BUILD YOUR OWN cortex 16 bit, 64k RAM colour computer

With this powerful machine (featured in Electronics Today International as a constructional project) you have access to highly advanced systems and software developed specially by MPE Ltd for the CORTEX. For business, education, R & D – or simply increasing your knowledge and understanding of computers – it beats comparably priced off-the-shelf machines hands down!

STATEMENTS
IF
ELSE
ON
GOTO
GOSUB
POP
REM
FOR
NEXT
ERROR
INPUT

PRINT
;
?
*
1 UNIT
BAUD
CALL
DATA
READ
RESTOR
RETURN
STOP

TIME
WAIT
SAVE
LOAD
MOTOR
ESCAPE
NOESC
RANDOM
ENTER
LIST
PURGE
NUMBER

RENUM
BOOT
GRAPH
TEXT
PLOT
UNPLOT
COLOUR
CHAR
SPRITE
SHAPE
SPUT
SGET

MAG
TOF
TON
DIM
LET
DEF
NEW
END
BIT
CRB
CRF
MEM

MWD
BASE

COMMANDS
RUN
SIZE
CONT
MON

DELIMITERS
TO
TAB
STEP
THEN

: @ # : ? % $ ' ! &

( )
[ ]

FUNCTIONS
FNA-FNZ
ABS
ADR
ASC
ATN
SIN
COS
EXP
FRA

INT
LOG
SQR
SYS
TIC
SGN
BIT
CRB
CRF
MEM
MWD
LEN
MCH

POS
COL
MOD
RND
KEY

OPERATORS
OR
LOR
AND
LAND
NOT
LNOT
LXOR

= = = > >= < <= <> - + / * ^

Standard features –

- High speed 24K byte extended basic interpreter
- Powerful TMS9995 16 bit microcprocessor
- 48 bit floating point gives 11 digit accuracy
- High resolution (256 x 192) colour graphics
- Screen memory does not use up user memory space
- 16 colours available on the screen together in graphic mode
- Fast line drawing and point plotting basic commands
- High speed colour shape manipulation from basic
- Full textual error messages
- String and Array size limited only by memory size
- Real time clock included in basic
- Interval timing with 10mS resolution via TIC function
- Named load and save of basic or machine code programs
- Auto-run available for any program
- Powerful machine code monitor
- Assembler and Disassembler included as standard
- Auto line numbering facility
- Full renumber command
- Simple but powerful line editor
- Flexible CALL statement allows linkage to machine code routines with up to 12 parameters
- Basic programs may contain spaces between key words to make programs readable without using more memory
- Over 34K bytes available for basic programs
- Extended basic includes IF-THEN-ELSE
- Interfaces for screen and cassette included.
- Supports bit manipulation of variables from basic
- Error trapping to a basic routine included
- Basic supports Hexadecimal numbers
- Separate 16K video RAM for graphics

Self assembly kit

£295

All prices exclusive of VAT. Carriage paid.

Optional Extras

Floppy disc interface electronics	£86.50
Hardware kit & connectors for disc drives	£49.50
RS232C interface kit	£9.20
Pair of 5¼" disc drives (SS)	£300.00
Pair of 5¼" disc drives (DS)	£590.00

Full assembly instructions and 216 page users manual.

All goods subject to availability.

POWERTRAN cybernetics ltd.

Portway Industrial Estate, Andover SP10 3ET. Tel: 0264 64455

to POWERTRAN CYBERNETICS LTD, Portway Industrial Estate, Andover, Hants SP10 3ET.

Please send me _____

_____

I enclose cheque for _____ or charge to: Access VISA

Access/ Barclaycard A/C No. _____

Allow 21 days for delivery

Name _____

Address _____

_____ Tel. _____

CORTEX CENTRONICS INTERFACE

Any follow-up on the Cortex is long overdue, for which we apologise. We're now about to start setting this to rights.

The Cortex 16-bit computer was designed in late 1982 and was featured by ETI in the November 1982, December 1982 and January 1983 editions. It is based on the TMS9995 16-bit microprocessor with a full 64K byte main memory and a separate 16K byte memory for the colour graphics display. The machine can be expanded by simply adding chips to provide features such as floppy disc support and an expansion bus for extra memory, input and output. This article details how to use the EBUS bus expansion to plug an external board into your Cortex to provide a parallel data interface for a Centronics printer.

The E-BUS was developed to provide a compact, high-performance and flexible interface for both memory and input/output expansion. The E-BUS system multiplexes both addresses and data information onto its lines at different times, so as to keep the number of lines needed as low as possible. It can be shared by multiple microprocessors for access to common memory or I/O.

