

# Czy nadszedł czas na zmianę?

## Migracja z platform 8-bitowych do 32-bitowych

*Od około 8 lat wspieramy specjalistów za pośrednictwem naszej społeczności element14. W tym czasie byliśmy świadkami wielu zmian na rynku mikrokontrolerów, w tym rozwoju technologii oraz udostępnienia architektury 32-bitowej, która umożliwi inżynierom tworzenie jeszcze lepszych projektów.*

Spośród miliardów systemów wbudowanych używanych codziennie na świecie duża część jest oparta na mikrokontrolerach z rdzeniami 8- lub 16-bitowymi. Wielu specjalistów uczyło się konstruowania i programowania urządzeń w oparciu o te mikrokontrolery i pomimo upływu lat nadal cenią sobie ich niskie koszty, niewielkie zużycie energii oraz prostą architekturę. Dzięki temu „stare” układy nadal utrzymywały swoją popularność nawet po tym, jak do powszechnego użytku wprowadzono układy 32-bitowe. Właśnie dlatego, z rosnącą popularnością rozwiązań 32-bitowych rośnie też potrzeba edukowania o różnicach, korzyściach i szansach oferowanych przez nowszą architekturę.

Przy każdym nowym projekcie pojawia się pytanie – z czego zrezygnuję wybierając rozwiązania 8-bitowe? Co naprawdę oferuje mi architektura 32-bitowa? Czy w związku z tym, że współcześnie najlepiej sprzedają się mikrokontrolery 32-bitowe przyszedł czas, aby je poznać, by nie zostać w tyle?

Na szczęście jedną z najbardziej ekscytujących cech branży elektronicznej jest to, że nigdy nie przestaje się ona rozwijać. Takie rdzenie, jak ARM Cortex-M0+, w który są wyposażone mikrokontrolery firm Freescale, Atmel, NXP i innych, sprawiły, że procesory 32-bitowe dotarły do punktu, w którym dorównują silnym stronom starszych mikroprocesorów 8- i 16-bitowych, a dodatkowo mają wiele korzyści, które przemawiają za zmianą. Rdzeń Cortex-M0+ został zaprojektowany specjalnie z myślą o tym, aby przejście na architekturę 32-bitową było łatwe i wygodne, a Twoje nowe urządzenia dysponowały większą mocą obliczeniową.

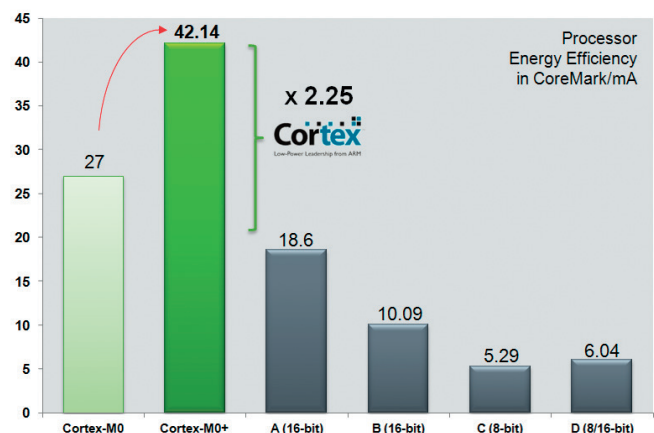
### Dlaczego 32-bity?

Jeśli zapytasz świeżego studenta politechniki, dlaczego warto przeprowadzić migrację z systemu 8-bitowego do 32-bitowego, prawdopodobnie otrzymasz odpowiedź – „4 razy więcej bitów, a więcej to zawsze lepiej!”. W rzeczywistości nie jest to jednak takie oczywiste. Istnieje jednak wiele przekonujących powodów świadczących o tym, że stosowanie mikrokontrolerów 32-bitowych ma sens nawet w tradycyjnych projektach dotychczas realizowanych na mikrokontrolerach 8- i 16-bitowych.

