



Gregoriaanse klok

Tijd van seconden tot eeuwen

M. Dohmen
R. Koekoek

Dit klokprogramma is zeer kort en zal zeker door menigeen nuttig kunnen worden gebruikt. Het telt decimaal en houdt decimaal kwart seconden, seconden, minuten, uren, dag van de week, datum, maand, jaar en eeuw bij. Daarbij wordt rekening gehouden met schrikkeljaren!

Gebruiksaanwijzing

Nadat het programma is ingetypt moet de NMI-vector worden geplaatst. (KIM: 7B op adres 17FA en 03 op 17FB.) Verder moet PB7 met NMI worden verbonden.

Nu wordt decimaal de secondestand op adres 0081 gezet, de minuutstand op 0082, de uurstand op 0083, de dag van de week (0 t/m 6) op 0084, de datum op 0085, de maand op 0086, het jaar op 0087 en de eeuw op 0088. Als alles naar behoren is uitgevoerd, wordt „ST” ingedrukt. Nu moet de klok lopen. Als u op adres 0081 kijkt, ziet u de secondestand. Zolang er niet op „RS” wordt gedrukt, blijft de klok lopen en stoort deze zich niet aan het uitvoeren van een ander programma (zoals de uitvoer van het monitorprogramma). Doordat deze klok het voordeel heeft te blijven werken, terwijl een ander programma wordt uitgevoerd, zijn de toepassingsmogelijkheden legio. Het is nog steeds mogelijk een programma van cassette in het RAM-geheugen te laden, zonder dat de klok gaat afwijken.

Met enige hardware is het mogelijk om op, ik noem maar iets, 12 april 2001, de radio aan te laten

gaan of om op de oneven dagen in de maand de tuinsproeier aan te laten gaan. Met een kleinscale software-truc is het mogelijk de computer uit te laten zoeken wanneer de 13e weer op een vrijdag valt.

Stroomdiagrammen

Het programma zal worden uitgelegd aan de hand van 2 stroomdiagrammen, die in complexiteit nogal van elkaar verschillen. In afb. 1 is de vereenvoudigde versie van het stroomdiagram te zien. In werkelijkheid bestaat het programma niet uit een lange rij vergelijkingen, maar uit een lus waaromheen zich allerlei conditionele sprongen bevinden. Deze behandelen afzonderlijk de verscheidene uitzonderingen, die tijdberekening met zich meebrengt.

De waarden, waarmee de tellerstanden worden vergeleken, staan in een tabel. Deze tabel wordt onderbroken voor een klein stukje programma. Dit is tussen de waarde voor de maand september en de waarde voor de maand oktober. September is namelijk de 9e maand (dec.) en oktober de 10e maand (dec.). Omdat register X (dit is de tabelwijzer) hexadecimaal telt, ziet hij het getal 10 (hex.) voor het getal 16 (dec.), zodat 6 plaatsen worden overgeslagen. Deze zes bytes worden nu nuttig gebruikt. Zodra er een interrupt komt, wordt de timer (zonder interrupt) goed gezet om ervoor te zorgen dat de interruptroutine altijd even lang duurt, onafhankelijk van de weg die het programma volgt. Het programma duurt nu, dankzij de timer, altijd ongeveer 101 µs. Deze tijdsduur compenseert ook de tijd, die de interrupttimer te weinig telt. Daarna wordt de kwart-secondeteller opgehoogd. Zolang deze

niet vier is, wacht de computer tot de gereserveerde tijd voor de interruptroutine om is. De routine, die wacht tot de tijd om is, begint bij het label „end”.

Is de kwart-secondeteller wel 4, dan maakt het programma deze nul en hoogt de secondeteller op. Zolang deze niet 60 is (overeenkomstig de waarde in de tabel), wordt naar „end” gesprongen. Als de tellerstand gelijk is aan de overeenkomstige waarde uit de tabel, wordt deze tellerstand nul gemaakt en wordt de minuutteller opgehoogd.

