Mikrokontrolery STM32 – który do czego?

Benchmarki porównujące wydajność w różnych aplikacjach

Duża liczba dostępnych modeli mikrokontrolerów STM32 oraz wyposażanie ich w różne rdzenie powoduje, że konstruktorzy nie zawsze wiedzą, jaki model wybrać do swojej aplikacji. Postanowiliśmy wykonać serię autorskich testów, których celem było zmierzenie i porównanie wydajności różnych mikrokontrolerów w typowych zastosowaniach, jak również sprawdzenie wpływu różnych ustawień na czas wykonania programu. W artykule przedstawiamy wyniki uzyskane dla STM32 z rdzeniami Cortex-M0, Cortex-M0+, dwóch wersji Cortex-M3, a także Cortex-M4 i Cortex-M7.

Jak w każdym teście lub badaniu ogromnie ważną rolę odgrywa używana metoda. Bez jej opisu informacje są niepełne i właściwie bezwartościowe, więc poświęćmy jej kilka słów.

Zastosowana metoda

Ponieważ zależało nam na podejściu maksymalnie praktycznym, mierzyliśmy czas wykonania zestawu testów reprezentujących typowe operacje wykonywane przez mikrokontrolery. W testach wykorzystano następujące typy mikrokontrolerów:

- STM32F030R8 (Cortex-M0),
- STM32L053R8 (Cortex-M0+),
- STM32F103RB (Cortex-M3),
- STM32L152RE (Cortex-M3),
- STM32F446RE (Cortex-M4),
- STM32F746ZG (Cortex-M7).

Kolejne testy obejmowały (na początku podano skrótowe oznaczenia stosowane dalej w tabelach):

- **int** – proste operacje arytmetyczne i logiczne na liczbach stałopozycyjnych: dodawanie, odejmowanie, mnożenie, dzielenie, modulo, suma logiczna, iloczyn logiczny, różnica symetryczna.
- **float** – proste operacje na liczbach zmiennopozycyjnych pojedynczej precyzji: dodawanie, odejmowanie, mnożenie, dzielenie, modulo, wartość bezwzględna, pierwiastek kwadratowy, podnoszenie do potęgi.

- **double** – proste operacje na liczbach zmiennopozycyjnych podwójnej precyzji – dokładnie takie same operacje jak dla liczb zmiennopozycyjnych pojedynczej precyzji.
- **matrix fl, matrix fix** – operacje na macierzach liczb odpowiednio zmiennopozycyjnych pojedynczej precyzji i stałopozycyjnych w formacie Q1.31. Wykorzystano funkcje z biblioteki CMSIS: dodawanie, odejmowanie, mnożenie, transpozycja i odwrotność (ostatnie tylko dla liczb zmiennopozycyjnych, ze względu na brak odpowiedniej funkcji operującej na liczbach stałopozycyjnych w bibliotece CMSIS).
- **fft fl, fft fix** – prosta i odwrotna, szybka transformata Fouriera w wersji operującej na liczbach odpowiednio zmiennopozycyjnych pojedynczej precyzji i stałopozycyjnych w formacie Q1.31. Test został przeprowadzony dla mikrokontrolerów, które miały wystarczająco dużą pamięć, aby można było wgrać program wykorzystujący funkcje z biblioteki CMSIS, tj. STM32L152RE, STM32F446RE, STM32F746ZG.
- **foc fl, foc fix** – algorytmy wykorzystywane przy Field Oriented Control: PID, transformata Clarka, transformata Parka (prosta i odwrotna). Wykorzystano funkcje z biblioteki CMSIS w wersji operującej na liczbach odpowiednio zmiennopozycyjnych pojedynczej precyzji i stałopozycyjnych w formacie Q1.31.
- **sm** – skoki i rozejścia warunkowe w Finite State Machine. Jako automat do testów wykorzystany został prosty hex loader.

W każdym przypadku częstotliwość taktowania mikrokontrolera została ustawiona na największą możliwą dla danego mikrokontrolera. Sygnał zegarowy mikrokontrolera pochodził z PLL. Tam, gdzie było to możliwe, jako źródło częstotliwości wejściowej PLL wybrano HSE. W przypadku mikrokontrolera STM32F030R8 używana płytka Nucleo była w konfiguracji „HSE not used", w związku z czym do taktowania PLL wykorzystano HSI.

Do pomiaru czasu wykorzystane zostały sprzętowe timery. Pomiary wykonano z rozdzielczością 1 µs dla mikrokontrolerów: STM32F103RB, STM32F446RE, STM32F746ZG i z rozdzielczością 10 µs dla STM32F030R8, STM32L053R8, STM32L152RE. Wyniki zostały następnie znormalizowane. Żeby nie zakłócać testów przez obsługę przerwań, pomiary polegały na odczytaniu rejestru CNT timera przed i po teście, a następnie wyliczeniu na tej podstawie czasu wykonania testu. Wyniki zostały następnie sprawdzone i ewentualnie skorygowane – w przypadku wyników większych niż $2^{16} - 1$ – na podstawie wyników pomiarów przeprowadzonych z 10 razy mniejszą rozdzielczością. Po zakończonych testach wyniki były przesyłane do komputera PC przez UART.

Dla wszystkich mikrokontrolerów sprawdzono wpływ następujących opcji:

- Poziom optymalizacji: –o2 vs –o3 – różnice w otrzymanych wynikach z reguły były niewielkie (w większości przypadków nie przekraczały ±1%), dlatego dalej przedstawiono wyniki dla poziomu optymalizacji –o3, a krótkie podsumowanie różnic między –o2 i –o3 umieszczono przy opisie wyników dla każdego z mikrokontrolerów.
- Wykorzystanie bądź nie sprzętowej jednostki zmiennopozycyjnej (FPU) tam, gdzie jest ona dostępna (tj. dla

1

czas wykonania [us]

microlib	PREFTEN	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1	34110	50840	133210	47120	13340	89850	36400	24180
0	0	48310	66250	171460	59900	17730	120400	48320	31950
1	1	123330	63970	220330	71600	13480	114810	306520	23900
1	0	158450	82890	286110	94440	17730	152760	389560	31950
min		34110	50840	133210	47120	13340	89850	36400	23900

2

microlib / std lib

PREFTEN	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
1	3,615655	1,258261	1,654005	1,519525	1,010495	1,277796	8,420879	0,98842
0	3,279859	1,25117	1,668669	1,576628	1	1,268771	8,062086	1

3

PREFTEN = 0 / 1

microlib	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1,4163	1,303108	1,287141	1,271222	1,329085	1,340011	1,327473	1,32134
1	1,284764	1,295764	1,298552	1,318994	1,315282	1,330546	1,270912	1,33682

4

microlib	PREFTEN		double / float			foc fl / foc fix	
0	1		2,620181			2,468407	
0	0		2,588075			2,491722	
1	1		3,444271			0,374560	
1	0		3,451683			0,392135	

5

czas wykonania [us]

microlib	DISAB_BUF	PREFTEN	PRE_READ	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	0	1	1	50730	70710	186670	64900	19130	127530	51970	35090
0	0	1	0	50730	70900	186400	64790	19140	127610	51970	35050
0	0	0	1	65460	87830	232100	79460	24760	165420	67630	44600
0	0	0	0	66710	88070	232120	79070	24760	165630	67810	44520
0	1	0	0	66710	89210	234620	80170	25140	166040	68660	44720
1	0	1	1	150620	88200	305880	100240	19340	162420	429060	34740
1	0	1	0	150610	88380	306000	100230	19340	162510	429100	34830
1	0	0	1	179970	106610	381750	125710	24760	209830	529530	44450
1	0	0	0	181230	106890	382060	125710	24750	210050	529720	44520
1	1	0	0	201090	110110	394030	127980	25130	210550	544160	44720
min				50730	70710	186400	64790	19130	127530	51970	34740

6

microlib / std lib

DISAB_BUF	PREFTEN	PRE_READ	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1	1	2,969052	1,247348	1,638614	1,54453	1,010978	1,273583	8,255917	0,990026
0	1	0	2,968855	1,246544	1,641631	1,546998	1,010449	1,27349	8,256687	0,993723
0	0	1	2,749313	1,213822	1,644765	1,582054	1	1,268468	7,829809	0,996637
0	0	0	2,716684	1,213694	1,645959	1,589857	0,999596	1,268188	7,811827	1
1	0	0	3,014391	1,234279	1,679439	1,596358	0,999602	1,268068	7,92543	1

7

DIAB_BUF = 1 / 0

microlib	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1	1,012944	1,01077	1,013912	1,015347	1,002475	1,012535	1,004492
1	1,109585	1,030124	1,03133	1,018057	1,015354	1,00238	1,02726	1,004492

8

PREFTEN = 0 / 1

microlib	DISAB_BU	PRE_READ	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	0	1	1,290361	1,242116	1,243371	1,224345	1,294302	1,297107	1,301328	1,271017
0	0	0	1,315001	1,242172	1,245279	1,220404	1,293626	1,297939	1,304791	1,270185
1	0	1	1,194861	1,20873	1,248038	1,25409	1,280248	1,291898	1,234163	1,279505
1	0	0	1,203307	1,209437	1,248562	1,254215	1,279731	1,292536	1,234491	1,278208

9

PRE_READ = 0 / 1

microlib	DISAB_BU	PREFTEN	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	0	1	1	1,002687	0,998554	0,998305	1,000523	1,000627	1	0,99886
0	0	1	1,019096	1,002733	1,000086	0,995092	1	1,001269	1,002662	0,998206
0	0	0	0,999934	1,002041	1,000392	0,9999	1	1,000554	1,000093	1,002591
0	0	0	1,007001	1,002626	1,000812	1	0,999596	1,001048	1,000359	1,001575

10

microlib	DISAB_BUF	PREFTEN	PRE_READ		double / float			foc fl / foc fix	
0	0	1	1		2,639938			2,453916	
0	0	1	0		2,629055			2,455455	
0	0	0	1		2,642605			2,445956	
0	0	0	0		2,635631			2,442560	
0	1	0	0		2,629974			2,418293	
1	0	1	1		3,468027			0,378548	
1	0	1	0		3,462322			0,378723	
1	0	0	1		3,580809			0,396257	
1	0	0	0		3,574329			0,396530	
1	1	0	0		3,578512			0,386927	

11

czas wykonania [us]

microlib	PREFTEN	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1	3892	30831	59646	29961	3672	51598	7305	14279
0	0	4797	44182	80394	41695	5313	80018	10569	18490
1	1	3963	35546	115359	45718	3776	62610	7444	15141
1	0	4799	48457	153618	62128	5312	87285	10569	18490
min		3892	30831	59646	29961	3672	51598	7305	14279

12

microlib / std lib

PREFTEN	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
1	1,018243	1,15293	1,934061	1,525917	1,028322	1,213419	1,019028	1,060368
0	1,000417	1,096759	1,910814	1,490059	0,999812	1,090817	1	1

STM32F446RE, STM32F746ZG – w obu przypadkach jest to jednostka zmiennopozycyjna pojedynczej precyzji).

- Wykorzystanie biblioteki Microlib zoptymalizowanej na niewielki rozmiar kodu względem biblioteki standardowej.

Dla konkretnych mikrokontrolerów sprawdzono wpływ specyficznych opcji mających przyspieszyć wykonanie programów, takich jak: prefetch, buforowanie, kieszenie.

Porównano również czas wykonania testów *fft* i *foc* w wersji operującej na liczb zmiennopozycyjnych i stałopozycyjnych oraz wyniki testów *double* i *float* (te same obliczenia na liczbach podwójnej i pojedynczej precyzji).

Wpływ dostępnych opcji na czas wykonywania programu

1. STM32F030R8 (Cortex-M0)

W przypadku STM32F030R8 ustawienie bitu PREFTEN w rejestrze FLASH_ACR włącza mechanizm *prefetch*, czyli spekulatywne pobieranie kolejnego słowa po tym, które ostatnio odczytano, jako instrukcję.

Wyniki: 1, 2, 3, 4

Wnioski:

- **–o2 vs –o3**
- Różnice między poziomem optymalizacji –o2 a –o3 praktycznie nie występowały (nie przekraczały ±1%).
- **microlib vs std lib**
- Rozmiar kodu wynikowego był o ok. ¼ większy dla biblioteki standardowej. W przypadku automatu i obliczeń na macierzach liczb stałopozycyjnych różnice nie występowały lub były bardzo małe (do ok. 1%). Dla obliczeń zmiennopozycyjnych w przypadku korzystania z Microlib czas wykonania był ok. 1,2…1,7 razy dłuższy.

Znaczne różnice wystąpiły w przypadku prostych obliczeń stałopozycyjnych, gdzie czas wykonania wersji wykorzystującej Microlib okazał się ok. 3,5 razy dłuższy. Jeszcze większe w przypadku FOC operującego na liczbach stałopozycyjnych, gdzie użycie Microlib spowodowało ponad 8-krotny wzrost czasu wykonania.

- **PREFTEN**

Włączenie mechanizmu prefetch spowodowało zmniejszenie czasu wykonania ok. 1,3…1,4 razy. Największy wpływ miało na operacje na macierzach liczb stałopozycyjnych, FOC operujące na liczbach zmiennopozycyjnych, automat oraz, w przypadku korzystania z biblioteki standardowej, na proste operacje stałopozycyjne i FOC operujące na liczbach stałopozycyjnych.

- **double vs float**

Czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2,5 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 3,5 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

- **floating point vs fixed point**

Czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 2,5 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 2,5 razy krótszy, jeżeli wykorzystywana była biblioteka Microlib.

