



Infineon XC800

51-ka z koproprocesorem do napędów

Współcześnie trudno jest spotkać napęd elektryczny bez mikrokontrolera. Nie dziwi więc fakt, że rosnące wymagania odnośnie jakości i sprawności napędów będą związane nie tylko z mechaniką, lecz również będą dotyczyć sterującej nią elektroniki, a zwłaszcza mikrokontrolera. Ten z jednej strony musi kontrolować aplikację w czasie rzeczywistym, a z drugiej ani on sam, ani też narzędzia uruchomieniowe nie mogą kosztować krocie, bo to zniechęci potencjalnego nabywcę i inżyniera konstruktora. Naprzeciw takim wymaganiom wychodzi oferta mikrokontrolerów XC800 firmy Infineon.

Firma Infineon opracowując swoje najnowsze mikrokontrolery wychodzi przede wszystkim naprzeciw wymaganiom aplikacji związanych z napędami. Od takiego przeznaczenia nie odbiega też seria XC800, oferująca użytkownikom peryferia umożliwiające sterowanie silnikami różnego typu, z zastosowaniem różnych metody ich kontroli. Są to: komutacja bloków dla bezszczotkowych silników prądu stałego (BLDC) z czujnikiem Halla lub bez niego, kontrola zorientowana połowo (FOC, *Field-Oriented-Control*) do sterowania synchronicznymi silnikami z magnesami stałymi, FOC oraz PFC dla złożonych, gotowych napędów.

FOC jest techniką sterowania silnikami elektrycznymi gwarantującą wysoką spraw-

ność ich pracy przy różnych prędkościach obrotowych. Stosując tę metodę można osiągnąć sprawność rzędu 95%, a tym samym ograniczyć pobór mocy, zmniejszyć hałas. Jednocześnie uzyskiwane są mniejszy poziom zakłóceń i bardzo wysoki moment obrotowy. Porównując metodę FOC z innymi, konkurencyjnymi algorytmami warto jest powiedzieć, że angażuje ona tylko 58% mocy obliczeniowej 8-bitowej jednostki CPU, pozostawiając resztę na realizację innych, ważnych funkcji, jak chociażby sterowanie parametrami i realizacja funkcji komunikacyjnych.

Metodę FOC zaimplementowano w 8-bitowych mikrokontrolerach CX878, 886 i 888 przez połączenie zmodyfikowanego rdzenia

8051 i komputera wektorowego z dodatkowym koproprocesorem matematycznym wykonującym operacje na liczbach 16-bitowych. Komputer wektorowy jest zbudowany przez dwie równoległe jednostki: MDU (16-bitowy układ mnożąco-dzielący) i CORDIC (16-bitowy koprocesor dedykowany do przekształceń wektorowych oraz obliczeń kątów).

Architektura

Pewnych Czytelników, którzy już dawno temu okrzyknęli koniec rdzeni 8-bitowych, a zwłaszcza leciwej 51-ki zdziwi fakt, że ta ma się całkiem dobrze i zapewne będzie oferowana jeszcze przez wiele lat. W ubiegłym roku ogólny udział mikrokontrolerów z rdzeniem Intel 8051 w wolumenie sprzedaży na całym świecie wyniósł aż 17%.

Infineon również postawił na to sprawdzone w wielu aplikacjach rozwiązanie. Mikrokontrolery XC800 wyposażono w rdzeń 8051, który jednak różni się od standardowego większą szybkością pracy. Otóż 1 cykl maszynowy zawiera 2 cykle zegarowe, a nie 12 jak w oryginale. Maksymalna częstotliwość zegara, to minimalnie 24 MHz (dla serii XC864 jest to 26,67 MHz, natomiast XC878 – 27 MHz), co daje cykl maszynowy równy ok. 83 ns.

