High Speed Memory Technology for Cache Applications

Introduction

Many processors today use a high speed cache to accelerate memory access. A level 2 or level 3 cache connected on a backside bus can take advantage of high SRAM bandwidth in providing low latency data access. Today's performance-oriented applications require this type of processor performance boost.

There are several SRAM options available when implementing cache memory solutions. Notably, the Synchronous Burst, Late Write, and Double Data Rate (DDR) SRAMs. Each protocol has interface peculiarities and advantages. The purpose of this document is to compare and contrast these different SRAM types for cache implementations.

Cache Basics

The basic principal behind cache memory is that performance can be improved if a subset of repeatedly accessed main memory is kept in a fast access memory store. A basic cache configuration is shown in **Figure 1**.

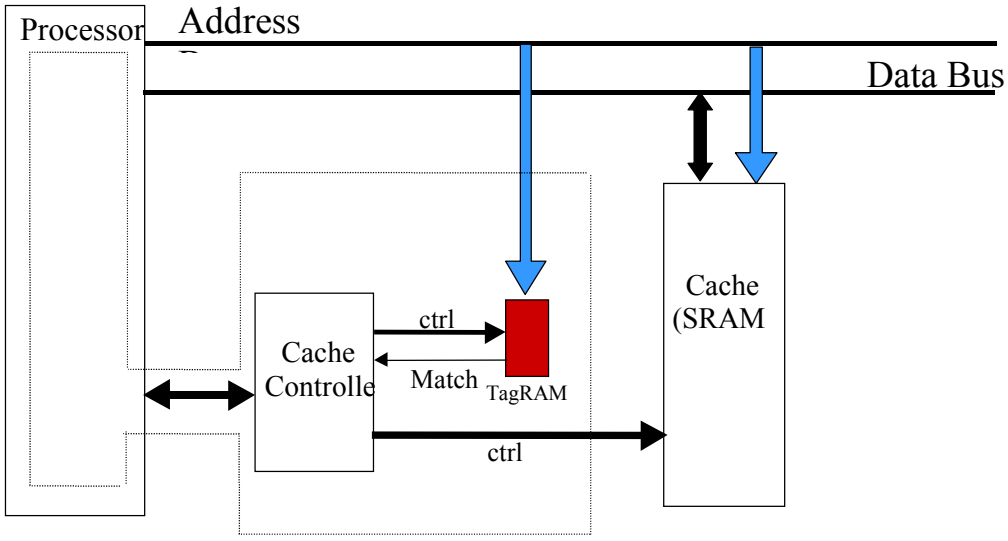


Figure 1: Basic cache configuration

The processor initiates a memory request that is sensed by the cache controller. The requested address is used to access a Tag RAM to determine if the data is located in the cache. If the data is located, it is called a cache hit; otherwise it is called a cache miss. On a cache hit the data is accessed, placed on the bus, and sent to the processor. The initial

main memory request is then cancelled. On a cache miss the main memory request is allowed to complete. The twin principals of spatial and temporal locality tell us that data requested now will be needed again soon, as will data near the target location. As a result, on a cache miss, the cache is updated with a new block of data, including the target location and nearby locations.

Early cache designs used discrete components for the controller, Tag RAM, and data cache. While this gave the designer flexibility, the cost of these specialty components became prohibitive. The market response was that some manufacturers like Intel and AMD eliminated external cache altogether. Their cache is either co-located on the processor chip or closely coupled in an MCM package. The remaining processor manufacturers integrated the controller and Tag RAM, but allow for external SRAM to be connected for data cache. **Table A** shows the list of processors still using external SRAM.

