INTRODUCTION

This Application Note provides the hardware circuit and software modifications required to use an IBM PC* or PC compatible keyboard with the Megatel Quark family of single board computers. The software modifications described apply to Megatel releases of CP/M 2.21 and 2.22 only. Further, the Note does not apply to IBM PC lookalike keyboards with RS-232C interfaces, which can be connected directly to the Quark's Full Duplex Serial Port.

HARDWARE DESCRIPTION

This implementation of the IBM PC keyboard connection utilizes the Full Duplex Serial Port on the Quark. As such, an external clock synchronization circuit is required. This circuit appears in Figure 1. Values for R1, R2 and C2 should be chosen to set the period of oscillator U1 to double the period of the keyboard clock (suggested values are shown). Figure 1 also shows the jumper connection required on the Quark.

SOFTWARE MODIFICATIONS

The code changes listed below permit the use of an IBM PC keyboard that is connected to a Quark. The original code was taken from BYTE magazine (May 1983, page 402). Some changes and additions have been made to the original code to further enhance the routine.

Each key on the IBM PC keyboard generates both a make and break scan code. For example key 1 (esc) produces a scan code 01 on make (when depressed) and code 81(hex) on break (when released). Break codes are formed by adding 80(hex) to the make codes.

The code changes include some Z80 instructions. These instructions are coded by defining bytes and are followed by a comment line stating the Z80 instuction.

for example:

```
DB 0CH,7*8+a+80H
; RES    7,a
```

To make the modifications to the operating system, a blank formatted disk, and a "SYSGENed" disk with an operating system and the following files are needed:

QSYSGEN.COM
DDT6.COM
QBIOS.ASM
ASM.COM
a text editor or word processor

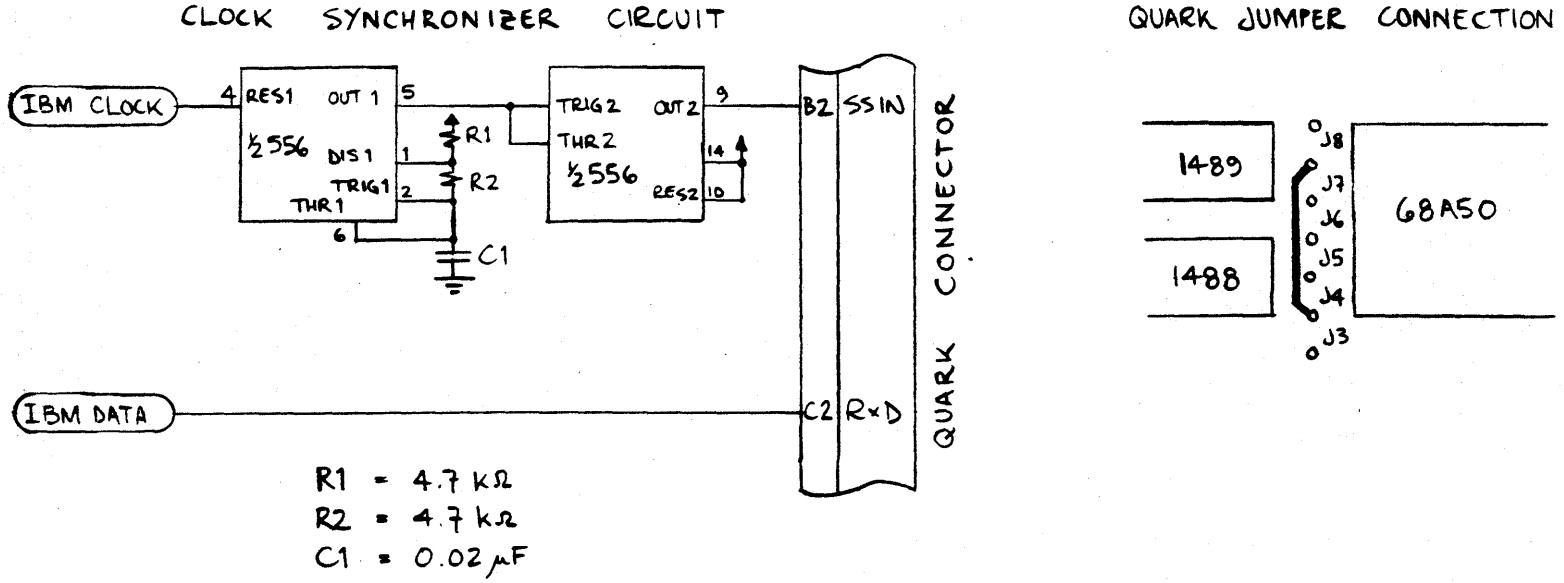

FIGURE 1. Clock Synchronizer Circuit and Quark Jumper Connection

 Order #QT000030-R1

The following code must replace the existing code in the QBIOS.ASM. The routines that must be changed are CONIN, CONST, and WBOOT.

