Hardware Interfacing to the TMS32020

APPLICATION REPORT: SPRA126

Authors: Jack Borninski, Jon Bradley, Charles Crowell, and Domingo Garcia
Digital Signal Processing – Semiconductor Group

Digital Signal Processing Solutions
1989

TEXAS INSTRUMENTS

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 application 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.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Hardware Interfacing to the TMS32020

Abstract

This report suggests Hardware design techniques for interfacing memory devices and peripherals to the TMS32020. Examples of PROM, EPROM, static RAM, and dynamic RAM circuits built around the TMS32020 are demonstrated, with timing requirements given for the processor and external devices. Interfaces to a combo-codec and a host computer through UART are also presented.

Product Support on the World Wide Web

Our World Wide Web site at www.ti.com contains the most up to date product information, revisions, and additions. Users registering with TI&ME can build custom information pages and receive new product updates automatically via email.

INTRODUCTION

The TMS32020 Digital Signal Processor has the power and flexibility to cost-effectively implement configurations that satisfy a wide range of system requirements. The large address space of the TMS32020 can be filled in those circuits that require external data or program memory. Peripheral devices can be interfaced to the TMS32020 to perform serial communication and analog signal acquisition.

This application report suggests hardware design techniques for interfacing memory devices and peripherals to the TMS32020. Examples of PROM, EPROM, static RAM, and dynamic RAM circuits built around the TMS32020 are demonstrated, with consideration given to the timing requirements of the processor and external devices. A memory-mapped UART (Universal Asynchronous Receiver-Transmitter) interface for communication with a host computer is presented, as well as an interface to a combo-codec (coder-decoder + filter) device for analog signal acquisition.

All circuits shown in this application report have been built and their operation verified at room temperature. Since the logic devices in these circuits have not been optimized as the most cost-effective, the designer may desire to make tradeoffs with respect to speed, cost, performance, and temperature.

TMS32020 CONSIDERATIONS

The TMS32020 has program, data, and I/O address spaces for interfacing with external memory and peripherals. Memory and I/O devices are usually selected by using the TMS32020 $\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$ signals (program, data, or I/O select, respectively), combined with the $\overline{\mathrm{STRB}}$ (strobe) signal. The signal ($\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$) used depends on the memory or I/O space to be addressed. Some of the read-only devices may be selected (enabled) using any of the $\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$ signals exclusively, whereas the read-write devices commonly use a $\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$ combined with the $\overline{\mathrm{STRB}}$ for an enable signal. The $\overline{\mathrm{STRB}}$, combined with a select line and the TMS32020 R/$\overline{\mathrm{W}}$ (read/write) signal, forms a convenient timing reference to control the write operation of the read/write devices. The read-only devices typically do not require the $\overline{\mathrm{STRB}}$ and R/$\overline{\mathrm{W}}$ combination.

Memory and I/O devices must respond with the data within a maximum of 90 ns when they are selected with either $\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$ during the read cycle. (Refer to the TMS32020 (20-MHz operation) timing diagram in the TMS32020 Data Sheet.) Consequently, the memory or I/O devices used should have correspondingly fast access time. For slower devices, one or more TMS32020 wait states must be inserted for proper operation.

When the $\overline{\mathrm{PS}}$, $\overline{\mathrm{DS}}$, or $\overline{\mathrm{IS}}$ is combined with the $\overline{\mathrm{STRB}}$ signal during the read cycle, the memory or I/O devices must respond within 50 ns of the $\overline{\mathrm{STRB}}$ signal going low, due to the data-read setup time required by the TMS32020. During the write cycle, the TMS32020 provides the minimum of a 55-ns data-write setup time to the memory or I/O devices (see the timing diagrams in the TMS32020 Data Sheet). Wait states also apply to the configuration using the $\overline{\mathrm{STRB}}$ signal.

Tradeoffs with respect to using faster memory and slower decode logic devices, and vice versa, may be made when designing a memory system around the TMS32020.

MEMORIES

Examples of four memory types, PROMs, EPROMs, static RAMs, and dynamic RAMs, are shown in interface to the TMS32020. The selection of which memory device to use in a particular application is determined by speed, cost, and functional requirements. If speed and maximum throughput are desired, the TMS32020 can run with zero wait states and perform memory accesses in a single machine cycle. The TMS32020 can access slower memories by inserting one or more wait states into the memory access operation by using the READY input signal. A circuit using each of the memory devices is described and illustrated in the following subsections of this application report.

PROM

When only fixed program memory is required and speed is a consideration, a PROM device may be chosen for memory interface. A Texas Instruments TBP28S166 PROM (2048 X 8) with a three-state output has been selected as an example interface to the TMS32020. The TBP28S166 has the maximum access time from address of 75 ns, which meets the TMS32020 timing requirements. A basic configuration showing this PROM interfaced to the TMS32020 as program memory is shown in Figure 1.

Another configuration that shows the TBP28S166 interface to the TMS32020 is shown in Figure 2. Here, more memory exists in the system, and the TBP28S166 is decoded and mapped into the program memory address space, starting at >4000.

The timing diagram for the interface of the TBP28S166 to the TMS32020 is shown in Figure 3. No wait states are necessary in both the basic and decoded configurations.

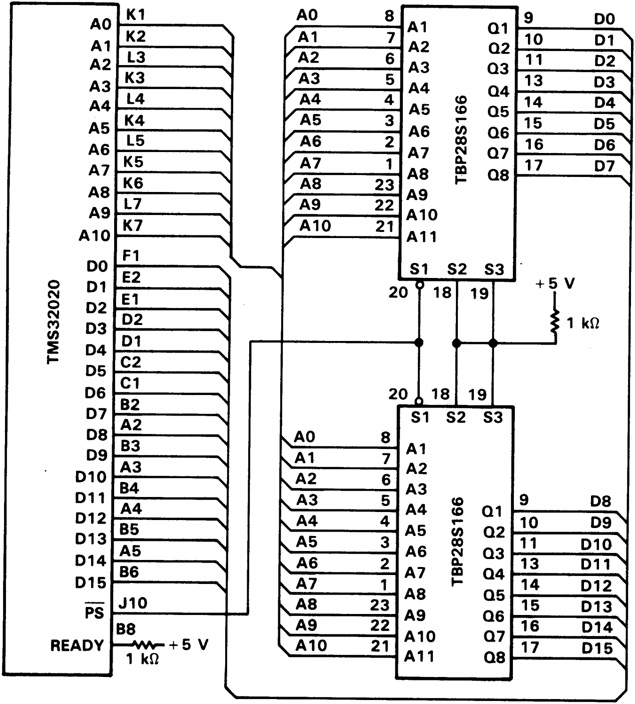
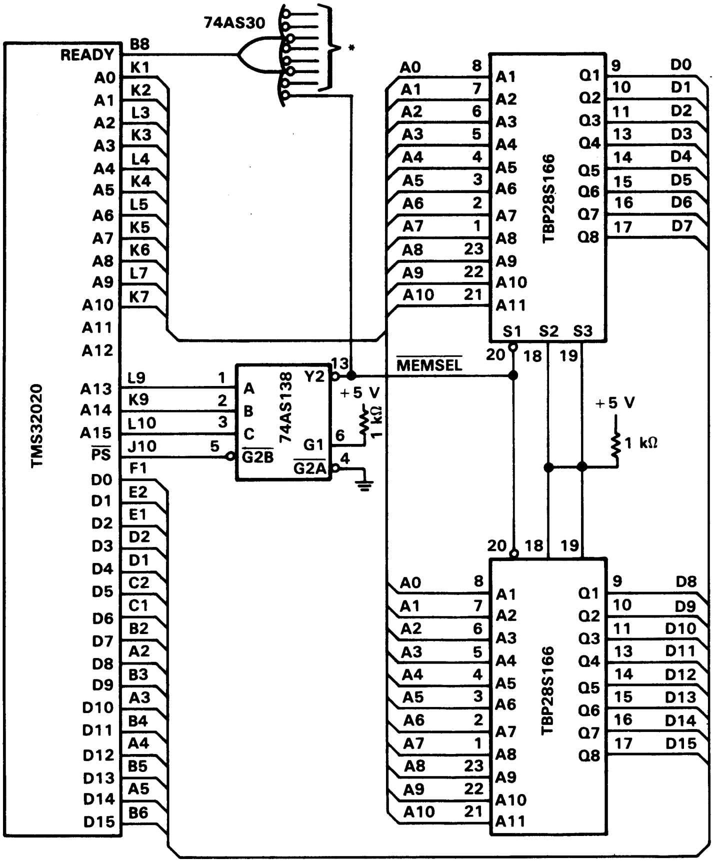

