

Obsługa wyświetlaczy graficznych z wbudowanym kontrolerem S6B1713, część 3

Przykłady dla mikrokontrolera AT91SAM7S256

Prezentację sposobu obsługi wyświetlacza z kontrolerem S6B1713 kończymy omówieniem sposobu sterowania podświetlaczem LED RGB. Do tego celu wykorzystano wielokanałowy generator PWM, którego efekty działania widać na fotografiach.

Generator PWM w mikrokontrolerze AT91SAM7S256

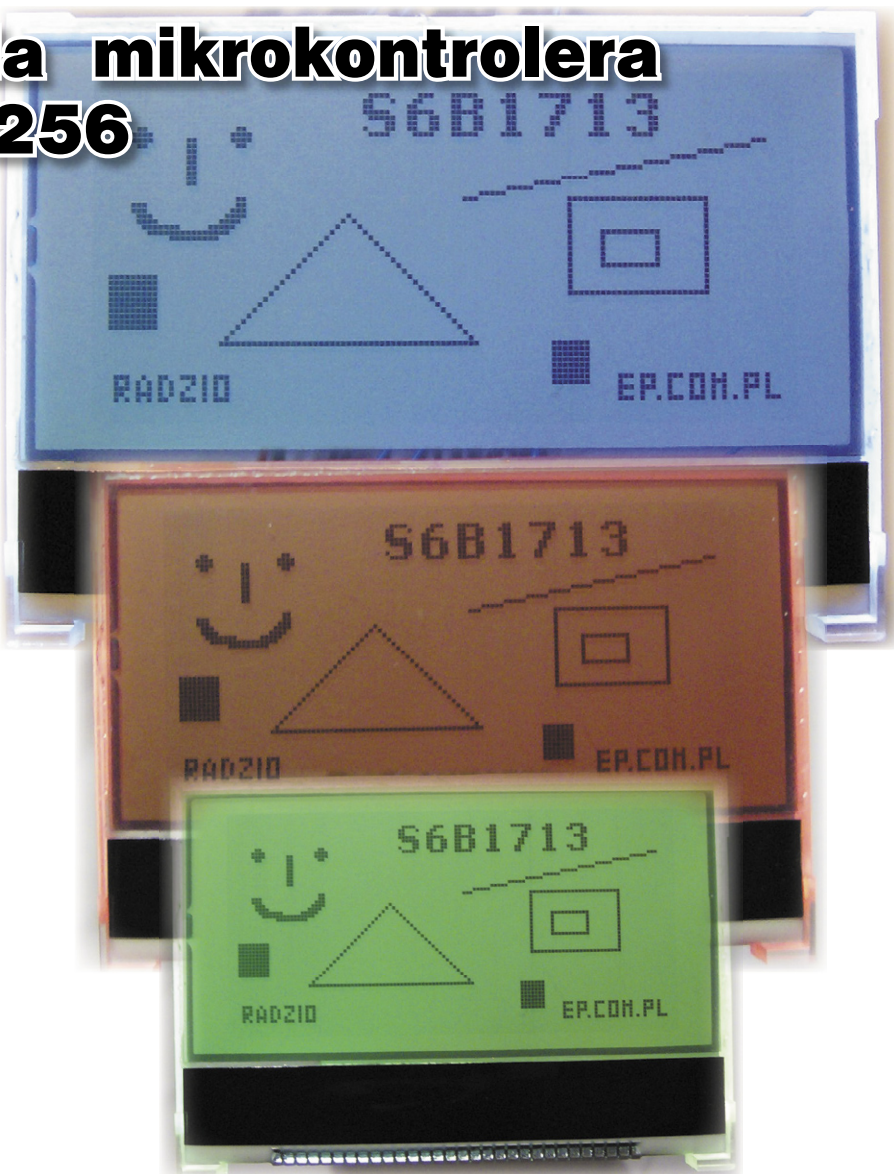
Budowę generatora PWM w mikrokontrolerze AT91SAM7S256 przedstawiono na rys. 5. Składa się on z czterech identycznych kanałów taktowanych sygnałem MCK za pośrednictwem układu Clock Generator, umożliwiającego wybór współczynnika podziału sygnału MCK w szerokim zakresie. Budowę układu Clock Generator przedstawiono na rys. 6.

Układ Clock Generator podzielono na trzy części:

- licznik modulo n umożliwiający uzyskanie 11 wartości sygnału zegarowego:

$$F_{MCK}, \frac{F_{MCK}}{2}, \frac{F_{MCK}}{4}, \frac{F_{MCK}}{8}, \frac{F_{MCK}}{16}, \frac{F_{MCK}}{32}, \frac{F_{MCK}}{64}, \frac{F_{MCK}}{128}, \frac{F_{MCK}}{256}, \frac{F_{MCK}}{512}, \frac{F_{MCK}}{1024}$$

- dwa liniowe dzielniki dostarczające dwa niezależne sygnały CLK_A i CLK_B otrzymane wskutek podziału dowolnego sygnału



wyjściowego licznika modulo n przez dowolną liczbę z zakresu 1...255.