Processor	Bus Width	Max External Cache	Tag	Organization	Line Size		Protocol Support	Interface Voltage
Motorola								
MPC7450	64-bit	2MB	Internal	8-Way		PL	SBurst/LW/DDR	1.5/1.8/2.5
MPC7410	64-bit	2MB	Internal	2-Way	32B, 64B, 128B	PL	SBurst/LW	1.8/2.5
MPC750	64-bit	1MB	Internal	2-Way	64B, 128B	PL/ FT	SBurst/LW	
MPC755	64-bit	1MB	Internal	2-Way		FT	SBurst/LW	2.5/3.3
QED/PMC-Sierra								
RM5270	64-bit	2MB	Internal	Direct	32B	PL	SBurst	
RM5271	64-bit	2MB	Internal	Direct	32B	PL	SBurst	
RM7065	64-bit	8MB	External	Direct	32B	PL	SBurst	

Note: PL = Pipeline mode, FT = Flow Through mode

Table A : External SRAM Cache

There are many technical variations in the line size, the size and number of memory regions cached, the conditions upon which a replacement is made, and the decision on which cached region will be replaced. These factors, though important in cache design, are not the focus of this paper and will not be discussed further here.

SRAM Options

Early cache systems were implemented using asynchronous memory. Increasing clock rates and the need for a self-timed memory caused the displacement of asynchronous memory by synchronous SRAMs. These synchronous SRAMs included a burst capability that was specifically designed for fast cache. Augmented with new features, the core burst architecture is very much alive today. Burst RAMs enhance throughput by eliminating the requirement that a new address be asserted for each word of memory accessed. Burst RAMs internally increment a beginning address, and by doing so drastically reduce the overall time to access multiple words of memory. Burst RAM timing is shown in **Figure 2**.

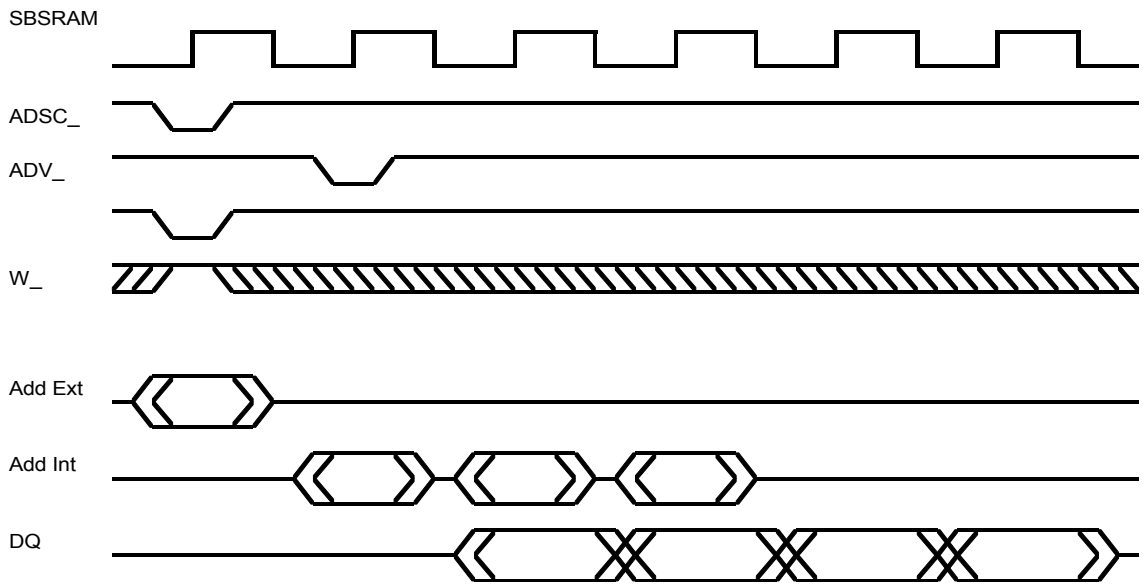


Figure 2: Burst RAM Timing

A cache is designed to accept memory requests from either the processor or the cache controller. The controller's role is to recognize a processor request and exercise the appropriate control signals. A controller may initiate write cycles to the cache when updating after a read miss. Because either processor or controller may make a request from cache, analogous control signals were created, ADSP (processor), and ADSC (controller).

