Memory Interface on TMS320C8x

Literature Number: BPRA069
Texas Instruments Europe
December 1997

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Contents

List of Figures

List of Tables

Memory Interface on TMS320C8x

ABSTRACT

The reason for developing the TMS320C8x - PC - Testboard was to gain experience with the hardware interface of the ’C8x and may be used as a reference design.

Therefore, we have implemented different memory devices on the board. This includes SRAM, SDRAM, VRAM and EPROM. The board has an interface for audio, video - and JTAG - signals so that it can be easily used for demonstration purposes.

A bi-directional FIFO builds the interface to the ISA-Bus and allows one to install the board in a PC and communicate with the ’C8x. A connector with almost all signals of the ’C8x gives the possibility of expanding the board with additional peripherals.

Currently, only the VRAM, the SRAM, the SDRAM and the EPROM are mounted on the board. The function of these memories has been tested. The test was based on a S/W test using the debugger via the JTAG connector.

The connections of Colour Palette, FIFO and TLC320AD65 to the ’C8x are available in the schematics but they have not been implemented or tested with hardware. In the following application report only the connection of the four memory types will be described in detail.

The reason for the implementation of CBTs in the data bus is to show how 5V-TTL-devices could easily be connected to Low Voltage components. Normally the 5V/3.3V conversion would be done by the ‘LVT16245 bus driver, which is also used on the board.

The TIBPAL20L8-5 devices, creating the Memory Identification Signals, should be replaced with 3.3V-PALs. In this case the CBTs in these signal lines are not necessary.

1. Configuration

1.1 System Overview

The block diagram (Figure 1) shows all memory and interface sections implemented in the TMS320C8x - PC - Testboard. The arrows give the realized data paths. Control logic is not presented in this figure.

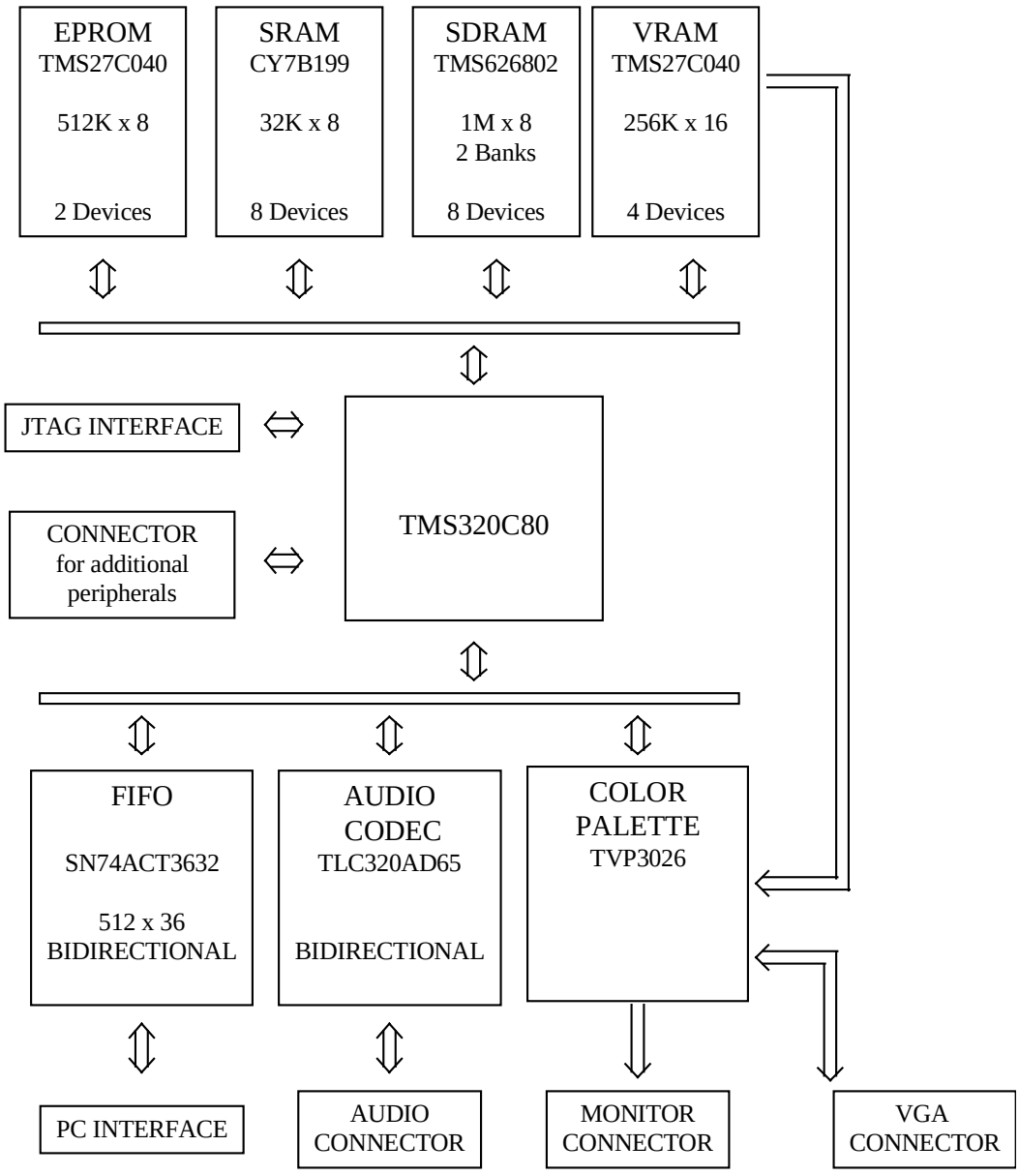

Figure 1: Functional block diagram

1.2 Memory Map

Except for EPROM, all memory arrays and interfaces can be placed in the memory map in any order. After hardware reset TC will service the MP instruction cache fill to fetch the cache block beginning at 0xFFFF FFC0. This block contains the address 0xFFFF FFF8 which is where the MP begins instruction execution. Therefore EPROM must be placed at the end of the memory map. The internal memory space is fixed, of course.

0x0000 0000 0x01FF FFFF	Internal Memory
0x0800 0000 0x081F FFFF	VRAM bank 2MByte (4 x 512K)
	free for additional VRAM banks
0x2000 0000 0x20FF FFFF	SDRAM bank 16Mbyte (8 x 2M)
	free for additional SDRAM banks
0x3800 0000 0x3803 FFFF	SRAM bank 256K (8 x 32K)
	free for additional SRAM banks
0x8000 0002 0x8000 0032	TLC320AD65 control registers
0x8800 0002	TLC320AD65 capture address
0x9800 0002	TLC320AD65 playback address
0x9800 0003 0x9800 00F3	TVP3026 control registers
0xA000 0004	FIFO AT-bus data I/O
0xA800 0006	‘LVT244 (FIFO flag register)
	free for additional EPROM banks
0xFFFF 0000 0xFFFF FFFF	EPROM bank 1MByte (2 x 512K)

Figure 2: Memory map

2. How to connect memory banks to the ’C8x

2.1 Connecting EPROM TMS27C040-10 to ’C8x

2.1.1 EPROM System Overview

In this chapter of the application report, the following system will be considered.

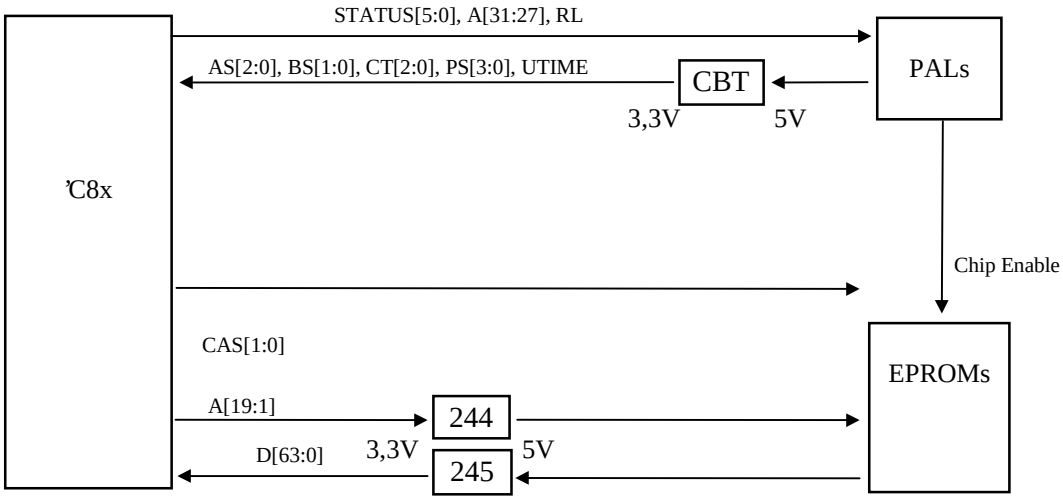

Figure 3: EPROM Interface Block Diagram

2.1.2 Affected Row Time Status Codes

(’C8x Rev. 3.0 Addendum p. 6-1)

Table 1: Status codes (row time) which affect the EPROM

STATUS[5:0]						Cycle Type	TC outputs valid Row Address on A[31:0]?	MNEMONIC
0	0	0	0	0	0	Normal Read	yes	READ
1	x	x	x	x	x	XPT-Reads	yes	READ

2.1.3 The ASI-Signal

Basically, the chip enable signals are generated by decoding the MSBs of the address bus. As the TC does not output a true address on A[31:0] for every access (e.g. Refresh Cycle), bus conflicts can occur. To avoid this, STATUS-lines must be evaluated in addition.

While the TC accesses DRAMs with multiplexed addressing, the MSBs of the address bus are invalid during Column Time. This may also cause bus conflicts. Therefore all chip enable signals must be latched with RL\.

In this application the signal AS\ (Address Shifted) takes all these problems into consideration and is used from every external logic that combines address lines. It shows if the TC outputs a valid address on A[31:0] during row- and column access time.

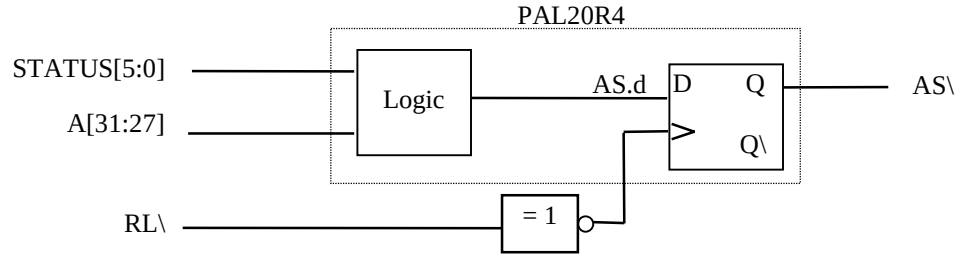

Figure 4: Logic diagram for AS

```
AS.d =!A31 & !A30 & !A29 & !A28 & A27                          /*VRAM-Address
     |!A31 & !A30 & A29 & !A28 & !A27                          /*SDRAM-Address
     |!STAT5 & !STAT4 & !STAT3 & !STAT2 &  STAT1 &  STAT0     /*SDRAM DCAB
     |!STAT5 & !STAT4 &  STAT3 &  STAT2 & !STAT1 & !STAT0     /*SDRAM MRS
     |!STAT5 & !STAT4 & !STAT3 & !STAT2 &  STAT1 & !STAT0;    /*REFRESH
```

2.1.4 Logic for E\ (Chip Enable) and G\ (Output Enable)

E\ must be set low for every access to the EPROM. As the TC outputs on a EPROM-access always the true address on A[31:0] the status lines can be left disregarded. In case of a special access cycle to another peripheral or memory device the value on the address bus could accidentally be the EPROM address. To avoid bus conflicts E\ is additionally tied up with AS\.

