TUC52 Circuit Description

Issue 1 – January 16, 1998

Tucson Amateur Packet Radio
8987-309 E. Tanque Verde Rd., #337
Tucson, AZ 85749-9399
Phone: +1 940-383-0000
Fax: +1 940-566-2544
tapr@tapr.org

1. INTRODUCTION

The name TUC-52 (TAPR Universal Controller based on the 8052). Additionally, because was designed to work with Intel's MCS BASIC-52 (and derivatives) we used the designation "52" instead of "32" (32 implying external ROM, which is the basis for TUC-52). It's somewhat of a misnomer, but we will live with it.

The constraints are to have a board that is capable of operating with internal masked ROM or external EPROM, and that the memory map provide both Harvard (separate code and data) or Von Neuman (combined code and data) addressing modes. RAM sites are provided such that up to 64K of RAM can be addressed on the board and both RAM sites may be battery backed up. There are several pre-programmed memory maps that the user can choose by placing shunts on shorting blocks. One particular memory map will allow the operation of Intel's MCS BASIC-52, a stand alone "tiny basic" system that includes floating point calculations and a rudimentary file system for storing and retrieving basic programs as well as the ability to begin executing a user developed preprogrammed basic program at reset, without any operator intervention.

Plenty of bit I/O has been provided for so that bus signals do not need to leave the board. However provision is made for a limited bus to be conveyed to another close-by board.

2. CPU AND ASSOCIATED GLUE

IC1 is the CPU socket that holds most any member of the 8052 family that uses a 40 pin DIP package, including the Dallas DS80C320. IC1 typically is stuffed with an 80C32 (the ROM-less version of the 80C52). Y1 provides a standard clock frequency of 11.0592 MHz that allows for common baud rates on the serial UART port. C14, a variable capacitor, is provided for in the board layout but it's not expected that the component will be provided for the user.

H1 selects the source of the UART input. A shunt across pins 1 & 2 of H1 will cause the input for IC1 pin 10 to be taken from an on-board "poor man's" RS-232 receiver. A shunt across pins 2 & 3 of H1 will cause the input for IC1 pin 10 to be taken from an off-board source provided through the PORT BLOCK.

H2 is used to tell the CPU to execute code from it's internal masked program memory or to get code from an external EPROM. A shunt across pins 1 and 2 of H2 will cause code to be executed from internal masked ROM of IC2. A shunt across pins 2 and 3 of H2 will cause code to be executed from external memory.

R12 and R13 are used to pull up P17 and P16 respectively so that these I/O leads can be used to communicate with I2C (pronounced EYE SQUARED SEE) protocol components (i.e., IC15 and IC16).

Part of SIP resistor RS2 is used to pull up INT1*, INT0*, T1 and T0 to ensure that these signals can be brought out to the outside world in an RF environment and not be likely to be inadvertently pulled low. It's understood that these ports already have an internal weak pull-up but the additional current source provided by RS2 is a form of insurance.

IC2 provides for the logical low ORing of the RD- and PSEN- signals to form the buffered PSRD* signal. Additionally IC2 provides buffering for the PSEN*, WR* and RD* signals. R14 is included to ensure that the WR- lead is pulled high during the time when the power is on but the reset signal is active. During the reset period WR- is tri-stated and its state is unknown. If it goes low it will send a write signal to any enabled RAMs. The pull-up is added to ensure it doesn't wander low during resets.

IC3 provide an address latch to capture and buffer the lower order address signals from the data bus. SIP resistor RS1 is used to pull up the open drain data bus.

3. MEMORY SITES

TUC-52 provides for up to three memory sites.

IC4 is for an EPROM of either the 27c256 (32K) or 27c512 (64k) type. The amount of EPROM actually available is determined by the PAL address selector.

H3 selects the signal for pin 1 of the EPROM, IC4. A shunt across pins 1 & 2 of H3 will cause IC4 pin 1 to be at +5 volts. This is needed when a 27C256 is installed in this socket. A shunt across pins 2 & 3 of H3 will cause IC4 pin 1 to be connected to A15. This is needed when a 27C512 is installed in this socket.

H4 selects the signal for IC4 pin 22, the output enable for the EPROM. A shunt across pins 1 & 2 of H4 will cause IC4 pin 22 to be connected to PSEN*. When in this position the EPROM will only be in the code space and not the data space of the memory map. A shunt across pins 2 & 3 of H4 will cause IC4 pin 22 to be connected to PSRD*. When in this position the EPROM will be in both the code and data space of the memory map.