TMS32020
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10
K1 K2 L3 K3 L4 K4 L5 K5 K6 L7 K7
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15
F1 E2 E1 D2 D1 C2 C1 B2 A2 B3 A3 B4 A4 B5 A5 B6
PS
J10
READY
B8
1 kΩ
+5 V
TBP28S166
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11
8 7 6 5 4 3 2 1 23 22 21
Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8
9 10 11 13 14 15 16 17
D0 D1 D2 D3 D4 D5 D6 D7
S1 S2 S3
20 18 19
+5 V
1 kΩ
TBP28S166
D8 D9 D10 D11 D12 D13 D14 D15

Figure 1. Basic Interface of TBP28S166 to TMS32020



*Connections to other devices in the system. (Inputs not used should be pulled up.)

Figure 2. Decoded Interface of TBP28S166 to TMS32020

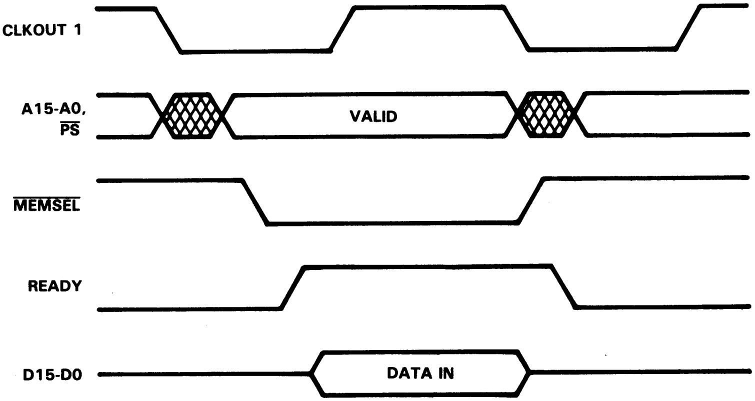


Figure 3. Interface Timing of TBP28S166 to TMS32020

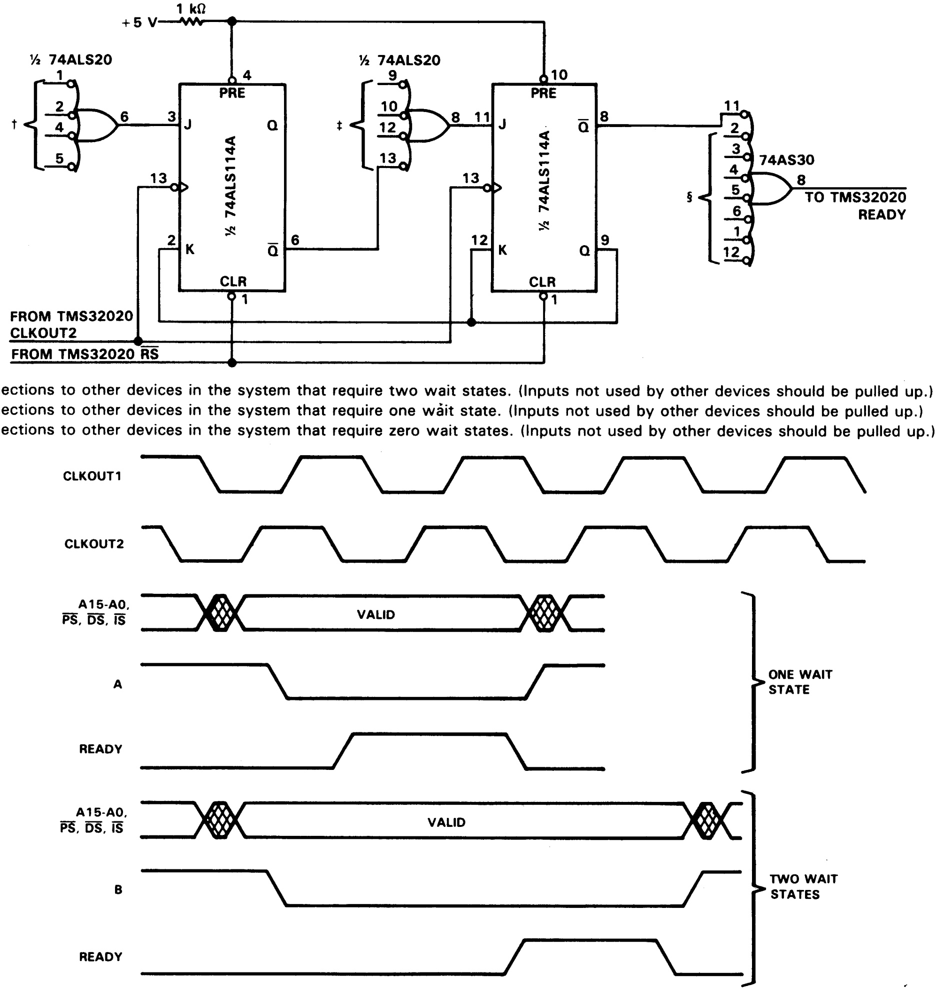
EPROM and Wait-State Generator

During the prototyping and development design stage, an EPROM may be selected as the memory device for interfacing to the TMS32020. The Texas Instruments TMS2764-35 EPROM (8192 X 8) has the access time from address of 350 ns. This does not directly meet the TMS32020 maximum allowable read-data access time of 90 ns, so a series of wait states is used to delay the TMS32020. A wait-state generator circuit for providing up to two 200-ns wait states is shown in Figure 4.

Point A in Figure 4 corresponds to one wait-state input and point B to two wait-state inputs. In the case of the TMS2764-35 EPROM with a 350-ns access time, two 200-ns wait states are needed for proper interfacing. Figure 5 shows the TMS2764 decoded into the program memory address space 0000-1FFF. Due to the long EPROM turn-off time, the 74ALS244 buffers at the EPROM output prevent data bus conflict.

