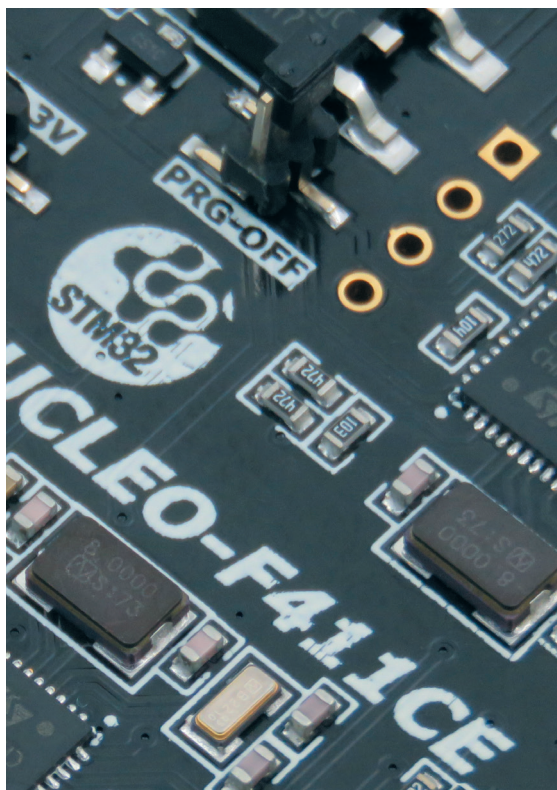
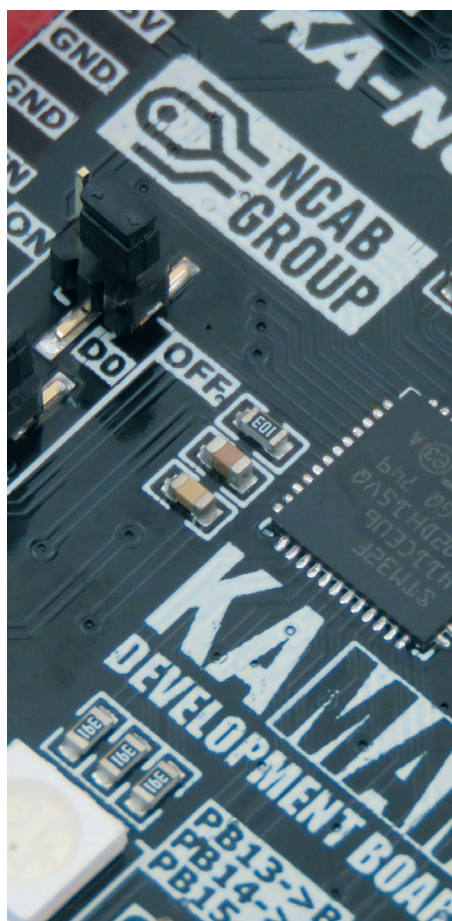


1



Wstęp

Swoją pierwszą styczność z układami STM32 miałem w trakcie kursu programowania tych układów zorganizowanego przez koło naukowe Microsystem Oriented Society na Politechnice Wrocławskiej. Podczas kursu przygotowałem kilka małych projektów i umieściłem je w serwisie GitHub; repozytorium to następnie rozbudowywałem po kursie, dodając kolejne projekty. Ponieważ wszystkie przykłady wykonywane były na płytkach Kamami KA-NUCLEO, które KN MOS otrzymało od producenta i rozdało między uczestników szkolenia, pracownik firmy Kamami p. Piotr Zbysiński skontaktował się ze mną i zapytał, czy nie chciałbym opisać kilku projektów na łamach miesięcznika „Elektronika Praktyczna”. Zgodziłem się – czego efektem była seria wydawnicza – „Kurs programowania układów STM32”. Następnie poproszony zostałem o przygotowanie książki na bazie tych artykułów – tej właśnie książki.

Opisuję w niej sporo tych samych projektów – część z nich nieco rozbudowałem, część całkiem zmieniłem, a część pozostawiłem w niemal niezmienionej formie. Względem serii wydawniczej zmieniło się także środowisko pracy – w artykułach wykorzystywane było starsze, tworzone przez społeczność – System Workbench for STM32, obecnie jest to Atollic TrueSTUDIO, wykupione przez producenta układów STM32. W dużej mierze jednak książka bazuje na wspomnianej serii wydawniczej.

