THE TARBELL CASSETTE INTERFACE

Plugs directly into ALTAIR* 8800 or IMSAI 8080 Computer. Serial-Parallel and Parallel-Serial Conversion performed on board.

It's fast — Up to 540 bytes per second (2200 bits per inch) with high-quality cassette recorder. 187 bytes per second (800 bits per inch) suggested standard with medium quality ($39.95) recorder. 27 bytes per second (with modification) for Kansas City format. 1000 bytes per second @ 10 inches per second with PHI-DECK.

Encoding method — Phase encoded self-clocking method requires only one channel, and withstands large amounts of wow and flutter. Has been used in the computer industry for many years. Can also be used to generate and detect Kansas City format tapes with small modification.

Will work with most audio cassette recorders. Has been used with reel-reel tape recorders. Can easily be adapted to automatically (software) controlled digital cassette units. The J. C. Penny #6536 is presently being recommended ($39.95). Suggested tape is Scotch Low-Noise, High-Density audio tape.

The device code (address) to which the interface responds is selectable in increments of 4 by an on-board dip-switch.

Four extra status lines are available to read external conditions.

Four extra control lines are available, which may be used to drive relays for extra cassette units, start-stop control, or controlled cassette drives, such as the PHI-DECK.

Two spare IC slots are provided on the board to do your own thing.

Software provided includes input/output driver routines, bootstrap, read-only memory program, and a BASIC program for saving and loading data. A Processor-Technology Software Package #1, which is an editor and assembler combined, and which has been extensively modified for saving and loading files to and from cassette, is available separately.

The 37-page owner's manual includes assembly instructions and drawings, parts and pin-function lists, soldering, cleaning, and installation notes, operating instructions, initial adjustment procedures, notes about using the PHI-DECK, start-stop control, interrupts, and writing programs for the cassette interface, the software mentioned above, and information about all the integrated circuits on the board.

The parts provided include all resistors, capacitors, integrated circuits, cassette cables, ribbon cables, dip connector, low-noise cassette with test stream, and a high-quality double-sided board with plated-through holes and gold-plated edge-connector fingers.

The first deliveries were made in September, 1975. Delivery time is from one to three weeks after receiving order. If you are not completely satisfied with the operation of the cassette interface, for whatever reason, the unit may be returned for refund or free repair within 90 days after you purchased it.

Kit price: $120. Assembled & Checked out: $175. Manual (incl): $4. California residents please add 6% sales tax.

*ALTAIR is a trademark/tradename of MITS, Inc.

TARBELL ELECTRONICS — 20620 S. Leapwood Ave., Suite P
Carson, California 90746 — Phone (213) 538-4251

HISTØRY AND SALES PITCH

I HAVE BEEN USING AN INEXPENSIVE AUDIØ CASSETTE RECØRDER IN MY HØME-DESIGNED CØMPUTER SYSTEM SINCE 1972. I HAVE ØVER 600 FILES ØN CASSETTES, MØSTLY ABØUT 4 KBYTES EACH. MY ESTIMATE IS THAT THE ERRØR RATE IS LESS THAN 1 ERRØR IN 1,000,000 BITS. I SAY THIS BECAUSE I CAN USUALLY RECØRD 30 4 KBYTE FILES ØN ØNE SIDE ØF A C-60 CASSETTE WITHØUT ANY ERRØRS. THIS INTERFACE GAVE ME VERY GØØD SERVICE WHILE I WAS WRITING THE DISK ØPERATING SYSTEM FØR MY 500 KBYTE DISK. SINCE I STARTED USING MY DISK SYSTEM A FEW YEARS AGØ, THE CASSETTE HAS SERVED AS BACKUP STØRAGE - A RELIABLE PLACE TØ STØRE DATA AND PRØGRAMS AFTER THEY ARE DEBUGGED.

THE ENCØDING METHØD I USE IS VERY SIMPLE, AND HAS BEEN IN USE IN INDUSTRY FØR QUITE SØME TIME. PICTURE A SHIFT REGISTER WHICH IS LØADED WITH THE DATA TØ BE RECØRDED. THE REGISTER IS THEN CLØCKED WITH A SQUARE WAVE. THE ØUTPUT ØF THE SHIFT REGISTER IS EXCLUSIVE-ØRED WITH THE CLØCK, PRØDUCING THE BI-PHASE DATA. THIS DATA GØES DIRECTLY TØ THE CASSETTE RECØRDER'S INPUT. THE MØST DIFFICULT PART ØF THE PRØCESS IS RECØVERING THE DATA. MANY LØNG HØURS WERE SPENT STUDYING THIS PRØBLEM AND TRYING DIFFERENT METHØDS. IN THE ØRIGINAL INTERFACE THIS WAS ACCØMPLISHED WITH A 760 HIGH-SPEED CØMPARATØR, A 74121 NØN-RETRIGGERABLE ØNE-SHØT, AND A DØUBLE-GLITCH GENERATØR MADE WITH AN EXCLUSIVE-ØR GATE. SINCE THEN, THE 8T20 HAS BEEN DEVELØPED, WHICH CØMBINES THESE THREE FUNCTIØNS ØN A SINGLE CHIP, AND IS THE UNIT USED IN THE ALTAIR INTERFACE.

THE PRESENT DESIGN IS EVEN MØRE RELIABLE THAN THE PREVIØUS ØNE, AND IS CAPABLE ØF RECØRDING AND RECØVERING ERRØR-FREE DATA AT A RATE ØF 540 BYTES PER SECØND ØN A STANDARD AUDIØ CASSETTE RECØRDER. (YES, THAT IS ØVER 2200 BITS PER INCH!) I AM STILL, HØWEVER, ENCØURAGING USERS TØ EXCHANGE DATA RECØRDED AT 187 BYTES PER SECØND (1500 BITS PER SECØND, 800 BITS PER INCH). THE MAIN ADVANTAGE ØF THIS METHØD ØVER ØTHERS IS IT'S ABILITY TØ WITHSTAND A LARGE AMØUNT ØF WØW AND FLUTTER, WHICH MAY BE INTRØDUCED BY CHEAP RECØRDERS, AND STILL RECØVER THE DATA RELIABLY. THIS FEATURE STEMS FRØM THE SELF-CLØCKING NATURE ØF THE RECØRDED SIGNAL: THE RECØVERED CLØCK VARIES RIGHT ALØNG WITH THE DATA, SØ THAT THE SPEED VARIATIØNS ARE ESSENTIALLY IGNØRED. THE MAIN DISADVANTAGE ØF THIS METHØD IS THAT IT REQUIRES GØØD LØW-NØISE TAPE, AND A DECENT FREQUENCY RESPØNSE ØN THE CASSETTE UNIT. THE CASSETTE UNIT I'VE BEEN USING LATELY (J.C. PENNY #6536) HAS A FREQUENCY RESPØNSE ØF 80-8,000 HZ. THE MØST IMPØRTANT PART IS THE HIGH END. THESE REQUIREMENTS ARE DUE TØ THE HIGH SPEED ØF THE INTERFACE, AND WØULD BE THE SAME FØR ANY HIGH SPEED DEVICE.

THE SPEED MAY NØT SEEM VERY IMPØRTANT TØ YØU NØW. BUT A GØØD PØRTIØN ØF YØUR TIME IS GØING TØ BE SPENT SAVING AND LØADING DATA, PRØGRAMS, AND ØTHER TEXT. THERE IS A WØRLD ØF DIFFERENCE BETWEEN LØADING BASIC AT SAY, 30 BYTES/SEC (4 MINUTES), AND AT 187 BYTES/SEC (40 SEC). IT DØESN'T SEEM LIKE MUCH, BUT WHEN YØU HAVE TØ DØ IT ØVER, AND ØVER, AND ØVER.... IT GETS TØ BE A BIT MUCH. ESPECIALLY WHEN YØU'RE DEVELØPING YØUR ØWN PRØGRAMS, AND THEY TEND TØ RUN AMUCK AND WIPE ØUT CØRE. THINK ABØUT IT, THEN BUY THE TARBELL CASSETTE INTERFACE. THE ØNLY METHØD PRØVEN WITH TIME. ASK YØUR FRIEND WHØ HAS ØNE.

SELECTING A CASSETTE UNIT FOR DIGITAL RECORDING

FIRST OF ALL, THE MOST EXPENSIVE CASSETTE RECORDERS ARE NOT NECESSARILY THE BEST FOR RECORDING DIGITAL DATA. THERE ARE SEVERAL FACTORS THAT COMBINE TO MAKE A GOOD UNIT FOR THE HOBBYIST:

1. IT SHOULD HAVE A GOOD HIGH-FREQUENCY RESPONSE, PREFERABLY UP TO AT LEAST 8,000 HZ.

2. IT SHOULD HAVE A TONE CONTROL, SO THAT THE INHERENT FREQUENCY RESPONSE MAY BE REALIZED.

3. ALTHOUGH AUTOMATIC VOLUME CONTROL IS MORE CONVENIENT FROM AN OPERATIONAL POINT OF VIEW, IT ALSO REQUIRES A FEW SECONDS OF SETTLING TIME BEFORE STARTING TO RECORD.

4. IF IT DOES NOT HAVE AUTOMATIC VOLUME CONTROL, IT IS GOOD TO HAVE A RECORDING LEVEL METER. THIS ALLOWS EASIER ADJUSTMENT FOR THE CORRECT RECORDING LEVEL.

5. IT IS VERY IMPORTANT TO HAVE A DIGITAL COUNTER. THIS MAKES IT POSSIBLE TO QUICKLY LOCATE THE DESIRED PROGRAM AMONG SEVERAL.

6. IT SHOULD BE CAPABLE OF RUNNING DIRECTLY ON THE AC LINE. BATTERIES TEND TO MAKE THE MOTOR GET SLOWER AS THEY WEAR.

7. IT IS HANDY TO HAVE AN AUXILLIARY INPUT, SO THAT A FAIRLY HIGH LEVEL MAY BE FED TO THE RECORDER, AND NOISE KEPT TO A MINIMUM.

8. A REMOTE INPUT JACK IS VALUABLE TO CONTROL START-STOP DURING ASSEMBLER AND COMPILER OPERATIONS.

9. JACKS FOR MIC, AUX, REMOTE, AND EARPHONE ARE USUALLY INCLUDED, BUT YOU SHOULD CHECK TO MAKE SURE THEY ARE THERE, ANYWAY.

10. LOW WOW AND FLUTTER CHARACTERISTICS ARE IMPORTANT, BUT ARE MUCH MORE IMPORTANT WHEN USING AN ASYNCRONOUS INTERFACE, WHICH IS NOT SELF-CLOCKING.

11. IF IT IS DESIRED TO DO AUTOMATIC REWIND, FAST-FORWARD, AND RECORD/PLAYBACK SWITCHING UNDER PROGRAM CONTROL, YOU MUST PURCHASE A RECORDER THAT HAS THESE FACILITIES. ANOTHER FEATURE TO LOOK FOR ON THIS TYPE OF UNIT IS A WAY TO KEEP TRACK OF WHERE YOU ARE ON THE CASSETTE TAPE.

12. PEOPLE HAVE EXPERIENCED PROBLEMS WITH PANASONIC RECORDERS. I AM PRESENTLY RECOMMENDING THE J.C. PENNEY MODEL 6536 AT $39.95 AS THE BEST BUY I KNOW, AND SEVERAL PEOPLE ARE USING THEM WITH MY INTERFACE WITH GOOD RESULTS. OTHER BRANDS THAT I KNOW HAVE BEEN SUCCESSFUL ARE SONY AND REALISTIC.

OF COURSE, NONE OF THE ABOVE ITEMS IS ABSOLUTELY NECESSARY FOR RECORDING DIGITAL DATA ON AN AUDIO CASSETTE. BUT THE MORE OF THESE REQUIREMENTS THAT ARE FILLED, THE MORE CAPABLE YOUR UNIT WILL BE, AND THE EASIER IT WILL BE TO USE.

IDEAS FOR USING THE CASSETTE INTERFACE

SAVING AND LOADING PROGRAMS

PROGRAMS MAY BE TOGGLED INTO MEMORY, LOADED FROM PAPER TAPE, OR LOADED BY SOME OTHER MEANS. THEY MAY THEN BE DUMPED ONTO CASSETTE USING THE CASSETTE OUTPUT ROUTINE. THE OUTPUT ROUTINE ITSELF MAY BE DUMPED ALONG WITH THE OTHER PROGRAM, SO THAT IT WILL BE AVAILABLE LATER FOR FURTHER DUMPS. ROUTINES MAY THEN BE LOADED FROM CASSETTE, MODIFIED, AND DUMPED BACK OUT TO CASSETTE IN A CONTINUOUS PROCESS OF DEVELOPMENT.

USING A BACKUP

A BACKUP IS A METHOD OF MAKING SURE THAT VALUABLE PROGRAMS OR DATA IS NOT LOST. ONE SIMPLE WAY OF PROVIDING A BACKUP IS TO RECORD A PARTICULAR PROGRAM IN TWO DIFFERENT PLACES ON THE SAME TAPE, OR ON TWO DIFFERENT TAPES. YOU MAY WANT TO GO BACK AND FORTH BETWEEN THE TWO COPIES, EACH TIME THE PROGRAM IS CHANGED. IN THIS WAY, YOU ALWAYS HAVE A COPY OF THE LAST PROGRAM, AND ONLY THE MOST RECENT CHANGES ARE LOST IF A POWER FAILURE OR OTHER EQUIPMENT TROUBLE DEVELOPS DURING THE PROCESS OF SAVING.

STARTING AND STOPPING AUTOMATICALLY

MOST AUDIO CASSETTE RECORDERS HAVE AN INPUT LABELED "REMOTE". THIS IS NORMALLY OPERATED FROM A SWITCH ON THE MICROPHONE, SO THAT THE RECORDER CAN BE STARTED AND STOPPED WHILE DICTATING. THIS INPUT CAN BE USED TO START AND STOP THE RECORDER UNDER PROGRAM CONTROL FROM THE COMPUTER. ONE OF THE 4 EXTRA CONTROL LINES COMING FROM THE CASSETTE INTERFACE (DIP-SOCKET PINS 5, 6, 7, OR 8) CAN BE USED TO DRIVE A RELAY WHICH WOULD HAVE ITS CONTACTS CONNECTED TO THE RECORDER REMOTE JACK. THE MAIN REQUIREMENTS ARE THAT THE 40 MA AVAILABLE FROM THE CONTROL LINE BE ABLE TO DRIVE THE RELAY, AND THAT THE RELAY CONTACTS ARE ABLE TO HANDLE THE CURRENT INTO THE REMOTE JACK, WHICH CAN BE AS HIGH AS ONE AMPERE.

OPERATING WITH MORE THAN ONE CASSETTE RECORDER

SOMETIMES IT IS DESIREABLE TO OPERATE WITH TWO OR MORE CASSETTE UNITS. WITH THE TARBELL CASSETTE INTERFACE, IT IS ALREADY POSSIBLE TO READ FROM ONE CASSETTE RECORDER, WHILE WRITING ONTO ANOTHER. THIS IS BECAUSE THE INPUT AND OUTPUT SECTIONS ARE ENTIRELY INDEPENDENT, AND MAY BE PROGRAMMED SEPERATELY. IF IT IS DESIRED TO READ FROM ONE OF TWO UNITS, AND WRITE ONTO ANOTHER, SUCH AS DURING A MERGING OPERATION, RELAYS MAY BE USED TO SWITCH BACK AND FORTH BETWEEN THE TWO INPUT UNITS, UNDER CONTROL OF ONE OF THE 4 CONTROL LINES. IF, HOWEVER, IT IS NECESSARY TO READ SIMULTANEOUSLY FROM TWO DIFFERENT UNITS, OR WRITE DIFFERENT INFORMATION ONTO TWO UNITS AT THE SAME TIME, IT WILL BE NECESSARY TO HAVE TWO CASSETTE INTERFACES.

TARBELL ELECTRONICS
20620 S. Leapwood Ave., Suite P
Carson, California 90746
(213) 538-4251

RANDY MILLER'S CHESS PROGRAM

THIS PROGRAM RUNS UNDER ALTAIR 8K BASIC 3.2, AND WILL PLAY A GAME OF CHESS AGAINST THE USER. IT REQUIRES A MEMORY OF ABOUT 20 KBYTES TOTAL FOR THE PROGRAM, THE BASIC INTERPRETER, AND THE VARIABLES. YOUR VERSION OF 8K BASIC 3.2 SHOULD BE MODIFIED TO DO A CLOAD FROM THE TARBELL CASSETTE INTERFACE.

MOVE THE TAPE TO THE MIDDLE OF THE LEADER (HIGH-PITCHED SOUND), THEN TYPE IN CLOAD C , THEN START THE TAPE PLAYING, THEN HIT RETURN. THE CONSOLE WILL COME BACK WITH "OK" WHEN THE COMPUTER HAS LOADED THE CHESS PROGRAM. THEN TYPE RUN .

WHEN THE CHESS PROGRAM ASKS FOR YOUR MOVE, TYPE TWO DIGITS FROM ONE TO EIGHT, THEN A COMMA, THEN TWO MORE DIGITS. THIS TELLS THE PROGRAM WHERE THE PIECE IS THAT YOU WANT TO MOVE, AND WHERE YOU WANT TO MOVE IT TO. THE CHESS BOARD IS SET UP WITH THE SQUARES NUMBERED AS SHOWN IN FIGURE A.

FIGURE A

	1	2	3	4	5	6	7	8
1	11	21	31	41	51	61	71	81
2	12	22	32	42	52	62	72	82
3	13	23	33	43	53	63	73	83
4	14	24	34	44	54	64	74	84
5	15	25	35	45	55	65	75	85
6	16	26	36	46	56	66	76	86
7	17	27	37	47	57	67	77	87
8	18	28	38	48	58	68	78	88

FIGURE B

1 = PAWN
2 = KNIGHT
3 = BISHOP
4 = ROOK
5 = QUEEN
6 = KING

SO, FOR EXAMPLE, TO MOVE YOUR PAWN FROM SQUARE 17 TO SQUARE 15 YOU WOULD TYPE 17,15 WHEN IT ASKS FOR YOUR MOVE. IF YOU WOULD LIKE TO SEE THE POSITION OF ALL THE PIECES ON THE BOARD, YOU MAY TYPE 0,0 WHEN IT IS YOUR MOVE. THE PIECES ARE IDENTIFIED BY THE NUMBERS 1 THROUGH 6 AS SHOWN IN FIGURE B.

TO HAVE THE COMPUTER GIVE A RUNNING ACCOUNT OF THE MOVES IT IS CONSIDERING AND THE EVALUATION EACH MOVE RECEIVES, TYPE IN THE FOLLOWING LINES:

```
681 PRINT PS(D,1);PS(D,2);
722 PRINT EV
```

DON'T EXPECT THE PROGRAM TO COME BACK WITH IT'S MOVE REAL QUICK. IT GENERALLY TAKES A FEW MINUTES TO MAKE UP IT'S MIND. THIS SHOULDN'T BOTHER YOU, SINCE MOST HUMAN PLAYERS TAKE AS LONG. DON'T BE SURPRISED IF YOU BEAT THE COMPUTER. SINCE THE PROGRAM IS FAIRLY SMALL, COMPARED TO THE BIG CHESS-PLAYING PROGRAMS, IT ISN'T HARD TO BEAT. IN FACT, I'VE BEAT IT 3 TIMES OUT OF 3, AND I DON'T CONSIDER MYSELF A VERY GOOD PLAYER.

TARBELL ELECTRONICS DOES NOT PROVIDE SUPPORT FOR THIS PROGRAM. ALL WE CLAIM IS THAT IT DOES PLAY CHESS, AND THAT WE WILL PROBABLY UPDATE FROM TIME TO TIME AS PEOPLE FIND BUGS OR MAKE IMPROVEMENTS. WE WILL NOT AUTOMATICALLY SEND OUT THESE IMPROVEMENTS, AND IF A NEW TAPE OR DOCUMENTATION IS REQUIRED, THE CUSTOMER IS EXPECTED TO PAY THE USUAL PRICE OF \$6. CALIFORNIA RESIDENTS PLEASE ADD 6%.

PROCESSOR TECHNOLOGY SOFTWARE PACKAGE #1

Modified for Low Memory and Cassette by Fred Clark,
Modified for Tarbell Cassette by Don Tarbell.

The main purpose of this software is to allow development of assembly-language programs. Programs may be entered from the keyboard in standard assembly-language format, edited, assembled, and saved on cassette in either source or machine format. These programs may be then loaded back into memory at some future time for listing, editing or running. Following is a summary of the commands. All address and byte information is in hex.

FILE /NAME/ AAAA	-Creates a file of name NAME at address AAAA (hex). Up to six text files can be in memory at the same time. The file just created is the current file.
FILE /NAME/ 0	-This command deletes the file named NAME.
FILE /NAME/	-Makes the named file the current file.
FILE	-Prints the starting and ending addresses of the current file, and it's name.
FILES	-Lists the name, starting & ending address for each of the files in the system. Current file at top.
DUMP SSSS EEEE	-Dumps the contants of memory from SSSS to EEEE.
EXEC AAAA	-Executes the machine-language program at address AAAA.
EXEC /NAME/	-Searches the cassette for named file, loads and exec.
ENTR AAAA	-Enter data into memory: B1 B2 B3 B4/
LIST NNNN	-Lists the current file starting at line NNNN.
DELT NNNN MMMM	-Deletes lines NNNN through MMMM from current file.
NNNN (text entered)	-Enter a line of text into the current file. Always use 4 decimal digits. It works similar to BASIC, in that the numbered lines are ordered automatically.
ASSM AAAA	-Assembles current file into address AAAA.
ASSM AAAA BBBB	-Assembles at address AAAA, but puts code at BBBB.
CUST	-Execute a customer (user) routine at address E000.
SFIL	-Save the current file on cassette.
LFIL	-Load the current file from cassette. The name of the file on tape must match the name of the current file. Be sure there is enough room in memory.
CFIL	-Check a cassette file written with the SFIL command above, for errors, without overwriting current file.
AFIL	-Append a file from cassette to the current file.
SAVE SSSS EEEE	-Save a block of memory from SSSS to EEEE on cassette.
LOAD SSSS	-Load a block of memory from cassette starting at SSSS.
CHEK SSSS	-Check a cassette file written with SAVE command.
NAME /NAME/	-Rename the current file to NAME.
RNUM N	-Renumber the current file by increment N.
NLIS NNNN	-Lists the current file at NNNN without line numbers.

This software is useful, but may not be completely free of bugs. It loads into the first 4096 bytes of memory, and uses about 2048 bytes following, for tables and scratch pad. Ctl-C escapes from any printing. Ctl-W freezes printing until another character is typed. The package is available from Tarbell Electronics for the prices listed below:

Cassette Tape with software and instructions: $5.00
A reassembled and patched (updated) listing: $5.00

TARBELL ELECTRONICS
20620 S. Leapwood Ave., Suite P
Carson, California 90746
(213) 538-4251

TARBELL ELECTRONICS
20620 S. Leapwood Ave., Suite P
Carson, California 90746
(213) 538-4251

INSTRUCTIONS FOR CASSETTE READ-ONLY MEMORY PROGRAM 1 (ROMP1)
AUG 12, 1976

THIS PROGRAM, WHICH WILL RUN IN EITHER READ-ONLY OR READ-WRITE MEMORY, ALLOWS THE USER TO PERFORM THE 6 FUNCTIONS LISTED BELOW:

LETTER	EXAMPLE	DESCRIPTION
E	E3000	EXECUTE PROGRAM AT 3000 (HEX).
C	C100020E7	CHECK CHECKSUM OF CASSETTE RECORD.
S	S	GENERATE SYNC STREAM.
O	O20000000	OUTPUT RECORD OF LENGTH 2000, STARTING AT 0000.
I	I10003000	INPUT RECORD OF LENGTH 1000, STARTING AT 3000.
L	L47DF1257	LOAD AND EXECUTE, LENGTH 47DF, STARTING AT 1257.

HAVE THE CASSETTE RECORDER RUNNING BEFORE TYPING THE LAST CHARACTER OF THE C, O, I, OR L COMMANDS, SINCE THE LAST CHARACTER IS WHAT STARTS THE TRANSFER GOING. IF AN E IS PRINTED AFTER THE TRANSFER IS MADE ON A C, I, OR L COMMAND, THIS INDICATES AN ERROR HAS BEEN DETECTED. OTHERWISE, THE C, OR I WILL BE PRINTED AFTER THE TRANSFER. IN THE CASE OF THE L COMMAND, IF THERE ARE NO ERRORS, THE PROGRAM WILL AUTOMATICALLY START EXECUTING AT THE STARTING ADDRESS SPECIFIED IN THE COMMAND.