Inicjalizacja układu PWM

Aby możliwe było wykorzystanie układu PWM, konieczne jest przeprowadzenie inicjalizacji według następującej sekwencji:

- przypisanie wyprowadzeniom

PA23, PA24 i PA25 funkcji wyjść układu PWM (*Peripheral B*) i dezaktywacja linii GPIO na tych wyprowadzeniach.

- włączenie taktowania układu PWM poprzez ustawienie odpowiedniej flagi w rejestrze PCER układu PMC.
- konfiguracja generatora PWM. Kod funkcji realizującej inicjacji-

List. 15.

```

void PWM_Init(void)
{
    // konfiguracja układu PIO
    AT91PS_PIO pPIO = AT91C_BASE_PIOA;
    pPIO->PIO_BSR = (AT91C_PA23_PWM0 | AT91C_PA24_PWM1 | AT91C_PA25_PWM2);
    pPIO->PIO_PDR = (AT91C_PA23_PWM0 | AT91C_PA24_PWM1 | AT91C_PA25_PWM2);

    // włączenie taktowania układu PWM
    AT91PS_PMC pPMC = AT91C_BASE_PMC;
    pPMC->PMC_PCEER = (1 << AT91C_ID_PWMC);

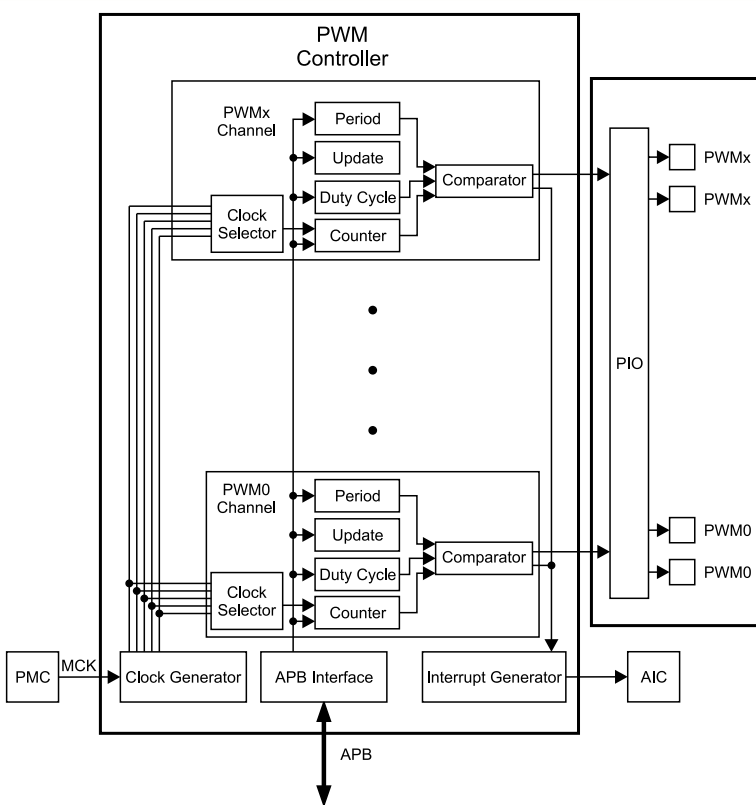
    // konfiguracja układu PWM
    AT91PS_PWMC pPWM = AT91C_BASE_PWMC;
    pPWM->PWM_MR = (1 << 0) | (0xA << 8) | (1 << 16) | (0xA << 24);

    pPWM->PWM_CH[0].PWM_CMR = 0x208; //
    pPWM->PWM_CH[1].PWM_CMR = 0x208; //
    pPWM->PWM_CH[2].PWM_CMR = 0x208; //

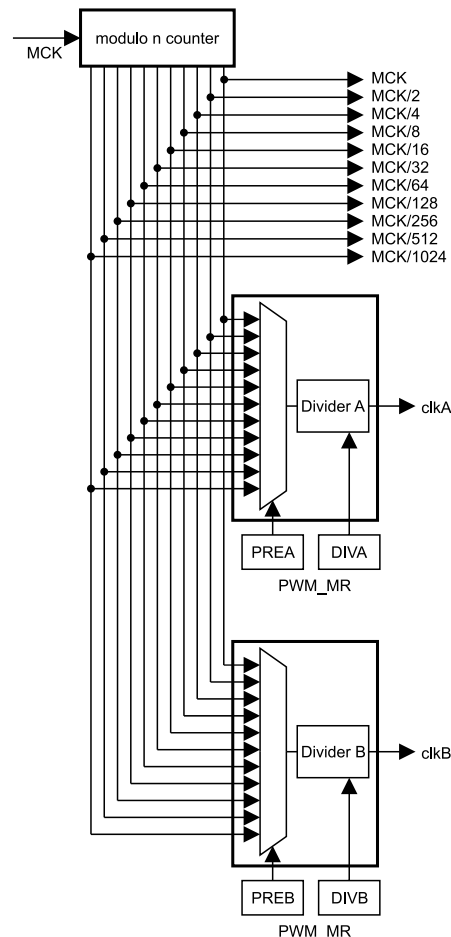