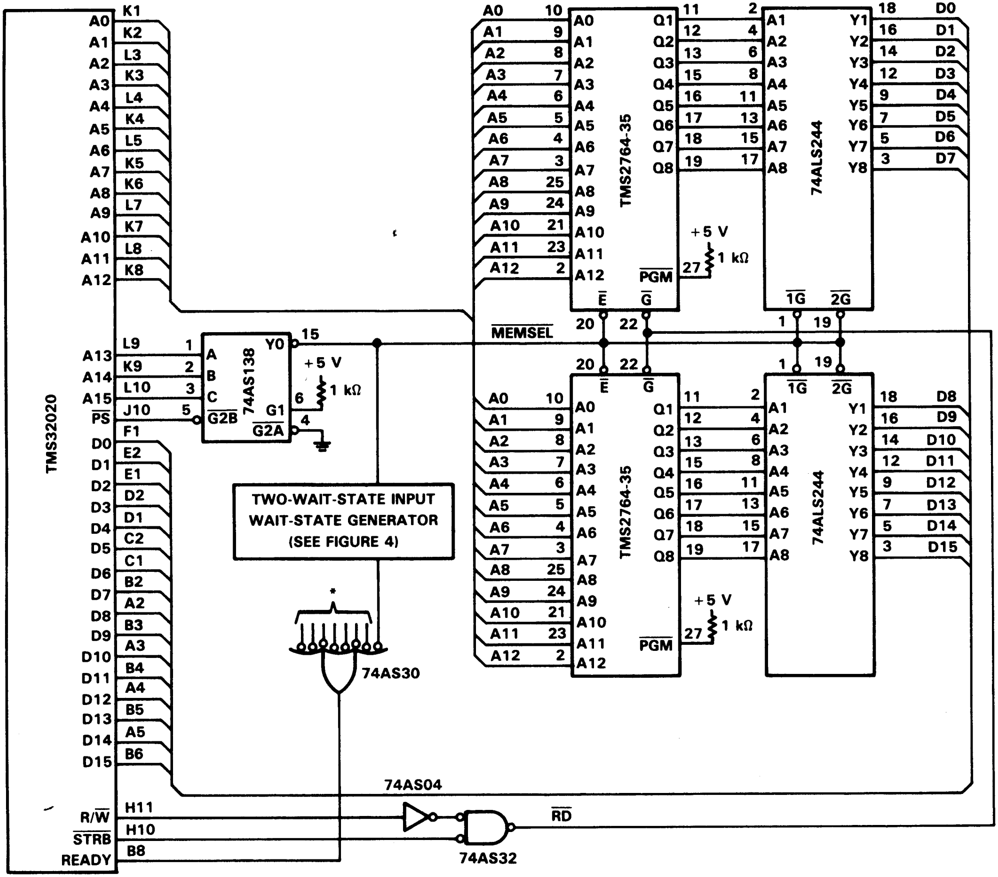
Figure 6 shows the timing considerations for the interface of the TMS2764 to the TMS32020.

If faster EPROMs are desired, TMS2764-25 (250-ns access time from address) EPROMs can be interfaced to the TMS32020. If the circuit shown in Figure 5 is used, the only change required is to connect the MEMSEL (memory select) signal to input A of the wait-state generator (see Figure 4). The TMS2764-25 operates with one wait state in this configuration.



†Connections to other devices in the system that require two wait states. (Inputs not used by other devices should be pulled up.)
‡Connections to other devices in the system that require one wait state. (Inputs not used by other devices should be pulled up.)
§Connections to other devices in the system that require zero wait states. (Inputs not used by other devices should be pulled up.)

Figure 4. Wait-State Generator and Timing



*Connections to other devices in the system. (Inputs not used should be pulled up.)

Figure 5. Decoded Interface of TMS2764 to TMS32020

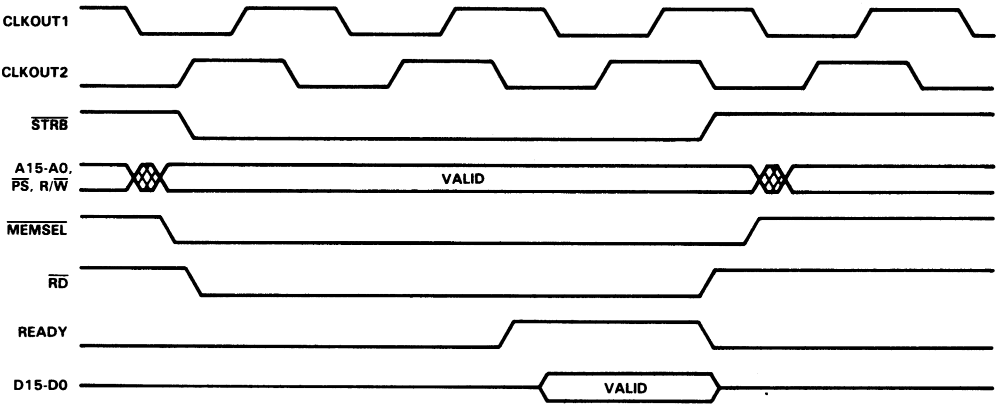


Figure 6. Interface Timing of TMS2764 to TMS32020

Static RAM

Static RAM as program memory is useful for program storage when a remote download capability exists. Static RAM as data memory is used when minimum design overhead and minimum chip count are desired.

Figure 7 shows an example of INMOS IMS1421-50 static RAMs (4096 X 4), interfaced to the TMS32020. These static RAMs are mapped into the program memory address space, starting at >6000.

The read/write timing considerations for the interface of the IMS1421 to the TMS32020 are shown in Figure 8.

During the memory-read cycle, the IMS1421-50 responds with data within 40 ns from being selected. This is in the specified range of the TMS32020, and no wait states are required.

During the memory-write cycle, the falling and rising edges of the $\overline{\text{MEMSEL}}$ signal form the beginning and end of the IMS1421-50 write cycle while the IMS1421 $\overline{\text{W}}$ (write) line is asserted early (early write).

During the early-write cycle, the IMS1421 data-bus drivers never turn on. This helps prevent bus conflicts with the TMS32020. The data supplied to the bus by the TMS32020 meets the timing parameters of the IMS1421-50, and no wait states are required.

The use of slower memories (e.g., IMS1420-45) is possible, provided the designer places external buffers on the data bus between the memory and the TMS32020. Since the internal IMS1420 output buffers do not turn off until the $\overline{\text{W}}$ line goes low, the external buffers (with the appropriate enable logic) prevent data bus conflicts during the write operation.

If CMOS RAMs are required, the IMS1423-35 devices can be used in the circuit shown in Figure 7. These devices operate faster, and their pinouts are identical to that of the IMS1421-50. Slower CMOS RAMs (e.g., IMS1423-45) can also be used; however, the data bus buffers must be added as outlined previously to prevent data bus buffers must be added as outlined previously to prevent data bus conflicts.

The design of the IMS1421 (NMOS) or IMS1423 (CMOS) interface as data memory is very similar to the one presented in Figure 7, and the data and program memory timing are identical. The only change needed is the use of the $\overline{\text{DS}}$ select line (instead of $\overline{\text{PS}}$) in the decode logic.

