

	Application Note
	Plix Chip and Plix E-Board
Quick Basic and the Plix	Date: 2/13/99
<p>Introduction: This application note demonstrates how to interface our PLIX E-Board with a computer using Quick Basic.</p>	
<p>Background: The PLIX is an 18-pin CMOS chip which provides an intelligent communication interface between a computer and X-10 AC power-line control modules. Using our example software and a TW523 power-line adapter module, you'll see how easy it is to remotely control appliances and lights through the power line.</p>	
<p>How it works: PLIX removes the burden of complex X-10 programming protocol from the designer by providing a simple parallel interface. It takes care of the complex zero-crossing timing for sending and receiving X-10 commands so you don't have to. An otherwise simple embedded controller can now also feature X-10 power-line control by simply adding a PLIX chip to the design.</p>	
<p>Program Listing:</p> <pre> 1 REM ***** VERSION 2.0 ***** 10 DATA &H278,&H378,&H3BC 20 FOR P=1 TO 3 30 READ DPORT:REM DATA OUTPUT PORT 40 SPORT=DPORT+1:REM STATUS INPUT PORT 50 CPORT=DPORT+2:REM CONTROL OUTPUT PORT 60 ACTION=0:REM THE FLAG 70 OUT DPORT,&HFF:REM WRITE DATA=255 80 STATUS=INP(SPORT):REM AND READ IT BACK 90 PRINT STATUS 100 OUT DPORT,&H0:REM WRITE DATA=0 110 IF(STATUS<>INP(SPORT)) THEN ACTION=1:REM FLAG THE CONNECTED PORT 120 PRINT INP(SPORT) 130 IF ACTION=1 THEN GOTO 170:REM LEAVE ROUTINE IF THE PORT WAS FLAGGED 140 NEXT P:REM IF NO FLAG CHECK THE NEXT PORT 150 PRINT"No PLIX found, check connections and try again" 160 END:REM NO PLIX FOUND AT ANY OF THE PORTS - HALT 170 PRINT"PLIX found at ";HEX\$(DPORT);"H" 175 OUT DPORT,&HFF:REM WRITE DATA=255 TO ALLOW PLIX TO STEAL POWER 180 PUP=0:DUP=4:NUP=0:SUP=0:REM DEFINE THE HIGH LEVEL BIT VALUES 190 PDOWN=8:DDOWN=0:NDOWN=2:SDOWN=1:REM DEFINE THE LOW LEVEL BIT </pre>	

```

VALUES
200 POWER=PUP:DIRECTION=DDOWN:NIBBLE=NDOWN:ENABLE=SDOWN:REM
INITIAL VALUES
210 CPORTV=POWER+DIRECTION+NIBBLE+ENABLE:REM INITIALIZE CONTROL
PORT VALUE
215 OUT CPORT,9:REM INSURE PLIX RESET & CS ARE TOGGLED
220 OUT CPORT,CPORTV
230 PRINT
240 PRINT
250 PRINT"Choose a function"
260 PRINT
270 PRINT"1 - Send a sync sequence"
280 PRINT"2 - Write a command sequence"
290 PRINT"3 - Display the last X-10 transmission received"
300 PRINT"4 - Display the power line status"
310 PRINT"5 - End"
320 PRINT
330 I$=INKEY$:IF I$="" THEN GOTO 330
340 IF I$="1" THEN GOTO 410
350 IF I$="2" THEN GOTO 520
360 IF I$="3" THEN GOTO 890
370 IF I$="4" THEN GOTO 1240
380 IF I$="5" THEN OUT CPORT,&HC:END:REM THAT'S ALL FOLKS
390 GOTO 230
400 REM
*****
410 REM SEND THE SYNC SEQUENCE
420 REM
*****
430 PRINT"Writing these bytes to PLIX as a sync sequence..."
440 RANGE=0
450 GOSUB 1480:REM WRITE ROUTINE
460 GOSUB 1480:REM WRITE ROUTINE
470 GOSUB 1480:REM WRITE ROUTINE
480 RANGE=31
490 GOSUB 1480:REM WRITE ROUTINE
500 GOTO 230
510 REM
*****
520 REM WRITE A COMMAND SEQUENCE
530 REM
*****
540 PRINT"Hit the appropriate HOUSECODE (A-P)"
550 I$=INKEY$:IF I$="" THEN GOTO 550
560 IF(I$>"`" AND I$<"q")THEN I$=CHR$(ASC(I$)-32):REM MAKE IT
UPPER CASE
570 IF(I$<"A" OR I$>"P")THEN GOTO 550
580 HCODE$=I$
590 PRINT HCODE$:REM THIS IS YOUR CHOICE
600 PRINT"Input your FUNCTION selection"
610 PRINT
620 PRINT"Enter number 1-16 as a UNIT code, or..."
630 PRINT
640 PRINT"17 - All units OFF      25 - All Lights ON"
650 PRINT"18 - Hail Request *    26 - Hail Acknowledge *"
660 PRINT"19 - Dim                27 - Bright"
670 PRINT"20 - Entended DATA *  28 - Status=ON *"