Synchronous Burst SRAMs (SBSRAMs) use ADSP_ or ADSC_ signals to load a new address, and ADV to continue the burst. The processor control signal (ADSP_) is timed so as to allow a complete clock period prior to initiating the cache memory access. This affords time for the Tag RAM to determine if the requested memory location is contained within the data cache. A cycle initiated by the controller (ADSC_) begins the cycle immediately. To prevent simultaneous signals one chip enable (E1) gates ADSP_.

All factors considered, an 8MB cache is considered large. Fortunately, the correct SRAM densities are available and commonplace.

Memory size	GSI part #	Organization	Cache size						
			128KB	256KB	512KB	1MB	2MB	4MB	8MB
256Kb		32K x 8		8-32K x 8					
512Kb		64K x 8					8-256K x 8		
1Mb		128K x 8				8-128K x 8			
		64K x 18			4-64K x 18				
	GS81032A	**32K x 32**		**2-32K x 32**					
2Mb		256K x 8					8-256K x 8		
		128K x 18				4-128K x 18			
	GS82032A	**64K x 32**			**2-64K x 32**				
4Mb		512K x 8						8-512K x 8	
	GS84018A	**256K x 18**					**4-256K x 18**		
	GS84032A	**128K x 32**				**2-128K x 32**			
		64K x 64			1-64K x 64				
8Mb		1Mb x 8							8-1Mb x 8
	GS88018A	**512K x 18**						**4-512K x 18**	
	GS88032A	**256K x 32**					**2-256K x 32**		
		512K x 64						1-512K x 64	
16Mb		2Mb x 8							
	GS816018	**1Mb x 18**							**4-1Mb x 18**
	GS816032	**512K x 32**						**2-512K x 32**	
	GS816272	**256K x 72**							**1-256K x 72**

Table B: SBRAM Data Cache Options Assuming a 64-bit Bus

While the SBSRAM interface is easy to connect, detailed timing can sometimes be tricky. Typically, the SRAM write cycle is usually the most contentious issue due to the constraints imposed by the processor output timing. As an example, we will look at a Motorola 7410 processor using SBSRAM cache.

The 7410 has programmable options for several types of SRAM. Based on using SBSRAM, the applicable timing parameters are shown in **Table C**.

From a timing perspective, the main issue is making sure there is enough time left over after the output delay of the processor to allow for SRAM setup and hold margin. The relationship between setup margin and propagation delay parameters is given by: t_{setup} margin = $t_{clock} - t_{jitter} - t_{skew} - t_{SRsetup}$. The 7410 can operate at a variety of clock rates with analogous options for the cache clock. Assuming a 450 MHz system clock and a 2:1 ratio for the cache clock, one period of the cache is 4.44 ns. Subtracting the output delay of the processor and the timing uncertainty due to clock jitter leaves 1.79 ns for SRAM setup. While this figure does not account for clock skew, in practice it would be subtracted from 1.79 ns. GSI SRAMs with minimum clock rates of 225 MHz require less than 1.79 ns of setup time and would be acceptable candidates.

The amount of hold time available based on propagation delay parameters is given by: t_{hold} margin = $t_{hold\text{-}source} - t_{hold\text{-}sink} - t_{jitter} - t_{skew}$. Given the example above, the 7410 hold requirement is 0.4 ns. Subtracting 0.15 ns for jitter leaves 0.25 ns. While this figure does not account for clock skew, in practice it would be subtracted from 0.25 ns. GSI SRAMs with minimum clock rates of 250 MHz require less than 0.25 ns of setup time and would be acceptable candidates. Adjustments in setup verses hold time can be made by varying the skew (propagation delay) of either the SRAM clock or its output (echo clock). These timing adjustments can translate to physical trace lengths. For more information on PCB tradeoffs see Motorola App Note 1794D L2 BackSide Timing Analysis.