The following code will replace the CONIN: and CONST: routines.

```
; I.B.M. P.C. KEYBOARD CONVERSION FROM BYTE
;
;ACIA EQUATES
STREG   EQU     78H             ; ACIA STATUS REGISTER
RDREG   EQU     79H             ; ACIA RECEIVE DATA REGISTER */
;
; CONCOLE STATUS CHECK
CONST:
        LDA     SAVECHAR
        ORA     A
        JZ      CONST1
        MVI     A,0FFH
        ORA     A
        RET
CONST1:
        IN      STREG
        ANI     1
        RZ                      ; NO CHARACTER TYPED
        PUSH    H
        PUSH    D
        PUSH    B
        CALL    KYBRD           ; GET CHARACTER
        POP     B
        POP     D
        POP     H
        JMP     CONST
;
;CONSOLE INPUT ROUTINE
CONIN:
        CALL    CONST
        JZ      CONIN
        LDA     SAVECHAR
        PUSH    H
        LXI     H,SAVECHAR
        MVI     M,0
        POP     H
        RET
;
;MAIN ROUTINE TO CONVERT I.B.M. KEY BOARD CODE TO ASCII CODE
;
KYBRD:
        CALL    KEYR
        MOV     C,A
        ORA     A
        RZ
        DB      0CBH,7*8+A+80H
;       RES     7,A
        LXI     H,TAB1-1
        MVI     D,0
        MOV     E,A
        DAD     D
        MOV     A,M
```

```
        MOV     E,A
        LXI     H,SHIFT
        CPI     83H             ;ALT KEY?
        MVI     B,80H
        JZ      SHF
        CPI     82H             ;CNTRL KEY?
        MVI     B,40H
        JZ      SHF
        CPI     81H             ;LEFT SHIFT KEY?
        MVI     B,2
        JZ      SHF
        CPI     80H             ;RIGHT SHIFT KEY
        MVI     B,4
        JZ      SHF
        DB      0CBH,7*8+C+40H
;       BIT     7,C
        JNZ     RET3
        CPI     84H             ;NUMLOCK KEY?
        MVI     B,20H
        JZ      SHFL
        CPI     85H             ;CAPSLOCKS KEY?
        MVI     B,1
        JZ      SHFL
;
;PROCESS NON SHIFT KEYS
;
        CPI     1BH
        JC      SPEC
        CPI     7BH
        JNC     OTHER
        CPI     61H
        JC      OTHER
        MOV     A,M
        DB      0CBH,0*8+A+40H
;       BIT     0,A
        JZ      KYBRD1
        ANI     6
        MOV     A,E
        JZ      KYBRD2
        JMP     RET0
KYBRD1:
        ANI     6
        MOV     A,E
        JZ      RET0
KYBRD2:
        DB      0CBH,5*8+A+80H
;       RES     5,A
        JMP     RET0
OTHER:
        MOV     A,M
        ANI     6
        MOV     A,E
        JZ      RET0
        CPI     60H
        JZ      OTH2
        CPI     5AH
        JNC     OTH1
        LXI     H,TAB2+1
```

```
        SBI     27H
        JC      RET0
        MVI     D,0
        MOV     E,A
        DAD     D
        MOV     A,M
        JMP     RET0
OTH1:
        DB      0CBH,5*8+A+0C0H
;       SETB    5,A
        JMP     RET0
OTH2:
        MVI     A,7EH
RET0:
        MVI     C,0
        LXI     H,SHIFT
        DB      0CBH,6*8+M+40H
;       BIT     6,M
        JZ      RET1
        ANI     1FH
RET1:
        DB      0CBH,7*8+M+40H
;       BIT     7,M
        STA     SAVECHAR
        RZ
        DB      0CBH,7*8+A+0C0H
;       SETB    7,A
        STA     SAVECHAR
        RET
;
;PROCESS SHIFT KEYS
;
SHF:
        DB      0CBH,7*8+C+40H
;       BIT     7,C
SHF1:
        MOV     A,B
        JNZ     SHFOFF
        ORA     M
SHF2:
        MOV     M,A
RET3:
        XRA     A
        RET
SHFOFF:
        CMA
        ANA     M
        JMP     SHF2
SHFL:
        MOV     A,B
        ANA     M
        JMP     SHF1
```

```
;
;PROCESS FUNCTION AND CURSOR PAD
;
SPEC:
        CPI     11H
        MOV     A,M
        JC      SPEC0
        DB      0CBH,5*8+A+40H
;       BIT     5,A
        JZ      SPEC0
        ANI     6
        MOV     A,E
        JNZ     SPEC2
        JMP     SPEC1
SPEC0:
        ANI     6
        MOV     A,E
        JZ      SPEC2
SPEC1:
        DB      0CBH,5*8+A+0C0H
;       SETB    5,A
SPEC2:
        DB      0CBH,6*8+M+40H
;       BIT     6,M
        JZ      SPEC3
        DB      0CBH,6*8+A+0C0H
;       SETB    6,A
        JMP     SPEC5
SPEC3:
        CPI     0EH
        JC      SPEC5
        CPI     1AH
        JNC     SPEC5
        PUSH    H
        PUSH    D
        LXI     H,TAB3
        MVI     D,0
        SUI     0EH
        MOV     E,A
        DAD     D
        MOV     A,M
        POP     D
        POP     H
SPEC5:
        MOV     E,A
        MOV     A,C
        MVI     C,0FEH
        CPI     0EH
        JZ      SPEC4
        CPI     0FH
        JZ      SPEC4
        CPI     1CH
        JZ      SPEC4
        INR     C
SPEC4:
        MOV     A,E
        JMP     RET1
```

```
KEYR:
        IN      STREG           ;CHECK STATUS
        ANI     1               ;IS THERE A CHARACTER
        JZ      KEYR
        IN      RDREG           ;GET CHARACTER
        CMA
        RET
CO:
        MVI     C,2
        CALL    5
        RET
;
SHIFT:          DB      0
;
SAVECHAR:       DB      0
;
TAB1:
        DB      1BH,`12345678`
        DB      `90-=`,8,9,`QWER`
        DB      `TYUIOP[]`,0DH,82H
        DB      `ASDFGHJKL;`
        DB      27H,60H,81H,5CH,`ZXCVBN`
        DB      `M,./`,80H,0FH,83H,20H,85H,1
        DB      2,3,4,5,6,7,8,9,0AH,84H
        DB      0CH,17H,18H,19H,2DH,14H,15H,16H,2BH,11H
        DB      12H,13H,10H,0EH
;
TAB2:
        DB      22H,0,0,0,0,3CH,5FH,3EH,3FH,29H
        DB      21H,40H,23H,24H,25H,5EH,26H,2AH,28H,0
        DB      3AH,00,2BH
TAB3:
        DB      7FH,0,16H,0,0AH,03,08,35H,0CH,1EH,0BH,12H
;
;*************************************************************************
```