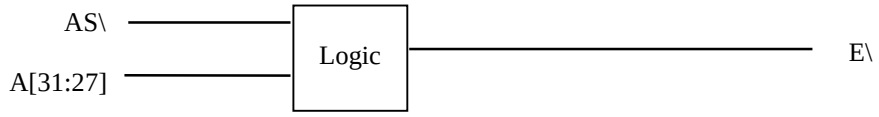

Figure 5: Logic diagram for E

```
!E = A31 & A30 & A29 & A28 & A27 & AS;                         /*EPROM-Address
```

Normally there is no need to switch the output enable signal of the EPROM. It can just be connected to Ground. The byte selection in case of a larger bus size than 8 bit is done by the TC after a READ cycle internal. In this application report the G\ pins of the EPROMs are linked to the CAS-signals. This should give an idea of how to design EEPROMs in a ’C8x based system.

2.1.5 Address bus

Dependent on bus size different address lines have to be used

Table 2: Connecting address bus

Bus Size	Address Bits connected to EPROM
8 bit	A[18:0]
16 bit	A[19:1]
32 bit	A[20:2]
64 bit	A[21:3]

2.1.6 Data and CAS signals

As the LSBs of the address bus are not connected to the EPROM, the TC creates the byte select with the CAS-signals. They are connected to the relevant output enable pin G\.
If bus size is less than 64 bits, Big- and Little Endian Mode need different signals to be connected to the memory bank.

Table 3: Connecting data bus and CAS\ signals

	Data Bus		CAS-Signals	
Bus Size	Big Endian	Little Endian	Big Endian	Little Endian
8 Bit	D[63:56]	D[7:0]	CAS7	CAS0
16 Bit	D[63:48]	D[15:0]	CAS[7:6]	CAS[1:0]
32 Bit	D[63:32]	D[31:0]	CAS[7:4]	CAS[3:0]
64 Bit	D[63:0]	D[63:0]	CAS[7:0]	CAS[7:0]

2.1.7 Timing Evaluation

E\:

$t_a(RAV\text{-}EL) = t_a(RAV\text{-}RL\_L) + t_{PLH}(inv) + t_{pd}(ff) + t_{pd}(nand);$ (1)

$t_a(RAV\text{-}EL) = t_a(RAV\text{-}RL\_L) + 5ns + 4{,}5ns + 5{,}5ns = t_a(RAV\text{-}RL\_L) + 15ns;$

G\ (G\ = CASx\):

$t_a(RAV\text{-}G\_L) = t_a(RAV\text{-}CASL) + t_{PHL}(244);$ (2)

$t_a(RAV\text{-}G\_L) = t_a(RAV\text{-}CASL) + 4{,}1ns;$

Address Bus

$t_a(RAV\text{-}AXV) = t_{PHL}(244) = 4{,}1ns;$ (3)

with the column cycle timing code the following parameters can be varied:

Table 4: Evaluating column timing code

CT2	CT1	CT0	t_a(RAV-DV)	t_a(RAV-CAS_L)	t_a(RAV-RL_L)
1	0	1	$6t_H$ - 7ns + $5t_H$ - 10ns = 120,5ns	$6t_H$ - 7 + $3t_H$ + 7 = 112,5ns	$3t_H$ - 12 + 8 = 33,5ns

1	1	0	$6t_H$ - 7ns + $8t_H$ - 10ns = 158ns	$6t_H$ - 7 + $5t_H$ + 7 = 137,5ns	$3t_H$ - 12 + 8 = 33,5ns
1	1	1	$8t_H$ - 7ns + $10t_H$ - 10ns = 208ns	8tH - 7 + $6t_H$ + 7 = 175ns	$3t_H$ - 12 + 8 = 33,5ns

'C8x-Frequency = 40MHz; t_H =12,5ns, UTIME\ = 1

$t_a(RAV\text{-}DV) = t_{su}(RAV\text{-}RASL) + t_a(RASL\text{-}DV);$ (4)

$t_a(RAV\text{-}CAS\_L) = t_{su}(RAV\text{-}RASL) + t_d(RASL\text{-}CASL);$ (5)

$t_a(RAV\text{-}RL\_L) = t_a(MIDV\text{-}CFGV) + t_{su}(CFGV\text{-}CKOH);$ (6)

TMS27C040-10:
$t_a(A)$ = 100ns;
$t_a(E)$ = 100ns;
$t_{en}(G)$ = 50ns;

$t_a(RAV\text{-}DV) - t_a(RAV\text{-}AXV) \geq t_a(A);$ (7)
CT[2:0] = 111 : 208ns - 4,1ns ≥ 100ns ✓

$t_a(RAV\text{-}DV) - t_a(RAV\text{-}E\_L) \geq t_a(E);$ (8)
CT[2:0] = 111 : 208ns - 15ns -33,5ns = 159,5ns ≥ 100ns ✓

$t_a(RAV\text{-}DV) - t_a(RAV\text{-}G\_L) \geq t_{en}(G);$ (9)
CT[2:0] = 111 : 208ns - 175ns -4,1ns = 28,9ns
CT[2:0] = 111, UTIME\ = 1: 208ns -175ns - 4,1ns + 25ns = 53,9ns ✓

2.1.8 Define names used in the evaluation

ta(RAV-EL)	maximum access time, E\ low from A (row address) valid
ta(RAV-RL_L)	maximum access time, RL_L from A (row address) valid
ta(RAV-AXV)	maximum access time, Address valid at EPROM
ta(RAV-G_L)	maximum access time, G\ low from A (row address) valid
ta(RAV-DV)	maximum access time, 'C8x expects data valid from row address valid
ta(RAV-CASL)	maximum access time, CAS\ low from A (row address) valid
ta(RAV-RL_L)	maximum access time, RL\ low from A (row address) valid
tsu(RAV-RASL)	defined in 'C8x specification “advance information”
ta(RASL-DV)	defined in 'C8x specification “advance information”
tsu(RAV-RASL)	defined in 'C8x specification “advance information”
td(RASL-CASL)	defined in 'C8x specification “advance information”
ta(MIDV-CFGV)	defined in 'C8x specification “advance information”
tsu(CFGV-CKOH)	defined in 'C8x specification “advance information”
tPLH(inv)	maximum delay time of SN74AS04, transition from low to high
tpd(ff)	maximum delay time of a PAL's Flip Flop after CLK↑
tpd(nand)	maximum delay time of a PAL's NAND
tPHL(244)	maximum delay time of SN74LVT16244, transition high to low
tH	Period of CLKIN, CLKOUT = 2tH
ta(A)	EPROM's maximum access time, Data valid from AX valid
ta(E)	EPROM's maximum access time, Data valid from E\ low

ten(G)	EPROM's maximum access time, Data valid from G\ low

2.1.9 Memory Identification Signals

In addition to providing the EPROM with various signals, external logic must also provide the 'C80 with some control signals; namely, AS[2:0], BS[1:0], PS[3:0], CT[2:0] and UTIME\ to describe the types of cycles that the 'C80 should generate. These signals are sampled at row time during the r^2 state. The logic which creates these signals must also be responsible for decoding this information for other memory types and peripherals in the system as well. [1].

For some special cycle types no true addresses occur on the address bus (e.g. DRAM Refresh Cycles). If one of those cycles is used in the system the logic must decode the status lines in addition to the MSBs of the address. For all commands affecting the EPROM the TC outputs a true row address during r^2 state (see above Table 1).

As shown in the timing evaluation the EPROMs require the 3-cycle per column access mode for proper operation.
⇒ CT2 = 1, CT1 = 1 and CT0 = 1 ([11], 'C8x Rev. 3.0 Addendum p.7-5)

Equation (9) gives the need for the user modified timing mode.
⇒ UTIME\ = 0 ([13], TC User's Guide p. 7-17)

The two EPROMs build a bus size of 16 bit.
⇒ BS1 = 0 and BS0 = 1 ([13], TC User's Guide p. 7-18)

For the EPROMs the address output must be unshifted during column time.
⇒ AS2 = 0, AS1 = 0 and AS0 = 0 ([13], TC User's Guide p. 7-12)

The very slow devices such as EPROMs need the whole row- and column time for every access. Therefore page mode must be disabled.
⇒ PS3 = 1, PS2 = 0, PS1 = 0 and PS0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.4-1)

The logic equations for the memory identification signals can be seen in the proLogic files in the appendix. The logic is realized in the parts U57, U58 (TIBPAL20L8) and U54 (TIBPAL16L8).
UTIME\ defines during a reset the endian mode for the 'C8x. Note that for this reason UTIME\ must be additional combined with the RESET\ signal.
([13], TC User's Guide p. 9-6).

2.2 Connecting SRAM CY7B199-12 to ’C8x

2.2.1 SRAM System Overview

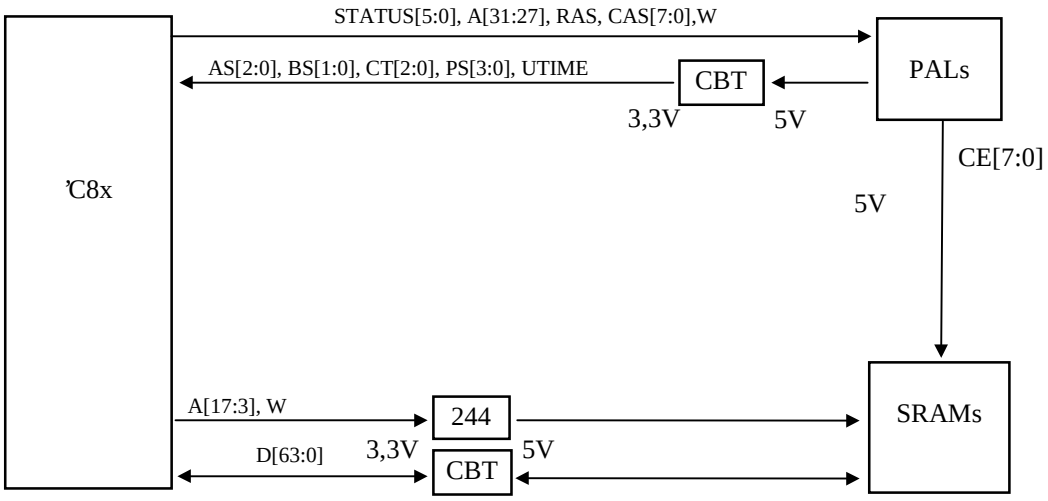

Figure 6: SRAM Interface Block Diagram

2.2.2 Affected Row Time Status Codes

Table 5: Status codes (row time) which affect the SRAM

STATUS[5:0]						Cycle Type	TC outputs valid Row Address on A[31:0]?	MNEMONIC
0	0	0	0	0	0	Normal Read	yes	READ
0	0	0	0	0	1	Normal Write	yes	WRT
1	x	x	x	x	x	XPT-Reads	yes	READ
1	x	x	x	x	x	XPT-Writes	yes	WRT

(’C8x Rev. 3.0 Addendum p. 6-1)

2.2.3 The ASl-Signal

Basically the chip enable signals are developed by decoding the MSBs of the address bus. As the TC outputs not for every access a true address on A[31:0] (e.g. Refresh Cycle) bus conflicts can occur. To avoid this, STATUS-lines must also be evaluated.

