

Copyright 1991 Lee Djavaherian

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

realprog8

ready.

```
1 dim wd$(100):l2=log(2)
2 read x:if x<>100 then read wd$(x):goto 2
5 poke 56579,63:gosub 10000
10 for x=0 to 7:ob(x)=2^x:ab(x)=255-ob(x):next x
20 p1=56844:c1=56845
30 p2=56846:c2=56847
40 p3=57100:c3=57101
50 p4=57102:c4=57103
60 poke c1,0:poke c2,0:poke c3,0:poke c4,0
70 poke p1,63:poke p2,240:poke p3,255:poke p4,0
80 poke c1,4:poke c2,4:poke c3,4:poke c4,4
90 poke p3,1+8:poke p1,peek(p1) or 31
100 def fn a1(q)=peek(p1) or ob(q)<
110 def fn a2(q)=peek(p2) or ob(q)
120 def fn a3(q)=peek(p3) or ob(q)
130 def fn a4(q)=peek(p4) or ob(q)
140 def fn b1(q)=peek(p1) and ab(q)
150 def fn b2(q)=peek(p2) and ab(q)
160 def fn b3(q)=peek(p3) and ab(q)
170 def fn b4(q)=peek(p4) and ab(q)
180 def fn ss(q)=peek(p2) and 15
190 def fn oh(q)=peek(p4) and 124
192 def fn r1(q)=peek(p1) and ob(q)
194 def fn r2(q)=peek(p2) and ob(q)
196 def fn r3(q)=peek(p3) and ob(q)
197 def fn r4(q)=peek(p4) and ob(q)
198 for x=1 to 5:ww(x)=ob(x+1):next x:ww(3)=0
199 rem main loop
200 hq=124:gosub 8300:gosub 8700
207 iu=0
210 print "main loop"
215 hr=fn oh(0):if hr<hq then gosub 3500:if fl=1 then gosub 8000:goto 240
220 if fn r1(7)=0 then gosub 7000
230 goto 210
240 b=ti/60
250 print "second loop":if ti/60>b+20 then iu=1:gosub 5000:goto 200310
260 dt=fn ss(0)
270 if dt=11 or dt=12 then gosub 6000:if iu=1 then gosub 5000:goto 200310
280 hr=fn oh(0)
290 if hr<hq then print "8000":gosub 8000
300 if hr=124 then gosub 3500:if fl=1 then 310
305 goto 250
310 ph(1)=38:ph(2)=35:gosub 3000:goto 200
1999 rem speech interface
2000 oz=1
2005 l=len(wd$(ph(oz))):yz$=wd$(ph(oz)):print "ph(oz)":ph(oz)
2010 for x=1 to l/step 3
2020 s=val(mid$(yz$,x,2))
2030 poke 56577,s:poke 56577,0
2040 if (peek(56577)and 64)<>64 then 2040
2050 next x
2060 ph(oz)=0
2070 oz=oz+1:print oz:if ph(oz)=0 then return
2080 poke 56577,3:goto 2005
```

```

2999 rem speaker to speech
3000 gosub 8700:gosub 2000
3010 poke p1,peek(p1) or 31
3020 return
3499 rem off hook debouncer
3500 fl=0
3510 for x=1 to 500:next x
3520 if fn oh(0)=hr then fl=1
3530 return
3999 rem answering machine starter
4000 r=0:ph(1)=37:ph(2)=26:gosub 3000
4010 a=ti/60:r=r+1:print "artificial ring on"
4020 poke p3,fn a3(6)
4035 if fn r4(7)=0 then ph(1)=26:ph(2)=34:gosub 3000:return
4040 poke p3,fn b3(6)
4060 if ti/60<a+2 then 4020
4070 if r=3 then ph(1)=26:ph(2)=32:gosub 3000:return
4075 a=ti/60:print "artificial ring off"
4080 if ti/60<a+4 then 4080
4090 goto 4010
4999 rem phone in use routine
5000 dt=fn ss(0)
5010 hr=fn oh(0):if hr=124 then return
5020 if hr<>hq then gosub 8000:goto 5030
5030 if dt=0 then 5000
5040 if dt=12 then 5170
5050 if dt>5 then 5000
5060 b=ti/60:pp=dtWW(pp)=0:WW(pp)=1
5070 if ti/60>b+20 then ph(1)=27:ph(2)=pp:ph(3)=32:gosub 3000:goto 5000
5080 poke p1,fn b1(dt=1)
5100 poke p2,fn b2(5) for x=1 to 5
5110 ph(1)=28:ph(2)=35:gosub 3000:goto 5000 if WW(x)=0 AND X>PP THEN PH(1)=27:PH(2)=X:
5120 next xPP gosub 3000
5130 ph(1)=33:ph(2)=27:ph(3)=dt:gosub 3000
5140 hr=fn oh(0):if hr=124 then return
5150 if hr<>hq then gosub 8000:goto 5000
5160 dt=fn ss(0):if dt=0 then 5070
5163 if dt=12 then 5170
5167 if dt=11 then gosub 4000:goto 5000
5168 if dt<6 then 5060
5169 goto 5070
5170 poke p2,fn a2(5) hold OPEN
5180 for x=1 to 100:next x
5190 poke p2,fn a2(4)
5200 a=ti/60
5210 b=ti/60:ph(1)=28:gosub 3000
5220 if ti/60>b+10 then 5210
5230 hr=fn oh(0)
5240 if hr<hq then 5280
5250 if hr=124 then hq=124:goto 5220
5260 dt=fn ss(0):if dt=0 then 5220
5263 if dt=12 then 5280
5267 if dt=11 then gosub 4000
5269 goto 5220 if dt<6 then 5280
5280 poke p2,fn b2(4)
5290 for x=1 to 100:next x huh
5999 rem phone not in use routine
6000 poke p2,fn a2(4) turn on power
6005 if dt=11 then gosub 4000
6007 if dt=12 then ph(1)=29:gosub 3000 ready