- **Lepsza wydajność.** Zmiana rdzenia na 32-bitowy może dać od 2- do 40-krotne zwiększenie wydajności na każdy MHz w porównaniu ze stosowanymi współcześnie architektuрами 8- i 16-bitowymi. Zyskuje się w ten sposób o wiele szybsze, 32-bitowe wykonywanie operacji matematycznych, w tym 32-bitową operację mnożenia wykonywaną w jednym cyklu zegara, a ponadto – również wykonywane w jednym cyklu – operacje I/O do programowego sterowania transmisją szeregową i ogólnie programowej

emulacji protokołów. Taka moc obliczeniowa otwiera zupełnie nowe możliwości umożliwiając używanie stosów programowych (USB, Bluetooth itp.), systemów operacyjnych czasu rzeczywistego, zaawansowanych interfejsów użytkownika i wielu innych technologii, a przy tym ma wystarczający zapas mocy obliczeniowej na realizowanie zasadniczej funkcjonalności aplikacji.

- **Oszczędność energii.** Rdzeń Cortex-M0+ jest również niezwykle energooszczędny osiągając blisko dwukrotnie lepszy wynik w teście Coremark/mA, niż jego konkurenci 16-bitowi. W połączeniu z silniejszym rdzeniem Cortex-M0+, system wbudowany może zużywać mniej energii, kończąc zadania znacznie szybciej i wcześniej przechodząc z powrotem w tryb uśpienia. Nawet w trybie uśpienia rdzeń osiąga imponujące wyniki, jeśli chodzi o zużycie energii. Na przykład w mikrokontrolerach z rodziny Freescale Kinetis L w najgłębszym, dziewiątym trybie oszczędzania energii, jest pobierany prąd o natężeniu poniżej 500 nA. Dzięki inteligentnie zaprojektowanym modułom peryferyjnym projektant systemu ma jeszcze większe możliwości, ponieważ niektóre operacje są wykonywane bez konieczności aktywowania rdzenia, co prowadzi do dodatkowych oszczędności energii.



- **Bardziej zwięzły kod.** Może się to wydawać nienaturalne, że korzystanie z rdzenia 32-bitowego oznacza mniejszą wielkość kodu wynikowego, jednak rdzeń Cortex-M0+ korzysta z instrukcji Thumb-2, z których wiele zajmuje tylko 16 bitów w pamięci Flash. Należy też pamiętać, że wiele instrukcji dla rdzeni 8-bitowych jest w rzeczywistości dłuższe niż 8 bitów. Ponadto, w zależności od aplikacji, kilka bajtów instrukcji 8-bitowych można zastąpić jedną instrukcją 32-bitową, podobnie jak w pokazanym poniżej przykładzie mnożenia 16-bitowego.

W rezultacie zwięzłość kodu jest wyraźnie poprawiona dzięki przejściu na architekturę 32-bitową z instrukcjami Thumb-2.

**Skalowalność.** Już nigdy więcej assembler związany z jedną, konkretną architekturą od konkretnego dostawcy mikrokontrolerów 8-bitowych nie będzie Cię ograniczał w sytuacji, gdy okaże się, że musisz wprowadzić zmiany w projekcie lub potrzebujesz większej wydajności. Tworzenie aplikacji dla procesorów 32-bitowych w C lub C++ zamiast w asem-

**Comparing 16-bit multiply operations across processor architectures**

48 Cycle, 48 Byte Code Size	8 Cycle, 8 Byte Code Size		1 Cycle, 2 Byte Code Size
8-bit example	16-bit example		ARM Cortex-M
MOV A, XL ; 2 bytes	MUL AB; 1 byte	MOV R4,#0130h	MULS r0,r1,r0
MOV B, YL ; 3 bytes	ADD A, R1; 1 byte	MOV R5,#0138h	
MUL AB; 1 byte	MOV R1, A; 1 byte	MOV SumLo,R6	
MOV R0, A; 1 byte	MOV A, B; 2 bytes	MOV SumHi,R7	
MOV R1, B; 3 bytes	ADDC A, R2; 1 byte	(Operands are moved to and from a memory mapped hardware multiply unit)	
MOV A, XL; 2 bytes	MOV R2, A; 1 byte		
MOV B, YH; 3 bytes	MOV A, XH; 2 bytes		
MUL AB; 1 byte	MOV B, YH; 3 bytes		
ADD A, R1; 1 byte	MUL AB; 1 byte		
MOV R1, A; 1 byte	ADD A, R2; 1 byte		
MOV A, B; 2 bytes	MOV R2, A; 1 byte		
ADDC A, #0; 2 bytes	MOV A, B; 2 bytes		
MOV R2, A; 1 byte	ADDC A, #0; 2 bytes		
MOV A, XH; 2 bytes	MOV R3, A; 1 byte		
MOV B, YL; 3 bytes			

