Freescale Semiconductor
Application Note

Document Number:AN4666
Rev 0, 07/2013

Optimizing MPC564xB/C System Performance Parameters

Effects of varying key system level parameters are measured using sample benchmarks

by: Viktor Fellinger and Steve Mihalik

Contents

1 Introduction....1
2 Architecture and optimization opportunities....1
3 Descriptions of benchmarks used for measurements....8
4 Effect of wait states versus frequency....10
5 Effect of flash BIU line buffer configuration....11
6 Effects of crossbar configuration....13
7 Effect of Branch Target Buffers when enabled....15
8 Effect of Small Data Area....16
9 Effect of crossbar configuration for dual core....17
10 Summary....28

1 Introduction

A common question is, what is the effect of various system level parameters on performance? This application note uses three sample benchmarks to explore the performance effects by varying those parameters.

The configurations tested do not use masters other than one or both cores. No DMA or other masters operate during the benchmarks. These results are useful starting points for users to get a feel on the system parameter effects on MPC5646B/C. The best benchmark is always your code.

2 Architecture and optimization opportunities

2.1 Block diagram

The MPC564xB/C block diagram is shown below. For the benchmarks, memory banks are dedicated to specific cores in the linker file.

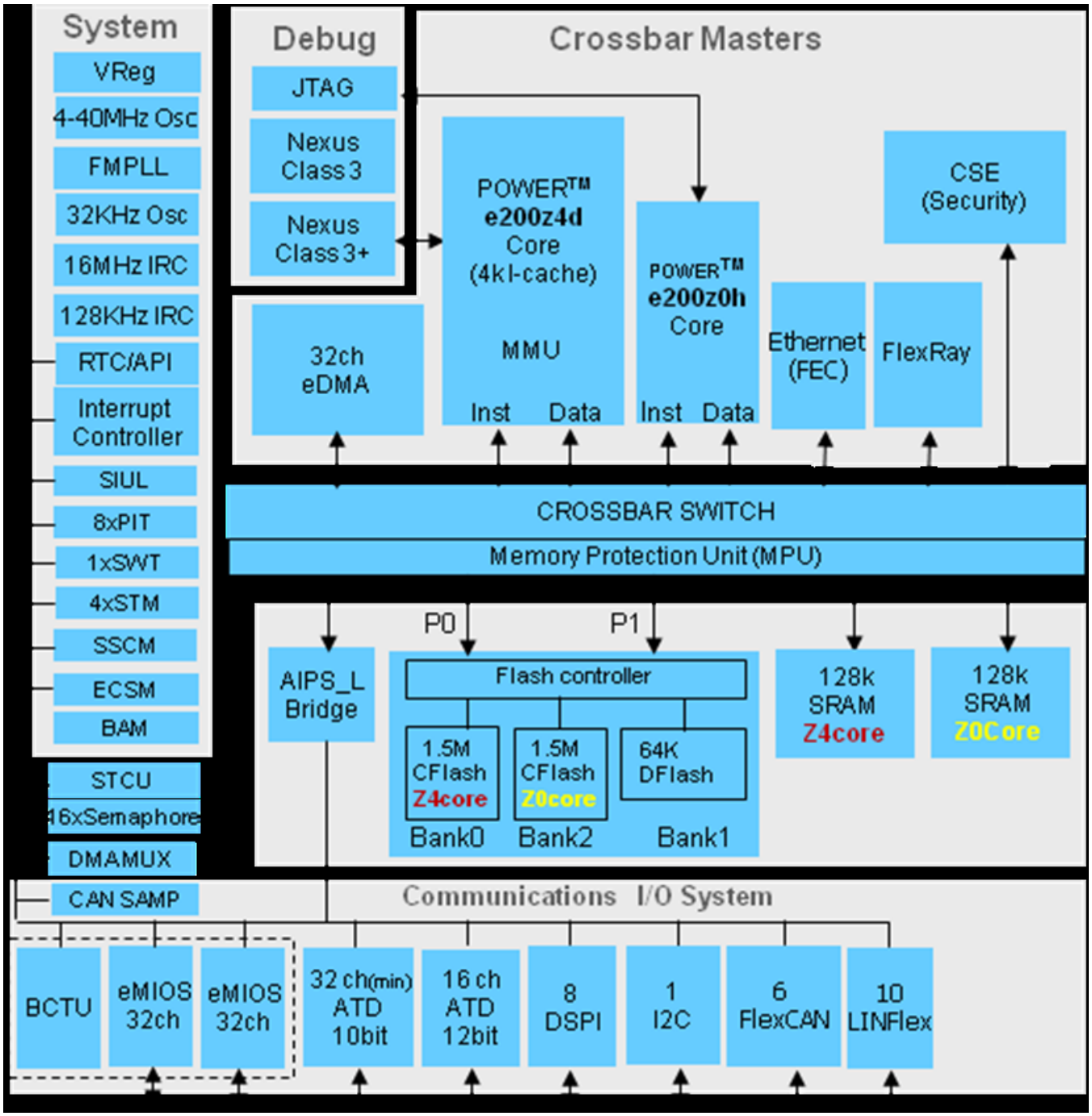

Figure 1. MPC5646C block diagram

2.2 Core comparison

The table below shows the features for the e200z0h and e200z4d cores.

Table 1. Core feature comparison

Feature	e200z0h	e200z4d
cache	no	4 KB
SPE	no	yes
Embedded floating-point (FPU) unit	no	yes
Branch target buffer	1 entry	8 entries
Memory management unit (MMU)	no	16 entries
VLE instruction set	yes	yes
Book-E instruction set	no	yes
AHB system buses	32-bit	64-bit
Independent instruction and data accesses	yes	yes
Dual issue	no	yes
Divider logic for signed and unsigned	5 to 34 clocks	divide in 14 clocks
Hardware multiplier array	8x32 bit	32x32 bit
32-bit single cycle barrel shifter	yes	yes
32-bit mask unit for data masking and insertion	yes	yes
32-bit ALU	yes	yes

2.3 Parameters to be varied for optimization testing

The following parameters varied for single core benchmark testing:

- Flash and RAM wait states versus frequency
- Flash line buffer and prefetching
- Crossbar configurations (priority and parking)
- Branch Target Buffer enabling
- Small Data Area (SDA) size

In addition, the following parameters varied for dual core benchmark testing:

- Crossbar priorities and parking
- Flash block partitioning for software
- RAM block partitioning for data

2.3.1 Flash and RAM wait states

RAM and flash modules need wait states according to the frequency. The trade off to be tested is, what is the impact of a higher frequency that requires an additional wait state? For example, if an additional wait state is needed above 100 MHz, is it better to run at a maximum frequency of 120 MHz with the extra wait state or 100 MHz without it?

The tables below show the wait state requirements.

Table 2. Flash memory read access timing[1]

Symbol		C	Parameter	Conditions[2]		Frequency range	Unit
				Code flash memory	Data flash memory		
f_{READ}	CC	P	Maximum frequency for Flash reading	5 wait states	13 wait states	120–100	MHz
		C		4 wait states	11 wait states	100–80	
		D		3 wait states	9 wait states	80–64	
		C		2 wait states	7 wait states	64–40	
		C		1 wait states	4 wait states	40–20	
		C		0 wait states	2 wait states	20–0	

1. MPC5646C Data Sheet, Rev. 5.1, j08/2012, page 61, Table 28
2. V_{DD}=3.3 V ± 10%/5.0 v ± 10%, T_A = -40 to 125 °C, unless otherwise specified

Table 3. SRAM memory wait states[1]

Frequency	Wait states
>= 64MHz + 4%	1
> 64 MHz + 4%	0

1. MPC5646C Microcontroller Reference Manual, Rev. 4, page 1181, note. RAM wait states are controlled in MUDCR register in the ECSM module.

2.3.2 Flash line buffer and prefetching

Line buffers and prefetching features in the flash controller minimize the effect of wait states. The flash array width is a multiple of the system bus width. When line buffers are enabled, any access to the flash results in the entire line being read and put into a line buffer, regardless of whether the access was for a byte, word, and so on. This makes other data, like next instruction from flash, available to the master without wait states. Hence it makes sense in general to enable line buffers.

Prefetching is the capability of multiple line buffers to work together so that after a transfer from the flash bank to the line buffer completes, additional transfers from the flash bank to a different line buffer will take place concurrently with a master reading the flash data from the first line buffer. The effect for sequential accesses is after the first access, wait states are not used for further contiguous accesses. Prefetching makes sense for sequential accesses, but in general does not help random accessing.