Konstruktor, który zdecydował się zastosować XC800 w swoim urządzeniu z całą

Tab. 1. Wykaz mikrokontrolerów z serii XC800													
Typ (SAX-)	F _{max} [MHz]	Pamięć programu [kB]	Typ pamięci	SRAM + Cache [kB]	Koprocesor	Cyfrowe I/O	Sterowane czasowo I/O (PWM, CAPCOM)	Zewn. szyny danych i adr.	Liczba kanałów A/C	Liczba węzłów CAN	Int. Szeregowe	Zakres temp.	Obudowa
Seria XC866													
XC864-1FR	26,27	4	Flash	0,75	-	13	7	nie	4	0	1×UART, 1×SCC	FK	TSSOP-20
XC866-1FR	26,27	4	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC	FK,A	TSSOP-38
XC866L-1FR	26,27	4	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC, LIN BSL	FK,A	TSSOP-38
XC866-2FR	26,27	8	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC	FK,A	TSSOP-38
XC866L-2FR	26,27	8	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC, LIN BSL	FK,A	TSSOP-38
XC866-4FR	26,27	16	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC	FK,A	TSSOP-38
XC866L-4FR	26,27	16	Flash	0,75	-	27	7	nie	8	0	1×UART, 1×SCC, LIN BSL	FK,A	TSSOP-38
XC866-2RR	26,27	8	ROM	0,75	-	27	7	nie	8	0	1×UART, 1×SCC	FK,A	TSSOP-38
XC866-4RR	26,27	16	ROM	0,75	-	27	7	nie	8	0	1×UART, 1×SCC	FK,A	TSSOP-38
Seria XC886/888LIM													
XC886-6FF	24	24	Flash	1,75	-	34	7	nie	8	0	2×UART, 1×SCC	FK	TQFP-48
XC886-8FF	24	32	Flash	1,75	-	34	7	nie	8	0	2×UART, 1×SCC	FK	TQFP-48
XC886LIM-6FF	24	24	Flash	1,75	VC	34	7	nie	8	0	2×UART, 1×SCC, LIN BSL	FK	TQFP-48
XC886LIM-8FF	24	32	Flash	1,75	VC	34	7	nie	8	0	2×UART, 1×SCC, LIN BSL	FK	TQFP-48
XC888-6FF	24	24	Flash	1,75	-	48	7	nie	8	0	2×UART, 1×SCC	FK	TQFP-64
XC888-8FF	24	32	Flash	1,75	-	48	7	nie	8	0	2×UART, 1×SCC	FK	TQFP-64
XC888LIM-6FF	24	24	Flash	1,75	VC	48	7	nie	8	0	2×UART, 1×SCC, LIN BSL	FK	TQFP-64
XC888LIM-8FF	24	32	Flash	1,75	VC	48	7	nie	8	0	2×UART, 1×SCC, LIN BSL	FK	TQFP-64
Seria XC886/888CLIM													
XC886C-6FF	24	24	Flash	1,75	-	34	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-48
XC886CM-6FF	24	24	Flash	1,75	VC	34	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-48
XC886C-8FF	24	24	Flash	1,75	-	34	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-48
XC886CM-8FF	24	32	Flash	1,75	VC	34	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-48
XC886CLIM-8FF	24	32	Flash	1,75	VC	34	7	nie	8	2	2×UART, 1×SCC, LIN BSL	FK	TQFP-48
XC888C-6FF	24	24	Flash	1,75	-	48	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-64
XC888CM-6FF	24	24	Flash	1,75	VC	48	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-64
XC888C-8FF	24	24	Flash	1,75	-	48	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-64
XC888CM-8FF	24	32	Flash	1,75	VC	48	7	nie	8	2	2×UART, 1×SCC	FK	TQFP-64
XC888CLIM-8FF	24	32	Flash	1,75	VC	48	7	nie	8	2	2×UART, 1×SCC, LIN BSL	FK	TQFP-64
XC886CM-8RF	24	32	ROM	1,75	VC	34	7	nie	8	2	2×UART, 1×SCC	K	TQFP-48
XC888CM-8RF	24	32	ROM	1,75	VC	48	7	nie	8	2	2×UART, 1×SCC	K	TQFP-64
Seria XC878													
XC878-13FF	27	52	Flash	3	-	48	13	tak	8	-	2×UART, 1×SCC	FX	TQFP-64
XC878M-13FF	27	52	Flash	3	eVC	48	13	tak	8	-	2×UART, 1×SCC	F	TQFP-64
XC878CM-13FF	27	52	Flash	3	eVC	48	13	tak	8	2	2×UART, 1×SCC	FX	TQFP-64
XC878-16FF	27	64	Flash	3	-	48	13	tak	8	-	2×UART, 1×SCC	FX	TQFP-64

Tab. 1. c.d.

Typ (SAX-)	F _{max} [MHz]	Pamięć programu [kB]	Typ pamięci	SRAM + Cache [kB]	Koprocesor	Cyfrowe I/O	Sterowane czasowo I/O (PWM, CAPCOM)	Zewn. szyny danych i adr.	Liczba kanałów A/C	Liczba węzłów CAN	Int. Szeregowe	Zakres temp.	Obudowa
XC878-16FF	27	64	Flash	3	eVC	48	13	tak	8	-	2×UART, 1×SCC	F	TQFP-64
XC878CM-16FF	27	64	Flash	3	eVC	48	13	tak	8	2	2×UART, 1×SCC	F,X	TQFP-64
XC878L-13FF	27	52	Flash	3	-	48	13	tak	8	-	2×UART, 1×SCC, LIN	X	TQFP-64
XC878C-13FF	27	52	Flash	3	-	48	13	tak	8	2	2×UART, 1×SCC	X	TQFP-64
XC878L-16FF	27	64	Flash	3	-	48	13	tak	8	-	2×UART, 1×SCC, LIN	X	TQFP-64
XC878C-16FF	27	64	Flash	3	-	48	13	tak	8	2	2×UART, 1×SCC	X	TQFP-64

VC – komputer wektorowy, UART – uniwersalny interfejs asynchroniczny, SSC – szeregowy interfejs synchroniczny
 Zakresy temperatur: F=-40...85°C, K=-40...125°C, A=-40...140°C, X=-40...105°C