THE SECOND AND THIRD BYTES OF THIS PROGRAM SPECIFY WHERE THE STACK IS LOCATED, AND SHOULD BE THE TOP OF SOME RANDOM-ACCESS MEMORY (NOT READ-ONLY MEMORY) WHICH IS NOT USED FOR OTHER THINGS. THE STACK WILL THEN EXPAND DOWN FOR ABOUT 8 OR 10 BYTES FROM THIS ADDRESS.

```
ASSM FF00 E400

FF00                    0005 *
FF00                    0010 *** READ-ØNLY MEMØRY PRØGRAM ***
FF00                    0015 *
FF00 31 00 EC           0020 START LXI  SP,0EC00H   SET STACK PTR.
FF03 3E 0D              0034       MVI  A,0DH       PRINT CR, LF.
FF05 CD 58 FF           0036       CALL PTCN
FF08 3E 0A              0038       MVI  A,0AH
FF0A CD 58 FF           0040       CALL PTCN        ØN CØNSØLE.
FF0D 3E 3F              0042       MVI  A,'?'       PRINT "?"
FF0F CD 58 FF           0043       CALL PTCN        ØN CØNSØLE.
FF12 CD 64 FF           0050       CALL RDCN        READ KEYBØARD.
FF15 FE 45              0060       CPI  'E'        IF E,
FF17 CC 36 FF           0070       CZ   EXEC    EXECUTE A PRØGRAM.
FF1A FE 43              0080       CPI  'C'        IF C,
FF1C CC B0 FF           0090       CZ   CINR     GØ TØ INPUT RØUTINE.
FF1F FE 53              0100       CPI  'S'        IF S,
FF21 CC 73 FF           0110       CZ   CSYNC   GØ TØ SYNC GEN.
FF24 FE 4F              0120       CPI  'Ø'        IF Ø,
FF26 CA 7B FF           0130       JZ   CØUTR       GØ TØ CASS ØUT.
FF29 FE 49              0140       CPI  'I'        IF I,
FF2B CC B0 FF           0150       CZ   CINR    GØ TØ CASS IN.
FF2E FE 4C              0160       CPI  'L'        IF L,
FF30 CC B0 FF           0170       CZ   CINR    DØ A LØAD & GØ.
FF33 C3 00 FF           0200       JMP  START       START ØVER.
FF36                    6000 *
FF36                    6010 *** EXECUTE THE PRØGRAM AT THE ADDRESS ***
FF36                    6020 *
FF36 CD 3A FF           6030 EXEC  CALL AHEX      READ ADDRESS FRØM KB.
FF39 E9                 6040       PCHL           JUMP TØ IT.
FF3A                    7000 *
FF3A                    7002 *** CØNVERT UP TØ 4 HEX DIGITS TØ BINARY *
FF3A                    7004 *
FF3A 21 00 00           7005 AHEX  LXI  H,0     GET A 16-BIT ZERØ.
FF3D 0E 04              7008       MVI  C,4     CØUNT ØF 4 DIGITS.
FF3F CD 64 FF           7009 AHE1  CALL RDCN    READ A BYTE.
FF42 29                 7010       DAD  H       SHIFT 4 LEFT.
FF43 29                 7012       DAD  H
FF44 29                 7014       DAD  H
FF45 29                 7016       DAD  H
FF46 CD 50 FF           7018       CALL AHS1    CØNVERT TØ BINARY.
FF49 85                 7020       ADD  L
FF4A 6F                 7022       MØV  L,A
FF4B 0D                 7024       DCR  C       4 DIGITS?
FF4C C2 3F FF           7026       JNZ  AHE1    KEEP READING.
FF4F C9                 7028       RET          RETURN FRØM AHEX.
FF50                    7100 *
FF50                    7102 *** CØNVERT AN ASCII DIGIT TØ HEX ***
FF50                    7104 *
FF50 D6 30              7106 AHS1  SUI  48      ASCII BIAS.
FF52 FE 0A              7108       CPI  10      DIGIT 0-10
FF54 D8                 7110       RC
FF55 D6 07              7112       SUI  7       ALPHA BIAS.
FF57 C9                 7114       RET          RETURN FRØM AHS1.
FF58                    8000 *
FF58                    8002 *** PRINT REGISTER A ØN CØNSØLE ***
```

```
FF58                    8004 *
FF58 F5                 8006 PTCN  PUSH PSW        SAVE REG A.
FF59 DB 00              8008 PTLØP IN   CØNC       READ PRINTER STATUS
FF5B E6 80              8010       ANI  80H        IF BIT 7 NØT 0,
FF5D C2 59 FF           8012       JNZ  PTLØP      WAIT TILL IT IS.
FF60 F1                 8014       PØP  PSW        THEN RECØVER A,
FF61 D3 01              8016       ØUT  CØND       AND PRINT IT.
FF63 C9                 8018       RET             RETURN FRØM PTCN.
FF64                    8100 *
FF64                    8102 *** READ FRØM CØNSØLE TØ REGISTER A ***
FF64                    8104 *
FF64 DB 00              8106 RDCN  IN   CØNC       READ KEYBØARD STATS
FF66 E6 01              8108       ANI  1          IF BIT 1 NØT 0,
FF68 C2 64 FF           8110       JNZ  RDCN       REPEAT UNTIL IT IS.
FF6B DB 01              8112       IN   CØND       READ FRØM KEYBØARD.
FF6D E6 7F              8114       ANI  7FH        STRIP ØFF MSB.
FF6F CD 58 FF           8115       CALL PTCN       ECHØ ØNTØ PRINTER.
FF72 C9                 8116       RET             RETURN FRØM RDCN.
FF73                    8118 CØNC  EQU  0          CØNSØLE STATUS PØRT
FF73                    8120 CØND  EQU  1          CØNSØLE DATA PØRT.
FF73                    9100 *
FF73                    9102 *** SYNC CØDE GENERATØR PRØGRAM ***
FF73                    9103 *
FF73 3E E6              9104 CSYNC MVI  A,0E6H     WRITE SYNC BYTE
FF75 CD A4 FF           9106       CALL CØUT       ØNTØ CASSETTE.
FF78 C3 73 FF           9108       JMP  CSYNC      KEEP DØING IT.
FF7B                    9200 *
FF7B                    9202 *** CASSETTE INTERFACE ØUTPUT RØUTINE ***
FF7B                    9204 *
FF7B CD 3A FF           9206 CØUTR CALL AHEX   READ BLØCK LENGTH.
FF7E EB                 9210       XCHG        PUT INTØ D,E.
FF7F CD 3A FF           9212       CALL AHEX   READ STARTING ADR.
FF82 06 00              9216       MVI  B,0    START CHECKSUM = 0.
FF84 CD A4 FF           9220       CALL CØUT       START BYTE ØUT.
FF87 3E E6              9222       MVI  A,0E6H     SEND SYNC BYTE
FF89 CD A4 FF           9224       CALL CØUT       TØ CASSETTE.
FF8C 7E                 9226 CØLØP MØV  A,M        GET DATA FRØM MEMØR
FF8D CD A4 FF           9228       CALL CØUT       SEND TØ CASSETTE.
FF90 80                 9230       ADD  B      ADD TØ CHECKSUM.
FF91 47                 9232       MØV  B,A
FF92 23                 9234       INX  H          INCREMENT PØINTER.
FF93 1B                 9236       DCX  D      DECREMENT CØUNTER.
FF94 97                 9238       SUB  A          CLEAR A.
FF95 BA                 9240       CMP  D      IF D NØT = 0,
FF96 C2 8C FF           9242       JNZ  CØLØP       REPEAT LØØP.
FF99 BB                 9244       CMP  E          IF E NØT = 0,
FF9A C2 8C FF           9246       JNZ  CØLØP       REPEAT LØØP.
FF9D 78                 9248       MØV  A,B        GET CHECKSUM.
FF9E CD A4 FF           9250       CALL CØUT       ØUTPUT IT.
FFA1 C3 00 FF           9252       JMP  START      GET ANØTHER CØMMAND
FFA4 F5                 9254 CØUT  PUSH PSW        SAVE A AND FLAGS.
FFA5 DB 6E              9256 CLØP  IN   CASC       READ CASSETTE STATS
FFA7 E6 20              9258       ANI  20H        LØØK AT BIT 5.
FFA9 C2 A5 FF           9260       JNZ  CLØP       TRY AGAIN?
FFAC F1                 9262       PØP  PSW        RESTØRE A.
FFAD D3 6F              9263       ØUT  CASD   SEND DATA TØ CASSETTE.
FFAF C9                 9264       RET             RETURN FRØM CØUT.
FFB0                    9300 *
FFB0                    9302 *** CASSETTE INPUT RØUTINE ***
FFB0                    9304 *
FFB0 F5                 9310 CINR  PUSH PSW    SAVE CØNTRØL CHAR.
FFB1 3E 10              9320       MVI  A,10H  USE BIT 4 IN REG A
FFB3 D3 6E              9330       ØUT  CASC   TØ RESET CASS INT.
FFB5 CD 3A FF           9340       CALL AHEX   READ BLØCK LENGTH.
FFB8 EB                 9350       XCHG        PUT INTØ D,E.
FFB9 CD 3A FF           9360       CALL AHEX   READ STARTING ADDRESS.
```

```
FFBC F1           9365       PØP  PSW    GET CØNTRØL CHAR.
FFBD E5           9370       PUSH H      SAVE STARTING ADDRESS
FFBE F5           9375       PUSH PSW    UNDER CØNTRØL CHAR.
FFBF 06 00        9380       MVI  B,0    SET CHECKSUM = 0.
FFC1 CD F2 FF     9390 CILØP CALL CIN    READ A BYTE FRØM CASS.
FFC4 4F           9400       MØV  C,A    SAVE IT IN REG C.
FFC5 F1           9410       PØP  PSW    GET CØNTRØL CHAR.
FFC6 F5           9420       PUSH PSW    SAVE IT BACK.
FFC7 FE 43        9430       CPI  'C'    IS IT A C?
FFC9 79           9440       MØV  A,C    GET BACK DATA BYTE.
FFCA CA CE FF     9450       JZ   CINØ   IF C, DØN'T STØRE IT.
FFCD 77           9460       MØV  M,A    IF NØT, DØ STØRE IT.
FFCE 80           9470 CINØ  ADD  B      ADD TØ CHECKSUM.
FFCF 47           9480       MØV  B,A
FFD0 23           9490       INX  H      INCREMENT PØINTER.
FFD1 1B           9500       DCX  D      DECREMENT CØUNTER.
FFD2 97           9510       SUB  A      CLEAR A.
FFD3 BA           9520       CMP  D      IF D NØT = 0,
FFD4 C2 C1 FF     9530       JNZ  CILØP  READ MØRE.
FFD7 BB           9540       CMP  E      IF E NØT = 0,
FFD8 C2 C1 FF     9550       JNZ  CILØP  READ MØRE.
FFDB CD F2 FF     9560       CALL CIN    READ LAST BYTE.
FFDE B8           9570       CMP  B      CØMPARE TØ CHECKSUM.
FFDF 3E 45        9580       MVI  A,'E'  PRINT E FØR ERRØR.
FFE1 C2 EC FF     9590       JNZ  CERR   PRINT NØW IF ERRØR.
FFE4 F1           9600       PØP  PSW    RECØVER CTL CHAR.
FFE5 FE 4C        9610       CPI  'L'    IF IT'S NØT L,
FFE7 C2 EC FF     9620       JNZ  CERR   DØN'T EXECUTE.
FFEA E1           9630       PØP  H      ØTHERWISE, EXECUTE
FFEB E9           9640       PCHL        AT STARTING ADDRESS.
FFEC CD 58 FF     9660 CERR  CALL PTCN   PRINT C,E, ØR I.
FFEF C3 00 FF     9690       JMP  START
FFF2 DB 6E        9700 CIN   IN   CASC   READ STATUS.
FFF4 E6 10        9710       ANI  10H    LØØK AT BIT 4.
FFF6 C2 F2 FF     9720       JNZ  CIN    WAIT UNTIL LØW.
FFF9 DB 6F        9730       IN   CASD   READ DATA FRØM CASS.
FFFB C9           9740       RET         RETURN FRØM CIN.
FFFC              9996 CASD  EQU  6FH    CASSETTE DATA PØRT.
FFFC              9997 CASC  EQU  6EH    CASSETTE STATUS PØRT.
FFFC              9998 PSW   EQU  6
FFFC              9999 SP    EQU  6
?
```

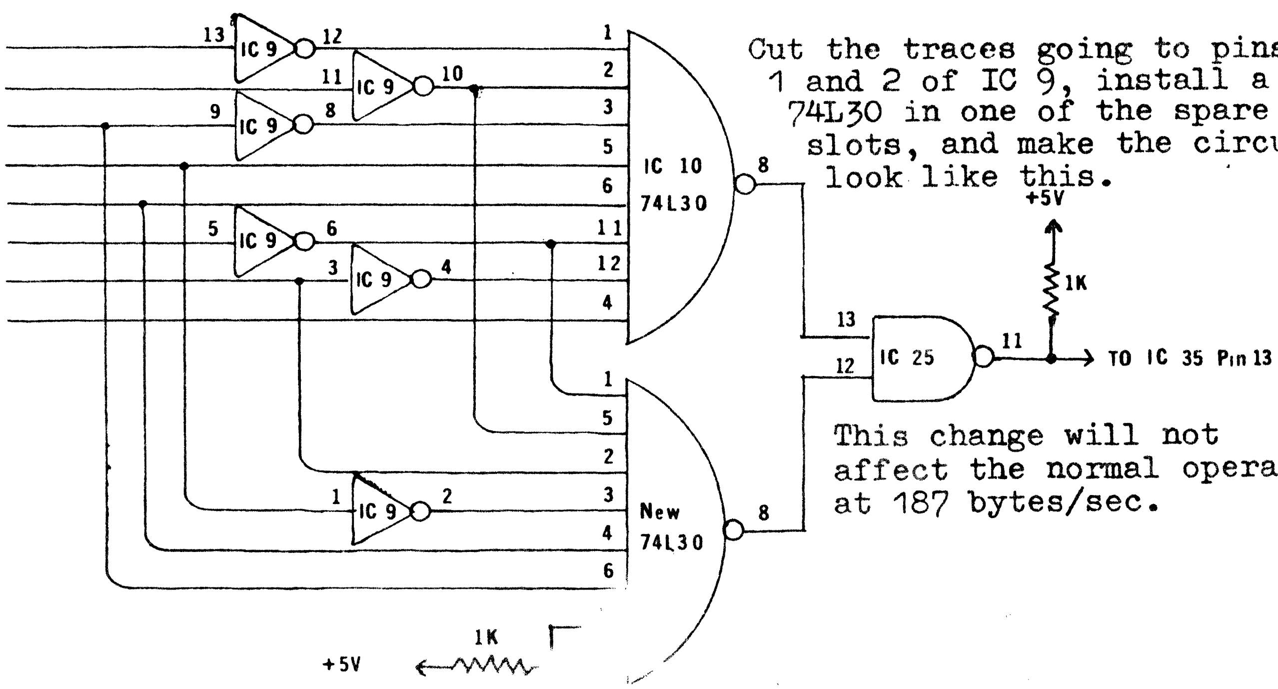
TARBELL

CONVERSION TO KANSAS CITY FORMAT

```
5F00                0010 ** KANSAS CITY ØUTPUT RØUTINE **
5F00                0020 ** ØUTPUTS THE BYTE IN REGISTER A. **
5F00                0030 * NØTE:  FREQUENCY ØF ØUTPUT ØSCILLATØR
5F00                0040 * SHØULD BE CHANGED TØ 4800 HZ FØR
5F00                0050 * ØUTPUT FREQUENCIES ØF 2400 AND 1200
5F00                0060 * FØR A 1 AND 0 RESPECTIVELY.
5F00                0070 *    ØCTØBER 27, 1976
5F00 C5             0080       PUSH B       SAVE REGISTERS B,C.
5F01 0E 08          0090       MVI  C,8     GET BIT CØUNT.
5F03 CD 1B 5F       0100       CALL ZERØ    DØ START BIT.
5F06 0F             0110 LØØP  RRC          LØØK AT LSB.
5F07 DC 15 5F       0120       CC   ØNE     IF 1, DØ A 1.
5F0A D4 1B 5F       0130       CNC  ZERØ    IF 0, DØ A 0.
5F0D 0D             0140       DCR  C       DECREMENT CØUNTER.
5F0E C2 06 5F       0150       JNZ  LØØP    DØ ALL 8 BITS.
5F11 CD 15 5F       0160       CALL ØNE     DØ TWØ STØP BITS.
5F14 C1             0165       PØP  B       RESTØRE B,C.
5F15 F5             0170 ØNE   PUSH PSW     SAVE REGISTER A.
5F16 3E 00          0180       MVI  A,0     GET 00000000 PATTERN.
5F18 C3 1E 5F       0190       JMP  REST    DØ REST ØF IT.
5F1B F5             0200 ZERØ  PUSH PSW     SAVE REGISTER A.
5F1C 3E 55          0210       MVI  A,55H   GET 01010101 PATTERN.
5F1E CD 23 5F       0220 REST  CALL CØUT    ØUTPUT TØ CASSETTE.
5F21 F1             0230       PØP  PSW     RESTØRE REGISTER A.
5F22 C9             0240       RET          RETURN.
5F23 F5             0250 CØUT  PUSH PSW     SAVE REGISTER A.
5F24 DB 6E          0260 ØLØØP IN   CASC    READ CASS. STATUS.
5F26 E6 20          0270       ANI  20H     LØØK AT ØUT RDY BIT.
5F28 C2 24 5F       0280       JNZ  ØLØØP   WAIT TILL READY.
5F2B F1             0290       PØP  PSW     RESTØRE REGISTER A.
5F2C D3 6F          0300       ØUT  CASD    ØUTPUT TØ CASSETTE.
5F2E C9             0310       RET          RETURN FRØM CØUT.
5F2F                0320 CASC  EQU  6EH     CASS STATUS PØRT.
5F2F                0330 CASD  EQU  6FH     CASS DATA PØRT.
5F2F                0340 PSW   EQU  6
5F2F                0350 SP    EQU  6
?
```

The frequency of the output oscillator may be changed from it's normal value of 3000 hz to 4800 hz by putting a resistor of about 14 kohms in parallel with R6 (27 kohm).

```
5F40                 0010 ** KANSAS CITY INPUT RØUTINE **
5F40                 0020 ** READS ØNE BYTE INTØ REGISTER A. **
5F40                 0022 * NØTE: IN ØRDER TØ USE THIS RØUTINE,
5F40                 0025 * MAKE THE CHANGES ØN YØUR BØARD
5F40                 0026 * AS SHØWN IN THE DIAGRAM BELØW.
5F40                 0030 *
5F40 C5              0034        PUSH B      SAVE B,C.
5F41 0E 08           0035 BLIN   MVI  C,8    SET CØUNT=8 BITS.
5F43 3E 10           0038        MVI  A,10H  RESET RECEIVER.
5F45 D3 6E           0039        ØUT  CASC
5F47 CD 73 5F        0040 BLØØP  CALL CASIN  READ A BYTE (BIT).
5F4A E6 3C           0050        ANI  3CH    LØØK AT MIDDLE 4 BITS.
5F4C CA 65 5F        0060        JZ   ØNE    IF XX0000XX, BIT=1.
5F4F FE 3C           0070        CPI  3CH    IF XX1111XX, BIT=1.
5F51 CA 65 5F        0080        JZ   ØNE
5F54 FE 14           0090        CPI  14H    IF XX0101XX, BIT=0.
5F56 CA 61 5F        0100        JZ   ZERØ
5F59 FE 28           0110        CPI  28H    IF XX1010XX, BIT=0.
5F5B CA 61 5F        0120        JZ   ZERØ
5F5E C3 41 5F        0130        JMP  BLIN   MUST BE NØISE.
5F61 BF              0140 ZERØ   CMP  A      CLEAR CARRY.
5F62 C3 66 5F        0150        JMP  REST   DØ REST LIKE ØNE.
5F65 37              0170 ØNE    STC         SET CARRY=1.
5F66 78              0180 REST   MØV  A,B    GET RESULT.
5F67 1F              0190        RAR         SHIFT CARRY INTØ MSB.
5F68 47              0200        MØV  B,A    PUT RESULT BACK.
5F69 0D              0210        DCR  C      DØNE WITH BYTE?
5F6A C2 47 5F        0220        JNZ  BLØØP  IF NØT, KEEP READING.
5F6D CD 73 5F        0230        CALL CASIN  READ AN EXTRA BYTE.
5F70 78              0240        MØV  A,B    GET RESULT.
5F71 C1              0250        PØP  B      RESTØRE B,C.
5F72 C9              0260        RET         RETURN.
5F73 DB 6E           0270 CASIN  IN   CASC   READ CASS STATUS.
5F75 E6 10           0280        ANI  10H    LØØK AT INPUT BIT.
5F77 C2 73 5F        0290        JNZ  CASIN  WAIT TILL READY.
5F7A DB 6F           0300        IN   CASD   READ DATA BYTE.
5F7C C9              0310        RET         RETURN FRØM CASIN.
5F7D                 0400 CASC   EQU  6EH    STATUS/CØNTRØL PØRT.
5F7D                 0410 CASD   EQU  6FH    DATA PØRT.
?
```



Cut the traces going to pins 1 and 2 of IC 9, install a 74L30 in one of the spare slots, and make the circuit look like this.