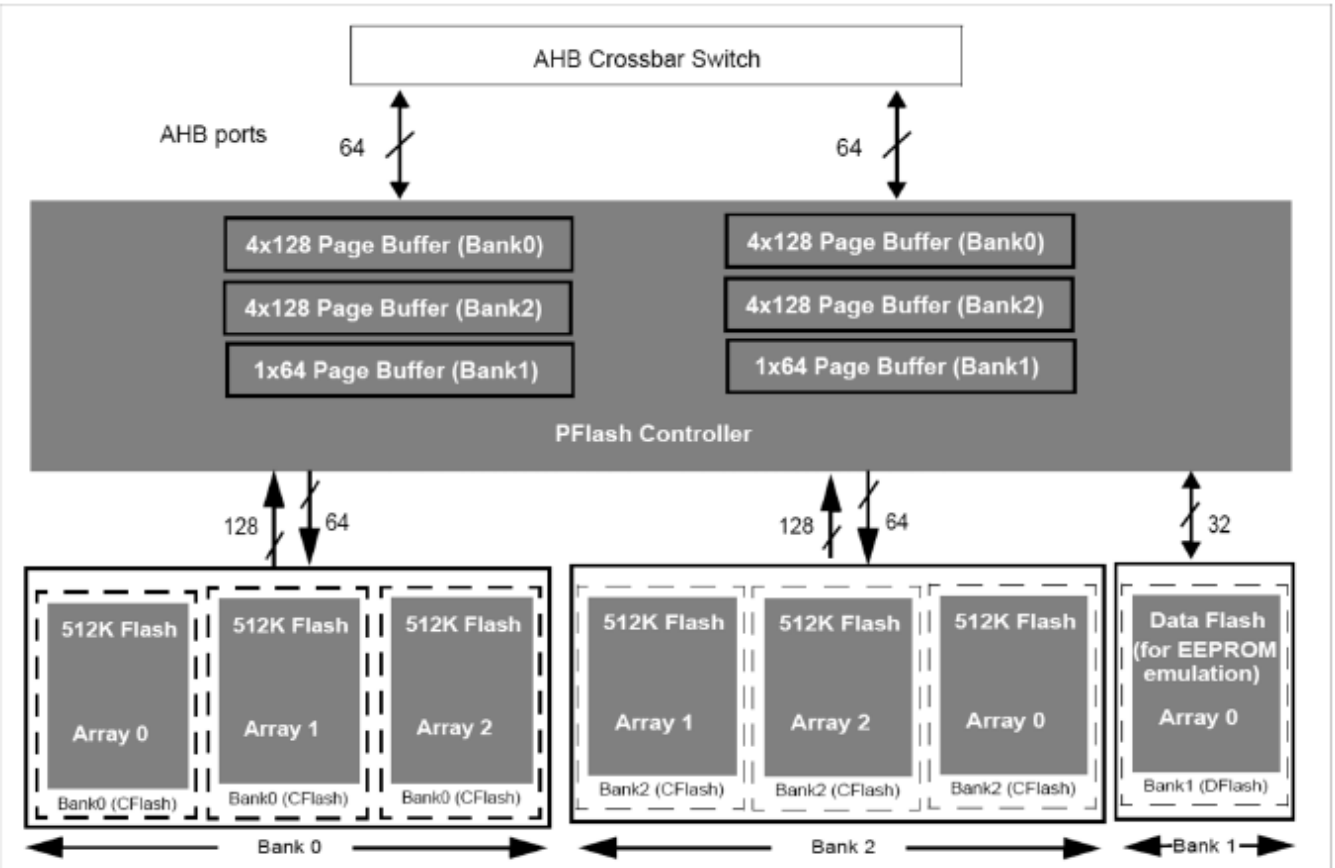

Figure 2. Flash memory architecture

There are two ports to the platform flash controller:

- Port P0 (crossbar port S0) always connects (dedicated) to the e200z4d instruction bus
- Port P1 (crossbar port S1) always connects to the e200z4d data bus and all other masters including e200z0h bus
- Each input (read access) port includes
  - four page (line) read buffers for flash bank 0
  - four page (line) read buffers for flash bank 2
  - one page (line) buffer for flash bank 1 (intended for data)

2.3.3 Crossbar

The crossbar connects modules that can initiate read/write bus access (masters) to modules that do not initiate access by their own (slaves). The crossbar allows simultaneous (concurrent) access between masters and slaves.

The two crossbar parameters affecting performance are:

- **Priority**. Each slave is configured to have a priority for each master. If more than one master simultaneously tries to access a slave, the one that gets access is the one with the higher assigned priority.
- **Parking**. An extra clock is required when a slave is not “parked” at the master requesting access. Slaves are configured to be either:
  - parked at a fixed master,
  - parked at the last master requesting access, or
  - parked at no master (this last option saves a slight amount of power at the expense of performance).

To evaluate the performance effect, benchmarks were run with different priority configurations and different parking configurations.

MPC564xB/C has eight crossbar master ports:

- e200z4d Instruction Fetch
- e200z4d Data Fetch
- e200z0h Instruction Fetch
- e200z0h Data Fetch
- eDMA
- Ethernet (FEC)
- FlexRay
- CSE (security)

MPC564xB/C has five crossbar slave ports:

- Flash controller (2 ports)
- SRAM (2 ports, one per module)
- PBRIDGE (peripheral bridge)

The block diagram for MPC564xB/C crossbar with the master and slave port number assignments is shown below.

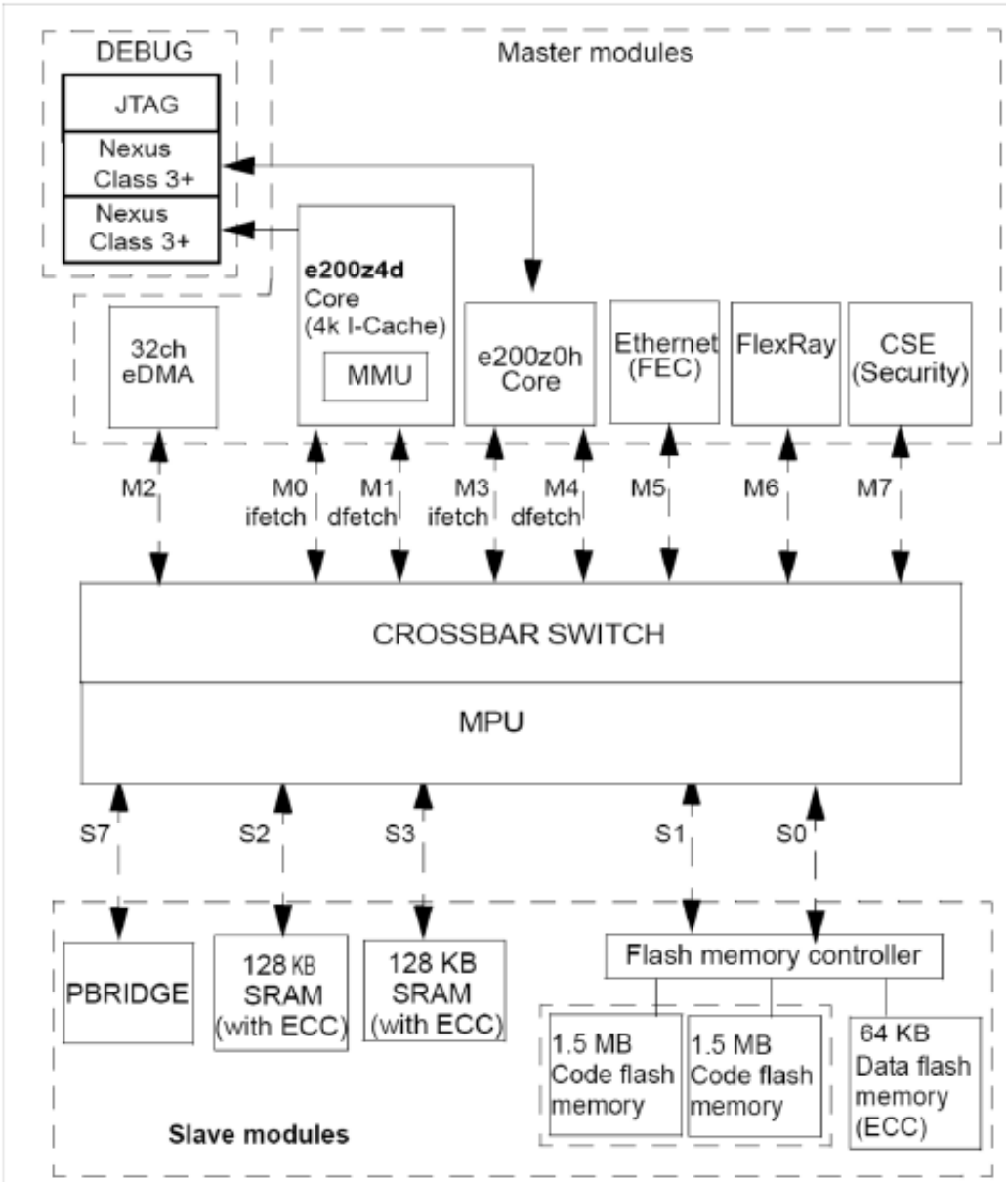

Figure 3. MPC564xB/C crossbar block diagram

2.3.4 Branch target buffer

The instruction fetching mechanism can use a branch target buffer to detect branch instructions early. This branch instruction lookahead scheme allows branch targets to be fetched early, thereby hiding some taken branch bubbles[1].

Out of reset, the contents of the buffers are undefined, so they must be invalidated by software before software enables them. The performance effect will vary with application code, but is often non-trivial.

MPC5646B/C cores have the following number of buffers:

- e200z0h: 1 branch target buffer
- e200z4d: 8 branch target buffers

1. e200z0 Power Architecture(TM) Core Reference Manual, Rev. 0, 4/2008, page 3-3

2.3.5 Small Data Area (SDA) size

The application binary interface (API) used with Qorivva devices defines certain registers to have dedicated uses including:

- GPR2: Small Data Area pointer
  - Typically used for read-only data (ROM data)
  - Linker symbol is .sdata
- GPR13: Small Data Area pointer
  - Typically used for read/write data (RAM data)
  - Linker symbols are .sdata for initialized data, .sbss for zero-initialized data

Using a dedicated general purpose register to create an SDA has the benefit of more efficient memory access: Only one assembly instruction is needed instead of two. If the linker has an SDA size allocation of 0, no SDA is used. If the SDA parameter is set to certain threshold value (that is, sda=1024) then compiler assigns variables of that size or smaller to the SDA area. The ROM and RAM SDA can contain each up to 64 KB of data. Because offset-based addressing on the Power Architecture® processor is signed, an SDA base register points 32 KB past and before the start of the SDA pointer to provide a full 64 KB of addressing. This optimization does not change the basic function of your program.