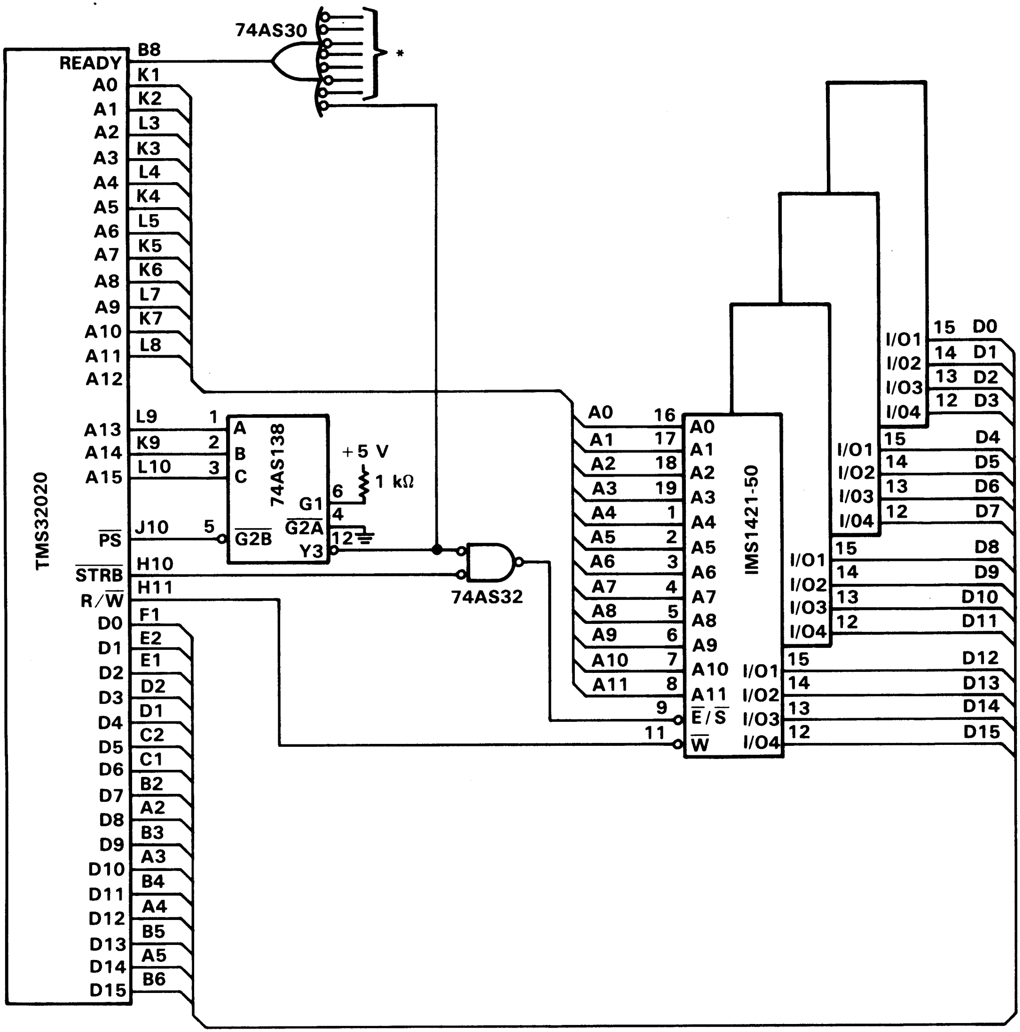
Static memories that do not meet the TMS32020 no-wait-state timing requirements can be interfaced, provided the necessary number of wait states is generated and used to drive the TMS32020 READY line.

Dynamic RAM

In systems where large inexpensive memory space is required, the use of dynamic RAM devices with the associated control circuitry may be justified. This section describes an interface between the Texas Instruments TMS4416-15 (16K X 4 dynamic RAM) and the TMS32020. The circuit shown in Figure 9 uses the TMS4500A Dynamic RAM Controller to supply the control functions for the dynamic RAM devices. The TMS4500A provides address generation, timing, access/refresh arbitration, and other functions to control a bank of TMS4416 dynamic RAMs.

The timing diagram for the memory access and refresh cycles in the interface of the TMS4416 to the TMS32020 is shown in Figure 10.

During the regular access cycle, the TMS4500A $\overline{\text{RAS}}$ (row address strobe) and $\overline{\text{CAS}}$ (column address strobe) signals both become active, and one automatic TMS32020 wait state is generated using the $\overline{\text{MSC}}$ (microstate complete) signal (see Figure 9). The generation of at least one TMS32020 wait state is necessary because of the TMS4416-15 access time of 150 ns. The regular (i.e., uninterrupted by refresh) memory access takes two TMS32020 cycles (400 ns at 20-MHz operation).



*Connections to other devices in the system. (Inputs not used should be pulled up.)