```

5225 if T1/60 > A+900 THEN 5280

if 5268 if dt < 6 then

5270 C = T1/60 : PP = dt : WW(pp) = 0
if T1/60 > C+20 then

```
6010 dt=fn ss(0):if dt>0 then 6050
6020 hr=fn oh(0):if hr=124 then gosub 3500:if fl=1 then 6170
6025 if hr<>hq then gosub 8000
6030 if fn r1(7)=0 then gosub 7000:if iu=1 then 6170
6040 goto 6010
6050 if dt=11 then gosub 4000:
6060 if dt=12 then 6170 iu=1:6070 6170
6065 if dt>5 then 6010
6070 b=ti/60:pp=dt: WW(pp)=0 al(i-)
6075 if ti/60>b+20 then ph(1)=27:ph(2)=pp:ph(3)=32:gosub 3000:goto 6010
6080 for x=1 to 5 WW(pp)=1:
6090 if WW(x)=0 AND x>pp then WW(pp)=1:
6095 next x
6100 ph(1)=33:ph(2)=27:ph(3)=pp:gosub 3000
6130 hr=fn oh(0):if hr=124 then 6170
6140 if hr<hq then gosub 8000:goto 6010
6160 dt=fn ss(0):if dt=0 then 6075
6163 if dt=12 then 6170
6167 if dt=11 then gosub 4000:goto 6010
6168 if dt<6 then 6070
6169 goto 6075
6170 poke p2,fn b2(4)
6180 return
6999 rem ring detector and counter
7000 rn=0
7010 if fn r1(7)=0 then a=ti/60
7020 if ti/60<a+.5 then 7010
7030 rn=rn+1:ph=(1)=31:gosub 3000
7035 if rn=5 then gosub 4000:return
7037 if fn oh(0)<>124 then iu=1:return
7040 if fn r1(7)=0 then a=ti/60:goto 7020
7050 if ti/60<a+4.5 then 7040
7060 ph(1)=31:ph(2)=35:gosub 3000
7070 return
7999 rem connect/disconnect routine
8000 gosub 3500:if fl=0 then return
8005 vv=abs(hq-hr)
8010 gosub 8300
8030 ph(1)=27:ph(2)=(log(vv)/12)-1:print ct:gosub 3000
8040 if hr<hq then ph(1)=34:ph(2)=39:goto 8060
8050 ph(1)=35:ph(2)=36
8060 gosub 3000
8070 for x=1 to 5
8080 if WW(x)=0 and WW(x)<>vv then ph(1)=27:ph(2)=x:gosub 3000
8085 next x
8090 poke p1,peek(p1) or 63
8100 hq=hr
8110 return
8299 rem phone in use variable loader
8300 ct=0
8310 for x=1 to 5
8320 WW(x)=fn r4(x+1)
8330 if WW(x)=0 then ct=ct+1
8335 next x
8340 return
8699 rem phone in use speaker switcher
8700 vl=31
8710 for x=1 to 5
8720 if WW(x)=0 then vl=vl-ob(x-1)
8730 next x
```

```
8740 poke p1,peek(p1) and (vl+192)
8750 poke p1,peek(p1) or vl
8760 return
8999 rem led counter
9000 for gy=1 to 2
9010 mg=0
9020 c=ti/60
9030 if ti/60>c+1 then 9070
9040 if fn r1(6)=0 then 9030
9050 if fn r1(6)>0 then 9050
9060 mg=mg+1:goto 9020
9070 next gy
9080 if mg=15 then print "there are at least";
9090 print mg;" messages."
9100 return