3 Descriptions of benchmarks used for measurements

Benchmarks offer data points -- the best benchmark is always your application. The following three benchmarks were used to provide three data points for varying of the system parameters. Green Hills compiler was used for all three benchmarks.

Dhrystone benchmarks results were in DMIPS, Benchmarks A and C measured cycles and time in microseconds.

3.1 Dhrystone 2.1: general integer performance

This simple, standard industry benchmark indicates general integer performance. It also includes string and buffer copy operations, but no floating point operations.

Compiler optimizations:

```
# Use faster addressing mode
sda=all

# Optimize for speed
-Ospeed -no-codefactor
-Omax
-inline prologue

# Enable linker optimizations
-Olink

# Stop linker from in-lining function x parameters.
# This could have negative performance side effects on other code fragments.
-no inline trivial

# Optimize application as a whole
-Owholeprogram

# Inform compiler which functions are referenced from assembly so they are not deleted
-external=decode command
-external=sc_tx_handler
```

Table 4. Link file map sections for Dhrystone 2.1 benchmark

Section	Address	Size
.vletext	0x00012000	0x2B14
.vectortab	0x00010000	0x1334
.sdata2	0x00014B1C	0x254
.rodata	0x00014D70	0x60
.sdata	0x40000000	0x24
.bss	0x40000004	0x418
.sbss	0x40000024	0x4ACD
.data	0x40004AF4	0x38
.bss	0x40004B2C	0x9C

3.2 Customer Benchmark A: Array and Math/Logic

This benchmark performs calculations with a relatively high percentage of different operations on array elements. Operations include compare, add, shift, AND, OR and XOR. About 40% of instructions are RAM read/write operations (can be seen as worst case for RAM access).

Code size is much higher on e200z0 than e200z4. e200z4 also has the benefit of the 4 KB instruction cache.

Compiler optimizations include:

- -isel: Use isel instruction
- -speed: Optimize for speed
- SDA = 1024: Allocate 1 KB for SDA

Table 5. Link file map sections for Customer Benchmark A

Section	Address	Size
.vletext	0x000002A4	0x37B06
.ROM.data	0x00001118	4
.ROM.sdata	0x0000111C	0x34
.data	0x40000000	4
.bss	0x40000004	0x201C
.sdata	0x4000041C	0X34
.sbss	0x4000041C	0xCA2
.sdata2	0x40000860	0XD10

3.3 Customer Benchmark C: Calculation Loops

Here math calculations are performed in loops. They are mainly integers such as multiply, add, subtract and divide, but include some floating point multiple and divider operations that are implemented in software for e200z0 core.

The code has a smaller code size which is well suited for the e200z4's 4 KB instruction cache.

Compiler optimizations include:

- -isel: Use isel instruction
- -speed: Optimize for speed
- SDA = 1024: Allocate 1KB for SDA

Table 6. Link File Map Sections for Customer Benchmark C

Section	Address	Size
.vletext	0x000002A4	0xE74
.ROM.data	0x00001118	4
.ROM.sdata	0x0000111C	0
.data	0x40000000	4
.bss	0x40000004	0x418
.sdata	0x4000041C	0
.sbss	0x4000041C	0x444
.sdata2	0x40000860	8

4 Effect of wait states versus frequency

Performance normally increases with frequency. However, it is not a linear relationship. One reason is additional wait states kick in for higher frequencies. The tests here give indications of what performance to expect when moving from executing at one frequency to another.

"Relative Performance" is calculated as the ratio of the metric between the fastest frequency and others:

- Relative Performance (Dhrystone 2.1) = (DMIPS) / (DMIPS at 120 MHz)
- Relative Performance (Benchmarks A, C) = (run time at 120 MHz) / (run time)

For example, from Table 7 the Dhrystone 2.1 Relative Performance for e200z4 frequency of 100 MHz = 291.60 / 249.00 = 85.39%.

Configuration 120 MHz with five flash wait states and one RAM wait state (first row) is considered to deliver 100% performance. The relative performance gives an indication of performance increase (value >100%) or decrease (value <100%) when using different wait state configurations.

Flash wait states are controlled by fields in the Platform Flash Controller, Platform Flash Configuration Register 0 (PFCR0). RAM wait states are controlled by a field in the Error Correction Status Module, Miscellaneous User-Defined Control Register (MUDCR).

Table 7. e200z4 Flash and RAM Wait States versus Frequency Benchmark Results[1].

e200z4 Freq. (MHz)	Flash Wait States	RAM Wait States	Dhrystone 2.1			Benchmark A			Benchmark C		
			DMIPS / MHz	DMIPS	Relative Performance	z4 cycles	z4 run time (µsec)	Relative Performance	z4 cycles	z4 run time (µsec)	Relative Performance
120	5	1	2.43	291.60	100.00%	5956	49.63	100.00%	25816	215.13	100.00%
100	4	1	2.49	249.00	85.39%	5581	55.81	88.93%	25785	257.85	83.43%
80	3	1	2.56	204.80	70.23%	5324	66.55	74.58%	25778	322.23	66.76%
64	2	0	3.02	193.28	66.28%	3831	59.86	82.92%	23452	366.44	58.71%
40	1	0	3.14	125.60	43.07%	3671	91.78	54.08%	23497	587.43	36.62%
20	0	0	3.21	64.20	22.02%	3605	180.25	27.54%	23497	1174.85	18.31%

1. Test conditions:
   - Flash bank 0 used for e200z4
   - RAM module 0 used for e200z4
   - Flash Prefetch buffers: buffers 0,1,2 for instruction, buffer 3 for data
   - SDA size 1 KB

NOTE

Shaded cells indicate performance did not increase at higher frequency that needed an additional wait state.

Table 8. e200z0 Flash and RAM Wait States versus Frequency Benchmark Results[1]

e200z4 Freq. (MHz)	e200z0 Freq.[2] (MHz)	Flash wait States	RAM wait States	Dhrystone 2.1			Benchmark A			Benchmark C		
				DMIPS / MHz	DMIPS	Relative performance	z0 cycles	z0 run time (µsec)	Relative performance	z0 cycles	z0 run time (µsec)	Relative performance
120	60	5	1	1.31	78.60	100.00 %	7083	118.05	100.00 %	64519	1075.32	100.00 %
100	50	4	1	1.41	70.50	89.69%	6901	138.02	85.53%	65872	1317.44	81.62%
80	80	3	1	1.34	107.20	136.39 %	7034	87.93	134.26 %	62288	778.60	138.11 %
64	64	2	0	1.68	107.52	136.79 %	5088	79.50	148.49 %	58846	919.47	116.95 %
40	40	1	0	1.92	76.80	97.71%	4790	119.75	98.58%	56070	1401.75	76.71%
20	20	0	0	2.19	43.80	55.73%	4578	228.90	51.57%	51504	2575.20	41.76%

1. Test conditions:
   - Flash Bank 2 used for e200z0
   - RAM module 0 used for e200z0
   - Flash Prefetch buffers: buffers 0,1,2 for instruction, buffer 3 for data
   - compiler parameter: SDA=1024
2. e200z0 frequency is e200z4 frequency for 80 MHz and below. Above 80 MHz e200z0 frequency is e200z4 frequency/2

Comments and Recommendations:

- Generally performance increases with frequency, but it is a nonlinear relationship.
- Since maximum e200z0 frequency is 80 MHz, e200z0 performance will decrease when e200z4 frequency is above 80 MHz.
- For all benchmarks a better Dhrystone/MHz metric was generally achieved at frequencies <= 64 MHz.
- Performance comparison between 80 MHz and 64 MHz
  - due to RAM wait state at >64 MHz and additional flash wait state overall performance increase is very dependent on application
  - RAM access ratio - that is, Benchmark_A has a very high RAM access rate (~40% of instructions are RAM read/write operations) and delivers better performance at 64 MHz comparing to 80 MHz
- e200z4 specific
  - higher cache hit rates for 4KB instruction cache mitigate impact of additional wait state
  - The higher the cache hit rate the more linear performance can be scaled with frequency (that is, refer to Benchmark_C)

5 Effect of flash BIU line buffer configuration

When line buffers are enabled, they can be configured to allow prefetching for instructions and / or data using controls in the BIUCR register for each port: Controls for line buffers are:

- Instruction Prefetch Enable (IPFE)

- Data Prefetch Enable (DPFE)
- Number of lines allocated to instruction, data or any fetches

How many buffers are configured for instructions and data is determined by the Buffer Configuration field for each port.

Code was only executed in these tests on the e200z4 core. Buffer configurations used in testing were identical for both ports. Since Port P0 is always connected to the e200z4 instruction bus, no data will be read from this port so in some configurations buffers 2 and 3 or buffer 2 will not be used.

"Relative Performance" is calculated as the ratio of the first configuration in the table compared to the other two below it:

- Relative Performance = (Run Time for Buffers 1,2 instruction & buffers 3,4 data) / (Run Time for different)

For example, from the table below, the e200z4 Relative Performance for line buffer configuration of any access is 51.36 / 50.37 = 101.97%.

Table 9. e200z4 Benchmark C flash BIU line buffer configuration results[1]

Flash prefetch buffer configuration (applies to both ports PO and P1)	e200z4 at 120 MHz			e200z0 64 MHz		
	z4 cycles	z4 run time (µsec)	Relative performance	z0 cycles	z0 run time (µsec)	Relative performance
Buffers 0, 1: Instruction Buffers 2, 3: data	6163	51.36	100.00%	7239	120.65	100.00%
Buffers 1, 2, 3: Instruction Buffer 3: data	6047	50.39	101.92%	7076	117.93	102.30%
All buffers: any access	6044	50.37	101.97%	7025	117.08	103.05%

1. Test conditions:
   - Flash bank 0 used for e200z4, flash bank 2 used for e200z0
   - RAM module 0 used for e200z4. RAM module 1 used for e200z0
   - Instruction and data prefetching enabled in all cases (IPFE=DPFE=1)
   - compiler parameter: SDA=1024
   - Wait states for 120 MHz: 5 flash, 1 RAM
   - Wait states for 64 MHz: 3 flash, 1 RAM
   - Prefetch limit (PFLIM) = 1- which means the referenced line is prefetched on a buffer miss, or the next sequential page is prefetched on a buffer hit (if not already present), that is, **prefetch on miss or hit**.