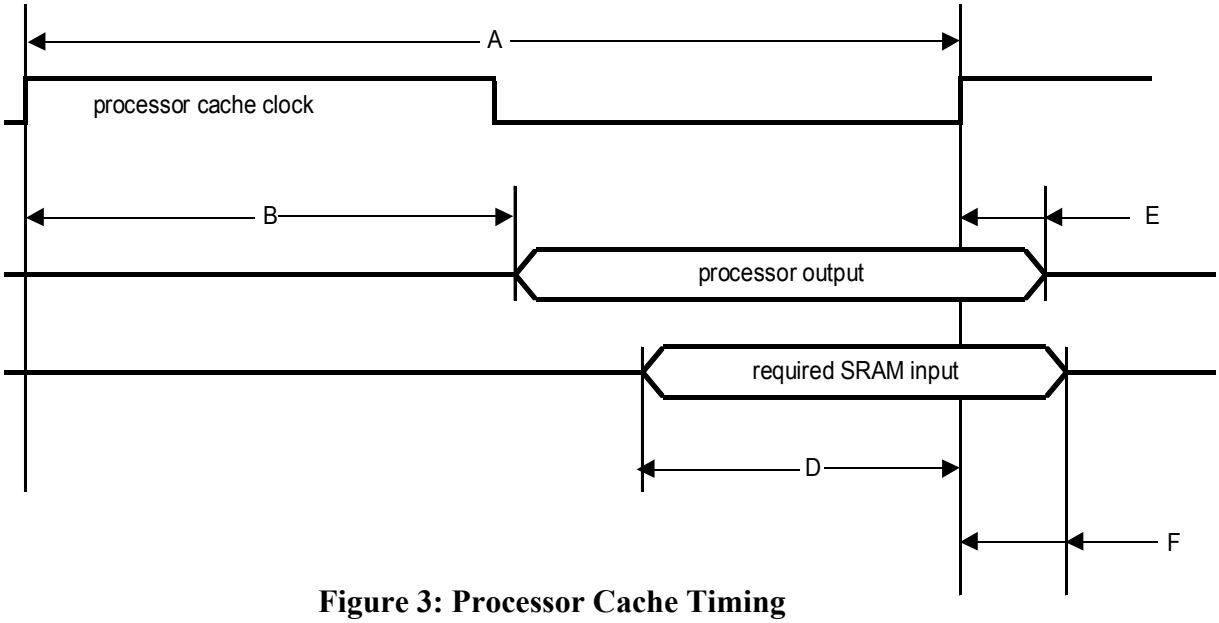


Figure 3: Processor Cache Timing

	A	B	C	A-B-C	D						E	F					
cache clk frequency MHz	cache clk period ns	7410 output delay (sync burst) ns	clock jitter ns	Available time for SRAM setup	SRAM setup ns GS816XXXA or GS880XXA						7410 hold time ns	SRAM hold ns GS816XXA or GS880XXA					
					333 MHz	300 MHz	250 MHz	225 MHz	200 MHz	166 MHz	0.4	333 MHz	300 MHz	250 MHz	225 MHz	200 MHz	166 MHz
300	3.33	2.5	0.15	0.68	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5
225	4.44	2.5	0.15	1.79	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5
180	5.55	2.5	0.15	2.9	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5
150	6.66	2.5	0.15	4.01	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5
128	7.81	2.5	0.15	5.16	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5
112	8.92	2.5	0.15	6.27	1	1.1	1.2	1.3	1.4	1.5	0.4	0	0.1	0.2	0.3	0.4	0.5

Table C: SBSRAM Timing Parameters

Another important factor in G4 timing analysis is the output drive strength. Differences in drive strength capability can significantly increase flight times between source and destination, further complicating the timing analysis. Many SRAMs have programmable drive strength options. Simulations show that the processor-to-SRAM delay could be up to 20 times greater than the SRAM to processor delay. These differences should be identified and translated to maximum PCB length restrictions.

GSI SRAMs offer a robust range of densities, speeds and interface timing. A GSI rev A 225 MHz SBSRAM will have a required setup time of just 1.3 ns, well within the allotted time of 1.79 ns, which affords plenty of extra margin for the layout. This timing is available in both 8Mb/9Mb and 16Mb/18Mb components.