2. STM32L053R8 (Cortex-M0+)

W przypadku STM32L053R8 ustawienie bitu DISAB_BUF w rejestrze FLASH_ACR wyłącza bufory używane jako kieszenie dla odczytów z NVM (*Non-Volatile Memory*). Jeśli DISAB_BUF = 0, to ustawienie w rejestrze FLASH_ACR bitu PREFTEN włącza mechanizm prefetch, czyli spekulatywne pobieranie kolejnego słowa po tym, które ostatnio odczytano, jako instrukcję, a ustawienie bitu PRE_READ włącza mechanizm pre-read, czyli spekulatywne pobieranie kolejnego słowa danych.

Wyniki: 5, 6, 7, 8, 9, 10

Wnioski:

- **–o2 vs –o3**

Dla automatu czas wykonania w przypadku poziomu optymalizacji o2 był ok. 2…3% krótszy. W pozostałych wypadkach różnice praktycznie nie występowały.

- **microlib vs std lib**

Rozmiar kodu wynikowego był o ok. 25% większy dla biblioteki standardowej. W przypadku automatu i obliczeń na macierzach liczb stałopozycyjnych różnice nie występowały lub były bardzo małe (do ok. 1%). Dla obliczeń zmiennopozycyjnych w przypadku korzystania z Microlib czas wykonania był ok. 1,2…1,6 razy dłuższy.

Znaczne różnice wystąpiły w przypadku prostych obliczeń stałopozycyjnych, gdzie czas wykonania wersji wykorzystującej Microlib okazał się ok. 3 razy dłuższy. Jeszcze większe w przypadku FOC operującego na liczbach stałopozycyjnych, gdzie użycie Microlib spowodowało ok. 8-krotny wzrost czasu wykonania.

- **DISAB_BUF**

Wyłączenie buforów spowodowało wydłużenie czasu wykonania o maksymalnie kilka procent. Wyłączenie buforów praktycznie nie miało wpływu w przypadku automatu, FOC operującego na liczbach zmiennopozycyjnych i prostych obliczeń stałopozycyjnych, jeżeli wykorzystywana była biblioteka standardowa. Natomiast jeżeli wykorzystywana była biblioteka Microlib, to czas wykonania dla prostych obliczeń stałopozycyjnych był ok. 1,1 razy dłuższy przy wyłączonych buforach.

- **PREFTEN**

Włączenie mechanizmu prefetch spowodowało zmniejszenie czasu wykonania o ok. 10…30%. Największy wpływ miało na operacje na macierzach liczb stałopozycyjnych, FOC operujące na liczbach zmiennopozycyjnych oraz, w przypadku korzystania z biblioteki standardowej, na proste operacje stałopozycyjne i FOC operujące na liczbach stałopozycyjnych.

- **PRE_READ**

Bardzo małe różnice.

- **double vs float**

Czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2,5 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 3,5 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

- **floating point vs fixed point**

Czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 2,5 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 2,5 razy krótszy, jeżeli wykorzystywana była biblioteka Microlib.

3. STM32F103RB (Cortex-M3)

W przypadku STM32F103RB ustawienie bitu PREFTEN w rejestrze FLASH_ACR włącza mechanizm prefetch, czyli spekulatywne pobieranie kolejnego słowa po tym, które ostatnio odczytano jako instrukcję.

Wyniki: 11, 12, 13, 14

Wnioski:

- **–o2 vs –o3**

Dla prostych obliczeń stałopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był do ok. 7% krótszy. W przypadku korzystania z biblioteki standardowej i przy włączonym prefetch dla prostych obliczeń zmiennopozycyjnych – ok. 6% krótszy, dla obliczeń na macierzach liczb zmiennopozycyjnych – ok. 9% krótszy, natomiast dla FOC operującego na liczbach zmiennopozycyjnych – ok. 2% dłuższy. Przy włączonym prefetch dla automatu – ok. 2–7% krótszy.

W pozostałych przypadkach różnice były niewielkie.

- **microlib vs std lib**

Rozmiar kodu wynikowego był o ok. 1/6 większy dla biblioteki standardowej. Różnice czasów wykonania w zależności od użycia

13

PREFTEN = 0 / 1

microlib	int	float	double	matrix fl	matrix fix	foc fl	foc fix	sm
0	1,232528	1,433038	1,347852	1,391642	1,446895	1,550797	1,446817	1,294909
1	1,210951	1,363219	1,331652	1,35894	1,40678	1,394106	1,419801	1,221188

14

microlib	PREFTEN		double / float			foc fl / foc fix	
0	1		1,934611			7,063381	
0	0		1,819610			7,571010	
1	1		3,245344			8,410801	
1	0		3,170192			8,258586	

biblioteki standardowej lub Microlib były średnio mniejsze niż dla Cortex-M0(+). Różnice były większe przy włączonym prefetch. Dla operacji na liczbach stałopozycyjnych i automatu różnice były niewielkie.

W przypadku korzystania z Microlib czas wykonania był ok. 2 razy dłuższy dla prostych obliczeń zmiennopozycyjnych podwójnej precyzji, ok. 1,5 razy dłuższy dla obliczeń na macierzach liczb zmiennopozycyjnych i ok. 1,1–1,2 razy dłuższy dla pozostałych obliczeń zmiennopozycyjnych.

- **PREFTEN**

Włączenie mechanizmu prefetch spowodowało zmniejszenie czasu wykonania ok. 1,2…1,5 razy. Największe znaczenie miało dla FOC, obliczeń na macierzach (szczególnie zmiennopozycyjnych) i prostych obliczeń na liczbach zmiennopozycyjnych pojedynczej precyzji. Większe różnice występowały, jeżeli wykorzystywana była biblioteka standardowa niż dla Microlib.

- **double vs float**

Czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2 razy dłuższy, jeżeli wykorzystywana jest biblioteka standardowa i ok. 3,2 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

- **floating point vs fixed point**

Czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 6…7 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ponad 8 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib. Dla FFT czasy wykonania różniły się odpowiednio ok. 7…8 razy i ok. 10 razy.

4. STM32L152RE (Cortex-M3)

W przypadku STM32L152RE ustawienie bitu PREFTEN w rejestrze FLASH_ACR włącza mechanizm prefetch, czyli spekulatywne pobieranie kolejnego słowa po tym, które ostatnio odczytano, jako instrukcję.

Wyniki: 15, 16, 17, 18

Wnioski:

- **–o2 vs –o3**

Dla prostych obliczeń stałopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był do ok. 2–3% krótszy, natomiast w przypadku korzystania z biblioteki standardowej dla prostych obliczeń zmiennopozycyjnych i obliczeń na macierzach liczb zmiennopozycyjnych – do ok. 4% dłuższy. W pozostałych przypadkach różnice były niewielkie.

- **micro lib vs std lib**

Rozmiar kodu wynikowego był o ok. 1/6 większy dla biblioteki standardowej. Różnice czasów wykonania w zależności od użycia

15

czas wykonania [us]

microlib	PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
0	1	8770	49390	99230	48370	7320	162590	21910	85990	13720	28120
0	0	8920	55770	110330	55250	7890	187580	23260	101480	14250	29540
1	1	8770	66440	215250	85280	7390	215640	21760	114780	13880	28520
1	0	8920	70670	227640	90730	7960	228560	22900	122250	14470	29430
min		8770	49390	99230	48370	7320	162590	21760	85990	13720	28120

16

microlib / std lib

PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	1	1,345212	2,169203	1,763076	1,009563	1,326281	0,993154	1,334806	1,011662	1,014225
0	1	1,267169	2,063265	1,642172	1,008872	1,218467	0,984523	1,204671	1,015439	0,996276

17

PREFTEN = 0 / 1

microlib	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
0	1,017104	1,129176	1,111861	1,142237	1,077869	1,153699	1,061616	1,180137	1,03863	1,050498
1	1,017104	1,063666	1,057561	1,063907	1,077131	1,059915	1,05239	1,065081	1,042507	1,031907

18

microlib	PREFTEN		double / float			fft fl / foc fix	foc fl / foc fix	
0	1		2,009111			7,420812	6,267493	
0	0		1,978304			8,064488	7,121404	
1	1		3,239765			9,909926	8,269452	
1	0		3,221169			9,980786	8,448514	

19

czas wykonania [us]

hw fpu	microlib	PREFTEN	ICEN	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	0	1	1	1	1334	1949	23132	1313	891	1867	2715	917	1695	4246
1	0	1	1	0	1334	2101	24537	1313	891	1904	2756	1111	2028	4524
1	0	1	0	1	1500	2148	30924	1621	1231	2363	3272	1106	1916	6244
1	0	1	0	0	1500	2259	32633	1629	1231	2412	3321	1338	2339	6659
1	0	0	1	1	1334	2593	26044	1419	991	1910	2725	917	1695	4246
1	0	0	1	0	1334	2663	27268	1426	991	1947	2765	1111	2029	4525
1	0	0	0	1	1973	2788	39380	2120	1655	3411	4512	1417	2678	7836
1	0	0	0	0	1973	2857	40659	2128	1656	3454	4564	1627	3095	8194
1	1	1	1	1	**1334**	**1669**	**36768**	**1313**	**891**	**1819**	**2642**	**917**	**1695**	**4246**
1	1	1	1	0	**1334**	**2090**	**36884**	**1312**	**891**	**1898**	**2655**	**1167**	**2000**	**4524**
1	1	1	0	1	**1612**	**2232**	**53022**	**1625**	**1220**	**2294**	**3230**	**1073**	**1916**	**6465**
1	1	1	0	0	**1612**	**2343**	**53139**	**1630**	**1220**	**2401**	**3247**	**1456**	**2339**	**6881**
1	1	0	1	1	**1334**	**2523**	**38730**	**1410**	**892**	**1862**	**2652**	**917**	**1695**	**4246**
1	1	0	1	0	**1333**	**2593**	**38833**	**1415**	**892**	**1940**	**2664**	**1167**	**2001**	**4525**
1	1	0	0	1	**1751**	**2827**	**64230**	**2216**	**1619**	**3337**	**4446**	**1389**	**2728**	**7795**
1	1	0	0	0	**1751**	**2896**	**64331**	**2222**	**1619**	**3430**	**4459**	**1678**	**3078**	**8103**
0	0	1	1	0	1334	8731	17304	7627	907	24937	2642	13554	1694	4246
0	0	1	0	1	1334	8773	18647	8452	907	25094	2656	13915	2000	4524
0	0	1	0	0	1529	12955	24104	11731	1217	40912	3247	21409	1945	6315
0	0	1	1	1	1528	13008	25592	12704	1217	41029	3260	21853	2389	6703
0	0	0	1	0	1334	9718	19374	7771	908	25762	2651	13983	1696	4246
0	0	0	0	1	1335	9760	20725	8595	909	25880	2665	14366	2001	4524
0	0	0	0	0	2084	16364	30193	14710	1687	54155	4489	28419	2750	8249
0	0	0	1	1	2084	16409	31551	15865	1686	54272	4505	28819	3128	8637
0	1	1	1	1	**1333**	**10938**	**35552**	**13483**	**907**	**34701**	**2639**	**18450**	**1695**	**4246**
0	1	1	1	0	**1334**	**10979**	**35655**	**13485**	**907**	**34903**	**2648**	**18955**	**2000**	**4524**
0	1	1	0	1	**1500**	**18009**	**54015**	**20664**	**1231**	**53329**	**3199**	**27723**	**1928**	**6244**
0	1	1	0	0	**1501**	**18054**	**54118**	**20664**	**1231**	**53451**	**3212**	**28238**	**2328**	**6659**
0	1	0	1	1	**1334**	**11429**	**37419**	**13844**	**908**	**35563**	**2648**	**18833**	**1695**	**4246**
0	1	0	1	0	**1334**	**11471**	**37518**	**13845**	**908**	**35683**	**2657**	**19299**	**2001**	**4525**
0	1	0	0	1	**1973**	**20979**	**66474**	**25549**	**1666**	**64939**	**4454**	**34009**	**2678**	**7836**
0	1	0	0	0	**1974**	**21023**	**66574**	**25548**	**1666**	**65059**	**4466**	**34486**	**3022**	**8195**
min					1333	1669	17304	1312	891	1819	2639	917	1694	4246

20

fpu: off / single precision

microlib	PREFTEN	ICEN	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
0	1	1	1	1	4,479733	0,748055	5,808835	1,017957	13,35672	0,973112	14,78081	0,99941	1
0	1	1	0	1	4,175631	0,759954	6,437167	1,017957	13,17962	0,963716	12,52475	0,986193	1
0	1	0	1	1,019333	6,031192	0,779459	7,236891	0,988627	17,31358	0,992359	19,35714	1,015136	1,011371
0	1	0	0	1,018667	5,7583	0,784237	7,798649	0,988627	17,01036	0,981632	16,33259	1,021377	1,006608
0	0	1	1	1	3,747782	0,743895	5,476392	0,916246	13,48796	0,972844	15,24864	1,00059	1
0	0	1	0	1,00075	3,665039	0,760048	6,027349	0,917255	13,29224	0,963834	12,93069	0,9862	0,999779
0	0	0	1	1,05626	5,86944	0,766709	6,938679	1,019335	15,87658	0,994902	20,05575	1,026886	1,052705
0	0	0	0	1,05626	5,743437	0,775991	7,455357	1,018116	15,7128	0,987073	17,71297	1,010662	1,054064
1	1	1	1	0,99925	6,553625	0,966928	10,26885	1,017957	19,07697	0,998864	20,11996	1	1
1	1	1	0	1	5,25311	0,966679	10,2782	1,017957	18,38936	0,997363	16,2425	1	1
1	1	0	1	0,930521	8,068548	1,018728	12,71631	1,009016	23,24717	0,990402	25,83691	1,006263	0,965816
1	1	0	0	0,931141	7,705506	1,018423	12,6773	1,009016	22,26197	0,989221	19,39423	0,995297	0,967737
1	0	1	1	1	4,529925	0,96615	9,81844	1,017937	19,09936	0,998492	20,53762	1	1
1	0	1	0	1,00075	4,423833	0,966137	9,784452	1,017937	18,3933	0,997372	16,53728	1	1
1	0	0	1	1,126785	7,420941	1,034937	11,52933	1,02903	19,46029	1,001799	24,48452	0,981672	1,00526
1	0	0	0	1,127356	7,259323	1,034867	11,49775	1,02903	18,96764	1,00157	20,55185	0,981806	1,011354

biblioteki standardowej lub Microlib były średnio mniejsze niż dla Cortex-M0(+). Dla operacji na liczbach stałopozycyjnych i automatu różnice były bardzo niewielkie.

