

パーソナルコンピュータとプリンタによるデータレコーダの試作

松原周信*

Development of a data recorder by a personal computer and printer

CHIKANOBU MATSUBARA *

Abstract : A simple 8 channel data recorder was developed which was constructed by a personal computer with A-D converter and a printer. Sampling interval was created with a built-in timer IC (8253), A-D converted data were written into memory ICs not by DMA but with data transfer instruction, and graphic data display on CRT was not executed with graphic controller IC but by writing data into VRAM directly. The printer was driven with graphic command (ESC/Page mode) directly. As the result, the execution speed was very high, and the software for data recording can run on many kinds of personal computers and A-D converter boards by not modifying the program or in little. As this software can be altered at will by each user, this equipment is convenient for anyone engaging in measuring various kinds of electrical signal. As an example, whole body reaction time, electromyogram, goniogram etc. were measured simultaneously: satisfactory result was acquired. (Accepted September 10, 2001)

各種の測定装置等によって得られる電気信号は、インクや感熱紙、あるいは磁気テープやメモリIC¹⁻⁹⁾を用いる装置によって記録することができる。これらのうち、メモリICに記録する装置においては、きわめて短時間内の事象から、かなり長時間にわたる変化までを記録し、また、測定後これらを液晶画面あるいは記録紙等に何度でも描画することができるなど、比較的安価な製品にあっても概して大変便利にできているものが多い。

しかし、たとえば某社製の装置（HIOKI 8808 MEMORY HiCORDER）では、測定のためには必ず「開始」キー、さらに「トリガ」キーと、2個のキーを一定以上の時間間隔をおいて押す必要があったり、また、RS-232Cポート通して装置の設定を変更する場合に、装置が待機または測定中は指令を受信はするがこれを無視するにもかかわらず、RS-232Cポートから待機または測定中であるかどうかを知る方法がないなどの問題点がある。これらは、一メーカーの一製品をもって、多種多様なユーザーの要求をすべて満たすことが不可能なために生起する問題であり、別なメーカーの別な製品において

もまた、あるユーザーにとって、ある類似の問題が発生する事態は、想像に難くない。

また、これらの装置を、パーソナルコンピュータからの指令で操作することが広く行われているが、測定装置に記録されたデータを、RS-232Cポートを通してパーソナルコンピュータに取り込むためには、通常かなりの時間を要する。パーソナルコンピュータにメモリICが搭載されているにもかかわらず、一旦測定装置にデータを記録しこれを転送することは、ハードウェア資源を余分に準備した上で、さらに時間を無駄遣いすることになる。

これに対し、パーソナルコンピュータにA-D変換ボード（またはカード）を装着し、直接データを取り込めば、ユーザの思い通りの操作性が、非常に安価に実現可能である。ただし、A-D変換ボードの入力信号のレンジは、通常、プログラムによって自由に変更できるようにはなっていないので、場合によってはそのためのプリアンプを使用する必要がある。しかし、各種測定装置のアナログ出力は、多くの場合そのレベルを調節できるようになっているので、直接A-D変換ボードに人力でき

* 京都府立大学人間環境学部食保健学科健康科学研究室

Laboratory of Health Science, Department of Food Sciences and Nutritional Health, Faculty of Human Environment, Kyoto Prefectural University

る場合が少なくない。

そこで、A-D変換ボードを装着したパーソナルコンピュータと、描画コマンドを有するプリンタを使用し、データレコーダとしての機能を有する装置を構成した。そのプログラムにおいては、どのようなA-D変換ボードでも使用できるように、また、ペンティアムに限らず、8086、V30を含む多くのCPUを搭載したパーソナルコンピュータで使用でき、さらに、多くの機種にできるだけ手数をかけず移植できるように配慮した。