Hetzelfde geldt voor de uurteller, de dag-van-de-week-teller en de jaarteller. De datum- en maandteller worden op het moment, dat de maximale waarde wordt overschreden, 01 gemaakt.

Bij de datumteller behoort wat toelichting. In de tabel staat voor de maximale waarde voor de datumteller 33. Omdat de datumteller nooit 33 wordt, voorkomen we zo, dat de datumteller onnodig 01 wordt gemaakt. Zodra de datumteller is opgehoogd, wordt deze waarde vergeleken met de maximale waarde, die bij de betreffende maand hoort.

Stel nu dat de klok op 29 maart staat. De computer vergelijkt dan 29 met de maximale waarde voor de maand maart, die 32 is. Hierbij is 29 kleiner dan 32, dus er is nog niets aan de hand. Zodra de waarde in de datumteller gelijk is aan de desbetreffende waarde in tabel B, wordt bepaald of er op dat moment wordt vergeleken met de maximale waarde voor de maand februari. Wordt er inderdaad met februari vergeleken, dan wordt nagegaan of het hier om een schrikkeljaar gaat. Anders wordt er 01 in de datumteller gezet en wordt de maandteller



opgehoogd. Als de maanteller 13 is wordt er 01 in de maanteller gezet en wordt de jaarteller opgehoogd. Is het jaar deelbaar door 4 dan is er een schrikkeljaar, maar daarover meer in de uitleg aan de hand van het tweede stroomdiagram.

Tweede stroomdiagram

In afb. 2 kunt u dit tweede stroomdiagram zien. Deze is een stuk ingewikkelder, maar past nauwer bij de lus van het programma. X is de lusteller. Hierin zal steeds de juiste plaats in de tabel staan. In feite is dit register een tabelwijzer. In het stroomdiagram is te zien dat telkens met (iets, X) wordt vergeleken. In Y staat meestal 00 of 01, omdat, wanneer een waarde wordt overschreden, Y in de betreffende geheugenplaats wordt gezet. De diverse geheugenplaatsen, die 00, respectievelijk 01 moeten worden gemaakt, kunnen dan snel vanuit

Y worden geladen, zodat de accu niet hoeft te worden aangetast. Zodra X groter dan of gelijk aan 5 wordt (er wordt met datum, maand of jaar vergeleken), wordt Y geladen met 01, omdat dan in plaats van 00, 01 in de diverse geheugenplaatsen moet worden gezet. Dit is duidelijker te zien in de eenvoudige stroomdiagram.

Er behoort wat toelichting bij de bepaling van een schrikkeljaar. Wij, mensen, delen het getal door vier en kijken dan of er een rest is. De computer kan dit, als hij hexadecimaal rekt, op dezelfde manier doen door naar de eerste twee bitjes te kijken. Maar dit gaat niet als hij decimaal moet rekenen. Het getal 12 (dec.) plus een aantal malen vier is ook deelbaar door vier, hoewel het tweede bitje 1 is. Het getal 10 (dec.) is niet deelbaar door vier hoewel de eerste twee bitjes beide nul zijn. Deze problemen zijn in het pro-

Afb. 1 Eenvoudig stroomdiagram van het klokprogramma.