IC5 is for a 32K RAM chip. The amount of RAM actually available is determined by the PAL address selector. The RAM is connected to the +5BB net for power so that it can retain memory data (if the user has installed a battery on TUC-52). H5 selects the signal source for IC5 pin 22, the output enable for the RAM. A shunt across pins 1 and 2 of H5 will cause IC5 pin 22 to be connected to RD*. When in this position the RAM will only be in the data space and not the code space of the memory map. A shunt across pins 3 and 4 of H5 will cause IC5 pin 22 to be connected to PSRD*. When in this position the RAM will be in both the code and data space of the memory map. A shunt across pins 5 and 6 of H5 will

cause IC5 pin 22 to be connected to PSEN*. When in this position the RAM will be in only the code space and not the data space of the memory map.

IC7 provides control of the chip select input to the RAM. When the system is under normal power conditions PF* is high and switch IC7A is closed while PF is low and switch IC7B is open, thus connecting the RAM chip-select pin to YCE*, the chip-select generated by the PAL. When normal power has been removed, PF* is low and switch IC7A is open while PF is high and switch IC7B is closed, thus connecting the RAM chip-select pin to +5BB and disabling the chip. IC7 is powered by the +5BB net and not the +5 net.

IC6 is a dual mode site (socket) that can accommodate either a 32K RAM chip or a 27C256 EPROM chip. With standard PAL programming, this part is in the address space from 8000H to EFFFH.

H6 selects the signal for pin 1 of the socket for IC6. A shunt across pins 1 and 2 of H6 will cause IC6 pin 1 to be connected to A14. This is needed when a RAM chip is installed in this socket. A shunt across pins 2 and 3 of H6 will cause IC6 pin 1 to be connected to the +5 net. This is needed when a 27C256 is installed in this socket.

H7 selects the signal for pin 27 of the socket for IC6. A shunt across pins 1 and 2 of H7 will cause IC7 pin 27 to be connected to WR*. This is needed when a RAM chip is installed in this socket. A shunt across pins 2 and 3 of H7 will cause IC7 pin 27 to be connected to A14. This is needed when a 27C256 is installed in this socket.

H8 selects the signal for pin 22, the output enable for the chip installed at IC6. A shunt across pins 1 and 2 of H8 will cause IC6 pin 22 to be connected to PSRD*. When in this position the chip will be in both the code and data space of the memory map. A shunt across pins 2 and 3 of H8 will cause IC6 pin 22 to be connected to RD*. When in this position the chip will only be in the data space and not the code space of the memory map.

H9 selects the signal for pin 20, the chip-select for IC6. A shunt across pins 1 and 2 of H9 will cause IC6 pin 20 to be connected directly to the PAL chip-select ZCE*. This is used when an EPROM chip is installed in this socket. It may also be used for a RAM chip that is not battery backed up. A shunt across pins 2 and 3 of H9 will cause IC6 pin 20 to be connected to the analog switches IC7C and IC7D, for control of the chip-select. This may be used when a RAM chip is installed in this socket.

H10 selects the source for power for the device in the socket at IC6. A shunt across pins 1 and 2 of H10 will cause IC6 pin 28 to be connected to the +5 net. This is used when an EPROM chip is installed in this socket or when a RAM that is not to be battery backed up is to be used in this socket. A shunt across pins 2 and 3 of H10 will cause IC6 pin 28 to be connected to the +5BB net. This is used only when a RAM chip is installed in this socket and the RAM is to be battery backed up.

Note also that the position of shunts on H9 and H10 must track each other for proper operation. If a shunt is installed across pins 1 & 2 of H9 then a shunt must be installed across pins 1 & 2 of H10 and then either a EPROM or RAM may be used as IC6 (but the RAM will not be battery backed up). The converse is also true. If a shunt is installed across pins 2 & 3 of H9 then a shunt must be installed across pins 2 & 3 of H10 and then only a RAM may be used as IC6.

Many different memory installations are possible with TUC-52. At the bare minimum (and really never expected to operate this way) it's possible that TUC-52 could operate with only internal code and data within the 8032 (8052 for internal code), leaving IC4, IC5 and IC6 unpopulated. More likely is that IC4 will contain an EPROM with code and that the 256 bytes of RAM internal to an 8032 will be used for read/write storage. Most simple applications can run in this mode.