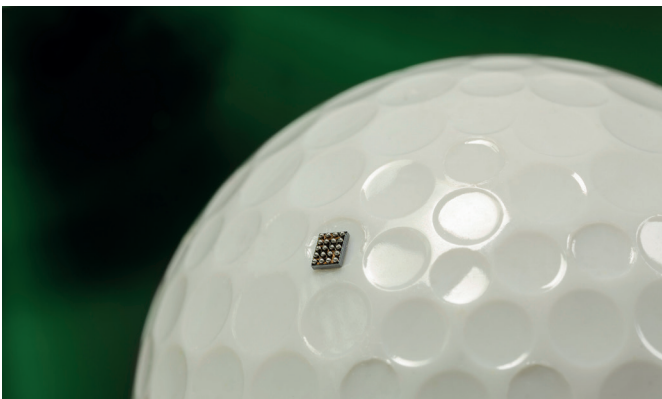
N.B. The Cortex-M multiply in fact performs a 32-bit multiply, here we assume r0 and r1 contain 16-bit data.

blerze skraca jej czas opracowywania i debugowania, a także upraszcza przenoszenie na nowsze procesory. Wzrastająca popularność mikrokontrolerów 32-bitowych ARM na całym świecie prowadzi do tego, że ekosystem oprogramowania dla tej platformy już jest ogromny i nadal rośnie.

## Ale...

Pomyślisz pewnie, że wszystko to brzmi świetnie, ale potrzebujesz mniejszej obudowy, niskiej ceny oraz łatwości użycia właściwej mikrokontrolerom 8- i 16-bitowym. Na szczęście dzisiejsze mikrokontrolery 32-bitowe to zupełnie nowy rozdział.

- **Wielkość.** Dzięki wymiarom obudowy np. w mikrokontrolerach z rodziny Kinetis KL03 wynoszącym zaledwie 1,6 mm×2,0 mm, te 32-bitowe mikrokontrolery z rdzeniem ARM mogą być stosowane w miejscach o ograniczonej ilości przestrzeni.



- **Cena.** Przy cenie układu wynoszącej od 49 centów, mikrokontrolery 32-bitowe sprawdzają się w zastosowaniach, w których dotychczas były za drogie. Ponadto, większa zwężość kodu oznacza mniejsze zapotrzebowanie na pamięć Flash.
- **Złożoność.** Niespodziewanie, zastosowanie architektury 32-bitowej może przynieść korzyści w niektórych zastosowaniach. Dzięki 32-bitowej przestrzeni adresowej nie ma potrzeby stronicowania, a dostęp do komórek pamięci odbywa się w sposób bezpośredni. Architektura rdzenia Cortex-M0+ obejmuje rozbudowany kontroler przerwań, co upraszcza ich obsługę. Bufor śledzenia zapewnia łatwiejsze debugowanie, a zaledwie 56 instrukcji sprawia, że dla osób korzystających z assemblera „przeskok” na architekturę 32-bitową jest w rzeczywistości bardzo łatwy.

## Projektowanie z wykorzystaniem 32-bitowych procesorów Kinetis

Jedną z głównych zalet korzystania z procesora 32-bitowego z rdzeniem ARM jest ogromna ilość dostępnego oprogramowania i narzędzi, w tym mnóstwo kompilatorów i debuggerów. Wielokrotne wykorzystywanie opro-

gramowania i sprzętu jest tym łatwiejsze, że projekty mogą być skalowane np. z rozwiązań wykorzystujących rdzeń Cortex-M0+ do „mocniejszych”, z rdzeniem np. Cortex-M4. Dzięki kompatybilności procesorów z rodziny Freescale Kinetis na poziomie wyprowadzeń poruszanie się po krzywej wydajności i ilości pamięci Flash jest teraz o wiele łatwiejsze.