Table 10. e200z4 Benchmark A flash BIU line buffer configuration results[1]

Flash prefetch buffer configuration (applies to both ports PO and P1)	e200z4 at 120 MHz			e200z0 64 MHz		
	z4 cycles	z4 run time (µsec)	Relative performance	z0 cycles	z0 run time (µsec)	Relative performance
Buffers 0, 1: Instruction Buffers 2, 3: data	25815	218.13	100.00%	65854	1097.57	100.00%

Table continues on the next page...

Table 10. e200z4 Benchmark A flash BIU line buffer configuration results[1] (continued)

Buffers 1, 2, 3: Instruction Buffer 3: data	25815	218.13	100.00%	64687	1078.12	101.80%
All buffers: any access	25815	218.13	100.00%	64498	1074.97	102.10%

1. Test conditions:
   - Flash bank 0 used for e200z4, flash bank 2 used for e200z0
   - RAM module 0 used for e200z4. RAM module 1 used for e200z0
   - Instruction and data prefetching enabled in all cases (IPFE=DPFE=1)
   - compiler parameter: SDA=1024
   - Wait states for 120 MHz: 5 flash, 1 RAM
   - Wait states for 64 MHz: 3 flash, 1 RAM
   - Prefetch limit (PFLIM) = 1- which means the referenced line is prefetched on a buffer miss, or the next sequential page is prefitched on a buffer hit (if not already present), that is, **prefetch on miss or hit**.

Comments and Recommendations:

- Test results showed performance increased with more line buffers available for prefetching instructions. Generally this makes sense since prefetching only benefits sequential accesses, such as instructions.
- Optimal prefetch buffer configuration is dependent on several application specific settings/parameters:
  - Core cache (for e200z4 only) hit rate – core only fetches instructions from Flash in case of cache miss
  - relative number of conditional and unconditional branches
  - taken versus not-taken branches
  - Branch Prediction Buffer (BTB) hits in e200x core
  - hit rate in 4x128 bit flash prefetch buffer
  - access frequency to constant data stored in flash
- Impact of prefetch buffer configuration on overall performance may differ from application to application
- Run application or critical sections using different prefetch buffer configurations to evaluate settings delivering best performance

6 Effects of crossbar configuration

Benchmarks were run on both e200z4 and e200z0 for different crossbar configuration parameters of:

- **Parking of slave at master**. After a slave is accessed by any master, it can be “parked” at an assigned master for the next access. When the slave is accessed again, the access time is faster if the slave was parked at that master.
- **Master priorities for slaves**. If two or more masters attempt to access a slave at the same time, arbitration is done as to who gets in first based on the software configured priority scheme. This testing was done using fixed priorities, round-robin.

The “Optimized Configuration” in the tables use parking and master priority assignments to the masters accessing them in these benchmark tests.

CAUTION

This benchmark only uses two masters for accessing memory: the two cores. The e200z0 core instruction bus will need a significant amount of its flash bank (Port S1) bandwidth. Testing should be done in an application to verify if the e200z0 instruction bus traffic does not “starve” other masters needing the same slave flash block. Adjust priorities as needed.

Table 11. Crossbar settings used for benchmarks

Slave port	Crossbar slave port's parking		Crossbar slave port's highest priority master	
	Default configuration	Optimized configuration	Default configuration	Optimized configuration
Port S0 (Flash bank for z4 Instr Bus only)	M0 - Z4 Instr Bus	M0 - Z4 Instr Bus	M0 - Z4 Instr Bus	M0 - Z4 Instr Bus
Port S1 (Flash bank for all masters)	M0 - Z4 Instr Bus	M3 - Z0 Instr Bus	M0 - Z4 Instr Bus	M3 - Z0 Instr Bus
Port S2 (SRAM0 module)	M0 - Z4 Instr Bus	M1 - Z4 Data Bus	M0 - Z4 Instr Bus	M1 - Z4 Data Bus
Port S3 (SRAM1 module)	M0 - Z4 Instr Bus	M4 - Z0 Data Bus	M0 - Z4 Instr Bus	M4 - Z0 Data Bus

Results are shown below. The default XBAR configuration is considered to deliver 100%. The relative performance gives an indication of performance increase (value >100%) or decrease (value <100%) when using optimized XBAR settings.(that is, 107.52% means that optimized settings deliver 7.52% better performance which corresponds to 7.52% lower runtime).

"Relative Performance" is calculated as the ratio of the metric between the optimized and default configurations:

- Relative Performance (Dhrystone 2.1) = (Optimized DMIPS) / (Default DMIPS)
- Relative Performance (Benchmarks A= (Default Run Time) / (Optimized Run Time)

For example, from the table below the Dhrystone 2.1 Relative Performance for e200z4 frequency of 120 MHz = 291.60 / 271.20 = 107.52%.

Table 12. Crossbar testing results for e200z4

e200Z4 Freq	Flash WS	RAM WS	Dhrystone benchmark					Benchmark A				
			XBAR default		Optimized configuration		Relative performance	XBAR default		Optimized configuration		Relative performance
			Z4 DMIPS / MHz	Z4 DMIPS	Z4 DMIPS / MHz	Z4 DMIPS		Z4 Cycles	Z4 Run Time	Z4 Cycles	Z4 Run Time	
120	5	1	2.26	271.20	2.43	291.60	107.52 %	6144	51.20	5956	49.63	103.16 %
100	4	1	2.31	231.00	2.49	249.00	107.79 %	5802	58.02	5581	55.81	103.96 %
80	3	1	2.37	189.60	2.56	204.80	108.02 %	5570	69.63	5324	66.55	104.62 %
64	2	0	2.75	176.00	3.02	193.28	109.82 %	4061	63.45	3831	59.86	106.00 %
40	1	0	2.83	113.20	3.14	125.60	110.95 %	3912	97.80	3671	91.78	106.56 %
20	0	0	2.91	58.20	3.18	63.60	109.28 %	3855	192.75	3605	180.25	106.93 %

Table 13. Crossbar test results for e200z0

e200Z4 Freq	Z0:Z4 Clock	Flash WS	RAM WS	Dhrystone Benchmark					Benchmark A				
				XBAR default		Optimized configuration		Relative performance	XBAR default		Optimized configuration		Relative performance
				Z0 DMIPS/MHz	Z0 DMIPS	Z0 DMIPS/MHz	Z0 DMIPS		Z0 Cycles	Z0 Run Time	Z0 Cycles	Z0 Run Time	
120	1:2	5	1	1.31	78.60	1.31	78.60	100.00 %	7083	118.05	7083	118.05	100.00 %
100	1:2	4	1	1.41	70.50	1.41	70.50	100.00 %	6901	138.02	6901	138.02	100.00 %
80	1:1	3	1	1.23	107.20	1.34	107.20	108.94 %	7594	94.93	7034	87.93	107.96 %
64	1:1	2	0	1.59	107.52	1.68	107.52	105.66 %	5588	87.31	5088	79.50	109.83 %
40	1:1	1	0	1.77	76.80	1.92	76.80	108.47 %	5344	133.60	4790	119.75	111.57 %
20	1:1	0	0	1.96	43.80	2.19	43.80	111.73 %	5160	258.00	4578	228.90	112.71 %

Comments and Recommendations:

- Master priority and parking on slave port settings have significant impact on single and dual core performance.
- One additional arbitration cycle will delay access to slave when PARK setting does not match master ID accessing the slave.
- For each application, XBAR settings need to be adjusted according to priority and performance needs.
- 0-12% Single Core Performance increase when XBAR settings are optimized for Flash and RAM ports.
- Highest priority on the RAM block containing FlexRay buffers should be granted to FlexRay master. Failure to do so may result in FlexRay timeout errors.
- More comments on XBAR settings can be found in dual core benchmarks.

7 Effect of Branch Target Buffers when enabled

MPC564xB/C Branch Target Buffers (BTB) are different by core:

- e200z4 8x branch target buffer entries
- e200z0 1x branch target buffer entry

For e200z4, there is a field for Branch Target Buffer Allocation Control, BUSCSR[BALLOC]. This controls if buffers are enabled for all branches, forward branches, backward branches or none. These benchmark tests used the default setting of enable buffers for all branches.

“Relative Performance” shows the improvement of enabling BTB for that core:

- Relative Performance = (Run time with BTB disabled) / (Run time with BTB enabled)

For example, from the table below the relative performance of enabling BTB for the e200z4 at 120 MHz is 50.48 / 50.39 = 100.18%, indicating a performance increase (value > 100%) after enabling BTB.