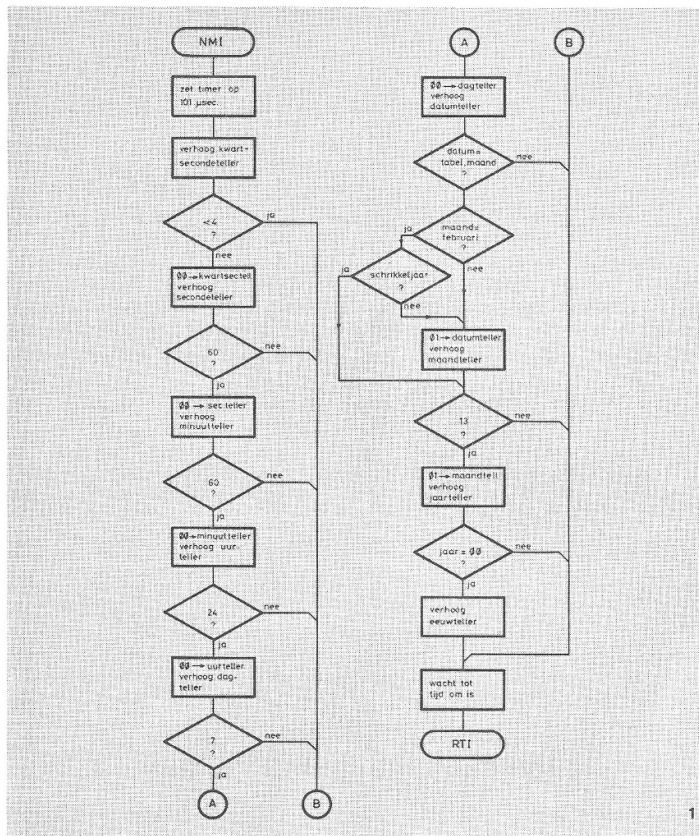
gramma (vanaf adres 03CC) als volgt opgelost. Allereerst wordt het eerste bitje in de carry geschoven. Treedt een carry op dan is het getal sowieso niet deelbaar door vier. Het is immers oneven. Als er geen carry optreedt, voert de computer een AND met 09 uit. De bitjes, die overblijven vormden vroeger het getal 12 (00001001 was 00010010: 12 dec.). Als het getal na deze AND nog steeds 09 is, was het getal 12 (plus een aantal malen vier) en dus deelbaar door 4.

Als het getal na deze AND 00 is geworden, was het getal voor deze AND XXX0XX0X (bin.) en hangt het van het eerste bitje af of het getal deelbaar door vier is. Aangezien de carryflag 0 was, voordat deze AND werd uitgevoerd, was vroeger het eerste bitje 0 en is dus het getal deelbaar door vier.

Werking van het programma

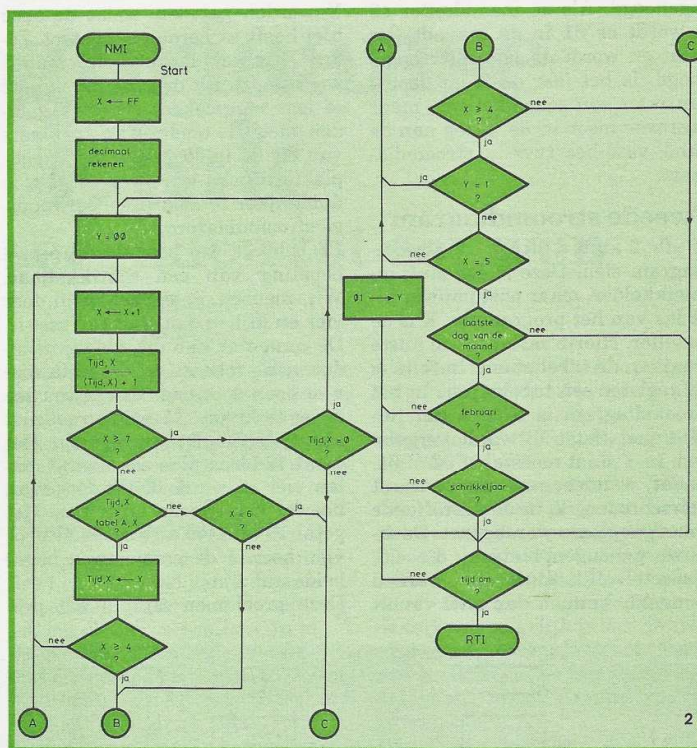
Zodra er een interrupt komt, wordt er naar adres 037B gesprongen. Hier worden allereerst de waarden, die op dat moment in de registers staan op de stack gezet. Daarna wordt de timer, die ervoor zorgt, dat de routine altijd even lang duurt, goed gezet en wordt de decimaal-flag gezet. Er wordt nu alleen nog maar decimaal gerekend.