方法

A-D変換ボードを装着したパーソナルコンピュータ、および描画コマンドを有するプリンタを用いて、データレコーダを構成することを意図した。その際、サンプリングインターバルは、パーソナルコンピュータにはじめから搭載されているタイマICによって作成し、A-D変換したデータは、DMAではなくデータ転送命令を用いてメモリICに書き込み、モニタ画面への描画はグラフィックコントローラICを介することなく直接VRAMにデータを書き込むことによって行わせることとした。

この条件のもと、手許に所蔵していたありあわせの資料を用い、データレコーダを構成した。すなわち、12ビット差動8チャンネル（またはシングルエンド16チャンネル）A-D変換ボード（カノープス ADX-98H）をI/O拡張ボックス（アドテックシステムサイエンス ANE-451）に装着し、これをノートタイプのパーソナルコンピュータ（NEC PC9801NX/C）に接続したうえ、パラレルポートにレーザプリンタ（エプソン LP-9200SX）を接続した。パーソナルコンピュータには、3倍クロックのオーバードライブプロセッサ（アイ・オー・データ機器 PK-NXC75）が装着されていた。また、A-D変換ボードには、4ビットずつのデジタル入出力端子が付属していた。

なお、実際に測定データを得て装置の動作を確認するため、第1図に示した通り、LEDの発光による刺激発生装置（自作）の作動に応じて跳躍する際の全身反応時間測定装置を構成し、1.024秒間にわたって、このときの外側広筋と腓腹筋の筋電図、腓腹筋の筋電図積分値、膝および足関節のゴニオグラム、跳躍用の台にかかる荷重のあわせて6種類のアナログ信号を測定した。その際、光刺激発生装置は、A-D変換ボード付属のデジタル出力によって作動させた。使用した測定装置は、筋電図アンプ（日本光電 AM-601G）2台、多用途積分ユニット（日本光電 EI-601G）1台、足関節および膝関節用各1組ずつの、ゴニオメータ（自作）ならびにアンプ（自作）、荷重測定台（自作）ならびにストレインゲージアンプ（共立電子産業 STR-1N-05）1組であった。

プログラムはAppendixに示した通り、高速な動作が

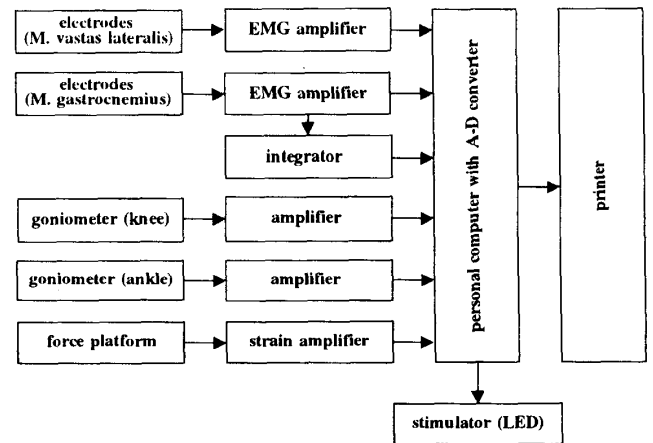


Fig. 1 An example of data recorder constitution. Analog data were inputted, A-D converted, and A4 sized charts were printed out.

要求される箇所、すなわちA-D変換とその結果をメモリに書き込む部分を機械語で記述し、その他の部分はMS-DOS版N-88 BASIC (86)を用いた。機械語のプログラミングに際しては、8086の有する命令のみを用いた。プリンタは、ESC/Pageモードで使用した。

結果

第2図は、2msごとに512回、1.024秒間にわたってサンプリングを行い、測定後A4サイズ用の用紙にプリントアウトした結果の一例で、上から順に、外側広筋の筋電図、膝関節のゴニオグラム、腓腹筋の筋電図、その全波整流波形、その積分値、足関節のゴニオグラム、跳躍用台への荷重、およびその上下反転波形である。筋電図の全波整流波形と荷重の上下反転波形は、筋電図ならびに荷重の測定値をもとに、ソフトウェアによって作図したものである。全身反応時間が、神経伝達時間ならびに筋収縮時間から構成されること、筋電図の発現から関節の角度変化の開始までには一定の時間が必要なことをはじめ、300ms前後の短時間に生起する現象を、この図から明瞭に読みとることが可能である。

