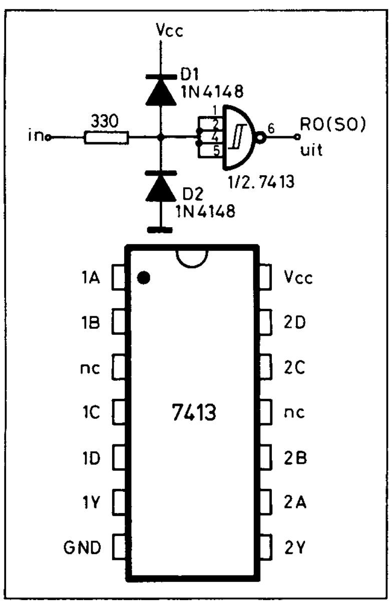
Frequency Meter

M. Dohmen
R. Koekoek

The program described here can measure a frequency up to, mind you, 100 kHz, with a deviation of no more than one cycle in 65536. The frequency is read out in five digits. The program was written for the KIM, but is also well suited to other 6502 microcomputers, notably the SYM and the Junior. Once started, the program measures a frequency continuously when key "6" is pressed, and once when key "0" is pressed.



Overflow

This program makes use of a feature of the 6502 that has not been used much until now. This microprocessor has an SO (Set Overflow) pin, pin 38, with which the overflow flag can be set from the outside. On most computers this pin is brought out directly to a connector pin (RO, pin E-5 on the KIM; SO, pin 14-c on the Junior). The overflow flag is set when the signal on this input goes from high to low. This provides fast, direct access to the microprocessor, without involving the rest of the computer. Once the overflow flag has been set, it can only be influenced through software.

Buffering

Because measurements have to be made directly in circuits, buffering and protection of the RO input is essential. The easiest way to achieve this is with a Schmitt-trigger gate, the 7413. This is then given voltage and current protection as shown in the figure. In this way, signals of any waveform can be measured, provided their amplitude is large enough and the signal is centered around 2.5 V.

Operation

The program starts at location 0200 hex. First, the interrupt vector is set to the start address of the interrupt routine (0298 hex). Next, a counter is initialized for the number of interrupts, after which the display is cleared. Then the timer is set to approximately 1/4 second, with interrupts enabled. For one complete cycle, four interrupts are processed, which results in a total time of approximately 1 second. To make this time exact, and to make it possible to calibrate the meter, an extra delay routine of approximately 3 ms follows. The accumulator and the carry and overflow flags are cleared, and then the number of overflows is counted, which takes place until the next interrupt. Counting is done in binary, with the least significant bits in the X register, the next in the Y register, and the most significant bits in the accumulator. Because this routine has been kept as short as possible, the maximum measurable frequency is approximately 100 kHz. The error of one in 65536 arises because once every FFFF hex counts the accumulator has to be incremented. This makes the counting routine slightly longer.
When an interrupt occurs, the program jumps to the interrupt routine. In it, the timer is restarted for 1/4 second, provided this was not the fourth interrupt. If it was, the binary value in the counter is converted to decimal. This is done only now, because with direct decimal counting everything would have to happen in the accumulator, and that takes more time. The conversion from binary to decimal works as follows: a table holds the binary representation of the decimal values 1, 10, 100, 1000 and 10000, respectively. First, 10000 is subtracted from the counter value, which is temporarily stored in a number of memory locations. As long as the result is not yet negative, a "10000 counter" is incremented. The result is each time put back into the original counter. If the result is negative, the value the counter had before the subtraction is taken, and the next value from the table is subtracted from it. This continues until the counter is zero. The decimal value is kept directly in the registers for the display. After the conversion the display lights up, making the value visible. Next, the program checks whether a key was pressed. If not, the previous key is retrieved. If so, then on a "0" the program stops measuring, and on a "6" the computer keeps indicating the frequency continuously.

Calibration

The frequency meter is calibrated as follows. A divider circuit, consisting of two divide-by-10 stages, is connected to the computer's 1 MHz clock. The frequency at its output will then be in the neighborhood of 10 kHz. This signal is fed to the measuring input and the program is started. By changing the values in memory locations 021C and 021E of the extra delay routine, the display is adjusted to 10000. A divide-by-100 was chosen because the accumulator is then not yet involved in the counting. The meter is thus at its most accurate. By placing the divide-by-100 circuit between the buffer (7413) and the RO (SO) input, the measuring range of the frequency counter is extended to 10 MHz. It is also important that, to make the timer work, on the KIM PB7 (A-15) must be connected to NMI (E-6), and on the Junior NMI (12-c) to IRQ (12-a). The timer generates an interrupt, and this has to be able to be acknowledged.

