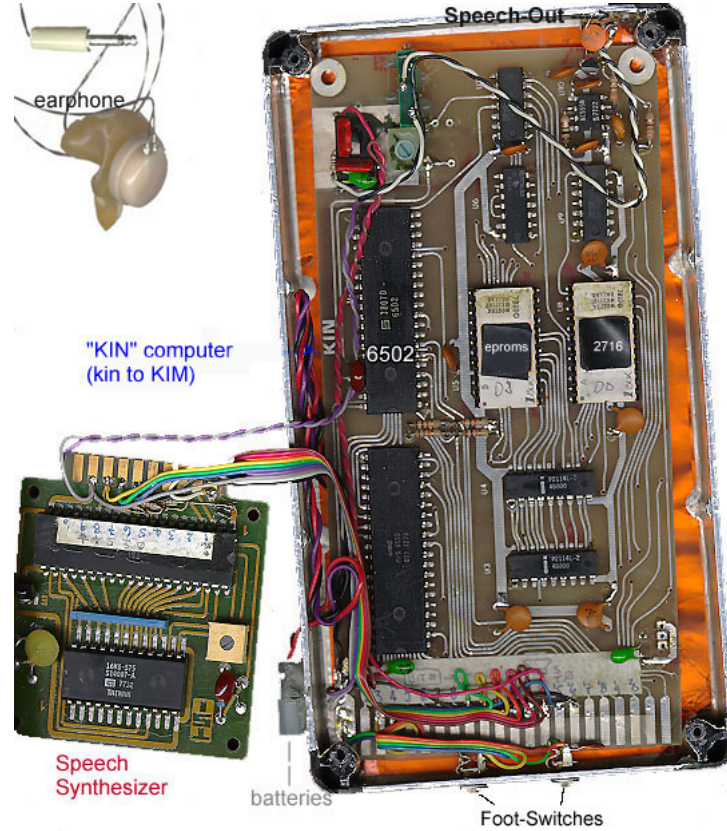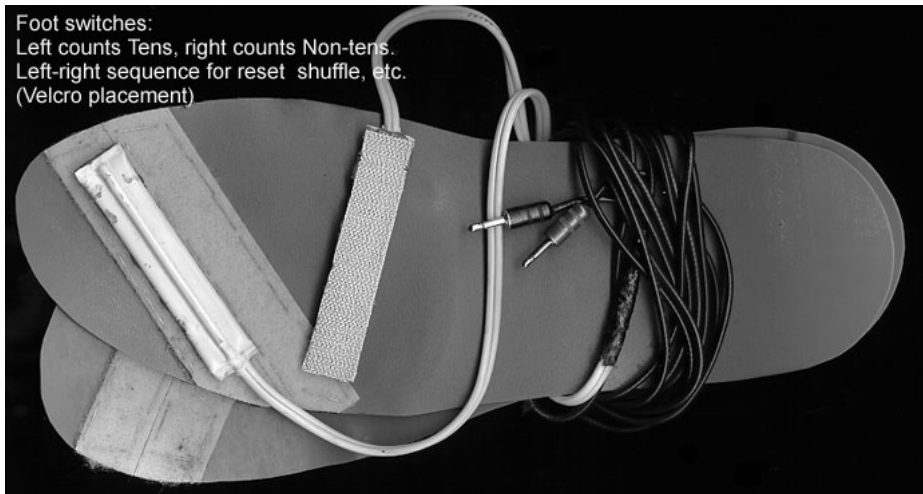The Kim-1 computer was cloned and a speech synthesizer was added... this became the KIN^{TM} computer.

The KIN microcomputer -- (kin to KIM)

Foot switches:
Left counts Tens, right counts Non-tens.
Left-right sequence for reset shuffle, etc.
(Velcro placement)

Foot switches.

Development

Top: Special ear mold to receive auditory suggestions from the blackjack computer.

Right: This shows the KIM computer in its case (center), the Keyboard and TV monitor (left) small audio amplifier on top of the monitor; two cassette recorders (right), KIM (center) and test instruments on the shelf (rear). The ribbon printer, made in SLC, USA, is at far left.
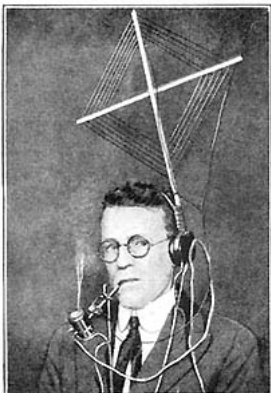Ca. 1983   Photo: a MINOX picture.

A later view (ca 1986) of the laboratory in which the Black Jack project was done... Now the lab has geared up to tackle the Roulette problem.

Bowl of Corn-Cob Pipe Holds Radio Set

THE most compact radio receiving set that has made its appearance is built on the bowl of a corn-cob pipe. It is the work of F. E. Wi'son, of Detroit, who can fill his pipe at the close of day and settle back for a smoke while he tunes in the radio stations that are "in the air."

For an aerial, Mr. Wilson uses a small loop attached to the headband of his 2000-ohm phone. The tuning coil is made of 100 turns of enameled wire—No. 26—wound around the bowl of the pipe. A piece of galena is balanced on the pipe stem, while the catwhisker is arranged to pivot on the stem. This small set has been remarkably successful in picking up concerts within 10 miles of the broadcasting stations.
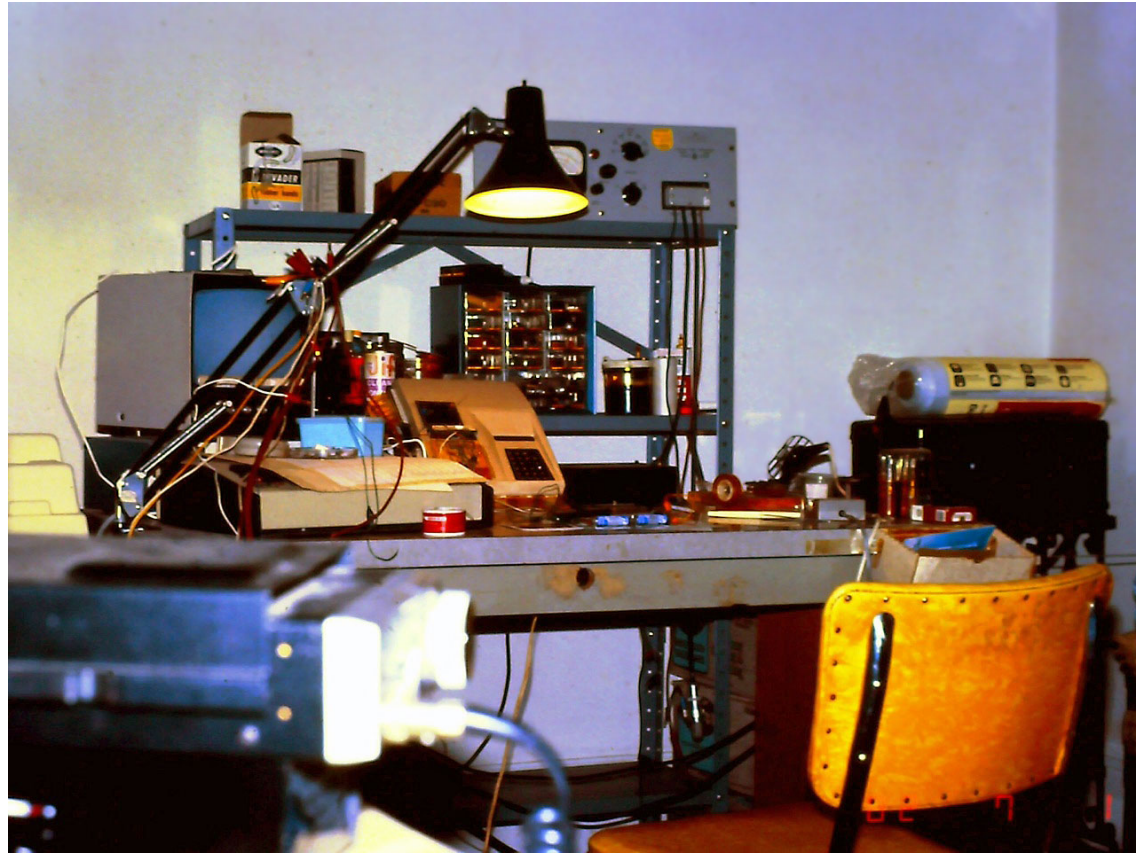
Danger - 2,000 Ohms!