With regard to I/O expansion, some explanation of the computer's operation is necessary. When doing input and output operations, the CPU does not communicate in eight or 16-bit words, in contrast to most common processors which support dedicated I/O commands. Instead, the 9995 sends and receives just *one* bit at a time. This means that multi-bit input and output functions, like those needed for a Centronics printer for instance, have to be built up or broken down one bit at a time. However, this operation is made simpler by the fact that each bit can be sent to or received from a unique address. So, to output eight bits to a peripheral, it is necessary to transfer the bits to a latch or similar device with eight successive single bit operations. Only then can the peripheral be told that the data is ready to be acted upon.

An input cycle occurs when the MEMEN (memory enable) control signal goes high (inactive); the bit address is output on the address/data bus and the single bit of data is sampled on the CRUIN signal. An output cycle is similar, except that the data bit is output on the CRUOUT signal and the CRUCLK control signal pulses active low. For multi-bit transfers, the cycle repeats with the address incrementing each time. A timing diagram for this is shown in Fig. 1.

During an I/O operation, the multiplexed address/data lines are forced to output the address throughout the I/O cycle so no address latches are required, consequently the ALATCH (address latch) control signal remains high. The data bit to be sent appears as the signal CRUOUT on the least significant address line (A15 by the TI convention) when an output operation is performed. The remaining 15 address lines define 32768 addresses to which it can be sent, although the Cortex and the 9995 CPU use some of these addresses internally (see Fig. 2, the Cortex I/O map).

A data bit to be read in is taken via the CRUIN line. Up to 16 bits can be input or output by a single instruction, the specified number of bits being processed serially to or from the target addresses.

The E-BUS

The kernel of the E-BUS interface on the Cortex is the 74LS-2001 gate array. This device is used to control access to the bus