While the TC accesses DRAMs with multiplexed addressing, the MSBs of the address bus are invalid during Column Time. This may also cause bus conflicts. Therefore all chip enable signals must be latched with RL\.

In this application the signal AS\ (Address Shifted) takes all these problems into consideration and is used from every external logic element that combines address lines. It shows if the TC outputs a valid address on A[31:0] during row- and column access time.

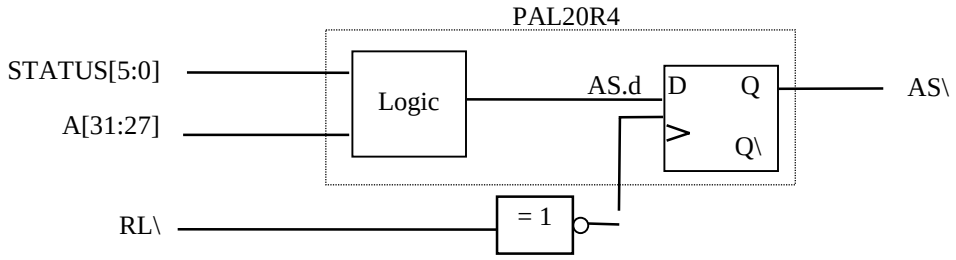

Figure 7: Logic diagram for AS

```
AS.d =!A31 & !A30 & !A29 & !A28 & A27                        /*VRAM-Address
     |!A31 & !A30 & A29 & !A28 & !A27                        /*SDRAM-Address
     |!STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & STAT0     /*SDRAM DCAB
     |!STAT5 & !STAT4 & STAT3 & STAT2 & !STAT1 & !STAT0     /*SDRAM MRS
     |!STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & !STAT0; /*REFRESH
```

2.2.4 Logic for CEx\ (Chip Enable), OE\ and W

OE\ is connected to GND. Control of the SRAM is realized only with the CE\ and WE\ pins. The W\ signal of the ’C8x can be connected to the memory device directly or via a driver.

The logic for CEx\ uses the MSBs of the address and the AS\ signal for memory identification. Using AS\ avoids all possible bus conflicts. Eight different CE[7:0]\ have to be created with CAS[7:0]. One for each device to make single Byte access possible.

To ensure that the TC outputs valid data while SRAM latches them for write-cycles RAS\ is used in addition. In the user modified timing mode (UTIME\ = 0 during r2 state) RAS\ is activated for only one column-cycle. Therefore it is used to break off CEx\ during earlier write cycles.

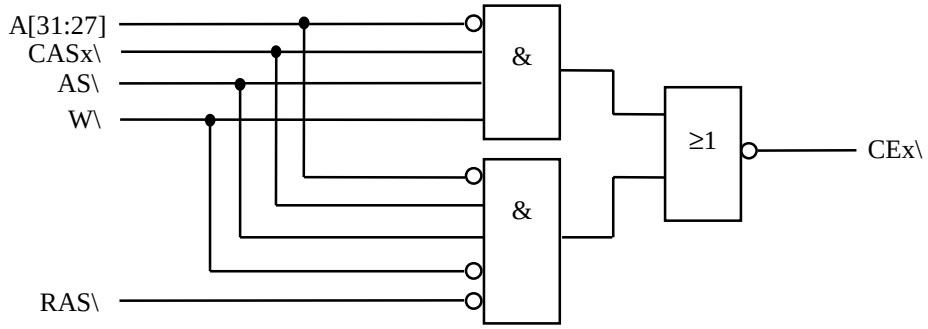

Figure 8: Logic diagram for CEx

```
!CEx = AS\ & A[31:27] & !RAS\ & !CASx\ & !W\                    (Write)
     | AS\ & A[31:27] & !CASx\ & W\                             (Read)
```

2.2.5 Address bus

Depending on bus size, different address lines have to be used:

Table 6: Connecting address bus

Bus Size	Address Bits connected to SRAM
8 bit	A[14:0]
16 bit	A[15:1]
32 bit	A[16:2]
64 bit	A[17:3]

2.2.6 Data and CAS signals

As the LSBs of the address bus are not connected to the SRAM, the TC creates the byte select with the CAS-signals or rather with the CEx\ signals. If bus size is less than 64 bits, Big- and Little Endian Mode need different signals to be connected to the memory bank or logic.

Table 7: Connecting data bus and CAS signals

	Data Bus		CAS-Signals	
Bus Size	Big Endian	Little Endian	Big Endian	Little Endian
8 Bit	D[63:56]	D[7:0]	CAS7	CAS0
16 Bit	D[63:48]	D[15:0]	CAS[7:6]	CAS[1:0]
32 Bit	D[63:32]	D[31:0]	CAS[7:4]	CAS[3:0]
64 Bit	D[63:0]	D[63:0]	CAS[7:0]	CAS[7:0]

2.2.7 Timing Evaluation

The very fast SRAMs can be accessed in page mode. The row-cycle will be generated only for the first address of an SRAM-access. Therefore the timing must be evaluated for column to column access. In the following, a clock frequency of 40MHz is given for the ’C8x.

Column Address - Data valid:

$t_a(CAV\text{-}DV) > t_{pdmin}(nand) + t_{ACE}$; (1)

$t_a(CAV\text{-}DV) > t_{PHLmax}(244) + t_{AA}$; (2)

For CY7B199-12 and Cycle Timing Code CT2 = 1, CT1 = 1 and CT0 = 0, UTIME\ = 0:

(1) 40ns > 5,5ns + 12ns; ✓

(2) 40ns > 4,1ns + 12ns; ✓

Address valid - CEx\ = low:

In the user modified mode, column address and CASx\ are simultaneously valid or active. Different signal routes are used to ensure correct timing:

$t_{SA} < t_{pdmin}(nand) + t_{PHLmin}(244) - t_{PHLmax}(244);$ (3)

For CY7B199-12 and Cycle Timing Code CT2 = 1, CT1 = 1 und CT0 = 0, UTIME\ = 0:
(3) 0ns < 1,5ns + 1 - 4,1ns;
wrong in worst case, but for $t_{PHLmax}(244) - t_{PHLmin}(244) < t_{pdmin}(nand)$ timing is correct.

Pulse duration of CEx\:
$t_w(RASL) > t_{SCE};$ (4)

CEx\, setup and hold of data valid:
$t_{SD} < t_w(RASL) + t_{pdmin}(nand);$ (5)
$t_w(DV) > t_w(RASL) + t_{pdmin}(nand);$ (6)
with $t_w(DV) = t_{su}(DV\text{-}CASL) + t_h(CASL\text{-}DV);$

For CY7B199-12 and Cycle Timing Code CT2 = 1, CT1 = 1 und CT0 = 0:
(4) 25ns > 9ns; ✓
(5) 7ns < 25ns + 1,5ns; ✓
(6) 5,5ns + 30,5ns > 25ns + 5,5ns; ✓

2.2.8 Define names used in the evaluation

$t_a(CAV\text{-}DV)$	defined in 'C8x specification “advance information”
$t_w(RASL)$	defined in 'C8x specification “advance information”
$t_w(RASL)$	defined in 'C8x specification “advance information”
$t_{su}(DV\text{-}CASL)$	defined in 'C8x specification “advance information”
$t_h(CASL\text{-}DV)$	defined in 'C8x specification “advance information”
$t_w(DV)$	minimum pulse duration, data valid
$t_{pdmin}(nand)$	minimum delay time of a PAL’s NAND
$t_{PHLmin}(244)$	minimum delay time of SN74LVT16244, transition high to low
$t_{PHLmax}(244)$	maximum delay time of SN74LVT16244, transition high to low
t_{ACE}	SRAM’s maximum access time, data valid from CE\ low
t_{AA}	SRAM’s maximum access time, data valid from address valid
t_{SA}	SRAM’s minimum address set-up time to Write Start
t_{SCE}	SRAM’s minimum pulse duration, CE\ low
t_{SD}	SRAM’s minimum set-up time, data valid to Write End

2.2.9 Memory Identification Signals

In addition to providing the EPROM with various signals, external logic must also provide the ‘C80 with some control signals; namely, AS[2:0], BS[1:0], PS[3:0], CT[2:0] and UTIME\ to describe the types of cycles that the ‘C80 should generate. These signals are sampled at row time during the r^2 state. The logic which creates these signals must also

be responsible for decoding this information for other memory types and peripherals in the system as well. [1]

For some special cycle types no true addresses occur on the address bus (e.g. DRAM Refresh Cycles). If one of those cycles is used in the system the logic must decode the status lines in addition to the MSBs of the address. For all commands affecting the SRAM the TC outputs a true row address during r^2 state (see above Table 5).

As shown in the timing evaluation the SRAMs require the 2-cycle per column access mode for proper operation.
⇒ CT2 = 1, CT1 = 1 and CT0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.7-5)

For the right timing of CEx\ user modified timing mode is necessary.
⇒ UTIME\ = 0 ([13], TC User's Guide p. 7-17)

Eight SRAMs build a bus size of 64 bit.
⇒ BS1 = 1 and BS0 = 1 ([13], TC User's Guide p. 7-18)

For the SRAMs the address output must be unshifted during column time.
⇒ AS2 = 0, AS1 = 0 and AS0 = 0 ([13], TC User's Guide p. 7-12)

The very fast devices such as SRAMs need column cycles for every access. Therefore page mode should be enabled. The whole memory space of the SRAMs is defined as one page:
⇒ PS3 = 1, PS2 = 0, PS1 = 1 and PS0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.4-1)

The logic equations for the memory identification signals can be seen in the proLogic files in the appendix. The logic is realized in the parts U57, U58 (TIBPAL20L8) and U54 (TIBPAL16L8).
UTIME\ defines during a reset the endian mode for the 'C8x. Note that for this reason UTIME\ must be additional combined with the RESET\ signal.
([13], TC User's Guide p. 9-6).