Moving up the scale of complexity, the next embellishment would be to have a RAM device at IC6. In this configuration IC4 could be mapped for code space from 0H to 7FFFH while IC6 could be mapped for code AND data from 8000H to EFFFH.

When TUC-52 is used for MCS BASIC-52 program development a RAM device would be added at IC5 which would be configured for data space from 0 to 7FFFH. In this case IC5 serves as working program memory while IC6 serves as the file system for storing programs and data. When TUC-52 is used as a stand-alone mountain-top MCS BASIC-52 program execution system, IC6 will hold a ROM that contains the user's BASIC source code program. A ROM is used in this case instead of BBRAM because it's more reliable.

4. PAL CHIP-SELECTS

A 16L8 PAL is used at IC11 to generate chip-selects for the different devices used on TUC-52. The actual memory mapping is determined by the programming of the PAL. The user selects a particular pre-programmed memory map by placing shunts on H13.

In general,

XCE* (active low) selects the EPROM at IC4,
YCE* (active low) selects the RAM at IC5 and
ZCE* (active low) selects the RAM/ROM at IC6.

IOA* is an I/O select line (active low) that's determined by address leads from the processor. It drives one half of IC9 for I/O device selection. IOB* is an I/O select line (active low) whose state is also determined by the address signals. Additionally, IOB* can also be configured to qualify its signal with RD* or WR* being active (low). This qualification may be needed for dumb ports (like 74hc374). The IOA* and IOB* signals are always inactive (high) when PSEN* is active (low).

TABLE 1 – MEMORY ADDRESS SELECTION

Shunts A and B determine main memory mapping

SHUNTS		MEMORY ADDRESS-WHEN-SIGNAL-IS-ACTIVE		
B	A	IC4 (XCE*)	IC5 (YCE*)	IC6 (ZCE*)
Open	Open	0000-7FFF	0000-7FFF	8000-EFFF
Open	Shunt	0000-1FFF	2000-7FFF	8000-EFFF
Shunt	Open	0000-3FFF	4000-7FFF	8000-EFFF
Shunt	Shunt	0000-FFFF	0000-7FFF	8000-EFFF

TABLE 2 – I/O ADDRESS SELECTION					
Shunt C determines I/O space mapping					
SHUNT	BUS-CNTL-SIGS			I/O ADDRESS-WHEN-ACTIVE	
C	PSEN*	RD*	WR*	IOA*	IOB*
Shunt	1	X	X	F800-FBFF	FC00-FFFF
Open	1	0	0	F800-FBFF	FC00-FFFF[1]
Open	1	0	1	F800-FBFF	FC00-FFFF
Open	1	1	0	F800-FBFF	FC00-FFFF
Open	1	1	1	F800-FBFF	no select

NOTE: Shunt D currently has no effect on the PAL.

5. I/O

There are two main types of I/O for TUC-52: processor I/O and memory mapped auxiliary I/O. Processor I/O is I/O provided by the processor IC itself while auxiliary I/O is that provided by memory mapped 82c55 parallel ports. First consider processor I/O.

There are two processor I/O ports available: P1 and P3 of 8 bits each. The P1 port usually provides general purpose bit I/O while P3 has special features associated with each lead. However, the special features of P3 can always be disabled and the ports used for simple bit input/output I/O. For advanced processors (like the Dallas DS80C320) even the P1 port has special features associated with it.

The features of P3 are always as follows (in addition to general purpose bit I/O):

P3.0	RXD	UART Data Input
P3.1	TXD	UART Data Output
P3.2	INT0*	Interrupt 0 Input
P3.3	INT1*	Interrupt 1 Input
P3.4	T0	Timer 0 Input
P3.5	T1	Timer 1 Input
P3.6	WR*	Bus Write (not available for bit I/O)
P3.7	RD*	Bus Read (not available for bit I/O)

For 80C31 devices port P1 provides only bit I/O, without special features. For 80C32 devices port P1 can provide the following special features (in addition to general purpose bit I/O).

P1.0	T2	Timer 2, External Count Input
P1.1	T2EX	Timer 2, Capture/Reload Trigger Input

Some Philips parts also allow each of the P1 ports to serve as an independent interrupt signal.