ただし、Appendixに示したプログラムは、チャンネル1から8までの差動入力データを順次A-D変換してそのままモニタ画面に表示し、測定終了後さらにプリンタに出力する基本的な動作を行わせるもので、第2図はこれに多少の修正を加えたプログラムによって描画させたものである。なお、使用したパーソナルコンピュータのCPUは80486DX 60MHz (20MHz×3)、外部（メモリ）クロック20MHzであったが、サンプリングインターバル2msでは、8チャンネル分のA-D変換、データのメモリへの書き込み、およびモニタ画面への描画が可能であった。しかし、1msごとにサンプリングを行わせると、プログラムが追従できなかった。



Fig. 2 An example of acquired data. Sampling interval was 2 ms, and 512 data were plotted on each channel: full scale of horizontal axis was 1.024 s. From upper part, EMG of musculus vastas lateralis, goniogram of knee joint, EMG of musculus gastrocnemius, its rectified wave form, its integrated wave form, goniogram of ankle joint, vertical load of the subject, and its inverted wave form were plotted, respectively.

考 察

パーソナルコンピュータにA-D変換ボードを装着し、これにプリンタを接続してデータレコーダを試作したが、その際、サンプリングのタイミングは、パーソナルコンピュータにはじめから搭載されているタイマIC(8253相当)を用いて割り込みを発生させることによって作成した。また、A-D変換したデータは、DMAではなくデータ転送命令を用いてメモリICに書き込んだ。このため、A-D変換機能を有しさえすればどのようなA-D変換ボードでも、そのボード固有の条件に合わせてプログラムをわずかに変更しさえすれば使用することができる。また、8086の有する命令しか使用していないので、同一のA-D変換ボードを使用する限り、NECのPC98シリーズのパーソナルコンピュータであれば、すべて同じプログラムを使用することができる。また、モニタ画面への描画を、グラフィックコントローラICを介することなく直接VRAMにデータを書き込むことによって行わせたので、PC/AT互換機をはじめ他機種のパーソナルコンピュータへの移植も、基本的には比較的容易である。

Appendixに示したプログラムにおいて、サンプリングのための割り込みは、割り込みコントローラIC(8259)の設定をデフォルトから変更し、優先順位を最下位とした。したがって、サンプリングインターバルが比較的長く、サンプリングのタイミングが多少ずれても差し支えない場合には、測定しながら何らかのデータをRS-232Cポートから入出力するなどしても、そのデータが欠損したり変化したりするようなことはない。

また、このプログラムは、N-88 BASIC(86)から機械語のプログラムを呼び出しているが、機械語部分は当然ながらFORTRANやCをはじめとする他のMS-DOS上の高級言語、あるいはWindows上の高級言語から呼び出す際にも全く同じでよいので、MS-DOSあるいはWindows上などの、BASIC以外の言語に変更することも容易である。データをフロッピーディスクやハードディスクなどに記録する場合、サイズが64k-2バイト以下であれば、BASICの場合バイナリセーブ命令(BSAVE)を使用することによって、プログラミングが容易にでき、かつ高速な処理が可能である。こうして記録したファイルは、BASICのバイナリロード命令(BLOAD)のほか、FORM='BINARY'の使えるFORTRANなどでもそのまま読み込み可能である。なお、AppendixのようにデフォルトでPC-98シリーズのB0000H番地から書き込めばデータはVRAMに記録されるが、VRAMに書ききれないサイズの場合には、エクステンドメモリまたはバンクメモリに書き込むように修正すればよい。