X wordt nu op FF gezet en hier begint de eigenlijke rekenroutine. Y wordt met 00 geladen. Waarom dit is, is al uitgelegd. X wordt opgehoogd en zal telkens als er een nieuwe plaats moet worden veranderd, opnieuw worden opgehoogd. Vervolgens wordt de desbetreffende geheugenlocatie met één verhoogd. X wordt met 07 vergeleken. Dit om te kijken of de rekenroutine al 7 maal is doorlopen. Als dit het geval is, springt het programma





Afb. 3 Stroomdiagram van de uitlees-subroutine voor de Gregoriaanse klok.



Als X niet gelijk aan 06 is, met andere woorden, als het programma nog niet bezig is met het ophogen van de maand, wordt er gekeken of X groter of gelijk is aan 04. Als X namelijk tussen 00 en 04 zit, of gelijk is aan 00 of 04, dan zijn er geen uitzonderingen bij het ophogen van de betreffende geheugenlocaties. Is X nog niet gelijk aan

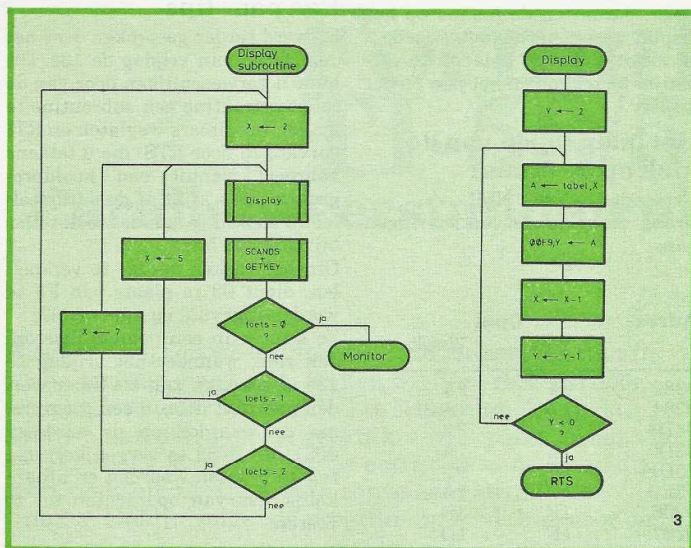
04, dan wordt naar het hoofdprogramma gesprongen. Vervolgens wordt gekeken of Y gelijk is aan 01. Y wordt namelijk pas 01 gemaakt als er een maand voorbij is. Als er een maand voorbij is, is het programma al voorbij de plaats, waar uitzonderingen worden gemaakt en kan het programma dat deel overslaan.

Als Y niet gelijk aan 01 is komen we bij de uitzonderingen. X wordt vergeleken met 05. En als X gelijk aan 05 is kijkt de computer dus naar de datum. Zoals u wellicht weet, heeft niet elke maand eenzelfde aantal dagen. Bovendien bestaat bijvoorbeeld 0 december niet. Op de eerste plaats moet worden gekeken met welke maand de com-

Lijst 1

[illegible]

Gregoriaanse klok



Lijst 1 Interruptroutine voor de Gregoriaanse klok.

Lijst 2 Uitleesroutine voor de Gregoriaanse klok.

puter te maken heeft en op de tweede plaats moet, als de hoogste waarde wordt overschreden, 01 in deze geheugenplaats worden gezet. Y wordt nu geladen met de maand en dient tevens als tabelwijzer. Als het bijvoorbeeld 31 januari is wordt Y geladen met 01 en wordt er, tevens via Y, gewezen naar de eerste waarde in de tabel, waar het aantal dagen in de maand wordt aangeduid.

De waarde in de tabel wordt vergeleken met de datum en zijn deze twee waarden niet aan elkaar gelijk, dan is er niets aan de hand. Als deze twee waarden wel gelijk zijn komt er nog een probleem op-

duiken. Wordt er vergeleken met de maximale waarde voor de maand februari (adres 03BA:C0 02)? Wordt er niet met de maximale waarde voor de maand februari vergeleken, dan wordt Y geladen met 01 en wordt deze waarde in de datumteller gezet. Hiermee is gelijk het tweede probleem opgelost. Wordt er wel met de maximale waarde voor de maand februari vergeleken, dan moet worden bepaald of er een schrikkeljaar is. (Deze routine begint bij het label „schrik“.) Hoe deze routine werkt is al eerder uitgelegd.