A variation on the SBSRAM is the Late Write (LW) SRAM. LW refers to the placement of data during a write to the SRAM. In standard (early write) SRAMs, data is presented simultaneously with the address. In LW SRAMs, on write cycles only, data is presented one clock cycle later than the address. This timing is shown in **Figure 4**. A timing analysis for LW would be exactly the same as for SBSRAM, except for the placement of the data relative to the address.

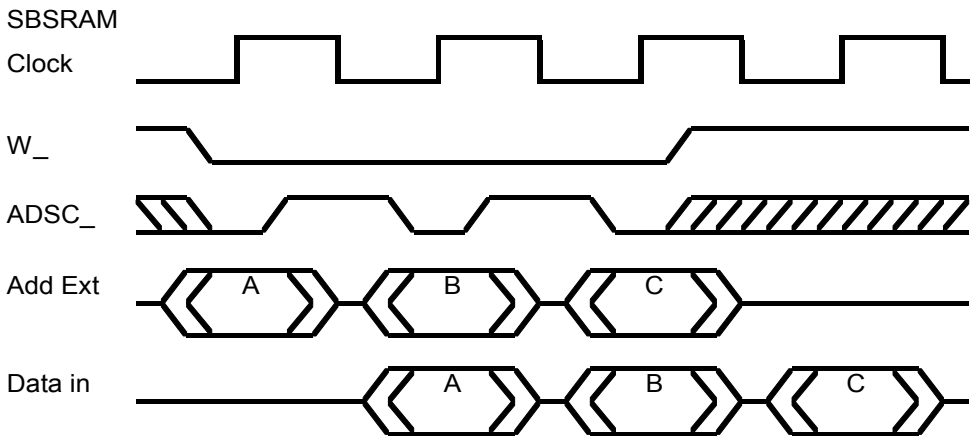


Figure 4: Late-Write Timing

A Double Data Rate (DDR) SRAM can also be used for cache. The attractive attribute for using DDR is that for a given throughput, DDR requires a lower clock rate and fewer I/O pins. To date, few processors have taken advantage of this option for cache, but the Motorola 7450 is one example.

Unlike SBRAMs, which were designed specifically to execute cache applications, the DDR SSRAM has a much broader application in mainstream communications designs. DDR SSRAMs are part of the SigmaRAM™ family, which begin at 18Mb density. As the name implies, a DDR SRAM uses both rising and falling clock edges to register data.

By utilizing both edges, the effective data rate is twice that of a single data rate part, at the same clock frequency. In other words, DDR SRAM affords the same throughput but at half the clock rate. **Figure 5** shows an example of DDR timing.