Pocket of T-Shirt Holds Computer

THE development of KIN computer ca. 1983. I am wearing the KIN Black Jack computer while holding a spare.

Also shown is the KIM-1 computer, attached to keyboard and monitor.

Photo Joyce Stonehocker Enlarge

Blackjack development lab using the Kim-1 computer. (Click ' tap on images to enlarge)

Link   Seeing with your ears - debunked!
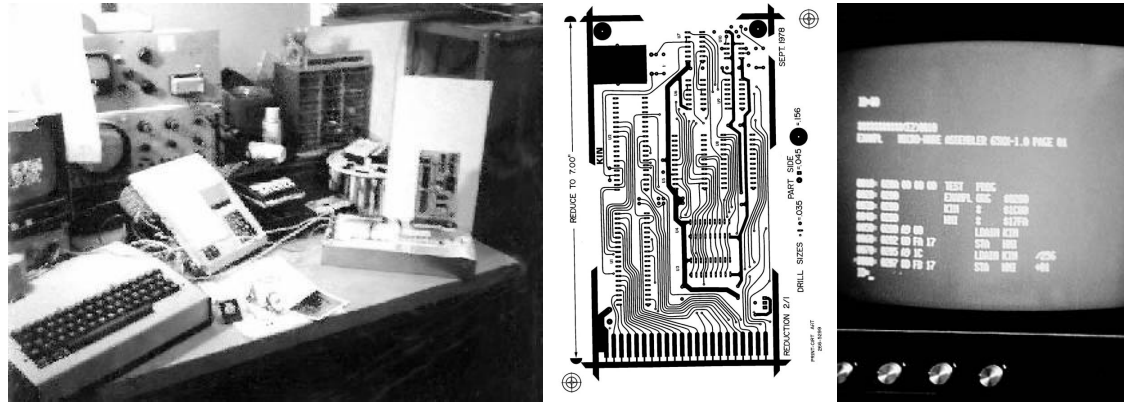
~@Com~   On the 'Net since 1994

The KIM & KIN Computers   Email

The KIN was constructed by Martek International, Salt Lake City, Utah, USA.

Links: Photo Gallery | Blackjack Project | Roulette Project | Roulette Timing | KIM-1 (6502) | Correspondence

Blackjack development lab using the Kim-1 computer. (Click / tap on images below to enlarge)

Development of KIN computer ca. 1983. Test-kludge box for KIN is at right, the KIN is inserted. Also shown is the KIM-1 computer in its plastic housing, attached to a keyboard and monitor. The KIN is inserted into socket on a test-kludge box (right); KIM is in its housing (center). Ca. 1978. Photo (left): MINOX 8mm film
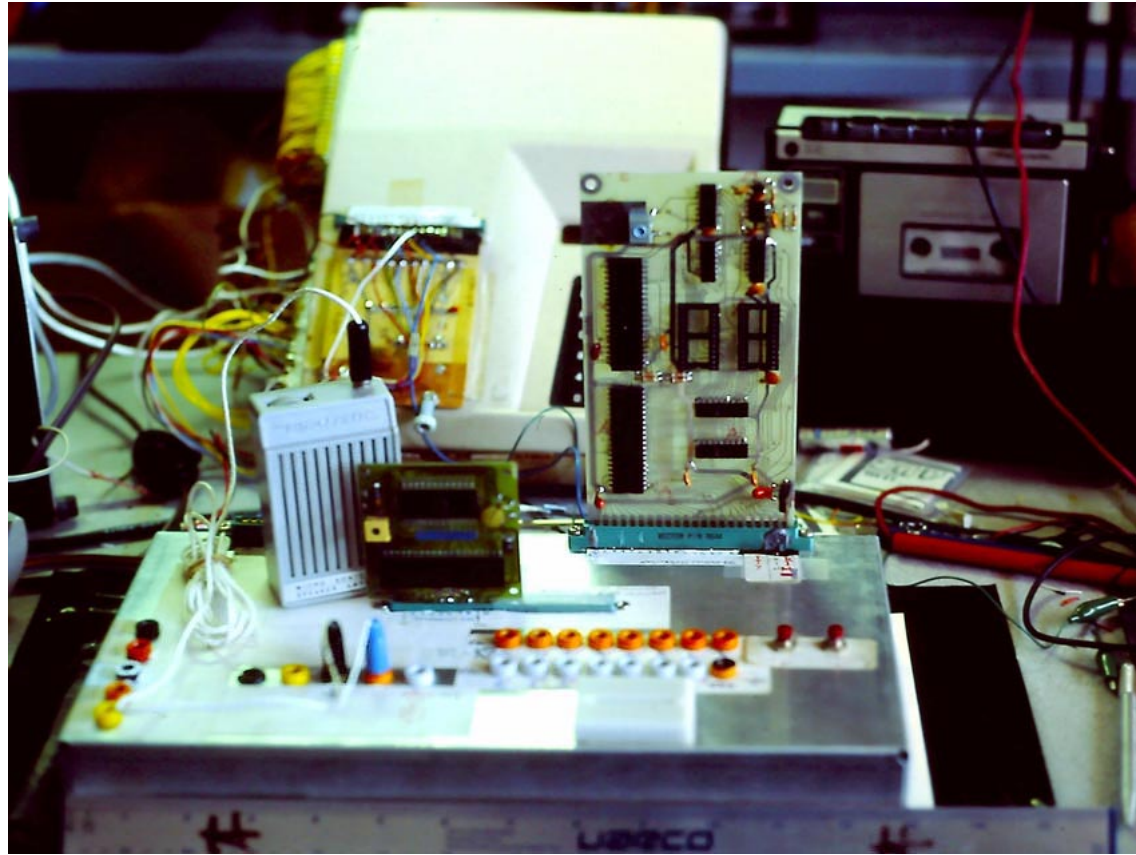
The KIN is inserted into socket on a test-kludge box; The KIM-1 computer is in its housing (rear).
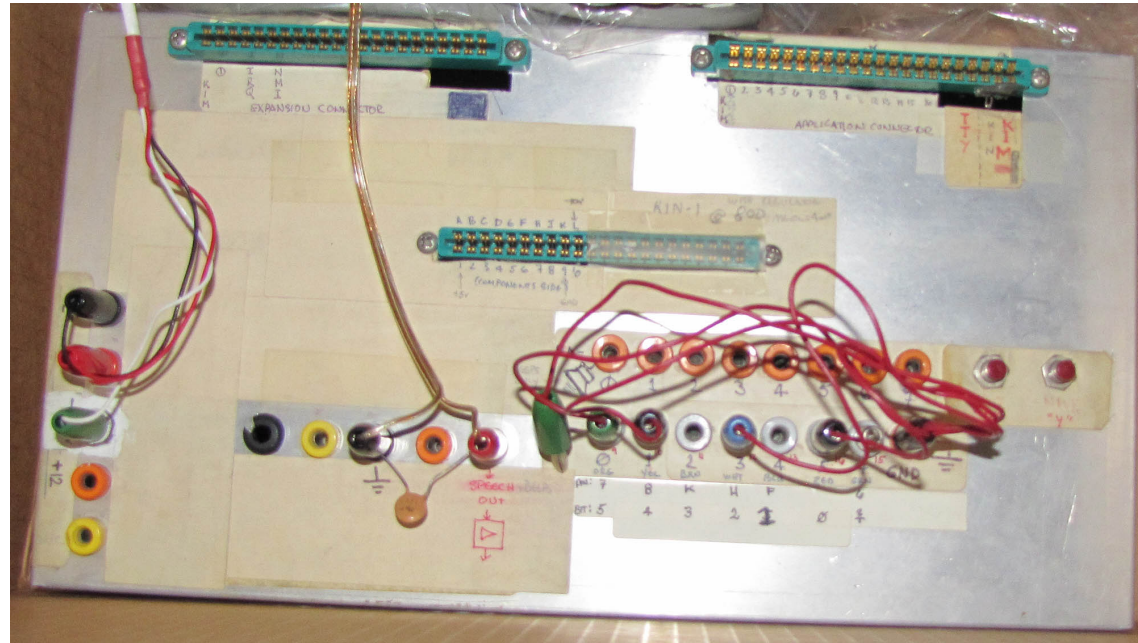
The KIN's test-kludge (click image to enlarge)