Figure 7. Interface of IMS1421 to TMS32020

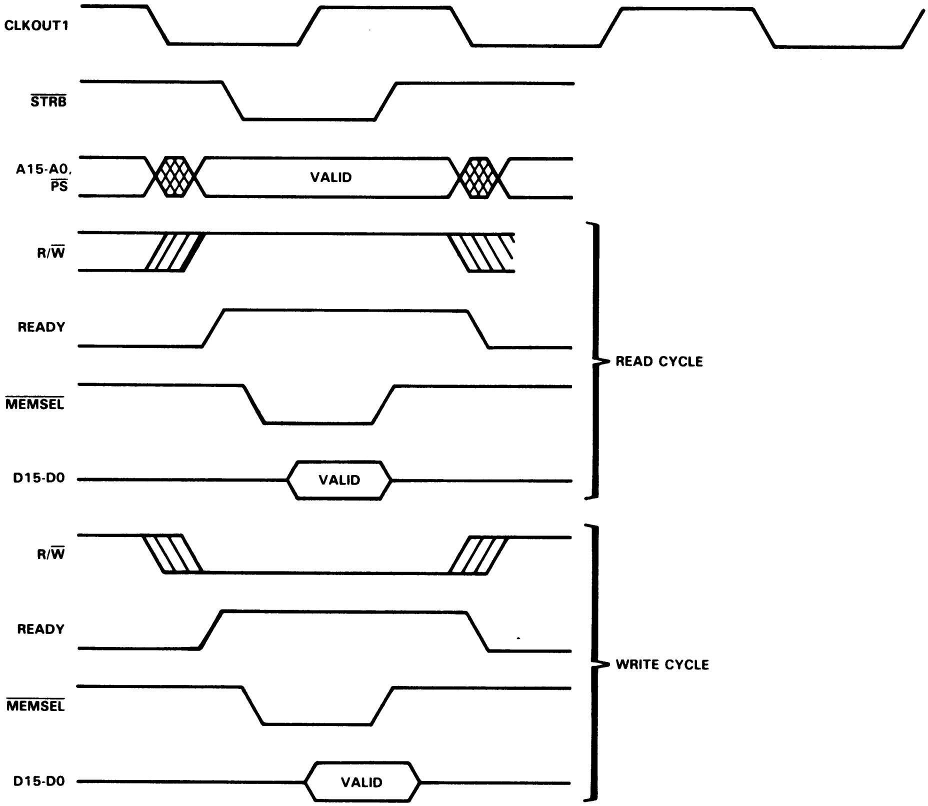


Figure 8. Interface Timing of IMS1421 to TMS32020

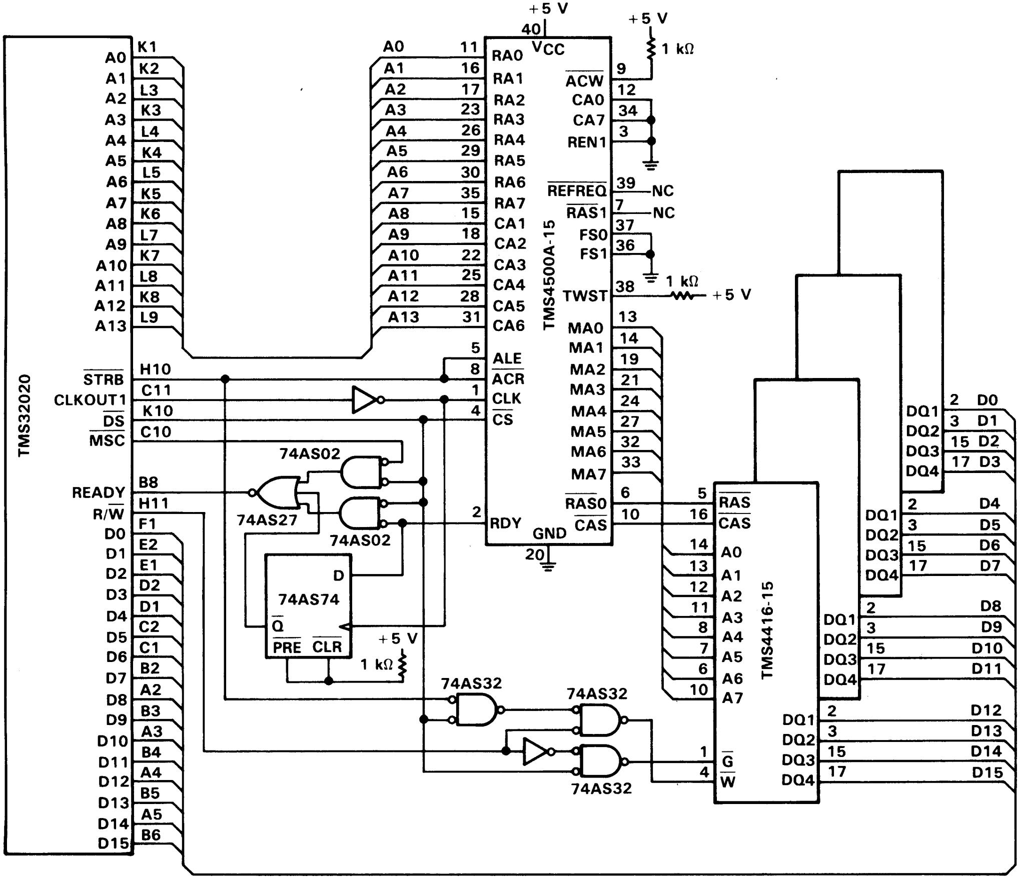
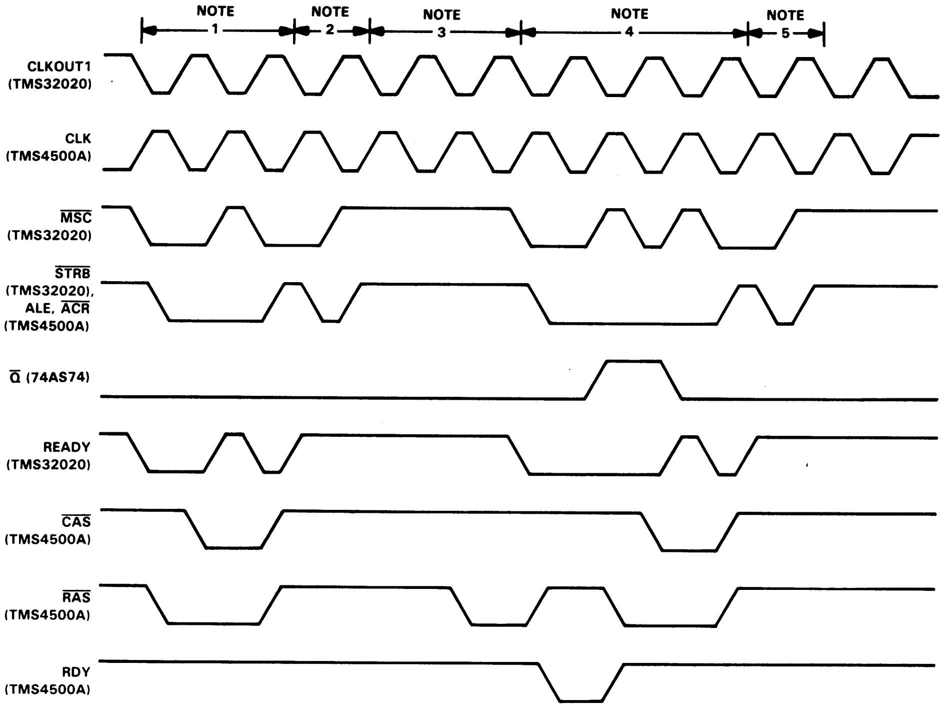


Figure 9. Interface of TMS4416 to TMS32020


NOTE 1
NOTE 2
NOTE 3
NOTE 4
NOTE 5
CLKOUT1 (TMS32020)
CLK (TMS4500A)
$\overline{\text{MSC}}$ (TMS32020)
$\overline{\text{STRB}}$ (TMS32020), ALE, $\overline{\text{ACR}}$ (TMS4500A)
$\overline{\text{Q}}$ (74AS74)
READY (TMS32020)
$\overline{\text{CAS}}$ (TMS4500A)
$\overline{\text{RAS}}$ (TMS4500A)
RDY (TMS4500A)

NOTES: 1. Dynamic RAM access cycle
2. Access cycle for memory other than dynamic RAM
3. Internal TMS32020 instruction cycles
4. Dynamic RAM access cycle with refresh
5. Access cycle for memory other than dynamic RAM

Figure 10. Interface Timing of TMS4416 to TMS32020

During the refresh cycle, which may be observed in Figure 10 at the point where $\overline{\text{RAS}}$ becomes low and $\overline{\text{CAS}}$ stays high, the TMS4500A performs memory access/refresh arbitration. The arbitration must be performed when the TMS32020 tries to access the dynamic RAM memory at the time the TMS4500A begins the refresh operation, as shown in Figure 10. In this case, the TMS32020 must be held for one or two additional wait states, until the refresh operation is completed. The READY signal to the TMS32020 is generated through a combination of three other signals. The $\overline{\text{MSC}}$ signal starts driving the TMS32020 READY line, the TMS4500A RDY (ready) signal continues to drive it, and the output of the 74AS74 flip-flop ($\overline{\text{Q}}$) extends it to the end of the required period, as shown in Figures 9 and 10. Such a circuit is required since the TMS4500A RDY signal alone is not of the proper duration for the TMS32020. After the memory refresh is performed, a regular access cycle takes place as previously described.

In an access/refresh arbitration case, it takes the TMS32020 three or four clock cycles to access the dynamic RAM, depending on the time relationship of the access and refresh requests. The worst-case RAM memory access timing is 800 ns at 20-MHz operation. This occurs when the requests for access and refresh take place nearly simultaneously. The access/refresh arbitration timing, shown in Figure 10, takes three TMS32020 clock cycles, i.e., 600 ns at 20-MHz operation (not the worst-case condition), since the request for access happens right after the end of the refresh cycle.