Table 14. Branch Target Buffer testing for e200z41

e200Z4 Freq	Flash WS	RAM WS	Benchmark A					Benchmark C				
			BTB Disabled (default)		BTB Enabled		Relative Performance	BTB Disabled (default)		BTB Enabled		Relative Performance
			Z4 Cycles	Z4 Run Time	Z4 Cycles	Z4 Run Time		Z4 Cycles	Z4 Run Time	Z4 Cycles	Z4 Run Time	
120	5	1	6058	50.48	6047	50.39	100.18 %	28157	234.64	25815	215.13	109.07 %
64	2	0	3840	60.00	3831	59.86	100.16 %	25768	402.63	23452	366.44	109.88 %

1. Test conditions:
   - Flash bank 0 used for e200z4, flash bank 2 used for e200z0
   - RAM module 0 used for e200z4. RAM module 1 used for e200z0
   - compiler parameter: SDA=1024

Table 15. Branch Target Buffer testing for e200z01

e200Z0 Freq	Flash WS	RAM WS	Benchmark A					Benchmark C				
			BTB Disabled (default)		BTB Enabled		Relative Performance	BTB Disabled (default)		BTB Enabled		Relative Performance
			Z0 Cycles	Z0 Run Time	Z0 Cycles	Z0 Run Time		Z0 Cycles	Z0 Run Time	Z0 Cycles	Z0 Run Time	
60	5	1	7086	118.10	7076	117.93	100.14 %	69245	1154.08	64687	1078.12	107.05 %
64	2	0	5096	79.63	5088	79.50	100.16 %	60866	951.03	58846	919.41	103.43 %

1. Test conditions:
   - Flash bank 0 used for e200z4, flash bank 2 used for e200z0
   - RAM module 0 used for e200z4. RAM module 1 used for e200z0
   - compiler parameter: SDA=1024

Comments and Recommendations:

- Enabling BTB improved performance. The improvement was non-trivial in one of the two benchmarks tested.
- Typically BTB delivers better performance for “prediction friendly“ code such as:
  - if-statements delivering the same condition multiple times (>4 times)
  - long loops (>4 iterations)
- Run application or critical sections with BTB OFF & ON to evaluate settings delivering best performance.

8 Effect of Small Data Area

Benchmarks were run on both e200z4 and e200z0 with SDA optimization turned on (sda=1024) and off (sda=0). When turned on, variables or arrays whose size is less than 1024 will be placed in the SDA section so it can be accessed with one assembly instruction instead of two.

The default configuration “SDA not optimized” is considered to deliver 100%. The relative performance gives an indication of performance increase (value >100%) or decrease (value <100%) after enabling SDA optimization (that is, in first row 103.26% means that SDA optimization enabled delivers 3.26% higher performance which corresponds to 3.26% lower runtime).

"Relative Performance" is calculated as the ratio of the metric between the fastest frequency and others:

- Relative Performance (Benchmarks A, C) = (run time not optimized) / (run time optimized)

For example, from the table below the Benchmark A relative performance for e200z4 frequency of 120 MHz = 51.25 / 49.63= 103.26%.

Table 16. SDA testing on e200z4

e200Z4 Freq	Flash WS	RAM WS	Benchmark A					Benchmark C				
			SDA not optimized		SDA optimized		Relative performance	SDA not optimized		SDA optimized		Relative performance
			Z4 Cycles	Z4 Run Time	Z4 Cycles	Z4 Run Time		Z4 Cycles	Z4 Run Time	Z4 Cycles	Z4 Run Time	
120	5	1	6150	51.25	6144	49.63	103.26 %	25782	214.85	25816	215.13	99.87%
100	3	1	5437	67.96	5570	66.55	102.12 %	26133	326.66	25778	322.23	101.38 %
80	2	0	3946	61.66	4061	59.86	103.00 %	23831	372.36	23452	366.44	101.62 %

Table 17. SDA testing on e200z0

e200Z0 Freq	Z4:Z0 Clock	Benchmark A					Benchmark C				
		SDA not optimized		SDA optimized		Relative Performance	SDA not optimized		SDA optimized		Relative performance
		Z0 Cycles	Z0 Run Time	Z0 Cycles	Z0 Run Time		Z0 Cycles	Z0 Run Time	Z0 Cycles	Z0 Run Time	
60	1:2	7361	122.68	7083	118.05	103.92%	67504	1125.07	64519	1075.32	104.63%
80	1:1	7300	91.25	7034	87.93	103.78%	64860	810.75	62288	778.60	104.13%
64	1:1	5324	88.73	5088	84.80	104.64%	61464	960.38	58846	919.47	104.45%

Conclusion:

- Performance increases due to Small Data Area Optimization is generally 1–5%
- Use SDA optimization as a default setup in your application
- Please note - SDA area is limited to 64 KB

9 Effect of crossbar configuration for dual core

The following configurations are expected user cases for the various memory size implementations of MPC564xB/C. For each of the CPU frequencies tested, the wait states for flash and RAM were set as in prior tests. A summary of the three configurations tested is in the following table.

Table 18. Summary of dual core crossbar configurations used for testing

Configuration	Memory contents				Crossbar configuration: parking & higher priority master			
	Flash Bank 0	Flash Bank 2	RAM Module 0	RAM Module 1	Port S0 (For all flash banks)	Port S1 (For all flash banks)	Port S2 (For RAM Module 0)	Port S3 (For RAM Module 1)
1	e200z4 instructions & e200z4 data constants	e200z0 instructions & e200z0 data constants	e200z4 data	e200z0 data	e200z4 instruction bus (fixed)	Benchmark Tests: e200z4 data bus vs. e200 instruction bus	e200z4 data bus	e200z0 data bus
2	e200z4 instructions & e200z4 data constants	e200z0 instructions & e200z0 data constants	e200z4 data & e200z0 data	Not used		e200z0 instruction bus	Benchmark Tests: e200z4 data bus vs. e200z0 data bus	Not Used
3	e200z4 instructions & e200z0 instructions & e200z0 data constants & e200z4 data constants	Not used	e200z4 data & e200z0 data			Benchmark Tests: Port S1: e200z4 data bus & Port S2: e200z4 data bus vs. Port S1: e200z0 instr. bus & Port S2: e200z0 data bus		

Benchmark tests are run at different frequencies. The measured results are compared with prior single core execution results earlier in this application note. A “Relative Performance” calculation is done.

“**Relative Performance**” compares performance of running when running in the **dual core configuration** (both cores are executing code) versus their **single core configuration** (standalone - only one core runs at a time.)

Dual core configuration DMIPS and run time measurements are in tables in this section. Single core configuration DMIPS and run time measurements are in prior sections.

For Dhrystone benchmarks relative performance is calculated from instructions per second as follows:

- DMIPS Relative Performance (single core) = core x DMIPS (dual core) / core x DMIPS (single core)
- DMIPS Relative Performance (dual core) = [z4 DMIPS (z4 dual core) + z0 DMIPS (z0 dual core)] / [z4 DMIPS (single core) + z0 DMIPS (single core)]

For run time benchmarks (Benchmarks A and C), relative performance is calculated from time measurements as follows:

- Run Time Relative Performance (single core) = core x run time (single core) / core x run time (dual core)
- Run Time Relative Performance (dual core) = [z4 run time (single core) + z0 run time (single core)] / [z4 run time (dual core) + z0 run time (dual core)]

9.1 Configuration 1: 2 flash banks, 2 SRAM modules

Configuration 1 uses the settings in Figure 4. One flash bank is dedicated to all e200z4 accesses and the other for all e200z0 accesses. Similarly e200z4 RAM is located in one SRAM module, and e200z0 RAM is in the other SRAM module.

Tests: Crossbar Port S1 (Flash Port 1) priority and parking for master e200z4 data bus versus master e200z0 instruction bus for different frequencies.

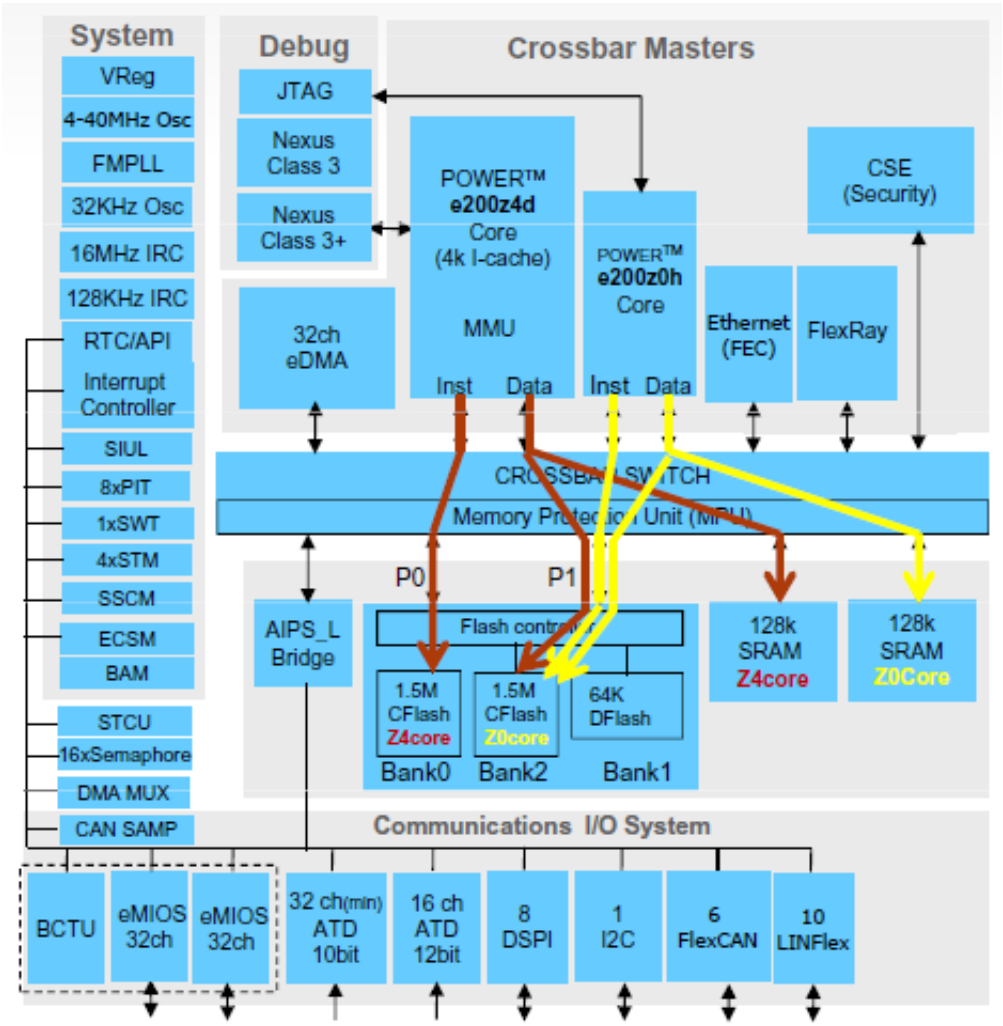

Figure 4. Instruction and Data Paths for configuration 1

Dhrystone Relative Performance Calculations Example: Per Configuration 1 Dhrystone Results table first XBAR Flash Port 1 Configuration, the “z4 Relative Performance” at 120 MHz is:

290.40 (120 Mhz z4 DMIPS dual core configuration per Table 19) /
291.60 (120 Mhz z4 DMIPS single core confuration per Table 7)= 99.59 %.