Blackjack development lab using the KIN-1 computer.   (Click ' tap on images to enlarge)
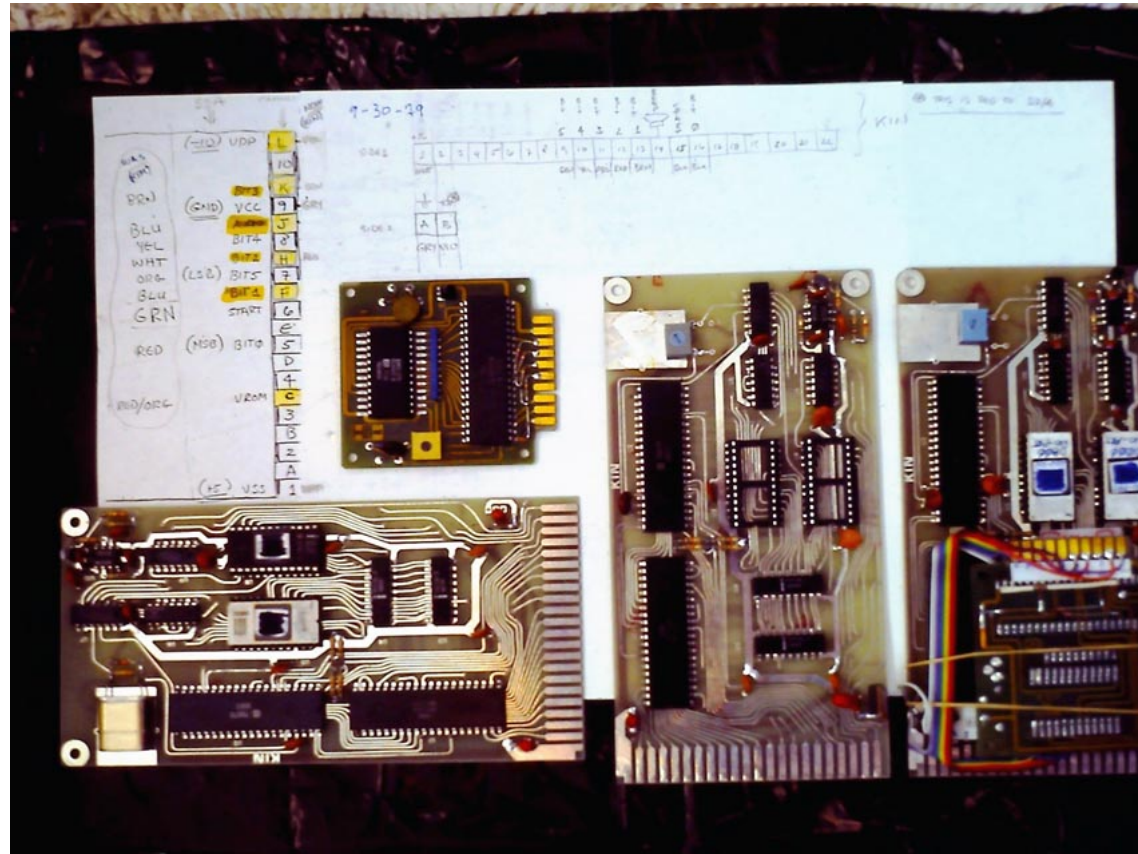
A speech synthesizer (top left) and three KIN computers below.
A speech synthesizer is wired to the third KIN (right) and folded over it..
(The middle KIN is without 2716 EPROMs)
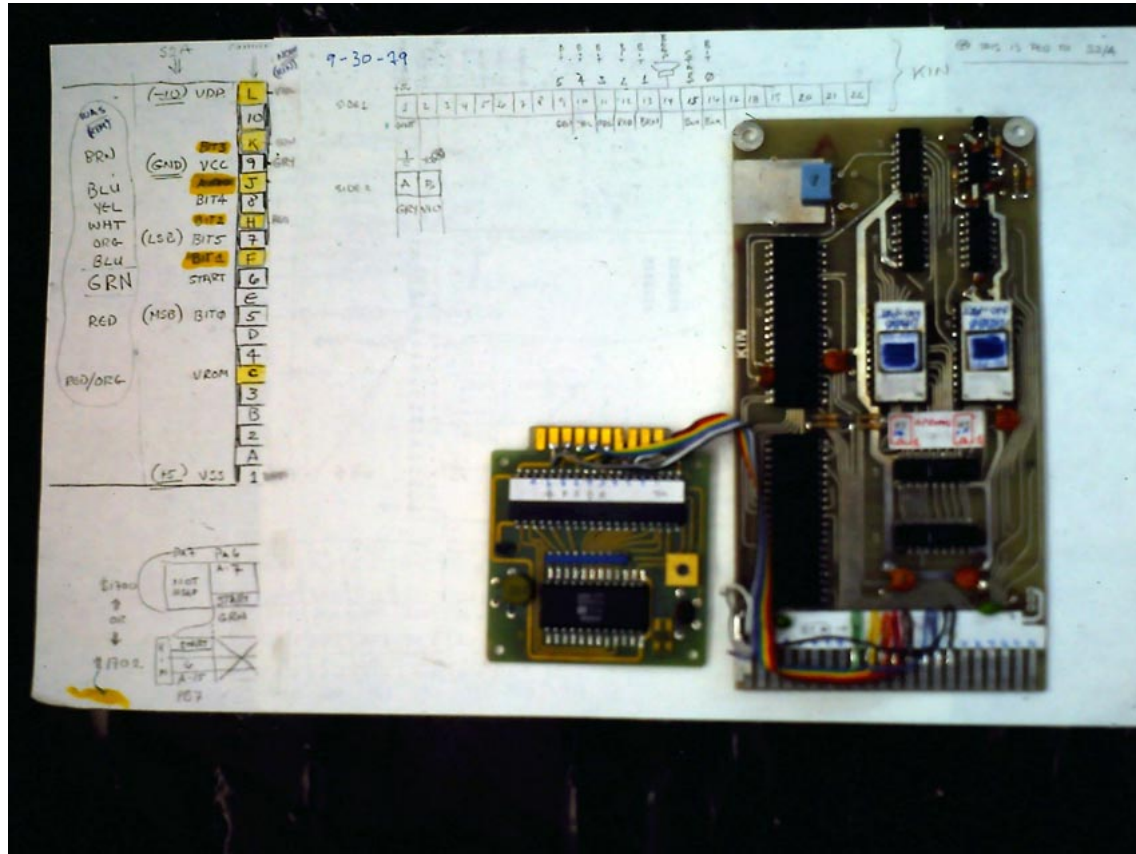
Speech synthesizer (left) wired to a KIN computer.