This change will not affect the normal operation at 187 bytes/sec.

```
5E00                     0005 *
5E00                     0010 *** READ-ØNLY MEMØRY PRØGRAM ***
5E00                     0015 * MØDIFIED FØR KANSAS-CITY *
5E00                     0016 *
5E00 31 00 60            0020 START LXI  SP,6000H    SET STACK PTR.
5E03 3E 0D               0034       MVI  A,0DH       PRINT CR, LF.
5E05 CD 58 5E            0036       CALL PTCN
5E08 3E 0A               0038       MVI  A,0AH
5E0A CD 58 5E            0040       CALL PTCN        ØN CØNSØLE.
5E0D 3E 3F               0042       MVI  A,'?'       PRINT "?"
5E0F CD 58 5E            0043       CALL PTCN        ØN CØNSØLE.
5E12 CD 64 5E            0050       CALL RDCN        READ KEYBØARD.
5E15 FE 45               0060       CPI  'E'        IF E,
5E17 CC 36 5E            0070       CZ   EXEC    EXECUTE A PRØGRAM.
5E1A FE 43               0080       CPI  'C'        IF C,
5E1C CC 9F 5E            0090       CZ   CINR     GØ TØ INPUT RØUTINE.
5E1F FE 53               0100       CPI  'S'        IF S,
5E21 CC 73 5E            0110       CZ   CSYNC   GØ TØ SYNC GEN.
5E24 FE 4F               0120       CPI  'Ø'        IF Ø,
5E26 CA 7B 5E            0130       JZ   CØUTR       GØ TØ CASS ØUT.
5E29 FE 49               0140       CPI  'I'        IF I,
5E2B CC 9F 5E            0150       CZ   CINR    GØ TØ CASS IN.
5E2E FE 4C               0160       CPI  'L'        IF L,
5E30 CC 9F 5E            0170       CZ   CINR    DØ A LØAD & GØ.
5E33 C3 00 5E            0200       JMP  START       START ØVER.
5E36                     6000 *
5E36                     6010 *** EXECUTE THE PRØGRAM AT THE ADDRESS ***
5E36                     6020 *
5E36 CD 3A 5E            6030 EXEC  CALL AHEX      READ ADDRESS FRØM KB.
5E39 E9                  6040       PCHL           JUMP TØ IT.
5E3A                     7000 *
5E3A                     7002 *** CØNVERT UP TØ 4 HEX DIGITS TØ BINARY *
5E3A                     7004 *
5E3A 21 00 00            7005 AHEX  LXI  H,0    GET A 16-BIT ZERØ.
5E3D 0E 04               7008       MVI  C,4    CØUNT ØF 4 DIGITS.
5E3F CD 64 5E            7009 AHE1  CALL RDCN   READ A BYTE.
5E42 29                  7010       DAD  H      SHIFT 4 LEFT.
5E43 29                  7012       DAD  H
5E44 29                  7014       DAD  H
5E45 29                  7016       DAD  H
5E46 CD 50 5E            7018       CALL AHS1   CØNVERT TØ BINARY.
5E49 85                  7020       ADD  L
5E4A 6F                  7022       MØV  L,A
5E4B 0D                  7024       DCR  C      4 DIGITS?
5E4C C2 3F 5E            7026       JNZ  AHE1   KEEP READING.
5E4F C9                  7028       RET         RETURN FRØM AHEX.
5E50                     7100 *
5E50                     7102 *** CØNVERT AN ASCII DIGIT TØ HEX ***
5E50                     7104 *
5E50 D6 30               7106 AHS1  SUI  48     ASCII BIAS.
5E52 FE 0A               7108       CPI  10     DIGIT 0-10
5E54 D8                  7110       RC
5E55 D6 07               7112       SUI  7      ALPHA BIAS.
5E57 C9                  7114       RET         RETURN FRØM AHS1.
5E58                     8000 *
5E58                     8002 *** PRINT REGISTER A ØN CØNSØLE ***
5E58                     8004 *
```

```
5E58 F5           8006 PTCN  PUSH PSW        SAVE REG A.
5E59 DB 00        8008 PTLØP IN   CØNC       READ PRINTER STATUS.
5E5B E6 80        8010       ANI  80H        IF BIT 7 NØT 0,
5E5D C2 59 5E     8012       JNZ  PTLØP      WAIT TILL IT IS.
5E60 F1           8014       PØP  PSW        THEN RECØVER A,
5E61 D3 01        8016       ØUT  CØND       AND PRINT IT.
5E63 C9           8018       RET             RETURN FRØM PTCN.
5E64              8100 *
5E64              8102 *** READ FRØM CØNSØLE TØ REGISTER A ***
5E64              8104 *
5E64 DB 00        8106 RDCN  IN   CØNC       READ KEYBØARD STATE
5E66 E6 01        8108       ANI  1          IF BIT 1 NØT 0,
5E68 C2 64 5E     8110       JNZ  RDCN       REPEAT UNTIL IT IS.
5E6B DB 01        8112       IN   CØND       READ FRØM KEYBØARD.
5E6D E6 7F        8114       ANI  7FH        STRIP ØFF MSB.
5E6F CD 58 5E     8115       CALL PTCN       ECHØ ØNTØ PRINTER.
5E72 C9           8116       RET             RETURN FRØM RDCN.
5E73              8118 CØNC  EQU  0          CØNSØLE STATUS PØRT
5E73              8120 CØND  EQU  1          CØNSØLE DATA PØRT.
5E73              9100 *
5E73              9102 *** SYNC CØDE GENERATØR PRØGRAM ***
5E73              9103 *
5E73 3E 55        9104 CSYNC MVI  A,55H      WRITE SYNC BYTE
5E75 CD 9C 5E     9106       CALL CØUT       ØNTØ CASSETTE.
5E78 C3 73 5E     9108       JMP  CSYNC      KEEP DØING IT.
5E7B              9200 *
5E7B              9202 *** CASSETTE INTERFACE ØUTPUT RØUTINE ***
5E7B              9204 *
5E7B CD 3A 5E     9206 CØUTR CALL AHEX   READ BLØCK LENGTH.
5E7E EB           9210       XCHG        PUT INTØ D,E.
5E7F CD 3A 5E     9212       CALL AHEX   READ STARTING ADR.
5E82 06 00        9216       MVI  B,0    START CHECKSUM = 0.
5E84 7E           9226 CØLØP MØV  A,M        GET DATA FRØM MEMØRY
5E85 CD 9C 5E     9228       CALL CØUT       SEND TØ CASSETTE.
5E88 80           9230       ADD  B      ADD TØ CHECKSUM.
5E89 47           9232       MØV  B,A
5E8A 23           9234       INX  H          INCREMENT PØINTER.
5E8B 1B           9236       DCX  D      DECREMENT CØUNTER.
5E8C 97           9238       SUB  A          CLEAR A.
5E8D BA           9240       CMP  D      IF D NØT = 0,
5E8E C2 84 5E     9242       JNZ  CØLØP       REPEAT LØØP.
5E91 BB           9244       CMP  E          IF E NØT = 0,
5E92 C2 84 5E     9246       JNZ  CØLØP       REPEAT LØØP.
5E95 78           9248       MØV  A,B        GET CHECKSUM.
5E96 CD 9C 5E     9250       CALL CØUT       ØUTPUT IT.
5E99 C3 00 5E     9252       JMP  START      GET ANØTHER CØMMAND
5E9C C3 00 5F     9254 CØUT  JMP  5F00H      GØ TØ BLØUT.
5E9F              9300 *
5E9F              9302 *** CASSETTE INPUT RØUTINE ***
5E9F              9304 *
5E9F F5           9310 CINR  PUSH PSW    SAVE CØNTRØL CHAR.
5EA0 3E 10        9320       MVI  A,10H  USE BIT 4 IN REG A
5EA2 D3 6E        9330       ØUT  CASC   TØ RESET CASS INT.
5EA4 CD 3A 5E     9340       CALL AHEX   READ BLØCK LENGTH.
5EA7 EB           9350       XCHG        PUT INTØ D,E.
5EA8 CD 3A 5E     9360       CALL AHEX   READ STARTING ADDRESS.
5EAB F1           9365       PØP  PSW    GET CØNTRØL CHAR.
5EAC E5           9370       PUSH H      SAVE STARTING ADDRESS
5EAD F5           9375       PUSH PSW    UNDER CØNTRØL CHAR.
5EAE 06 00        9380       MVI  B,0    SET CHECKSUM = 0.
5EB0 CD E1 5E     9390 CILØP CALL CIN    READ A BYTE FRØM CASS.
5EB3 4F           9400       MØV  C,A    SAVE IT IN REG C.
5EB4 F1           9410       PØP  PSW    GET CØNTRØL CHAR.
5EB5 F5           9420       PUSH        SAVE IT BACK.
5EB6 FE 43        9430       C           IS IT A C?
5EB8 79           9440       MØV  A,C    GET BACK DATA BYTE.
```

```
5EB9 CA BD 5E          9450         JZ    CINØ     IF C, DØN'T STØRE IT.
5EBC 77                9460         MØV   M,A      IF NØT, DØ STØRE IT.
5EBD 80                9470 CINØ    ADD   B        ADD TØ CHECKSUM.
5EBE 47                9480         MØV   B,A
5EBF 23                9490         INX   H        INCREMENT PØINTER.
5EC0 1B                9500         DCX   D        DECREMENT CØUNTER.
5EC1 97                9510         SUB   A        CLEAR A.
5EC2 BA                9520         CMP   D        IF D NØT = 0,
5EC3 C2 B0 5E          9530         JNZ   CILØP    READ MØRE.
5EC6 BB                9540         CMP   E        IF E NØT = 0,
5EC7 C2 B0 5E          9550         JNZ   CILØP    READ MØRE.
5ECA CD E1 5E          9560         CALL  CIN      READ LAST BYTE.
5ECD B8                9570         CMP   B        CØMPARE TØ CHECKSUM.
5ECE 3E 45             9580         MVI   A,'E'    PRINT E FØR ERRØR.
5ED0 C2 DB 5E          9590         JNZ   CERR     PRINT NØW IF ERRØR.
5ED3 F1                9600         PØP   PSW      RECØVER CTL CHAR.
5ED4 FE 4C             9610         CPI   'L'      IF IT'S NØT L,
5ED6 C2 DB 5E          9620         JNZ   CERR     DØN'T EXECUTE.
5ED9 E1                9630         PØP   H        ØTHERWISE, EXECUTE
5EDA E9                9640         PCHL           AT STARTING ADDRESS.
5EDB CD 58 5E          9660 CERR    CALL  PTCN     PRINT C,E, ØR I.
5EDE C3 00 5E          9690         JMP   START
5EE1 C3 40 5F          9700 CIN     JMP   5F40H    GØ TØ BLIN.
5EE4                   9996 CASD    EQU   6FH      CASSETTE DATA PØRT.
5EE4                   9997 CASC    EQU   6EH      CASSETTE STATUS PØRT.
5EE4                   9998 PSW     EQU   6
5EE4                   9999 SP      EQU   6
?
```

THE

TARBELL

CASSETTE

INTERFACE

TARBELL ELECTRONICS
20620 S. Leapwood Ave., Suite P
Carson, California 90746
(213) 538-4251

TABLE OF CONTENTS

PAGE

January 14, 1977

TARBELL ELECTRONICS
20620 S. Leapwood Ave., Suite P
Carson, California 90746
(213) 538-4251

THE TARBELL CASSETTE INTERFACE (FOR ALTAIR 8800 USERS)

SPEED: UP TO 540 BYTES PER SECOND (2200 BITS PER INCH).
187 BYTES PER SECOND FOR TARBELL STANDARD 800 BITS/INCH.
30 BYTES PER SECOND FOR "BYTE/LANCASTER" STANDARD.

ENCODING METHOD: PHASE-ENCODED (EXCLUSIVE-OR OF CLOCK AND DATA).
SELF-CLOCKING (CLOCK VARIES ALONG WITH TAPE SPEED).
USED ON MY OWN SYSTEM FOR THE LAST 4 YEARS.
CAN BE USED TO GENERATE "BYTE/LANCASTER" TAPES.

CASSETTE: WILL WORK WITH MOST AUDIO CASSETTE UNITS. MAY BE ADAPTED TO AUTOMATIC DIGITAL CASSETTE UNITS. WILL ALSO WORK WITH REGULAR REEL-REEL TAPE RECORDERS. I HAVE BEEN USING A REALISTIC (RADIO SHACK) CTR-19 AND A J.C. PENNY 6536 ($39.95). TAPE SHOULD BE OF LOW-NOISE TYPE.

8192 BYTE LOAD TIME: 15 SEC @ 540 BYTES PER SECOND.
43 SEC @ 187 BYTES PER SECOND.
4 MINUTES @ 30 BYTES PER SECOND.

DEVICE-CODE EASILY SELECTED WITH ON-BOARD DIP-SWITCH.

STATUS: 4 EXTRA STATUS LINES AVAILABLE FOR INPUT.

CONTROL: 4 EXTRA CONTROL LINES AVAILABLE FOR OUTPUT, WHICH MAY BE USED TO DRIVE RELAYS FOR EXTRA CASSETTE UNITS.
2 SPARE IC SLOTS TO LET YOU DO YOUR OWN THING.

COMPATIBILITY: PLUGS DIRECTLY INTO ALTAIR 8800 OR IMSAI 8080.
HAS SERIAL-PARALLEL AND PARALLEL-SERIAL CONVERSION ON BOARD. PATCHES PROVIDED FOR POPULAR SOFTWARE.

SOFTWARE: COMES WITH COMPLETE SET OF INPUT/OUTPUT SUBROUTINES, BOOTSTRAP, AND "BYTE STANDARD" (LANCASTER) SOFTWARE.

COST: $120 FOR COMPLETE KIT, $175 ASSEMBLED AND CHECKED-OUT.

MANUAL:
ASSEMBLY INSTRUCTIONS AND DRAWING, PARTS AND PIN FUNCTION LISTS
SOLDERING, CLEANING, AND INSTALLATION NOTES; OPERATING INSTRUCTIONS
INITIAL ADJUSTMENT PROCEDURES, INPUT/OUTPUT ROUTINES WITH CHECKSUM
BOOTSTRAP PROGRAM AND TEST-STREAM GENERATOR PROGRAM

PARTS:
ALL RESISTORS, CAPACITORS, AND INTEGRATED CIRCUITS
CASSETTE CABLE, RIBBON CABLE, AND DIP CONNECTOR
LOW-NOISE CASSETTE WITH TEST STREAM
DOUBLE-SIDED BOARD WITH PLATED-THRU HOLES AND GOLD EDGE CONNECTOR

WARRANTY: IF NOT COMPLETELY SATISFIED, RETURN BOARD FOR REFUND OR FREE REPAIR WITHIN 90 DAYS AFTER PURCHASE.

FIRST DELIVERIES WERE MADE IN SEPTEMBER, 1975. DELIVERY IS 1 TO 3 WEEKS AFTER RECEIVING ORDER. THE 25-PAGE MANUAL IS AVAILABLE AT $4. CALIFORNIA RESIDENTS PLEASE ADD 6% SALES TAX. MAKE CHECK OR MONEY ORDER PAYABLE TO TARBELL ELECTRONICS.

HISTORY AND SALES PITCH

I HAVE BEEN USING AN INEXPENSIVE AUDIO CASSETTE RECORDER IN MY HOME-DESIGNED COMPUTER SYSTEM SINCE 1972. I HAVE OVER 600 FILES ON CASSETTES, MOSTLY ABOUT 4 KBYTES EACH. MY ESTIMATE IS THAT THE ERROR RATE IS LESS THAN 1 ERROR IN 1,000,000 BITS. I SAY THIS BECAUSE I CAN USUALLY RECORD 30 4 KBYTE FILES ON ONE SIDE OF A C-60 CASSETTE WITHOUT ANY ERRORS. THIS INTERFACE GAVE ME VERY GOOD SERVICE WHILE I WAS WRITING THE DISK OPERATING SYSTEM FOR MY 500 KBYTE DISK. SINCE I STARTED USING MY DISK SYSTEM A FEW YEARS AGO, THE CASSETTE HAS SERVED AS BACKUP STORAGE - A RELIABLE PLACE TO STORE DATA AND PROGRAMS AFTER THEY ARE DEBUGGED.

THE ENCODING METHOD I USE IS VERY SIMPLE, AND HAS BEEN IN USE IN INDUSTRY FOR QUITE SOME TIME. PICTURE A SHIFT REGISTER WHICH IS LOADED WITH THE DATA TO BE RECORDED. THE REGISTER IS THEN CLOCKED WITH A SQUARE WAVE. THE OUTPUT OF THE SHIFT REGISTER IS EXCLUSIVE-ORED WITH THE CLOCK, PRODUCING THE BI-PHASE DATA. THIS DATA GOES DIRECTLY TO THE CASSETTE RECORDER'S INPUT. THE MOST DIFFICULT PART OF THE PROCESS IS RECOVERING THE DATA. MANY LONG HOURS WERE SPENT STUDYING THIS PROBLEM AND TRYING DIFFERENT METHODS. IN THE ORIGINAL INTERFACE THIS WAS ACCOMPLISHED WITH A 760 HIGH-SPEED COMPARATOR, A 74121 NON-RETRIGGERABLE ONE-SHOT, AND A DOUBLE-GLITCH GENERATOR MADE WITH AN EXCLUSIVE-OR GATE. SINCE THEN, THE 8T20 HAS BEEN DEVELOPED, WHICH COMBINES THESE THREE FUNCTIONS ON A SINGLE CHIP, AND IS THE UNIT USED IN THE ALTAIR INTERFACE.

THE PRESENT DESIGN IS EVEN MORE RELIABLE THAN THE PREVIOUS ONE, AND IS CAPABLE OF RECORDING AND RECOVERING ERROR-FREE DATA AT A RATE OF 540 BYTES PER SECOND ON A STANDARD AUDIO CASSETTE RECORDER. (YES, THAT IS OVER 2200 BITS PER INCH!) I AM STILL, HOWEVER, ENCOURAGING USERS TO EXCHANGE DATA RECORDED AT 187 BYTES PER SECOND (1500 BITS PER SECOND, 800 BITS PER INCH). THE MAIN ADVANTAGE OF THIS METHOD OVER OTHERS IS IT'S ABILITY TO WITHSTAND A LARGE AMOUNT OF WOW AND FLUTTER, WHICH MAY BE INTRODUCED BY CHEAP RECORDERS, AND STILL RECOVER THE DATA RELIABLY. THIS FEATURE STEMS FROM THE SELF-CLOCKING NATURE OF THE RECORDED SIGNAL: THE RECOVERED CLOCK VARIES RIGHT ALONG WITH THE DATA, SO THAT THE SPEED VARIATIONS ARE ESSENTIALLY IGNORED. THE MAIN DISADVANTAGE OF THIS METHOD IS THAT IT REQUIRES GOOD LOW-NOISE TAPE, AND A DECENT FREQUENCY RESPONSE ON THE CASSETTE UNIT. THE CASSETTE UNIT I'VE BEEN USING LATELY (J.C. PENNY #6536) HAS A FREQUENCY RESPONSE OF 80-8,000 HZ. THE MOST IMPORTANT PART IS THE HIGH END. THESE REQUIREMENTS ARE DUE TO THE HIGH SPEED OF THE INTERFACE, AND WOULD BE THE SAME FOR ANY HIGH SPEED DEVICE.

THE SPEED MAY NOT SEEM VERY IMPORTANT TO YOU NOW. BUT A GOOD PORTION OF YOUR TIME IS GOING TO BE SPENT SAVING AND LOADING DATA, PROGRAMS, AND OTHER TEXT. THERE IS A WORLD OF DIFFERENCE BETWEEN LOADING BASIC AT SAY, 30 BYTES/SEC (4 MINUTES), AND AT 187 BYTES/SEC (40 SEC). IT DOESN'T SEEM LIKE MUCH, BUT WHEN YOU HAVE TO DO IT OVER, AND OVER, AND OVER.... IT GETS TO BE A BIT MUCH. ESPECIALLY WHEN YOU'RE DEVELOPING YOUR OWN PROGRAMS, AND THEY TEND TO RUN AMUCK AND WIPE OUT CORE. THINK ABOUT IT, THEN BUY THE TARBELL CASSETTE INTERFACE. THE ONLY METHOD PROVEN WITH TIME. ASK YOUR FRIEND WHO HAS ONE.

SELECTING A CASSETTE UNIT FOR DIGITAL RECORDING

FIRST OF ALL, THE MOST EXPENSIVE CASSETTE RECORDERS ARE NOT NECESSARILY THE BEST FOR RECORDING DIGITAL DATA. THERE ARE SEVERAL FACTORS THAT COMBINE TO MAKE A GOOD UNIT FOR THE HOBBYIST:

1. IT SHOULD HAVE A GOOD HIGH-FREQUENCY RESPONSE, PREFERABLY UP TO AT LEAST 8,000 HZ.

2. IT SHOULD HAVE A TONE CONTROL, SO THAT THE INHERENT FREQUENCY RESPONSE MAY BE REALIZED.

3. ALTHOUGH AUTOMATIC VOLUME CONTROL IS MORE CONVENIENT FROM AN OPERATIONAL POINT OF VIEW, IT ALSO REQUIRES A FEW SECONDS OF SETTLING TIME BEFORE STARTING TO RECORD.

4. IF IT DOES NOT HAVE AUTOMATIC VOLUME CONTROL, IT IS GOOD TO HAVE A RECORDING LEVEL METER. THIS ALLOWS EASIER ADJUSTMENT FOR THE CORRECT RECORDING LEVEL.

5. IT IS VERY IMPORTANT TO HAVE A DIGITAL COUNTER. THIS MAKES IT POSSIBLE TO QUICKLY LOCATE THE DESIRED PROGRAM AMONG SEVERAL.

6. IT SHOULD BE CAPABLE OF RUNNING DIRECTLY ON THE AC LINE. BATTERIES TEND TO MAKE THE MOTOR GET SLOWER AS THEY WEAR.

7. IT IS HANDY TO HAVE AN AUXILLIARY INPUT, SO THAT A FAIRLY HIGH LEVEL MAY BE FED TO THE RECORDER, AND NOISE KEPT TO A MINIMUM.

8. A REMOTE INPUT JACK IS VALUABLE TO CONTROL START-STOP DURING ASSEMBLER AND COMPILER OPERATIONS.

9. JACKS FOR MIC, AUX, REMOTE, AND EARPHONE ARE USUALLY INCLUDED, BUT YOU SHOULD CHECK TO MAKE SURE THEY ARE THERE, ANYWAY.

10. LOW WOW AND FLUTTER CHARACTERISTICS ARE IMPORTANT, BUT ARE MUCH MORE IMPORTANT WHEN USING AN ASYNCRONOUS INTERFACE, WHICH IS NOT SELF-CLOCKING.

11. IF IT IS DESIRED TO DO AUTOMATIC REWIND, FAST-FORWARD, AND RECORD/PLAYBACK SWITCHING UNDER PROGRAM CONTROL, YOU MUST PURCHASE A RECORDER THAT HAS THESE FACILITIES. ANOTHER FEATURE TO LOOK FOR ON THIS TYPE OF UNIT IS A WAY TO KEEP TRACK OF WHERE YOU ARE ON THE CASSETTE TAPE.

12. PEOPLE HAVE EXPERIENCED PROBLEMS WITH PANASONIC RECORDERS. I AM PRESENTLY RECOMMENDING THE J.C. PENNEY MODEL 6536 AT $39.95 AS THE BEST BUY I KNOW, AND SEVERAL PEOPLE ARE USING THEM WITH MY INTERFACE WITH GOOD RESULTS. OTHER BRANDS THAT I KNOW HAVE BEEN SUCCESSFUL ARE SONY AND REALISTIC.

OF COURSE, NONE OF THE ABOVE ITEMS IS ABSOLUTELY NECESSARY FOR RECORDING DIGITAL DATA ON AN AUDIO CASSETTE. BUT THE MORE OF THESE REQUIREMENTS THAT ARE FILLED, THE MORE CAPABLE YOUR UNIT WILL BE, AND THE EASIER IT WILL BE TO USE.

USING THE TARBELL CASSETTE INTERFACE FOR THE KANSAS CITY FORMAT

Some time ago, there was a meeting of various cassette interface manufacturers to determine a standard for exchange of programs and data on cassettes among computer hobbyists. The format that was proposed as a result of the meeting is a modified version of the coding technique described by Don Lancaster in the first issue of "BYTE" Magazine. In this format, each 8-bit byte is written on tape in an asyncronous format, with one start bit (zero), 8 data bits (zero or one), and two stop bits (ones). A one is defined as 8 cycles at 2400 bits per second, and a zero is defined as 4 cycles at 1200 bits per second. This provides a data transfer speed of 300 baud, or a little less than 30 bytes per second, and may be generated and decoded using a variety of techniques.

Since the standard is fairly slow, it suggests that many people may want to have two methods available. One that provides for the Kansas City (BYTE/Lancaster) format, and another that is much faster, to speed program loading and development. The Tarbell Cassette Interface may easily be modified for both methods.

First, the output oscillator frequency will have to be raised from 3000 hz to 4800 hz. This is because a higher bit density is required of the tape, although the actual data transfer rate is much slower. A one may be generated by writing a word of all zeroes (00000000), and a zero may be generated by writing a word of alternating ones and zeroes (01010101). An output subroutine converts each byte to be written in this format from parallel to serial form (required only for this format).

On the input side, the adjustment of the potentiometer (R8) will have to be changed for the higher frequency. The sync detector circuit(IC's 9 and 10) will have to be changed so that it recognizes the alternating bit pattern as a sync byte in addition to the normal sync byte of E6 (hex). An input subroutine converts each byte from it's serial form to it's parallel form (required only for this format).

Using the method outlined above, the Tarbell Cassette Interface can be modified so that a double-pole, single-throw switch will determine which frequency will be used. The software determines the format. Another alternative is to change to the higher frequency permanently, so that no switch is necessary. The disadvantage of this is that you would have to readjust the potentiometer to read tapes made with the standard 3000 hz oscillator (187 bytes per second), and that a slightly higher frequency response is required on the part of your recorder.

A description of the hardware modifications, and listings of the subroutines for the operations described above are available upon request. Please include a self-addressed stamped envelope.