Using the same table and XBAR Flash Port 1 Configuration, the “Dual Core Relative Performance” at 120 and 60 MHz is:

357.60 DMIPS (120/60 MHz dual core per Table 19: sum of 290.40 z4 DMIPS + 67.20 z0 DMIPS) /
370.20 DMIPS (120/60 MHz single core per Tables 7, 8: sum of 291.60 z4 DMIPS + 78.60 z0 DMIPS)
= 96.60 %.

Effect of crossbar configuration for dual core

Single core performance (z4 runs standalone or z0 runs standalone per Table 7 or 8) is considered to deliver 100% performance. Benchmarks A and C do not contain any constant data in Flash, and so benchmarks were not run. The dual core performance is equal to the sum of single core standalone performance z0+z4.

Table 19. Configuration 1 Dhrystone Results

CPU freq. z4/z0 (MHz)	Highest XBAR Priority for Flash Port 1: z4 Data Bus							Highest XBAR Priority for Flash Port 1: z0 Instruction Bus						
	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (single core)	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (single core[1])	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)
120/60	290.40	99.59 %	67.20	85.50 %	370.20	357.60	96.60 %	289.20	99.18 %	73.80	93.89 %	370.20	363.00	98.06 %
100/50	248.00	99.60 %	59.50	84.40 %	319.50	307.50	96.24 %	224.00	89.96 %	66.50	94.33 %	319.50	290.50	90.92 %
80/80	203.20	99.22 %	98.40	91.79 %	312.00	301.60	96.67 %	167.20	81.64 %	103.20	96.27 %	312.00	270.40	86.67 %
64/64	192.00	99.34 %	99.84	92.86 %	300.80	291.84	97.02 %	117.12	81.64 %	107.52	100.00 %	300.80	224.64	74.68 %
40/40	125.20	99.68 %	72.80	94.79 %	202.40	198.00	97.83 %	75.60	60.19 %	75.20	97.92 %	202.40	150.80	74.51 %
20/20	63.40	99.69 %	41.80	95.43 %	107.40	105.20	97.95 %	43.60	68.55 %	42.60	97.26 %	107.40	86.20	80.26 %

1. Data copied from Table 7 and Table 8

NOTE

See Tables 7, 8 for single core configuration DMIPS numbers. Shaded cells indicate higher dual core relative performance.

Comments and Recommendations:

- Optimal Crossbar Port S1 setting depended on frequency. For these benchmarks, having a higher priority on Port S1 for e200z0 resulted in higher dual core performance at the highest frequency.
- When using Configuration 1 please make sure that XBAR slave ports are configured to grant higher priority and parking for cores shown below:
  - Crossbar Port S0 (Flash Port0) -> z4 Instr Bus
  - Crossbar Port S2 (RAM Block 0) -> z4 Instr Bus
  - Crossbar Port S3 (RAM Block 1) -> z0 Data Bus
- Performance decreases for dual core versus single core performance due to access conflicts on Crossbar Port S1 (Flash Port1) between
  - z4 data bus (constant data stored in flash)
  - z0 instruction bus
  - z0 data bus (constant data stored in flash)
- To evaluate best system performance, also consider:
  - placing constant data to RAM rather then flash
  - accesses from other masters to RAM block 0 and 1 (that is, DMA, FlexRay)

- please note that it may be needed to grant FlexRay the highest priority on the RAM block containing massage buffers to avoid FlexRay timing violations
- delay due to accesses to common peripheral bridge (mainly for the core with lower priority on XBAR PBRIDGE slave port)
- access to peripherals running on divided clock (output of peripheral clock divider) will slow down overall execution time as compared to a configuration with non-divided peripheral clock

9.2 Configuration 2: 2 Flash Banks, 1 SRAM Module

Configuration 2 is the same as Configuration 1 except only one SRAM module is used, so both cores have their data located in that block.

Tests: Crossbar Port S2 (RAM Module 0) priority and parking for master e200z4 data bus versus master e200z0 data bus for different frequencies.

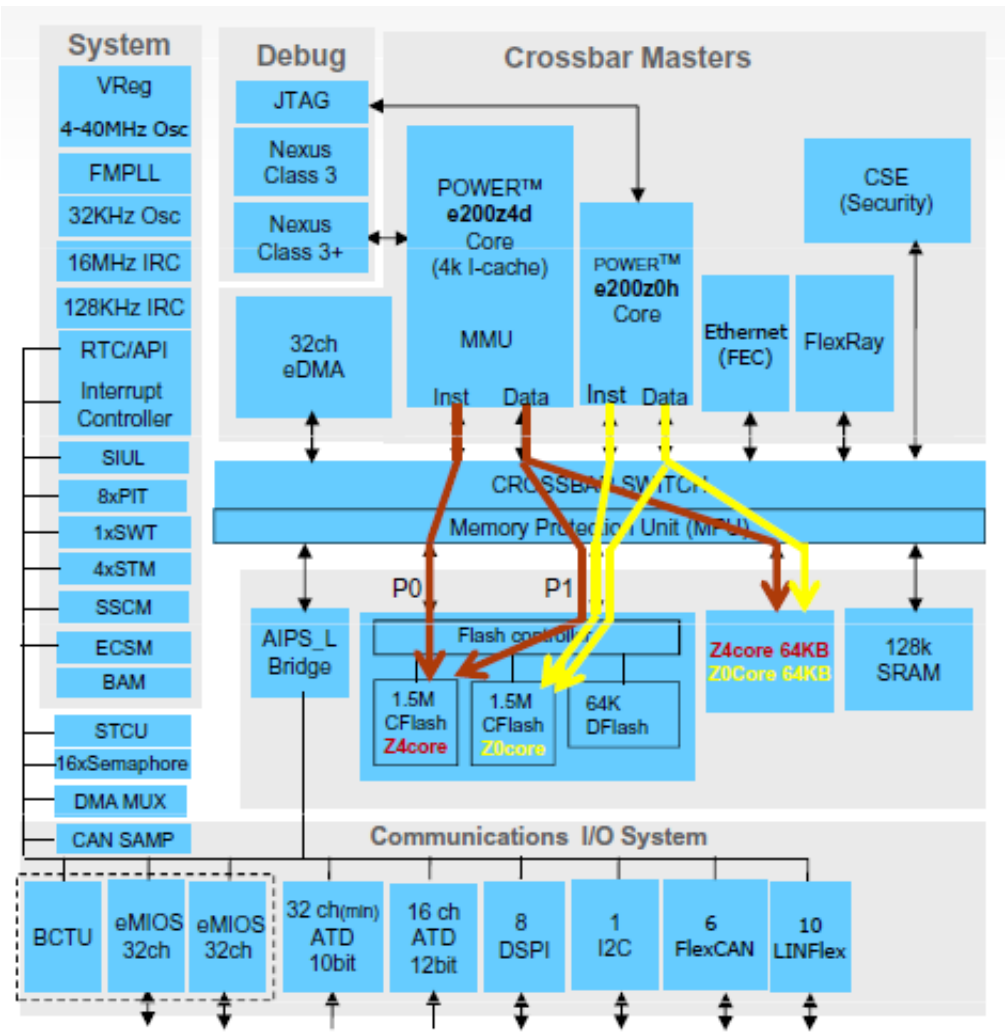

Figure 5. Instruction and Data Paths for configuration 3

Dhrystone Relative Performance Calculations Example: Per Configuration 2 Dhrystone Results table first XBAR Flash Port 1 Configuration, the “z4 Relative Performance” at 120 MHz is:

290.40 (120 MHz z4 DMIPS dual core configuration per Table 20) /
291.60 (120 MHz z4 DMIPS single core configuration per Table 7)= 99.59 %.

Using the same table and XBAR Flash Port 1 Configuration, the “Dual Core Relative Performance” at 120 and 60 MHz is:

354.60 DMIPS (120/60 MHz dual core per Table 20: sum of 290.40 z4 DMIPS + 64.20 z0 DMIPS) /
370.20 DMIPS (120/60 MHz single core per Tables 7, 8: sum of 291.60 z4 DMIPS + 78.60 z0 DMIPS)
= 95.79 %.