The Kim-1 in a test enclosure (right); printer document and papers (left.)

Ron Ward and Larry Walker of Martek International, Salt Lake City, made this, the KIN computer form schematics of the KIM:
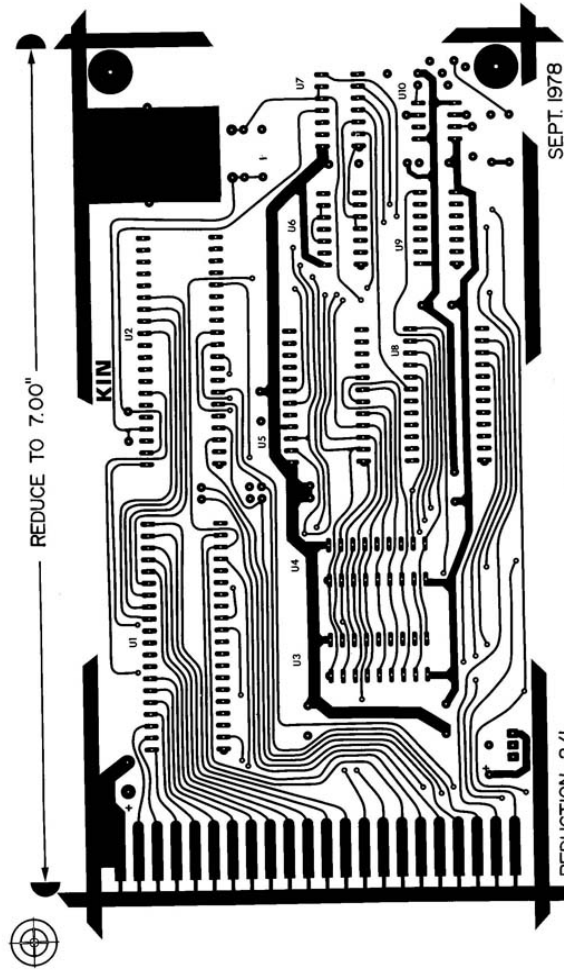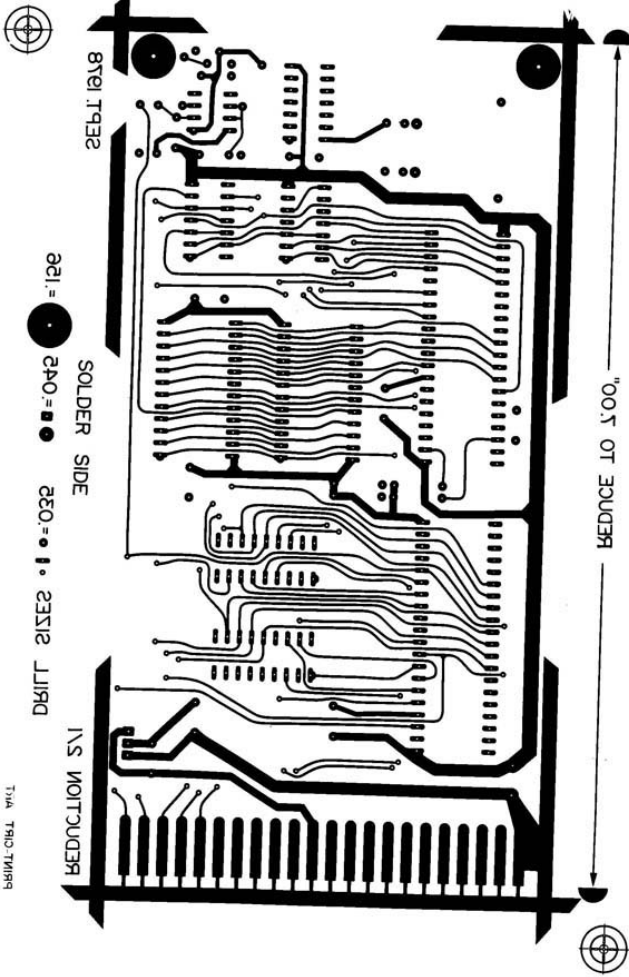
Click or tap on images to enlarge
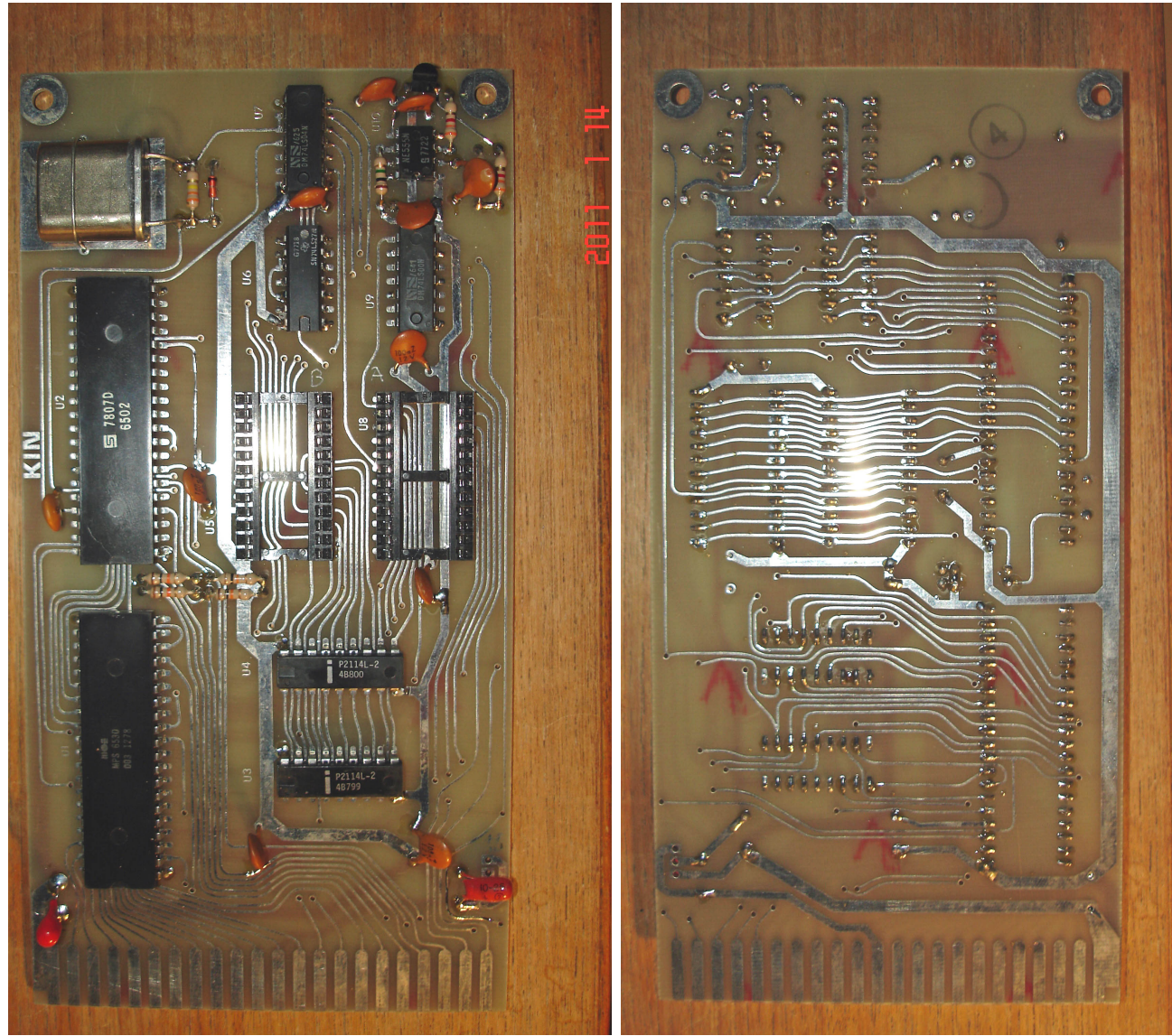
Schematic diagrams for the KIN computer -
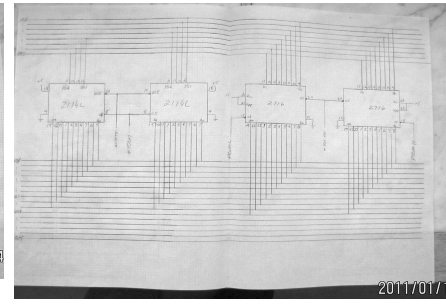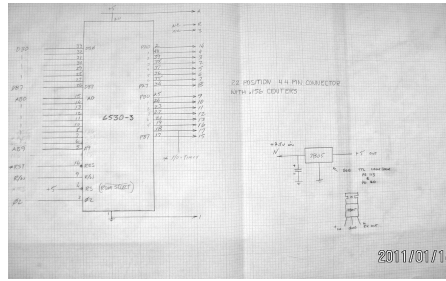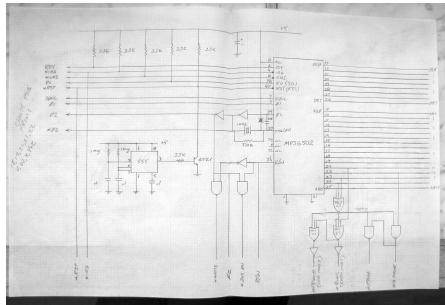
A mild digression...