2.3 Connecting VRAM TMS55160-70 to ’C8x

2.3.1 VRAM System Overview

In this chapter of the application report, the following system will be considered

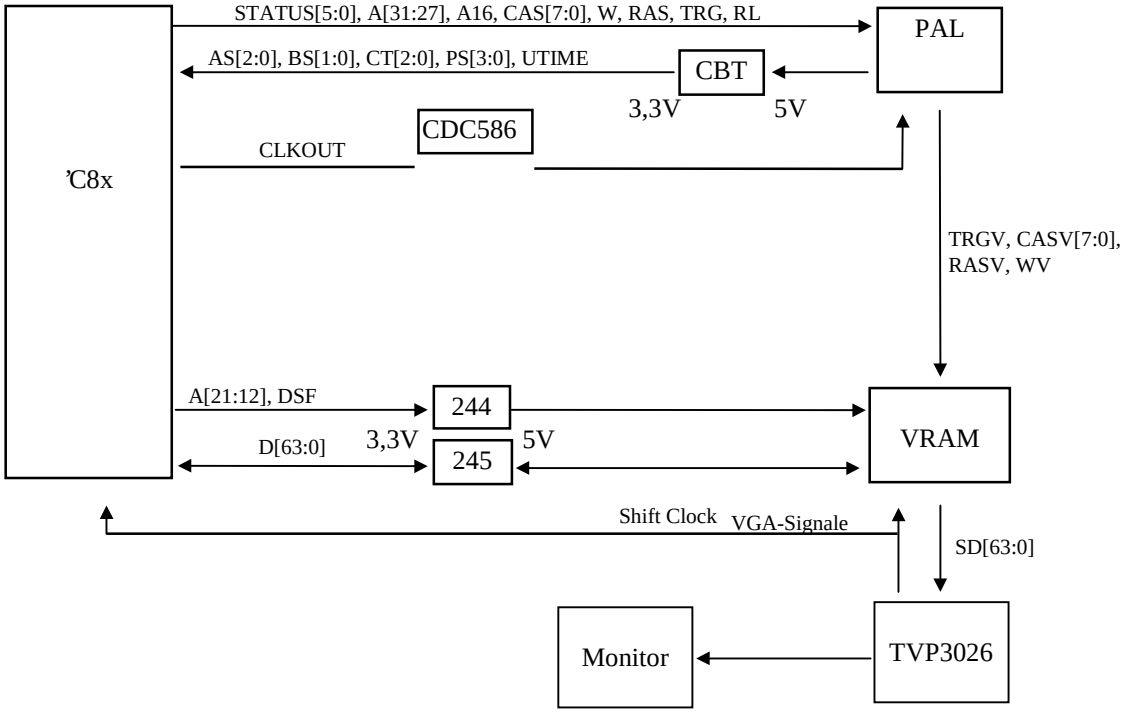

Figure 9: VRAM Interface Block Diagram

2.3.2 VRAM Control Cycles

VRAMs use various commands to control their operation and enable various features. The following cycles are implemented on ’C8x Rev. 3.0:

CBR CAS-before-RAS refresh (option reset), ends persistent write-per-bit mode and stop-point mode

RT Full-register transfer read
Full-register transfer write
(not supported by VRAM TMS55160)

SRT Split-register transfer read
Split-register transfer write
(not supported by VRAM TMS55160)

RW DRAM write (nonmasked)

BW DRAM block write (nonmasked)

LCR Load color register

VRAM cycles not supported by the TMS320C80 are:

CBRS CAS-before-RAS refresh (no reset) and stop point set

CBRN CAS-before-RAS refresh (no reset), does not end persistent write-per-bit mode and stop-point mode

RWM DRAM write in write-per-bit mode

BWM DRAM block write in write-per-bit mode

LMR Load write mask register

2.3.3 Affected Row Time Status Codes

('C8x Rev. 3.0 Addendum p. 6-1)

Table 8: Status codes (row time) which affect the VRAM

STATUS[5:0]						Cycle Type	TC outputs valid Row Address on A[31:0]?	MNEMONIC
0	0	0	0	0	0	Normal Read	yes	READ
0	0	0	0	0	1	Normal Write	yes	WRT
0	0	0	0	1	0	CBR Refresh	no	REFR
0	0	0	1	0	0	Peripheral Device PT Read	yes	RT
0	0	1	0	0	1	Block Write PT	yes	BW
0	0	1	1	0	1	Load Color Register	yes	LCR
0	1	0	0	0	0	Frame 0 Read Transfer	yes	RT
0	1	0	0	1	0	Frame 0 Split Read Transfer	yes	SRT
0	1	0	1	0	0	Frame 1 Read Transfer	yes	RT
0	1	0	1	1	0	Frame 1 Split Read Transfer	yes	SRT
0	1	1	1	0	0	PT Read Transfer	yes	RT
1	x	x	x	x	0	XPT-Reads	yes	READ
1	x	x	x	x	1	XPT-Write	yes	WRT

2.3.4 Logic for VRAM control signals

The TMS55160 has no special chip enable or output enable pin. To avoid bus conflicts the signals TRG\ and CASx\ must be combined with additional logic. A_VRAM\ gives the necessary information if the VRAM is accessed or not. To achieve this the MSBs of the address must be decoded. For refresh cycles no true row address is given from the TC. Therefore the status signals must activate A_VRAM in that case. To keep A_VRAM activated during Column time, A_VRAM must be latched with the falling edge of RL\.

In the TMS320C8x - PC - Testboard SDRAMs are implemented, too. Both need different refresh cycles. The LSB of the refresh pseudo address, A16, alternates VRAM- and SDRAM-refresh cycles.

```
!A_VRAM = (!A31 & !A30 & !A29 & !A28 & A27)
| (!STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & !STAT0 & A16);
```

DSF is connected via driver or directly to the VRAM.

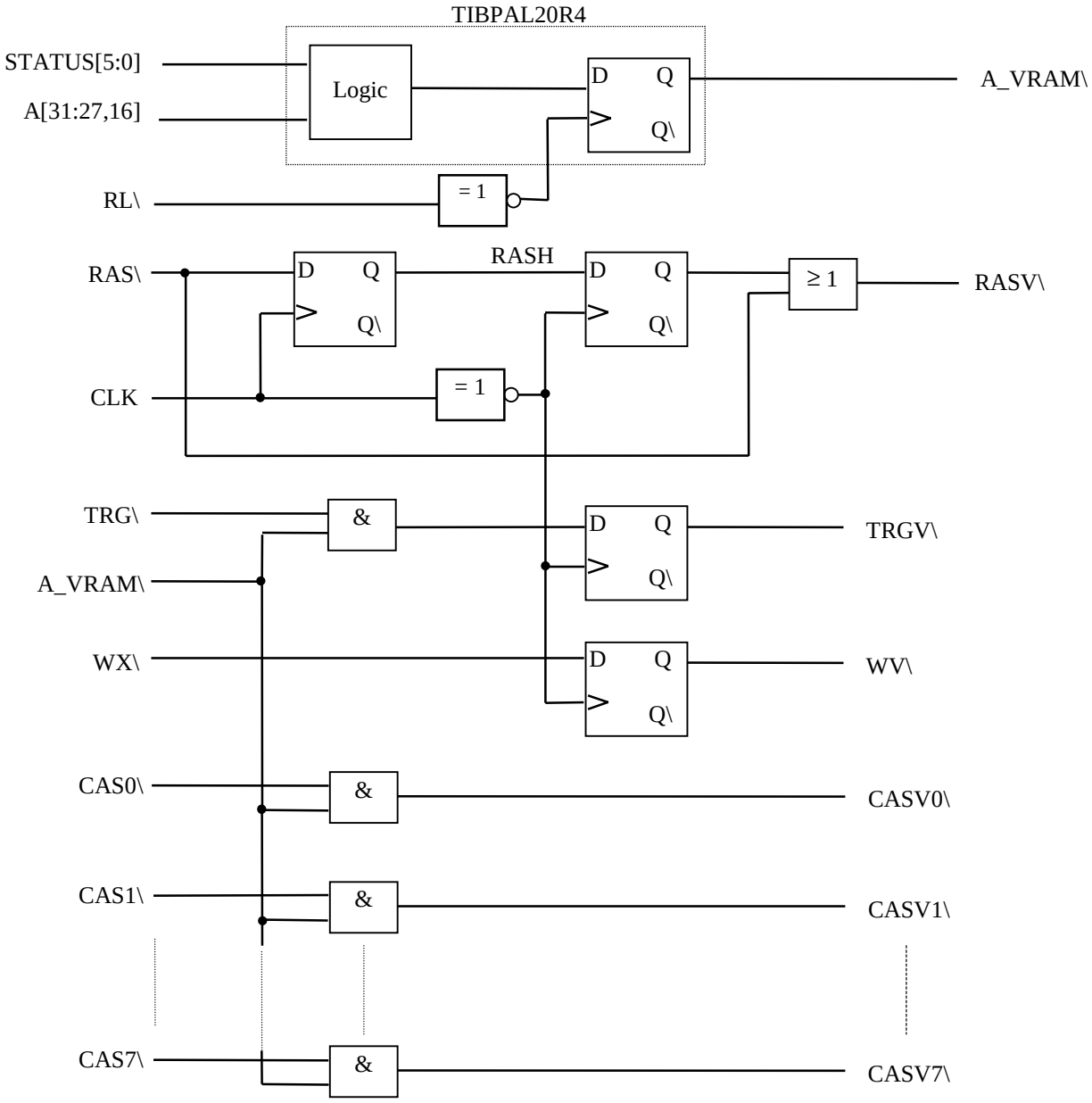

Figure 10: Logic diagram for VRAM control signals

2.3.5 Data and CAS signals

To make block write mode possible for the ‘C80 the data bus must be connected to the VRAM in a special order. The 'C8x supports block write mode only for a bus size of 64 bit.

For big and little endian mode the data bus and CAS signals must be connected in two different ways to the VRAM.

Little Endian Mode:

Table 9: Connecting data bus and CAS signals (little endian mode)

	signals 'C8x, logic outputs		to pins VRAM	
VRAM 0	CASV0\	D[7:0]	CASL\	DQ[7:0]
	CASV1\	D[15:8]	CASU\	DQ[15:8]
VRAM 1	CASV2\	D[23:16]	CASL\	DQ[7:0]
	CASV3\	D[31:24]	CASU\	DQ[15:8]
VRAM 2	CASV4\	D[39:32]	CASL\	DQ[7:0]
	CASV5\	D[47:40]	CASU\	DQ[15:8]
VRAM 3	CASV6\	D[55:48]	CASL\	DQ[7:0]
	CASV7\	D[63:56]	CASU\	DQ[15:8]

Big Endian Mode:

Table 10: Connecting data bus and CAS signals (big endian mode)

	signals 'C8x, logic outputs		to pins VRAM	
VRAM 0	CASV0\	D[7:0]	CASU\	DQ[8:15]
	CASV1\	D[15:8]	CASL\	DQ[0:7]
VRAM 1	CASV2\	D[23:16]	CASU\	DQ[8:15]
	CASV3\	D[31:24]	CASL\	DQ[0:7]
VRAM 2	CASV4\	D[39:32]	CASU\	DQ[8:15]
	CASV5\	D[47:40]	CASL\	DQ[0:7]
VRAM 3	CASV6\	D[55:48]	CASU\	DQ[8:15]
	CASV7\	D[63:56]	CASL\	DQ[0:7]

Connect SAM Outputs to Color Palette:
(Big and Little Endian Mode)

Table 11: Connect VRAM's SAM to color palette

	SAM Output VRAM	Data Input Color Palette
VRAM 0	DQ[7:0]	SD[7:0]
	DQ[15:8]	SD[15:8]
VRAM 1	DQ[7:0]	SD[23:16]
	DQ[15:8]	SD[31:24]
VRAM 2	DQ[7:0]	SD[39:32]
	DQ[15:8]	SD[47:40]
VRAM 3	DQ[7:0]	SD[55:48]
	DQ[15:8]	SD[63:56]

2.3.6 Address bus

For the bus size of 64 bit the 3 LSBs of the address lines can be left unconnected. 'C8x selects single Bytes with its CAS\-signals. The length of the column address word adds another offset of 9 bits. Therefore address bits A12-A21 of the 'C8x must be connected to VRAM's A0-A8.