試作に用いたA-D変換ボードには4ビットずつのデジタル入出力端子が備わっていたので、測定の開始、プリンタへの出力およびキャンセルの3指令は、外部に設

けたスイッチボックスからの入力によって与え、また、出力3ビットを用い、測定準備完了、測定中、および測定が終了してプリントアウトさせるかどうかの指示待ちの3状態を、LEDを用いたパイロットランプによって表示させ、残る1ビットの出力を、光刺激発生装置のトリガに用いた。デジタル入出力端子のないA-D変換ボードの場合には、RS-232CポートのDSRとCDを2ビットの入力、TXD、RTS、およびDTRを3ビットの出力として用いればよい。あるいは、入力はキーボードまたはマウス、状態の表示にはモニタ画面を使用することもできる。このような方法ではどうしても入出力端子数が不足する場合には、A-D変換ボードのほかにパラレルI/Oボードを併用するか、速度の点で問題がなければ、本来のRS-232Cプロトコルによればよい。

今回パーソナルコンピュータを用いて試作したデータレコーダは、CPUが80486で、現在一般に多く使用されている機種とくらべるとかなり非力であるが、それでも2msごとにサンプリングを行うことができた。したがって、筋電図、心電図、脳波などの生体電気現象や、ヒトの動作に伴う荷重や関節角度の変化などを記録するには十分である。これより高速なCPUを搭載したパーソナルコンピュータを使用すれば、さらに短いサンプリングインターバルでのデータ収集が可能である。また、複数のA-D変換ボードを装着すれば、16チャンネル、32チャンネル、あるいはそれ以上多チャンネルのデータ収集が可能である。さらに、プログラムは自由に変更することができるので、たとえばキーボードのキー、外部に設けたスイッチ、またはマウスのボタンのいずれを用いて操作するか、また、操作1回で測定を開始するか、誤操作防止のために2回の操作を必要とさせるか、あるいは操作後ランダムな時間間隔の後測定を開始させるか、その他、外部からの信号によって開始させるかというような点に関しては、思い通りに動作させることができる。また、データの線幅、縦横の罫線のデザインなど記録紙への表示方法や、さらにはサンプリングインターバル、波形の合成等も必要に応じ自由に設定することが可能であり、これらに関しキー操作等によって、プログラムを選択したり、パラメータを入力したりすることについても自由な設定が可能である。

また、この装置は、基本的にはパーソナルコンピュータ、A-D変換ボード、およびプリンタがありさえすれば構成し、使用することができる。パーソナルコンピュータとプリンタがすでにあるとすれば、これにA-D変換ボードを装着するだけでよい。なお、パーソナルコンピュータとA-D変換ボードについては、前述の通り原則としてどのような機種でも使用することができるが、プリンタについては、描画コマンドの公開されている機種でなければならない。そのうち、ESP/Pageコマンドを受け付けるプリンタであれば、プログラムを全く変更することなく使用することができる。PC-PR201、

HP/GL,あるいはその他の系統のコマンドを受け付けるプリンタやプロッタも,その機種にあわせプログラムを修正すれば使用可能である。ただし,機種により,その修正が軽微なものにとどまるとはかぎらない。

なお,一般にインク書きまたは感熱記録方式のレコーダは,ペンのスピードにもチャートスピードにも限界がある。また,8チャンネルまたはそれ以上のレコーダは,かなり高価である。これに対し,試作した装置のほうが,かなり高い周波数や大きな振幅,および速い現象をも記録することができるうえ,かなり安価である。また,メモリICに記録し,測定後液晶画面あるいは記録紙等に描画する装置については,各メーカーの各機種がそれぞれに特長を有するとしても,すべてのユーザーに満足に行く仕様というものはあり得ない。今回試作した装置では,この問題を完全に回避することが可能である。