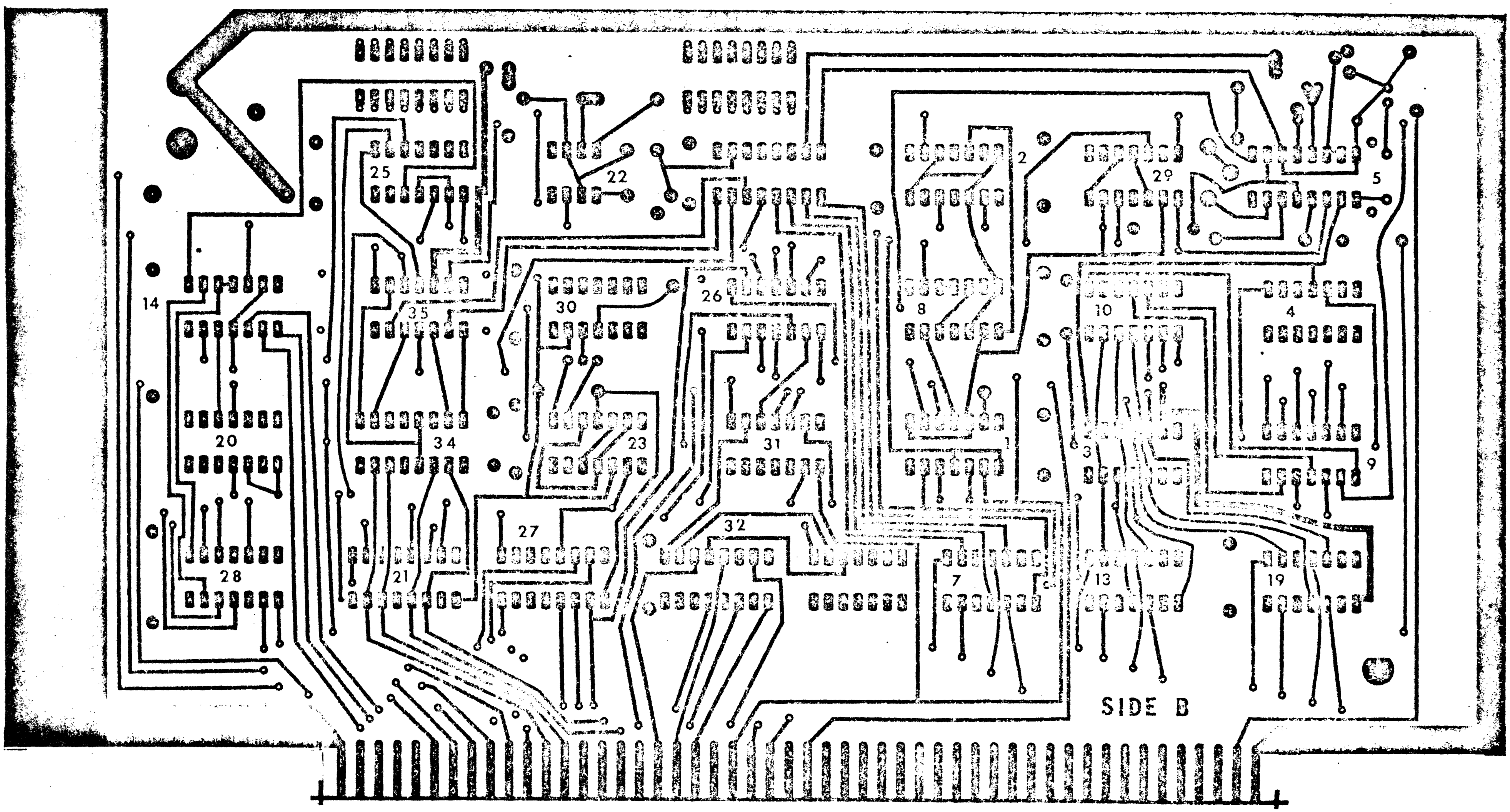
CASSETTE INTERFACE PARTS LIST

INTEGRATED CIRCUITS

REF NOS	DESCRIPTION	QTY	PT NO	+5	GND	-5
1	QUAD 2-INPUT AND	1	7408	14	7	
2	DUAL J-K FLIP-FLOP	1	7473	4	11	
3	8-BIT SHIFT REGISTER	1	74L164	14	7	
4,23	DUAL TYPE-D FLIP-FLOP	2	74L74	14	7	
6	5-VOLT REGULATOR	1	LM309K			
7,13,19,25	QUAD 2-INPUT NAND O/C	4	7403	14	7	
8,29	DUAL J-K FLIP-FLOP	2	74L73	4	11	
5	BI-DIRECTIONAL ONE-SHOT	1	8T20	16	8	4
9,31	HEX INVERTER	2	74L04	14	7	
10	8-INPUT NAND	1	74L30	14	7	
14	TRIPLE 3-INPUT AND	1	74L11	14	7	
20	HEX INVERTER	1	7404	14	7	
21,27	5-BIT SHIFT REGISTER	2	7496	5	12	
22	*TIMER	1	NE555	8	1	
26	QUAD 2-INPUT NAND	1	74L00	14	7	
28	DUAL 4-INPUT NOR	1	7425	14	7	
30	QUAD EXCLUSIVE-OR	1	74L86	14	7	
32	6-BIT COMPARATORS	1	DM8131	16	8	
34	QUAD LATCH	1	74L75	5	12	
35	HEX INVERTER BUFFER O/C	1	7406	14	7	

C1-C5,C10-C15	.1 MFD CAPACITOR	
C6	.02 MFD CAPACITOR	
C7	.033 OR .039 MFD CAPACITOR	
C8	*.01 MFD CAPACITOR	
C9	22 OR 25 MFD CAPACITOR	
C16	2200 PF CAPACITOR	
C17	.01 MFD CAPACITOR	
R1	2.4 KOHM RESISTOR	(RED,YELLOW,RED)
R2	4.7 KOHM RESISTOR	(YELLOW,VIOLET,RED)
R3	1.5 KOHM RESISTOR	(BROWN,GREEN,RED)
R4	330 OHM RESISTOR	(ORANGE,ORANGE,BROWN)
R5	220 OHM RESISTOR 1W	(RED,RED,BROWN)
R6	*27 KOHM RESISTOR	(RED,VIOLET,ORANGE)
R7	*APPROX 10 KOHM RESISTOR	(BROWN,BLACK,ORANGE)
R8	50 KOHM POTENTIOMETER	
R9	100 OHM RESISTOR	(BROWN,BLACK,BROWN)
R10-R17	1 KOHM RESISTOR	(BROWN,BLACK,RED)
CR1	1N914 SIGNAL DIODE	
CR2	1N750 4.7 VOLT ZENER DIODE	
CR3	LIGHT-EMITTING DIODE	
S1	DIP-SWITCH	
J1	DIP-SOCKET	
P1	DIP-PLUG	
	2 CASSETTE COAX CABLES	
	FLAT RIBBON CABLE	
PC1	PRINTED CIRCUIT BOARD	
CS1	SCOTCH LOW-NOISE CASSETTE	
	TO-3 INSULATING WAFER	
	2 SETS OF 4-40 NUTS, SCREWS, AND WASHERS	

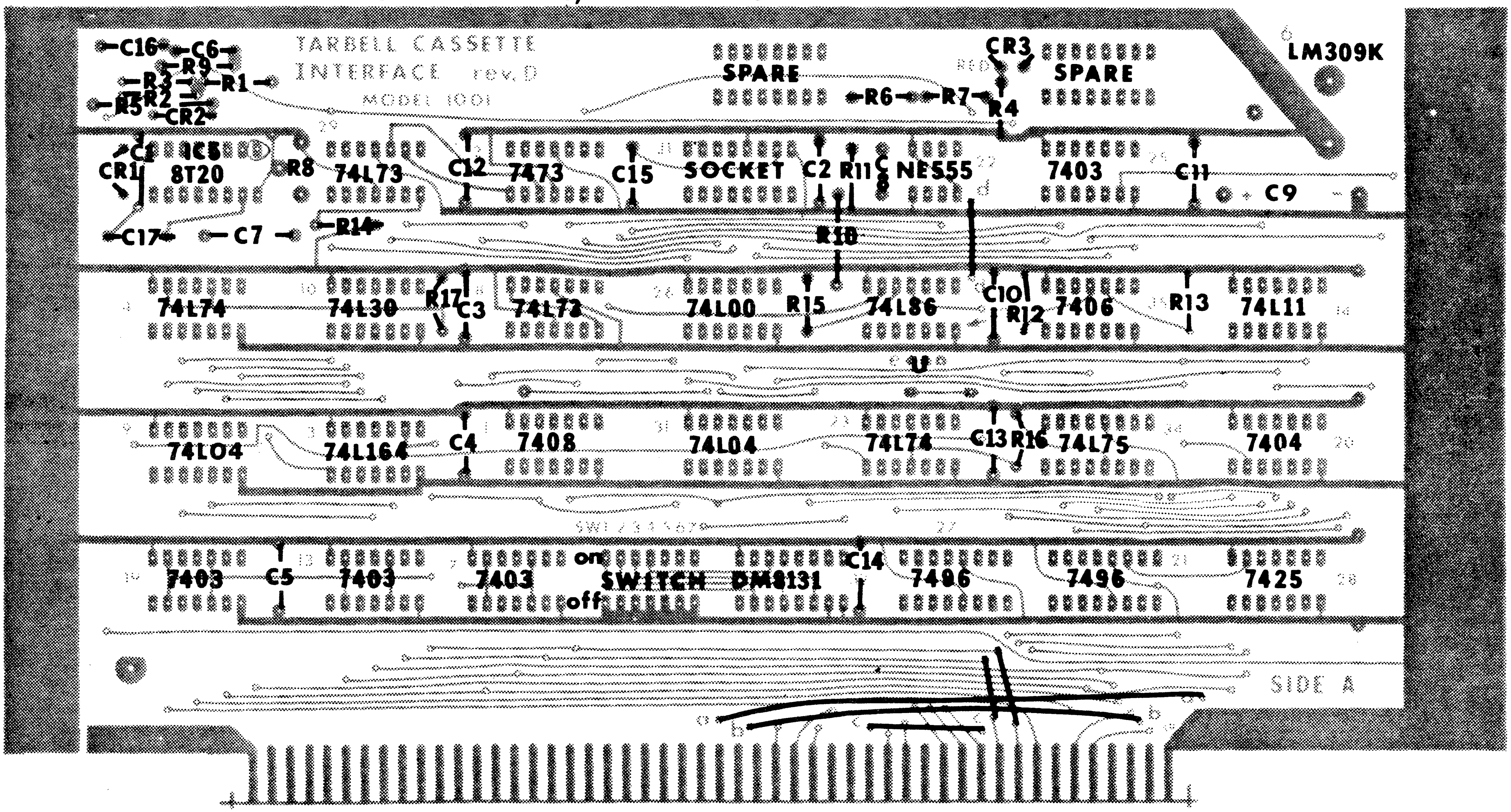
*** THE ITEMS MARKED WITH AN ASTERISK ARE MATCHED AND BAGGED SEPARATELY.**
74LS MAY BE SUBSTITUTED FOR 74L SERIES IC'S IN MOST CASES.



SOLDER SIDE

ASSEMBLY DRAWING

NOTE: IC5 HAS PIN 1 IN UPPER RIGHT, REST IN LOWER LEFT.



COMPONENT SIDE

ASSEMBLY INSTRUCTIONS

1. TAKE OUT ALL THE PARTS AND CHECK THEM AGAINST THE PARTS LIST. IF THERE ARE ANY PARTS MISSING, DROP US A NOTE, AND WE WILL SEND THEM TO YOU. NOTE THAT THE NE555, 10K (APPROX.) RESISTOR, 27K RESISTOR, AND .01 MFD CAPACITOR WHICH ARE PACKAGED IN THE SEPERATE BAG ARE MATCHED FOR 1500 HZ AND ARE MARKED WITH ASTERISKS ON THE PARTS LIST. OTHER COMPONENTS USED FOR THESE WILL CAUSE THE OUTPUT TO RUN AT A DIFFERENT FREQUENCY.

2. USING THE ASSEMBLY DRAWING ON PAGE 7, INSERT THE DISCRETE COMPONENTS (BAG WITH CAPACITORS, RESISTORS, LED, DIP-SWITCH) INTO THEIR PROPER LOCATIONS. ON THE LIGHT-EMITTING DODE (LED), THE RED LEAD GOES TO THE RESISTOR. IF THE RED HAS BEEN RUBBED OFF THE LEAD, USE AN OHMMETER TO DETERMINE WHICH LEAD GOES TO THE RESISTOR. THE 1N750 ZENER DIODE (SMALL GLASS DEVICE) SHOULD BE MOUNTED WITH THE LINE SIDE TOWARD THE LEFT. THE 1N914 DIODE (OTHER SMALL GLASS DEVICE) SHOULD BE MOUNTED WITH THE LINE SIDE TOWARD THE BOTTOM. THE ONLY CAPACITOR WHICH SHOULD BE ORIENTED IN A PARTICULAR DIRECTION IS THE FILTER CAPACITOR C9, WHICH SHOULD BE MOUNTED WITH THE PLUS SIDE TOWARD THE LEFT, AS MARKED ON THE BOARD. THE DIP-SOCKET SHOULD BE MOUNTED IN THE J1 POSITION, AS IS SHOWN ON THE ASSEMBLY DRAWING.

3. INSERT ALL THE INTEGRATED CIRCUITS (IC'S) (SMALL BLACK DEVICES AND ONE LARGE METAL DEVICE) INTO THEIR PROPER LOCATIONS AS INDICATED BY THE ASSEMBLY DRAWING AND THE PARTS LIST. NOTE THAT IC #5 (8T20) IS ORIENTED WITH PIN 1 AT UPPER RIGHT, WHEREAS ALL THE OTHERS ARE ORIENTED WITH PIN1 AT LOWER LEFT. THERE ARE 2 WAYS TO TELL WHICH IS PIN 1 ON AN IC. ON SOME, THERE IS A SMALL DOT BY PIN 1. ON OTHERS, THERE IS A NOTCH AT ONE END OF THE IC, AND PIN 1 IS ON THE LOWER LEFT WHEN THE NOTCH IS AT THE LEFT. BE SURE TO PUT THE MICA INSULATOR UNDER THE REGULATOR (IC 6) BEFORE MOUNTING IT. BE SURE TO PUT IC 6 IN SO THAT THE SCREW HOLES LINE UP, AND THEN INSTALL THE SCREWS, WITH THE WASHERS UNDER THE NUTS. BE SURE THAT THE WASHERS DON'T TOUCH THE TRACE ON SIDE B OF THE PC BOARD, AND THAT CAPACITOR C9 IS NOT TOUCHING ANY PART OF IC 6.

4. INSTALL THE SEVEN JUMPER WIRES, PREFERABLY USING SMALL INSULATED SINGLE-STRAND COPPER WIRE. THE JUMPER WIRES ARE MARKED ON THE ASSEMBLY DRAWING IN HEAVY BLACK LINES. FIVE OF THEM ARE ALSO MARKED ON THE BOARD WITH MATCHING LETTERS (A,B,C,D,E) BESIDE THEIR HOLES. NOTICE THAT THE BOTTOM OF JUMPER D GOES TO THE HOLE ABOVE THE "D". THERE ARE TWO MORE VERTICAL JUMPERS DOWN NEXT TO THE CONNECTOR. THE LAST IS A SHORT ONE (E), BELOW IC 30, TO SELECT THE OUTPUT PHASE. MINE IS CONNECTED BETWEEN THE CENTER AND LEFT HOLES, AS SHOWN IN THE SCHEMATIC. THIS MAY HAVE TO BE CHANGED LATER, AS NOTED IN THE ADJUSTMENT PROCEDURE.

5. CONSULT THE INSTALLATION NOTES FOR INSTRUCTIONS ABOUT THE CASSETTE CABLES.

6. SEE THE MODIFICATION SHEET (PAGE 8) FOR ANY LATE MODIFICATIONS. REVISION C HAS ALL MODIFICATIONS THROUGH MOD. #6. BE SURE THAT ALL MODIFICATIONS WHICH ARE NOT ON YOUR REVISION ARE INSTALLED.

MODIFICATIONS ON THE TARBELL CASSETTE INTERFACE AS OF SEPT 1, 1976.

INCLUDED ON REVISION B:

1. RESISTOR R1 HAS BEEN CHANGED TO A 2.4 KOHM RESISTOR.
2. ON THE OUTPUT VOLTAGE DIVIDER, R10 IS RECOMMENDED TO BE 1 KOHM FOR AUXILLIARY CASSETTE INPUTS, INSTEAD OF 10 KOHM.

INCLUDED ON REVISION C:

3. A .1 MFD CAPACITOR HAS BEEN ADDED IN PARALLEL WITH R11. THIS IMPROVES RELIABILITY WITH SOME TYPES OF RECORDERS.
4. A 1 KOHM RESISTOR HAS BEEN ADDED BETWEEN PINS 10 AND 14 OF IC 35. THIS PROVIDES PULL-UP TO DRIVE THE 7475 LATCH.
5. THE TRACE TO PIN 8 OF IC 5 (GROUND) HAS BEEN CUT AND A LINE RUN DIRECTLY TO THE GROUND BUS ON THE BOTTOM. THIS ELIMINATES CROSS-TALK FROM THE NE555 OSCILLATOR.
6. A 2200 PF CAPCITOR SHOULD BE ADDED BETWEEN PINS 6 AND 8 ON IC 5. THIS REDUCES THE EFFECTS OF HIGH-FREQUENCY NOISE GENERATED IN SOME COMPUTERS.

INCLUDED ON REVISION D:

7. ADD A .01 MFD CAPACITOR BETWEEN PINS 15 AND 8 OF IC 5.
8. ADD A 1K RESISTOR BETWEEN PINS 12 AND 14 OF IC 30.
9. CONNECT ONE SIDE OF A 1K RESISTOR TO 5 VOLTS, THE OTHER TO IC29-7&10, IC2-14&3&7&10, IC8-14&3&7.
10. CONNECT ONE SIDE OF A 1K RESISTOR TO 5 VOLTS, THE OTHER TO IC23-4&10&13.
11. CONNECT ONE SIDE OF A 1K RESISTOR TO 5 VOLTS, THE OTHER TO IC4-13&10, IC3-9.
12. CONNECT PIN 1 TO PIN 2 ON IC3.

THERE HAVE BEEN SEVERAL QUESTIONS REGARDING THE USE OF RATES HIGHER THAN THE STANDARD 187 BYTES PER SECOND. CHANGES ARE REQUIRED ON BOTH THE INPUT AND OUTPUT SECTIONS. ON THE INPUT SECTION, THE POTENTIOMETER THAT IS PROVIDED ON THE BOARD MAY BE ADJUSTED TO CHANGE THE FREQUENCY. ON THE OUTPUT SECTION, ANY OF THREE COMPONENTS MAY BE CHANGED TO CHANGE THE FREQUENCY: R6, R7, OR C8. IF YOU WANT TO OPERATE AT TWO FREQUENCIES, FOR EXAMPLE A HIGHER ONE, AND THE STANDARD, IT IS FEASIBLE TO INSTALL A SWITCH FOR THE ABOVE MENTIONED COMPONENTS. FOLLOWING WOULD BE A REASONABLE PROCEDURE FOR EXPERIMENTING WITH THE HIGHER RATES:

1. REDUCE THE VALUE OF R6, R7, OR C8 BY ABOUT THE AMOUNT YOU WANT TO INCREASE THE FREQUENCY.

2. USE THE CASSETTE OUTPUT ROUTINE AND A STOP WATCH TO VERIFY THAT THE INCREASE IN OUTPUT SPEED HAS BEEN ATTAINED.

3. USE THE SYNC GENERATOR PROGRAM TO MAKE A TAPE WITH A LONG STREAM OF SYNC BYTES AT THE NEW FREQUENCY.

4. PLAY IT BACK, ADJUSTING THE POT FOR THE LED TO COME ON. CLOSER ADJUSTMENT OF THE CASSETTE VOLUME CONTROL MAY ALSO BE NECESSARY.

5. THE .1 CAPACITOR MODIFICATION MENTIONED IN ITEM 3 ABOVE WILL ALSO HAVE TO BE REDUCED ACCORDINGLY.

SOLDERING, CLEANING, AND INSTALLATION NOTES

SOLDERING:

Be sure to use good resin-core solder. Acid-core solder will corrode. Use a small soldering element, preferably about 27 watts. Keep your tip clean by wiping on a sponge. Apply heat to the joint first, then solder, then remove solder, then remove the heat (soldering iron). Don't leave the heat applied to the connection more than a few seconds at a time. Some of the components can be destroyed by too much heat, especially the integrated circuits (IC's). Be sure there is a smooth flow of solder over the complete connection, and that the joint looks shiny.

CLEANING:

After you finish soldering, there will be many small conductive particles on the board which you cannot always see. Take a small pointed instrument of some sort, such as a jeweler's screwdriver, and scrape between all printed wiring which is close together, such as those leading to the IC pins. This may take some time, but it is well worth it. Then scrub the bottom (side B) of the board with alcohol. Then visually inspect the board under a strong light, and again remove any dangerous looking particles.

INSTALLATION:

First set the dip-switches to the following positions: = 6E

1-off, 2-off, 3-on, 4-off, 5-off, 6-on, 7-off (input phase inversion)

Switches 1 through 6 correspond to address bits 2 through 7 respectively, and off is a one, on is a zero. Address bit 1 can be either way, as it is ignored by the present interface. Address bit 0 is zero for status/control, and one for data. Therefore, the switch settings above correspond to device address 011011XX (most significant bit first), where X indicates bits that can be either way. This is the device select code that is used in all software for the cassette interface that is supplied by Tarbell Electronics.

Then insert the board into the 100-pin socket, being sure that the component side of the board is to the right (Altair) or front (IMSAI) as viewed from the front of the computer. Then install the ribbon cable between the dipconnector on the interface board and the 25-pin connector slot in the rear. Then install the coax cables between the 25-pin connector and the cassette recorder. The coax cables may also be connected directly between the interface board and the recorder. See the pin function list page for the proper connections.

INITIAL ADJUSTMENT INSTRUCTIONS

THIS INTERFACE WAS DESIGNED TO BE AS EASY AS POSSIBLE TO GET UP AND GOING. THERE ARE, HOWEVER, SOME INITIAL ADJUSTMENTS TO BE MADE, AFTER WHICH THERE NEED NEVER BE ANY MORE.

1. PUT THE TEST CASSETTE INTO YOUR CASSETTE RECORDER.

2. IF YOUR RECORDER HAS A TONE CONTROL, TURN IT TO THE MAXIMUM (BEST HIGH-FREQUENCY RESPONSE) POSITION.

3. TURN YOUR VOLUME CONTROL TO A MIDDLE POSITION.

4. TURN THE POTENTIOMETER ON THE INTERFACE TO A MIDDLE POSITION.

5. PRESS THE "PLAY" BUTTON ON YOUR RECORDER.

6. IF THE LED (RED LIGHT) ON THE INTERFACE DOES NOT COME ON AFTER A FEW SECONDS, ADJUST YOUR VOLUME AND THE INTERFACE POTENTIOMETER UNTIL THE LIGHT COMES ON.

7. IF THE LED STILL DOESN'T COME ON, CHANGE SWITCH #7 (INPUT PHASE REVERSAL) ON THE DIP-SWITCH TO THE OPPOSITE POSITION, THEN REPEAT STEP 6.

8. IF THE LED STILL DOESN'T COME ON AFTER ADJUSTING YOUR VOLUME AND THE INTERFACE POT, SOMETHING IS WRONG WITH YOUR RECEIVER SECTION.

9. WHEN THE LED COMES ONE, THIS INDICATES THAT THE RECEIVER IS OPERATING PROPERLY, AND IS DETECTING THE CONTINUOUS STREAM OF SYNC BYTES WHICH IS ON THE TEST TAPE. FURTHER ADJUST BOTH THE VOLUME CONTROL AND THE INTERFACE POT SO THAT YOU CAN TURN EACH OF THEM FROM SIDE TO SIDE A LITTLE WITHOUT THE LIGHT GOING OUT. THE LIGHT SHOULD BE VERY STABLE, WITH NO FLICKER.

NOTE: SINCE THE LED ONLY STAYS ON WHEN IT IS DETECTING CONTINUOUS SYNC BYTES, IN NORMAL OPERATION (WITH REAL DATA) IT WILL JUST FLICKER ONCE IN A WHILE.

1. RUN THE SYNC CODE GENERATOR PROGRAM WITH YOUR RECORDER IN RECORD MODE, ONTO A BLANK TAPE.

2. TRY THE PROCEDURE ABOVE. IF THE LIGHT DOES NOT COME ON CONTINUOUSLY, YOU MAY BE RECORDING AT TOO HIGH OR TOO LOW A LEVEL. TRY DIFFERENT LEVELS UNTIL YOU FIND THE BEST PLACE. YOU ALSO MAY BE RECORDING IN THE OPPOSITE PHASE. IF SO, CHANGE THE JUMPER FROM PIN 9 TO PIN 8 ON IC 23.

NOTE: IF YOU HAVE AN OSCILLOSCOPE, IC 4 PIN 11 SHOULD SHOW A NICE CLEAN WAVE FORM, WITH ABOUT 25% DUTY CYCLE. ALTHOUGH THERE MAY BE LONG-TERM JITTER, BECAUSE OF THE FLUTTER AND WOW ON THE CASSETTE RECORDER, FAST JITTER ON THE EDGES OF THE WAVE-FORM SHOULD BE FAIRLY SMALL. THE LESS THIS HIGH-SPEED JITTER IS, THE MORE TOLERANCE YOUR INTERFACE WILL HAVE TO TAPE SPEED VARIATIONS.

CASSETTE INTERFACE OPERATING INSTRUCTIONS

THESE INSTRUCTIONS PERTAIN TO OPERATING THE INTERFACE WITH AN ORDINARY AUDIO CASSETTE RECORDER, AND ASSUME THAT THE PROPER SOFTWARE (PROGRAMS, SUBROUTINES) IS IN THE COMPUTER TO COMMUNICATE WITH THE INTERFACE.

TO PERFORM AN OUTPUT (SAVE, WRITE) OPERATION:

1. IF YOUR VOLUME CONTROL HAS AN EFFECT ON THE RECORDING FUNCTION, FIRST TURN IT ALL THE WAY DOWN.

2. GET TO THE POINT IN YOUR PROGRAM WHERE ALL IT TAKES IS A PUSH OF A BUTTON TO START INTO THE CASSETTE OUTPUT ROUTINE.

3. USE FAST-FORWARD OR REWIND TO MOVE TO THE DESIRED LOCATION ON THE CASSETTE TAPE.

4. START YOUR CASSETTE RECORDING.

5. IF YOUR VOLUME CONTROL HAS AN EFFECT DURING RECORD, SLOWLY INCREASE THE VOLUME UNTIL YOUR INDICATOR SHOWS A CORRECT RECORDING LEVEL.

6. WAIT FOR ABOUT 5 SECONDS TO RECORD LEADER.

7. PUSH THE BUTTON THAT STARTS THE OUTPUT ROUTINE ON THE RECORDER. (THIS MIGHT BE THE CARRIAGE-RETURN AFTER "CSAVE" IN BASIC, OR THE FRONT-PANEL "RUN" BUTTON FOR STAND-ALONE PROGRAMS.)