If the TMS32020 continuously accesses the dynamic memory, it will be interrupted by a refresh cycle once out of ten thousand memory access cycles, i.e., 99.99 percent of the memory accesses will happen without interruption. This low interruption rate is based on the relationship between the TMS32020 dynamic-memory cycle length (400 ns) and the required TMS4416 refresh period (4 ms).

The transition in the TMS32020 READY line, at the end of the READY signal, is caused by the corresponding behavior of the $\overline{\text{MSC}}$ signal during the TMS32020 operation. The transition does not affect the circuit operation since the READY line is sampled on the rising edge of CLKOUT1, i.e., before the transition takes place.

The interface logic between the TMS32020, TMS4500A, and TMS4416 devices, i.e., the gates and a flip-flop (see Figure 9), can be replaced with one Programmable Array Logic (PAL)† integrated circuit for reduced IC count. One such PAL that is well suited to this application is SN74PAL16R4A (see Appendix).

PERIPHERALS

Peripheral devices, such as the UART and combo-codec, can be interfaced to the TMS32020 to perform serial communication and analog signal acquisition. Communication to a host computer can be provided by a memory- or I/O-mapped UART interface. A codec provides analog signal conversion for telecommunications and speech processing.

UART

The UART and TMS32020 configuration may be useful when a general-purpose TMS32020 program development system or upload/download capability from/to the TMS32020-based signal processing system is required.

As an interface example of an asynchronous communication controller to the TMS32020, a General Instruments AY-3-1015D UART is memory-mapped into the TMS32020 data memory space. Figure 11 shows the circuit for interfacing the AY-3-1015D UART to the TMS32020. The transmitter is data memory-mapped into location C000 and the receiver into location E000.

The receiver/transmitter timing diagrams for the interface of the AY-3-1015D to the TMS32020 are shown in Figure 12.

The three-state buffer of the UART receiver has received-data-enable (RDE) timing constraints that preclude its use with the TMS32020 (see the AY-3-1015D specifications sheet). The 74LS244 buffer placed on the UART's receiver port is fast enough to perform the switching function on the TMS32020 data bus.

The one-processor wait state, present during the transmitter operation, allows for positive data strobing into the UART's transmitter data register as required by the AY-3-1015D.

The UART's receiver and transmitter status lines, DAV (data available) and TBMT (transmitter buffer empty), are synchronized with the rest of the system and connected to either the interrupt or $\overline{\text{BIO}}$ (branch on I/O) pin of the TMS32020. This allows either interrupt or polled I/O techniques to be implemented and assures the synchronization of the UART device with the TMS32020 regarding the receiving and transmitting rates. For example, the TMS32020 does not try to read the UART's receiver buffer at a rate greater than the rate at which the characters are coming in or write to the UART's transmitter buffer at a rate greater than the maximum character transmission rate.

The UART can equivalently be mapped into the I/O space of the TMS32020, since the memory and I/O cycles of the TMS32020 are identical. The only change required is the use of the $\overline{\text{IS}}$ (instead of the $\overline{\text{DS}}$) line and of address lines A0-A3 to select an I/O port.

Combo-Codec

In some areas of telecommunications, speech processing, and other applications that require low-cost analog I/O devices, a combo-codec device may be useful. A codec consists of nonlinear A/D and D/A converters with all the associated filters and data-holding registers.

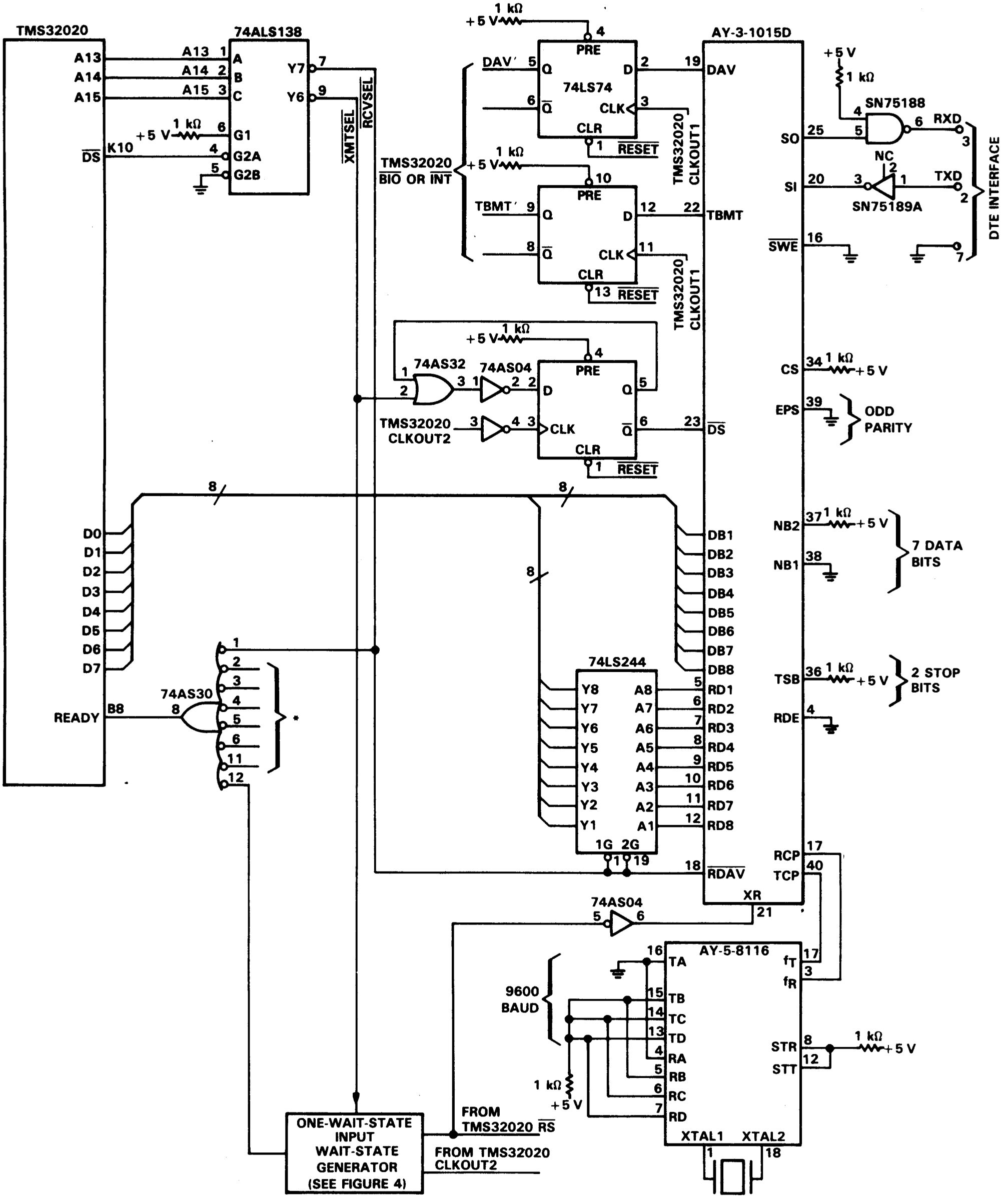
The TMS32020 contains a serial port for communicating to serial devices, such as codecs. The speed and versatility of the TMS32020 allow it to compand (COMpress and exPAND) a PCM (Pulse Code Modulation) data stream, acquired by the codec, through the TMS32020 execution of software conversion routines (see the application report, 'Companding Routines for the TMS32010'). Figure 13 shows an interface example of a Texas Instruments TCM2913 codec to the TMS32020 serial port.

