_b (czynniki)
;rezultaty:
; - liczba_a (młodszy bajt iloczynu)
; - liczba_b (starszy bajt iloczynu)
;modyfikowane:
; - flagi H, C
;
;PRZYKŁAD UŻYCIA:
;ld A,#$A0
;ld liczba_a,A
;ld A,#$10
;ld liczba_b,A
;call mul_ab
;*****
.mul_AB
;zapamiętanie modyfikowanych rejestrów
push X
push A
;załadowanie zmiennych
ld X,liczba_A
ld A,liczba_B
mul X,A
ld liczba_a,A
ld liczba_b,X
;odtworzenie rejestrów
pop A
pop X
;powrót z podprogramu
ret

```

ku z tym polecenie mnoży zawartość rejestru indeksowego X lub Y (opB) przez zawartość akumulatora (A), a wynik umieszcza w parze rejestrów: odpowiednio X i Y, lub Y i A. Akumulator zawiera mniej znaczący bajt iloczynu, natomiast

rejestr indeksowy – bardziej znaczący. Uproszczony schemat funkcjonowania polecenia MUL umieszczono na rys. 6.

Jacek Bogusz, EP
jacek.bogusz@ep.com.pl

STALEWSKI i RESZCZYK

SPÓŁKA KOMANDYTOWA

Usługi kontraktowej produkcji elektronicznej

- Małe, jak i duże serie oraz prototypy
- Krótkie terminy realizacji
- Najwyższa jakość wykonania
- Technika bezolowiowa
 - montaż powierzchniowy SMT
 - montaż przewlekany THT
 - lakierowanie płytek
 - testowanie funkcjonalne
 - linia do montażu urządzeń elektronicznych

Usługi formowania wtryskowego tworzyw sztucznych

- Cztery wtryskarki poj. wtrysku od max 180 cm³ do max 820 cm³, waga wypraski od max 162 g do max 738 g, siła zwarcia od max 1200 KN do max 2680 KN.
- produkcja z formy Klienta
- doradztwo w zakresie konstrukcji i wykonywania oprzyrządowania form wtryskowych
- doradztwo w zakresie doboru materiału
- obróbka powierzchni form wtryskowych



ul. Okrężna 1 b, 19-300 Elk, Tel/fax: +48 87 6201630, E-mail: z.holdyk@rsc.com.pl, www.rsc.com.pl