2.3.7 Timing Evaluation

All timing values for $f_{'C8x}$= 40MHz, tH = 12.5ns and Column Timing Code CT[2:0] = 110 (2 cycles/column access) and VRAM type TMS55160-70.

Address Bus:

$t_h(RA) < t_h(RASL\text{-}RAV) - 2t_H - t_{PDmax}(D\text{-}FF) - t_{pdmax}(nand) + t_{PHLmin}(244);$ (1)

$t_{su}(CA) < t_{su}(CAV\text{-}CASL) + t_{pdmin}(nand) - t_{PHLmax}(244);$ (2)

$t_h(CLCA) < t_h(CASL\text{-}CAV) - t_{pdmax}(nand) + t_{PHLmin}(244);$ (3)

$t_d(RLCA) > t_d(RASL\text{-}CAV) - 2t_H - t_{PDmin}(D\text{-}FF) - t_{pdmin}(nand) + t_{PHLmax}(244);$ (4)

(1): 10ns < 43ns - 25ns -3,5ns -5ns + 1ns; ✓
10ns < 10,5ns;
(2): 0ns < 5,5ns + 1,5ns - 4,1ns; ✓
(3): 10ns < 30,5ns - 5,5ns +1ns; ✓
(4): 35ns > 57ns - 25ns - 1,5ns - 1,5ns + 4,1ns; ✓
35ns > 33,1ns

Write Cycle Timing for CASx\ and Data valid:

$t_h(CLD) < t_h(CASL\text{-}DV) - t_{pdmax}(nand) + t_{PHLmin}(244);$ (5)

15ns < 30,5ns - 5,5ns + 1ns; ✓

Read Cycle Timing for CASx\ and Data valid:

$t_a(CASL\text{-}DV) > t_a(C) + t_{pdmax}(nand) + t_{PLHmax}(244);$ (6)

24,5ns > 20ns + 5,5ns + 4,1ns; †

3 cycles/column access (CT[2:0] = 111): t_a(CASL-DV) = 40ns:
40ns > 20ns + 5,5ns + 4,1ns; ✓

2.3.8 Define names used in the evaluation

$t_h(CLD)$	VRAM's minimum hold time, data valid after CASx\ low
$t_h(CLCA)$	VRAM's minimum hold time, column address valid after CASx\ low
$t_d(RLCA)$	VRAM's maximum delay time, RAS\ low to column address valid
$t_{su}(CA)$	VRAM's minimum set-up time, col. addr. valid before CASx\ low
$t_a(C)$	VRAM's maximum access time, col. addr. valid to data valid
$t_h(RA)$	VRAM's minimum hold time, row address valid after RAS\ low
$t_{PDmin}(D\text{-}FF)$	minimum delay time of a PAL's Flip Flop after CLK↑
$t_{PDmax}(D\text{-}FF)$	maximum delay time of a PAL's Flip Flop after CLK↑
$t_{pdmin}(nand)$	minimum delay time of a PAL's NAND

t_{pdmax}(nand) maximum delay time of a PAL's NAND
t_{PHLmin}(244) minimum delay time of SN74LVT16244, transition high to low
t_{PHLmax}(244) maximum delay time of SN74LVT16244, transition high to low
t_H defined in 'C8x specification "advance information"
t_a(CASL-DV) defined in 'C8x specification "advance information"
t_h(RASL-RAV) defined in 'C8x specification "advance information"
t_h(CASL-CAV) defined in 'C8x specification "advance information"
t_h(CASL-DV) defined in 'C8x specification "advance information"
t_d(RASL-CAV) defined in 'C8x specification "advance information"
t_{su}(CAV-CASL) defined in 'C8x specification "advance information"

2.3.9 Memory Identification Signals

In addition to providing the EPROM with various signals, external logic must also provide the 'C80 with some control signals; namely, AS[2:0], BS[1:0], PS[3:0], CT[2:0] and UTIME\ to describe the types of cycles that the 'C80 should generate. These signals are sampled at row time during the r^2 state. The logic which creates these signals must also be responsible for decoding this information for other memory types and peripherals in the system as well. [1].

For some special cycle types no true addresses occur on the address bus (e.g. DRAM Refresh Cycles). If one of those cycles is used in the system the logic must decode the status lines additionally to the MSBs of the address (see above Table 8).

As shown in the timing evaluation the VRAMs require the 3-cycle per column access mode for proper operation.
⇒ CT2 = 1, CT1 = 1 and CT0 = 1 ([11], 'C8x Rev. 3.0 Addendum p.7-5)

User modified timing mode must be suppressed.
⇒ UTIME\ = 1 ([13], TC User's Guide p. 7-17)

Four VRAMs build a bus size of 64 bit.
⇒ BS1 = 1 and BS0 = 1 ([13], TC User's Guide p. 7-18)

The 'C8x supports block write cycles only for a bus size of 64 bit. Therefore BS[1:0] are used to determine the block write mode in the beginning of a block write access. In case of STATUS[5:0] = 001101 (load color register) or 001001 (block write PT) BS[1:0] must be set on 10. This will select the 4x mode supported by the TMS55160 VRAMs.

For the VRAMs the address output must be shifted during column time for 12 bits.
⇒ AS2 = 0, AS1 = 1 and AS0 = 0 ([13], TC User's Guide p. 7-12)

Every new page of the VRAM bank starts with a different value build by address bits A21-A12.
⇒ PS3 = 1, PS2 = 0, PS1 = 1 and PS0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.4-1)

The logic equations for the memory identification signals can be seen in the proLogic files in the appendix. The logic is realized in the parts U57, U58 (TIBPAL20L8) and U54 (TIBPAL16L8).

UTIME\ defines during a reset the endian mode for the 'C8x. Note that for this reason UTIME\ must be additional combined with the RESET\ signal.

([13], TC User's Guide p. 9-6).

2.4 Connecting SDRAM TMS626802-15 to 'C8x

2.4.1 SDRAM System Overview

In this chapter of the application report, the following system will be considered.

* Driver is optional. Not realized on TMS320C8x - PC - Testboard Rev. 1.0

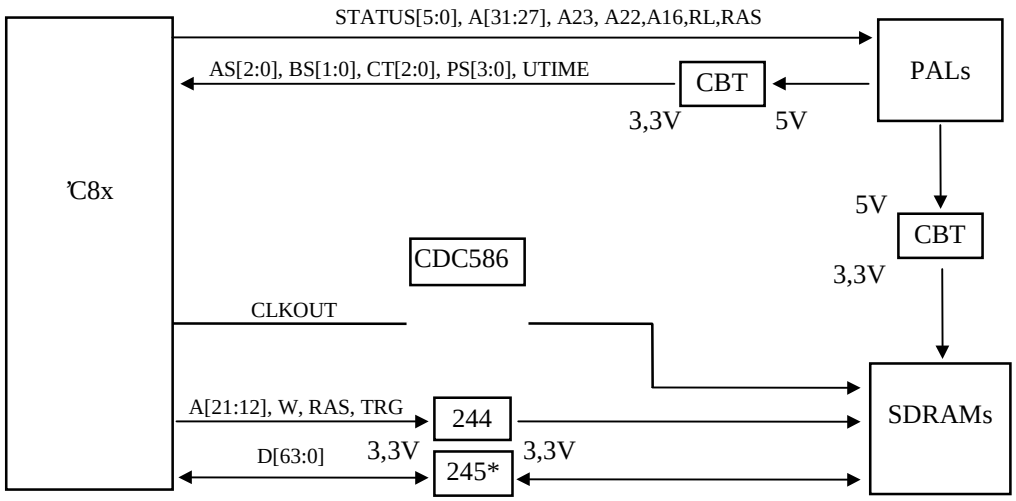

Figure 11: SDRAM Interface Block Diagram

2.4.2 Affected Row Time Status Codes

('C8x Rev. 3.0 Addendum p. 6-1)

Table 12: Status codes (row time) which affect the SRAM

STATUS[5:0]						Cycle Type	TC outputs valid Row Address on A[31:0]?	MNEMONIC
0	0	0	0	0	0	Normal Read	yes	READ
0	0	0	0	0	1	Normal Write	yes	WRT
0	0	0	0	1	0	CAS-before-RAS Refresh	no	REFR
0	0	0	0	1	1	Deactivate all banks	no	DCAB
0	0	1	1	0	0	Mode register set	no	MRS

1	x	x	x	x	x	XPT-Reads	yes	READ
1	x	x	x	x	x	XPT-Writes	yes	WRT

2.4.3 Logic for CS

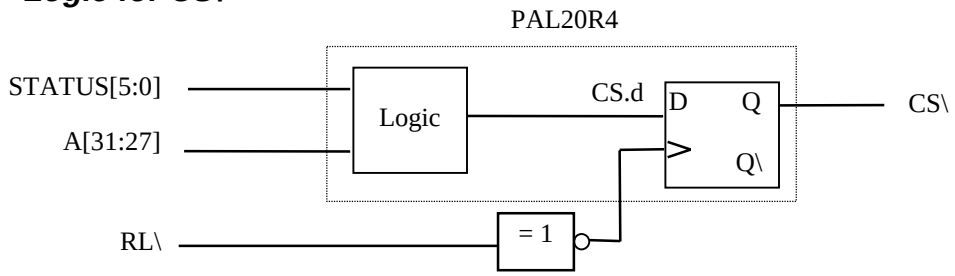

Figure 12: Logic diagram for CS*

A16 alternates VRAM- and SDRAM-Refresh Cycles

```
!CS.d = !A31 & !A30 & A29 & !A28 & !A27                              /*READ
      & !STAT5 & !STAT4 & !STAT3 & !STAT2 & !STAT1                   /*or WRT
    | !STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & STAT0              /*DCAB
    | !STAT5 & !STAT4 & STAT3 & STAT2 & !STAT1 & !STAT0              /*MRS
    | !STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & !STAT0 & !A16;     /*REFR
```

2.4.4 Address bus

2.4.4.1 A10 (Automatic-Precharge Select) and A11 (Bank Select)

The address output for A11 and A10 is correct only during row-time. The row-time value of A11 (Bank Select) must be held during Column accesses. Therefore it is latched with the falling edge of RL\. The requirements of A10 are more complicated. During row time SDRAM needs the normal row address output on A10. At column time it decides between accesses with and without automatic deactivation. As the 'C8x supports only access modes without automatic deactivation, A10 must be set on a low level during column time.

RAS\ is low only at the time of the ACTV-command. During column time it is always high. Therefore it is used in the logic for A10 as a criterion to decide between row and column time.