Figure 14 shows the TMS32020 serial port receive and transmit timing considerations in the interface of the TCM2913 to the TMS32020.

In this configuration, the TCM2913 codec functions in the fixed data-rate mode (2.048 MHz) with μ-law operation selected. All timing and synchronization signals are externally generated using independent oscillator and frequency-dividing hardware (an 8-bit counter such as a 74AS867 may be used in place of two 74LS161 4-bit counters to minimize the chip count). Alternatively, the designer may decide to generate the timing signals from the TMS32020 clock by subdividing its frequency.

In some circuits, it may be necessary to wire an op-amp to the analog output of the codec. In such cases or if variable output gain is required, a gain-setting resistor network must be provided as specified in the TCM2913 documentation.

Other linear A/D and D/A converters may be interfaced to the TMS32020 through its parallel ports.



*Connections to other devices in the system. (Inputs not used by other devices should be pulled up.)

Figure 11. Interface of AY-3-1015D to TMS32020

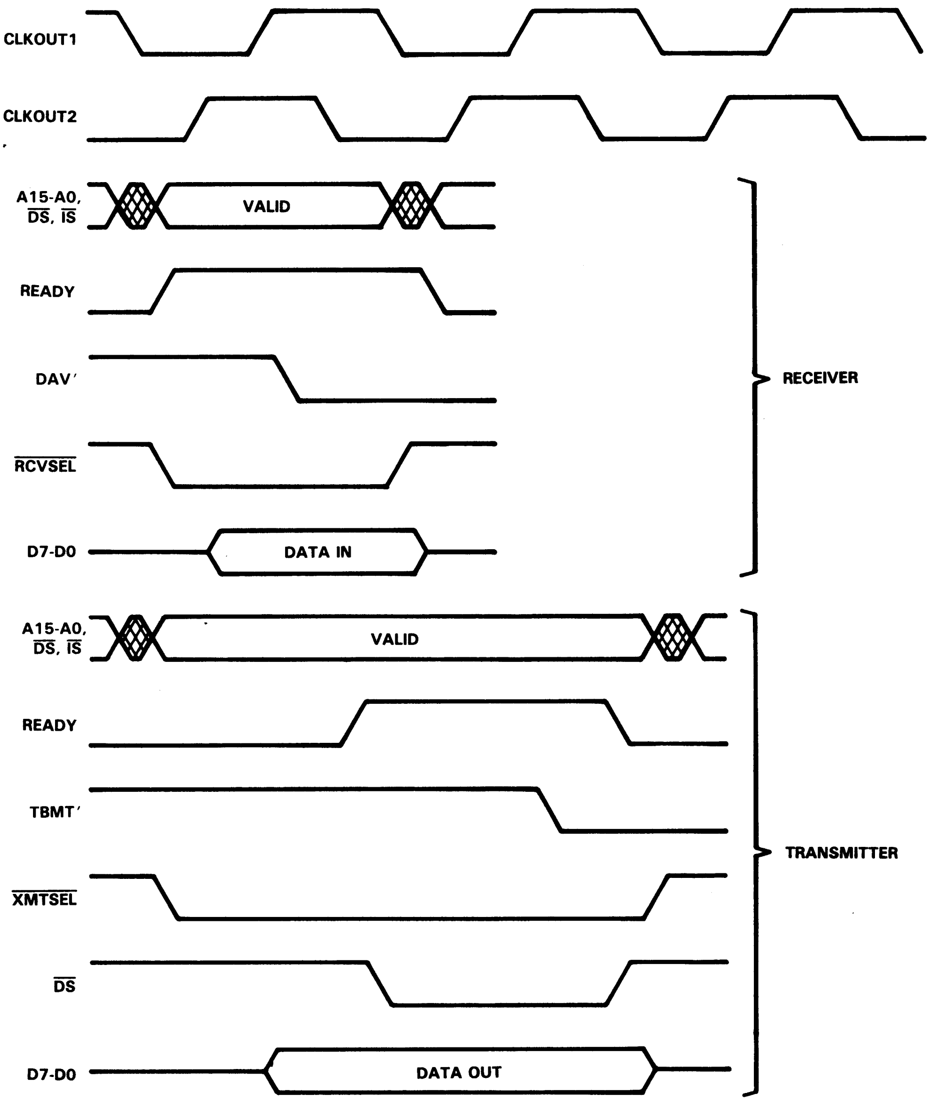


Figure 12. Interface Timing of AY-3-1015D to TMS32020

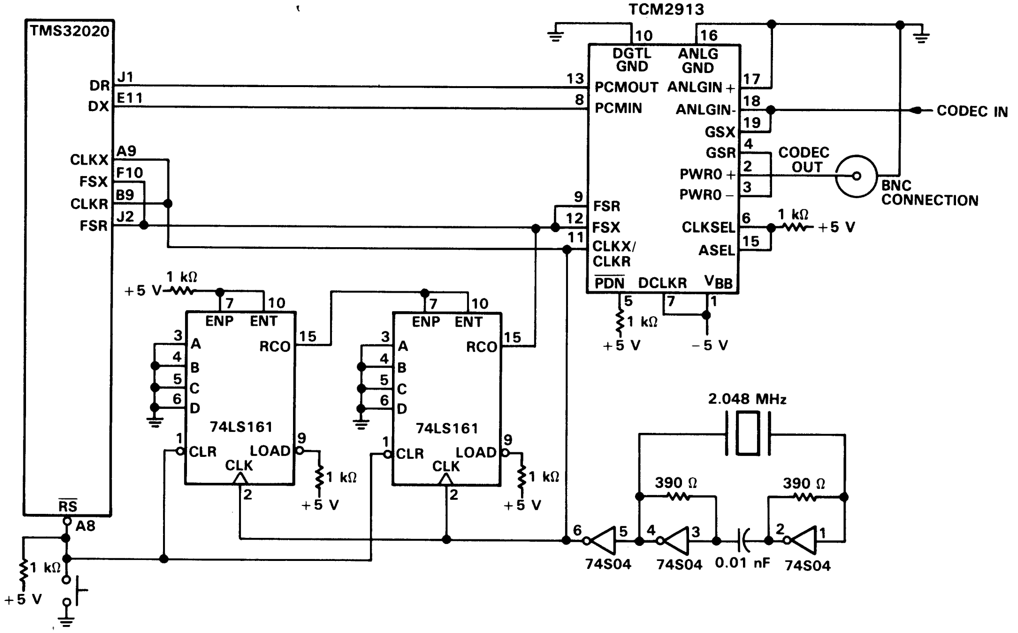


Figure 13. Interface of TCM2913 to TMS32020

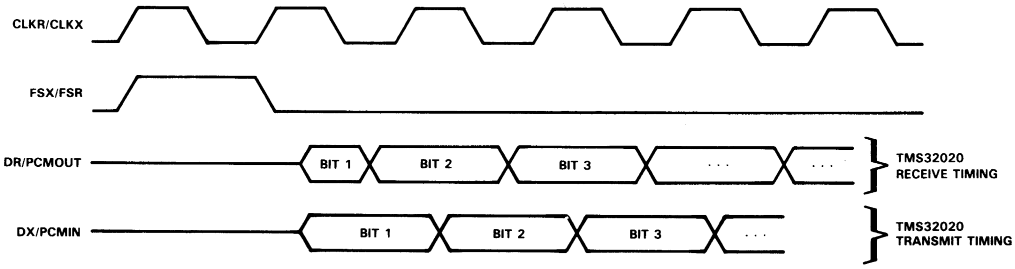


Figure 14. Interface Timing of TCM2913 to TMS32020

SUMMARY

The speed, performance, and flexibility of the TMS32020 allow it to cost-effectively implement configurations that satisfy a wide range of system requirements. This application report has described and demonstrated hardware design techniques for interfacing memory devices and peripherals to the TMS32020.