Table 20. Configuration 2 Dhrystone Results

CPU freq. z4/ z0 (MHz)	XBAR RAM 0 Port Configuration: z4 Data Bus has Higher Priority						XBAR RAM 0 Port Configuration: z0 Data Bus has Higher Priority					
	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)
120/60	290.40	99.59%	64.20	81.68%	354.60	95.79%	278.40	95.47%	73.80	93.89%	352.20	95.14%
100/50	230.00	92.37%	61.50	87.23%	291.50	91.24%	220.00	88.35%	66.50	94.33%	286.50	89.67%
80/80	178.40	87.11%	80.80	75.37%	259.20	83.08%	166.40	81.25%	98.40	91.79%	264.80	84.87%
64/64	174.72	90.40%	82.56	76.79%	257.28	85.53%	104.32	53.97%	103.68	96.43%	208.00	69.15%
40/40	106.00	84.39%	56.40	73.44%	162.40	80.24%	71.60	57.01%	72.80	94.79%	144.40	71.34%
20/20	62.80	98.74%	34.00	77.63%	96.80	90.13%	39.20	61.64%	40.00	91.32%	79.20	73.74%

NOTE

See Table 7 and Table 8 for single core configuration DMIPS numbers. Shaded cells indicate higher dual core relative performance.

Run Time Relative Performance Calculations Example: Per Configuration 2 Benchmark A Results table, first XBAR Flash Port 1 Configuration, the “z4 Relative Performance” at 120 MHz is:

49.63 (120 Mhz z4 single core configuration per Table 7) /
50.94 (120 Mhz z4 dual core configuration per Table 21)
= 97.43%.

Using the same table and XBAR Flash Port 1 Configuration, the “Dual Core Relative Performance” at 120 and 60 MHz is:

167.68 (120/60 MHz single core per Tables 7, 8: sum of 49.63 z4 +118.05 z0 run times) /
270.14 (120/60 MHz dual core per Table 21: sum of z4 + z0 run times)
= 62.07 %.

Table 21. Configuration 2 Benchmark A Results.

CPU freq. z4/ z0 (MHz)	XBAR RAM 0 Port Configuration: z4 Data Bus has Higher Priority	XBAR RAM 0 Port Configuration: z0 Data Bus has Higher Priority

Table continues on the next page...

Table 21. Configuration 2 Benchmark A Results. (continued)

	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)
120/60	50.94	97.43%	219.20	53.85%	270.14	62.07%	57.79	85.88%	142.13	83.06%	199.93	83.87%
100/50	57.65	96.81%	274.74	50.24%	332.39	58.31%	66.91	83.41%	168.66	81.83%	235.57	82.28%
80/80	73.05	91.10%	212.23	41.43%	285.28	54.15%	120.76	55.11%	104.30	84.30%	225.06	68.64%
64/64	63.50	94.27%	165.66	47.99%	229.16	60.81%	83.84	71.39%	89.41	88.92%	173.25	80.44%
40/40	98.10	93.55%	267.00	44.85%	365.10	57.94%	159.70	57.47%	137.00	87.41%	296.70	71.29%
20/20	193.60	93.10%	529.00	43.27%	722.60	56.62%	324.00	55.63%	264.70	86.48%	588.70	69.50%

NOTE

See Table 7 and Table 8 for Single Core configuration run time numbers. Shaded cells indicate higher dual core relative performance.

Table 22. Configuration 2 Benchmark C Results

CPU freq. z4/ z0 (MHz)	XBAR RAM 0 Port Configuration: z4 Data Bus has Higher Priority						XBAR RAM 0 Port Configuration: z0 Data Bus has Higher Priority					
	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)
120/60	215.58	99.79%	1142.02	94.16%	1357.60	95.05%	217.33	98.99%	1096.62	98.06%	1313.94	98.21%
100/50	258.71	99.67%	1422.98	92.58%	1681.69	93.67%	261.07	98.77%	1341.88	98.18%	1602.95	98.27%
80/80	325.13	99.11%	918.10	84.81%	1243.23	88.55%	335.40	96.07%	838.53	92.85%	1173.93	93.77%
64/64	368.64	99.40%	984.22	93.42%	1352.86	95.05%	375.14	97.68%	936.81	98.15%	1311.95	98.01%
40/40	589.53	99.64%	1507.30	93.00%	2096.83	94.87%	601.20	97.71%	1438.60	97.44%	2039.80	97.52%
20/20	1179.60	99.60%	2788.40	92.35%	3968.00	94.51%	1179.75	99.58%	2765.20	93.13%	3944.95	95.06%

NOTE

See Tables 7, 8 for Single Core configuration run time numbers. Shaded cells indicate higher dual core relative performance.

Comments and Recommendations:

- Configuration 2 delivers lower performance comparing to Configuration 1 due to common RAM array for e200z4 and e200z0 data sections.
- Overall dual core performance is highly dependent on code being executed by each core as well as XBAR RAM array priority and parking settings.
  - Benchmarks with more RAM variable usage (Benchmarks A and C) had higher dual core performance when the e200z0 data bus had higher crossbar priority on the shared RAM module.
  - Use different XBAR settings to balance run time between e200z4 and e200z0.
  - Benchmark A shows a worst case scenario where both cores are executing an algorithm with high degree of RAM read/write operations (~40% of all instructions) combined with non-optimal XBAR configuration.
- To estimate overall performance take into account access to common peripheral bridge (mainly for the core with lower priority on XBAR PBRIDGE slave port) and accesses from other masters (DMA, FlexRay)
  - please note that it may be needed to grant FlexRay the highest priority on the RAM block containing massage buffers to avoid FlexRay timing violations
- Please note that access to peripherals running on divided clock (output of peripheral clock divider) will slow down overall execution time comparing to a configuration with non-divided peripheral clock

9.3 Configuration 3: 1 Flash Bank, 1 SRAM Module

Configuration 3 uses the one flash block and one SRAM module for both cores. This testing compared two configurations. One is the e200z4 that has higher priority (and parking) on the single flash port and single SRAM port. In the other configuration the e200z0 has higher priority (and parking) on both of those same ports.

Tests: For Crossbar Port S0 (Flash Port 0) and Crossbar Port S2 (RAM Module 0), test higher priority and parking for master e200z4 instruction bus and e200z4 data bus versus master e200z0 instruction bus and e200z0 data bus for different frequencies.

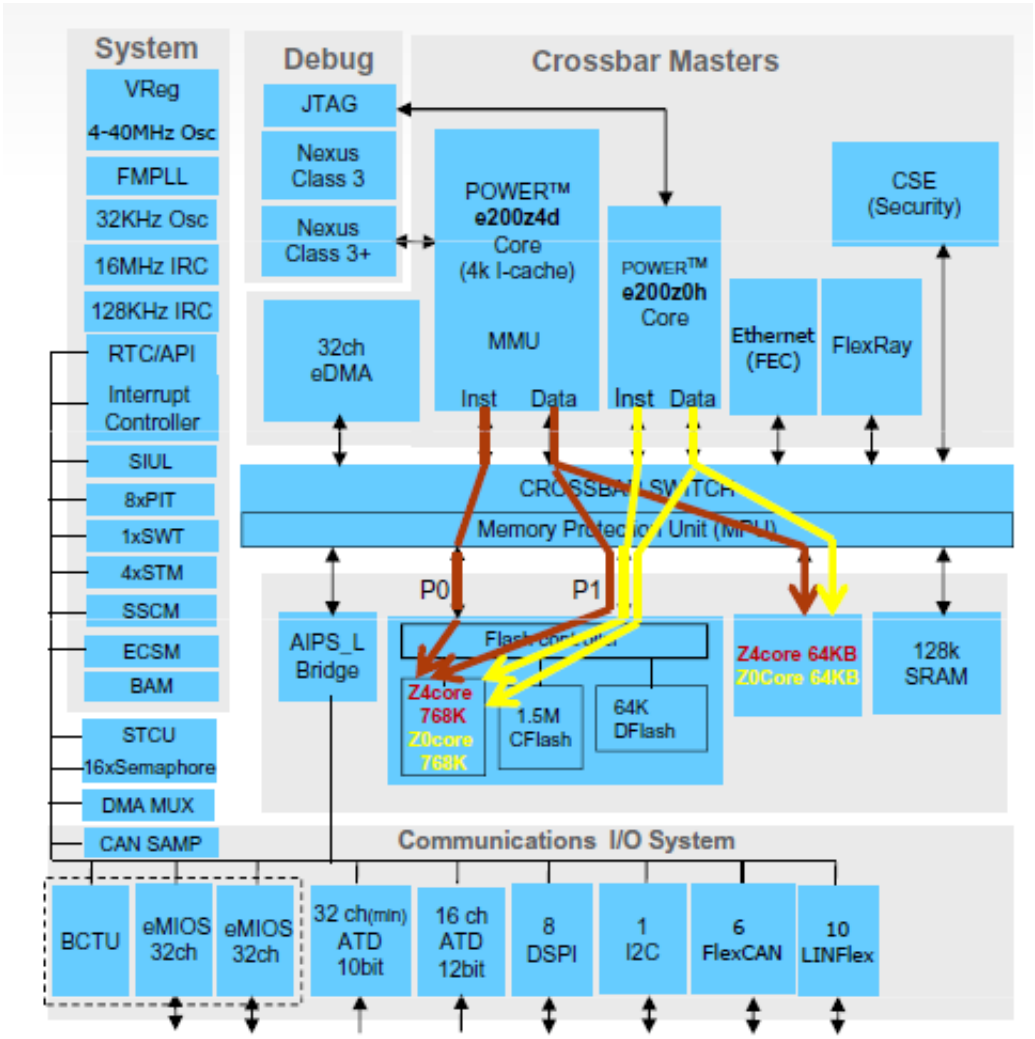

Figure 6. Instruction and Data Paths for configuration 3

Dhrystone Relative Performance Calculations Example: Per Configuration 3 Dhrystone Results table first XBAR Flash Port 1 Configuration, the “z4 Relative Performance” at 120 MHz is:

292.80 (120 Mhz z4 DMIPS dual core configuration per Table 23) /
291.60 (120 Mhz z4 DMIPS single core configuration per Table 7)
= 100.41 %.

