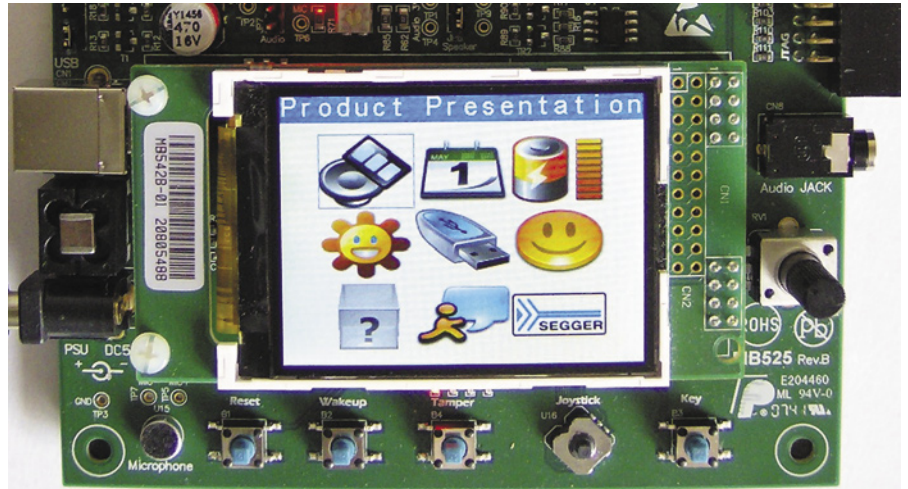


STM3210B-EVAL

Corteksowy zestaw ewaluacyjny

Zestaw ewaluacyjny STM3210B-EVAL/A jest kolejnym omawianym na łamach Elektroniki Praktycznej zestawem uruchomieniowym dla mikrokontrolerów STM32 firmy STMicroelectronics. Są to szybkie i tanie 32-bitowce, o doskonałym wyposażeniu, nowoczesnym rdzeniu Cortex-M3, dostępne w wielu wariantach obudów.

Zestaw składa się z dwóch płytek drukowanych: głównej, oznaczonej symbolem MB525 oraz mniejszej płytki, na której umieszczono kolorowy wyświetlacz LCD TFT 320x240. Schemat blokowy zestawu przedstawiono na rys. 1. Najważniejszym elementem płyty głównej jest mikrokontroler SMT32F103VBT6 w 100-wyprowadzeniowej obudowie LQFP. Zestaw posiada następujące układy peryferyjne: 8 MB pamięci Flash, złącze karty pamięci MicroSD podłączone do interfejsu SPI1 mikrokontrolera, czujnik temperatury LM75 podłączony do interfejsu I²C1, dwa złącza RS232 wraz z konwerterami poziomów napięć podłączone do interfejsów USART1 i USART2, transceiver IrDA podłączony do interfejsu USART3 oraz interfejs kart SmartCard (do zestawu dołączono kartę SmartCard do testów). Na płycie głównej znajdu-



Fot. 2.

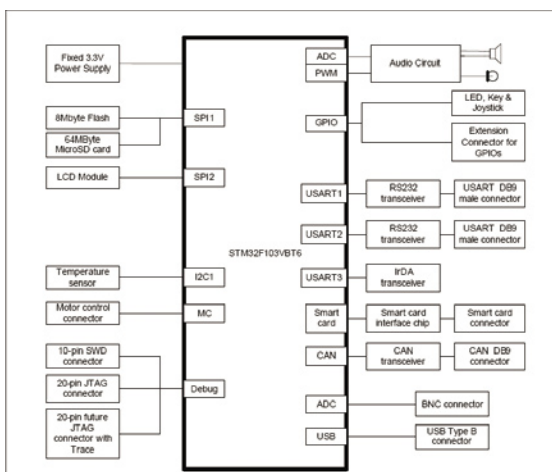
ją się również obwody umożliwiające nagrywanie dźwięku (mikrofon i wzmacniacz wejściowy) oraz jego odtwarzanie (wzmacniacz i głośnik), oraz obwody komunikacji z użytkownikiem (joystick, przyciski oraz diody LED). Zestaw ewaluacyjny jest fabrycznie zaprogramowany aplikacją demonstracyjną, która przedstawia najważniejsze cechy zestawu STM3210B-EVAL. Na wyświetlaczu LCD wyświetlane jest dziewięciopozycyjne menu (fot. 2), po którym użytkownik porusza się za pomocą joysticka. Wybór zaznaczonej ikonki następuje po naciśnięciu joysticka, natomiast kursor jest przemieszczany po ikonach po przechyleniu joysticka w wybranym kierunku. Pierwsza pozycja menu

kalendrza to, oczywiście, kalendarz i zegar. Możliwe jest ustawienie lub wyświetlenie czasu, daty oraz alarmu (fot. 4).