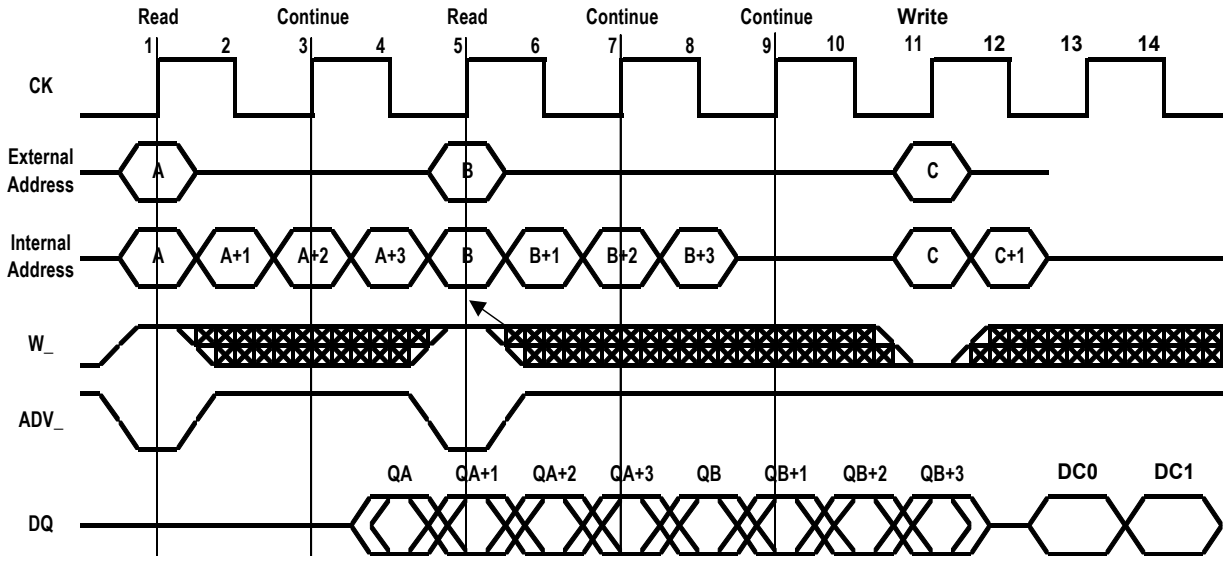
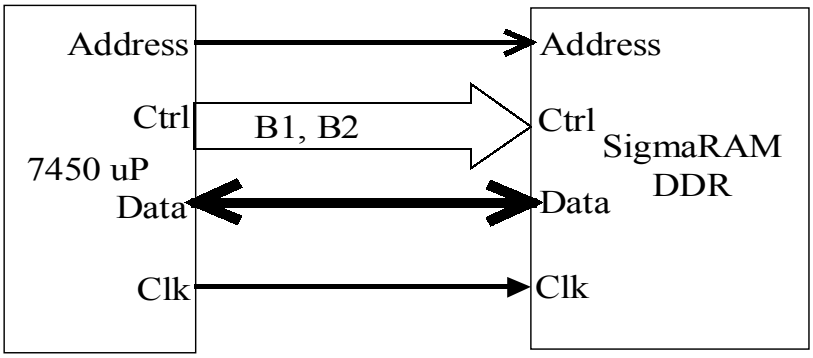


Figure 5: DDR Read and Write Timing

In this simplified drawing a read cycle is initiated with ADV_ at time 1. The external address A is used to access memory location A and A+1. By deasserting ADV_ high, the internal burst increases the access to A+2 and A+3. Had ADV_ remained low after time 1, the SRAM would, at times 4 and 5, output two consecutive words corresponding to A and A+1 only. A normal "non-burst" DDR read cycle takes one starting address and outputs two consecutive words of memory. Because data is output on alternate edges of the clock, the next read cycle could begin as early as the next full clock period or time 3 in the diagram above. A "burst" DDR cycle as depicted in the diagram requires one deselect between reads because the output is four consecutive words. Writing to the DDR RAM can be done with one cycle of deselect when operating in "non-burst" mode. Operating in Burst mode requires two cycles of deselect between read and write.

The 7450 signaling is familiar with the exception of B1 and B2. To illustrate the operation of these signals, **Figure 6** shows the correlation to SigmaRAM DDR signals. One of the differences between SBSRAMs and DDR is the control for cycle initiation. While SBSRAMs have ADSP_ and ADSC_ signals to load an address and ADV_ to continue the burst, DDR SRAMs have only one signal, ADV_. The ADV_ control combines the equivalent function of ADSC_ and ADV_ in a SBSRAM.



7450			SigmaRAM DDR		
B1	B2		E_	W_	ADV
0	0	Load, Write	0	0	0
0	1	Load, Read	0	1	0
1	0	Deselect	1	X	0
1	1	Continue	X	X	1

Figure 6: DDR and 7450 control signaling

Because the 7450 is designed for DDR, the interface in unremarkable. The timing analysis, however, presents some unique challenges. In DDR mode, the 7450 outputs one data word every half-clock period. To meet SRAM setup expectations data is actually launched using an internal 90 degree phase shifted clock. As a result, data is available ¼ clock period before the sample clock edge[1]. As an example, we assume a 550 MHz core clock with a 3:1 cache clock ratio. The cache clock period is 3.64, ns which results in setup time of 0.56 ns. Hold time is similarly stated at ¼ clock period[2].