Elephant Parts by Michael Nesmith
★★★★★ Customer reviews

Picture taken from *Elephant Parts* by Michael Nesmith
Link You Tube video

```
L
0010:              J U M P   T A B L E
0020:
0030:         PATCHES & VECTORS :
0040:         THE PATCHES (JUMP TABLE) IS
0050:         USED DURING DEBUGGING
0060:         TO FACILITATE EASIER
0070:         RELOCATION OF MODULES.
0080:         (YOU JUMP TO A LOCATION
0090:          IN THE JUMP TABLE, AND
0100:          THAT LOCATION JUMPS TO
0110:          WHERE YOU WANT TO GO.
0120:          DURING TESTING, THE JUMP
0130:          TABLE MUST BE IN READ/WRITE
0140:          MEMORY, TO BE USEFUL)
0150:         THE MODULES ARE LISTED BELOW:
0160:
0170:         JUMP TABLE :    (3 BYTES EA)
0180:         1) BET          D1C0   1ST 1/2
0190:         2) ENTERS       D266   1ST 1/2
0200:         3) TO 2ND HALF  D370   2ND 1/2
0210:         4) CKSW         D1FD
0220:         5) MORCRD       D570
0230:
0240:         'KIN' VECTORS   (2 BYTES EA)
0250:         6) NMI          2FFA, 2FFB
0260:         7) RST          2FFC, 2FFD
0270:         8) IRQ          2FFE, 2FFF
0280:
0290:
0300: JMPTBL ORG $DFEB  (FIRST LOC.)
0310:              P O I N T E R S
      :
0320:    = $4C    (JMP)   DFEB  BET
0330:    = $C0    (LO)
0340:    = $D1    (HI)
0350:
0360:    = $4C            DFEE  ENTERS
0370:    = $66
0380:    = $D2
0390:
0400:    = $4C            DFF1  TO 2ND 1/
2
0410:    = $70
0420:    = $D3
0430:
0440:    = $4C            DFF4  CKSW
0450:    = $FD
0460:    = $D1
0470:
0480:    = $4C            DFF7 SUBR. MORC
RD
0490:    = $70
0500:    = $D5            <-- LAST LOC.
0510:
0520:    NMI  (NOT USED)
0530:    = $00
0540:    = $20
0550:
0560:    RST  (JUMP TO 2000
0570:          ON POWER ON)
0580:    = $70   (LO)  PROGRAM
0590:    = $D1   (HI)  START LOC.
0600:
0610:    IRQ  (NOT USED)
0620:    = $00
0630:    = $20
0640: @
```