```

```

680 PRINT"21 - ON                29 - OFF"
690 PRINT"22 - Preset Dim Lo *   30 - Preset Dim Hi *"
700 PRINT"23 - All Lights OFF * 31 - Extended CODE *"
710 PRINT"24 - Status=OFF *      32 - Status Request ***"
720 PRINT"*=not currently supported by X-10 modules **=RF gateway
only"
730 INPUT FCODE
740 IF (FCODE<1 OR FCODE>32) THEN GOTO 600
750 PRINT
760 PRINT"Enter the number of times to send the X-10 command (1-
30)"
770 INPUT REPEAT
780 IF(REPEAT<1 OR REPEAT>30) THEN GOTO 760
790 PRINT"Sending these bytes to PLIX for your selection..."
800 CODE=ASC(HCODE$)-64:REM SHIFT HOUSE CODE FOR TABLE ENTRY 1-16
810 GOSUB 1310:REM GET ACTUAL COMMAND BYTE FROM TABLE
820 GOSUB 1480:REM WRITE ROUTINE
825 IF (FCODE<17) THEN CODE=FCODE
830 IF(FCODE<17)THEN GOSUB 1310 ELSE RANGE=FCODE-1
840 GOSUB 1480:REM WRITE ROUTINE
850 RANGE=REPEAT
860 GOSUB 1480:REM WRITE ROUTINE
870 GOTO 230
880 REM
*****
890 REM DISPLAY LAST X-10 TRANSMISSION RECEIVED
900 REM
*****
910 DIRECTION=DDOWN
920 GOSUB 1750:REM SET DIRECTION
930 RANGE=0:REM  THIS ALLOWS THE PLIX CHIP TO DRIVE THE STATUS
PORT
940 GOSUB 1860:REM SET DATA
950 FOR Z=0 TO 1:REM GET 2 BYTES FROM PLIX
960 ENABLE=SUP
970 GOSUB 1800:REM SET ENABLE
980 GOSUB 1640:REM CHECK READY
990 IF READY=0 THEN PRINT". ";GOTO 980:REM BUSY
1000 NIBBLE=NDOWN
1010 GOSUB 1690:REM  SET NIBBLE
1020 RANGELO=INP(SPORT)
1030 NIBBLE=NUP
1040 GOSUB 1690:REM  SET NIBBLE
1050 RANGEHI=INP(SPORT)
1060 RANGE=0
1070 IF (RANGELO AND &H10)=&H10 THEN RANGE=RANGE+&H1
1080 IF (RANGELO AND &H20)=&H20 THEN RANGE=RANGE+&H2
1090 IF (RANGELO AND &H80)=&H0 THEN RANGE=RANGE+&H4
1100 IF (RANGEHI AND &H10)=&H10 THEN RANGE=RANGE+&H8
1110 IF (RANGEHI AND &H20)=&H20 THEN RANGE=RANGE+&H10
1120 IF Z=0 THEN PRINT:PRINT"This is ";
1130 IF Z=0 THEN IF RANGE>15 THEN RANGE=RANGE-16 ELSE PRINT"not
";
1140 IF Z=0 THEN PRINT"a new entry",
1150 IF Z=0 THEN PRINT "HC=";ELSE PRINT "FC=";
1160 PRINT RANGE,
1170 ENABLE=SDOWN