W wypadku korzystania z Microlib czas wykonania był ponad 2 razy dłuższy dla prostych obliczeń zmiennopozycyjnych podwójnej precyzji, ok. 1,7 razy dłuższy dla obliczeń na macierzach liczb zmiennopozycyjnych i ok. 1,3 razy dłuższy dla pozostałych obliczeń zmiennopozycyjnych.

- **PREFTEN**

Włączenie mechanizmu prefetch spowodowało zmniejszenie czasu wykonania maksymalnie ok. 1,2 razy. Największe znaczenie miało dla obliczeń na liczbach zmiennopozycyjnych pojedynczej i podwójnej precyzji, jeżeli wykorzystywana była biblioteka standardowa.

- **double vs float**

Czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 3,2 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

- **floating point vs fixed point**

Czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 6…7 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ponad 8 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib. Dla FFT czasy wykonania różniły się odpowiednio ok. 7…8 razy i ok. 10 razy.

5. STM32F446RE (Cortex-M4)

W przypadku STM32F446RE ustawienie w rejestrze FLASH_ACR bitu ICEN włącza kieszeń instrukcji o pojemności 64 linii po 128 bitów każda, natomiast ustawienie bitu DCEN włącza kieszeń danych o pojemności 8 linii po 128 bitów każda. Ustawienie PREFTEN w rejestrze FLASH_ACR włącza mechanizm prefetch, czyli spekulatywne pobieranie kolejnej 128-bitowej linii zawierającej instrukcje.

Wyniki: 19, 20

Wnioski:

- **–o2 vs –o3**

W większości wypadków różnice były nieznaczne (poniżej 1%). Dla prostych obliczeń stałopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był do ok. 3% krótszy, jeśli włączona była kieszeń instrukcji oraz do ok. 20% krótszy, jeśli kieszeń instrukcji była wyłączona.

Dla operacji zmiennopozycyjnych pojedynczej precyzji w przypadku używania biblioteki standardowej przy wyłączonej kieszeni instrukcji i FPU czas wykonania dla poziomu optymalizacji –o2 był od kilku do kilkunastu procent krótszy.

W przypadku używania FPU, dla FOC operującego na liczbach zmiennopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był od kilku do ok. 10% dłuższy.

Dla prostych obliczeń stałopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był do ok. 2…3% krótszy, natomiast w przypadku korzystania z biblioteki standardowej dla prostych obliczeń zmiennopozycyjnych i obliczeń na macierzach liczb zmiennopozycyjnych – do ok. 4% dłuższy. W pozostałych wypadkach różnice były niewielkie.

- **FPU**

Wykorzystanie FPU przyspieszyło obliczenia na liczbach zmiennopozycyjnych pojedynczej precyzji ok. 3,5…20 razy dla biblioteki standardowej i ok. 4,5…25 razy dla biblioteki Microlib. Użycie wsparcia sprzętowego dla obliczeń zmiennopozycyjnych było najbardziej korzystne w przypadku FOC i FFT, gdzie przyspieszenie wynosiło 15…20 razy dla biblioteki standardowej i 20…25 dla biblioteki Microlib. Trochę mniejsze różnice występowały w przypadku obliczeń na macierzach (6…8 razy dla biblioteki standardowej i 10…13 dla biblioteki Microlib) i prostych obliczeń (3,5…6 razy dla biblioteki standardowej i 4,5…8 dla biblioteki Microlib).

Obliczenia na liczbach zmiennopozycyjnych podwójnej precyzji w przypadku korzystania z FPU wykonywały się o ok. 30% wolniej dla biblioteki standardowej i ok. 4% wolniej dla biblioteki Microlib i włączonej kieszeni instrukcji.

Wyniki cd.: 21, 22, 23

Wnioski cd.:

- **micro lib vs std lib**

Rozmiar kodu wynikowego był o ok. 1/6 większy dla biblioteki standardowej. Dla biblioteki Microlib czas wykonania dla prostych obliczeń zmiennopozycyjnych podwójnej precyzji był ok. 1,5 razy dłuższy w przypadku używania FPU, i ok. 2 razy dłuższy, jeśli FPU nie było wykorzystywane. W przypadku niekorzystania z FPU czas wykonania dla biblioteki Microlib był ok. 1,7 razy dłuższy dla obliczeń na macierzach liczb zmiennopozycyjnych i ok. 1,2…1,4 razy dłuższy dla pozostałych obliczeń zmiennopozycyjnych.

- **PREFTEN**

Jeśli kieszeń instrukcji była włączona, to mechanizm prefetch miał niezbyt duży wpływ na czas wykonania. W większości przypadków wyniki były bardzo podobne. Różnice występowały w kilku przypadkach: przy włączonym prefetch dla obliczeń na macierzach i prostych obliczeń zmiennopozycyjnych podwójnej precyzji czas wykonania był do ok. 1,1 razy krótszy, dla prostych obliczeń

21

microlib / std lib

hw fpu	PREFTEN	ICEN	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	1	1	1	1	0,856337	1,589486	1	1	0,97429	0,973112	1	1	1
1	1	1	0	1	0,994764	1,503199	0,999238	1	0,996849	0,963353	1,050405	0,986193	1
1	1	0	1	1,074667	1,039106	1,714591	1,002468	0,991064	0,9708	0,987164	0,970163	1	1,035394
1	1	0	0	1,074667	1,037185	1,628382	1,000614	0,991064	0,995439	0,977718	1,088191	1	1,033338
1	0	1	1	1	0,973004	1,487099	0,993658	0,900101	0,974869	0,973211	1	1	1
1	0	1	0	0,99925	0,973714	1,424124	0,992286	0,900101	0,996405	0,963472	1,050405	0,9862	1
1	0	0	1	0,887481	1,013989	1,631031	1,045283	0,978248	0,978305	0,985372	0,98024	1,018671	0,994768
1	0	0	0	0,887481	1,013651	1,582208	1,044173	0,977657	0,993052	0,976994	1,031346	0,994507	0,988894
0	1	1	0	0,99925	1,252777	2,054554	1,767799	1	1,391547	0,998864	1,361222	1,00059	1
0	1	0	1	1	1,251453	1,912104	1,59548	1	1,39089	0,996988	1,362199	1	1
0	1	0	0	0,981033	1,39012	2,240914	1,761487	1,011504	1,303505	0,985217	1,294923	0,99126	0,988757
0	1	1	1	0,98233	1,387915	2,114645	1,626574	1,011504	1,302761	0,985276	1,29218	0,974466	0,993436
0	0	1	0	1	1,176065	1,931403	1,781495	1	1,380444	0,998868	1,34685	0,99941	1
0	0	0	1	0,999251	1,175307	1,810277	1,61082	0,9989	1,378787	0,996998	1,34338	1	1,000221
0	0	0	0	0,946737	1,282022	2,201636	1,736846	0,987552	1,199132	0,992203	1,196699	0,973818	0,949933
0	0	1	1	0,947217	1,281187	2,110044	1,610337	0,988138	1,198758	0,991343	1,196641	0,966113	0,948825

22

PREFTEN = 0 / 1

hw fpu	microlib	ICEN	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	0	1	1	1	1,330426	1,125886	1,080731	1,112233	1,023032	1,003683	1	1	1
1	0	1	0	1	1,267492	1,111301	1,086062	1,112233	1,022584	1,003266	1	1,000493	1,000221
1	0	0	1	1,315333	1,297952	1,273445	1,307835	1,344435	1,443504	1,378973	1,281193	1,397704	1,254965
1	0	0	0	1,315333	1,264719	1,245947	1,306323	1,345248	1,432007	1,374285	1,215994	1,323215	1,230515
1	1	1	1	1	1,511684	1,053362	1,073877	1,001122	1,023639	1,003785	1	1	1
1	1	1	0	0,99925	1,24067	1,052841	1,078506	1,001122	1,022129	1,00339	1	1,0005	1,000221
1	1	0	1	1,086228	1,266577	1,211384	1,363692	1,327049	1,454664	1,376471	1,294501	1,4238	1,205723
1	1	0	0	1,086228	1,236022	1,210617	1,36319	1,327049	1,428571	1,373268	1,152473	1,315947	1,17759
0	0	1	0	1	1,113045	1,119626	1,01888	1,001103	1,033083	1,003407	1,031651	1,001181	1
0	0	0	1	1,00075	1,112504	1,111439	1,016919	1,002205	1,031322	1,003389	1,032411	1,0005	1
0	0	0	0	1,362982	1,263142	1,252614	1,253943	1,386196	1,323695	1,382507	1,327432	1,413882	1,306255
0	0	1	1	1,363874	1,261454	1,232846	1,248819	1,385374	1,322772	1,381902	1,318766	1,309334	1,288528
0	1	1	1	1,00075	1,044889	1,052515	1,026774	1,001103	1,024841	1,00341	1,020759	1	1
0	1	1	0	1	1,044813	1,052251	1,026696	1,001103	1,022348	1,003399	1,018148	1,0005	1,000221
0	1	0	1	1,315333	1,164918	1,230658	1,236401	1,353371	1,217705	1,39231	1,226743	1,389004	1,254965
0	1	0	0	1,315123	1,164451	1,230164	1,236353	1,353371	1,217171	1,390411	1,221262	1,29811	1,230665

zmiennopozycyjnych pojedynczej precyzji – do ok. 1,3 razy krótszy (w pojedynczym przypadku 1,5 razy).

Jeśli kieszeń instrukcji była wyłączona, prefetch miało znacznie większy wpływ. Włączenie prefetch spowodowało skrócenie czasu wykonania ok. 1,1…1,5 razy (w większości przypadków ponad 1,2 razy).

- **Instruction Cache (ICEN)**

Włączenie kieszeni instrukcji spowodowało zmniejszenie czasu wykonania ok. 1,1…2 razy. Miało większe znaczenie, jeśli mechanizm prefetch był wyłączony.

W wypadku obliczeń zmiennopozycyjnych miało większe znaczenie, jeśli FPU nie było wykorzystywane.

Wyniki cd.: 24, 25

Wnioski cd.:

- **Data Cache (DCEN)**

Włączenie kieszeni danych największe znaczenie miało przy FOC – spowodowało skrócenie czasu wykonania ok. 1,2 razy dla wersji stałopozycyjnej i ok. 1,2–1,3 razy dla wersji zmiennopozycyjnej w przypadku nieużywania FPU. Miało niewielkie znaczenie dla pozostałych obliczeń stałopozycyjnych.

Włączenie kieszeni danych spowodowało skrócenie o ok. kilka procent czasu wykonania dla automatu, jak również dla prostych obliczeń zmiennopozycyjnych podwójnej precyzji w przypadku używania biblioteki standardowej, prostych obliczeń zmiennopozycyjnych pojedynczej precyzji, jeśli FPU nie było wykorzystywane, oraz dla obliczeń na macierzach liczb zmiennopozycyjnych, jeśli używana była biblioteka standardowa i FPU nie było wykorzystywane.

- **double vs float**

W wypadku nieużywania FPU czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 3 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

W przypadku używania FPU czas wykonania był odpowiednio ok. 10 i ok. 20 razy dłuższy.

- **floating point vs fixed point**

W wypadku nieużywania FPU, czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 7–10 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 10–15 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib. Dla FFT czasy wykonania różniły się odpowiednio ok. 10…12 razy i ok. 13…16 razy.

W wypadku używania FPU czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 2 razy krótszy, natomiast FFT ok. 1,4 razy krótszy.