Changes for the Junior

The program as it appears in the listing works on the KIM. To make it suitable for the Junior, the following changes must be made:

Address	KIM	Junior
0203	FB	7B
0204	17	1A
0208	FA	7A
0209	17	1A
0219	0F	FF
021A	17	1A
0283	1F	8E
0284	1F	1D
02A0	0F	FF
02A1	17	1A

```
MICRO-WARE ASSEMBLER 65XX-1.0 PAGE 01

0010:
0020:                ******************************************
0030:                *                                        *
0040:                *          FREQUENTIEMETER               *
0050:                *          VOOR DE 6502                  *
0060:                *                                        *
0070:                *      AUTEURS : M. DOHMEN               *
0080:                *                R. KOEKOEK              *
0090:                *                                        *
0100:                ******************************************
0110:
0120: 0200                  ORG    $0200
0130:
0140:                ZEROPAGE ADRESSEN
0150:
0160: 0200           INTCNT *      $00D0
0170: 0200           ZEROP  *      $00D1
0180: 0200           TEMPC  *      $00E0
0190: 0200           TEMPB  *      $00E1
0200: 0200           TEMPA  *      $00E2
0210: 0200           INL    *      $00F8
0220: 0200           INH    *      $00F9
0230: 0200           POINTL *      $00FA
0240: 0200           POINTH *      $00FB
0250: 0200           CHAR   *      $00FE
0260:
0270:                TABEL
0280:
0290: 0200           TABLO  *      $02AF
0300: 0200           TABHI  *      $02B0
0310:
0320:                TIMERADRES
0330:
0340: 0200           TIMER  *      $170F
0350:
0360:                INTERRUPTVECTOR
0370:
0380: 0200           INTLO  *      $17FA
0390: 0200           INTHI  *      $17FB
0400:
0410:                MONITOR SUBROUTINE
0420:
0430: 0200           SCANDS *      $1F1F
0440:
0450:                HOOFDPROGRAMMA
0460:
0470: 0200 A9 02     INIT   LDAIM $02
0480: 0202 8D FB 17         STA   INTHI
0490: 0205 A9 98            LDAIM $98     ZET INTERRUPTVECTOR
0500: 0207 8D FA 17         STA   INTLO
0510: 020A A9 FC     START  LDAIM $FC     ZET INTERRUPTTELLER
0520: 020C 85 D0            STA   INTCNT OP -4
0530: 020E A9 00            LDAIM $00
0540: 0210 85 F9            STA   INH     MAAK DISPLAY EN DUS OOK
0550: 0212 85 FA            STA   POINTL DE TELLER SCHOON
0560: 0214 85 FB            STA   POINTH
0570: 0216 A9 F5            LDAIM $F5     ZET TIMER EN MAAK
0580: 0218 8D 0F 17         STA   TIMER   INTERRUPTS MOGELIJK
0590:
0600:                EXTRA DELAY ROUTINE
0610:
0620: 021B A2 03            LDXIM $03     PAS DEZE WAARDEN AAN
0630: 021D A0 00     BULOOP LDYIM $00     VOOR DE VEREISTE
0640: 021F 88        BILOOP DEY           NAUWKEURIGHEID
0650: 0220 D0 FD            BNE   BILOOP
0660: 0222 CA               DEX
0670: 0223 D0 F8            BNE   BULOOP
0680:
0690:                TELPROGRAMMA
0700:
0710: 0225 B8               CLV
0720: 0226 18               CLC
0730: 0227 98               TYA           MAAK ACCU SCHOON
0740: 0228 50 FE     OVERFL BVC   OVERFL OVERFLOW GESET?
0750: 022A B8               CLV
0760: 022B E8               INX           HOOG X OP MET 1
0770: 022C D0 FA            BNE   OVERFL IS X = FF?
0780: 022E C8               INY           HOOG Y OP MET 1
0790: 022F D0 F7            BNE   OVERFL IS Y = FF?
0800: 0231 69 01            ADCIM $01     HOOG ACCU OP MET 1
0810: 0233 D0 F3            BNE   OVERFL IS ACCU = FF?

0830:                BINAIR NAAR DECIMAAL CONVERSIE
0840:
0850: 0235 85 E2     BINDEC STA   TEMPA  BEWAAR ACCU
0860: 0237 84 E1            STY   TEMPB  BEWAAR Y
0870: 0239 86 E0            STX   TEMPC  BEWAAR X
0880: 023B A2 08            LDXIM $08    ZET POINTER