Tab. 2. Zestawy ewaluacyjne dla mikrokontrolerów XC800

Typ zestawu	Typ mikrokontrolera	Opis
KIT_XC864_EK_V1	SAF-XC864-1FRI	Zestaw podst. XC864
KIT_XC878_EK_V1	SAX-SX878CM-16FFA	Zestaw podst. XC878
KIT_XC800_USCALE_USB	XC866 XC886 XC888	Zestaw z interfejsem USM XC800
KIT_XC888_SK	SAK-XC886CM	Zestaw startowy XC888
KIT_XC866_SK	SAK-XC866	Zestaw startowy XC866
KIT_XC866_EK	SAK-XC866	Zestaw podst.XC866
KIT_AK_XC800_FOC_V1	XC886/888	Zestaw aplikacyjny do implementacji FOC (silnik PMSM/24 V)
KIT_AK_XC866_BLDC	XC866	Zestaw demonstracyjny do implementacji algorytmu komutacji bloków
KIT_AK_3PHASE_DRIVE_V1	XC886	Zestaw ewaluacyjny o budowy driverów silników PMSM i indukcyjnych (110/220 V)
KIT_AK_XC886_LIN_STEPPER	XC866	Zestaw do budowy driver silnika krokowego z XC866
KIT_AK_DaveDrive_V2	XC886	Zestaw do budowy drivera niskonapięciowego silnika PMSM z oprogramowaniem DAVE Drive
KIT_AK_FOCDRIVE_V1	XC878, XE164F	Zestaw do budowy wysokonapięciowego driver FOC
KIT_AK_2MOTORDRIVE_V1	XC878, XE164F	Zestaw do budowy wysokonapięciowych driverów FOC i PFC

gotowych i sprawdzonych przez Infineon rozwiązań można szybko wybrać metodę odpowiednią dla silnika danego typu i sprawdzić ją w praktyce. Listę dostępnych płytek ewaluacyjnych umieszczono w **tab. 2**. Każdy zestaw zawiera preinstalowane, gotowe do uruchomienia przykłady oraz instrukcje ilustrujące krok po kroku co i dlaczego jest wykonywane. Firma przyjęła bardzo dobrą strategię nauki przez przykłady.

Ciekawym narzędziem oferowanym przez Infineon jest program o nazwie DAVE (*Digital Application Virtual Engineer*) umożliwiający bardzo szybkie tworzenie kodu w języku C i/lub asemblerze, a to dzięki wbudowanemu kreatorom. DAVE pozwala skonfigurować układ tak, aby ten pracował zgodnie z oczekiwaniami i automatycznie wygenerować kod w języku C z funkcjami driverów dla wszystkich układów peryferyjnych i do kontroli przerwań. Wygenerowany program jest zgodny z formatem stosowanym przez popularne narzędzia softwareowe, takie jak Keil czy Tasking.

DAVE Drive ver. 2 (www.infineon.com/davedrive/) jest generatorem gotowego kodu źródłowego służącego do sterowania większością typów silników dostępnych na rynku. Z jednej strony, programista ma wgląd do kompletnego kodu i możliwość wykonania niezbędnych zmian, natomiast z drugiej, jest on zwalniany z konieczności wpiśnięcia setek linii programu. Dodatkowo należy wspomnieć, że ze sterowaniem silnikami wiąże się pewne wiedza na ten temat. Trudno jest znaleźć ekspertów, którzy mieli-

by odpowiednie umiejętności w programowaniu w asemblerze oraz doskonałą znajomość zagadnień związanych z kinematyką. DAVE Drive generuje zoptymalizowany kod metody FOC np. dla XC886 lub XC878 używający ich komputera wektorowego.

DAVE Drive wspiera silniki typu BLDC i PMSM oraz różne algorytmy kontroli: komutację bloków z sensorami Halla i FOC. Konstruktorzy kontrolerów napędów mogą skupić się na oprogramowaniu aplikacji, takim jak np. programowanie wentylatorów czy pomp, bez zagłębiania się w szczegóły kontroli. W porównaniu z innymi, konkurencyjnymi narzędziami, DAVE Drive może wygenerować odpowiedni kod niezależnie od posiadanych bibliotek. Dodatkowo, zapewnia on generowanie kodu dla silników (napędów) zdefiniowanych przez użytkownika. Zmiana modelu silnika wiąże się tylko z wyborem odpowiedniego modelu z biblioteki lub określeniem jego nowych parametrów. DAVE Drive po jednym kliknięciu myszką usłuźnie wygeneruje kod, który (zapewne po niewielkich zmianach) można od razu zastosować we własnej aplikacji.

Narzędzie doceni każdy, kto kiedykolwiek zajmował się napędami elektrycznymi. Mając gotowe algorytmy sterowania i będąc zwolnionym z konieczności mozolnego ich parametryzowania, projektant czy programista może zaoszczędzić nawet rok pracy!

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