文 献

- 1) Baharestani, H., W. J. Tompkins, & J. G. Webster (1979) Heart rate recorder. *Med. & Biol. Eng. & Comput.* 17, 719-723.
- 2) 松原周信 (1991) 携帯式心拍数, 体温, 歩数長時間記録装置の試作. *京府医大誌* 100, 623-630.
- 3) 柴田政広, 山越憲一, 島津秀昭, 田中志信, 豊島健, 戸川達男 (1980) 生体情報無拘束簡易収集記録装置の試作. *医器材研報* 14, 119-123.
- 4) 柴田政広, 川原田淳, 島津秀昭, 山越憲一, 神谷瞭 (1983) 超小型デジタルレコーダによる運動時生体情報の無拘束計測. *体力科学* 32, 58-65.
- 5) 島津秀昭, 柴田政広, 山越憲一, 戸川達男 (1979) 無拘束簡易心拍数記憶装置. *医器材研報* 13, 49-52.
- 6) 田村俊世, 戸川達男 (1983) 携帯可能な多目的生体情報監視装置. *日生誌* 45, 245-250.
- 7) Taylor, C. B, H. C. Kraemer, D. A. Bragg, L. E. Miles, B. Rule, W. M. Savin, & R. F. DeBusk (1982) A new system for long-term recording and processing of heart rate and physical activity in outpatients. *Computers and Biomedical Research*. 15, 7-17.
- 8) 豊島健, 戸川達男 (1980) 汎用無拘束生体監視装置 — その深部体温および心拍数記録への応用 — . *医器材研報* 14, 27-41.
- 9) 山本高司, 真鍋篤広, 加藤好信, 藤松博 (1981) 携帯式24時間心拍数記録装置. *中京体育学研究* 22 (1), 15-22.