Firma Freescale jeszcze bardziej ułatwia przejście na technologię 32-bitową dzięki oprogramowaniu narzędziowemu do modelowania *Processor Expert* oraz systemowi czasu rzeczywistego *MQX-Lite*. Ułatwiają one i przyspieszają powstawanie nowych projektów 32-bitowych.

*Processor Expert* to program narzędziowy z graficznym interfejsem użytkownika służący do generowania kodu i tworzenia własnych sterowników, znacznie skracający czas opracowywania aplikacji.

System *RTOS MQX-Lite* to uproszczona wersja popularnego rozwiązania MQX, która została zaprojektowana specjalnie na potrzeby mikrokontrolerów dysponujących mniej niż 4 kB pamięci. Sterowniki dla tego systemu RTOS tworzy się za pomocą dostarczanego z nim oprogramowania *Processor Expert*, a ponieważ jest on podzbiorem „dużego” systemu MQX, względne przeniesienie aplikacji na mocniejszą platformę będzie łatwe.

Co ważne, dla procesorów z obsługą USB jest dostępny również stos USB.

Chcesz sprawdzić, co oferuje architektura 32-bitowa, ale nie wiesz, od czego zacząć? Na szczęście na rynku jest wiele zestawów uruchomieniowych stanowiących świetną platformę do oceny rozwiązań 32-bitowych, a wiele z nich dostępnych jest w niskiej cenie. Bardzo dobrym wyborem w celu zapoznania się z rozwiązaniami 32-bitowymi będzie platforma FRDM-KL05Z. Dodatkowo, zapewniona zgodność na poziomie wyprowadzeń z 8-bitowymi mikrokontrolerami S08 firmy Freescale. FRDM-KL26Z ma interfejs USB i bardziej zaawansowane moduły peryferyjne, a FRDM-KL46Z oferuje jeszcze więcej dzięki obsłudze segmentowego wyświetlacza LCD. Jeśli dla Twoich potrzeb najważniejsza jest obsługa wejść i wyjść o poziomach TTL oraz wydajność prądowa, dobrym wyborem będzie FRDM-KE02.

Każda z tych 4 płytek wielkości pudełka miętówek zawiera procesor z rdzeniem Cortex-M0+ i ma dołączone narzędzie do modelowania oprogramowania *Processor Expert*, system *MQX-Lite*, stopy USB i wiele więcej. Ponadto, dzięki wbudowanemu sprzętowemu debuggerowi *OpenSDA*, debugowanie kodu i programowanie pamięci Flash oraz komunikację szeregową można zrealizować wykorzystując do tego jedynie zwykły kabel USB.

Dodatkowo, wszystkie te płytki i generalnie wszystkie zestawy uruchomieniowe Freescale Freedom są zgodne z platformą Arduino i zapewniają prosty dostęp do cyfrowych i analogowych pinów I/O. Zestaw *MEMS Sensors Evaluation Kit* to świetny przykład możliwości, ponieważ stanowi on połączenie platformy deweloperskiej *Freescale Freedom FRDM-KL25Z* z modułem *MEMS Sensor Evaluation*, zapewniając mikrokontrolerowi Kinetis L dostęp do zestawu najnowocześniejszych czujników.

Podczas całego procesu nauki ważne jest, aby utrzymywać kontakt z innymi inżynierami, i właśnie tutaj z pomocą przychodzi społeczność element14. Często rozwiązanie Twojego problemu ktoś już wcześniej znalazł i właśnie do tego najlepiej przydają się społeczności online.

## 32-bitowa przyszłość

Technologia nigdy nie przestaje iść naprzód, a trend stosowania mikrokontrolerów 32-bitowych w coraz większej liczbie systemów wbudowanych będzie dominował. W element14 wspieramy inżynierów na każdym etapie procesu projektowania, od zestawów uruchomieniowych, po prototypy i produkcję. Chcemy, aby czuli się oni pewnie przy migracji do nowszej architektury 32-bitowej. Lepsza wydajność, imponująca oszczędność energii oraz większa zwężość kodu sprawiają, że nadszedł czas na wykonanie pierwszego kroku w stronę lepszej, przyszłościowej platformy 32-bitowej.

**Ankur Tomar, Farnell element14  
Anthony Huereca, Freescale**