Trzecia pozycja menu umożliwia uruchomienie trybów oszczędzania energii dostępnych w mikrokontrolerach STM32. Ikonka przedstawiająca „uśmiechnięte słoneczko” wbrew pierwszemu skojarzeniu nie uruchamia komunikatora internetowego, lecz wyświetla aktualną temperaturę odczytaną z czujnika LM75 (fot. 5).

Jedną z najciekawszych opcji zestawu jest dostępna pod ikonką przedstawiającą pendrive'a. Po wybraniu tej opcji i podłączeniu zestawu kablem USB do kompute-

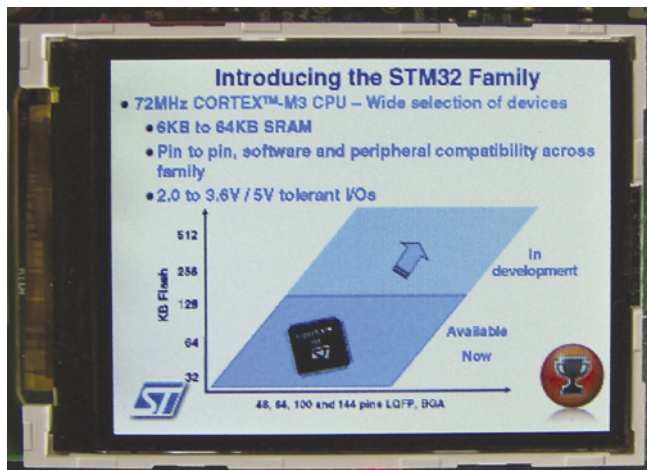


Rys. 1.

Product presentation pozwala uruchomić prezentację omawiającą rodzinę mikrokontrolerów STM32 (fot. 3). Kolejne slajdy prezentacji są omawiane przez lektora, którego głos wydobywa się z wbudowanego w zestaw głośniczka. Jakość dźwięku generowanego przez wbudowany głośnik nie jest najlepsza, jednak po podłączeniu zewnętrznego zestawu audio przez złącze minijack znacznie się poprawia.

Druga pozycja menu, oznaczona ikonką kartki

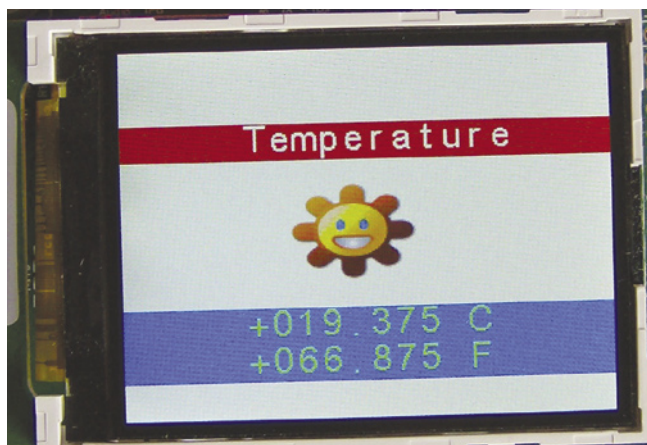




Fot. 3.



Fot. 4.



Fot. 5.

ra PC, karta MicroSD umieszczona w złączu na płytce zestawu jest dostępna w systemie jako wymienny nośnik pamięci. Prędkość zapisu danych na taki dysk USB waha się w pobliżu 280 kB/s, natomiast prędkość odczytu danych ok. 320 kB/s.

Pozostałe opcje menu nie są już tak interesujące jak omówione do tej pory, więc omówię je skrótowo: uśmiechnięte kółeczko uruchamia program rysujący różnokolorowe

okregi na wyświetlaczu, pudełko ze znakiem zapytania wyświetla pomoc na temat joysticka, natomiast człowiek z dymkiem wyświetla informacje na temat autorstwa i wersji oprogramowania demonstracyjnego. Oprócz oprogramowania demonstracyjnego, którym fabrycznie zaprogramowany jest mikrokontroler znajdujący się na płytce zestawu STM3210B-EVAL, dostępne są również bardziej praktyczne programy będące przykładami wykorzystania przygotowanej przez firmę STMicroelectronics biblioteki ARM®-based 32-bit MCU STM32F101xx and STM32F103xx firmware library.