1.1. Do kogo kieruję niniejszą książkę?

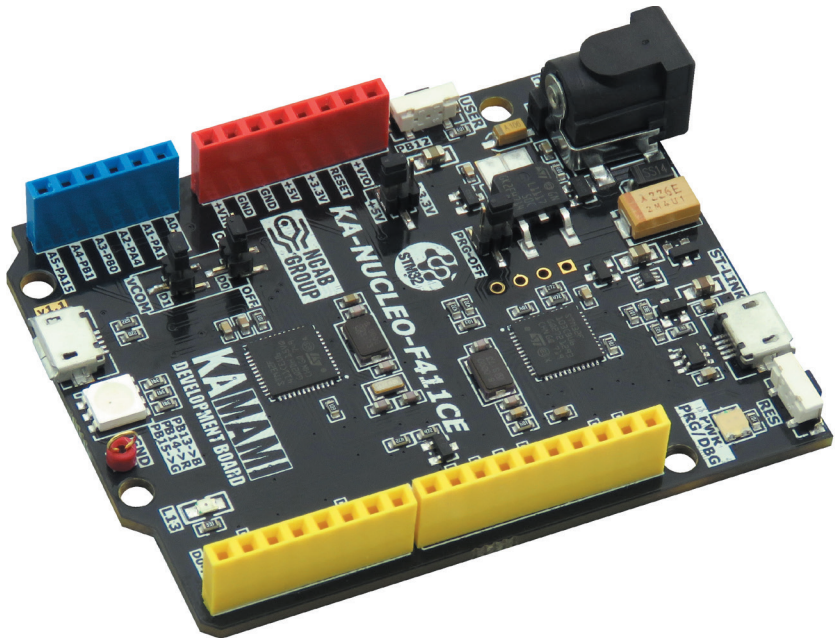
Książkę tę kieruję przede wszystkim do osób takich jak ja w momencie pierwszego spotkania z mikrokontrolerami STM32 – do osób, które mają już jakieś podstawowe doświadczenie z popularnymi płytkami Arduino lub jakimikolwiek mikrokontrolerami – czy to na zajęciach w szkole, na uczelni, czy prywatnie w domu. Zakładam też, że Czytelnik zna podstawy programowania w języku C/C++ i rozumie działanie prostego kodu – zna pojęcia takie jak: zmienne, instrukcje warunkowe czy pętle.

1.2. Czego będzie można się z niej nauczyć?

W kolejnych rozdziałach przedstawię narzędzia oraz biblioteki pozwalające na programowanie układów z rodziny STM32. Podczas tworzenia małych projektów zapoznamy się z zagadnieniami takimi jak: obsługa pinów GPIO, liczniki, przerywania, generowanie sygnału PWM, interfejsy komunikacyjne UART, SPI czy I²C, obsługa konwertera analogowo-cyfrowego i wielowątkowość oferowana przez system operacyjny FreeRTOS. Uruchomimy także urządzenia zewnętrzne takie jak: moduł GPS, kolorowy dotykowy wyświetlacz LCD, czujniki temperatury, ciśnienia i wilgotności czy paski adresowalnych diod LED oraz połączymy się z siecią Wi-Fi i komputerem w standardzie Bluetooth.

1.3. Płytką rozwojową Kamami KA-NUCLEO

Wszystkie opisywane w książce przykłady tworzone były pod kątem płytki rozwojowej Kamami KA-NUCLEO-F411CE (**rysunek 1.1**). Choć większość spośród nich będzie możliwa do wykonania na dowolnej innej płytce z układem STM32, zachęcam do zakupu tej konkretnej płytki, ponieważ przenoszenie projektów może być chwilami problematyczne.



Rys. 1.1. Płytki KA-NUCLEO-F411CE

KA-NUCLEO-F411CE występuje w dwóch, nieznacznie różniących się wersjach i każdy przykład jest kompatybilny z obiema wersjami. Obie wersje płytki bazują na układzie STM32F411CEU6 i są wyposażone w interfejs USB 2.0 (do dyspozycji użytkownika), pięć interfejsów SPI, pięć magistral I²S, trzy magistrale I²C, trzy interfejsy USART/UART, sześć liczników 16-bitowych oraz dwa 32-bitowe. KA-NUCLEO-F411CE może być taktowany zegarem o maksymalnej częstotliwości 100 MHz. Na płytce oprócz właściwego mikrokontrolera znajduje się również prostszy układ STM32F103C8U6, pełniący funkcję programatora ST-LINK z wyjściem USB. Urządzenie ma wyprowadzenia pinów zgodne z Arduino, wbudowany przycisk do dyspozycji użytkownika oraz trójkolorową diodę LED RGB.