6. STM32F746ZG (Cortex-M7)

W przypadku STM32F746ZG możliwe są różne konfiguracje z różnym ulokowaniem kodu i danych w pamięci. Dla różnych konfiguracji różne

23

ICEN = 0 / 1

hw fpu	microlib	PREFTEN	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	0	1	1	1,124438	1,102104	1,336849	1,234577	1,381594	1,265667	1,205157	1,206107	1,130383	1,470561
1	0	1	0	1,124438	1,075202	1,329951	1,24067	1,381594	1,266807	1,205007	1,20432	1,153353	1,471927
1	0	0	1	1,47901	1,075202	1,512057	1,49401	1,67003	1,785864	1,65578	1,545256	1,579941	1,845502
1	0	0	0	1,47901	1,07285	1,491088	1,492286	1,671039	1,774011	1,650633	1,464446	1,525382	1,810829
1	1	1	1	1,208396	1,337328	1,442069	1,237624	1,369248	1,261132	1,222559	1,17012	1,130383	1,52261
1	1	1	0	1,208396	1,121053	1,440706	1,242378	1,369248	1,265016	1,222976	1,247644	1,1695	1,520999
1	1	0	1	1,312594	1,120491	1,658404	1,571631	1,815022	1,792159	1,676471	1,514722	1,60944	1,835846
1	1	0	0	1,313578	1,116853	1,656606	1,570318	1,815022	1,768041	1,673799	1,437875	1,538231	1,790718
0	0	1	0	1,146177	1,483793	1,392973	1,538088	1,341786	1,640614	1,228993	1,579534	1,14817	1,487282
0	0	0	1	1,145427	1,482731	1,372446	1,503076	1,341786	1,635012	1,22741	1,570464	1,1945	1,481653
0	0	0	0	1,562219	1,683886	1,558429	1,892935	1,85793	2,102127	1,693323	2,032396	1,621462	1,94277
0	0	1	1	1,561049	1,68125	1,522364	1,845841	1,854785	2,097063	1,690432	2,006056	1,563218	1,909151
0	1	1	1	1,125281	1,646462	1,519324	1,532597	1,357222	1,536815	1,212202	1,502602	1,137463	1,470561
0	1	1	0	1,125187	1,644412	1,517824	1,532369	1,357222	1,531416	1,212991	1,489739	1,164	1,471927
0	1	0	1	1,47901	1,835594	1,776477	1,845493	1,834802	1,826027	1,682024	1,80582	1,579941	1,845502
0	1	0	0	1,47976	1,832709	1,774455	1,845287	1,834802	1,823249	1,680843	1,786932	1,510245	1,81105

24

DCEN = 0 / 1

hw fpu	microlib	PREFTEN	ICEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
1	0	1	1	1	1,077989	1,060738	1	1	1,019818	1,015101	1,211559	1,19646	1,065473
1	0	1	0	1	1,051676	1,055265	1,004935	1	1,020736	1,014976	1,209765	1,220772	1,066464
1	0	0	1	1	1,026996	1,046997	1,004933	1	1,019372	1,014679	1,211559	1,19705	1,065709
1	0	0	0	1	1,024749	1,032478	1,003774	1,000604	1,012606	1,011525	1,1482	1,155713	1,045687
1	1	1	1	1	1,252247	1,003155	0,999238	1	1,04343	1,004921	1,272628	1,179941	1,065473
1	1	1	0	1	1,049731	1,002207	1,003077	1	1,046643	1,005263	1,356943	1,220772	1,064346
1	1	0	1	0,99925	1,027745	1,002659	1,003546	1	1,04189	1,004525	1,272628	1,180531	1,065709
1	1	0	0	1	1,024407	1,001572	1,002708	1	1,027869	1,002924	1,208063	1,128299	1,039513
0	0	1	0	1	1,00481	1,077612	1,108168	1	1,006296	1,005299	1,026634	1,180638	1,065473
0	0	0	1	0,999346	1,004091	1,061732	1,082943	1	1,00286	1,004004	1,020739	1,228278	1,061441
0	0	0	0	1,00075	1,004322	1,069733	1,106035	1,001101	1,00458	1,005281	1,02739	1,179835	1,065473
0	0	1	1	1	1,00275	1,044977	1,078518	0,999407	1,00216	1,003564	1,014075	1,137455	1,047036
0	1	1	1	1,00075	1,003748	1,002897	1,000148	1	1,005821	1,00341	1,027371	1,179941	1,065473
0	1	1	0	1,000667	1,002499	1,001907	1	1	1,002288	1,004064	1,018577	1,207469	1,066464
0	1	0	1	1	1,003675	1,002646	1,000072	1	1,003374	1,003399	1,024744	1,180531	1,065709
0	1	0	0	1,000507	1,002097	1,001504	0,999961	1	1,001848	1,002694	1,014026	1,128454	1,045814

25

hw fpu	microlib	PREFTEN	ICEN	DCEN		double / float			fft fl / foc fix	fft fl / foc fix	
1	0	1	1	1		11,86865059			0,687661142	0,54100295	
1	0	1	1	0		11,67872442			0,690856313	0,547830375	
1	0	1	0	1		14,39664804			0,722188264	0,577244259	
1	0	1	0	0		14,44577247			0,726287263	0,572039333	
1	0	0	1	1		10,04396452			0,700917431	0,54100295	
1	0	0	1	0		10,23957942			0,704159132	0,547560375	
1	0	0	0	1		14,12482066			0,755984043	0,529126214	
1	0	0	0	0		14,23136157			0,756792287	0,525686591	
1	1	1	1	1		22,02995806			0,688493565	0,54100295	
1	1	1	1	0		17,64784689			0,714877589	0,5835	
1	1	1	0	1		23,75537634			0,710216718	0,560020877	
1	1	1	0	0		22,67989757			0,739451802	0,622488243	
1	1	0	1	1		15,35077289			0,702111614	0,54100295	
1	1	0	1	0		14,97608947			0,728228228	0,583208396	
1	1	0	0	1		22,72019809			0,750562303	0,509164223	
1	1	0	0	0		22,21374309			0,769230769	0,545159194	
0	0	1	1	0		1,981903562			9,438682816	8,001180638	
0	0	1	0	1		2,125498689			9,448042169	6,9575	
0	0	1	0	0		1,860594365			12,5999384	11,00719794	
0	0	1	1	1		1,967404674			12,58558282	9,147341984	
0	0	0	1	0		1,993620086			9,717842324	8,244693396	
0	0	0	0	1		2,123463115			9,711069418	7,179410295	
0	0	0	0	0		1,845086776			12,06393406	10,33418182	
0	0	0	1	1		1,922786276			12,04705882	9,213235294	
0	1	1	1	1		3,250319985			13,14929898	10,88495575	
0	1	1	1	0		3,24756353			13,18089124	9,4775	
0	1	1	0	1		2,999333667			16,67052204	14,37914938	
0	1	1	0	0		2,997562867			16,64103362	12,12972509	
0	1	0	1	1		3,274039724			13,43013595	11,11091445	
0	1	0	1	0		3,270682591			13,42980805	9,644677661	
0	1	0	0	1		3,168597169			14,57992815	12,69940254	
0	1	0	0	0		3,166722161			14,56762203	11,41164792	

są też mechanizmy wspomagające wykonanie programu. W przypadku korzystania z pamięci wewnętrznej mikrokontrolera, kod i stałe (*Read Only Memory Area*) znajdują się w pamięci Flash, a dostępy do nich mogą następować przez interfejs AHB/AXI (Flash-AXIM) albo przez interfejs ITCM – *Instruction Tightly-Coupled Memory* (Flash-ITCM), natomiast dane (*Read/Write Memory Area*) mogą być umieszczone w pamięci SRAM, do której dostęp następuje przez interfejs AHB, lub w pamięci DTCM-RAM, do której dostęp następuje przez interfejs DTCM.

26

								czas wykonania [us]									
r/o mem	r/w mem	hw fpu	microlib	ICEN	DCEN	ART	PREFT	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	1	0	1	1	x	x	996	1596	18724	650	475	1076	1398	700	983	2533
		1	0	1	0	x	x	1845	2389	23534	1511	1522	2107	3614	1362	2654	4870
		1	0	0	1	x	x	1482	2435	44407	2044	1539	2483	3208	1186	2190	9539
		1	0	0	0	x	x	2093	3180	47260	2278	1874	3072	4328	1866	3333	10098
FLASH-ITCM	SRAM1	1	0	x	1	1	1	995	1663	15579	654	458	1028	1316	695	972	2534
		1	0	x	1	1	0	995	1735	16807	666	458	1035	1318	695	973	2534
		1	0	x	1	0	1	1367	2236	30245	1199	882	1661	2169	1334	1852	6118
		1	0	x	1	0	0	1390	2317	32655	1356	1000	1989	2572	1427	2079	6622
		1	0	x	0	1	1	1714	1839	16941	1468	1497	1868	3390	918	2047	4252
		1	0	x	0	1	0	1713	1874	18037	1489	1497	1889	3394	917	2046	4252
		1	0	x	0	0	1	1744	2364	31522	1715	1625	2295	3692	1463	2676	7005
		1	0	x	0	0	0	1744	2448	33957	1768	1662	2455	3811	1519	2722	7390
FLASH-AXIM	DTCM-RAM	1	0	1	1	x	x	996	1578	18252	655	467	893	1227	700	988	2465
		1	0	1	0	x	x	1135	2200	22147	667	478	1210	1496	1191	1631	3120
		1	0	0	1	x	x	1483	2493	44668	2050	1537	2393	3115	1185	2190	9539
		1	0	0	0	x	x	1552	3103	47592	2063	1536	2710	3321	1713	2811	9902
FLASH-ITCM	DTCM-RAM	1	0	x	x	1	1	996	1663	15614	653	454	956	1247	695	991	2467
		1	0	x	x	1	0	996	1735	16843	666	455	964	1254	695	991	2467
		1	0	x	x	0	1	1366	2235	30277	1195	879	1623	2128	1334	1862	6122
		1	0	x	x	0	0	1390	2317	32658	1355	999	1961	2540	1426	2106	6626
FLASH-AXIM	SRAM1	1	1	1	1	x	x	996	1327	22901	677	470	1073	1402	855	1115	2517
		1	1	1	0	x	x	1869	1950	30718	1531	1531	2125	3633	1574	2861	4772
		1	1	0	1	x	x	1437	2397	72128	2077	1507	2464	3400	1265	2176	8789
		1	1	0	0	x	x	2178	3150	77671	2397	1886	3075	4512	1991	3172	10302
FLASH-ITCM	SRAM1	1	1	x	1	1	1	1019	1397	21371	638	457	1030	1311	695	973	2516
		1	1	x	1	1	0	1020	1614	22522	644	458	1032	1318	695	973	2516
		1	1	x	1	0	1	1436	2224	44551	1233	866	1644	2189	1302	1949	6211
		1	1	x	1	0	0	1437	2361	47498	1397	1008	1947	2590	1366	2291	6621
		1	1	x	0	1	1	1725	1567	27958	1474	1495	1883	3415	917	2047	4237
		1	1	x	0	1	0	1725	1816	28820	1475	1497	1904	3423	917	2047	4237
		1	1	x	0	0	1	1783	2348	49201	1781	1619	2244	3739	1431	2552	7115
		1	1	x	0	0	0	1784	2475	51440	1869	1631	2436	3787	1454	2769	7577
FLASH-AXIM	DTCM-RAM	1	1	1	1	x	x	997	1327	22816	670	470	893	1229	885	1167	2450
		1	1	1	0	x	x	1158	1807	24183	684	470	1218	1516	1260	1695	3104
		1	1	0	1	x	x	1437	2397	72127	2077	1502	2381	3330	1264	2176	8790
		1	1	0	0	x	x	1901	3023	73269	2087	1502	2749	3601	1843	2690	9277
FLASH-ITCM	DTCM-RAM	1	1	x	x	1	1	1019	1398	21345	637	453	953	1247	695	991	2449
		1	1	x	x	1	0	1019	1614	22507	642	454	961	1254	695	991	2449
		1	1	x	x	0	1	1436	2224	44489	1227	864	1629	2157	1301	1945	6215
		1	1	x	x	0	0	1437	2361	47475	1391	1007	1933	2557	1366	2301	6624
FLASH-AXIM	SRAM1	0	0	1	1	x	x	995	6564	13053	5138	482	20882	1405	10687	1006	2517
		0	0	1	0	x	x	1856	7503	18720	8246	1533	25013	3636	12249	2635	4772
		0	0	0	1	x	x	1436	20739	34162	16326	1520	67557	3474	35784	2180	8790
		0	0	0	0	x	x	2155	21342	36803	16920	1861	69040	4586	36686	3255	10302
FLASH-ITCM	SRAM1	0	0	x	1	1	1	995	6706	11889	4446	455	20765	1319	11525	987	2534
		0	0	x	1	1	0	996	7303	12759	4517	456	21590	1319	12033	987	2534
		0	0	x	1	0	1	1436	12938	23396	10671	887	41238	2215	21959	1917	5919
		0	0	x	1	0	0	1436	14153	25000	11560	1027	45392	2611	24529	2172	6355
		0	0	x	0	1	1	1703	6926	13359	5461	1495	24171	3388	12083	1982	4252
		0	0	x	0	1	0	1704	7476	14234	5506	1495	24689	3395	12539	1982	4253
		0	0	x	0	0	1	1770	13033	24243	11438	1650	42102	3748	22204	2634	7041
		0	0	x	0	0	0	1761	14266	26013	12230	1659	46250	3787	24778	2685	7321
FLASH-AXIM	DTCM-RAM	0	0	1	1	x	x	996	6510	12902	5168	486	20789	1222	10523	1020	2450
		0	0	1	0	x	x	1159	7278	17091	7115	486	21320	1472	11420	1761	3104
		0	0	0	1	x	x	1437	20606	34155	14449	1520	68422	3354	36066	2180	8790
		0	0	0	0	x	x	1899	21112	36841	16057	1518	69038	3547	36861	2870	9278
FLASH-ITCM	DTCM-RAM	0	0	x	x	1	1	995	6706	11891	4432	451	20715	1247	11525	1010	2467
		0	0	x	x	1	0	995	7303	12761	4508	452	21544	1255	12042	1010	2467
		0	0	x	x	0	1	1436	12939	23397	10662	884	41237	2184	21954	1885	5918
		0	0	x	x	0	0	1436	14154	25000	11550	1027	45465	2583	24524	2171	6356
FLASH-AXIM	SRAM1	0	1	1	1	x	x	996	8159	21995	9140	481	23716	1402	12841	1002	2533
		0	1	1	0	x	x	1841	9423	29926	11483	1529	30182	3633	15116	2635	4887
		0	1	0	1	x	x	1136	24565	70831	27531	1506	82019	3376	44050	2134	7960
		0	1	0	0	x	x	2218	25038	75987	28182	1903	83343	4545	44790	3144	8764
FLASH-ITCM	SRAM1	0	1	x	1	1	1	1019	7847	20952	8938	458	23991	1293	13374	996	2516
		0	1	x	1	1	0	1020	8278	21844	9095	457	24258	1300	13773	1001	2516
		0	1	x	1	0	1	1309	15282	42232	16516	851	48481	2238	26329	1949	6050
		0	1	x	1	0	0	1297	16241	45565	17683	982	52756	2645	28707	2246	5779
		0	1	x	0	1	1	1725	8560	27169	11376	1496	30343	3368	14686	1973	4237
		0	1	x	0	1	0	1725	8866	27962	11480	1496	30342	3373	15008	1973	4237
		0	1	x	0	0	1	1783	15486	46677	17629	1634	50261	3797	26865	2676	6980
		0	1	x	0	0	0	1783	16448	49882	18756	1658	54478	3887	29286	2681	6982
FLASH-AXIM	DTCM-RAM	0	1	1	1	x	x	997	8153	21598	9110	473	23559	1219	12856	1015	2466
		0	1	1	0	x	x	1135	8695	22835	9111	479	24122	1488	13793	1663	3136
		0	1	0	1	x	x	1135	24730	70658	27505	1500	81763	3275	44050	2135	7960
		0	1	0	0	x	x	1436	25188	71826	27504	1500	82423	3557	44735	2676	8315
FLASH-ITCM	DTCM-RAM	0	1	x	x	1	1	1019	7869	20400	8927	454	23977	1223	13365	1014	2449
		0	1	x	x	1	0	1019	8303	21321	9086	454	24245	1228	13773	1014	2449
		0	1	x	x	0	1	1309	15282	41927	16514	845	48483	2196	26325	1964	5654
		0	1	x	x	0	0	1297	16241	45436	17679	978	52720	2609	28702	2260	5779
		min						995	1327	11889	637	451	893	1219	695	972	2449