[1] RD* and WR* should never be low at same time -- this state is shown for completeness of PAL programming.

For the Dallas DS80C320 device, port P1 can also provide the following special features (in addition to general purpose bit I/O).

P1.0	T2	Timer 2, External Count Input
P1.1	T2EX	Timer 2, Capture/Reload Trigger Input
P1.2	RXDX	Second UART Data Input
P1.3	TXDX	Second UART Data Output
P1.4	INT2	Active high interrupt
P1.5	INT3*	Active low interrupt
P1.6	INT4	Active high interrupt
P1.7	INT5*	Active low interrupt

Auxiliary I/O is that provided by up to two 82c55 parallel ports. The first 82c55 is IC13 which has non-inverting high voltage open collector drivers connected to port A. These drivers can easily be omitted by simply replacing the driver IC with a "straight across" jumper block. Pull-up resistors, formed by RS3, are provided for port B. These too can be omitted for applications that don't need them (or don't want them). Ports A, B and C of IC13 are brought out to the PORT BLOCK for connection to another board for specific interfacing requirements.

The second 82c55 (IC15) ports are connected directly to the SEC BLOCK (secondary block) without any other embellishments.

6. CONNECTOR BLOCKS

There are three major connector blocks: the PORT BLOCK, SEC BLOCK and POWER BLOCK. There is one minor block: the AUX BLOCK. Only the PORT block is expected to have a connector installed on the standard TUC-52 board. All other blocks are expected to be unpopulated.

The PORT BLOCK is the block that's expected to be used in almost all applications. Additional blocks will usually be unnecessary. The PORT BLOCK conveys nets for P1 and P3 from the processor as well as the three ports from the primary 82c55 (A=P4, B=P5 and C=P6). Essential power and control signals also pass through the PORT BLOCK. The only area for concern is the voltage drop on the GND, +5 and +V nets through the PORT BLOCK interconnections. If this drop is considered to be excessive then the POWER BLOCK should be used as well.

The SEC BLOCK is not expected to be used often but it is available for those applications that need its circuits. The SEC BLOCK conveys nets for the three ports from the secondary 82c55 (A=P7, B=P8 and C=P9) as well as power and ground.

The POWER BLOCK is used to provide a low voltage drop connection between TUC-52, any interface board(s) and the external power source. At this point it's not clear if this block would always need to be used since its circuits are redundant with those provided by the PORT BLOCK. The only reason for the user to make use of the POWER BLOCK is if there is an excessive voltage drop in the +5 and/or ground nets through the PORT BLOCK.

The AUX BLOCK is not expected to be used often but it is available for those applications that need its circuits. The AUX BLOCK is a subset of the PORT BLOCK and serves as a low cost alternative for applications that don't need the 50 pin connector and cable of the PORT BLOCK.

7. POWER CONTROL

The power control system is the heart of the battery backup system for the RAM devices and the time of day clock (if provided). It also includes the system reset circuitry.

Unregulated power is applied to the +V net either from the interface board through the PORT BLOCK or through the POWER BLOCK. This voltage is regulated to +5 volts by VR1. Option block H18 is provided so that the regulator can be isolated from the +5 net for those cases when the +5 net is back-fed from the interface board.

As far as the BBRAM system goes, there are three states to be concerned with. They are (State-A) when the voltage on the +5 net is so low that there is no way the CPU is going to execute code (less than 2 volts), (State-B) when there is enough voltage on the +5 net that the CPU might actually run (more than 2 volts) but we don't want it to run, and (State-C) when there is enough voltage on the +5 net that the CPU should run (about 5 volts) and we want it to run. The goal of the power control circuitry are as follows: For State-A the system should hold the BBRAM disabled and not worry about the reset signal. For State-B the system should hold the BBRAM disabled and make reset active so that the CPU won't execute code. For State-C the system should enable the BBRAM for operation and, after a one second delay upon entering State-C, release the reset signal so that the CPU can execute code.