Tenslotte worden aan het eind van

de interruptroutine de registers teruggedhaald van de stack, zodat de computer met de juiste waarden in de registers verder kan gaan op het punt waar hij werd geïnterrupteerd.

Afregeling

Toen wij het programma lieten werken, was er niets af te regelen, maar als iemand het programma tot bijvoorbeeld het jaar 2000 wil laten lopen, zonder bij te hoeven regelen, zal hij zeer nauwkeurige meetinstrumenten, een zeer stabiel kristal, een zeer stabiele netvoeding en een constante temperatuur in de computerruimte moeten hebben. Iemand, die sneller tevreden is, zal, na uitproberen, de waarde op adres 0381 aanpassen.

Gelijkzetten

De waarden op de adressen 0081 tot 0089 kunnen, terwijl de klok loopt worden gewijzigd, door op die adressen de juiste waarden in te typen.

```
00700 0000 00 10      BCC  END      HAVING FEBRUARY?
00800 0000 00 02      CPYH 402
00900 0000 00 06      BCD  SCHRIK  JN 00H ER EEN SCHRIKKELJAAR ZIJN
00920 0000 00 01      RESET
00940 0000 00 05      LDIH 401  W=01.0 IS GEEN DAG
00960 0000 00 06      STC  DATUM
00980 0000 00 06      BNE  GEENH  ONCONDITIONEEL
009A0 0000 00 20      SCHRIK  CPYH 403  MFG HET 29 FEBRUARI?
009C0 0000 00 20      BCD  RESET  JN 00H GEEN AANPASSING
009E0 0000 00 07      LDIH 404  HIER JAAR
00A00 0000 00 07      NOP
00A20 0000 00 07      NOP
00A40 0000 00 07      LDIH 405  SCHUIF BIT 0 IN CARRY
00A60 0000 00 07      BCS  RESET  ONEVEN, GEEN SCHRIKKELJAAR
00A80 0000 00 07      ANDH 999  WAKEER
00AA0 0000 00 07      BCC  END  00H GEEN SCHRIKKELJAAR
00AC0 0000 00 07      CPYH 405  ZIJN DE "1" DAN
00AE0 0000 00 07      BNE  RESET  SCHRIKKELJAAR
00B00 0000 00 07      BIT  TINEH  WICHT TOT DE
00B20 0000 00 07      BPL  END  TIJD OM IS
00B40 0000 00 07      LDIH 404  STA  TINEH
00B60 0000 00 07      STA  TINEH
00B80 0000 00 07      PLA
00BA0 0000 00 07      TRV
00BC0 0000 00 07      PLA
00BE0 0000 00 07      TRV
00C00 0000 00 07      PLA
00C20 0000 00 07      RTI
00C40 0000 00 07      TAECLA = #04
00C60 0000 00 07      = #60 SECONDEN
00C80 0000 00 07      = #60 MINUTEN
00CA0 0000 00 07      = #24 UREN
00CC0 0000 00 07      = #30 DAGEN
00CE0 0000 00 07      = #12 DAGEN IN EEN MAAND
00D00 0000 00 07      TAECLA = #12
00D20 0000 00 07      = #12 JANUARI
00D40 0000 00 07      = #12 FEBRUARI
00D60 0000 00 07      = #12 MARCH
00D80 0000 00 07      = #12 APRIL
00DA0 0000 00 07      = #12 MEI
00DC0 0000 00 07      = #12 JUNI
00DE0 0000 00 07      = #12 JULI
00E00 0000 00 07      = #12 AUGUSTUS
```

```
02400 0000 00 31      = #31 SEPTEMBER
02420 0000 00 31      = #31 OKTOBER
02440 0000 00 31      = #31 NOVEMBER
02460 0000 00 31      = #31 DECEMBER
```