27

fpu: off / single precision

r/o mem	r/w mem	microlib	ICEN	DCEN	ARTEN	PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	0	1	1	x	x	0,998996	4,112782	0,697127	7,904615	1,014737	19,40706	1,005007	15,26714	1,023398	0,993683
		0	1	0	x	x	1,005962	3,140645	0,795445	5,457313	1,007227	11,87138	1,006087	8,993392	0,992841	0,979877
		0	0	1	x	x	0,968961	8,517043	0,769293	7,98728	0,987654	27,20781	1,082918	30,17201	0,995434	0,92148
		0	0	0	x	x	1,029623	6,711321	0,778735	7,427568	0,993063	22,47396	1,059612	19,66024	0,976598	1,020202
FLASH-ITCM	SRAM1	0	x	1	1	1	1	4,032471	0,763143	6,798165	0,99345	20,19942	1,00228	16,58273	1,015432	1
		0	x	1	1	0	1,001005	4,209222	0,759148	6,782282	0,995633	20,8599	1,000759	17,31367	1,014388	1
		0	x	1	0	1	1,050475	5,786225	0,773549	8,899917	1,005669	24,82721	1,021208	16,46102	1,035097	0,967473
		0	x	1	0	0	1,033094	6,10833	0,76558	8,525074	1,027	22,82152	1,015163	17,18921	1,044733	0,95968
		0	x	0	1	1	0,993582	3,766177	0,78856	3,720027	0,998664	12,93951	0,99941	13,16231	0,968246	1
		0	x	0	1	0	0,994746	3,989328	0,789156	3,697784	0,998664	13,06988	1,000295	13,67394	0,968719	1,000235
		0	x	0	0	1	1,014908	5,513113	0,769082	6,669388	1,015385	18,3451	1,015168	15,17703	0,984305	1,005139
		0	x	0	0	0	1,009748	5,827614	0,766057	6,917421	0,998195	18,8391	0,993702	16,31205	0,986407	0,990663
FLASH-AXIM	DTCM-RAM	0	1	1	x	x	1	4,125475	0,706881	7,890076	1,040685	23,27996	0,995925	15,03286	1,032389	0,993915
		0	1	0	x	x	1,021145	3,308182	0,771707	10,66717	1,016736	17,61983	0,983957	9,588581	1,079706	0,994872
		0	0	1	x	x	0,968982	8,265544	0,764641	7,048293	0,988939	28,59256	1,076726	30,43544	0,995434	0,92148
		0	0	0	x	x	1,223582	6,803738	0,774101	7,783325	0,988281	25,47528	1,068052	21,51839	1,020989	0,936982
FLASH-ITCM	DTCM-RAM	0	x	x	1	1	0,998996	4,032471	0,76156	6,787136	0,993392	21,66841	1	16,58273	1,019173	1
		0	x	x	1	0	0,998996	4,209222	0,757644	6,768769	0,993407	22,34855	1,000797	17,32662	1,019173	1
		0	x	x	0	1	1,051245	5,789262	0,772765	8,922176	1,005688	25,40789	1,026316	16,45727	1,012352	0,966678
		0	x	x	0	0	1,033094	6,108761	0,765509	8,523985	1,028028	23,1846	1,016929	17,19776	1,030864	0,959251
FLASH-AXIM	SRAM1	1	1	1	x	x	1	6,148455	0,960438	13,50074	1,023404	22,10252	1	15,01871	0,898655	1,006357
		1	1	0	x	x	0,985019	4,832308	0,974217	7,500327	0,998694	14,20329	1	9,603558	0,921007	1,024099
		1	0	1	x	x	0,790536	10,24823	0,982018	13,25518	0,999336	33,28693	0,992941	34,82213	0,980699	0,905678
		1	0	0	x	x	1,018365	7,948571	0,978319	11,7572	1,009014	27,10341	1,007314	22,49623	0,991173	0,850709
FLASH-ITCM	SRAM1	1	x	1	1	1	1	5,617037	0,980394	14,0094	1,002188	23,29223	0,98627	19,24317	1,023638	1
		1	x	1	1	0	1	5,128872	0,965896	14,12267	0,997817	23,50581	0,986343	19,81727	1,028777	1
		1	x	1	0	1	0,91156	6,871403	0,947947	13,39497	0,982679	29,48966	1,022385	20,22197	1	0,974078
		1	x	1	0	0	0,902575	6,878865	0,959304	12,65784	0,974206	27,09605	1,021236	21,01537	0,980358	0,872829
		1	x	0	1	1	1	5,462668	0,971779	7,717775	1,000669	16,11418	0,986237	16,01527	0,96385	1
		1	x	0	1	0	1	4,882159	0,970229	7,783051	0,999332	15,93592	0,985393	16,36641	0,96385	1
		1	x	0	0	1	1	6,5954	0,9487	9,898372	1,009265	22,39795	1,015512	18,77358	1,048589	0,981026
		1	x	0	0	0	0,999439	6,645657	0,969712	10,03531	1,016554	22,36371	1,026406	20,14168	0,96822	0,921473
FLASH-AXIM	DTCM-RAM	1	1	1	x	x	1	6,143934	0,946616	13,59701	1,006383	26,38186	0,991863	14,52655	0,869751	1,006531
		1	1	0	x	x	0,980138	4,811843	0,944258	13,32018	1,019149	19,8046	0,98153	10,94683	0,981121	1,010309
		1	0	1	x	x	0,78984	10,31706	0,979633	13,24266	0,998668	34,33977	0,983483	34,84968	0,981158	0,905575
		1	0	0	x	x	0,755392	8,33212	0,980305	13,17873	0,998668	29,9829	0,987781	24,27292	0,994796	0,896303
FLASH-ITCM	DTCM-RAM	1	x	x	1	1	1	5,628755	0,955727	14,01413	1,002208	25,1595	0,980754	19,23022	1,023209	1
		1	x	x	1	0	1	5,144362	0,947305	14,15265	1	25,22893	0,979266	19,81727	1,023209	1
		1	x	x	0	1	0,91156	6,871403	0,942413	13,45884	0,978009	29,76243	1,018081	20,23444	1,009769	0,909735
		1	x	x	0	0	0,902575	6,878865	0,957051	12,70956	0,971202	27,27367	1,020336	21,01171	0,982182	0,872434

28

microlib / std lib

r/o mem	r/w mem	hw fpu	ICEN	DCEN	ARTEN	PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	1	1	1	x	x	1	0,831454	1,223083	1,041538	0,989474	0,997212	1,002861	1,221429	1,134283	0,993683
		1	1	0	x	x	1,013008	0,816241	1,30526	1,013236	1,005913	1,008543	1,005257	1,155653	1,077995	0,979877
		1	0	1	x	x	0,969636	0,984394	1,624248	1,016145	0,979207	0,992348	1,05985	1,06661	0,993607	0,921375
		1	0	0	x	x	1,040612	0,990566	1,643483	1,052239	1,006403	1,000977	1,042514	1,066988	0,951695	1,020202
FLASH-ITCM	SRAM1	1	x	1	1	1	1,024121	0,840048	1,371783	0,975535	0,997817	1,001946	0,996201	1	1,001029	0,992897
		1	x	1	1	0	1,025126	0,930259	1,340037	0,966967	1	0,997101	1	1	1	0,992897
		1	x	1	0	1	1,050475	0,994633	1,473004	1,028357	0,981859	0,989765	1,009221	0,976012	1,052376	1,015201
		1	x	1	0	0	1,033813	1,01899	1,45454	1,030236	1,008	0,978884	1,006998	0,957253	1,101972	0,999849
		1	x	0	1	1	1,006418	0,852094	1,650316	1,004087	0,998664	1,00803	1,007375	0,998911	1	0,996472
		1	x	0	1	0	1,007005	0,96905	1,597827	0,990598	1	1,007941	1,008544	1	1,000489	0,996472
		1	x	0	0	1	1,022362	0,993232	1,560846	1,038484	0,996308	0,977778	1,01273	0,978127	0,953662	1,015703
		1	x	0	0	0	1,022936	1,011029	1,514857	1,057127	0,981348	0,992261	0,993702	0,957209	1,017267	1,025304
FLASH-AXIM	DTCM-RAM	1	1	1	x	x	1,001004	0,840938	1,250055	1,022901	1,006424	1	1,00163	1,264286	1,181174	0,993915
		1	1	0	x	x	1,020264	0,821364	1,091931	1,025487	0,983264	1,006612	1,013369	1,057935	1,03924	0,994872
		1	0	1	x	x	0,968982	0,961492	1,614735	1,013171	0,977228	0,994985	1,069021	1,066667	0,993607	0,92148
		1	0	0	x	x	1,224871	0,974218	1,539523	1,011634	0,977865	1,014391	1,084312	1,07589	0,956955	0,936881
FLASH-ITCM	DTCM-RAM	1	x	x	1	1	1,023092	0,840649	1,367042	0,975498	0,997797	0,996862	1	1	1	0,992704
		1	x	x	1	0	1,023092	0,930259	1,336282	0,963964	0,997802	0,996888	1	1	1	0,992704
		1	x	x	0	1	1,051245	0,995078	1,469399	1,026778	0,982935	1,003697	1,013628	0,975262	1,044576	1,015191
		1	x	x	0	0	1,033813	1,01899	1,453702	1,026568	1,008008	0,985722	1,006693	0,957924	1,092593	0,999698
FLASH-AXIM	SRAM1	1	1	1	x	x	1,001005	1,242992	1,685053	1,778902	0,997925	1,135715	0,997865	1,201553	0,996024	1,006357
		1	1	0	x	x	0,991918	1,255898	1,598611	1,392554	0,997391	1,206653	0,999175	1,23406	1	1,024099
		1	0	1	x	x	0,791086	1,184483	2,073386	1,686329	0,990789	1,214071	0,97179	1,230997	0,978899	0,905575
		1	0	0	x	x	1,029234	1,17318	2,064696	1,665603	1,022569	1,20717	0,99106	1,220902	0,965899	0,850709
FLASH-ITCM	SRAM1	1	x	1	1	1	1,024121	1,170146	1,762301	2,010346	1,006593	1,155358	0,980288	1,160434	1,009119	0,992897
		1	x	1	1	0	1,024096	1,133507	1,712046	2,013505	1,002193	1,123576	0,985595	1,144602	1,014184	0,992897
		1	x	1	0	1	0,91156	1,181172	1,805095	1,547746	0,959414	1,175639	1,010384	1,199007	1,016693	1,022132
		1	x	1	0	0	0,903203	1,147531	1,8226	1,529671	0,956183	1,162231	1,013022	1,170329	1,03407	0,909363
		1	x	0	1	1	1,012918	1,235923	2,03376	2,083135	1,000669	1,255347	0,994097	1,215427	0,995459	0,996472
		1	x	0	1	0	1,012324	1,185928	1,964451	2,084998	1,000669	1,228968	0,99352	1,196906	0,995459	0,996238
		1	x	0	0	1	1,007345	1,188215	1,925381	1,541266	0,990303	1,193791	1,013074	1,209917	1,015945	0,991336
		1	x	0	0	0	1,012493	1,152951	1,91758	1,533606	0,999397	1,177903	1,026406	1,181936	0,99851	0,953695
FLASH-AXIM	DTCM-RAM	1	1	1	x	x	1,001004	1,252381	1,674004	1,762771	0,973251	1,133244	0,997545	1,221705	0,995098	1,006531
		1	1	0	x	x	0,979292	1,194696	1,336083	1,280534	0,985597	1,131426	1,01087	1,207793	0,94435	1,010309
		1	0	1	x	x	0,78984	1,200136	2,068745	1,903592	0,986842	1,194981	0,976446	1,221372	0,979358	0,905575
		1	0	0	x	x	0,756187	1,193066	1,949621	1,712898	0,988142	1,193879	1,002819	1,213613	0,932404	0,896206
FLASH-ITCM	DTCM-RAM	1	x	x	1	1	1,024121	1,173427	1,715583	2,014215	1,006652	1,15747	0,980754	1,159653	1,00396	0,992704
		1	x	x	1	0	1,024121	1,13693	1,670794	2,015528	1,004425	1,125371	0,978486	1,143747	1,00396	0,992704
		1	x	x	0	1	0,91156	1,18108	1,791982	1,548865	0,955882	1,175716	1,005495	1,199098	1,04191	0,95539
		1	x	x	0	0	0,903203	1,147449	1,81744	1,530649	0,952288	1,159573	1,010066	1,170364	1,040995	0,90922