The voltage at the input to the regulator is sensed by D3 and Q1. In general, Q1 should be saturated only when there is sufficient voltage (about 8 volts) at the input of VR1 to ensure that the +5 net is stable at about +5 volts (that's State-C). When the voltage at +V is below about 8 volts (State-A or State-B), Q1 is cut off and the input to IC16A is high (IC16 is powered from the +5BB net, so it's "always powered up"). In this case IC16A's output is low and C7 is discharged very quickly through D4 and R6. In this state PF (power failure active high) is high and PF* (power failure active low) is low. These PF signals go to IC7, the 74HC4066 analog switch, to ensure that the RAM chips are not selected by the PAL's chip-select outputs. Additionally, when Q1 is cut off, the output of IC16E is low causing C8 to be discharged through D5 and R9 which causes the reset signal (RST) to go active (high) as long as power is available to IC8. IC8 is powered by the +5 net and can operate correctly with as little as 2 volts as a power source. So, when a voltage is applied to +V and the +5 net starts to rise, once it gets to 2 volts or so the RST signal is in the high (active state) and won't be released until the voltage at the input to VR1 is 8 volts or more (plus the time delays caused by R6/C7 and R9/C8). Note also that during power down conditions that the output of IC16E is low so there's no current drain caused by IC16E trying to drive an unpowered IC8C's input to a logic high. TUC-52's battery backup feature only works when a source of +V is available and that +V is used to derive the +5 power source. The location of the regulator doesn't matter (i.e., the regulator can be on the TUC-52 board or some other board, so long as the +V net drops below 7.5 volts before the +5 net drops below 4.5 volts). In those cases where TUC-52 is powered exclusively from a source of +5 VDC and no associated +V is available, the battery backed ram feature will not work reliably. In addition, to enable the RAM devices when only +5 VDC is available, a shunt must be placed on H19. Do NOT install a shunt on H19 when a source of +V is available.

Power for the +5BB net can come from either of two sources. One is the +5 net and the other is a 3 volt battery. Both sources are combined by low voltage drop diodes (D8 and D9). The +5BB net is filtered by a tantalum capacitor (tantalum for low leakage) as well as a few standard bypassing caps. Note that no battery is required to operate TUC-52, even though the RAMs and IC9, the time of day clock, are powered from the +5BB net. The +5BB net will be powered when the +5 net is powered, even if no battery is installed. The only down-side of not having a battery installed is that the RAM and clock will lose their data when the power is removed from the +V net.

There is a slight difference in the R/C/D timing network for IC16B and IC8C. In the case of IC16B the goal is to get charge off C5 as fast as possible and not to get the input of IC16B low as fast as possible. In the case of IC8C the goal is to get input of IC8C low as fast as possible while discharging C8 at a more

leisurely pace than that of C6. Since any low at the cathode of D5 will be sustained for tens of milliseconds there's no rush to get the charge off of C8 immediately.

In addition to having the power failure detection system generating reset signals for the CPU, resets can also be generated by a mechanical switch that the user controls (the big red button). This switch, if needed, would be connected to H21.

An additional reset signal can be developed from the received UART signal on detection of a long spacing or BREAK condition. When RXD goes low the output of IC8E goes high, charging C9 through R10. When it goes high enough IC8D's output will go low. If a shunt is installed at H20 this will cause a system reset. I've included this functionality because I've found it to be very useful for commercial systems. The "reset on BREAK" function is of great benefit when one computer controls another (or controls many devices networked together). Rather than having someone go to each unit and push the big red button, the central computer only needs to send a spacing condition for a few seconds and all networked controllers are reset.

8. CLOCK AND NON-VOLATILE MEMORY

IC9 provides for a simple time of day clock and 240 bytes of BBRAM while IC10 provides for data retention by an EEPROM. Both devices communicate with the CPU by means of the Philips I2C (pronounced EYE SQUARED SEE) protocol.

9. MINIMAL SERIAL I/O

IC8A and IC8B provide a "poor man's" RS-232 interface. It's not expected that this interface will take the place of a real RS-232 interface (which goes on the interface board). I decided not to put a MAX232 interface on the TUC-52 board because it would have required running the TTL and RS-232 versions of the control signals twice though the PORT BLOCK. It was much simpler to have the interface board contain the RS-232 interface (or RS-422 or RS-485) and send the control signals to whatever I/O port was desired.

The "poor" interface is included for simple systems that don't make use of an interface board. Not to advance this interface as a desirable alternative to a MAX232 interface, but I've used this "poor" interface on many systems over the years and have yet to find an RS-232 device (manufactured after 1980) that it won't work with over short runs (less than 20 feet).

———ooOoo———