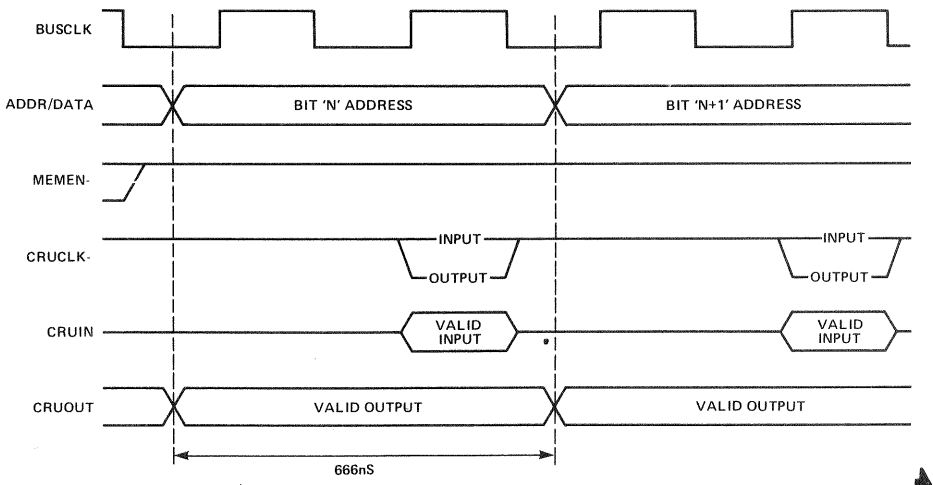


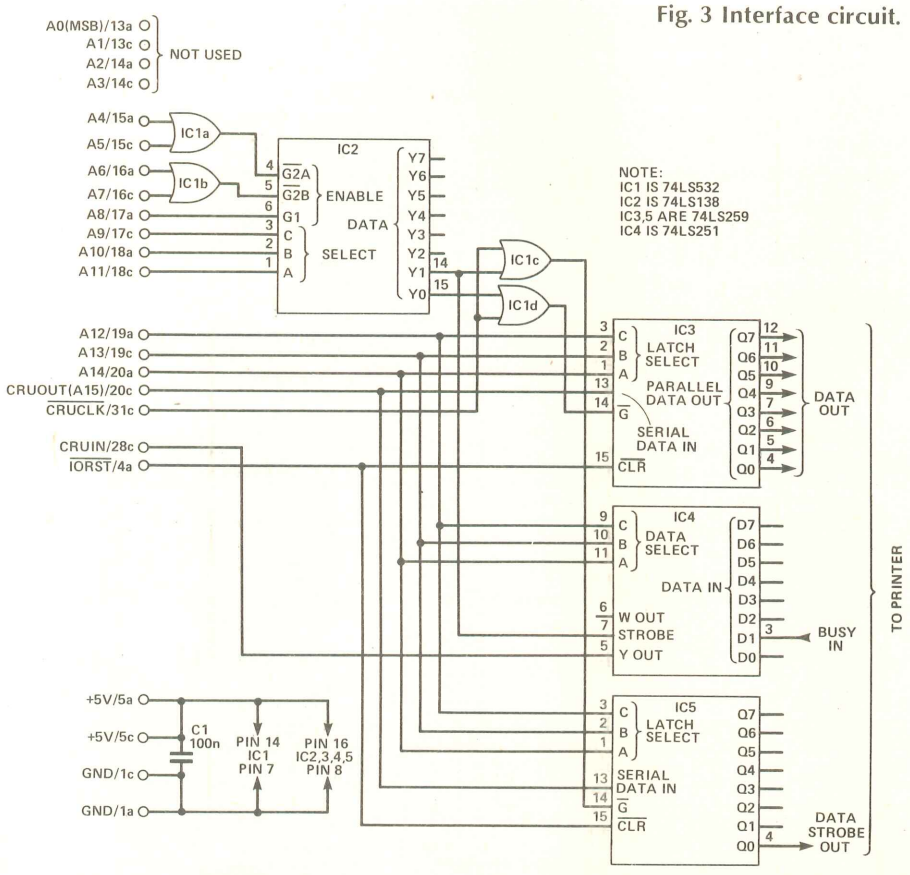
Fig. 1 Timing diagram for multi-bit transfers.

Address	Input Function	Output Function
0000	NOT USED*	"BASIC" LED
0002	NOT USED	KEYBOARD ACKNOWLEDGE
0004	DISK SIZE JUMPER	EBUS INTERRUPT ACK
0006	DISK DENSITY JUMPER	EBUS TIMEOUT ENABLE
0008	FLOPPY INTERRUPT FLAG	DISK SIZE
000A	KEYBOARD INTERRUPT FLAG	EPROM ON/OFF
000C	DISPLAY INTERRUPT FLAG	"BELL" ENABLE
000E	EBUS TIMEOUT INTERRUPT FLAG	NOT USED
0010 to 001E	KEYBOARD DATA	DUPLICATE OF ABOVE
0020 to 003E	DUPLICATE OF ABOVE	DUPLICATE OF ABOVE
0040 to 007E	NOT USED	NOT USED
0080 to 00BE	TMS 9902 RS232 PORT	TMS 9902 RS232 PORT
00C0 to 017E	NOT USED	NOT USED
0180 to 01BE	TMS 9902 CASSETTE PORT	TMS 9902 CASSETTE PORT
01C0 to 01FE	TMS 9911 DMA CONTROLLER	TMS9911 DMA CONTROLLER
0200 to 07FE	external via EBUS	external via EBUS
0800 to 080E	external via EBUS	Centronics data via EBUS
0810	external via EBUS	Centronics strobe via EBUS
0812	Centronics status via EBUS	external via EBUS
0814 to 1EDE	external via EBUS	external via EBUS
1EE0	CPU internal	CPU internal timer control
1EE2	" "	" " timer interrupt
1EE4	" " level 1 interrupt flag	" " enable
1EE6	" " " 3 " "	" "
1EE8	" " " 4 " "	" "
1EEA	" " **	" " **
1EEC	" " ***	" " ***
1F00 to 1FD8	" " not used	" " not used
1FDA	CPU internal MID flag	CPU internal
1FDC to FFFE	external via EBUS	external via EBUS

* Some systems may have this used for floppy disk interface
** Cortex Basic uses this bit to enable the display on the TV screen of all control characters.
*** Cortex Basic uses this bit to disable the scrolling of a text display on the TV screen.

Fig. 2 (above) Cortex I/O map.

Fig. 3 Interface circuit.



for multiple microprocessors and to synchronise all data transfers, as well as to provide time-out controls to avoid a permanent lock-up of the bus. However, for simple expansion of the Cortex facilities, it is possible to avoid using this device. Four changes to the main PCB have to be made, as follows:

1. Cut the connection to IC99 pin 18 and connect this pin to IC94 pin 11;
2. Connect IC94 pin 9 to IC11 pin 13;
3. Connect IC94 pin 19 to IC27 pin 2;
4. Connect IC89 pin 18 to IC89 pin 19.