(Please compare 'C8x Rev 3.0 Addendum pages 7-11 to 7-16)

```
A10(SDRAM) = A22(TC) & !RAS\
```

SDRAMs need multiplexed addressing. Therefore 'C8x's A23 and A22 correspond SDRAMs A11 and A10. (bus size = 64 bit)

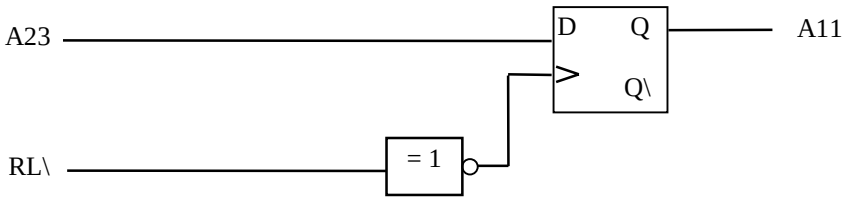

Figure 13: Logic diagram for A11

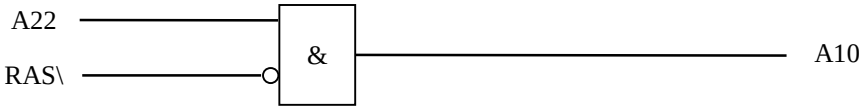

Figure 14: Logic diagram for A10

2.4.4.2 Address bus bits A0 - A9

With respect to the bus size and the length of the column address word, address lines must be connected in the following way.

Table 13: Connecting address bus

Bus Size	'C8x Outputs	SDRAM Inputs
8 bit	A[18:9]	A[9:0]
	A19 + logic	A10
	A20 + latched	A11
16 bit	A[19:10]	A[9:0]
	A20 + logic	A10
	A21 + latched	A11
32 bit	A[20:11]	A[9:0]
	A21 + logic	A10
	A22 + latched	A11
64 bit	A[21:12]	A[9:0]
	A22 + logic	A10
	A23 + latched	A11

2.4.5 Data and CAS signals

As the LSBs of the address bus are not connected to the SDRAM, the TC creates the byte select with the CAS-signals.

If bus size is less than 64 bits, Big- and Little Endian Mode need different signals to be connected to the memory bank.

Table 14: Connecting data bus and CAS signals

	Data Bus		CAS-Signals	
Bus Size	Big Endian	Little Endian	Big Endian	Little Endian
8 Bit	D[63:56]	D[7:0]	CAS7	CAS0
16 Bit	D[63:48]	D[15:0]	CAS[7:6]	CAS[1:0]
32 Bit	D[63:32]	D[31:0]	CAS[7:4]	CAS[3:0]
64 Bit	D[63:0]	D[63:0]	CAS[7:0]	CAS[7:0]

2.4.6 Timing Evaluation

Read Latency 2 is the fastest mode ’C8x and SDRAM support. At first timing evaluation tests values for this mode.

The ’C8x works with a frequency of 40MHz (t_H = 12.5ns).

A10

$t_{su}(A10) < t_{su}(RASL\text{-}CKOH) - t_{pdmax}(nand)$

2ns < 12,5ns - 5,5ns - 5ns ✓

$t_h(A10) < t_h(CKOH\text{-}RASL) + t_{pdmin}(nand)$

4ns < 12,5ns -5,5ns + 1,5ns ✓

A11 and CS\

$t_{su}(A11) < t_{wmin}(RL\text{-}RASL) - t_{PLHmax}(inv) - t_{pdmax}(D\text{-}FF)$

2ns < 75ns - 7ns - 5ns - 4,5ns ✓

RAS\

Without Driver

$t_{su}(RAS) < t_{su}(RASL\text{-}CKOH);$ 2ns < 12,5ns - 5,5ns; ✓

$t_h(RAS) < t_h(CKOH\text{-}RASL);$ 4ns < 12,5ns - 5,5ns; ✓

With Driver

$t_{su}(RAS)<t_{su}(RASL\text{-}CKOH)-t_{PHLmax}(244);$ 2ns < 12,5ns - 5,5ns - 4,1ns; ✓

$t_h(RAS)<t_h(CKOH\text{-}RASL)+t_{PHLmin}(244);$ 4ns < 12,5ns - 5,5ns +1ns ✓

Same Evaluations for Data I/O during Write, A[9:0], W\, CAS\ and DQM.

Data I/O during Read

$t_{ac}(CKOH\text{-}DV) > t_{ac}(SDRAM) + t_{PXXmax}(245)$

Read Latency = 2

16,5ns > 18ns + 4,1ns; †

Take Read Latency = 3 or faster device without driver!

Table 15: SDRAM access time

	maximum access time t_{ac}			unit
	TMS626802-10	TMS626802-12	TMS626802-15	
Read Latency = 1	29	33	38	ns
Read Latency = 2	14	15	18	ns
Read Latency = 3	9	10	12	ns

2.4.7 Define names used in the evaluation:

t_{su}(RASL-CKOH)	minimum setup time, RAS\ low before CLK↑ (= tH - 5.5ns)
t_h(CKOH-RASL)	minimum hold time, RAS\ low after CLK↑ (= tH - 5.5ns)
t_{wmin}(RL-RASL)	defined in 'C8x specification “advance information”
t_H	defined in 'C8x specification “advance information”
t_{ac}(CKOH-DV)	maximum delay time, 'C8x expects data valid after CLK↑(= 16.5ns)
t_{su}(RAS)	SDRAM’s minimum set-up time, RAS\ valid before CLK↑
t_h(RAS)	SDRAM’s minmum hold time, RAS\ valid after CLK↑
t_{su}(A10)	SDRAM’s minimum set-up time, A10 valid before CLK↑
t_h(A10)	SDRAM’s minimum hold time, A10 valid before CLK↑
t_{su}(A11)	SDRAM’s minimum set-up time, A11 valid before CLK↑
t_{ac}(SDRAM)	SDRAM’s maximum access time, data valid after CLK↑
t_{pdmax}(nand)	maximum delay time of a PAL’s NAND
t_{pdmin}(nand)	minimum delay time of a PAL’s NAND
t_{PLHmax}(inv)	maximum delay time of SN74AS04, transition from high to low
t_{pdmax}(D-FF)	maximum delay time of a PAL’s Flip Flop after CLK↑
t_{PHLmax}(244)	maximum delay time of SN74LVT16244, transition high to low
t_{PHLmin}(244)	minimum delay time of SN74LVT16244, transition high to low
t_{PXXmax}(245)	minimum delay time of SN74LVT16245, any transition

2.4.8 Memory Identification Signals

In addition to providing the EPROM with various signals, external logic must also provide the ‘C80 with some control signals; namely, AS[2:0], BS[1:0], PS[3:0], CT[2:0] and UTIME\ to describe the types of cycles that the ‘C80 should generate. These signals are sampled at row time during the r^2 state. The logic which creates these signals must also be responsible for decoding this information for other memory types and peripherals in the system as well. [1]

For some special cycle types no true addresses occur on the address bus (e.g. DRAM Refresh Cycles). If one of those cycles is used in the system the logic must decode the status lines in addition to the MSBs of the address (see above Table 12).

As shown in the timing evaluation the SDRAMs require read latency 3 and burst length 2 for proper operation. CT2 selects SDRAM column cycle timing.
⇒ CT2 = 0, CT1 = 1 and CT0 = 1 ([11], 'C8x Rev. 3.0 Addendum p.7-5)

User modified timing mode is not supported for SDRAM cycles and must be suppressed.
⇒ UTIME\ = 1 ([13], TC User’s Guide p. 7-17)

Eight SDRAMs build a bus size of 64 bit.
⇒ BS1 = 1 and BS0 = 1 ([13], TC User’s Guide p. 7-18)

Address bit A3 must be shifted during column time at bus position A12.
⇒ AS2 = 0, AS1 = 1 and AS0 = 0 ([13], TC User’s Guide p. 7-12)

Every new page of the SDRAM bank starts with a different value build by address bits A23-A12.
⇒ PS3 = 1, PS2 = 0, PS1 = 1 and PS0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.4-1)

The logic equations for the memory identification signals can be seen in the proLogic files in the appendix. The logic is realized in the parts U57, U58 (TIBPAL20L8) and U54 (TIBPAL16L8).

UTIME\ defines during a reset the endian mode for the 'C8x. Note that for this reason UTIME\ must be additional combined with the RESET\ signal.

([13], TC User’s Guide p. 9-6).

2.5 Connecting bi-directional FIFO SN74ACT3632-15 to ’C8x

On the TMS320C8x - PC - Testboard a bi-directional FIFO is used as interface to the ISA-bus. It gives the opportunity to transmit or receive commands and data to or from a host PC. But the FIFO could be used as interface to any other bus system or as a buffer for data I/O of video or audio channels as well.

This chapter will describe only the connection of the SN74ACT3632 to the ’C8x in detail.

2.5.1 I/O System Overview

In this chapter of the application report, the following system will be considered.

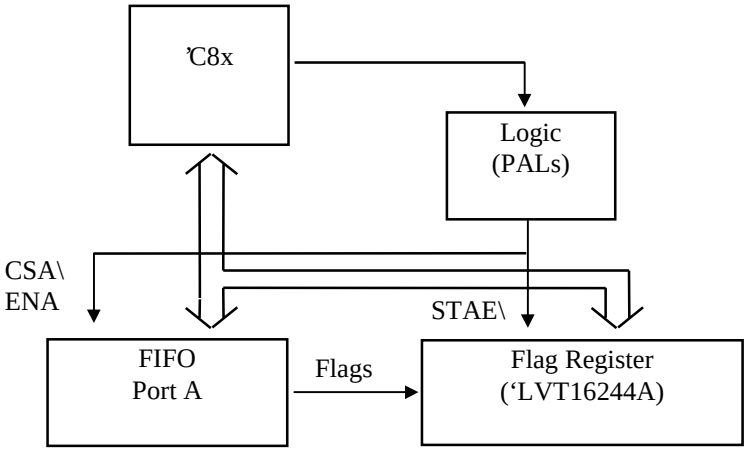

Figure 15: Functional block diagram

Basically connection of the FIFO to the ’C8x can be handled just like the memory interfaces in the chapters before. The main difference is the addressing of this I/O port. In case of transmission of a data block to or from memory devices, every word has its own source or destination address. The FIFO accepts all data under one single address.

As memory space of the FIFO is limited, flags show the actual state of the shift register. They are connected to the inputs of a ‘LVT16244A busdriver. It builds for the 'C8x a read only register with an individual address in the external memory map. The processor can read its value before every FIFO access to avoid data overflow or a read of an empty FIFO.

Flags are used as well to create interrupts in case of invalid FIFO accesses.

2.5.2 Affected Row Time Status Codes

('C8x Rev. 3.0 Addendum p. 6-1)

Table 16: Status codes (row time) which affect the SRAM

STATUS[5:0]						Cycle Type	TC outputs valid Row Address on A[31:0]?	MNEMONIC
0	0	0	0	0	0	Normal Read	yes	READ
0	0	0	0	0	1	Normal Write	yes	WRT
0	0	0	1	0	0	Periph. Device PT Read	yes	READ
0	0	0	1	0	1	Periph. Device PT Write	yes	WRT
0	1	1	1	0	0	PT Read Transfer	yes	READ
0	1	1	1	0	1	PT Write Transfer	yes	WRT
1	x	x	x	x	x	XPT-Reads	yes	READ
1	x	x	x	x	x	XPT-Writes	yes	WRT

2.5.3 The AS\-Signal

Basically the chip enable signals are developed by decoding the MSBs of the address bus. As the TC does not output a true address on A[31:0] (e.g. Refresh Cycle) for every access, bus conflicts can occur. To avoid this, STATUS-lines must be evaluated additionally.