8. WHEN THE PROGRAM INDICATES THAT THE DATA TRANSFER IS COMPLETE, STOP YOUR CASSETTE RECORDER.

TO PERFORM AN INPUT (LOAD, READ) OPERATION:

1. BE SURE YOUR VOLUME CONTROL IS AT THE POSITION THAT YOU LEFT IT IN THE ADJUSTMENT PROCEDURE.

2. GET TO THE POINT IN YOUR PROGRAM WHERE ALL IT TAKES IS A PUSH OF A BUTTON TO START INTO THE CASSETTE INPUT ROUTINE.

3. USE FAST-FORWARD OR REWIND TO MOVE TO THE DESIRED LOCATION ON THE CASSETTE TAPE. THIS SHOULD BE A FEW SECONDS INTO THE LEADER OF A PREVIOUS RECORDING.

4. START YOUR CASSETTE IN THE PLAYBACK MODE.

5. PRESS THE BUTTON WHICH CAUSES THE INPUT ROUTINE TO START RUNNING. (IN BASIC, THIS MIGHT BE THE CARRIAGE-RETURN AFTER A CLOAD, OR THE FRONT-PANEL "RUN" BUTTON FOR STAND-ALONE PROGRAMS, SUCH AS BOOTSTRAPS.)

6. WHEN THE PROGRAM INDICATES THAT THE DATA TRANSFER IS COMPLETE, STOP YOUR CASSETTE RECORDER.

NOTE: ALWAYS BE SURE THAT ALL MEMORY INTO WHICH PROGRAMS OR DATA IS TO BE READ, IS UNPROTECTED FIRST.

CASSETTE BOOTSTRAP PROGRAM

THIS PROGRAM LOADS DATA STARTING AT ZERO AND KEEPS ON GOING. THERE IS NO COUNT OF BYTES, AND NO CHECKSUM. IT IS ASSEMBLED TO RUN AT 2F00 (HEX), BUT MAY BE ASSEMBLED TO RUN ANYWHERE, PROVIDED THAT IT DOES NOT LOAD DATA OVER ITSELF. USE THE SECOND HAND ON YOUR WATCH TO DETERMINE HOW LONG TO WAIT UNTIL STOPPING THIS PROGRAM. ALLOW ABOUT 45 SECONDS TO LOAD AN 8 KBYTE BLOCK.

```
2F00  3E 10             MVI   A,10H     SET BIT 4 OF A = 1.
2F02  D3 6E             OUT   CASC      RESET INTERFACE.
2F04  21 00 00          LXI   H,0       PUT STARTING ADDRESS IN H,L.
2F07  DB 6E       LOOP  IN    CASC      READ STATUS.
2F09  E6 10             ANI   10H       CLEAR ALL BUT BIT 4.
2F0B  C2 07 2F          JNZ   LOOP      WAIT IN LOOP UNTIL READY.
2F0E  DB 6F             IN    CASD      READ A DATA BYTE.
2F10  FB                EI              SIGNAL OPERATOR.
2F11  77                MOV   M,A       PUT DATA INTO MEMORY.
2F12  23                INX   H         INCREMENT MEMORY POINTER.
2F13  C3 07 2F          JMP   LOOP      REPEAT THE ABOVE OPERATION.
                  CASC  EQU   6EH       CASSETTE STATUS PORT.
                  CASD  EQU   6FH       CASSETTE DATA PORT.
                        END
```

NOTE: IF YOU HAVE AN IMSAI OR ALTAIR WITH AN OUTPUT PORT ON THE FRONT PANEL (8 LED'S), YOU CAN USE THE BOOTSTRAP PROGRAM FOR TROUBLESHOOTING THE INPUT SECTION WITH THE FOLLOWING MODIFICATION:

AT	INSTEAD OF	SUBSTITUTE
2F10	EI (FB)	CMA (2F)
2F11	MOV M,A (77)	OUT (D3)
2F12	INX H (23)	LEDS (FF)

SYNC CODE GENERATOR PROGRAM

THIS PROGRAM MAY BE USED TO GENERATE A CONTINUOUS STREAM OF E6 (HEX), THE SAME AS IS ON THE SUPPLIED CASSETTE. IF YOU FIND THAT THE RECEIVED STREAM IS INVERTED FROM THE ONE SUPPLIED, YOU MAY CHANGE THE CIRCUIT SO THAT IC23-8 IS HOOKED TO THE 74L86 INSTEAD OF IC23-9. THIS WILL MAKE YOUR RECORDINGS THE SAME PHASE AS MINE.

```
0000  DB 6E       LOOP  IN    CASC      READ STATUS.
0002  E6 20             ANI   20H       LOOK AT BIT 5.
0004  C2 00 00          JNZ   LOOP      WAIT UNTIL READY.
0007  3E E6             MVI   A,0E6H    GET SYNC BYTE.
0009  D3 6F             OUT   CASD      WRITE IT ONTO CASSETTE.
000B  C3 00 00          JMP   LOOP      REPEAT.
                  CASC  EQU   6EH       STATUS PORT.
                  CASD  EQU   6FH       DATA PORT.
                        END
```

INPUT ADDR FROM CAS
OUTPUT ADDR TO CAS
SYNC BYTE

CASSETTE INTERFACE OUTPUT ROUTINE

THIS PROGRAM WRITES A BLOCK OF MEMORY OUT ONTO CASSETTE TAPE. THE PROGRAM IS ASSEMBLED TO START AT 3100 (HEX), BUT MAY BE REASSEMBLED TO START ANYWHERE. THE BLOCK STARTING ADDRESS IS LOCATED AT ADDRESS 3104 (HEX). THE BLOCK LENGTH (2 BYTES) IS LOCATED AT ADDRESS 3107 (HEX). THE PROGRAM WILL WRITE A "W" ON THE COMMENT DEVICE WHEN IT IS THROUGH WITH IT'S DATA TRANSFER.

```
3100  31 43 31       LXI   SP,STAK     SET STACK POINTER.
3103  21 00 00       LXI   H,0         GET BLOCK ADDRESS.
3106  01 00 20       LXI   B,2000H     SET BLOCK LENGTH = 8192.
3109  1E 00          MVI   E,0         SET E=0.
310B  3E 3C          MVI   A,3CH       GET START BYTE.
310D  CD 32 31       CALL  COUT        OUTPUT START BYTE TO CASSETTE.
3110  3E E6          MVI   A,0E6H      GET SYNC BYTE.
3112  CD 32 31       CALL  COUT        OUTPUT SYNC BYTE TO CASSETTE.
3115  7E        LOOP MOV   A,M         GET A DATA BYTE FROM MEMORY.
3116  CD 32 31       CALL  COUT        OUTPUT DATA BYTE TO CASSETTE.
3119  83             ADD   E           ADD E (CHECKSUM) TO A.
311A  5F             MOV   E,A         PUT NEW CHECKSUM INTO E.
311B  23             INX   H           INCREMENT MEMORY POINTER.
311C  0B             DCX   B           DECREMENT COUNTER.
311D  3E 00          MVI   A,0         MAKE A=0.
311F  B8             CMP   B           IF B NOT = 0,
3120  C2 15 31       JNZ   LOOP          REPEAT LOOP.
3123  B9             CMP   C           IF C NOT = 0,
3124  C2 15 31       JNZ   LOOP          REPEAT LOOP.
3127  7B             MOV   A,E         OTHERWISE, GET CHECKSUM
3128  CD 32 31       CALL  COUT        AND OUTPUT IT.
312B  3E 57          MVI   A,"W"       WRITE "W" (END OF WRITE).
312D  D3 01          OUT   1           PRINT ON CONSOLE.
312F  C3 2F 31  WAIT JMP   WAIT        WAIT HERE WHEN DONE.
3132  F5        COUT PUSH  PSW         SAVE A AND FLAGS.
3133  DB 6E     CLOP IN    CASC        READ CASSETTE STATUS.
3135  E6 20          ANI   20H         CLEAR ALL BUT BIT 5.
3137  C2 33 31       JNZ   CLOP        TRY AGAIN IF NOT READY.
313A  F1             POP   PSW         RESTORE A AND FLAGS.
313B  D3 6F          OUT   CASD        OUTPUT DATA TO CASSETTE.
313D  C9             RET               RETURN FROM COUT.
313E  00             0
313F  00             0
3140  00             0
3141  00             0
3142  00             0
3143  00        STAK 0
                CASD EQU   6FH
```

CASSETTE INTERFACE INPUT ROUTINE

THIS PROGRAM READS A BLOCK OF BYTES FROM CASSETTE INTO MEMORY. THE PROGRAM IS ASSEMBLED TO START AT 3100 (HEX), BUT MAY BE REASSEMBLED TO START ANYWHERE, ALTHOUGH CARE SHOULD BE TAKEN TO INSURE THAT THE DATA IT IS READING DOES NOT WRITE OVER THE PROGRAM ITSELF. THIS MAY BE ACCOMPLISHED BY LOCATING THE PROGRAM IMMEDIATELY BELOW OR A BLOCK LENGTH ABOVE THE DATA TO BE READ IN. THE STARTING ADDRESS FOR THE BLOCK IS LOCATED IN ADDRESS 3185 (HEX). THE BLOCK LENGTH IS LOCATED IN ADDRESS 3188 (HEX) (TWO BYTES).

```
3180  3E 10            MVI  A,10H      SET BIT 4 OF A=1.
3182  D3 6E            OUT  CASC       RESET INTERFACE.
3184  21 00 00         LXI  H,0        GET STARTING ADDRESS.
3187  11 00 20         LXI  D,2000H    GET BLOCK LENGTH.
318A  06 00            MVI  B,0        SET CHECKSUM = 0.
318C  DB 6E      LOOP  IN   CASC       READ CASSETTE STATUS.
318E  E6 10            ANI  10H        LOOK AT BIT 4.
3190  C2 8C 31         JNZ  LOOP       WAIT IF NOT READY.
3193  DB 6F            IN   CASD       READ DATA FROM CASSETTE.
3195  77               MOV  M,A        PUT DATA INTO MEMORY.
3196  80               ADD  B          ADD CHECKSUM TO A.
3197  47               MOV  B,A        PUT IT BACK IN B.
3198  23               INX  H          INCREMENT MEMORY POINTER.
3199  1B               DCX  D          DECREMENT COUNTER.
319A  3E 00            MVI  A,0        CLEAR A.
319C  BA               CMP  D          IF D NOT = 0,
319D  C2 8C 31         JNZ  LOOP         READ MORE.
31A0  BB               CMP  E          IF E NOT = 0,
31A1  C2 8C 31         JNZ  LOOP         READ MORE.
31A4  DB 6E      CHEK  IN   CASC       READ STATUS.
31A6  E6 10            ANI  10H        LOOK AT BIT 4.
31A8  C2 A4 31         JNZ  CHEK       WAIT IF NOT READY.
31AB  DB 6F            IN   CASD       READ CHECKSUM.
31AD  B8               CMP  B          COMPARE TO B.
31AE  3E 45            MVI  A,"E"      PUT CODE FOR "E" IN A.
31B0  C2 B5 31         JNZ  ERR        IF CHECKSUMS NOT EQUAL, ERROR.
31B3  C6 02            ADI  2          ADD A 2 TO MAKE "G" IF EQUAL.
31B5  D3 01      ERR   OUT  CRTD       PRINT "E" FOR "G".
31B7  C3 B7 31   END   JMP  END        WAIT HERE WHEN DONE.
                 CASC  EQU  6EH        CASSETTE STATUS/CONTROL PORT.
                 CASD  EQU  6FH        CASSETTE DATA PORT.
                 CRTD  EQU  01H        CONSOLE DATA PORT.
```

BASIC STORE & LOAD

```
1 REM   THIS PRØGRAM SHØWS HØW TØ SAVE DATA ØNTØ A CASSETTE
2 REM   AND LØAD IT BACK INTØ MEMØRY FRØM A PRØGRAM RUNNING
3 REM   UNDER 8K BASIC 3.1.  IT ALLØWS YØU TØ ENTER LINES ØF
4 REM   TEXT FRØM THE CØNSØLE KEYBØARD, SAVE THEM ØN CASSETTE,
5 REM   LØAD THEM BACK INTØ MEMØRY, AND PRINT THEM.
10 CLEAR 3000:L=50:DIM A$(50):REM RESERVE FØR UP TØ 50 LINES.
20 CC=110:CD=111:REM  CASSETTE CØNTRØL AND DATA PØRT NUMBERS.
30 T$="%":REM  END-ØF-FILE CHARACTER.
40 D=100
100 INPUT "CØMMAND";C$
110 IF C$="ENTER" THEN 1000
120 IF C$="PRINT" THEN 2000
130 IF C$="SAVE" THEN 3000
140 IF C$="GET" THEN 4000
900 PRINT "INVALID CØMMAND.":GØTØ 100
1000 REM  ENTER TEXT FRØM THE CØNSØLE KEYBØARD
1020 FØR N=1 TØ L:REM  ENTER A MAXIMUM ØF L LINES.
1030 INPUT B$:REM  READ A LINE FRØM KEYBØARD.
1040 IF B$="%" THEN 1070:REM  A % TERMINATES THE INPUT.
1050 A$(N)=B$:NEXT N:REM PUT LINE INTØ BUFFER.
1070 N=N-1:GØTØ 100:REM  N=THE NUMBER ØF LINES ENTERED.
2000 REM  PRINT THE BUFFER AREA ØN THE CØNSØLE.
2010 FØR I=1 TØ N:PRINT A$(I):NEXT I:GØTØ 100
3000 REM  SAVE THE BUFFER ØNTØ CASSETTE TAPE.
3010 S$=CHR$(195)+CHR$(230):REM  S$=START & SYNC BYTES.
3019 REM  CHANGE THE CØNSØLE ØUTPUT RØUTINE FØR CASSETTE.
3020 PØKE 1230,CC:PØKE 1232,32:PØKE 1238,CD
3030 FØR I=1 TØ N
3040 FØR K=1 TØ D:NEXT K:REM  DELAY FØR CØUNT ØF D.
3050 B$=S$+A$(I):REM  HØØK START & SYNC BYTES TØ LINE.
3060 PRINT B$:REM  WRITE LINE ØNTØ CASSETTE.
3070 NEXT I
3080 B$=S$+T$:REM  HØØK START & SYNC BYTES TØ TERMINATØR.
3085 FØR K=1 TØ D:NEXT K:REM  DELAY FØR CØUNT ØF D.
3090 PRINT B$:REM  WRITE THE END-ØF-FILE MARK.
3099 REM  CHANGE CØNSØLE RØUTINE BACK TØ NØRMAL.
3100 PØKE 1230,0:PØKE 1232,128:PØKE 1238,1
3110 GØTØ 100
4000 REM  GET TEXT FRØM CASSETTE AND PUT INTØ BUFFER.
4010 REM  CHANGE CØNSØLE INPUT RØUTINE FØR CASSETTE.
4011 PØKE 1241,CC:PØKE 1243,16:PØKE 1248,CD:PØKE 1232,0
4012 PØKE 1238,255
4020 FØR I=1 TØ L
4030 ØUT CC,16:REM  RESET CASSETTE INPUT SECTIØN.
4040 INPUT B$:REM  READ A LINE ØF TEXT FRØM CASSETTE.
4050 IF B$=T$ THEN 4080
4060 A$(I)=B$
4070 NEXT I
4080 N=I-1
4090 REM  CHANGE CØNSØLE RØUTINE BACK TØ NØRMAL.
4091 PØKE 1241,0:PØKE 1243,1:PØKE 1248,1:PØKE 1232,128
4092 PØKE 1238,1
4100 GØTØ 100
ØK
```

SAVING AND LOADING DATA TO AND FROM CASSETTE FROM BASIC PROGRAMS

It is quite often desireable to save and load data to and from cassette, while running a BASIC program. For example, you might have a nice inventory program running in BASIC, but it's of limited utility if there is no way to save the inventory on cassette for overnight storage. You can leave the computer running, but if there is an interruption in power, or a computer failure, your inventory is lost. Other applications include accounts receivable, mailing lists, and payroll.

There are several ways of handling this function, depending on the version of BASIC you use. Some versions of BASIC have a command which can save and load a numerical array to and from cassette. ALTAIR* DISK BASIC is one of these. The only problem with this way, is that DISK BASIC is fairly large, and that strings have to be converted to numerical arrays. Another way is described explicitly by a program on page 15 of the Tarbell Cassette Interface Manual. In this method, the console (TTY, CRT) routines are modified by POKE commands, so that they are temporarily cassette I/O routines. Then the PRINT and INPUT statements may be used to transfer the data. After the data is transferred, POKE statements restore the console routines to their original form. The disadvantage here is that different versions of BASIC have their console I/O routines in different places, so the program has to be adapted when changing from one version to another.

Another possibility is to POKE the data into an unused area of memory. The USR function then is used to transfer to your own output routine (possibly an adaptation from one in the manual). This routine only needs to transfer a block of memory onto tape. The USR function can be used with a different argument to run an input routine, then the BASIC program can retrieve the data with the PEEK function. The main problem with this method is that the I/O routines generally have to be loaded seperately, and are not yet written and tested.

Still another way is to use a general purpose monitor program. This program would also be loaded seperately, but might reside in read-only-memory (ROM). The monitor would handle all input/output functions including console and cassette I/O. One of the commands in the monitor is to assign different functions to different I/O devices than normally handle these functions. For example, the console function could be assigned to the cassette interface instead of the TTY. Then PRINT and INPUT statements could be used for I/O to and from cassette.

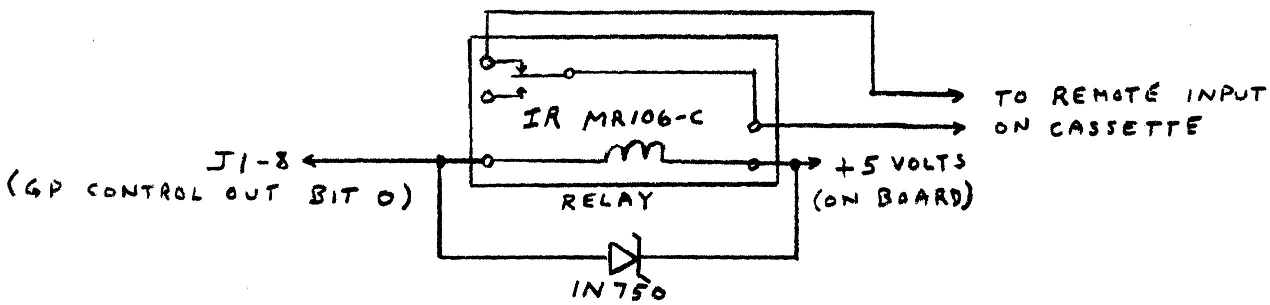
The monitor described above is presently under development by Tarbell Electronics. In addition to the Assign command, it has commands for dumping and loading memory, checking records for errors, and moving data from one area in memory to another.

CONTROLLING THE START-STOP (REMOTE) FUNCTION ON YOUR CASSETTE

MOST CASSETTE RECORDERS HAVE A REMOTE CONTROL INPUT, MEANT FOR CONTROL FROM A MICROPHONE SWITCH. THIS FACILITY CAN BE OPERATED BY A COMPUTER PROGRAM TO START AND STOP THE TAPE AUTOMATICALLY, ACCORDING TO THE NEEDS OF THE PROGRAM.
THIS IS PARTICULARLY IMPORTANT IF THE AMOUNT OF DATA ON TAPE IS MORE THAN WILL FIT INTO MAIN MEMORY ALL AT ONCE. IN THIS CASE, THE DATA MAY BE "BLOCKED"; THAT IS, GAPS MAY BE INSERTED BETWEEN BLOCKS OF DATA, WHICH ALLOW TIME FOR THE TAPE TO START AND STOP. THE PROGRAM MAY THEN START THE TAPE, READ SOME DATA, STOP THE TAPE, WORK ON THE DATA, START ANOTHER TAPE, WRITE SOME DATA, AND STOP THE TAPE. THIS PROCESS MAY BE REPEATED UNTIL ALL THE DATA IS PROCESSED. SOME EXAMPLES WHERE THIS OPERATION MIGHT BE NECESSARY ARE AS FOLLOWS:

1) AN ASSEMBLER, WHERE THE SOURCE IS LARGER THAN MEMORY, MAY READ SOURCE FROM ONE TAPE AND WRITE MACHINE CODE TO ANOTHER.
2) A COMPILER, IN THE SAME SITUATION.
3) A MERGING PROGRAM, WHERE AN OLD FILE IS UPDATED WITH CHANGES TO FORM A NEW FILE.

THE CIRCUIT FOR CONTROLLING ONE TAPE UNIT IS SHOWN BELOW:



THESE ROUTINES MAY BE USED FOR STARTING AND STOPPING THE CASSETTE BEFORE AND AFTER INPUT AND OUTPUT OPERATIONS:

```
START   LDA   CTLS   GET CONTROL STATUS BYTE.
        ORI   01     SET BIT 0 = ONE.
        STA   CTLS   UPDATE CONTROL STATUS BYTE.
        OUT   CASC    START THE TAPE.
        CALL  DELAY   WAIT FOR TAPE TO GET UP TO SPEED.
        RET           RETURN (NEXT DO YOUR I/O).

STOP    LDA   CTLS    GET CONTROL STATUS BYTE.
        AND   OFEH    SET BIT 0 TO ZERO.
        STA   CTLS    UPDATE CONTROL STATUS BYTE.
        OUT   CASC    STOP THE TAPE.
        RET           RETURN FROM I/O ROUTINE.

CTLS    DB    0       CONTROL STATUS BYTE.
CASC    EQU   6EH     CASSETTE CONTROL PORT.
```

THE START-UP DELAY IS DETERMINED BY YOUR RECORDER, AND SHOULD BE LONGER BEFORE A WRITE THAN BEFORE A READ OPERATION.

NOTE: A MODULE WHICH ALLOWS THE CONTROL OF UP TO FOUR CASSETTE RECORDERS WITH A TARBELL CASSETTE INTERFACE IS AVAILABLE FROM: RO-CHE SYSTEMS, 7101 MAMMOTH AVE, VAN NUYS, CA 91405.

THEORY OF CASSETTE INTERFACE OPERATION

OUTPUT SECTION

THE PURPOSE OF THIS SECTION IS TO CONVERT 8-BIT PARALLEL BYTES FROM THE COMPUTER TO A SERIAL BI-PHASE ENCODED DATA STREAM FOR THE RECORDER. THE NE555 (IC 22) IS CONNECTED AS AN OSCILLATOR TO OSCILLATE AT TWICE THE FREQUENCY OF THE REQUIRED CLOCK RATE. FOR 800 BITS PER INCH, THE CLOCK RATE NEEDS TO BE 1500 HZ FOR A RECORDER RUNNING AT 1 7/8 INCHES PER SECOND. SO IC 22 RUNS AT 3000 HZ, DETERMINED BY THE MATCHED SET OF COMPONENTS (IC 22, C8, R6, AND R7). THE OUTPUT ON PIN 3 IS FED TO A J-K FLIP-FLOP, WHICH DIVIDES THE FREQUENCY BY TWO. THE MAIN FUNCTION OF THIS FLIP-FLOP IS TO MAKE THE WAVE-FORM PERFECTLY SYMMETRICAL. THIS SQUARE-WAVE IS THEN FED TO AN INPUT OF THE EXCLUSIVE-OR GATE AT IC 30 PIN 6.

IC 32, THE DM8131, IS A 6-BIT DIGITAL COMPARATOR. IT'S PURPOSE IS TO COMPARE THE ADDRESS ON BITS 2 THROUGH 7 OF THE ADDRESS BUS WITH THE SETTING ON THE DIP SWITCH. WHEN THEY MATCH, THE OUTPUT AT PIN 9 GOES LOW, INDICATING THAT THIS DEVICE IS BEING SELECTED. THIS SIGNAL IS INVERTED AND ANDED WITH THE WRITE SIGNAL AND THE STATUS OUTPUT SIGNAL ON THE BUS. THIS PRODUCES A HIGH OUTPUT AT IC 14 PIN 6 WHEN THIS DEVICE IS BEING WRITTEN TO.

THIS SIGNAL IS IN TURN ANDED WITH ADDRESS BIT 0 TO STROBE DATA INTO THE SHIFT REGISTER (IC 26 PIN 6 AND IC 20 PINS 4 AND 2). THE DATA IS SHIFTED OUT OF THE SHIFT REGISTER BY THE CLOCK PREVIOUSLY MENTIONED. THE SHIFT REGISTER IS MADE UP OF IC'S 21, 27, AND 23. DATA ENTERS AT THE BOTTOM AND IS SHIFTED OUT TOWARD THE RIGHT, WHERE IT IS COMBINED WITH THE CLOCK TO FORM THE BI-PHASE SIGNAL FOR THE RECORDER.