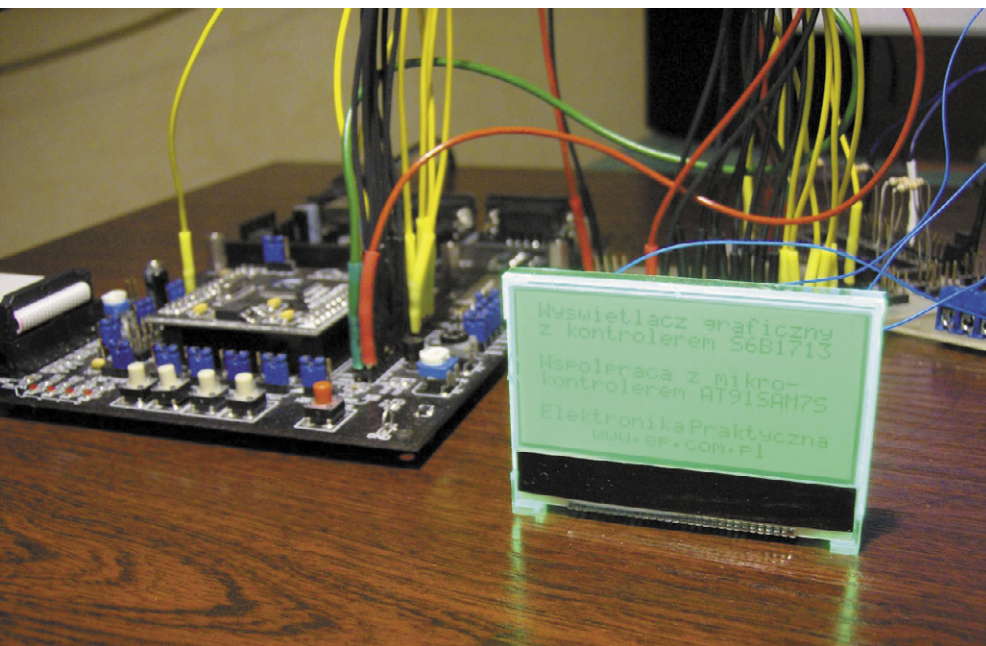
    pPWM->PWM_CH[0].PWM_CPRDR = 101;
    pPWM->PWM_CH[1].PWM_CPRDR = 101;
    pPWM->PWM_CH[2].PWM_CPRDR = 101;

    pPWM->PWM_CH[0].PWM_CDTYR = 0;
    pPWM->PWM_CH[1].PWM_CDTYR = 0;
    pPWM->PWM_CH[2].PWM_CDTYR = 0;

    pPWM->PWM_ENA = (AT91C_PWM_CHID0 | AT91C_PWM_CHID1 | AT91C_PWM_CHID2);
}
    
```



Rys. 5.



Rys. 6.

List. 16.

```

void PWM_Update(int ch0, int ch1, int ch2)
{
    AT91PS_PWMC pPWM = AT91C_BASE_PWMC;

    pPWM->PWM_CH[0].PWM_CUPDR = ch0;
    pPWM->PWM_CH[1].PWM_CUPDR = ch1;
    pPWM->PWM_CH[2].PWM_CUPDR = ch2;
}
    
```

zaczę układu PWM przedstawiono na list. 15.

Uaktualnienie wartości współczynnika wypełnienia

Po uruchomieniu układu PWM zmiana zawartości rejestrów CPRD lub CDTY (w zależności od stanu bitu CPD w rejestrze CMR) jest możliwa wyłącznie poprzez zapis do rejestru CUPD. Kod funkcji uaktualniającej zawartość rejestru CDTY przedstawiono na list. 16.

Wywołanie tej funkcji w dowolnym miejscu programu spowoduje zapis nowych wartości współczynnika wypełnienia sygnału generowanego przez układ PWM, co objawi się zmianą koloru świecenia podświetlacza RGB.

Radosław Kwiecień, EP
radoslaw.kwiecien@ep.com.pl