KIM-1 program listing. Assembly language code for the
software of the Blackjack machine.

```
0470:   CPYIM $01
0480:   BEQ ADDFAC
0490:   CPYIM $02
0500:   BEQ ADDFAC
0510:
0520:   SUBFAC
0530:   SEC
0540:   LDA TOTFAC
0550:   SBCIY ZBASE
0560:   STA TOTFAC
0570:
0580:  LOW_TF
0590:   LDA TOTFAC
0600:   BPL HI_TF
0610:
0620:  DECACC
0630:   CLC
0640:   LDA NEGACC
0650:   ADCIM $01
0660:   STA NEGACC
```

```
1540:   = $FF
1550:   = $FF
1560:   = $FF
1570:   = $FF
1580:   = $FF
1590:   = $FF
1600:   = $FF
1610:
1620:  PRT_K
1630:   RTS
1640:   RTS
1650:   CMPIM $03
1660:   BNE GOPRTV
1670:   RTS
1680:  GOPRTV
1690:   LDAIM $0D  CARRIAGE RETURN
1700:   JSR OUTCH
```

LDA KEY
JX ON REV G.0 -TAPE
(NOT ON 01-03)

PRT LOC 50 - 65
WAS LDA KEY
DO NOT PRINT

EXIT IF KEY=3

The code runs in 8K of eprom.
There is quite a bit of it.

```
0470: JSR BT * $0221
0480: JSR CO * $0778
0490: SHUFF * $D085
0500: DECOT * $DA2C
0510: INCOT * $DA23
0520: C_REV
0530: NOT_REV              PRODUCTION
0540:                      9-22-85
0550: ORG $D370  SEC 1/2 PROB
0560: BT * $162F  KIM - NOT KIM   03
0570: RY_ADJ * $DE07
0580: RD_ADJ * $DE2F         LOADS 200-3CF
0590: ...  * $DE4E
0080: RE_ADJ * $DE95          0  00
0090: ILLIG * $DEA2           1  02
0100: INITC * $D9B8           2  90
0110: PRDAT * $D915           3  D3
0120: UNHOLD * $DB45          4  DO
0130: DECTEN * $DA09          5
0140: INCTEN * $DA00
0150: MULTI1 * $D866
0160: LOWERR * $DA50       EPROM I 03
0170: ... * $DA2F
0180: CALCAR * $D833       EXIT
0190: SIZBET * $D995
```

Here are some images from the Black Jack project ca. 1975 - 1978.

Early development for the Black Jack prediction project - Surplus keyboard, shown, ca. 1977 (left).

KIM-1 program listing on the monitor (center); me, in the lab.

Adhoc Notes -

RON WARD - Principle

LARRY WALKER - Technician

MARTEK INTERNATIONAL (Bob Brickey)

MARTEK ENGINEERING

3572 So. 500 West

801- 262 1552

- Revision on schematics

- PC LAYOUT
(Orlene at Printset Art)

- Parts Procured
And Board Assembled
and Debugged.

1 801 557 212

2000   STARTING LOCATON

22
21 F0

2 FFC (LO)   00   STARTING LOC
2 FFD (HI)   20   VECTOR

0500:	NMI (NOT USED) 2FFA, 2FFB
0510:	= $00
0520:	= $20
0530:	
0540:	RST (JUMP TO 2000
0550:	ON POWER ON)
0560:	= $00 (LO) 2FFC, 2FFD
0570:	= $20 (HI)
0580:	
0590:	IRQ (NOT USED)
0600:	= $00 2FFE, 2FF
0610:	= $20
0620:	@

00 HOW MANY   FF
02 WHAT   17
02] WHAT FROM F0
03]   21

SEC

OUT
4:26

FROM   00
        01

TO   02
        03

WHAT   3 00

2000 - 3050 = 26 = (05 MANY)

17F5 00   FROM (LO)
17F6 01   (HI)
17F7 02   TO   (LO)
17F8 03   HI
17F9 04   WHAT

0219

last = 022E

9:35
out

0	00	00	00	00	00	00	00
1	20	28	30	38	02	02	C0
2	0	0	0	0	00	92	00
3	0	0	0	0	00	D0	00
4	00	0	00	00	00	17	00
5	28	30 2F	38	40	92 02	D1 C8

EPROM 1   EPROM 2

---

16 # THREAD - start work

HIRED - 2 weeks from Monday

come in, fill out
forms any time

MRS
MURDOCK
5733-6301

01 x3
02
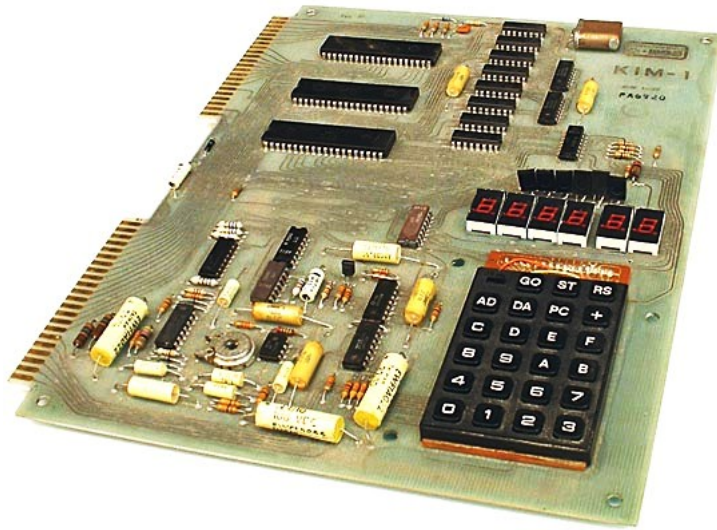
KIM-1 computer
The KIM-1 (**K**eyboard **I**nput **M**onitor)
micro computer is the grand daddy of
all 6502 microprocessors systems.
It was originally created by MOS
Technology as a development tool for
the 6502 chip.