THE FIRST EIGHT OUTPUTS OF THE SHIFT REGISTER ARE FED INTO IC 28. SINCE A ZERO IS BEING FED INTO THE LEFT END OF THE SHIFT REGISTER FOR EACH SHIFT (IC 21 PIN 9), THESE LEFT EIGHT BITS WILL ALL BECOME ZERO AFTER 8 SHIFTS. AT THIS TIME THE OUTPUTS AT IC 28 PINS 6 AND 8 WILL BOTH BE HIGH, CAUSING THE OUTPUT AT IC 14 PIN 8 TO GO HIGH. THIS SIGNAL WILL BE USED BY THE COMPUTER TO DETERMINE WHEN THE INTERFACE NEEDS MORE OUTPUT DATA.

WHEN ADDRESS BIT 0 IS LOW, THE OUTPUT AT IC 14 PIN 6 ANDED WITH IC 31 PIN 2 CAUSES A HIGH OUTPUT AT IC 35 PIN 10. THIS STROBES BITS 0, 1, 2, AND 3 ON THE DATA BUS INTO THE LATCH OF IC 34. THE OUTPUTS OF THIS LATCH ARE BUFFERED WITH IC 35, AND APPEAR AT PINS 2, 4, 6, AND 8. THESE ARE THEN FED TO THE DIP-SOCKET. IF DATA BUS BIT 4 IS HIGH, A PULSE APPEARS AT PIN 8 OF IC 26. THIS IS USED TO RESET THE COUNTER IN THE INPUT SECTION.

THEORY OF OPERATION

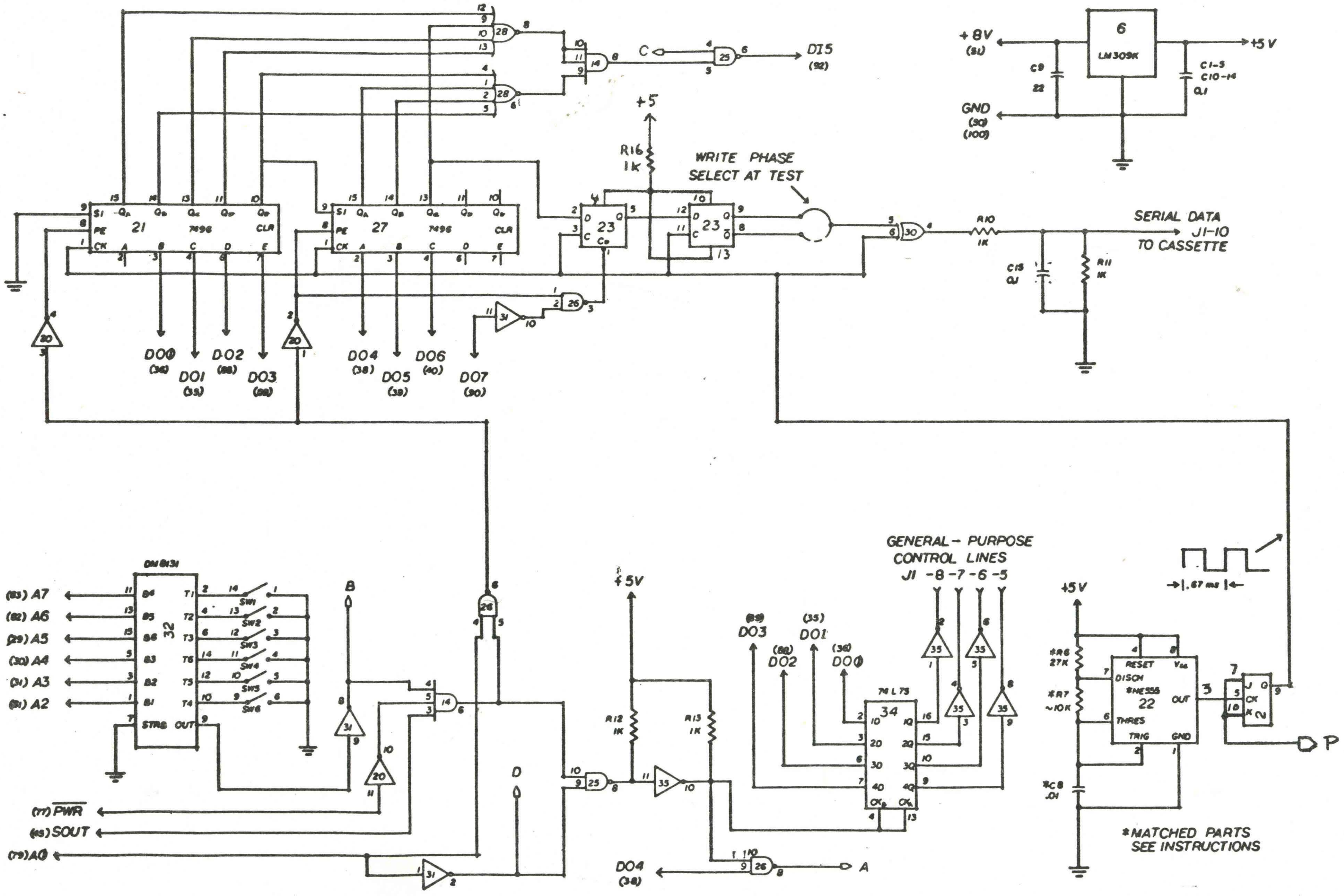
INPUT SECTION

THE PURPOSE OF THIS SECTION IS TO CONVERT THE BI-PHASE AUDIO SIGNAL COMING FROM THE CASSETTE RECORDER TO 8-BIT PARALLEL BYTES FOR THE COMPUTER. THE AUDIO SIGNAL COMING DIRECTLY FROM THE CASSETTE EARPHONE OR SPEAKER OUTPUT IS TERMINATD BY THE 100 OHM RESISTOR AND FED THROUGH THE .02 CAPACITOR TO THE INPUT OF THE 8T20 (IC 5 PINS 6 AND 7). THIS INPUT HAS A BUILT-IN VOLTAGE-DIVIDER, WHICH BIASES THE DC LEVEL TO A GOOD MIDWAY TTL REFERENCE VOLTAGE BETWEEN ONE AND ZERO. THIS IS ONE INPUT OF A HIGH-SPEED COMPARATOR. THE OTHER INPUT (IC 5 PIN 5) IS CONNECTED THROUGH A RESISTOR DIVIDER AND A 1N914 DIODE TO ONE OF THE COMPARATOR OUTPUTS (IC 5 PIN 9). THIS PROVIDES A SMALL AMOUNT OF HYSTERESIS TO COMBAT NOISE PROBLEMS. THE OTHER OUTPUT OF THE COMPARITOR (IC 5 PIN 1) IS EXCLUSIVE-ORED WITH SWITCH 7 ON THE DIP-SWITCH TO PROVIDE A WAY TO INVERT THE INPUT DATA STREAM. THIS IS THEN FED TO A D-TYPE FLIP-FLOP (IC 4 PIN 12).

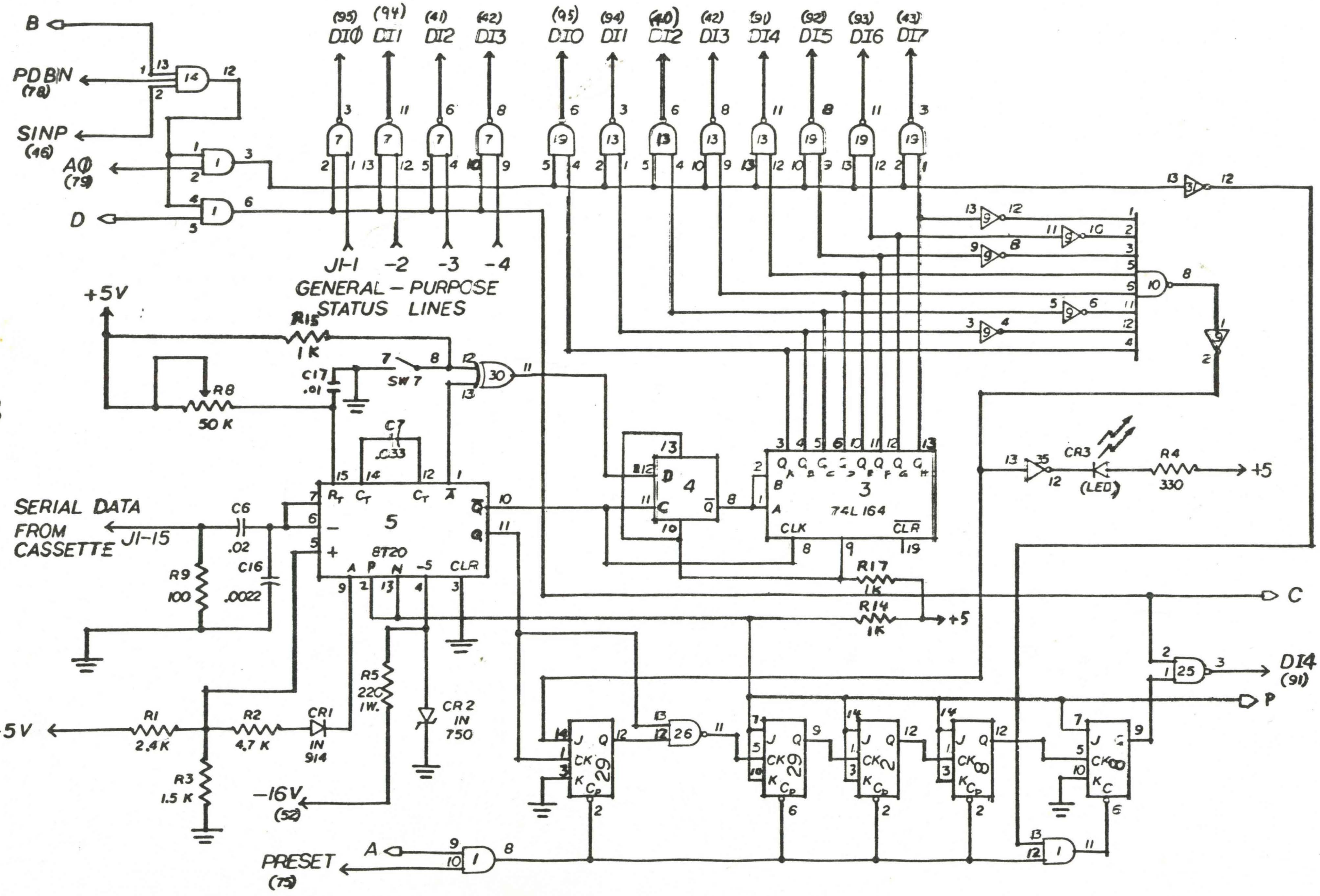
MEANWHILE, INSIDE IC 5, THE OUTPUTS OF THE COMPARITOR SECTION ARE FED TO AN EDGE-DETECTOR, WHICH DETECTS BOTH POSITIVE AND NEGATIVE-GOING TRANSITIONS. THE OUTPUT OF THIS DETECTOR IS THEN USED TO TRIGGER A STABLE NON-RETRIGGERABLE ONE-SHOT. THE CAPACITOR FOR THIS ONE-SHOT IS BETWEEN PINS 12 AND 14 OF IC 5, AND THE RESISTOR IS THE 50 KOHM POTENTIOMETER. THE OUTPUT OF THE ONE-SHOT IS THE RECOVERED CLOCK (IC 5 PINS 10 AND 11). THIS IS FED TO THREE DIFFERENT PLACES: 1) IC 4 PIN 11, WHERE IT TRIGGERS THE FLIP-FLOP TO RECOVER THE SERIAL DATA STREAM, 2) IC 3 PIN 8 WHERE IT IS USED TO SHIFT THE SERIAL-PARALLEL SHIFT-REGISTER, 3) IC 29 PIN 1 AND IC 26 PIN 13, WHERE IT IS USED TO STEP THE 8 COUNTER.

IN A START-UP POSITION, IC 29 PIN 12 IS ZERO, HAVING BEEN RESET EITHER BY THE RESET SWITCH OR BY A RESET COMMAND FROM THE PROGRAM. THIS STOPS THE CLOCK FROM TRIGGERING THE 8 COUNTER, WHICH HAS ALSO BEEN RESET. AS THE SERIAL STREAM FLOWS THROUGH THE SHIFT-REGISTER (IC 3), IT IS CONTINUALLY INSPECTED BY THE SYNC DECODER MADE UP OF IC'S 9 AND 10. WHENEVER A SYNC CODE APPEARS IN THE SHIFT REGISTER, IC 10 PIN 8 GOES LOW. THIS LIGHTS THE SYNC LED AND ALSO ALLOWS THE FLIP-FLOP AT IC 29 PIN 12 TO GO HIGH. THIS ALLOWS THE CLOCK TO APPEAR AT IC 26 PIN 11, AND TRIGGER THE 8 COUNTER. WHEN THE COUNTER HAS COUNTED TO 8, IC 8 PIN 12 GOES HIGH, TRIGGERING THE READY FLIP-FLOP AT IC 8 PIN 9. THIS READY CONDITION INDICATES TO THE COMPUTER THAT THERE IS A BYTE IN THE SHIFT REGISTER READY TO READ. THE COMPUTER MAY THEN READ THIS BYTE THROUGH GATES OF IC'S 13 AND 19.

IN ORDER TO READ DATA THE SIGNAL AT POINT B, WHICH COMES FROM THE OUTPUT SECTION IS ANDED WITH THE TWO INPUT GATE SIGNALS PDBIN AND SINP (IC 14 PINS 13, 1 AND 2). WHEN ADDRESS BIT 0 IS HIGH, THIS SIGNAL IS ANDED WITH IT TO GATE DATA FROM THE SHIFT REGISTER ONTO THE INPUT DATA BUS (IC 1 PIN 3). WHEN ADDRESS BIT 0 IS LOW, THIS SIGNAL IS ANDED WITH IT TO GATE VARIOUS STATUS BITS ONTO THE INPUT DATA BUS. FOUR OF THESE INPUTS ARE GENERAL-PURPOSE, AND COME FROM THE DIP-SOCKET. THE OTHERS ARE PICKED UP AT IC 25 PIN 3 (INPUT STATUS), AND IC 25 PIN 6 (OUTPUT STATUS).

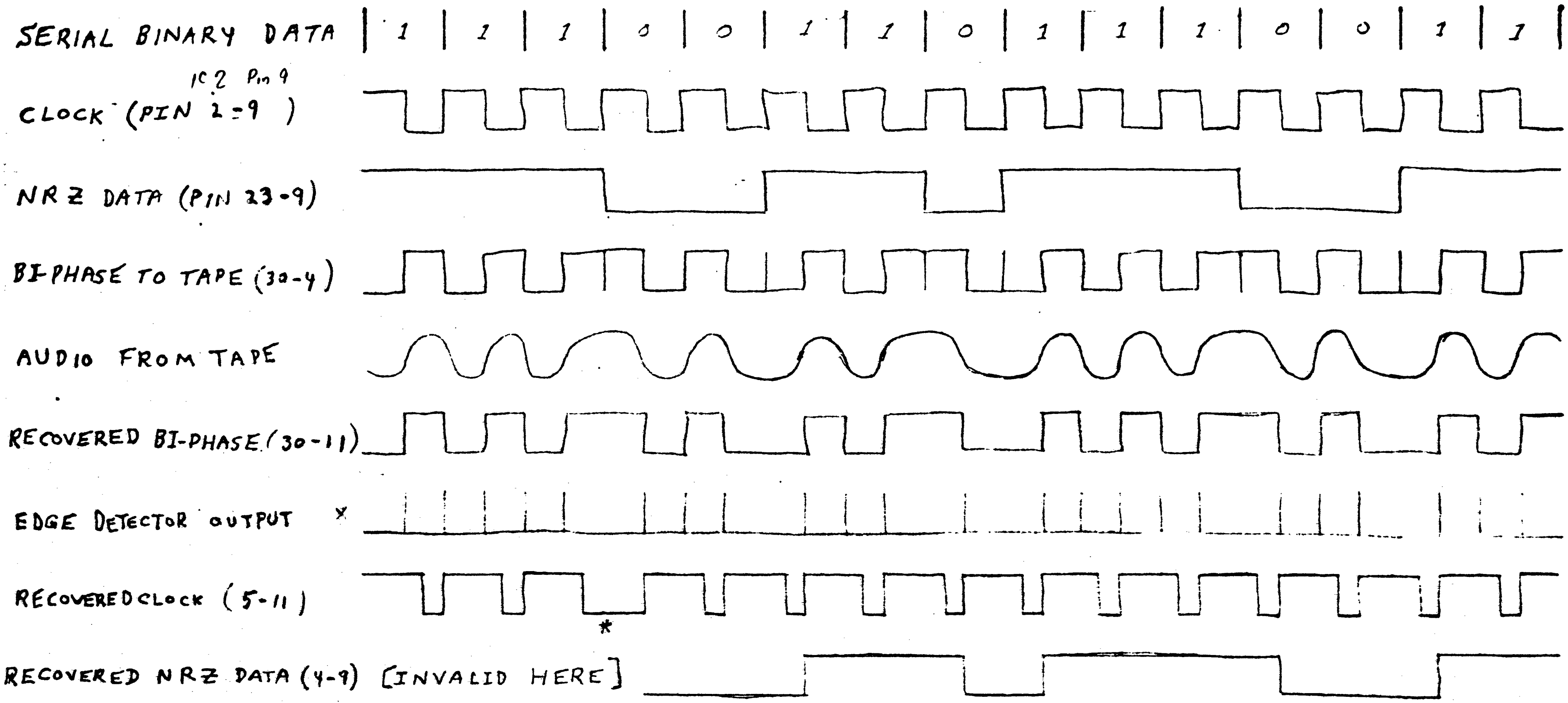


CASSETTE INTERFACE OUTPUT SECTION



CASSETTE INTERFACE INPUT SECTION

CASSETTE INTERFA TIMING DIAGRAM



* CLOCK IS IN PROPER PHASE AFTER FIRST DATA CHANGE.

CASSETTE INTERFACE PIN FUNCTIONS

J1 (DIP SOCKET)

1	GP STATUS IN BIT 0	16	GROUND
2	GP STATUS IN BIT 1	15	DATA FROM CASSETTE
3	GP STATUS IN BIT 2	14	SPARE
4	GP STATUS IN BIT 3	13	SPARE
5	GP CONTROL OUT BIT 3	12	+5 VOLTS
6	GP CONTROL OUT BIT 2	11	SPARE
7	GP CONTROL OUT BIT 1	10	DATA TO CASSETTE
8	GP CONTROL OUT BIT 0	9	GROUND

NOTES: GP STANDS FOR GENERAL-PURPOSE
DATA FROM CASSETTE SHOULD BE CONNECTED TO EARPHONE JACK
DATA TO CASSETTE SHOULD BE CONNECTED TO AUXILLIARY JACK.
IF MIKE JACK IS USED, OR IF THE RECORDER ONLY OPERATES WITH AUTOMATIC VOLUME CONTROL ON RECORD, THEN DIVIDER RESISTORS MAY HAVE TO BE CHANGED FOR THE PROPER RECORDING LEVEL.
THE DIVIDER RESISTORS ARE R10 AND R11.

P2 (8800 BUS PINS)

1	+8V
29	ADDRESS LINE #5
30	ADDRESS LINE #4
31	ADDRESS LINE #3
35	DATA OUT LINE #1
36	DATA OUT LINE #0
38	DATA OUT LINE #4
39	DATA OUT LINE #5
40	DATA OUT LINE #6
41	DATA IN LINE #2
42	DATA IN LINE #3
43	DATA IN LINE #7
45	OUT
46	INP
50	GROUND
51	+8V
52	-16V
75	RESET-NOT
77	WRITE-NOT
78	DATA BUS IN
79	ADDRESS LINE #0
81	ADDRESS LINE #2
82	ADDRESS LINE #6
83	ADDRESS LINE #7
88	DATA OUT LINE #2
89	DATA OUT LINE #3
90	DATA OUT LINE #7
91	DATA IN LINE #4
92	DATA IN LINE #5
93	DATA IN LINE #6
94	DATA IN LINE #1
95	DATA IN LINE #0
100	GROUND

IF YOU CANNOT MAKE AT LEAST TEN 8K-BYTE TRANSFERS WITH NO ERRORS, YOU HAVE A PROBLEM, AND THE ITEMS BELOW MAY BE OF SOME HELP:

1. CHECK TO MAKE SURE THAT ALL THE COMPONENTS AND JUMPERS ARE IN THEIR PROPER LOCATIONS, AND THAT THEY ARE ORIENTED AS SHOWN IN THE ASSEMBLY DRAWING.

2. MAKE SURE THAT THE BOARD IS CLEAN, ESPECIALLY THAT THERE IS NO FLUX RESIDUE BETWEEN IC PINS OR OTHER CLOSE LINES.

3. DEMAGNETIZE AND CLEAN THE RECORD/PLAYBACK HEAD ON YOUR RECORDER.

4. WHEN YOU PLUG IN THE BOARD, BE SURE THAT THE PINS ON THE BOARD EDGE CONNECTOR LINE UP WITH THE PINS IN THE MOTHERBOARD CONNECTOR.

5. HAVE YOU TRIED BOTH PHASES WITH THE PHASE SWITCH, AND ARE THE OTHER SETTINGS ON THE DIP-SWITCH CORRECT?

6. YOU SHOULD BE ABLE TO ADJUST THE VOLUME ON YOUR RECORDER BY ABOUT 50% DURING PLAYBACK, AND STILL HAVE THE SYNC LIGHT LIT WHEN READING THE SYNC STREAM. IF IT DOESN'T LIGHT AT ALL, THERE IS PROBABLY SOME GROSS PROBLEM ON THE BOARD, SUCH AS A BAD PLATED THROUGH HOLE, A SOLDER BRIDGE, OR A BAD INTEGRATED CIRCUIT. IF YOU ARE USING A TAPE DECK THAT HAS ONLY A PREAMP, YOU MAY NEED TO ADD AN EXTRA STAGE OF AMPLIFICATION IN ONE OF THE EXTRA IC SLOTS.

7. IF YOU HAVE AN OSCILLOSCOPE, THE BEST PLACE TO LOOK TO SEE HOW THE RECEIVER INPUT SECTION IS OPERATING IS AT IC 4, PIN 11. THIS SIGNAL SHOULD BE FAIRLY CLEAN, WITH SOME OVERALL JITTER, DUE TO THE TAPE WOW AND FLUTTER, AND SOME HIGH-SPEED JITTER ON THE EDGE OF THE WAVEFORM. IT IS THIS HIGH-SPEED JITTER THAT YOU SHOULD TRY TO MAKE A MINIMUM.

8. IF YOU HAVE A VIDEO INTERFACE, OR OTHER SOURCE OF HIGH-FREQUENCY NOISE, TRY LOCATING IT FURTHER AWAY FROM THE CASSETTE INTERFACE.

9. ARE YOU SURE THAT YOUR RECORDER HAS A FREQUENCY RESPONSE TO 8KHZ?

10. HAVE YOU USED THE PROPER VOLTAGE DIVIDER (R10, R11) FOR YOUR PARTICULAR RECORDER? IF YOU ARE ABLE TO RECOVER THE SYNC STREAM I WROTE SATISFACTORILY, BUT ARE HAVING TROUBLE WITH RECORDINGS YOU MAKE YOURSELF, THE LEVEL GOING FROM THE INTERFACE TO THE RECORDER MAY BE TOO HIGH OR TOO LOW, ESPECIALLY IF YOU HAVE AUTOMATIC LEVEL CONTROL. YOU MAY ALSO WANT TO TRY OPERATING WITHOUT C15.

11. HAVE YOU CHECKED YOUR +5 VOLT POWER? TOO MANY BOARDS IN YOUR COMPUTER COULD INTRODUCE RIPPLE ON THIS SUPPLY.

12. DON'T USE DIGITALLY CERTIFIED TAPE, ONLY AUDIO LOW-NOISE TAPE.

13. IF YOUR RECORDER HAS AN INTERNAL MICROPHONE, BE SURE IT IS NOT ACTIVE WHILE YOU ARE MAKING A RECORDING (THE J.C. PENNY HAS A SWITCH ON THE TONE CONTROL WHICH CUTS OFF THE INTERNAL MIC).

14. IF YOU STILL HAVE PROBLEMS, PLEASE RETURN THE UNIT, PREFERABLY WITH YOUR CASSETTE RECORDER, AND I WILL GET IT OPERATING PERFECTLY WITHOUT CHARGE. THE REPAIR TURNAROUND TIME IS 1 TO 3 WEEKS.

15. IF YOU ARE COMPLETELY DISSATISFIED, YOU MAY RETURN THE INTERFACE FOR A COMPLETE REFUND WITHIN 90 DAYS AFTER YOU ACCEPTED DELIVERY.