Examples of PROM, EPROM, static RAM, and dynamic RAM circuits built around the TMS32020 have been provided, with consideration given to the timing requirements of the processor and external devices. Interface examples of UART and codec chips to the TMS32020 are also presented.

Table 1 summarizes the interface requirements of various memory devices to the TMS32020, as described in this report. The table also includes maximum affordable access times for each circuit configuration.

Table 1. Memory Type and TMS32020 Interface Requirements

MEMORY TYPE	PART NUMBER	ACCESS TIME	NUMBER OF WAIT STATES REQUIRED BY THE TMS32020	ACCESS TIME FOR CIRCUIT CONFIGURATION
PROM	TBP28S166	75 ns max (from address)	0	85 ns max (from address)
EPROM	TMS2764-35	350 ns max (from address)	2	371 ns max (from address)
Static RAM	IMS1421-40	30 ns max (from select)	0	39.2 ns max (from select)
Static RAM	IMS1420-55	55 ns max (from select)	0 (buffers reqd)	62 ns max (from select)
Static RAM (CMOS)	IMS1423-35	35 ns max (from select)	0	39.2 ns max (from select)
Static RAM (CMOS)	IMS1423-45	45 ns max (from select)	0 (buffers reqd)	62 ns max (from select)
Dynamic RAM	TMS4416-15	80 ns max (from $\overline{\text{CAS}}$)	1	170 ns max (from $\overline{\text{CAS}}$)

APPENDIX
PAL[†] DESIGN

PAL Assembler Code

```
PAL16R4                                  PAL DESIGN DOCUMENT
0000000-0000                             REV *  5/02/85 BY DANA CROWELL
SHIVA DRAM CONTROLLER

CLK MSC DS RDY CLKOUT1 STROBE RW NC1 NC2 GND
GND /CLK1 /READY /NC3 /NC4 /Q /NC5 /G /W VCC

IF (VCC)
READY      = /MSC*/DS +
             /RDY*/DS +
             Q

IF (VCC)
CLK1       = CLKOUT1

IF (VCC)
W          = /STROBE*/DS*/RW

IF (VCC)
G          = /DS*RW

IF (VCC)
Q          = /RDY


FUNCTION TABLE

CLK MSC DS RDY CLKOUT1 STROBE RW
GND /CLK1 /READY /W /G /Q

;-------------------+-----------------+
;                   ¦                 ¦
;   C               ¦           S     ¦
; C L         A     ¦   R M     B M   ¦
; L K       O R R R ¦   D E     M E   ¦
; K O       F E E E ¦ S Y M W I A M   ¦
; O U   M   F A A S ¦ Y P P A D C C   ¦
; U T R E A I D D E ¦ N A A I L Y Y W ¦
; T 2 W M O O Y Y T ¦ C L L T E C C E ¦    PATH   VECTOR
--------------------------------------------------------------
  C H H H L H H L L   L L L L H L L L                  1
--------------------------------------------------------------
```

[†] PAL is a trademark of Monolithic Memories, Inc.

PAL Fuse Map

```
DESCRIPTION:

SHIVA DRAM CONTROLLER

                11 1111 1111 2222 2222 2233
   0123 4567 8901 2345 6789 0123 4567 8901

 0 ---- ---- ---- ---- ---- ---- ---- ----
 1 ---- -X-- ---- ---- -X-- -X-- ---- ---- /STROBE*/DS*/RW
 2 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
 3 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
 4 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
 5 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
 6 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
 7 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

 8 ---- ---- ---- ---- ---- ---- ---- ----
 9 ---- -X-- ---- ---- ---- X--- ---- ---- /DS*RW
10 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
11 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
12 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
13 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
14 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
15 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

16 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
17 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
18 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
19 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
20 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
21 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
22 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
23 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

24 ---- ---- -X-- ---- ---- ---- ---- ---- /RDY
25 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
26 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
27 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
28 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
29 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
30 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
31 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

32 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
33 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
34 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
35 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
36 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
37 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
38 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
39 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

40 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
41 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
42 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
43 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
44 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
45 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
46 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
47 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

48 ---- ---- ---- ---- ---- ---- ---- ----
49 -X-- -X-- ---- ---- ---- ---- ---- ---- /MSC*/DS
50 ---- -X-- -X-- ---- ---- ---- ---- ---- /RDY*/DS
51 ---- ---- ---- ---X ---- ---- ---- ---- Q
52 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
53 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
54 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
55 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

56 ---- ---- ---- ---- ---- ---- ---- ----
57 ---- ---- ---- X--- ---- ---- ---- ---- CLKOUT1
58 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
59 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
60 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
61 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
62 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
63 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

LEGEND:  X : FUSE NOT BLOWN (L,N,0)   - : FUSE BLOWN   (H,P,1)

NUMBER OF FUSES BLOWN =  340
```

PAL Fuse Map Summary

```
                 11 1111 1111 2222 2222 2233
   0123 4567 8901 2345 6789 0123 4567 8901

 0 ---- ---- ---- ---- ---- ---- ---- ----
 1 ---- -X-- ---- ---- -X-- -X-- ---- ---- /STROBE*/DS*/RW

 8 ---- ---- ---- ---- ---- ---- ---- ----
 9 ---- -X-- ---- ---- ---- X--- ---- ---- /DS*RW


24 ---- ---- -X-- ---- ---- ---- ---- ---- /RDY



48 ---- ---- ---- ---- ---- ---- ---- ----
49 -X-- -X-- ---- ---- ---- ---- ---- ---- /MSC*/DS
50 ---- -X-- -X-- ---- ---- ---- ---- ---- /RDY*/DS
51 ---- ---- ---- ---X ---- ---- ---- ---- Q

56 ---- ---- ---- ---- ---- ---- ---- ----
57 ---- ---- ---- X--- ---- ---- ---- ---- CLKOUT1

LEGEND:  X : FUSE NOT BLOWN (L,N,0)   - : FUSE BLOWN   (H,P,1)

NUMBER OF FUSES BLOWN =  340
```

PAL Object Code

```
B B B B B B B B B 3 B B B B B B B B B B B B B B B B B B B B B B .
3 3 3 3 3 0 3 3 3 3 3 3 3 3 3 3 3 2 3 3 1 2 3 3 3 3 3 3 3 3 3 3 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 .
C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C .
C 8 C C C 8 C C C C C C 4 C C C C C C C C C C C C C C C C C C C .
```