Mikrokontroler wyposażony jest w kieszenie instrukcji i danych, które można włączyć korzystając z funkcji *SCB_EnableICache()* i *SCB_EnableDCache()* z biblioteki CMSIS. Każda z kieszeni ma rozmiar 4 KiB. Kieszenie mają znaczenie w przypadku, gdy dostępy do pamięci wykonywane są przez interfejs AHB. Mikrokontroler jest również wyposażony w *Adaptive Real Time Accelerator* (ART). ART dysponuje kieszenią o pojemności 64 linii po 256 bitów każda. Włączenie tej kieszeni następuje przez ustawienie bitu ARTEN w rejestrze FLASH_ACR. ART odpowiada także za mechanizm prefetch. Mechanizm prefetch jest włączany przez ustawienie bitu PREFTEN w rejestrze FLASH_ACR. ART i ART-prefetch mają znaczenie w przypadku, gdy instrukcje są pobierane z Flash przez interfejs ITCM.

Przetestowano następujące konfiguracje i wpływ odpowiednich mechanizmów wspomagających:

- RO-Mem: Flash-AXI, R/W-Mem: SRAM1 – kieszeń instrukcji, kieszeń danych,
- RO-Mem: Flash-ITCM, R/W-Mem: SRAM1 – ART i ART-prefetch, kieszeń danych,
- RO-Mem: Flash-AXI, R/W-Mem: DTCM-RAM – kieszeń instrukcji, kieszeń danych (ze względu na możliwy odczyt stałych z pamięci Flash przez interfejs AHB),
- RO-Mem: Flash-ITCM, R/W-Mem: DTCM-RAM – ART i ART-prefetch.

Wyniki: 26, 27, 28

Wnioski:

- **–o2 vs –o3**

W większości wypadków różnice były nieznaczne (poniżej 2%). Dla prostych obliczeń stałopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był ok. 5…10% krótszy, jeśli odpowiednie mechanizmy wspomagające pobieranie instrukcji (kieszeń instrukcji lub ART i ART-prefetch) były włączone oraz do ok. 40% krótszy, jeśli nie były włączone.

Dla FOC operującego na liczbach stałopozycyjnych oraz, jeżeli używane było FPU, dla FOC operującego na liczbach zmiennopozycyjnych czas wykonania w przypadku poziomu optymalizacji –o2 był do ok. 15% dłuższy.

Jeśli odpowiednie mechanizmy wspomagania pobierania instrukcji nie były włączone, czas wykonania dla automatu był do ok. 15% krótszy przy poziomie optymalizacji –o2, jeśli używane było FPU lub biblioteka standardowa oraz do ok. 15% dłuższy, jeśli FPU nie było używane i korzystano z Microlib.

- **FPU**

Wykorzystanie FPU przyspieszyło obliczenia na liczbach zmiennopozycyjnych pojedynczej precyzji ok. 3,5–30 razy dla biblioteki standardowej i ok. 5–35 razy dla biblioteki Microlib. Użycie wsparcia sprzętowego dla obliczeń zmiennopozycyjnych było najbardziej korzystne w przypadku FOC i FFT, gdzie przyspieszenie wynosiło od ok. 10 do ok. 30 razy. Trochę mniejsze różnice występowały w wypadku obliczeń na macierzach (4…10 razy dla biblioteki standardowej i 7…14 razy dla biblioteki Microlib) i prostych obliczeń (3…8 razy dla biblioteki standardowej i 5…10 razy dla biblioteki Microlib).

Obliczenia na liczbach zmiennopozycyjnych podwójnej precyzji w przypadku korzystania z FPU wykonywały się o ok. 30% wolniej dla biblioteki standardowej i ok. 2…5% wolniej dla biblioteki Microlib.

- **micro lib vs std lib**

Rozmiar kodu wynikowego był o ok. 1/6 większy dla biblioteki standardowej. Dla biblioteki Microlib czas wykonania dla prostych obliczeń zmiennopozycyjnych podwójnej precyzji był ok. 1,2…1,6 razy dłuższy w przypadku używania FPU i ok. 1,3…2 razy dłuższy, jeśli FPU nie było wykorzystywane. W wypadku niekorzystania z FPU czas wykonania dla biblioteki Microlib był ok. 1,5…2 razy dłuższy dla obliczeń na macierzach liczb zmiennopozycyjnych i ok. 1,2 razy dłuższy dla pozostałych obliczeń zmiennopozycyjnych.

W wypadku korzystania z FPU czas wykonania dla biblioteki Microlib był do ok. 1,2 razy krótszy dla prostych obliczeń zmiennopozycyjnych pojedynczej precyzji. Dla konfiguracji, gdzie dostęp do kodu następował przez interfejs AXI, czas wykonania dla biblioteki Microlib był do ok. 1,25 razy dłuższy dla FOC.

Wyniki cd.: 29, 30, 31

Wnioski cd.:

- **Instruction Cache (IC)**

Włączenie kieszeni instrukcji spowodowało skrócenie czasu wykonania ok. 1,2…3,8 razy. Największe znaczenie miało w przypadku automatu, operacji na macierzach i na liczbach zmiennopozycyjnych podwójnej precyzji, a w przypadku, gdy FPU nie było wykorzystywane również dla obliczeń na liczbach zmiennopozycyjnych pojedynczej precyzji. Przyspieszenie uzyskane dzięki włączeniu kieszeni instrukcji było większe, jeśli jednocześnie włączona była kieszeń danych.

- **Data Cache (DC)**

Włączenie kieszeni danych spowodowało skrócenie czasu wykonania maksymalnie ok. 3,3 razy. Włączenie kieszeni danych miało większy wpływ na przyspieszenie wykonania programu, jeśli jednocześnie włączone były odpowiednie mechanizmy wspomagające pobieranie instrukcji (kieszeń instrukcji w przypadku korzystania z interfejsu AHB/AXI, ART w przypadku korzystania z interfejsu ITCM).

Włączenie kieszeni danych miało mniejszy wpływ, jeśli dane do odczytu i zapisu znajdowały się w DTCM-RAM, ponieważ wtedy mogło przyspieszyć jedynie odczyt stałych.

Włączenie kieszeni danych miało największe znaczenie w przypadku obliczeń na macierzach liczb stałopozycyjnych oraz FFT i FOC operujących na liczbach stałopozycyjnych. Dość istotne było również dla tych samych testów w wersjach operujących na liczbach zmiennopozycyjnych, jeżeli wykorzystywane było FPU. Najmniejsze znaczenie miało dla prostych obliczeń na liczbach zmiennopozycyjnych pojedynczej i podwójnej precyzji oraz dla pozostałych

ICEN = 0 / 1

29

r/o mem	r/w mem	hw fpu	microlib	DCEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	1	0	1	1,487952	1,525689	2,371662	3,144615	3,24	2,307621	2,294707	1,694286	2,227874	3,76589
		1	0	0	1,134417	1,331101	2,008158	1,507611	1,231275	1,457997	1,197565	1,370044	1,25584	2,073511
FLASH-AXIM	DTCM-RAM	1	0	1	1,488956	1,579848	2,447293	3,129771	3,291221	2,679731	2,538712	1,692857	2,216599	3,869777
		1	0	0	1,367401	1,410455	2,148914	3,092954	3,213389	2,239669	2,21992	1,438287	1,723483	3,173718
FLASH-AXIM	SRAM1	1	1	1	1,442771	1,80633	3,149557	3,067947	3,206383	2,296365	2,425107	1,479532	1,95157	3,491855
		1	1	0	1,165329	1,615385	2,528517	1,565643	1,231875	1,447059	1,241949	1,26493	1,108703	2,158843
FLASH-AXIM	DTCM-RAM	1	1	1	1,441324	1,80633	3,161246	3,1	3,195745	2,666293	2,70952	1,428249	1,86461	3,587755
		1	1	0	1,641623	1,672939	3,029773	3,05117	3,195745	2,256979	2,37533	1,462698	1,587021	2,988724
FLASH-AXIM	SRAM1	1	0	1	1,443216	3,159506	2,617176	3,177501	3,153527	3,235179	2,472598	3,348367	2,166998	3,492253
		1	0	0	1,161099	2,844462	1,965972	2,051904	1,21396	2,760165	1,261276	2,99502	1,235294	2,158843
FLASH-AXIM	DTCM-RAM	1	0	1	1,442771	3,165284	2,647264	2,795859	3,127572	3,29126	2,744681	3,42735	2,137255	3,587755
		1	0	0	1,638481	2,900797	2,155579	2,256781	3,123457	3,23818	2,409647	3,227758	1,629756	2,989046
FLASH-AXIM	SRAM1	1	1	1	1,140562	3,010786	3,220323	3,012144	3,130977	3,458383	2,407989	3,430418	2,129741	3,142519
		1	1	0	1,20478	2,657116	2,539163	2,454237	1,244604	2,761348	1,251032	2,963085	1,193169	1,793329
FLASH-AXIM	DTCM-RAM	1	1	1	1,138415	3,033239	3,271507	3,01921	3,171247	3,470563	2,586628	3,426416	2,103448	3,227899
		1	1	0	1,265198	2,896837	3,145435	3,018769	3,131524	3,416922	2,390457	3,243312	1,60914	2,651467

obliczeń zmiennopozycyjnych, jeśli FPU nie było używane. Jednak nawet w tym przypadku, jeśli dane do odczytu i zapisu znajdowały się w SRAM1 i używane były odpowiednie mechanizmy wspomagające pobieranie instrukcji, to przyspieszenie wynikające z włączenia kieszeni danych było dość duże.

- **ART Accelarator (ARTEN)**

Włączenie ART spowodowało skrócenie czasu wykonania maksymalnie ok. 2,6 razy. Włączenie ART miało największe znaczenie w przypadku automatu oraz, jeżeli FPU nie było używane i wykorzystano bibliotekę standardową, w przypadku obliczeń na macierzach liczb zmiennopozycyjnych. Dość duże również dla obliczeń na macierzach w pozostałych konfiguracjach, FOC (szczególnie operującego na liczbach zmiennopozycyjnych) oraz dla prostych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji.

Włączenie ART miało większy wpływ na czas wykonania, jeśli mechanizm prefetch nie był włączony. W konfiguracjach, gdzie dane do odczytu i zapisu były umieszczone w SRAM1, przyspieszenie uzyskane dzięki włączeniu ART Accelerator było znacznie mniejsze, jeśli kieszeń danych nie była włączona.

Wyniki cd.: 32, 33, 34, 35

Porównanie konfiguracji

Wnioski cd.:

- **ART-prefetch (PREFTEN)**

Włączenie mechanizmu prefetch spowodowało skrócenie czasu wykonania maksymalnie ok. 1,2 razy. Włączenie mechanizmu prefetch miało większe znaczenie, jeśli ARTEN nie było ustawione. Jeśli ARTEN=1, to włączenie mechanizmu prefetch spowodowało skrócenie czasu wykonania dla prostych obliczeń na liczbach zmiennopozycyjnych pojedynczej i podwójnej precyzji oraz, jeśli FPU nie było używane, także dla pozostałych obliczeń zmiennopozycyjnych. Jeśli ARTEN=0, to istotna poprawa czasu wykonania wystąpiła również w przypadku obliczeń na macierzach liczb stałopozycyjnych oraz FFT i FOC operujących na liczbach stałopozycyjnych.