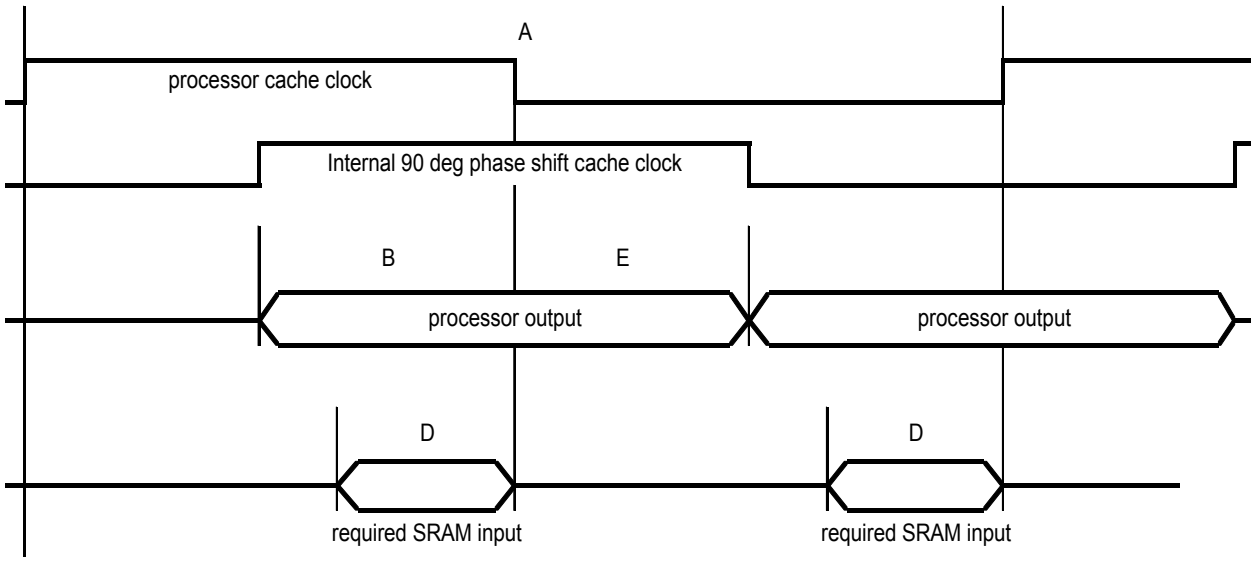


Figure7: Processor-cache (DDR) Timing

[1] Subtract 0.35 ns for data
[2] Add 0.35 ns for data

	A	B	C	(A/4)-B-C	D			E	F		
cache clk frequency MHz	cache clk period ns	7450 setup time (DDR) ns	clock jitter ns	Available time for SRAM setup ns	SRAM setup ns GS8170DXX			7410 hold time (DDR) ns	SRAM hold ns GS8170DXX		
					333 MHz	300 MHz	250 MHz		333 MHz	300 MHz	250 MHz
250	4.00	1.00	0.35	0.65	0.32	0.35	0.4	1.35	0.27	0.3	0.35
266	3.76	0.94	0.35	0.59	0.32	0.35	0.4	1.29	0.27	0.3	0.35
275	3.64	0.91	0.35	0.56	0.32	0.35	0.4	1.26	0.27	0.3	0.35

Table D: DDR Timing Parameters

In this case, a 250 MHz SRAM was selected in order to meet the setup requirements. At present, the GSI DDR SRAM meeting these requirements is part number GS8170DDxx. GSI supports custom cache designs by offering a range of SBSRAMs from 1Mb to18Mb, including DDR. All of these parts feature standard packaging with voltage ranges from 1.8 to 3.3 volts. For more information, access www.gsitechnology.com or contact nharris@gsitechnology.com.