For this modification to work, the

HOW IT WORKS

IC1a, IC1b and IC2 decode address bits A4 to A11 inclusive. Two outputs from IC2 are used each of which will go low when one of their particular group of eight addresses is accessed. Note that the address decoding is not complete and may well respond at other locations (2K interval). The Y1 output from IC2 (address 0088h to 008Fh) enables the output of IC4 (74LS251) the 1 of 8 selector. This device selects the signal on the input specified by the least significant address lines and passes it to the CRUIN line. This would normally be the printer status line in this case and its state will signal whether the printer is ready to receive data.

The data to be output to the printer appears on the CRUOUT line in serial form. This has to be converted to parallel form before being presented to the printer. This is done by IC3 which is an eight bit addressable latch (74LS259). Address lines A12, 13 and 14 determine which bit is to be written into IC3 and the output Y0 of IC2 gated with CRUCLK in IC1d actually causes it to happen. To write a complete eight bits of new data requires eight output cycles. Once this has been done the data is ready to be acted upon by the printer. IC5 is another 74LS259 like IC3 and one bit of its output is used as a strobe signal to tell the printer that the data is ready. Output Q0 is used in this case; it is normally low and is pulsed high under software control. This happens when a 1 and then a 0 is output to 0088h.

For the transfer of one character the sequence is as follows: eight bits of data are written into the eight locations in IC3. The data then appears in parallel form at its output and is available to the printer. The DATA STROBE is taken high and then low to signal to the printer that valid data is ready. The printer signals that is is BUSY by taking the BUSY STATUS line high. This is read by the computer and no further action is taken until it goes low again. Now new data can be sent to IC3 and the process repeated.

Note that some printers give a low on the status line to indicate the BUSY condition, this can be accommodated by connecting the EBUS CRUIN line to IC4 pin 6 instead of pin 5. More difficult is when the STROBE has to be inverted, this requires an additional inverter.

following devices must be present on the PCB (if your Cortex does not have all the options available, then some of these may be missing, so you will need to check): IC90 (74LS08), IC91 (74LS32), IC92 (74LS74A), IC93 (74LS00), IC94,95,96 and 99 (all 74LS244).

If you wish to use memory expansion, then you must break the links next to IC26 and drill out the shorting links next to IC94, and ensure that you have IC97 (74LS245) and IC26 (74LS612). Note that all the component numbers used here are those given in the original Cortex article.

It's Already There

Although the Cortex has an RS232 interface for a printer or terminal, it also has the software necessary to drive a parallel data printer port. The hardware necessary to implement this was left off the main PCB to save space. Building the interface described here will therefore free the RS232 for other tasks, and make it possible to use the more common Centronics printers.

The circuit diagram for the interface is shown in Fig. 3 and the circuit itself is discussed in the 'How It Works' section.

I/O	ROW A	PIN	ROW C	I/O
	GND 0V	1	GND 0V	
I/O	PRES- (RESET-)	2	BUSCLK (3MHz)	O
	+12V	3	−12V	
O	IORST- (I/O RESET-)	4	NMI-(NON-MASKABLE INT)	I
	+5V	5	+5V	
	—	6	—	
	—	7	—	
	—	8	—	
	—	9	—	
I	INTEN- (LEVEL 1)	10	ALATCH (ADDRESS LATCH)	O
O	XA0 (MSB ADDRESS)	11	XA1	O
O	XA2	12	XA3	O
O	A0	13	A1	O
O	A2	14	A3	O
O	A4	15	A5	O
O	A6	16	A7	O
I/O	A8/D8 (DATA MSB)	17	A9/D9	I/O
I/O	A10/D10	18	A11/D11	I/O
I/O	A12/D12	19	A13/D13	I/O
I/O	A14/D14	20	A15/D15/CRUOUT (LSB)	I/O
	—	21	MEMEN- (MEMORY ENABLE)	O
O	DEN- (MEMORY READ)	22	READY- (MEMORY READY)	I
I	GRANTIN	23	GRANTOUT	O
	—	24	BUSY- (BUS BUSY)	I/O
	GND	25	GND	
	—	26	—	
	—	27	—	
	—	28	CRUIN (I/O INPUT)	I
O	WE- (MEMORY WRITE)	29	—	
	+5V	30	+5V	
	—	31	CRUCLK (I/O WRITE STROBE)	O
	GND	32	GND	

Fig. 4 EBUS signals and their uses.

Next month we will publish the PCB, overlay, parts list and buylines.

ETI