The First Book of KIM, a Manual
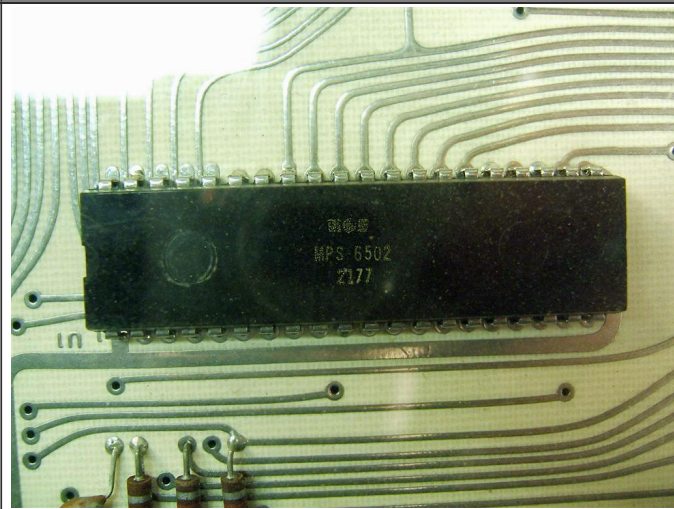
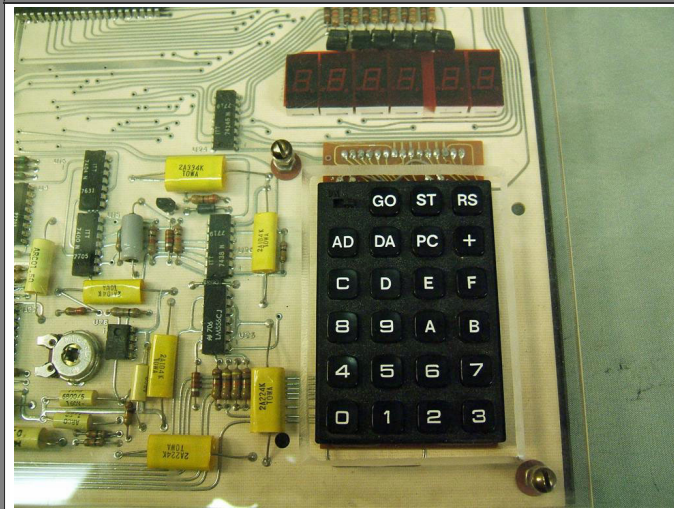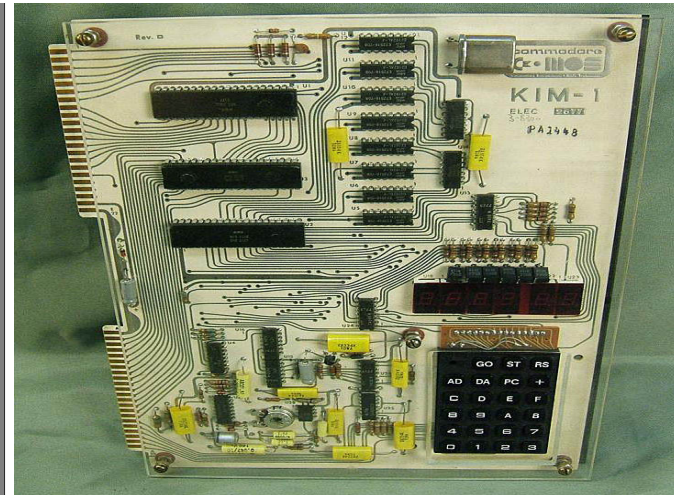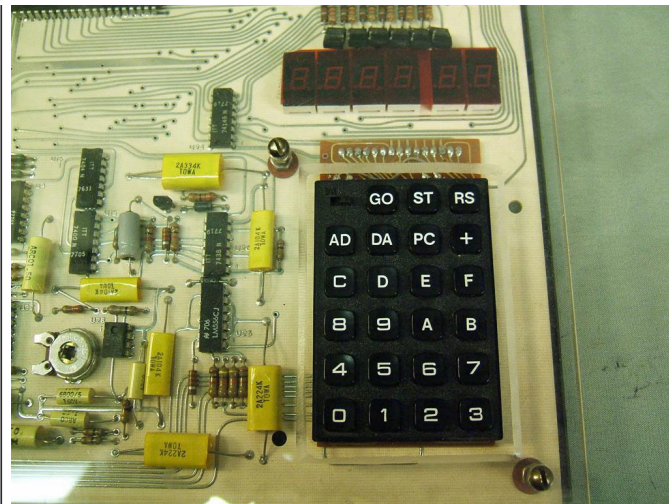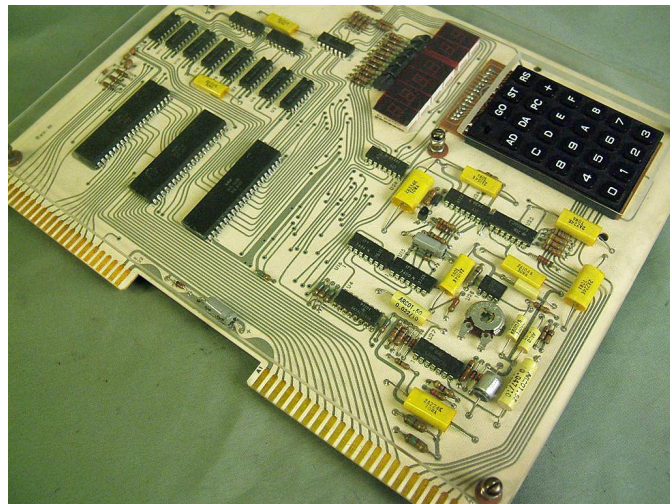| Please click / tap the images to enlarge. | In 2011 this pictured KIM-1 sold was on eBay - Link |
| A Plexiglas cover is fastened onto the circuit board, with a cut-out to allow access to the hexadecimal key pad. This configuration was commonly used in schools and universities. | |

A KIM-1 Emulator: Try it yourself -

Submitted by a reader, Jonathan H. (January 2015) - Here's an emulator I found:

The Amazing JavaScript Kim1!
http://www.robsayers.com/jskim1/

Links   The First Book of KIM
         Write programs in AppleSoft BASIC.

Just for fun, and for testing the newly-built KIN computer...
OF approximately 50 runs on my KIM-1, I was able to land the Lunar Excursion Module without crashing it -- maybe twice... it's goddamn difficult to land it !

```
   Lunar Lander -- A program for KIM / 6502  By JIM BUTTERFIELD

***** Hex Dump - Lunar Lander *****

0200 A2 0D BD CC 02 95 D5 CA 10 F8 A2 05 A0 01 F8 18
0210 B5 D5 75 D7 95 D5 CA 88 10 F6 B5 D8 10 02 A9 99
0220 75 D5 95 D5 CA 10 E5 A5 D5 10 0D A9 00 85 E2 A2
0230 02 95 D5 95 DB CA 10 F9 38 A5 E0 E5 DD 85 E0 A2
0240 01 B5 DE E9 00 95 DE CA 10 F7 B0 0C A9 00 A2 03
0250 95 DC CA 10 FB 20 BD 02 A5 DE A6 DF 09 F0 A4 E1
0260 F0 20 F0 9C F0 A4 A2 FE A0 5A 18 A5 D9 69 05 A5
0270 D8 69 00 B0 04 A2 AD A0 DE 98 A4 E2 F0 04 A5 D5
0280 A6 D6 85 FB 86 FA A5 D9 A6 D8 10 05 38 A9 00 E5
0290 D9 85 F9 A9 02 85 E3 D8 20 1F 1F 20 6A 1F C9 13
02A0 F0 C0 B0 03 20 AD 02 C6 E3 D0 ED F0 B7 C9 0A 90
02B0 05 49 0F 85 E1 60 AA A5 DD F0 FA 86 DD A5 DD 38
02C0 F8 E9 05 85 DC A9 00 E9 00 85 DB 60 45 01 00 99
02D0 81 00 99 97 02 08 00 00 01 01
(Red marks beginning and end of program code)
```

Description -
This program starts at 0200. when started, you will find
yourself at 4500 feet and falling. The thrust on your machine
is set to low; so you will pick up speed due to the force of
gravity.