IDEAS FOR USING THE CASSETTE INTERFACE

SAVING AND LOADING PROGRAMS

PROGRAMS MAY BE TOGGLED INTO MEMORY, LOADED FROM PAPER TAPE, OR LOADED BY SOME OTHER MEANS. THEY MAY THEN BE DUMPED ONTO CASSETTE USING THE CASSETTE OUTPUT ROUTINE. THE OUTPUT ROUTINE ITSELF MAY BE DUMPED ALONG WITH THE OTHER PROGRAM, SO THAT IT WILL BE AVAILABLE LATER FOR FURTHER DUMPS. ROUTINES MAY THEN BE LOADED FROM CASSETTE, MODIFIED, AND DUMPED BACK OUT TO CASSETTE IN A CONTINUOUS PROCESS OF DEVELOPMENT.

USING A BACKUP

A BACKUP IS A METHOD OF MAKING SURE THAT VALUABLE PROGRAMS OR DATA IS NOT LOST. ONE SIMPLE WAY OF PROVIDING A BACKUP IS TO RECORD A PARTICULAR PROGRAM IN TWO DIFFERENT PLACES ON THE SAME TAPE, OR ON TWO DIFFERENT TAPES. YOU MAY WANT TO GO BACK AND FORTH BETWEEN THE TWO COPIES, EACH TIME THE PROGRAM IS CHANGED. IN THIS WAY, YOU ALWAYS HAVE A COPY OF THE LAST PROGRAM, AND ONLY THE MOST RECENT CHANGES ARE LOST IF A POWER FAILURE OR OTHER EQUIPMENT TROUBLE DEVELOPS DURING THE PROCESS OF SAVING.

STARTING AND STOPPING AUTOMATICALLY

MOST AUDIO CASSETTE RECORDERS HAVE AN INPUT LABELED "REMOTE". THIS IS NORMALLY OPERATED FROM A SWITCH ON THE MICROPHONE, SO THAT THE RECORDER CAN BE STARTED AND STOPPED WHILE DICTATING. THIS INPUT CAN BE USED TO START AND STOP THE RECORDER UNDER PROGRAM CONTROL FROM THE COMPUTER. ONE OF THE 4 EXTRA CONTROL LINES COMING FROM THE CASSETTE INTERFACE (DIP-SOCKET PINS 5, 6, 7, OR 8) CAN BE USED TO DRIVE A RELAY WHICH WOULD HAVE ITS CONTACTS CONNECTED TO THE RECORDER REMOTE JACK. THE MAIN REQUIREMENTS ARE THAT THE 40 MA AVAILABLE FROM THE CONTROL LINE BE ABLE TO DRIVE THE RELAY, AND THAT THE RELAY CONTACTS ARE ABLE TO HANDLE THE CURRENT INTO THE REMOTE JACK, WHICH CAN BE AS HIGH AS ONE AMPERE.

OPERATING WITH MORE THAN ONE CASSETTE RECORDER

SOMETIMES IT IS DESIREABLE TO OPERATE WITH TWO OR MORE CASSETTE UNITS. WITH THE TARBELL CASSETTE INTERFACE, IT IS ALREADY POSSIBLE TO READ FROM ONE CASSETTE RECORDER, WHILE WRITING ONTO ANOTHER. THIS IS BECAUSE THE INPUT AND OUTPUT SECTIONS ARE ENTIRELY INDEPENDENT, AND MAY BE PROGRAMMED SEPERATELY. IF IT IS DESIRED TO READ FROM ONE OF TWO UNITS, AND WRITE ONTO ANOTHER, SUCH AS DURING A MERGING OPERATION, RELAYS MAY BE USED TO SWITCH BACK AND FORTH BETWEEN THE TWO INPUT UNITS, UNDER CONTROL OF ONE OF THE 4 CONTROL LINES. IF, HOWEVER, IT IS NECESSARY TO READ SIMULTANEOUSLY FROM TWO DIFFERENT UNITS, OR WRITE DIFFERENT INFORMATION ONTO TWO UNITS AT THE SAME TIME, IT WILL BE NECESSARY TO HAVE TWO CASSETTE INTERFACES.

MODIFICATIONS ON CASSETTE BASIC FOR CSAVE AND CLOAD
USING TARBELL CASSETTE INTERFACE

8K BASIC 3.2

ADDR	OLD	NEW
1008	06	6E
1009	E6	E6
100A	01	10
100B	C2	C2
100C	07	07
100D	10	10
100E	DB	DB
100F	07	6F
1016	06	6E
1017	E6	E6
1018	80	20
1019	C2	C2
101A	15	15
101B	10	10
101C	F1	F1
101D	D3	D3
101E	07	6F
101F	C9	C9
1020	E5	E5
1021	3E	CD
1022	D3	14
1023	CD	10
1024	14	3E
1025	10	E6
1053	06	3E
1054	03	10
1055	CD	D3
1056	07	6E
1057	10	00
1058	D6	00
1059	D3	00
105A	C2	00
105B	A3	00
105C	10	00
105D	05	00
105E	C2	00
105F	A5	00
1060	19	00

12K BASIC 3.2

ADDR	OLD	NEW
155A	06	6E
155B	E6	E6
155C	01	10
155D	C2	C2
155E	59	59
155F	15	15
1560	DB	DB
1561	07	6F
1568	06	6E
1569	E6	E6
156A	80	20
156B	C2	C2
156C	67	67
156D	15	15
156E	F1	F1
156F	D3	D3
1570	07	6F
1571	C9	C9
1572	E5	E5
1573	3E	CD
1574	D3	66
1575	CD	15
1576	66	3E
1577	15	E6
15A5	06	3E
15A6	03	10
15A7	CD	D3
15A8	59	6E
15A9	15	00
15AA	D6	00
15AB	D3	00
15AC	C2	00
15AD	A5	00
15AE	15	00
15AF	05	00
15B0	C2	00
15B1	A7	00
15B2	15	00

DISK BASIC

ADDR	OLD	NEW
0E58	06	FA
0E59	D2	67
0E5A	FA	0E
0E5B	67	3E
0E5C	0E	19
0E5D	78	CD
0E5E	CD	FD
0E5F	FA	1D
0E60	1D	2F
0E61	CD	CD
0E62	FA	FD
0E63	1D	1D
0E64	C3	C3
0E65	74	74
0E66	0E	0E
0E67	0E	3E
0E68	04	10
0E69	CD	D3
0E6A	F0	6E
0E6B	1D	00
0E6C	B8	00
0E6D	C2	00
0E6E	67	00
0E6F	0E	00
0E70	0D	00
0E71	C2	00
0E72	69	00
0E73	0E	00
0E74	F7	F7
1DF0	DB	DB
1DF1	06	6E
1DF2	E6	E6
1DF3	01	10
1DF4	C2	C2
1DF5	F0	F0
1DF6	1D	1D
1DF7	DB	DB
1DF8	07	6F

ADDR	OLD	NEW
1DFE	DB	DB
1DFF	06	6E
1E00	E6	E6
1E01	80	20
1E02	C2	C2
1E03	FE	FE
1E04	1D	1D
1E05	F1	F1
1E06	D3	D3
1E07	07	6F
1E08	C9	C9
1E09	06	06
1E0A	01	01
1E0B	FE	FE
1E0C	2E	2E
1E0D	CA	CA
1E0E	12	12
1E0F	0E	0E
1E10	E5	E5
1E11	3E	CD
1E12	D3	FD
1E13	CD	1D
1E14	FD	3E
1E15	1D	E6
1E16	CD	CD
1E17	FA	FD
1E18	1D	1D
1E48	06	3E
1E49	03	10
1E4A	CD	D3
1E4B	F0	6E
1E4C	1D	00
1E4D	D6	00
1E4E	D3	00
1E4F	C2	00
1E50	48	00
1E51	1E	00
1E52	05	00
1E53	C2	00
1E54	4A	00
1E55	1E	00

PROCESSOR TECHNOLOGY SOFTWARE PACKAGE #1

Modified for Low Memory and Cassette by Fred Clark,
Modified for Tarbell Cassette by Don Tarbell.

The main purpose of this software is to allow development of assembly-language programs. Programs may be entered from the keyboard in standard assembly-language format, edited, assembled, and saved on cassette in either source or machine format. These programs may be then loaded back into memory at some future time for listing, editing or running. Following is a summary of the commands. All address and byte information is in hex.

FILE /NAME/ AAAA	-Creates a file of name NAME at address AAAA (hex). Up to six text files can be in memory at the same time. The file just created is the current file.
FILE /NAME/ 0	-This command deletes the file named NAME.
FILE /NAME/	-Makes the named file the current file.
FILE	-Prints the starting and ending addresses of the current file, and it's name.
FILES	-Lists the name, starting & ending address for each of the files in the system. Current file at top.
DUMP SSSS EEEE	-Dumps the contents of memory from SSSS to EEEE.
EXEC AAAA	-Executes the machine-language program at address AAAA.
EXEC /NAME/	-Searches the cassette for named file, loads and exec.
ENTR AAAA	-Enter data into memory: B1 B2 B3 B4/
LIST NNNN	-Lists the current file starting at line NNNN.
DELT NNNN MMMM	-Deletes lines NNNN through MMMM from current file.
NNNN (text entered)	-Enter a line of text into the current file. Always use 4 decimal digits. It works similar to BASIC, in that the numbered lines are ordered automatically.
ASSM AAAA	-Assembles current file into address AAAA.
ASSM AAAA BBBB	-Assembles at address AAAA, but puts code at BBBB.
CUST	-Execute a customer (user) routine at address E000.
SFIL	-Save the current file on cassette.
LFIL	-Load the current file from cassette. The name of the file on tape must match the name of the current file. Be sure there is enough room in memory.
CFIL	-Check a cassette file written with the SFIL command above, for errors, without overwriting current file.
AFIL	-Append a file from cassette to the current file.
SAVE SSSS EEEE	-Save a block of memory from SSSS to EEEE on cassette.
LOAD SSSS	-Load a block of memory from cassette starting at SSSS.
CHEK SSSS	-Check a cassette file written with SAVE command.
NAME /NAME/	-Rename the current file to NAME.
RNUM N	-Renumber the current file by increment N.
NLIS NNNN	-Lists the current file at NNNN without line numbers.

This software is useful, but may not be completely free of bugs. It loads into the first 4096 bytes of memory, and uses about 2048 bytes following, for tables and scratch pad. Ctl-C escapes from any printing. Ctl-W freezes printing until another character is typed. The package is available from Tarbell Electronics for the prices listed below:

Cassette Tape with software and instructions: \$5.00
A reassembled and patched (updated) listing: \$5.00

WRITING PROGRAMS FOR THE CASSETTE INTERFACE

SOMETIMES IT IS NECESSARY TO WRITE ASSEMBLY OR MACHINE LANGUAGE PROGRAMS FOR A PARTICULAR INTERFACE. THE CLASS OF PROGRAMS WE ARE TALKING ABOUT HERE ARE CALLED "DRIVERS". THESE WOULD BE REQUIRED FOR LINKAGE TO A PIECE OF SOFTWARE FOR WHICH PATCHES ARE NOT PROVIDED IN THE MANUAL.

THIS INTERFACE IS A SYNCRONOUS DEVICE. ONE OF THE IMPLICATIONS OF THIS IS THAT DATA OR PROGRAMS ARE MOST EFFICIENTLY WRITTEN AS A CONTIGUOUS BLOCK, RATHER THAN AS SEPERATE BYTES. THERE ARE A FEW RULES THAT MUST BE FOLLOWED WHEN WRITING SOFTWARE FOR THIS DEVICE:

1. THE FIRST BYTE MUST BE A "START BYTE" WHICH MAY BE ANY BYTE EXCEPT 00, FF, OR E6 (HEXADECIMAL).
2. THE SECOND BYTE MUST BE A "SYNC BYTE" WHICH MUST BE E6 (HEX).
3. THE SOFTWARE MUST BE ABLE TO DELIVER BYTES TO THE INTERFACE AS FAST AS IT CAN ACCEPT THEM, WHICH AT THE STANDARD SPEED, IS 187 BYTES PER SECOND. THIS MEANS THAT ANY LOOP THAT THE PROGRAM GOES THROUGH WHICH IS BETWEEN BYTES, MUST LAST LESS THAN 5.3 MILLISECONDS. AN AVERAGE INSTRUCTION TIME ON THE 8080 WITH NO WAIT STATES IS 2 CYCLES, OR 1 MICROSECOND. THUS, THERE SHOULD BE NO MORE THAN ABOUT 5300 INSTRUCTION EXECUTIONS BETWEEN BYTES BEING SENT OUT. THIS IS NORMALLY NOT A PROBLEM.
4. A SIMILAR CONSTRAINT MUST BE OBSERVED WITH RESPECT TO THE INPUT SOFTWARE, WHICH SHOULD BE ABLE TO ACCEPT DATA AS FAST AS IT IS BEING MADE READY BY THE INTERFACE.

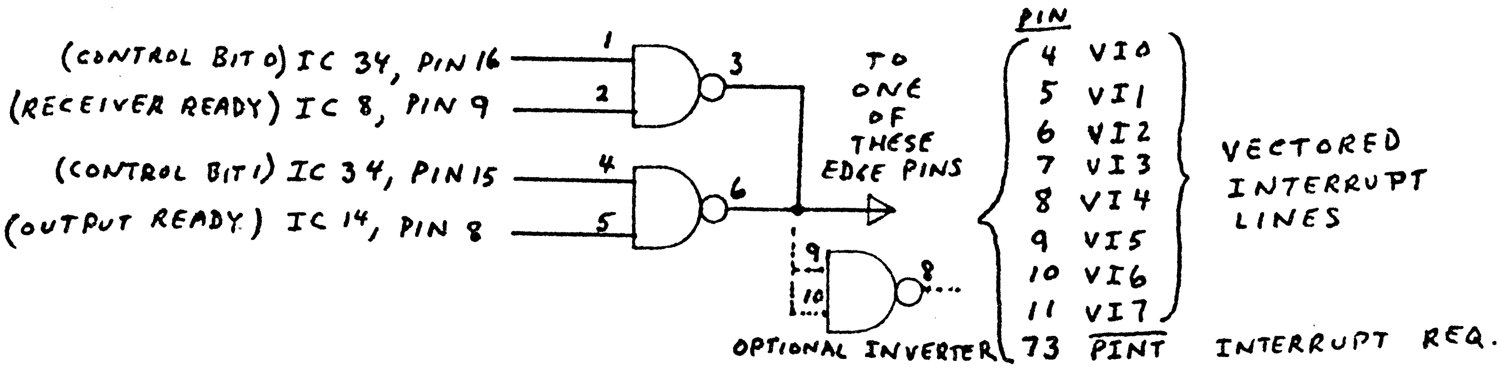
THERE ARE SITUATIONS IN WHICH THE DATA CANNOT BE PROVIDED OR ACCEPTED FAST ENOUGH BY THE SOFTWARE. ONE EXAMPLE OF THIS IS DATA WHICH IS BEING GENERATED BY A PROGRAM RUNNING IN BASIC. THIS PROBLEM IS SOLVED BY SENDING THE DATA OUT AND READING IT IN A LINE AT A TIME, WITH NULLS IN BETWEEN. THE PAGE ENTITLED "HOW TO SAVE AND LOAD DATA FROM A BASIC PROGRAM" IS A SAMPLE PROGRAM THAT SHOWS HOW TO DO THIS. THERE ARE AT LEAST TWO OTHER WAYS TO HANDLE THIS PROBLEM: 1) SEND EACH BYTE AS A SEPERATE BLOCK WITH IT'S OWN START AND SYNC BYTES. 2) ACCUMULATE BYTES IN A BUFFER AREA OF MEMORY, AND START AND STOP THE CASSETTE RECORDER UNDER CONTROL OF THE COMPUTER WHEN IT IS TIME TO DUMP AND REFILL THE BUFFER.

OTHER ITEMS TO BE CONSIDERED WHEN WRITING SOFTWARE:

1. SINCE TAPE IS AN IMPERFECT MEDIUM, IT IS GENERALLY USEFUL TO INCORPORATE AN ERROR-CHECKING SCHEME, SUCH AS THE CHECKSUM SYSTEM THAT IS DEMONSTRATED ON THE PAGES ENTITLED "OUTPUT ROUTINE WITH CHECKSUM" AND "INPUT ROUTINE WITH CHECKSUM". THERE ARE MANY SYSTEMS POSSIBLE, EACH PROVIDING DIFFERENT KINDS AND LEVELS OF PROTECTION, AND BOOKS HAVE BEEN WRITTEN ABOUT THESE.
2. IT IS SOMETIMES HANDY TO HAVE AN IDENTIFIER, SUCH AS A NAME, WRITTEN ALONG WITH THE FILE ONTO CASSETTE. AN EXAMPLE IS THE ONE-LETTER NAME GIVEN TO ALTAIR BASIC PROGRAMS WHEN WRITING TO CASSETTE.
3. ANOTHER ITEM THAT IS USEFUL IS A WAY FOR THE PROGRAM TO TELL HOW LONG THE FILE IS. A ONE OR TWO-BYTE HEADER TO INDICATE LENGTH OF A FILE OR BLOCK IS SOMETIMES USED. IN THE MODIFIED PROCESSOR TECHNOLOGY SOFTWARE PACKAGE #1 THAT I SUPPORT, BLOCKS ARE PRECEEDED BY ONE LENGTH BYTE THAT MAY BE FROM 1 TO 255. A LENGTH OF ZERO INDICATES THE END OF THE FILE.
4. A "TYPE BYTE" IS A UNIQUE BYTE FOR A PARTICULAR FORMAT OF FILE, SO THAT A SOPHISTICATED LOADER MAY DISTINGUISH BETWEEN DIFFERENT TYPES, AND LOAD THEM APPROPRIATELY. I USE A 90 (HEX) TYPE BYTE IMMEDIATELY FOLLOWING THE SYNC BYTE ON THE PROC. TECH. SOFTWARE.

USING THE TARBELL CASSETTE INTERFACE UNDER INTERRUPT CONTROL

There is no built-in provision for interrupts in the design. However, it is not too difficult to make a modification to the board to provide for interrupts. Two bits in the control line register are used as interrupt-enable flip-flops. It is necessary to install another 7403 quad 2-input open-collector NAND gate in one of the spare IC positions at the top of the board. The circuit diagram below shows how to connect the 7403:



If you do not have a vectored interrupt card, connect the output pins 3 and 6 to the edge connector pin 73. An interrupt will then be caused when the following conditions are true: 1) Interrupts are enabled with an EI instruction. 2) Control bit 0 is high <u>and</u> the receiver is ready with an input byte; <u>or</u> control bit 1 is high <u>and</u> the transmitter is ready for an output byte. Both conditions 1 and 2 must be true for an interrupt to be caused.

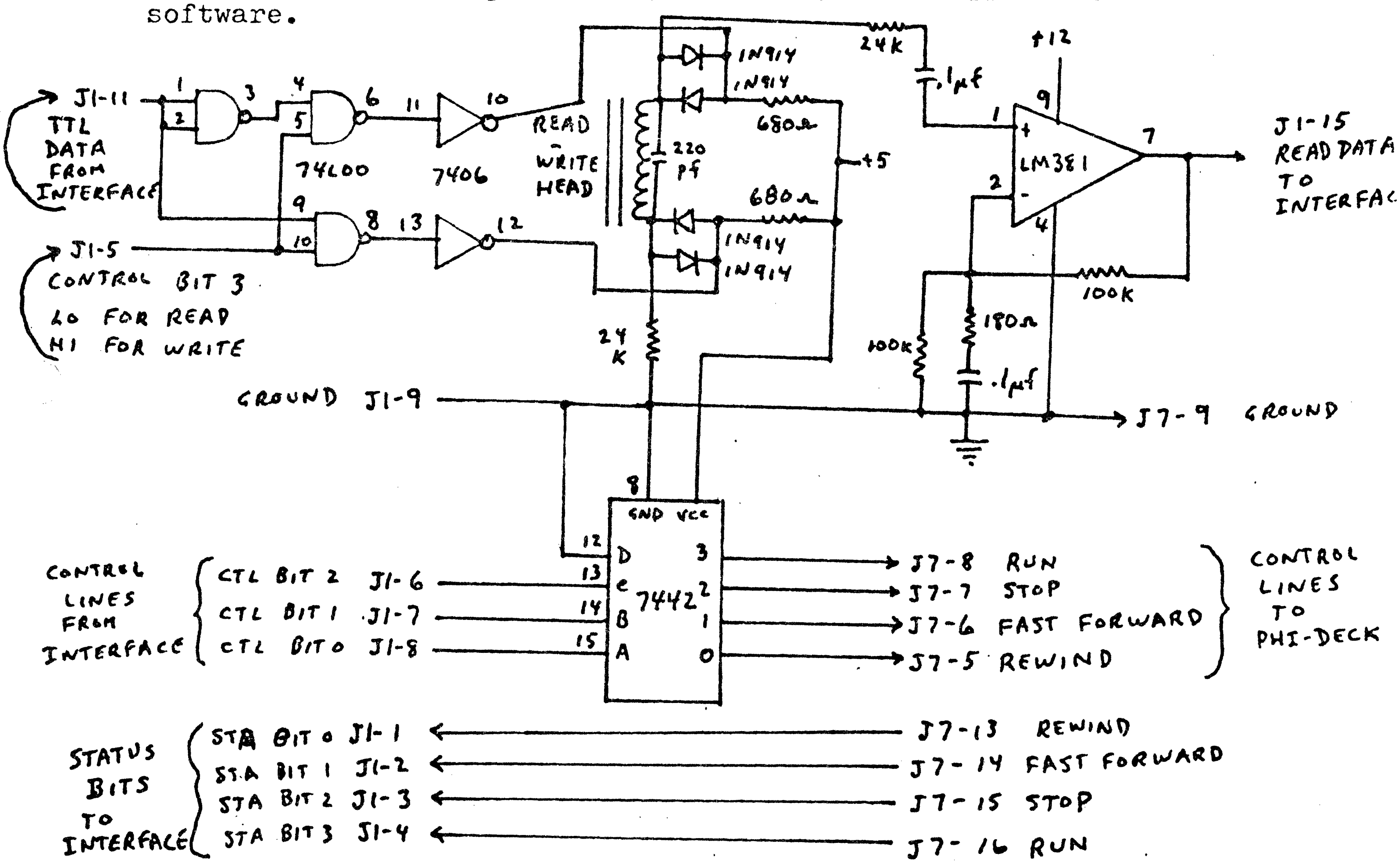
If you do have a vectored interrupt card, connect the output pins 3 and 6 to one of the edge connector pins 4,5,6,7,8,9, 10, or 11, depending on what level you want the cassette interface. Be sure that you have the correct polarity to cause an interrupt. The output at pins 3 and 6 goes low to cause an interrupt. If you have a vectored interrupt card that requires a high-going signal to cause an interrupt, you may use the extra gate shown above to invert the line. Read your manual on your interrput board to make sure.

There is not enough room here to explain how to use the interrupt system, but the following lines show how to enable and disable the two interrput bits on this board.

enable receiver interrupt:	MVI	A,1
	OUT	6EH
enable transmitter interrupt:	MVI	A,2
	OUT	6EH
enable both interrupts:	MVI	A,3
	OUT	6EH
disable both interrupts:	MVI	A,0
	OUT	6EH

PRELIMINARY PHI-DECK ADAPTER INFORMATION

We have purchased the introductory variable-speed PHI-DECK package, which includes the deck, control electronics, power supply, and control box. We have been experimenting with an adapter which connects between the Tarbell Cassette Interface, the read-write heads, and the control electronics. This adapter provides start-stop, forward, rewind, and fast-forward control for the PHI-DECK from the Tarbell Cassette Interface. It also includes read-write electronics for one channel. It is still in the experimental stage, but speeds of 1000 bytes per second have been attained. The information is being provided here for those who would like to experiment further along these lines. The circuit is not guaranteed to work for your application, and will probably not be exactly what we end up with for our use. As progress is made, further refinements will be published, including more supporting software.



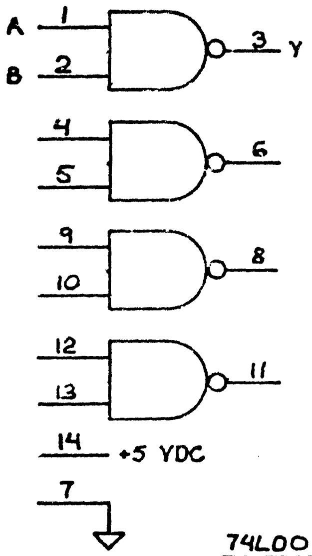
The simple program below is handy for experimenting. Flip sense switch 2 up momentarily to pulse the control line.

LOOP	IN FFH	DB FF	Read Sense Switches.
	OUT 6EH	D3 6E	Write to Control Port.
	IN 6EH	DB 6E	Read Status Lines.
	OUT FFH	D3 FF	Write To Display Lights.
	JMP LOOP	C3 00 00	Do it all over again.