While the TC accesses DRAMs with multiplexed addressing, the MSBs of the address bus are invalid during Column Time. This may also cause bus conflicts. Therefore all chip enable signals must be latched with RL\.

In this application the signal AS\ (Address Shifted) takes all these problems into consideration and is used from every external logic that combines address lines. It shows if the TC outputs a valid address on A[31:0] during row- and column access time.

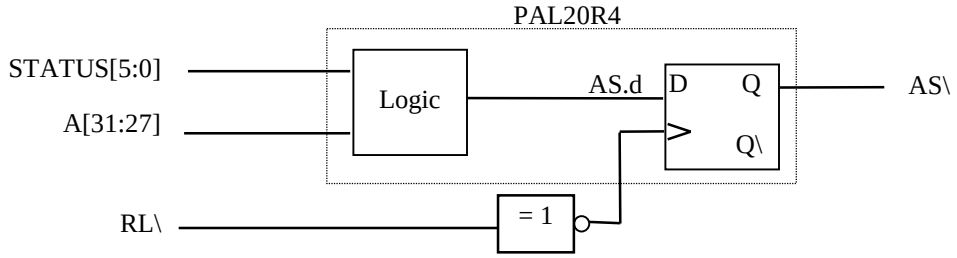

Figure 16: Logic diagram for AS

```
AS.d = !A31 & !A30 & !A29 & !A28 & A27                          /*VRAM-Address
     | !A31 & !A30 & A29 & !A28 & !A27                          /*SDRAM-Address
     | !STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & STAT0       /*SDRAM DCAB
     | !STAT5 & !STAT4 & STAT3 & STAT2 & !STAT1 & !STAT0       /*SDRAM MRS
     | !STAT5 & !STAT4 & !STAT3 & !STAT2 & STAT1 & !STAT0;     /*REFRESH
```

2.5.4 Logic for CSA\ and ENA

'C8x is connected to the port A of the FIFO. Independently of port B, it is controlled by the pins CSA\ and ENA.

```
!CSA = A31 & !A30 & A29 & !A28 & !A27 & !AS;
```

CSA\ and W/RA control the output driver of the data pins. After a read access both must be active long enough to keep data valid when latched by the TC. Thus CSA\ is additionally triggered with the rising edge of CLKOUT.

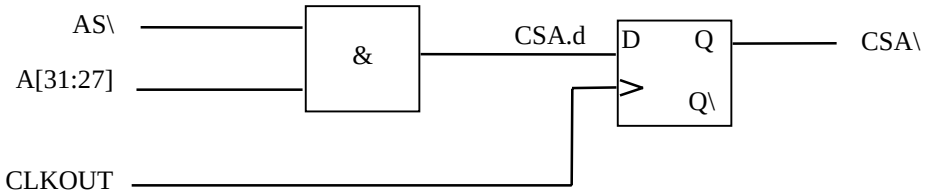

Figure 17: Logic diagram for CSA

Note that every rising edge at the clock input during ENA = high and CSA = low shifts the FIFO registers. Therefore duration of access time must be exactly one clock cycle. If user modified timing is selected, RAS\ goes low for exactly one machine state with every new column access. Connecting RAS\ to the ENA input ensures the right timing in this mode. As ENA is active high, RAS\ must be inverted.

2.5.5 Interrupt Memory Cycle with FAULT

To ensure that there is enough space in the FIFO for a beginning write access, FAULT\ is combined with the almost full flag AFA\ of port A. If there is no memory free in the FIFO, AFA\ is low and FAULT\ will be sampled low at the end of r^2 state. This will cause the 'C8x to break off

the memory access. The repetition of the access is controlled by software, running on the MP. STATUS0 distinguishes between read and write cycles.

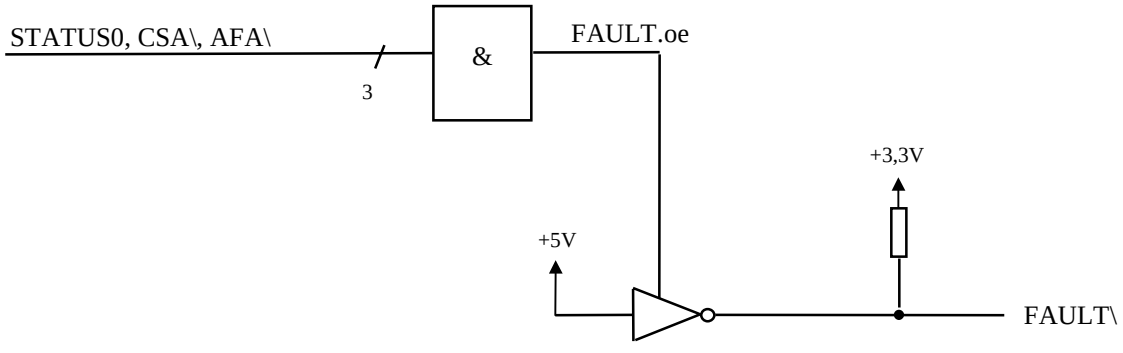

Figure 18: Logic diagram for FAULT

```
!FAULT = 1;
FAULT.oe = STATUS0 & !CSA & !AFA;
```

2.5.6 Logic for STAE

STAE\ controls the output drivers of the FIFO flag register, built by a ‘LVT16244A block.

STAE\ = Output Enable Flag Register Port A

```
!STAE = A31 & !A30 & A29 & !A28 & A27 & AS;
```

2.5.7 Data bus

If bus size is less than 64 bits, Big- and Little Endian Mode need different lines to be connected to the FIFO.

Table 17: Connecting data bus to ‘ACT3632

	Data Bus	
Bus Size	Big Endian	Little Endian
8 Bit	D[63:56]	D[7:0]
16 Bit	D[63:48]	D[15:0]
32 Bit	D[63:32]	D[31:0]
64 Bit	D[63:0]	D[63:0]

In this application several devices with small bus sizes are connected to the ’C8x. To distribute bus load to all lines of the data bus, the FIFO and the flag register are connected to upper bytes of the data bus. Bus size is set to 64 bits in those cases. The offset on the data bus is taken into account by the LSBs of the address word. Corresponding to them, TC will expect valid data on bits as follows:

Little endian mode, bus size code BS1 = 1, BS0 = 1:

Table 18: Shift data bits with LSBs of address word (little endian mode)

Address value	Corresponding data bits			
	Bus size of peripheral device			
	8 bit	16 bit	32 bit	64 bit
0x.....0	D[7:0]	D[15:0]	D[31:0]	D[63:0]
0x.....1	D[15:8]	D[23:8]	D[39:8]	-
0x.....2	D[23:16]	D[31:16]	D[47:16]	-
0x.....3	D[31:24]	D[39:24]	D[55:24]	-
0x.....4	D[39:32]	D[47:32]	D[63:32]	-
0x.....5	D[47:40]	D[55:40]	-	-
0x.....6	D[55:48]	D[63:48]	-	-
0x.....7	D[63:56]	-	-	-

Big endian mode, bus size code BS1 = 1, BS0 = 1:

Table 19: Shift data bits with LSBs of address word (big endian mode)

Address value	Corresponding data bits			
	Bus size of peripheral device			
	8 bit	16 bit	32 bit	64 bit
0x.....0	D[63:56]	-	-	-
0x.....1	D[55:48]	D[63:48]	-	-
0x.....2	D[47:40]	D[55:40]	-	-
0x.....3	D[39:32]	D[47:32]	D[63:32]	-
0x.....4	D[31:24]	D[39:24]	D[55:24]	-
0x.....5	D[23:16]	D[31:16]	D[47:16]	-
0x.....6	D[15:8]	D[23:8]	D[39:8]	-
0x.....7	D[7:0]	D[15:0]	D[31:0]	D[63:0]

2.5.8 Timing Evaluation

For timing evaluation CT2 = 1, CT1 = 1, CT0 = 0 is assumed at first. This is the fastest mode FIFO could be accessed. Only two equations show critical timing:

ENA

$t_{su}(EN) < t_H - t_{pdmax}(nand);$ (1)

4,5ns < 12,5ns - 5,5ns; ✓

Timing Data valid, CLKOUT↑ (Read Cycle)

$t_{amax}(FIFO) < 2t_H - t_{su}(DV\text{-}CKOH) - t_{Pxxmax}(245);$ (2)

11ns < 25ns - 8ns - 4,1ns; ✓

2.5.9 Define names used in the evaluation

$t_{su}(EN)$	FIFO's minimum set-up time, ENA high before CLK↑
t_H	defined in 'C8x specification "advance information"
$t_{pdmax}(nand)$	maximum delay time of a PAL's NAND
$t_{amax}(FIFO)$	FIFO's maximum access time, data valid after CLK↑
$t_{su}(DV\text{-}CKOH)$	defined in 'C8x specification "advance information"
$t_{PXXmax}(245)$	minimum delay time of SN74LVT16245, any transition

2.5.10 Memory Identification Signals

In addition to providing the EPROM with various signals, external logic must also provide the 'C80 with some control signals; namely, AS[2:0], BS[1:0], PS[3:0], CT[2:0] and UTIME\ to describe the types of cycles that the 'C80 should generate. These signals are sampled at row time during the r^2 state. The logic which creates these signals must also be responsible for decoding this information for other memory types and peripherals in the system as well. [1]

For some special cycle types no true addresses occur on the address bus (e.g. DRAM Refresh Cycles). If one of those cycles is used in the system the logic must decode the status lines in addition to the MSBs of the address (see Table 16).

As shown in the timing evaluation the FIFO works in the fast 1 cycle per column access mode.
⇒ CT2 = 1, CT1 = 0 and CT0 = 1.

Accesses to the flag register (‘LVT16244A) need valid address output during TC samples the data. The 2 cycle per column mode is, for that reason, the fastest possibility.
⇒ CT2 = 1, CT1 = 1 and CT0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.7-5)

Timing evaluation showed the need of the user modified timing mode during FIFO accesses.
⇒ UTIME\ = 0 ([13], TC User’s Guide p. 7-17)

As the logic that generates STAE\ does not combine RAS\ nor CAS\, UTIME is optional during a flag register access.

As discussed above, a bus size of 64 bit must be selected for FIFO and flag register.
⇒ BS1 = 1 and BS0 = 1 ([13], TC User’s Guide p. 7-18)

At FIFO and flag register accesses, the address output must be unshifted during column time.
⇒ AS2 = 0, AS1 = 0 and AS0 = 0 ([13], TC User’s Guide p. 7-12)

FIFO is always accessed with the same address. TC’s mechanism which releases a new row access after a change of the actual page will not work. Nevertheless TC is set in page mode for FIFO accesses to speed up transfers of more than one single access. Any PS code except of 1000 can be selected.
⇒ PS3 = 1, PS2 = 0, PS1 = 1 and PS0 = 0 ([11], 'C8x Rev. 3.0 Addendum p.4-1)

There is no point in accessing the flag register more than one time in a row. Thus page size code can be set on any value in that case.

The logic equations for the memory identification signals can be seen in the proLogic files in the appendix. The logic is realized in the parts U57, U58 (TIBPAL20L8) and U54 (TIBPAL16L8).