You can look at your fuel at any time by pressing the
"F" button. Your fuel (initially 800 pounds) will be shown
in the first four digits of the KIM display.
The first two digits of the KIM display always show
your rate of descent or ascent. "A" restores altitude.
Set your thrust by pressing buttons 1 through 9.
Warning: button 0 turns you motor off, and it will not
reignite! A thrust of 1, minimum, burns very little fuel;
but gravity will be pulling your craft down faster and
faster. A thrust of 9, maximum, overcomes gravity
and reduces your rate of descent very sharply A thrust of 5
exactly counterbalances gravity; you will continue to descend
(or ascend) at a constant rate. If you run out of fuel,
your thrust controls will become inoperative.
A safe landing is considered to be one where you land
at a descent rate of 5 or less. After you land, your thrust
controls will be inoperative, since the motor is automatically
turned off; but you can still preff "F" to look at your fuel.
Pressing "GO" starts a new flight.

        Suggestions for a safe flight:

        (1) Conserve fuel at the beginning by pressing 1.
        You begin to pick up speed downwards.

        (2) When your rate of descent gets up to the 90's,
        you're falling fast enough. Press 5 to steady the rate.

        (3) When your altitude reaches about 1500 feet,
        you'll need to slow down. Press 9 and slow down fast.

        (4) When your rate of descent has dropped to 15 or 20,
        steady the craft by pressing 5 or 6.
        Now you're on your own.

```
; main routine - initialization
0200 A2 0D GO LDX #13 fourteen bytes
0202 BD CC 02 LP1 LDA INIT,X
0205 95 D5 STA ALT,X
0207 CA DEX
0208 10 F8 BPL LP1
; Update height and velocity
020A A2 05 CALC LDX #5
020C A0 01 RECAL LDY #1
020E F8 SED
020F 18 CLC
0210 B5 D5 DIGIT LDA ALT,X
0212 75 D7 ADC ALT+2,X add each digit
0214 95 D5 STA ALT,X
0216 CA DEX
0217 88 DEY
0218 10 F6 BPL RECAL next digit
021A B5 D8 LDA ALT+3,X hi-order .. zero ..
021C 10 02 BPL INCR .. or ..
021E A9 99 LDA #$99
0220 75 D5 INCR ADC ALT,X
0222 95 D5 STA ALT,X
0224 CA DEX
0225 10 E5 BPL RECAL do next addition
0227 A5 D5 LDA ALT
0229 10 0D BPL UP still flying?
022B A9 00 LDA #0 nope, turn off
022D 85 E2 STA DOWN
022F A2 02 LDX #2
0231 95 DD DD STA ALT,X
0233 95 DB STA TH2,X
0235 CA DEX
0236 10 F9 BPL DD
0238 38 UP SEC update fuel
0239 A5 E0 LDA FUEL+2
023B E5 DD SBC THRUST
023D 85 E0 STA FUEL+2
023F A2 01 LDX #1 two more digits to go
0241 B5 DE LP2 LDA FUEL,X
0243 E9 00 SBC #0
0245 95 DE STA FUEL,X
0247 CA DEX
0248 10 F7 BPL LP2
024A B0 0C BCS TANK still got fuel?
024C A9 00 LDA #0 nope, kill motor
024E A2 03 LDX #3
0250 95 DD LP3 STA THRUST,X
0252 CA DEX
0253 10 FB BPL LP3
; show alt, fuel, or messages
0255 20 BD 02 JSR THHRUST
0258 A5 DE TANK LDA FUEL fuel into registers
025A A6 DF LDX FUEL+1
025C 09 F0 ORA #$F0 plus F flag
025E A4 E1 LDY MODE
0260 F0 20 BEQ ST
0262 F0 9C GOLINK BEQ GO
0264 F0 A4 CLINK BEQ CALC
0266 A2 FE LDX #$FE
0268 A0 5A LDY #$5A
026A 18 CLC
026B A5 D9 LDA VEL+1
026D 69 05 ADC #5
026F A5 D8 LDA VEL
0271 69 00 ADC #0
0273 B0 04 BCS GOOD
0275 A2 AD LDX #$AD
0277 A0 DE LDY #$DE
0279 98 GOOD TYA
027A A4 E2 LDY DOWN
```

```
027C F0 04 BEQ ST
027E A5 D5 LDA ALT
0280 A6 D6 LDX ALT+1
0282 85 FB ST STA POINTH
0284 86 FA STX PONTL
; show rate of ascent/descent as absolute
0286 A5 D9 LDA VEL+1
0288 A6 D8 LDX VEL up or down?
028A 10 05 BPL FLY .. up, we're OK
028C 38 SEC
028D A9 00 LDA #0
028F E5 D9 SBC VEL+1
0291 85 F9 FLY STA INH
0293 A9 02 LDA #2 loop twice thru display
0295 85 E3 STA DECK
0297 D8 FLITE CLD display & key test
0298 20 1F 1F JSR SCANDS light 'em up!
029B 20 6A 1F JSR GETKEY check keys
029E C9 13 CMP #$13 GO key?
02A0 F0 C0 BEQ GOLINK ...yes
02A2 B0 03 BCS NOKEY .. if no key
02A4 20 AD 02 JSR DOKEY
02A7 C6 E3 NOKEY DEC DECK
02A9 D0 ED BNE FLITE
02AB F0 B7 BEQ CLINK to CALC
; subroutine to test keys
02AD C9 0A DOKEY CMP #$0A test numeric
02AF 90 05 BCC NUMBER
02B1 49 0F EOR #$0F Fuel F gives 0 flag
02B3 85 E1 STA MODE
02B5 60 RETRN RTS
02B6 AA NUMBER TAX
02B7 A5 DD LDA THRUST test; is motor off?
02B9 F0 FA BEQ RETRN yes, ignore key
02BB 86 DD STA THRUST no, store thrust
; calculate accel as thrust minus 5
02BD A5 DD THRSET LDA THRUST
02BF 38 SEC
02C0 F8 SED
02C1 E9 05 SBC #5
02C3 85 DC STA TH2+1
02C5 A9 00 LDA #0
02C7 E9 00 SBC #0
02C9 85 DB STA TH2
02CB 60 RTS
; initial values
02CC 45 01 00 .BYTE $45,1,0 altitude
02CF 99 81 00 .BYTE $99,$81,0 rate of ascent
02D2 99 97 .BYTE $99,$97 acceleration
02D4 02 .BYTE 2 thrust
02D5 08 00 00 .BYTE 8,0,0 fuel
02D8 01 .BYTE 1 display mode
02D9 01 .BYTE 1 in flight/landed
; end
00D5 ALT *=*+3
00D8 VEL *=*+3
00DB TH2 *=*+2
00DD THRUST *=*+1
00DE FUEL *=*+1
00E1 MODE *=*+1
00E2 DOWN *=*+1
00E3 DECK *=*+1
; linkages to KIM monitor
SCANDS = $1F1F
GETKEY = $1F6A
POINTH = $FB
POINTL = $FA
INH = $F9
```

The above is the assembly language for the 6502 microprocessor which is used in the KIM and in this KIN computer. Just for fun, here is a link to a web page which lets you write programs in AppleSoft BASIC. Basic is what we used to analyze our Roulette data on the Apple II.

Frank Mellon and I in my bedroom computer lab, Summer 1978.

Pick one up for a friend!

Pictures from our roulette prediction project

KIM-1 technology (6502)

Back to MAIN

MINOX camera

KIM microcomputer Redux

These pages show computer work done on Black Jack strategies and Roulette prediction. © 2013 by @Com