STM32 firmware library

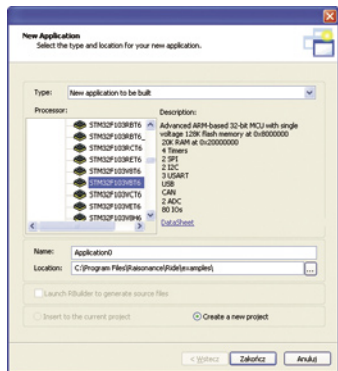
Biblioteka ta jest zbiorem procedur, struktur danych oraz makroinstrukcji, umożliwiających wykorzystanie wszystkich możliwości układów peryferyjnych mikrokontrolerów STM32. Biblioteka została napisana w języku ANSI-C i co najważniejsze – w odróżnieniu

od bibliotek dostarczanych przez producentów komercyjnych narzędzi uruchomieniowych, może być zastosowana z dowolnym kompilatorem języka C dla mikrokontrolerów STM32. Biblioteka została podzielona na osobne sterowniki dla każdego typu układu peryferyjnego występującego w mikrokontrolerach STM32. Nazewnictwo struktur danych, funkcji i ich parametrów jest ujednolicone i intuicyjne. Korzy-

stanie z tej biblioteki pozwala na szybkie i przyjemne tworzenie aplikacji dla mikrokontrolerów STM32. Osoby czujące awersję do grzebania się w rejestrach sprzętowych i ich bitach, których nazwy z konieczności są skrótowe i zawiłe, odczuwają przyjemność płynącą z pisania programów korzystających z omawianej biblioteki. Niestety, jeśli coś jest przyjazne programiście jest mniej przyjazne oprogramowanemu sprzętowi. Oczywistą konsekwencją zastosowania omawianej biblioteki jest spadek wydajności oraz wzrost zajętości pamięci programu. Jeśli więc wydajność wykonywania programu nie jest priorytetem, to zastosowanie biblioteki jest w pełni uzasadnione. Jeśli natomiast chcemy wydusić z jednostki centralnej ostatki mocy obliczeniowej, należy zrezygnować z zastosowania biblioteki i skorzystać z rozwiązań innych dostawców oprogramowania lub stworzyć własne funkcje obsługi urządzeń peryferyjnych. W takim przypadku biblioteka może zostać potraktowana jako praktyczny przewodnik korzystania z poszczególnych układów peryferyjnych, będący wskazówką przy pisaniu własnych sterowników do układów peryferyjnych.

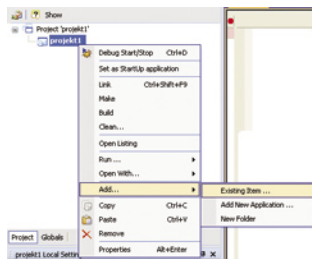
Środowisko Ride7 + RKitARM

Francuska firma Raisonance jest jednym ze stałych dostawców środowisk uruchomieniowych dla mikrokontrolerów firmy STMicroelectronics. Czytelnicy Elektroniki Praktycznej doskonale znają środowisko Ride w wersji szóstej przeznaczonej dla mikrokontrolerów ST7 – w jednym z odcinków kursu „LITEcomp – aplikacje” przedstawiono aplikację stworzoną w środowisku Ride dla mikrokontrolerów ST7. Środowisko Ride jest również dostępne w wersji dla mikrokontrolerów z rdzeniem ARM produkowanych przez firmę STMicroelectronics: od dłuższego czasu STR7 i STR9, a ostatnio również STM32. Co ważne, środowisko wykorzystuje kompilator GCC i jest dostępne za darmo bez ograniczeń co do rozmiaru tworzonej aplikacji, czy też czasu działania. Jedynym ograniczeniem jest limit kodu debugowanego za pomocą interfejsu RLink-Std do 32 kB. Jeśli jednak zrezygnujemy z funkcji debugowa-



Rys. 6.

nia wewnątrzukładowego, to środowisko Ride nie posiada praktycznie żadnych ograniczeń. Jest to ważna cecha, gdyż w chwili pisania artykułu najpopularniejszy darmowy pakiet uruchomieniowy WinARM nie



Rys. 7.

obsługuje jeszcze mikrokontrolerów z rdzeniem ARM Cortex-M3.

Środowisko Ride 7 w porównaniu z wersją szóstą otrzymało całkowicie nowy, przyjemniejszy dla oka wygląd. Tworzenie projektu przebiega w sposób podobny jak we wcześniejszych wersjach środowiska Ride: z menu *Project* należy wybrać opcję *New project*. Ukáže się okno

dialogowe (rys. 6), w którym należy wybrać typ mikrokontrolera, dla którego tworzony jest projekt oraz podać jego nazwę i lokalizację. Następnie należy utworzyć plik źródłowy i dodać go do drzewa projektu (rys. 7). Utworzony projekt jest domyślnie skonfigurowany do współpracy z omówioną wcześniej biblioteką funkcji opracowaną przez firmę STMicroelectronics.

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

Alternatywne narzędzia dla mikrokontrolerów STM32 opisywaliśmy w EP12/2007. Mikrokontrolery STM32 przedstawialiśmy w artykułach publikowanych w EP7/2007, 10/2007 oraz 2/2008. Artykuły te publikujemy na CD-EP4/2008B.