0890: 023D A5 E2     POS    LDA   TEMPA
0900: 023F C9 00            CMPIM $00    IS DE TELLER
0910: 0241 D0 0C            BNE   SUBT   NOG STEEDS
0920: 0243 A5 E1            LDA   TEMPB  GROTER DAN
0930: 0245 DD B0 02         CMPAX TABHI  DE HUIDIGE
0940: 0248 D0 05            BNE   SUBT   WAARDE
0950: 024A A5 E0            LDA   TEMPC  IN DE
0960: 024C DD AF 02         CMPAX TABLO  TABEL
0970: 024F 90 2D     SUBT   BCC   NEXT   NEEN,HAAL VOLGENDE TABELWAARDE
0980: 0251 38               SEC          JA, TREK
0990: 0252 A5 E0            LDA   TEMPC  DE HUIDIGE
1000: 0254 FD AF 02        SBCAX TABLO  WAARDE UIT
1010: 0257 85 E0            STA   TEMPC  DE TABEL
1020: 0259 A5 E1            LDA   TEMPB  AF VAN
1030: 025B FD B0 02         SBCAX TABHI  DE
1040: 025E 85 E1            STA   TEMPB  STAND IN
1050: 0260 A5 E2            LDA   TEMPA  DE TELLER
1060: 0262 E9 00            SBCIM $00
1070: 0264 85 E2            STA   TEMPA  EN ZET TELLERSTAND TERUG
1080: 0266 8A               TXA          BEWAAR DE TABELPOINTER
1090: 0267 48               PHA          IN X OP DE STACK
1100: 0268 4A               LSRA         BEPAAL IN WELK DIGIT DEZE
1110: 0269 4A               LSRA         TELWAARDE MOET KOMEN
1120: 026A AA               TAX          DOOR POINTER IN X
1130:
1140:                TELLERSTAND DECIMAAL IN DISPLAY
1150:
1160: 026B B0 04            BCS   TIENT  TIENTAL?
1170: 026D F6 F9            INCZX INH    NEEN, EEN ERBIJ
1180: 026F D0 09            BNE   NOG    EN TERUG
1190: 0271 B5 F9     TIENT  LDAZX INH    HAAL VORIGE STAND
1200: 0273 69 0F            ADCIM $0F    EN TEL TIENTAL OP
1210: 0275 95 F9            STAZX INH    DISPLAY

0040: 027A 68        NOG    PLA          HAAL TABELPOINTER VAN STACK
0050: 027B AA               TAX          EN PLAATS DIE IN X
0060: 027C 10 BF            BPL   POS    ALLES KLAAR?
0070: 027E CA        NEXT   DEX          TABELPOINTER NAAR
0080: 027F CA               DEX          VOLGENDE WAARDE
0090: 0280 10 BB            BPL   POS    EN TERUG
0100: 0282 20 1F 1F  DISPL  JSR   SCANDS DISPLAY TELLER
0110: 0285 D0 0B            BNE   KEY    TOETS INGEDRUKT?
0120: 0287 A5 F8            LDA   INL    NEEN, HAAL VORIGE
0130: 0289 D0 02            BNE   DI     WAS DIT NUL?
0140: 028B C6 D1            DEC   ZEROP  NEEN, DELAY
0150: 028D D0 F3     DI     BNE   DISPL  JA, BLIJF DISPLAYEN
0160: 028F 4C 0A 02  BACK   JMP   START  OPNIEUW
0170: 0292 4A        KEY    LSRA
0180: 0293 85 F8            STA   INL    BEWAAR TOETS
0190: 0295 4C 0A 02         JMP   START  OVERNIEUW
0200:
0210:                INTERRUPTROUTINE
0220:
0230: 0298 48               PHA          BEWAAR ACCU
0240: 0299 E6 D0            INC   INTCNT VERLAAG DE TELLER
0250: 029B F0 07            BEQ   VIER   ZIJN ER VIER GEWEEST?
0260: 029D A9 F5            LDAIM $F5    NEEN, LAADT TIMER EN
0270: 029F 8D 0F 17         STA   TIMER  MAAK INTERRUPT MOGELIJK
0280: 02A2 68               PLA          HAAL ACCU TERUG
0290: 02A3 40               RTI          EN KEER TERUG
0300: 02A4 68        VIER   PLA          HAAL ACCU TERUG
0310: 02A5 85 FE            STA   CHAR   BEWAAR DATA
0320: 02A7 68               PLA
0330: 02A8 68               PLA          ZET STACK GOED
0340: 02A9 68               PLA
0350: 02AA A5 FE            LDA   CHAR   HAAL DATA OP
0360: 02AC 4C 35 02         JMP   BINDEC CONVERTEER
0370: 02AF 01               =     $01
0380: 02B0 00               =     $00
0390: 02B1 0A               =     $0A
0400: 02B2 00               =     $00
0410: 02B3 64               =     $64
0420: 02B4 00               =     $00
0430: 02B5 E8               =     $E8
0440: 02B6 03               =     $03
0450: 02B7 10               =     $10
0460: 02B8 27               =     $27
```