UTIME\ defines during a reset the endian mode for the 'C8x. Note that for this reason UTIME\ must be additional combined with the RESET\ signal.

([13], TC User’s Guide p. 9-6).

3. Further Hardware Requirements

3.1 Refresh

The DRAM-technology of SDRAMs and VRAMs need regular refresh cycles to keep data valid in their memory cells. In the MP's control register REFCNTL the refresh rate must be programmed to request refreshes often enough to fulfil DRAM requirements. On the other hand, refreshes should not occur too often, to keep TC's effectiveness.

To program the refresh rate, the number of clock cycles must be written in the REFRATE register, which should occur on average between two refresh cycles.

REFRATE is the lower half of 'C8x's REFCNTL register.

MP Byte Address
0x0182 0000

REFCNTL register

RPARLD	REFRATE
31 16	15 0

Every refresh cycle can refresh a whole row of a DRAM-memory. The VRAM array contains 512 rows. Within VRAM's refresh time t_{REF} = 8ms every row must be refreshed at least one time. SDRAM array contains 4096 rows which must be refreshed within 64ms. This results in 512 rows in 8ms, just as for the VRAMs.

SDRAM and VRAM need different refresh cycles, adjusted by column timing control pins CT[2:0]. This results in a refresh rate of 2 · 512 refresh cycles in 8ms.

During refresh cycles TC outputs on address bits A[31:16] the refresh pseudo address. As it is decremented with every refresh cycle, A16 is used to alternate CT-codes between SDRAM and VRAM refresh.

Evaluating the value to be written into REFRATE:

Maximum time between two refresh cycles:
T_{REF} = 8ms / (2 · 512) = 7,8125µs;

Maximum clock cycles between two refresh cycles ($f_{'C8x}$ = 40MHz, T_{CLKOUT} = 25ns):
REFRATE = T_{REF} / T_{CLKOUT} = 7,8125µs / 25ns = 312,5 = 312. (round off!)

Load the REFRATE value 312 into REFCNTL during boot routine!

3.2 Reset

With RESET\ = low a hardware reset can be released at any time. This makes the 'C8x drive all control registers to their default values. The minimum pulse duration for RESET\ = low is not determined yet. To be certain that the pulse length was adequate, a period of 1 ms was chosen for TMS320C8x - PC - Testboard. With the following rising edge of RESET\ the state of 'C8x inputs UTIME\ and HREQ\ decide the mode, processor will come up running.

For UTIME\ = low 'C8x will work in big endian mode. For a high level during reset, little endian mode will be selected. Notice, that endian mode can be changed only with a hardware reset. Therefore endian mode is set by the hardware designer.

HREQ\ determines the power-up state of the MP during reset. If HREQ\ is low at the rising edge of RESET\, the MP comes up running. If HREQ\ is high, the MP remains halted until the first interrupt occurrence on EINT3\.

For these reasons HREQ\ and UTIME\ must be combined with RESET\, in addition to their other system specific requirements.

After the rising edge of RESET\ no hold time for HREQ\ and UTIME\ is required. Thus logic for both signals can be realized very easily. Timing evaluation is not necessary.

3.3 5-V to 3V Translation

The ’C8x is a pure low voltage device. Any voltage level on a ’C8x input higher than 3.3V + 10% can destroy the device. If 3-V and 5-V devices are used jointly on a board, a level translation may be necessary for several signals, to protect low voltage devices.

In most cases data bus signals are connected via a driver to the 5-V components. For drivers of the low voltage family input levels of 5-V are no problem. Their output level is limited to 3.3V. Thus no additional level conversion is necessary.

For some signals timing evaluation may not allow implementation of a driver. In that cases CBTs can realize level translation in a very easy way. They introduce next to no propagation delay.

As shown in figure C3.1, a CBT implements only a N-channel MOS transistor, driven by a C-MOS gate. With a diode Vcc can be easily reduced to +4.3V due to the diode drop of approximately 0.7V. This drop, coupled with the gate-to-source voltage drop of 1V, brings the output voltage to a maximum 3.3-V level that can be used to drive a signal in a 3-V environment.

3.4 Reduction of reflection noise

To keep reflections on bus system under control, bus termination arrays are implemented at the end of every critical signal line. The SN74S1053 contain two schottky barrier diodes for every signal line. They clamp signal level to GND or Vcc. Voltage peaks will be cut off.

Furthermore all pins of a net are connected from point to point to reduce the number of reflection sources. Adaptive resistors near signal sources are not implemented in this application. They may provide an additional opportunity to reduce reflection noise.

3.5 Data Bus

Bi-directional bus drivers share data bus into two parts. Due to this, bus load is halved for every output. The faster devices are directly connected to the ’C8x. Therefore 5-V SRAM must be decoupled from ‘C80 with CBTs. All other devices transmit their data to the 'C8x via drivers. 5-V TTL to 3-V TTL is achieved with the ‘LVT16245 in these cases. To reduce bus load on the lower bits of the data bus, devices with small bus size are laid next to each other on the bus. Shifting devices on the data bus is only possible for I/O

components. Memory devices have to build a continuous address space. They must be exactly connected in the way, BS-code shows to the TC.

Both parts of the data bus need their own bus termination.

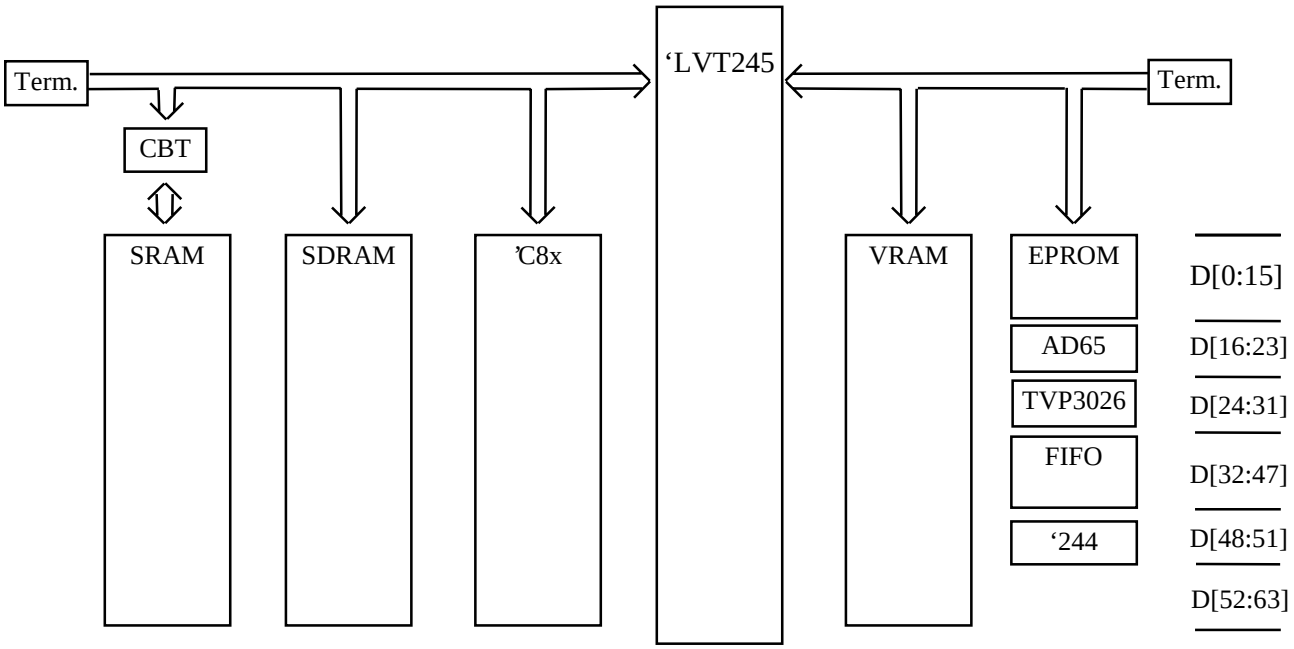

Figure 19: Schematic diagram data bus laying

3.6 Checklist before lay-out should be started

- Is it certain that all 'C8x inputs cannot receive a higher voltage level than 3.3V ± 10%?
- Are all power pins connected to the right supply (5V - 3.3V!)?
- Is the power supply with its additional circuits adequate for all the components employed?
- Have you discussed the endian mode with your software designer?
- Are HREQ\ and UTIME\ defined in the required way during RESET\ = low?
- Are all inputs, even those of unused gates, at a defined level?
- Do by a memory bank disregarded address bits match with the bus size?

disregarded address bits	bus size
-	8 bit
A0	16 bit
A1, A0	32 bit
A2, A1, A0	64 bit

- Did you ensure total bus load does not exceed maximum values specified for device output drivers? (compare Digital Design Workshop [18])

- Are all timing evaluations verified? Did you take into consideration signal delay caused by running time ?
- Are you sure, that all drivers and converters are taken into consideration for timing evaluation?
- Did you ensure that in all cases every signal line is driven by only one output? (bus conflicts suppressed?)
- Is it really possible to realize the logic you have developed in the PALs, GALs, FPGAs or whatever you have designed in?

4. Conclusion

This application report gives you only some tips for your ’C8x design. To be able to develop the hardware for ’C8x peripherals it is still necessary to read at least the chapters of the TC user’s guide, mentioned in this report. For the TMS320C8x - PC - Testboard we have chosen the type and speed of the devices we wanted to connect to the 'C8x before we started the design. By this means we were able to show the high flexibility of the TC.

For a practical configuration you should spend more time to find the right types matching to the 'C8x.

References

1. TMS320C80 SDRAM Application Report, Texas Instruments, 1995
2. Bus Termination Arrays, Texas Instruments, 1993
3. Bus Interface Circuits Widebus™/Widebus+™ Application and Data Book, Texas Instruments, 1995
4. High-Performance FIFO Memories Data Book, Texas Instruments, 1994
5. The proLogic™ Compiler User's Guide, Release 3.00, Texas Instruments, 1993
6. MOS Memory Commercial and Military Specifications Data Book, Texas Instruments, 1993
7. TMS55160 262144 by 16-Bit Multiport Video RAM, Texas Instruments, 1992, Revised March 1994
8. TMS626802 1048576-Word by 8-Bit by 2-Bank Synchronous Dynamic Random-Access Memory, Texas Instruments, 1994, Revised February 1995
9. Programmable Logic Data Book, Texas Instruments, 1990
10. SRAMs PROMs PLDs FPGAs High Performance Data Book, Cypress Semiconductor, 1993
11. TMS320C80 MVP REV 3.0 Addendum, Texas Instruments, 1994
12. TMS320C80 Multimedia Video Processor Advance Information, Texas Instruments, 1994
13. TMS320C80 Transfer Controller User's Guide, Texas Instruments, 1995
14. TMS320C80 Master Processor User's Guide, Texas Instruments, 1995
15. TMS320C80 (MVP) System-Level Synopsis, Texas Instruments, 1994
16. TMS320C80 Video Controller User's Guide, Texas Instruments, 1994
17. TMS34020 User's Guide, Texas Instruments, 1990, p. 3-20
18. Digital Design Workshop, E. Haseloff, P. Forstner, Texas Instruments 1995
19. 'C8x-Workshop, Texas Instruments, 1995