- **Mechanizmy wspomagające (ogólnie)**

Różnice czasów wykonania między przypadkiem, gdy mechanizmy wspomagające wykonanie programu są włączone, a przypadkiem, gdy są wyłączone, wyniosły od 1,5 do 7 razy. Jeśli mechanizmy wspomagające wykonanie programu były wyłączone, to czasy wykonania

30

DCEN = 0 / 1

r/o mem	r/w mem	hw fpu	microlib	ICEN	ARTEN	PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	1	0	1	x	x	1,85241	1,496867	1,25689	2,324615	3,204211	1,958178	2,585122	1,945714	2,699898	1,922621
		1	0	0	x	x	1,412281	1,305955	1,064247	1,114481	1,217674	1,237213	1,349127	1,573356	1,521918	1,058602
FLASH-ITCM	SRAM1	1	0	x	1	1	1,722613	1,105833	1,087425	2,244648	3,268559	1,817121	2,575988	1,320863	2,105967	1,677979
		1	0	x	1	0	1,721608	1,080115	1,073184	2,235736	3,268559	1,825121	2,575114	1,319424	2,102775	1,677979
		1	0	x	0	1	1,275786	1,057245	1,042222	1,430359	1,842404	1,381698	1,702167	1,096702	1,444924	1,144982
		1	0	x	0	0	1,254676	1,056539	1,039871	1,303835	1,662	1,234289	1,481726	1,064471	1,309283	1,115977
FLASH-AXIM	DTCM-RAM	1	0	1	x	x	1,139558	1,39417	1,213401	1,018321	1,023555	1,354983	1,219234	1,701429	1,65081	1,26572
		1	0	0	x	x	1,046527	1,244685	1,065461	1,006341	0,999349	1,13247	1,066132	1,44557	1,283562	1,038054
FLASH-AXIM	SRAM1	1	0	1	x	x	1,876506	1,46948	1,341339	2,261448	3,257447	1,980429	2,591298	1,840936	2,565919	1,895908
		1	0	0	x	x	1,515658	1,314143	1,076849	1,154068	1,251493	1,247971	1,327059	1,573913	1,457721	1,172147
FLASH-ITCM	SRAM1	1	0	x	1	1	1,692836	1,121689	1,308221	2,310345	3,271335	1,828155	2,604882	1,319424	2,103803	1,684022
		1	0	x	1	0	1,691176	1,125155	1,279638	2,290373	3,268559	1,844961	2,597117	1,319424	2,103803	1,684022
		1	0	x	0	1	1,241643	1,055755	1,104375	1,444444	1,869515	1,364964	1,708086	1,099078	1,309389	1,145548
		1	0	x	0	0	1,241475	1,048285	1,082993	1,337867	1,618056	1,251156	1,462162	1,064422	1,208643	1,144389
FLASH-AXIM	DTCM-RAM	1	0	1	x	x	1,161484	1,361718	1,059914	1,020896	1	1,363942	1,233523	1,423729	1,452442	1,266939
		1	0	0	x	x	1,322895	1,26116	1,015833	1,004815	1	1,154557	1,081381	1,45807	1,236213	1,055404
FLASH-AXIM	SRAM1	1	0	1	x	x	1,865327	1,143053	1,434153	1,604905	3,180498	1,197826	2,5879	1,146159	2,619284	1,895908
		1	0	0	x	x	1,500696	1,029076	1,077308	1,036384	1,224342	1,021952	1,320092	1,025207	1,493119	1,172014
FLASH-ITCM	SRAM1	1	0	x	1	1	1,711558	1,032806	1,123644	1,228295	3,285714	1,164026	2,568613	1,048416	2,008105	1,677979
		1	0	x	1	0	1,710843	1,023689	1,115605	1,218951	3,278509	1,143539	2,57392	1,042051	2,008105	1,678374
		1	0	x	0	1	1,232591	1,007343	1,036203	1,071877	1,860203	1,020952	1,692099	1,011157	1,374022	1,189559
		1	0	x	0	0	1,226323	1,007984	1,04052	1,057958	1,615385	1,018902	1,450402	1,010151	1,236188	1,152006
FLASH-AXIM	DTCM-RAM	1	0	1	x	x	1,163655	1,117972	1,324678	1,376741	1	1,025542	1,204583	1,085242	1,726471	1,266939
		1	0	0	x	x	1,321503	1,024556	1,078641	1,111288	0,998684	1,009003	1,057543	1,022043	1,316514	1,055518
FLASH-AXIM	SRAM1	1	0	1	x	x	1,848394	1,154921	1,360582	1,256346	3,178794	1,272643	2,591298	1,177167	2,629741	1,929333
		1	0	0	x	x	1,952465	1,019255	1,072793	1,023646	1,263612	1,016143	1,346268	1,016799	1,47329	1,101005
FLASH-ITCM	SRAM1	1	0	x	1	1	1,692836	1,090863	1,296726	1,272768	3,266376	1,264766	2,604795	1,098101	1,980924	1,684022
		1	0	x	1	0	1,691176	1,071032	1,280077	1,262232	3,273523	1,250804	2,594615	1,089668	1,971029	1,684022
		1	0	x	0	1	1,362108	1,013349	1,105252	1,067389	1,920094	1,036715	1,696604	1,020358	1,373012	1,153719
		1	0	x	0	0	1,374711	1,012746	1,094744	1,06068	1,688391	1,032641	1,469565	1,020169	1,193678	1,208168
FLASH-AXIM	DTCM-RAM	1	0	1	x	x	1,138415	1,066479	1,057274	1,00011	1,012685	1,023897	1,220673	1,072884	1,638424	1,271695
		1	0	0	x	x	1,265198	1,01852	1,01653	0,999964	1	1,008072	1,086107	1,015551	1,253396	1,044598

31

ARTEN = 0 / 1

r/o mem	r/w mem	hw fpu	microlib	DCEN	PREFTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-ITCM	SRAM1	1	0	1	1	1,373869	1,344558	1,941395	1,833333	1,925764	1,615759	1,648176	1,919424	1,90535	2,414365
		1	0	1	0	1,396985	1,335447	1,94294	2,036036	2,183406	1,921739	1,951442	2,053237	2,136691	2,61326
		1	0	0	1	1,017503	1,285481	1,860693	1,168256	1,085504	1,228587	1,089086	1,593682	1,307279	1,64746
		1	0	0	0	1,018097	1,306297	1,88263	1,187374	1,11022	1,299629	1,122864	1,656489	1,330401	1,738006
FLASH-ITCM	DTCM-RAM	1	0	x	1	1,371486	1,343957	1,939093	1,830015	1,936123	1,697699	1,706496	1,919424	1,87891	2,481557
		1	0	x	0	1,395582	1,335447	1,938966	2,034535	2,195604	2,034232	2,025518	2,051799	2,125126	2,685853
FLASH-ITCM	SRAM1	1	1	1	1	1,409225	1,591983	2,084647	1,932602	1,894967	1,596117	1,669718	1,873381	2,003083	2,468601
		1	1	1	0	1,408824	1,462825	2,10896	2,169255	2,200873	1,886628	1,965099	1,965468	2,354573	2,631558
		1	1	0	1	1,033623	1,498405	1,759818	1,208277	1,082943	1,191715	1,094876	1,560523	1,246702	1,679254
		1	1	0	0	1,034203	1,362885	1,784872	1,267119	1,089512	1,279412	1,106339	1,585605	1,352711	1,788294
FLASH-ITCM	DTCM-RAM	1	1	x	1	1,409225	1,590844	2,084282	1,926217	1,907285	1,709339	1,729751	1,871942	1,962664	2,537771
		1	1	x	0	1,410206	1,462825	2,109344	2,166667	2,218062	2,011446	2,039075	1,965468	2,321897	2,704777
FLASH-ITCM	SRAM1	0	0	1	1	1,443216	1,929317	1,967869	2,400135	1,949451	1,985938	1,679303	1,905336	1,942249	2,335833
		0	0	1	0	1,441767	1,937971	1,959401	2,559221	2,252193	2,102455	1,97953	2,038478	2,200608	2,507893
		0	0	0	1	1,039342	1,88175	1,814732	2,094488	1,103679	1,741839	1,106257	1,837623	1,328961	1,655927
		0	0	0	0	1,033451	1,90824	1,827526	2,221213	1,109699	1,873304	1,115464	1,976075	1,354692	1,721373
FLASH-ITCM	DTCM-RAM	0	0	x	1	1,443216	1,929466	1,967623	2,405686	1,960089	1,990683	1,751403	1,904902	1,866337	2,398865
		0	0	x	0	1,443216	1,938108	1,959094	2,562112	2,272124	2,110332	2,058167	2,036539	2,149505	2,576409
FLASH-ITCM	SRAM1	0	1	1	1	1,284593	1,947496	2,015655	1,847841	1,858079	2,020799	1,730858	1,968671	1,956827	2,40461
		0	1	1	0	1,271569	1,961947	2,085927	1,944255	2,148796	2,174788	2,034615	2,084295	2,243756	2,2969
		0	1	0	1	1,033623	1,809112	1,718024	1,549666	1,092246	1,656428	1,127375	1,829293	1,35631	1,647392
		0	1	0	0	1,033623	1,855177	1,783921	1,633798	1,108289	1,795465	1,152387	1,951359	1,358844	1,647864
FLASH-ITCM	DTCM-RAM	0	1	x	1	1,284593	1,942051	2,055245	1,849894	1,861233	2,022063	1,795585	1,969697	1,936884	2,308697
		0	1	x	0	1,272816	1,95604	2,131045	1,945741	2,154185	2,174469	2,124593	2,083932	2,228797	2,359739

PREFTEN = 0 / 1

r/o mem	r/w mem	hw fpu	microlib	DCEN	ARTEN	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-ITCM	SRAM1	1	0	1	1	1	1,043295	1,078824	1,018349	1	1,006809	1,00152	1	1,001029	1
		1	0	1	0	1,016825	1,036225	1,079683	1,130942	1,133787	1,197471	1,1858	1,069715	1,12257	1,08238
		1	0	0	1	0,999417	1,019032	1,064695	1,014305	1	1,011242	1,00118	0,998911	0,999511	1
		1	0	0	0	1	1,035533	1,077248	1,030904	1,022769	1,069717	1,032232	1,038278	1,01719	1,054961
FLASH-ITCM	DTCM-RAM	1	0	x	1	1	1,043295	1,078711	1,019908	1,002203	1,008368	1,005613	1	1	1
		1	0	x	0	1,01757	1,036689	1,078641	1,133891	1,136519	1,208256	1,193609	1,068966	1,131042	1,082326
FLASH-ITCM	SRAM1	1	1	1	1	1,000981	1,155333	1,053858	1,009404	1,002188	1,001942	1,005339	1	1	1
		1	1	1	0	1,000696	1,061601	1,066149	1,133009	1,163972	1,184307	1,183189	1,049155	1,175475	1,066012
		1	1	0	1	1	1,158902	1,030832	1,000678	1,001338	1,011152	1,002343	1	1	1
		1	1	0	0	1,000561	1,054089	1,045507	1,04941	1,007412	1,085561	1,012838	1,016073	1,085031	1,064933
FLASH-ITCM	DTCM-RAM	1	1	x	1	1	1,154506	1,054439	1,007849	1,002208	1,008395	1,005613	1	1	1
		1	1	x	0	1,000696	1,061601	1,067118	1,133659	1,165509	1,186618	1,185443	1,049962	1,183033	1,065809
FLASH-ITCM	SRAM1	0	0	1	1	1,001005	1,089025	1,073177	1,015969	1,002198	1,03973	1	1,044078	1	1
		0	0	1	0	1	1,093909	1,068559	1,08331	1,157835	1,100732	1,178781	1,117036	1,13302	1,073661
		0	0	0	1	1,000587	1,079411	1,065499	1,00824	1	1,021431	1,002066	1,037739	1	1,000235
		0	0	0	0	0,994915	1,094606	1,073011	1,069243	1,005455	1,098523	1,010406	1,115925	1,019362	1,039767
FLASH-ITCM	DTCM-RAM	0	0	x	1	1	1,089025	1,073165	1,017148	1,002217	1,040019	1,006415	1,044859	1	1
		0	0	x	0	1	1,093902	1,068513	1,083286	1,161765	1,102529	1,182692	1,117063	1,151724	1,074011
FLASH-ITCM	SRAM1	0	1	1	1	1,000981	1,054925	1,042574	1,017565	0,997817	1,011129	1,005414	1,029834	1,00502	1
		0	1	1	0	0,990833	1,062754	1,078921	1,070659	1,153937	1,088179	1,181859	1,090319	1,152386	0,955207
		0	1	0	1	1	1,035748	1,029188	1,009142	1	0,999967	1,001485	1,021926	1	1
		0	1	0	0	1	1,062121	1,068663	1,063929	1,014688	1,083902	1,023703	1,090117	1,001868	1,000287
FLASH-ITCM	DTCM-RAM	0	1	x	1	1	1,055153	1,045147	1,017811	1	1,011177	1,004088	1,030527	1	1
		0	1	x	0	0,990833	1,062754	1,083693	1,070546	1,157396	1,087391	1,188069	1,090294	1,150713	1,022108

porównanie 4 dostępnych konfiguracji (względem pierwszej)