Dokładny opis KA-NUCLEO-F411CE, karta katalogowa oraz schematy są do pobrania na stronie producenta:

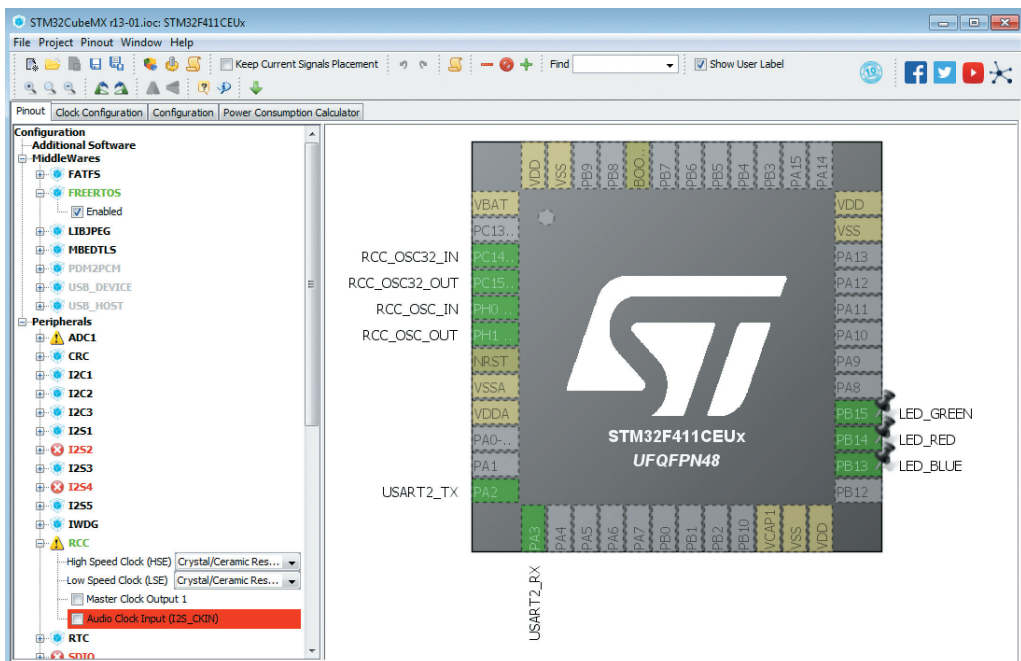
<https://kamami.pl/stm32-nucleo/557904-ka-nucleo-f411ce-plytka-rozwojowa-z-mikrokontrolerem-stm32f411ce.html>.

STM32 to rodzina 32-bitowych mikrokontrolerów wyprodukowanych przez firmę STMicroelectronics. Układy te bazują na rdzeniu obliczeniowym z serii ARM Cortex-M. W zależności od wersji cechują się małym poborem mocy lub wysokimi osiąganiami i możliwościami. Układy mogą być taktowane zegarem o maksymalnej częstotliwości od 32 do 216 MHz, odpowiednio w wersji STM32L0 oraz STM32F7. Wersje STM32F7, STM32F4 i STM32F3, oparte na rdzeniach Cortex-M4F, mają

wsparcie dla obliczeń zmiennoprzecinkowych (FPU) oraz cyfrowego przetwarzania sygnałów (DSP). Mikrokontrolery te mają również wiele interfejsów SPI, UART, I2C, I2S, a w niektórych wersjach również wbudowany kontroler USB 1.1 lub 2.0 i Ethernet.

1.4. W jakim programie będziemy pracować?

Producent układów STM32 – firma STMicroelectronics – wydał zestaw bibliotek – frameworków, noszący nazwę *Hardware Abstraction Layer*, umożliwiający programowanie tych mikrokontrolerów w bardzo prosty, wysokopoziomowy, sposób. Dzięki temu nie ma konieczności zagłębiania się w obszerną specyfikację układu ani rdzenia ARM. Wraz z bibliotekami HAL producent dostarcza również program STM32CubeMX (**rysunek 1.2**). Jest to graficzny generator konfiguracji mikrokontrolera, pozwalający na skonfigurowanie wszystkich wyprowadzeń, interfejsów, liczników oraz taktowania całego układu, a następnie na wygenerowanie gotowego projektu dla różnych środowisk IDE. Wszystko to sprawia, że programowanie tak zaawansowanego mikrokontrolera jest niemal równie proste i intuicyjne, jak programowanie platformy Arduino, niemniej mamy tutaj znacznie większe możliwości.



Rys. 1.2. Program STM32CubeMX