Appendix Program of the data recorder

```

10000 CLEAR &H21,...&H2000
10010 *START
10020 OUT &HD6,&H10          ' PLO ON
10030 DEF SEG=SEGPTR(2)
10040 SCREEN 2
10050 RESTORE 10080
10060 FOR I=0 TO 286:READ D$:POKE I,VAL("&H"+D$):NEXT
10070 FOR I=287 TO 521:READ D$:POKE I,VAL("&H"+D$):NEXT
10080 DATA 00,00          ' counter = data writing address
10090 '
10100 DATA 31,c0          ' XOR AX,AX
10110 DATA 8e,d8          ' MOV DS,AX
10120 DATA c7,06,20,00,29,00 ' MOV WORD PTR[0020],0029
10130 DATA 8c,0e,22,00    ' MOV [0022],CS
10140 DATA b0,34          ' MOV AL,34          ; 8253 counter 0 LSB->MSB
10150 DATA e6,77          ' OUT 77,AL          ; "
10160 DATA b0,33          ' MOV AL,33          ; 8253 counter 0 write LSB
10170 DATA e6,71          ' OUT 71,AL          ; "
10180 DATA b0,13          ' MOV AL,13          ; 8253 counter 0 write MSB
10190 DATA e6,71          ' OUT 71,AL          ; "
10200 '[2.4576 MHz]      1 ms: 2457.6->2458=&H099A 10 ms: 24576=&H6000
10210 '                  2 ms: 2457.6*2=4915.2->4915=&H1333
10220 DATA e4,02          ' IN AL,02          ; 8259 interrupt control
10230 DATA 24,fe          ' AND AL,FE          ; "
10240 DATA e6,02          ' OUT 02,AL          ; "
10250 DATA b0,00          ' MOV AL,00          ; set up A-D board
10260 DATA e6,d0          ' OUT D0,AL          ; "
10270 DATA e6,d3          ' OUT D3,AL          ; "
10280 DATA cf            ' IRET
10290 '
10300 DATA fb            ' STI              0029;
10310 DATA 50            ' PUSH AX
10320 DATA b0,20          ' MOV AL,20          ; 8259 interrupt control
10330 DATA e6,00          ' OUT 00,AL          ; "
10340 DATA 1e            ' PUSH DS
10350 DATA 53            ' PUSH BX
10360 DATA 56            ' PUSH SI
10370 DATA b8,00,b0      ' MOV AX,B000
10380 DATA 8e,d8          ' MOV DS,AX
10390 DATA 2e            ' CS:
10400 DATA 8b,1e,00,00   ' MOV BX,[0000]
10410 ' ----- CH 1 to CH N -----
10420 ' I/O address: D0 -->
10430 DATA be,00,00      ' MOV SI,0000
10440 DATA b4,20          ' MOV AH,00          ; 00 = CH 1
10450 DATA 88,e0          ' MOV AL,AH          0041; CH select and sample
10460 DATA e6,d6          ' OUT D6,AL          ; "
10470 DATA b0,10          ' MOV AL,10          ; delay
10480 DATA fe,c8          ' DEC AL              0047; "
10490 DATA 75,fc          ' JNZ 0047           ; "
10500 DATA b0,04          ' MOV AL,04          ; start A-D conversion
10510 DATA e6,D0          ' OUT D0,AL          ; "
10520 DATA e4,d0          ' IN AL,D0           004F; end of A-D conversion ?
10530 DATA a8,80          ' TEST AL,80         ; "
10540 DATA 74,fa          ' JZ 004F            ; no then LOOP
10550 DATA e4,d2          ' IN AL,D2           ; get value of LSB 4 bits
10560 DATA 88,00          ' MOV [BX+SI],AL     ; write to memory (bit 7->4)
10570 DATA e4,d4          ' IN AL,D4           ; get value of MSB 8 bits
10580 DATA 88,40,01       ' MOV [BX+SI+1], AL ; write to memory (bit 7->0)
10590 'DATA 81,c6,00,10   ' ADD SI,1000
10600 'DATA 81,c6,00,20   ' ADD SI,2000
10610 'DATA 81,c6,00,04   ' ADD SI,0400
10620 'DATA fe,c4          ' INC AH              ; next CH
10630 'DATA 80,fc,28      ' CMP AH,08           ; 08 --> CH 1 to 8
10640 'DATA 75,d8          ' JNZ 0041
10650 ' -----
10660 DATA 51            ' PUSH CX
10670 DATA 52            ' PUSH DX
10680 DATA 89,d8          ' MOV AX,BX
10690 DATA 31,d2          ' XOR DX,DX
10700 'DATA b9,0e,00      ' MOV CX,000E        ' 0EH=7*2 (2048 data)
10710 'DATA b9,1c,00      ' MOV CX,001C        ' 0EH=14*2 (4096 data)
10720 'DATA b9,04,00      ' MOV CX,0004        ' 04H=2*2 (1 sec 2 data/dot)
10730 'DATA f7,f1          ' DIV CX              ' AX=(DX*256^2+AX) \ CX
10740 '                      '                      ' DX=(DX*256^2+AX) MOD CX
10750 'DATA 80,fa,0c      ' CMP DL,0C          ' IF DL MOD 14=12 THEN
10760 'DATA 80,fa,1a      ' CMP DL,1A          ' IF DL MOD 28=26 THEN
10770 'DATA 80,fa,02      ' CMP DL,02          ' IF DL MOD 04=02 THEN
10780 'DATA 75,7c          ' JNZ L10            ' (1AH=26, 28=14*2)
10790 '
10800 'DATA 31,d2          ' XOR DX,DX          ' ( AX = previous result )
10810 'DATA b9,08,00      ' MOV CX,0008
10820 'DATA f7,f1          ' DIV CX              ' AX=(DX*256^2+AX) \ CX
10830 '                      '                      ' DX=(DX*256^2+AX) MOD CX
10840 '                      '                      ' DL <- 0,1,...,7
10850 'DATA 88,d1          ' MOV CL,DL
10860 'DATA 53            ' PUSH BX
10870 '
10880 '                      -- 80*65-80*64+3 --
10890 'DATA be,53,00      ' MOV SI,0053        ' 53H=80*65-80*64+3
10900 '
10910 '                      -- J8 --
10920 'DATA 01,c6          ' ADD SI,AX          ' ( AX = previous result )
10930 '                      ----- (CH)
10940 'DATA b5,08          ' MOV CH,08
10950 '
10960 'DATA b8,00,b0      ' MOV AX,B000 L1:
10970 'DATA 8e,d8          ' MOV DS,AX
10980 '                      ----- (AH)
10990 'DATA 53            ' PUSH BX
11000 'DATA b4,07          ' MOV AH,07
11010 'DATA b4,1c          ' MOV AH,1C          14 data/dot
11020 'DATA b4,04          ' MOV AH,04          2 data/dot
11030 'DATA 43            ' INC BX
11040 'DATA 8a,07          ' MOV AL,[BX] L2:
11050 '
11060 'DATA a8,80          ' TEST AL,80         ' CMP -> OFB
11070 'DATA 75,04          ' JNZ L3
11080 'DATA 0c,80          ' OR AL,80
11090 'DATA eb,02          ' JMP L4
11100 'DATA 24,7f          ' AND AL,7F L3:
11110 '
11120 'DATA f6,d0          ' NOT AL             L4:
11130 'DATA d0,e8          ' SHR AL,1
11140 '
11150 'DATA 80,fc,07      ' CMP AH,07          ' initialize MAX, MIN
11160 'DATA 80,fc,1c      ' CMP AH,1C          ' initialize MAX, MIN
11170 'DATA 80,fc,04      ' CMP AH,04          ' initialize MAX, MIN
11180 'DATA 75,06          ' JNZ L5
11190 'DATA 88,c2          ' MOV DL,AL
11200 'DATA 88,c6          ' MOV DH,AL
11210 'DATA eb,0c          ' JMP L7
11220 '
11230 'DATA 38,f0          ' CMP AL,DH L5:
11240 'DATA 72,02          ' JB L6
11250 'DATA 88,c6          ' MOV DH,AL
11260 'DATA 38,d0          ' CMP AL,DL L6:
11270 'DATA 77,02          ' JA L7
11280 'DATA 88,c2          ' MOV DL,AL
11290 'DATA 4b            ' DEC BX L7:
11300 'DATA 4b            ' DEC BX
11310 'DATA fe,cc          ' DEC AH
11320 'DATA 75,d3          ' JNZ L2
11330 'DATA 5b            ' POP BX
11340 '
11350 'DATA b8,00,a8      ' MOV AX,A800
11360 'DATA 8e,d8          ' MOV DS,AX
11370 '
11380 '                      -- number of l=dot --
11390 'DATA 28,d6          ' SUB DH,DL          ' DH = max value