Control Table

s1	s0	function
0	0	RUN
0	1	STOP
1	0	FF
1	1	REW

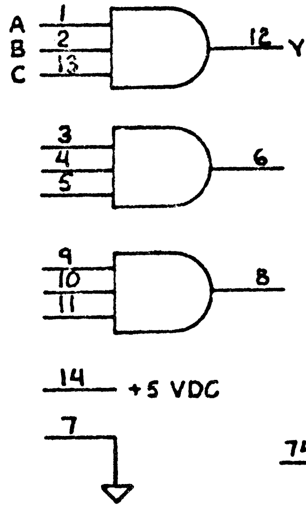
Sense switch 3 should be up for read, down for write.



INPUTS		OUTPUT
A	B	Y
0	0	1
1	0	1
0	1	1
1	1	0

$Y = \overline{AB}$ $Y = \bar{A} + \bar{B}$

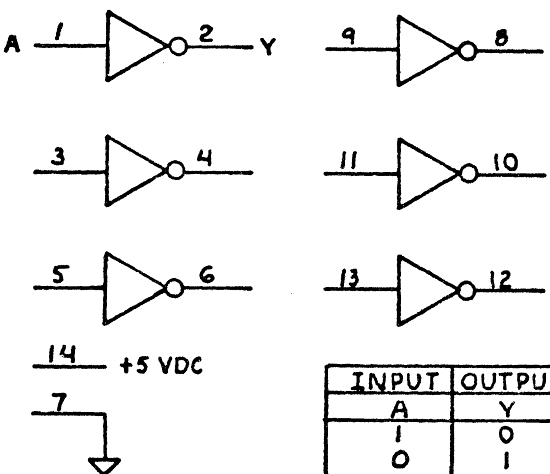
74L00/7403 NAND/OR



SEE TRUTH TABLE FOR 7408

$Y = ABC$ $\bar{Y} = \bar{A} + \bar{B} + \bar{C}$

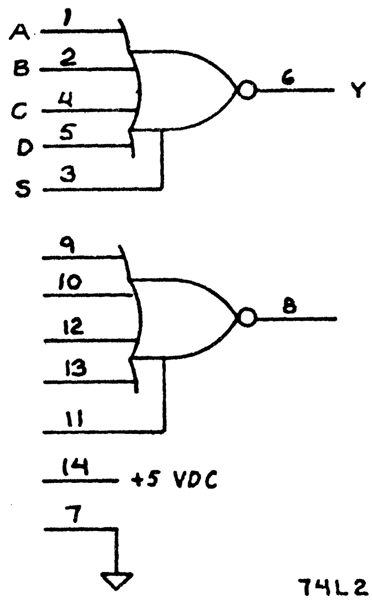
74L11 AND



INPUT	OUTPUT
A	Y
1	0
0	1

$Y = \bar{A}$

7404/74L04/7406 INVERTER



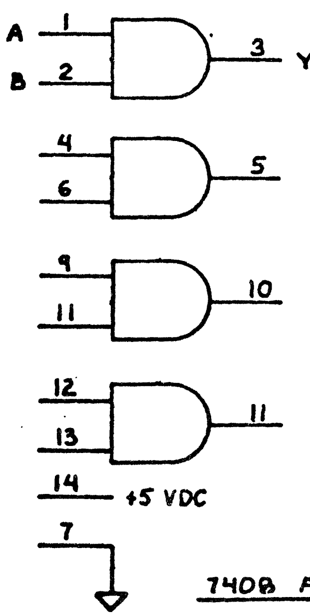
INPUTS	OUTPUT
S A B C D	Y
1 X X X X	0
0 0 0 0 0	1
0 0 0 0 1	1
0 0 0 1 0	1
0 0 1 0 0	1
0 1 0 0 0	1
↓	
0 0 1 1 1	1
0 1 1 1 1	0

X= DON'T CARE

$Y = \overline{S(A+B+C+D)}$

$Y = S(ABCD)$

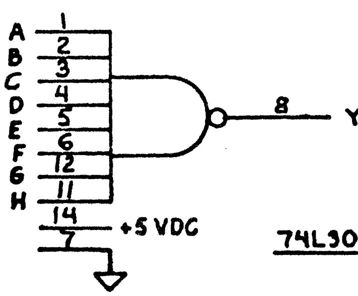
74L25 NOR/AND



INPUTS		OUTPUT
A	B	Y
0	0	0
1	0	0
0	1	0
1	1	1

$Y = AB$ $\bar{Y} = \bar{A} + \bar{B}$

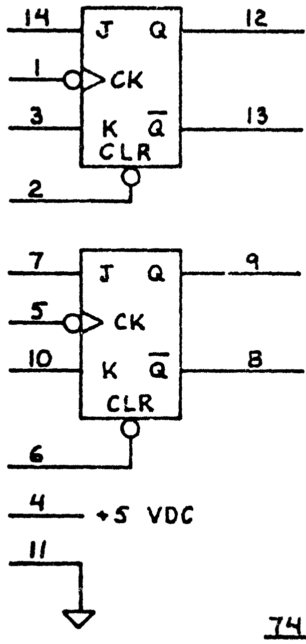
7408 AND



SEE TRUTH TABLE
TABLE FOR 74L00/7403

$Y = \overline{ABCDEFGH}$

74L30 AND

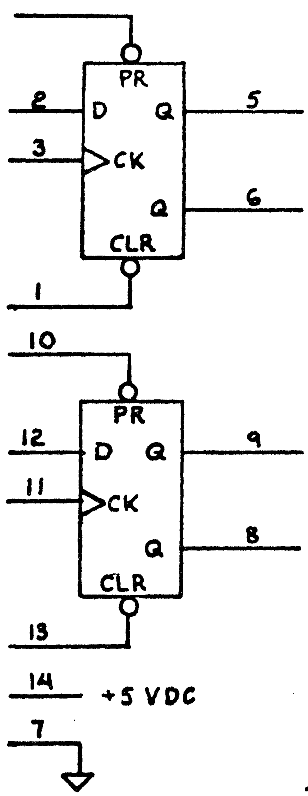


INPUTS				OUTPUTS	
CLR	CK	J	K	Q	$\overline{Q}$
0	X	X	X	0	1
1	⎍	0	0	NO CHANGE	
1	⎍	1	0	1	0
1	⎍	0	1	0	1
1	⎍	1	1	$\overline{Q}$	Q

NO CHANGE IN OUTPUT UNTIL FALLING EDGE OF CLOCK

X = DON'T CARE

74L73/7473 J-K FLIP-FLOP WITH CLEAR

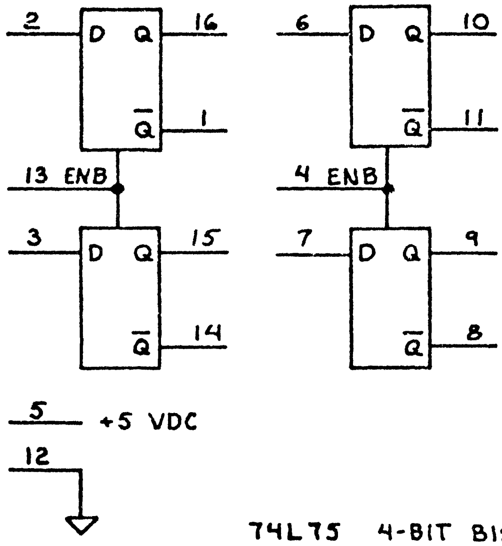


INPUTS				OUTPUTS	
CLR	PR	CK	D	Q	Q
1	0	X	X	1	0
0	1	X	X	0	1
0	0	X	X	ILLEGAL	
1	1	↑	1	1	0
1	1	↑	0	0	1

NO CHANGE IN OUTPUT UNTIL RISING EDGE OF CLOCK

X = DON'T CARE

74L74 D-TYPE FLIP-FLOP WITH PRESET AND CLEAR

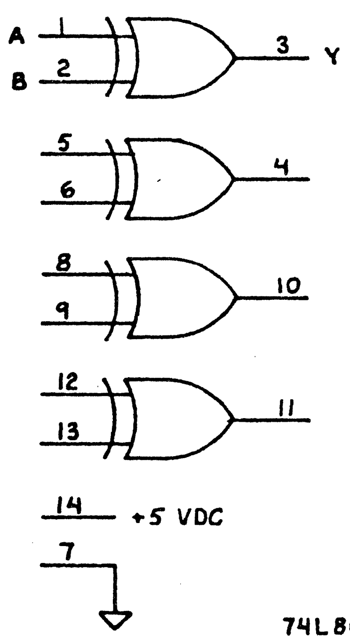


INPUTS		OUTPUTS	
ENB	D	Q	$\bar{Q}$
0	X	Q_0	$\bar{Q}_0$
1	0	0	1
1	1	1	0

X= DON'T CARE

Q_0= THE STATE OF Q BEFORE ENB IS SET LOW

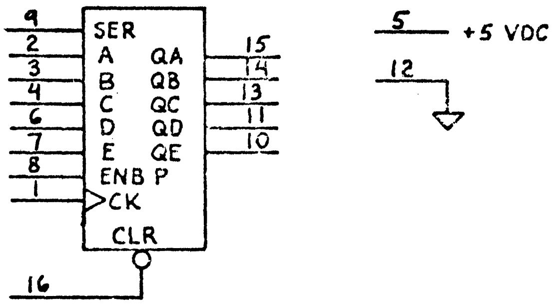
74L75 4-BIT BISTABLE LATCH



INPUTS		OUTPUT
A	B	Y
0	0	0
1	0	1
0	1	1
1	1	0

$Y = A \oplus B$ $\quad$ $Y = A\bar{B} + \bar{A}B$

74L86 EXCLUSIVE-OR



SER = SERIAL DATA INPUT. DATA IS MOVED FROM SERIAL INPUT TO QA AFTER THE FIRST CLOCK.

A-B = PARALLEL DATA INPUT. DATA IS PRESENT ON ASSOCIATED OUTPUT (A→QA) AFTER CLOCK. IF MODE IS SHIFTED FROM PARALLEL, DATA WILL MOVE TOWARDS QE ON EACH FOLLOWING CLOCK.

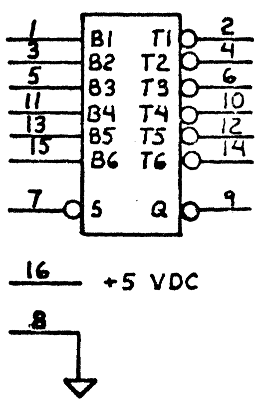
ENB P = ENABLE PARALLEL MODE. WHEN SET, REGISTER IS IN PARALLEL MODE. WHEN RESET, REGISTER IS IN SERIAL MODE.

CK = CLOCK. NO CHANGE IN OUTPUT UNTIL RISING EDGE OF CLOCK.

CLR = CLEAR. WHEN RESET ALL OUTPUTS ARE FORCED TO TO RESET WITHOUT REGARDS TO ANY OTHER INPUT.

QA-QE = OUTPUTS. ALL OUTPUTS USED FOR PARALLEL OUTPUT. FOR SERIAL OUTPUT, ONE OF THE OUTPUTS IS USED DEPENDING ON THE LENGTH DESIRED.

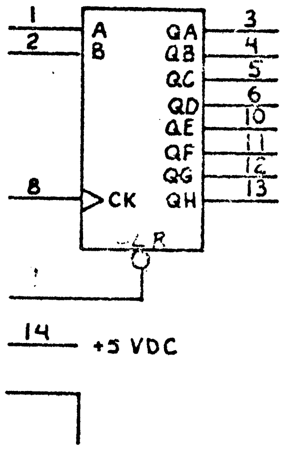
7496 5-BIT SHIFT REGISTER



IF B1 THROUG B6 ARE BIT-FOR-BIT EQUAL TO THE BITS RESET ON T1 THROUGH T6, Q IS RESET. FOR ALL OTHER CONDITIONS Q IS SET

S = STROBE. WHEN SET Q IS DISABLED

DM8131 6-BIT COMPARITOR



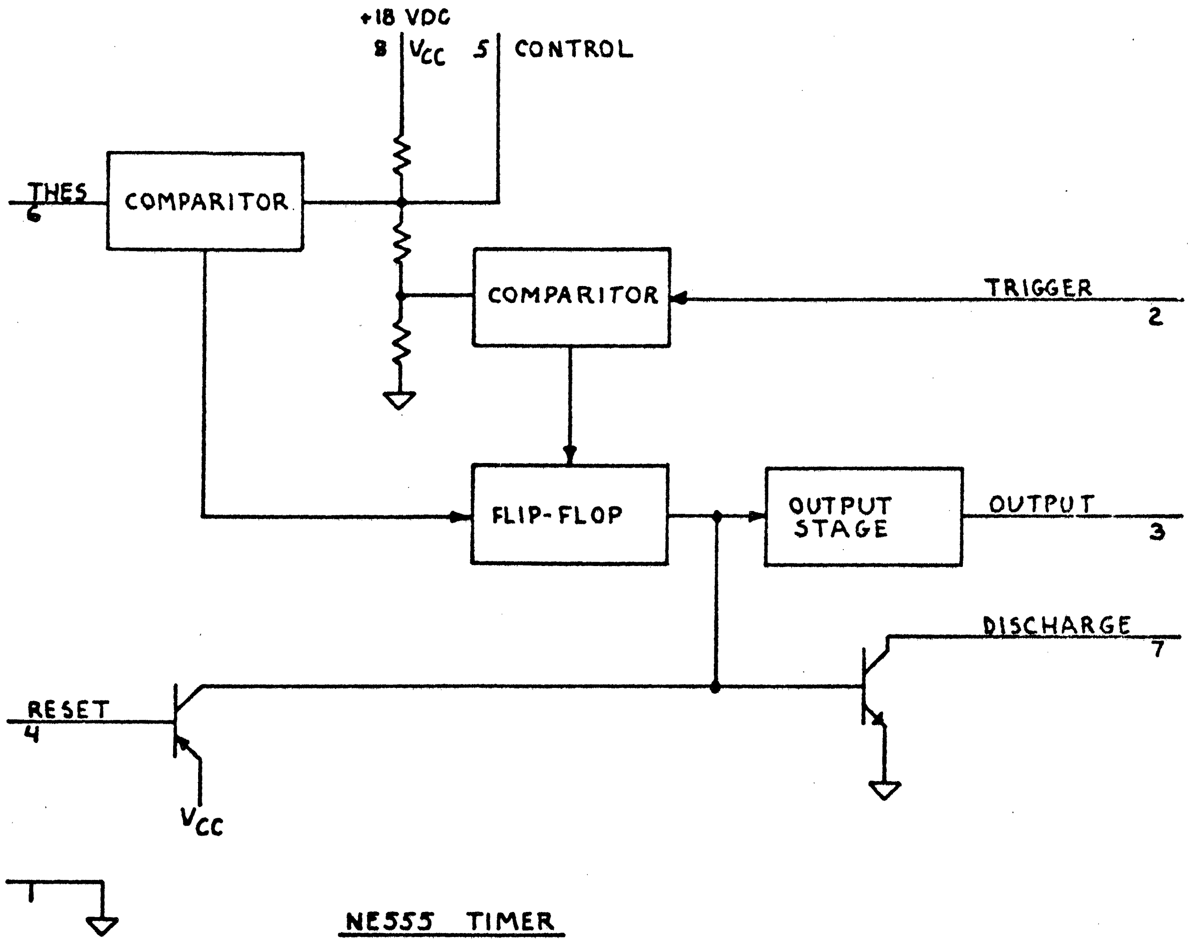
A,B = SERIAL DATA INPUTS. ONE OF THE TWO MUST BE SET TO ENABLE THE OTHER. DATA IS PRESENT AT QA AFTER ONE CLOCK AND MOVES TOWARD QH ON EACH FOLLOWING CLOCK.

CK = CLOCK. NO CHANGE IN OUTPUT UNTIL RISING EDGE OF CLOCK

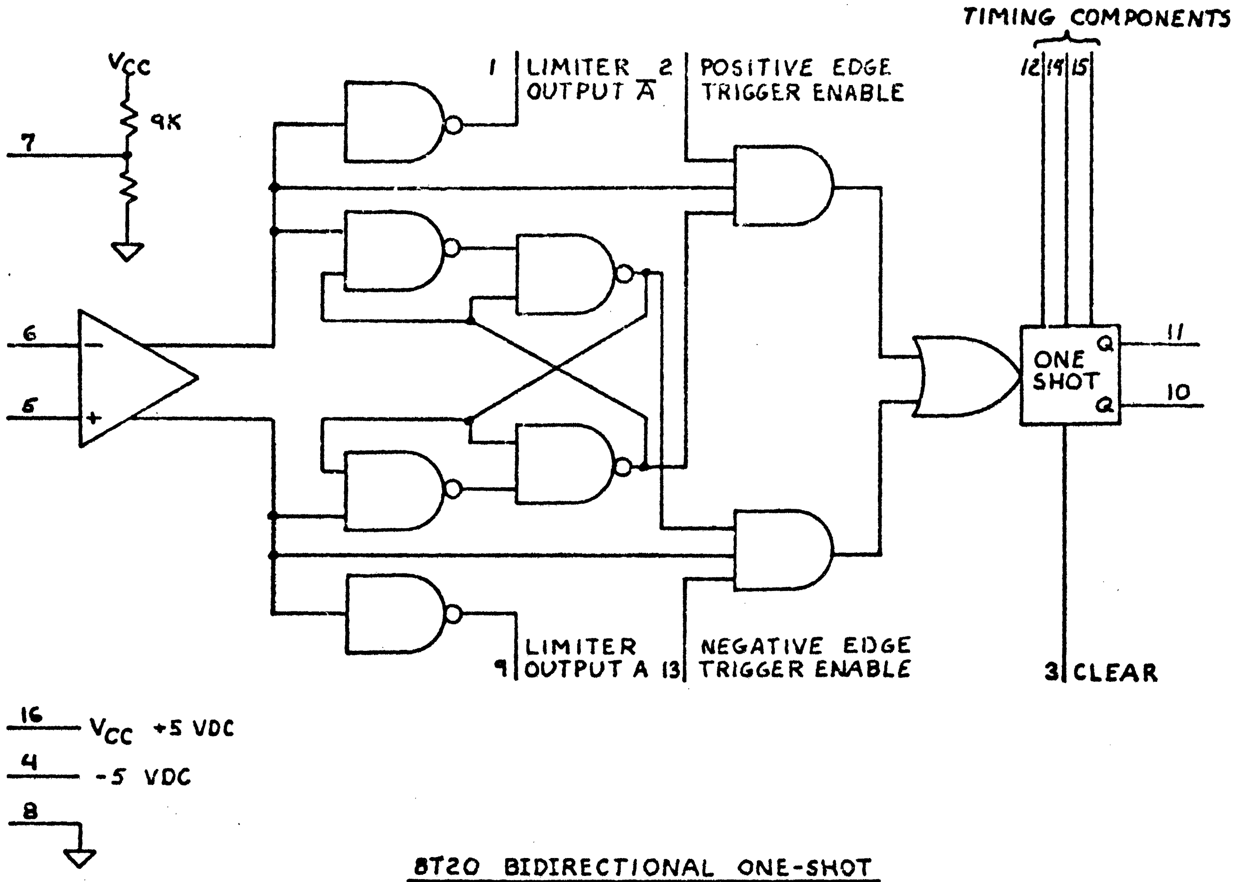
CLR = CLEAR. WHEN RESET, ALL OUTPUTS ARE RESET WITHOUT REGARDS TO TO OTHER INPUTS.

QA-QH = OUTPUTS. DATA IS MOVED FROM A AND B TO QA ON EACH CLOCK AND THE DATA AT QA IS MOVED TOWARDS QH.

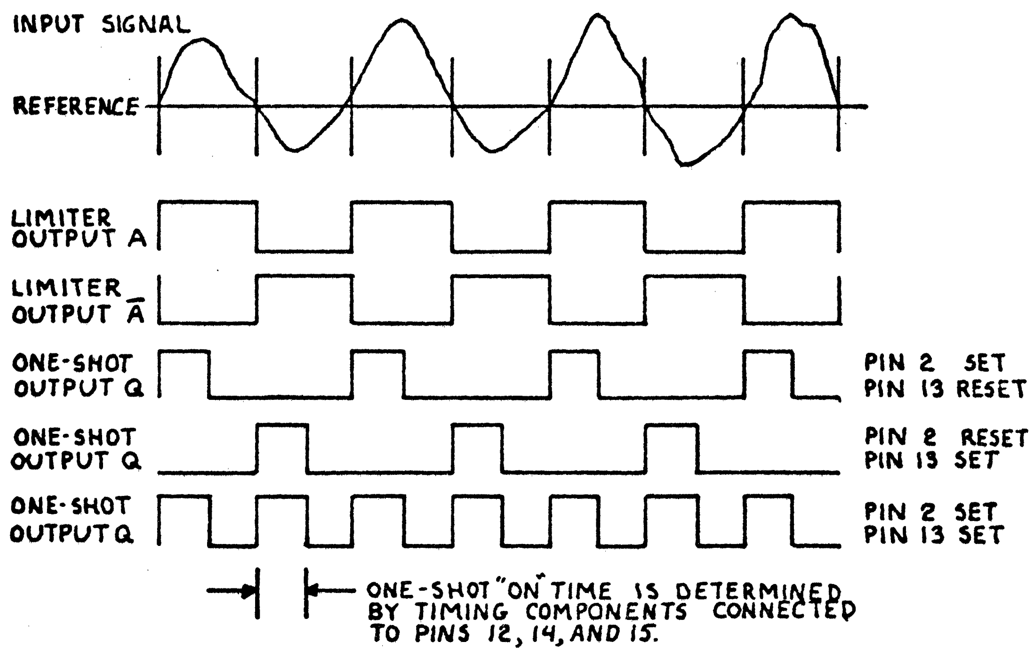
74L164 8-BIT PARALLEL-OUT SHIFT REGISTER



NE555 TIMER



8T20 BIDIRECTIONAL ONE-SHOT



8T20 BIDIRECTIONAL ONE-SHOT INPUT AND OUTPUT WAVEFORMS

DEAR CUSTOMER,

THANK YOU VERY MUCH FOR PURCHASING A CASSETTE INTERFACE. I AM INTERESTED TO KNOW WHAT KIND OF PROGRESS YOU HAVE MADE WITH YOUR TARBELL CASSETTE INTERFACE. I WOULD REALLY APPRECIATE IT IF YOU WOULD TAKE TIME TO FILL THIS QUESTIONAIRE OUT. THIS WILL HELP ME TO PROVIDE YOU WITH BETTER SERVICE IN THE FUTURE.

DID YOUR INTERFACE ARRIVE IN A REASONABLE LENGTH OF TIME? YES NO

WERE ANY OF THE ITEMS DAMAGED IN SHIPMENT? YES NO

WERE ANY OF THE ITEMS MISSING? IF SO, WHAT? YES NO

WAS THE QUALITY OF WORKMANSHIP ON THE BOARD REASONABLE? YES NO

HAVE YOU STARTED CONSTRUCTION YET? YES NO

HAVE YOU HAD ANY PROBLEM UNDERSTANDING THE MANUAL? WHERE? YES NO

WERE THE COMPONENTS OF REASONABLE QUALITY? WHAT WASN'T? YES NO

HAVE YOU COMPLETED THE CONSTRUCTION YET? YES NO

HAVE YOU TESTED THE INTERFACE YET? YES NO

WHAT IS THE MAKE AND MODEL OF YOUR RECORDER?

HAVE YOU ENCOUNTERED ANY PROBLEMS? WHAT? YES NO

IF THERE WERE PROBLEMS, ARE THEY FIXED? IN WHAT WAY? YES NO

HAVE YOU FOUND ANY OF THE ADJUSTMENTS CRITICAL? WHICH? YES NO

ARE YOU DISATISFIED IN ANY WAY WITH THE UNIT? HOW? YES NO

DO YOU HAVE ANY SUGGESTIONS FOR IMPROVEMENT OF THE DESIGN YES NO

WHAT DO YOU FEEL IS THE MAJOR DISADVANTAGE OF THIS UNIT?

HAVE YOU DEVELOPED ANY SOFTWARE OR HARDWARE RELATIVE TO THE INTERFACE THAT MIGHT BE OF USE TO OTHER PEOPLE? WHAT?

HAVE YOU TRIED PACKING DENSITIES HIGHER THAN THE STANDARD 800 BITS/INCH? WHAT DENSITY? WITH WHAT RESULTS?

WHAT IS YOUR NAME, ADDRESS, AND PHONE NUMBER?

DO YOU MIND IF I GIVE OUT YOUR NAME TO OTHER CASSETTE USERS?

PLEASE USE THIS SPACE, AND THE REVERSE SIDE, IF NECESSARY, TO MAKE ANY COMMENTS ON THE INTERFACE THAT MAY BE HELPFUL.

SINCERELY,

Donald E. Tarbell

DONALD E. TARBELL
144 MIRALESTE DRIVE #106
MIRALESTE, CALIF. 90732