The following must be added to the beginning of the WBOOT routine. The purpose of this code is to reset the control flag in the shift byte.

```
    LDA  SHIFT
    ANI  21H
    STA  SHIFT
```

The following procedure should be executed: Place the disk with the required files in drive A and the blank disk in drive B. Make all of the above changes to the QBIOS.ASM. Now enter the following instructions (the instructions in **bold face** are entered by the operator, **RET** means a return is entered):

```
A0>ASM QBIOS.AAZ

A0>QSYSGEN
Megatel Quark Floppy Sysgen VER 2.22
SOURCE DRIVE NAME (OR RETURN TO SKIP) A
SOURCE ON A, THEN TYPE RETURN RET
FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT) RET
```

```
A0>SAVE 53 QCPM.SYS
A0>DDT6 QCPM.SYS
DDT VERS 2.2
NEXT  PC
3600 0100
-IQBIOS.HEX
-R3880       (this is for Megatel release CP/M 2.22; for release CP/M 2.21 enter R4980)
-GO

A0>SAVE 53 QIBM.SYS
A0>QSYSGEN QIBM.SYS
Megatel Quark Floppy Sysgen VER 2.22
DESTINATION DRIVE NAME (OR RETURN TO REBOOT)B
DESTINATION ON A, THEN TYPE RETURN RET
FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT) RET

A0>
```

The disk in drive B: will now have an operating system that is compatible with the other operating system but will allow input from an IBM PC keyboard. This disk should be used to boot up an operating system after an IBM PC keyboard has been connected.

Please note that the above code is not completely free of bugs. For example, key strokes may be lost if there is a disk access at the same time as keyboard input.

Megatel recommends that the keyboard interrupt routine be enabled at all times. The disk routine should be changed to selectively disable unwanted interrupts instead of disabling all interrupts.

Finally, the ALT key when used in conjunction with two or more keys may cause incorrect interpretations from the keyboard.

Megatel Computer Technologies
A Division of F. & K. MFG. CO. LIMITED
150 Turbine Drive, Weston, Ontario, Canada M9L 2S2
1051 Clinton Street, Buffalo, New York, USA 14206
Telephone (416) 745-7214 Telex 065-27453 MEGATEL TOR