Using the same table and XBAR Flash Port 1 Configuration, the “Dual Core Relative Performance” at 120 and 60 MHz is:

352.20 DMIPS (120/60 MHz dual core per Table 23: sum of 292.80 z4 DMIPS + 59.40 z0 DMIPS) /
370.20 DMIPS (120/60 MHz single core per Tables 7, 8: sum of 291.60 z4 DMIPS + 78.60 z0 DMIPS)
= 95.14%.

Table 23. Configuration 3 Dhrystone Results

CPU freq. z4/ z0 (MHz)	XBAR RAM 0 Port Configuration: z4 Data Bus has Higher Priority XBAR Flash Port 0 Port Configuration: z4 Instruction Bus has Higher Priority						XBAR RAM 0 Port Configuration: z0 Data Bus has Higher Priority XBAR Flash Port 0 Port Configuration: z0 Instruction Bus has Higher Priority					
	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)	z4 DMIPS (dual core)	z4 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z0 DMIPS (dual core)	z0 Rel. Perf. (z4 DMIPS dual core / z4 DMIPS single core)	z4 + z0 DMIPS (dual core)	Dual Core Rel. Perf. (Total DMIPS single cores / Total DMIPS dual core)
120/60	292.80	100.41 %	59.40	75.57%	352.20	95.14%	282.00	96.71%	72.60	92.37%	352.20	95.79%
100/50	250.00	100.40 %	50.00	70.92%	300.00	93.90%	244.00	97.99%	63.50	90.07%	286.50	96.24%
80/80	199.20	97.27%	74.40	69.40%	273.60	87.69%	101.60	49.61%	102.40	95.52%	264.80	65.38%
64/64	186.88	96.69%	80.00	74.40%	266.88	88.72%	85.12	44.04%	100.48	93.45%	208.00	61.70%
40/40	119.60	95.22%	60.80	79.17%	180.40	89.13%	65.60	52.23%	70.80	92.19%	144.40	67.39%
20/20	62.20	97.80%	31.00	70.78%	93.20	86.78%	41.80	65.72%	42.00	95.89%	79.20	78.03%

NOTE

See Tables 7, 8 for single core configuration DMIPS numbers.

Run Time Relative Performance Calculations Example: Per Configuration 3 Benchmark A Results table, first XBAR Flash Port 1 Configuration, the “z4 Relative Performance” at 120 MHz is:

49.63 (120 Mhz z4 single core configuration per Table 7) /
59.95 (120 Mhz z4 dual core configuration per Table 24)= 82.79 %.

Using the same table and XBAR Flash Port 1 Configuration, the “Dual Core Relative Performance” at 120 and 60 MHz is:

167.68 (120/60 MHz single core per Tables 7, 8: sum of 49.63 z4 +118.05 z0 run times) /
269.10 (120/60 MHz dual core per Table 24: sum of z4 + z0 run times)
= 62.31 %.

Table 24. Configuration 3 Benchmark A Results

CPU freq. z4/ z0 (MHz)	Highest XBAR Priority and Parking for RAM0: z4 data Highest XBAR Priority and Parking for Flash Bank 0: z4 instr.	Highest XBAR Priority and Parking for RAM0: z0 data Highest XBAR Priority and Parking for Flash Bank 0: z0 instr

Table continues on the next page...

Table 24. Configuration 3 Benchmark A Results (continued)

	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)
120/60	59.95	82.79%	209.15	56.44%	269.10	62.31%	67.28	73.77%	147.77	79.89%	215.05	77.97%
100/50	64.13	87.03%	267.62	51.57%	331.75	58.43%	73.41	76.03%	172.06	80.22%	245.47	78.96%
80/80	78.81	84.44%	206.80	42.52%	285.61	54.09%	124.68	53.38%	109.78	80.10%	234.45	65.89%
64/64	70.17	85.30%	164.44	48.35%	234.61	59.40%	102.41	58.45%	96.91	82.04%	199.31	69.92%
40/40	100.23	91.57%	270.35	44.29%	370.58	57.08%	160.28	57.26%	140.10	85.47%	300.38	70.42%
20/20	194.05	92.89%	536.80	42.64%	730.85	55.98%	325.05	55.45%	264.80	86.44%	589.85	69.37%

NOTE

See Tables 7, 8 for Single Core configuration run time numbers

Table 25. Configuration 3 Benchmark C Result

CPU freq. z4/ z0 (MHz)	Highest XBAR Priority & Parking for RAM0: z4 data Highest XBAR Priority & Parking for Flash Bank 0: z4 instr.						Highest XBAR Priority & Parking for RAM0: z0 data Highest XBAR Priority & Parking for Flash Bank 0: z0 instr					
	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)	z4 run time (dual core)	z4 Rel. Perf. (z4 run time single core / z4 run time dual core)	z0 run time (dual core)	z0 Rel. Perf (z0 run time single core / z0 run time dual core)	z4+z0 run time (dual core)	z4+z0 Rel. Perf. (sum of z4+z0 run times single core / sum of z4+z0 run times dual core)
120/60	215.75	99.71%	1467.05	73.30%	1682.80	76.68%	220.60	97.52%	1225.97	87.71%	1446.57	89.21%
100/50	258.87	99.61%	1769.18	74.47%	2028.05	77.68%	264.95	97.32%	1325.88	99.36%	1590.83	99.02%
80/80	323.83	99.51%	968.95	80.36%	1292.78	85.15%	339.10	95.02%	884.30	88.05%	1223.40	89.98%
64/64	369.22	99.25%	1034.22	88.90%	1403.44	91.63%	381.17	96.13%	952.97	96.48%	1334.14	96.38%

Table continues on the next page...

Table 25. Configuration 3 Benchmark C Result (continued)

40/40	590.50	99.48%	1535.45	91.29%	2125.95	93.57%	608.55	96.53%	1420.05	98.71%	2028.60	98.06%
20/20	1181.10	99.47%	2805.50	91.79%	3986.60	94.07%	1233.05	95.28%	2597.20	99.15%	3830.25	97.91%

NOTE

See Tables 7, 8 for Single Core configuration run time numbers

Comments and Recommendations:

- Configuration 3 delivers lower performance comparing to Configurations 1 and 2:
  - Configuration 1 performance is higher due to separate flash bank for code/constants and separate RAM arrays for each core
  - Configuration 2 performance is higher due to separate RAM array for each core
- Again, overall dual core performance is highly dependent on code being executed by each core as well as XBAR RAM array priority and parking settings.
  - Benchmarks with more RAM variable usage (Benchmarks A and C) had higher dual core performance when the e200z0 data bus had higher crossbar priority on the shared RAM module.
  - Use different XBAR settings to balance run time between e200z4 and e200z0.
  - Benchmark A shows a worst case scenario where both cores are executing an algorithm with high degree of RAM read/write operations (~40% of all instructions) combined with non-optimal XBAR configuration.
  - The dual core performance between configurations 2 and 3 is not very big because of independent prefetch buffers in the flash module configured for each core. This minimizes effect of accessing the same flash block.
- To estimate overall performance take into account access to common peripheral bridge (mainly for the core with lower priority on XBAR PBRIDGE slave port) and accesses from other masters (DMA, FlexRay)
  - please note that it may be needed to grant FlexRay the highest priority on the RAM block containing massage buffers to avoid FlexRay timing violations
- Please note that access to peripherals running on divided clock (output of peripheral clock divider) will slow down overall execution time comparing to a configuration with non-divided peripheral clock.

10 Summary

A high level summary from the benchmark testing is shown below. Be sure to see the respective sections for more details.

Remember -- “mileage will vary”! In other words, it is best to try varying parameters with your application. These benchmark results are a starting point.

Table 26. Summary of Parameter Comments and Recommendations

Parameter	Key Comments on Performance Test Results
Wait States	Generally performance increased with frequency, but it is a nonlinear relationship. At the higher frequency (> 64 MHz for this chip), lower performance can occur when an additional RAM wait state is required. Since maximum e200z0 frequency is 80 MHz, e200z0 performance will decrease when e200z4 frequency is above 80 MHz due to the integer divider.
Flash Line Buffer Configuration	Generally, line buffer prefetching improves performance for sequential accesses, such as instructions.

Table continues on the next page...

Table 26. Summary of Parameter Comments and Recommendations (continued)

Parameter	Key Comments on Performance Test Results
Crossbar Configuration	Master priority and parking configurations have significant impact on performance. Optimize configuration is to place parking and priority with masters accessing the slave most frequently. However users must verify with their application if any additional masters (such as DMA) are not starved when changing master priorities on slaves.
Branch Target Buffer Enabling	Enabling BTB improved performance. The improvement was non-trivial in one of the two benchmarks tested.
SDA Enabling	SDA should be used. Testing indicated performance increase generally of 1% to 5%.
Dual Core: 2 Flash Banks, 2 SRAM Modules	Configuration offered best overall performance. Crossbar settings for Flash Port 1 should be changed from the default values (see recommendations in section 8.1).
Dual Core: 2 Flash Banks, 1 SRAM Module	Middle performance, due to arbitration delays from concurrent attempts to access the single SRAM module. Crossbar settings for Flash Port 1 should be changed from the default values (see recommendations in section 8.2).
Dual Core: 1 Flash Bank, 1 SRAM Module	Lowest performance due to arbitration delays from concurent attempts to access the single SRAM module and single flash port. Crossbar settings for Flash Port 1 should be changed from the default values (see recommendations in section 8.3).

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number AN4666
Revision 0, 07/2013