r/o mem	r/w mem	hw fpu	microlib	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	SRAM1	1	0	1	1	1	1	1	1	1	1	1	1
FLASH-ITCM	SRAM1	1	0	0,998996	1,04198	0,832034	1,006154	0,964211	0,95539	0,941345	0,992857	0,98881	1,000395
FLASH-AXIM	DTCM-RAM	1	0	1	0,988722	0,974792	1,007692	0,983158	0,829926	0,877682	1	1,005086	0,973154
FLASH-ITCM	DTCM-RAM	1	0	1	1,04198	0,833903	1,004615	0,955789	0,888476	0,891989	0,992857	1,008138	0,973944
FLASH-AXIM	SRAM1	1	1	1	1	1	1	1	1	1	1	1	1
FLASH-ITCM	SRAM1	1	1	1,023092	1,052751	0,933191	0,942393	0,97234	0,959925	0,935093	0,812865	0,872646	0,999603
FLASH-AXIM	DTCM-RAM	1	1	1,001004	1	0,996288	0,98966	1	0,832246	0,876605	1,035088	1,046637	0,973381
FLASH-ITCM	DTCM-RAM	1	1	1,023092	1,053504	0,932055	0,940916	0,96383	0,888164	0,889444	0,812865	0,888789	0,972984
FLASH-AXIM	SRAM1	0	0	1	1	1	1	1	1	1	1	1	1
FLASH-ITCM	SRAM1	0	0	1	1,021633	0,910825	0,865317	0,943983	0,994397	0,93879	1,078413	0,981113	1,006754
FLASH-AXIM	DTCM-RAM	0	0	1,001005	0,991773	0,988432	1,005839	1,008299	0,995546	0,869751	0,984654	1,013917	0,973381
FLASH-ITCM	DTCM-RAM	0	0	1	1,021633	0,910978	0,862592	0,935685	0,992003	0,887544	1,078413	1,003976	0,980135
FLASH-AXIM	SRAM1	0	1	1	1	1	1	1	1	1	1	1	1
FLASH-ITCM	SRAM1	0	1	1,023092	0,96176	0,95258	0,977899	0,952183	1,011596	0,922254	1,041508	0,994012	0,993289
FLASH-AXIM	DTCM-RAM	0	1	1,001004	0,999265	0,98195	0,996718	0,983368	0,99338	0,869472	1,001168	1,012974	0,973549
FLASH-ITCM	DTCM-RAM	0	1	1,023092	0,964456	0,927484	0,976696	0,943867	1,011005	0,872325	1,040807	1,011976	0,966838

r/w mem: SRAM1 / DTCM-RAM

r/o mem	hw fpu	microlib	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
FLASH-AXIM	1	0	1	1,011407	1,02586	0,992366	1,017131	1,204927	1,139364	1	0,994939	1,027586
FLASH-ITCM	1	0	0,998996	1	0,997758	1,001531	1,008811	1,075314	1,055333	1	0,980827	1,027158
FLASH-AXIM	1	1	0,998997	1	1,003725	1,010448	1	1,201568	1,140765	0,966102	0,955441	1,027347
FLASH-ITCM	1	1	1	0,999285	1,001218	1,00157	1,00883	1,080797	1,051323	1	0,981837	1,027358
FLASH-AXIM	0	0	0,998996	1,008295	1,011704	0,994195	0,99177	1,004474	1,149755	1,015585	0,986275	1,027347
FLASH-ITCM	0	0	1	1	0,999832	1,003159	1,008869	1,002414	1,057739	1	0,977228	1,027158
FLASH-AXIM	0	1	0,998997	1,000736	1,018381	1,003293	1,016913	1,006664	1,150123	0,998833	0,987192	1,02717
FLASH-ITCM	0	1	1	0,997204	1,027059	1,001232	1,008811	1,000584	1,057236	1,000673	0,982249	1,027358

RO-Mem: FLASH-AXIM / FLASH-ITCM

r/w mem	hw fpu	microlib	int	float	double	matrix fl	matrix fix	fft fl	fft fix	foc fl	foc fix	sm
SRAM1	1	0	1,001005	0,959711	1,201874	0,993884	1,037118	1,046693	1,06231	1,007194	1,011317	0,999605
DTCM-RAM	1	0	1	0,948888	1,168951	1,003063	1,028634	0,9341	0,983962	1,007194	0,996973	0,999189
SRAM1	1	1	0,977429	0,949893	1,071592	1,061129	1,028446	1,041748	1,069413	1,230216	1,14594	1,000397
DTCM-RAM	1	1	0,97841	0,949213	1,068915	1,051805	1,037528	0,937041	0,985565	1,273381	1,177598	1,000408
SRAM1	0	0	1	0,978825	1,097906	1,155646	1,059341	1,005634	1,065201	0,927289	1,01925	0,993291
DTCM-RAM	0	0	1,001005	0,970772	1,085022	1,166065	1,077605	1,003572	0,979952	0,913059	1,009901	0,993109
SRAM1	0	1	0,977429	1,03976	1,04978	1,0226	1,050218	0,988537	1,0843	0,960147	1,006024	1,006757
DTCM-RAM	0	1	0,97841	1,036091	1,058725	1,0205	1,04185	0,982567	0,996729	0,961915	1,000986	1,006942

testów były nieraz dość znacząco gorsze niż dla mikrokontrolera STM32F446RE.

- **Porównanie konfiguracji**

Porównywano wyniki uzyskane, gdy dla danej konfiguracji wszystkie mechanizmy wspomagające były włączone.

- **Read Only Memory Area**

Różnice czasu wykonania między konfiguracjami różniącymi się interfejsem dostępu do RO-Mem wynosiły maksymalnie 1,2 razy, ale w większości przypadków były małe.

Jeżeli wykorzystywany był interfejs ITCM, to dla prostych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji, czasy wykonania były 1,05–1,2 razy krótsze, dla obliczeń na macierzach – do ok. 1,5 razy krótsze.

W wypadku używania FPU i Microlib, wykorzystanie interfejsu ITCM dało ok. 1,15 razy krótszy czas wykonania dla FOC operującego na liczbach zmiennopozycyjnych i ok. 1,25 razy dla FOC operującego na liczbach stałopozycyjnych. Natomiast jeśli FPU nie było używane, to FOC operujące na liczbach zmiennopozycyjnych wykonywało się 5–10% szybciej w przypadku interfejsu AHB/AXI.

- **Read/Write Memory Area**

Dla konfiguracji, w których dane do odczytu i zapisu znajdowały się w DTCM-RAM w przypadku FFT operującego na liczbach stałopozycyjnych oraz zmiennopozycyjnych, jeżeli używane było FPU, czasy wykonania były ok. 1,05–1,2 razy krótsze, natomiast w przypadku automatu – 1,03 razy krótsze.

Dla FOC operującego na liczbach stałopozycyjnych czasy wykonania były o kilka procent krótsze, jeśli dane do odczytu i zapisu znajdowały się w SRAM1.

Wyniki cd.:

Wnioski cd.:

- **double/float**

W wypadku nieużywania FPU, czas wykonania tych samych obliczeń na liczbach zmiennopozycyjnych podwójnej precyzji był ok. 2 razy dłuższy, jeżeli wykorzystywana była biblioteka standardowa i ok. 3 razy dłuższy, jeżeli wykorzystywana była biblioteka Microlib.

W wypadku używania FPU, czas wykonania był odpowiednio ok. 10–15 i ok. 15–30 razy dłuższy.

- **floating/fixed point**

W wypadku nieużywania FPU, czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 10…20 razy dłuższy. Dla FFT czasy wykonania różniły się typowo ok. 10…12 razy.

W wypadku używania FPU, czas wykonania FOC operującego na liczbach zmiennopozycyjnych był ok. 1,3 razy krótszy, natomiast FFT ok. 1,3…2 razy krótszy.

Katarzyna Kosowska

r/o mem	r/w mem	hw fpu	microlib	ICEN	DCEN	ARTEN	PRETEN		double / float			fft fl / foc fix	foc fl / foc fix	
FLASH-AXIM	SRAM1	1	0	1	1	x	x		11,731830			0,769671	0,712106	
		1	0	1	0	x	x		9,850984			0,583011	0,513188	
		1	0	0	1	x	x		18,236961			0,774002	0,541553	
		1	0	0	0	x	x		14,861635			0,709797	0,559856	
FLASH-ITCM	SRAM1	1	0	x	1	1	1		9,368010			0,781155	0,715021	
		1	0	x	1	1	0		9,687032			0,785281	0,714286	
		1	0	x	1	0	1		13,526386			0,765791	0,720302	
		1	0	x	1	0	0		14,093656			0,773328	0,686388	
		1	0	x	0	1	1		9,212072			0,551032	0,448461	
		1	0	x	0	1	0		9,624867			0,556570	0,448192	
		1	0	x	0	0	1		13,334179			0,621614	0,546712	
		1	0	x	0	0	0		13,871324			0,644188	0,558046	
FLASH-AXIM	DTCM-RAM	1	0	1	1	x	x		11,566540			0,727791	0,708502	
		1	0	1	0	x	x		10,066818			0,808824	0,730227	
		1	0	0	1	x	x		17,917369			0,768218	0,541096	
		1	0	0	0	x	x		15,337415			0,816019	0,609392	
FLASH-ITCM	DTCM-RAM	1	0	x	x	1	1		9,389056			0,766640	0,701312	
		1	0	x	x	1	0		9,707781			0,768740	0,701312	
		1	0	x	x	0	1		13,546756			0,762688	0,716434	
		1	0	x	x	0	0		14,094950			0,772047	0,677113	
FLASH-AXIM	SRAM1	1	1	1	1	x	x		17,257724			0,765335	0,766816	
		1	1	1	0	x	x		15,752821			0,584916	0,550157	
		1	1	0	1	x	x		30,090947			0,724706	0,581342	
		1	1	0	0	x	x		24,657460			0,681516	0,627680	
FLASH-ITCM	SRAM1	1	1	x	1	1	1		15,297781			0,785660	0,714286	
		1	1	x	1	1	0		13,954151			0,783005	0,714286	
		1	1	x	1	0	1		20,031924			0,751028	0,668035	
		1	1	x	1	0	0		20,117747			0,751737	0,596246	
		1	1	x	0	1	1		17,841736			0,551391	0,447973	
		1	1	x	0	1	0		15,870044			0,556237	0,447973	
		1	1	x	0	0	1		20,954429			0,600160	0,560737	
		1	1	x	0	0	0		20,783838			0,643253	0,525099	
FLASH-AXIM	DTCM-RAM	1	1	1	1	x	x		17,193670			0,726607	0,758355	
		1	1	1	0	x	x		13,382955			0,803430	0,743363	
		1	1	0	1	x	x		30,090530			0,715015	0,580882	
		1	1	0	0	x	x		24,237182			0,763399	0,685130	
FLASH-ITCM	DTCM-RAM	1	1	x	x	1	1		15,268240			0,764234	0,701312	
		1	1	x	x	1	0		13,944857			0,766348	0,701312	
		1	1	x	x	0	1		20,004047			0,755216	0,668895	
		1	1	x	x	0	0		20,108005			0,755964	0,593655	
FLASH-AXIM	SRAM1	0	0	1	1	x	x		1,988574			14,862633	10,623260	
		0	0	1	0	x	x		2,495002			6,879263	4,648577	
		0	0	0	1	x	x		1,647235			19,446459	16,414679	
		0	0	0	0	x	x		1,724440			15,054514	11,270661	
FLASH-ITCM	SRAM1	0	0	x	1	1	1		1,772890			15,742987	11,676798	
		0	0	x	1	1	0		1,747090			16,368461	12,191489	
		0	0	x	1	0	1		1,808317			18,617607	11,454877	
		0	0	x	1	0	0		1,766410			17,384910	11,293278	
		0	0	x	0	1	1		1,928819			7,134298	6,096367	
		0	0	x	0	1	0		1,903959			7,272165	6,326438	
		0	0	x	0	0	1		1,860124			11,233191	8,429765	
		0	0	x	0	0	0		1,823426			12,212833	9,228305	
FLASH-AXIM	DTCM-RAM	0	0	1	1	x	x		1,981874			17,012275	10,316667	
		0	0	1	0	x	x		2,348310			14,483696	6,484952	
		0	0	0	1	x	x		1,657527			20,400119	16,544037	
		0	0	0	0	x	x		1,745027			19,463772	12,843554	
FLASH-ITCM	DTCM-RAM	0	0	x	x	1	1		1,773188			16,611868	11,410891	
		0	0	x	x	1	0		1,747364			17,166534	11,922772	
		0	0	x	x	0	1		1,808254			18,881410	11,646684	
		0	0	x	x	0	0		1,766285			17,601626	11,296177	
FLASH-AXIM	SRAM1	0	1	1	1	x	x		2,695796			16,915835	12,815369	
		0	1	1	0	x	x		3,175846			8,307735	5,736622	
		0	1	0	1	x	x		2,883411			24,294727	20,641987	
		0	1	0	0	x	x		3,034867			18,337294	14,246183	
FLASH-ITCM	SRAM1	0	1	x	1	1	1		2,670065			18,554524	13,427711	
		0	1	x	1	1	0		2,638802			18,660000	13,759241	
		0	1	x	1	0	1		2,763513			21,662645	13,508979	
		0	1	x	1	0	0		2,805554			19,945558	12,781389	
		0	1	x	0	1	1		3,173949			9,009204	7,443487	
		0	1	x	0	1	0		3,153846			8,995553	7,606690	
		0	1	x	0	0	1		3,014142			13,237029	10,039238	
		0	1	x	0	0	0		3,032709			14,015436	10,923536	
FLASH-AXIM	DTCM-RAM	0	1	1	1	x	x		2,649086			19,326497	12,666010	
		0	1	1	0	x	x		2,626222			16,211022	8,294047	
		0	1	0	1	x	x		2,857178			24,965802	20,632319	
		0	1	0	0	x	x		2,851596			23,172055	16,717115	
FLASH-ITCM	DTCM-RAM	0	1	x	x	1	1		2,592451			19,605070	13,180473	
		0	1	x	x	1	0		2,567867			19,743485	13,582840	
		0	1	x	x	0	1		2,743555			22,077869	13,403768	
		0	1	x	x	0	0		2,797611			20,206976	12,700000	