Lijst 2

```
00000 0000 00 00      HOORBEELD VAN EEN PROGRAMMA
00020 0000 00 00      OM DE KLOK UIT TE LEZEN
00040 0000 00 00      ORG #0200
00060 0000 00 02      RUN
00080 0000 00 02      TERUG
00100 0000 00 02      JSR 0000 00 02  HAVEL DE TIJD OP
00120 0000 00 02      JSR 0000 00 02  SCHRIK LUCHT DE UITLEESING OP
00140 0000 00 02      JSR 0000 00 02  GETKEY HAVEL DE INGEDEPUNTE
00160 0000 00 02      CPYH 400  TOETS "0" INGEDEPUNTE?
00180 0000 00 02      BNE  GEEN  NIET VERDER
00200 0000 00 02      JMP  MONTR  JA NAAR MONITOR
00220 0000 00 02      CPYH 401  TOETS "1" INGEDEPUNTE?
00240 0000 00 02      BNE  GEEN  NIET VERDER
00260 0000 00 02      LDIH 405  JA NAAR DAG EN MAAND
00280 0000 00 02      JSP  TERUG  BIJ DISPLAY
00300 0000 00 02      CPYH 402  TOETS "2" INGEDEPUNTE?
00320 0000 00 02      BNE  RUN  NIET, OPHEID
00340 0000 00 02      LDIH 407  JA NAAR DATUM
00360 0000 00 02      BNE  TERUG  ONCONDITIONEEL
00380 0000 00 02      DISPL 0000 00 02  KIES DISPLAY
00400 0000 00 02      LDIH 408  HAVEL INHOUT
00420 0000 00 02      STAV  INH  ZET DIT IN HET DISPLAY
00440 0000 00 02      DEY
00460 0000 00 02      BPL  D15
00480 0000 00 02      RTS
```



Gregoriaanse klok

Voorbeeld van een uitleesroutine voor de klok (lijst 2)

Het stroomdiagram ziet u in afb. 3. Register X is in dit programma de pointer, die bepaalt, wat er op het display verschijnt. In het begin is X geladen met 02, zodat de uren, minuten en seconden zichtbaar worden. Eerst wordt de subroutine aangeroepen, die de door X aangegeven waarden op het display zet. Daarna wordt de subroutine „SCANDS” aangeroepen. Vervolgens wordt de waarde van de ingedrukte toets opgehaald. Is de toetswaarde 00, dan stopt het programma; er wordt rechtstreeks naar de monitor gesprongen en de klok loopt dus gewoon door (er is geen rezet nodig). Als de toetswaarde 1 is dan wordt X met 05 geladen en worden de dag van de

week, datum en de maand op het display gezet. Als de toetswaarde 2 is, wordt X met 07 geladen en verschijnt de maand en het jaar op het display.

Het laten lopen van de klok op de Junior

Verbindt 1RQ met NMI.

Breng de volgende veranderingen aan:

Adres	Data	
	Was	Wordt
0383	04	F4
0384	17	1A
03D8	07	F7
03D9	17	1A
03DF	0F	FF
03E0	17	1A
0206	1F	8E
0207	1F	1D
0209	6A	F9
020A	1F	1D
0210	22	33

Een paar tips

Er werd eerder gesproken over het detecteren van vrijdag de 13e. Dit kunt u verwezenlijken door van de interruptroutine een subroutine te maken (de timers weglaten en RTI vervangen door RTS) die u telkens aanroept vanuit een hoofdprogramma dat kijkt of de datumteller 13 en de dag-van-de-week-teller bijvoorbeeld 5 is.

Om het zoeken verder te versnellen, dient 03 in plaats van FF te worden ingevuld op adres 0387.

Zo kunt u in een paar milliseconden zien, wanneer het vrijdag de 13e is. De klok kan als timer worden gebruikt door in een programma de waarden op de adressen 0081 t/m 0088 te vergelijken met zelf te bepalen waarden en afhankelijk daarvan opdrachten uit te voeren.