9999 rem speech synthesizer vocabulary
10000 data 41,"":rem
10010 data 42,"10 23 35 23 27 47 19 07 11 55":rem djavaherians
10020 data 43,"45 19 43":rem lees
10030 data 44,"40 26 16 12 45 19 03 14 31 16":rem family room
10040 data 45,"":rem
10050 data 33,"12 12 43 03 08 23 23 45 12 44":rem is calling
10060 data 1,"46 15 15 11":rem one
10070 data 2,"13 31":rem two
10080 data 3,"29 14 19":rem three
10090 data 4,"40 40 58":rem four
10100 data 5,"40 40 06 35":rem five
10110 data 6,"55 55 12 12 02 41 55":rem six
10120 data 7,"55 55 07 07 35 12 11":rem seven
10130 data 8,"20 02 13":rem eight
10140 data 9,"11 24 06 11":rem nine
10150 data 10,"13 07 07 11":rem ten
10160 data 11,"12 45 07 07 35 12 11":rem eleven
10170 data 12,"13 48 07 07 45 35":rem twelve
10180 data 13,"29 51 01 02 13 19 11":rem thirteen
10190 data 14,"40 58 01 02 13 19 11":rem fourteen
10200 data 15,"40 12 40 01 02 13 19 11":rem fifteen
10210 data 16,"55 55 12 02 41 55 01 02 13 19 11":rem sixteen
10220 data 17,"55 55 07 35 12 11 01 02 13 19 11":rem seventeen
10230 data 18,"20 01 02 13 19 11":rem eighteen
10240 data 19,"11 06 11 01 02 13 19 11":rem nineteen
10250 data 20,"13 48 07 07 11 01 02 13 19":rem twenty
10260 data 30,"29 52 01 02 13 19":rem thirty
10270 data 40,"40 58 02 13 19":rem forty
10280 data 50,"40 40 12 40 40 01 02 13 19":rem fifty
10290 data 21,"20 01 07 07 16":rem am
10300 data 22,"09 19 01 07 07 16":rem pm
10310 data 23,"53 02 42 45 24 24 02 41":rem o'clock
10320 data 24,"18 15 15 03 13 24 06 16 03 12 12 43":rem the time is
10330 data 25,"53":rem o
10340 data 26,"26 11 55 51 19 44 03 16 15 37 19 11":rem answering machine
10350 data 27,"40 40 53 11":rem phone
10360 data 28,"57 53 45 21":rem hold
10370 data 29,"14 07 07 00 33 19":rem ready
10380 data 31,"14 19 44":rem ring
10390 data 32,"33 12 12 21 03 56 24 24 13 03 14 19 55 09 24 24 11 01 21"
10395 rem ^did not respond
10400 data 34,"42 15 11 07 41 13 07 21":rem connected
10410 data 35,"33 12 55 42 15 11 07 41 13 07 21":rem disconnected
10420 data 36,"40 40 39 15 15 16":rem from
```

```
10430 data 37,"55 55 13 59 13 19 44":rem starting  
10440 data 38,"23 b3 45 b3 40 40 53 11 55 b3 59":rem all phones are  
10450 data 39,"13 31":rem to  
19998 data 100  
19999 return
```

ready.