```

```

1180 GOSUB 1800:REM SET ENABLE
1190 GOSUB 1640:REM CHECK READY
1200 IF READY=1 THEN PRINT". ";GOTO 1190:REM BUSY
1210 NEXT Z:REM GET ANOTHER BYTE FROM PLIX
1220 GOTO 230
1230 REM
*****
1240 REM DISPLAY POWER LINE STATUS
1250 REM
*****
1260 POWER=(INP(SPORT) AND &H40):REM READ SPORT D6 (ACK PP.10)
1270 PRINT"AC power is ";
1280 IF (POWER=&H40) THEN PRINT"ON" ELSE PRINT"OFF"
1290 GOTO 230
1300 REM
*****
1310 REM GET ACTUAL COMMAND BYTE FROM TRANSLATION TABLE
1320 REM
*****
1330 IF (CODE=1) THEN RANGE=6
1340 IF (CODE=2) THEN RANGE=7
1350 IF (CODE=3) THEN RANGE=4
1360 IF (CODE=4) THEN RANGE=5
1370 IF (CODE=5) THEN RANGE=8
1380 IF (CODE=6) THEN RANGE=9
1390 IF (CODE=7) THEN RANGE=10
1400 IF (CODE=8) THEN RANGE=11
1405 IF (CODE=9) THEN RANGE=14
1410 IF (CODE=10) THEN RANGE=15
1420 IF (CODE=11) THEN RANGE=12
1425 IF (CODE=12) THEN RANGE=13
1430 IF (CODE=13) THEN RANGE=0
1440 IF (CODE=14) THEN RANGE=1
1445 IF (CODE=15) THEN RANGE=2
1450 IF (CODE=16) THEN RANGE=3
1460 RETURN
1470 REM
*****
1480 REM  write routine
1490 REM
*****
1500 GOSUB 1860:REM SET RANGE
1510  DIRECTION=DUP
1520 GOSUB 1750:REM SET DIRECTION
1530  ENABLE=SUP
1540 GOSUB 1800:REM SET ENABLE
1550 GOSUB 1640:REM CHECK READY
1560 IF READY=0 THEN PRINT". ";GOTO 1550:REM BUSY
1570 PRINT RANGE,
1580  ENABLE=SDOWN
1590 GOSUB 1800:REM SET ENABLE
1600 GOSUB 1640:REM CHECK READY
1610 IF READY=1 THEN PRINT". ";GOTO 1600:REM BUSY
1620 RETURN
1630 REM
*****
1640 REM  CHECK READY

```

```

1650 REM
*****
1660 IF ((INP(SPORT) AND &H8)=&H8) THEN READY=1 ELSE READY=0
1670 RETURN
1680 REM
*****
1690 REM SET NIBBLE (NDOWN=LOWER 3 BITS NUP=UPPER 2 BITS)
1700 REM
*****
1710 CPORTV=((CPORTV AND &HFD) OR NIBBLE)
1720 OUT CPORT,CPORTV
1730 RETURN
1740 REM
*****
1750 REM SET DIRECTION (DUP=WRITE DDOWN=READ)
1760 REM
*****
1770 CPORTV=((CPORTV AND &HFB) OR DIRECTION)
1780 RETURN
1790 REM
*****
1800 REM SET ENABLE (SUP=CHIP ENABLED SDOWN=CHIP DISABLED)
1810 REM
*****
1820 CPORTV=((CPORTV AND &HFE) OR ENABLE)
1830 OUT CPORT,CPORTV
1840 RETURN
1850 REM
*****
1860 REM SET DATA (RANGE=DATA)
1870 REM
*****
1880 OUT DPORT,(RANGE OR &HE0)
1890 RETURN
1900 REM
*****
1910 REM SET POWER (PUP=ON PDOWN=OFF)
1920 REM
*****
1930 CPORTV=((CPORTV AND &HF7) OR POWER)
1940 OUT CPORT,CPORTV
1950 RETURN
1960 IF (CODE=9) THEN RANGE=14
1970 IF (CODE=12) THEN RANGE=13
1980 IF (CODE=15) THEN RANGE=2

```