Pierwszy program

Ponieważ uważam, że nie ma sensu omawiać jakichkolwiek narzędzi uruchomieniowych bez ich praktycznego przetestowania, przedstawiam prosty przykładowy program korzystający z omówionej biblioteki. Będzie on bardzo prosty – jego działanie sprowadzi się do migania diodą LED. W przypadku zestawu STM3210B-EVAL dioda LED podłączona do wyprowadzenia PC6, jak również pozostałe diody są sterowane stanem wysokim. Ustawienie na wyprowadzeniu portu stanu wysokiego spowoduje zaświecenie się diody LED. W związku z tym, wyprowadzenia sterujące diodami LED należy skonfigurować jak pracujące w trybie przeciwsobnym (*push-pull*). Kod programu przedstawiony jest na listingu:

```
#include „stm32f10x_lib.h”

GPIO_InitTypeDef GPIO_InitStructure;

int main(void)
{
    vu32 i;

    RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOC, ENABLE);

    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_6;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
    GPIO_Init(GPIOC, &GPIO_InitStructure);

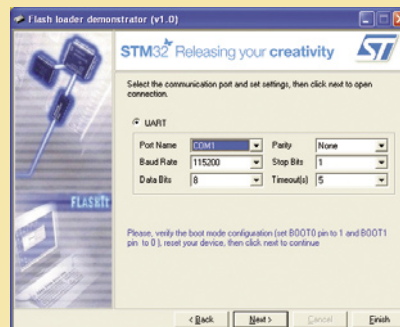
    do{
        GPIO_SetBits(GPIOC, GPIO_Pin_6);
        for(i = 0; i < 0x10000; i++);
        GPIO_ResetBits(GPIOC, GPIO_Pin_6);
        for(i = 0; i < 0x50000; i++);
    }while(1);
    return 0;
}
```

Program rozpoczyna się od włączenia taktowania portu GPIOC. Następnie do odpowiednich pól struktury *GPIO_InitStructure* zapisywane są dane konfigurujące wyprowadzenie PC6 w tryb wyjścia typu *push-pull*. Maksymalna prędkość pracy portu nie ma w tym przykładzie większego znaczenia, jednak ustawiona została na wartość maksymalną – 50 MHz. Następnie w pętli nieskończonej wykonywane jest naprzemienne ustawianie i zerowanie stanu na wyprowadzeniu. Zarówno po ustawieniu, jak i wyzerowaniu wyprowadzenia program wytraca czas w pętli, co pozwoli na zaobserwowanie zmian stanu diody LED. Po wybraniu z menu *Project* polecenia *Build Project* następuje kompilacja programu oraz konsolidacja z plikami biblioteki, następnie jest generowany plik w formacie *Intel Hex*, który zawiera kod gotowy do wgrania do pamięci mikrokontrolera.

Programowanie pamięci mikrokontrolerów STM32

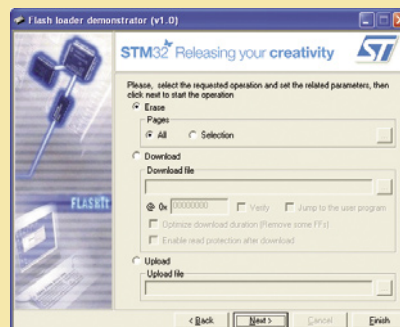
Pamięć Flash mikrokontrolera STM32F103VBT6 może być programowana poprzez łącze szeregowo z wykorzystaniem zapisanego w pamięci systemowej programu bootloadera. Płytkę STM3210B-

EVAL należy połączyć kablem *null-modem* z komputerem PC. Po uruchomieniu programu *Flash loader demonstrator* ukáže się okno powitalne. Następne okno służy do określenia parametrów komunikacji i wyboru numeru portu RS-232 (rys. 8). Aby



Rys. 8.

nawiązanie komunikacji z programem bootloadera było możliwe, należy ustawić odpowiednią kombinację na wejściach BOOT0 i BOOT1 (odpowiednio 1 i 0) oraz wyzerować mikrokontroler. Po naciśnięciu przycisku *Next* w przypadku poprawnego nawiązania komunikacji ukáže się okno z wyszczególnionymi sektorami pamięci Flash. W polu *Target* należy wybrać odpowiedni typ mikrokontrolera i klikając przycisk *Next* przejść do następnego okna, w którym należy wybrać rodzaj wykonywanej na pamięci operacji oraz ustawić powiązane z nią parametry (rys. 9). Ostatnie okno prezentuje postęp



Rys. 9.

procesu programowania pamięci Flash. Jeśli operacja programowania przebiegła pomyślnie, pozostaje tylko przestawić zworkę BOOT0 w pozycję 0 i wyzerować mikrokontroler. Dioda LED powinna emitować cyklicznie krótkie błysnięcia.