```

```

11400 DATA fe,c6      'INC DH          '-> DH = counter
11410 '                (dot number of 1 line)
11420 DATA 80,fd,04   'CMP CH,04      'move to right half area
11430 DATA 75,04      'JNE L8
11440 DATA 81,ee,a8,70 'SUB SI,70A8    '70A8H=80*90*4+40
11450 '
11460 DATA 56         'PUSH SI
11470 '                (dot number of 1 line)
11480 ' -- initial value of I*80 --
11490 DATA b0,50      'MOV AL,50 L8:  '50H=80
11500 DATA f6,e2      'MUL DL         'AX <- AL*DL
11510 DATA 01,c6      'ADD SI,AX
11520 '
11530 DATA b0,80      'MOV AL,80
11540 DATA d2,e8      'SHR AL,CL     'CL = 0,1.....,7
11550 DATA 08,04      'OR [SI],AL L9:
11560 '
11570 ' -- add I*80 --
11580 DATA 83,c6,50   'ADD SI,+50    '50H=80
11590 DATA fe,ce      'DEC DH
11600 DATA 75,f7      'JNZ L9
11610 '
11620 DATA 5e         'POP SI
11630 ' -----
11640 DATA 81,c6,20,1c 'ADD SI,1C20   '1C20H=80*90
11650 'DATA 81,c3,00,10 'ADD BX,1000
11660 'DATA 81,c3,00,20 'ADD BX,2000
11670 'DATA 81,c3,00,04 'ADD BX,0400
11680 DATA fe,cd      'DEC CH
11690 DATA 75,96      'JNZ L1
11700 ' -----
11710 DATA 5b         'POP BX
11720 '                L10:
11730 DATA 5a         'POP DX
11740 DATA 59         'POP CX
11750 ' -----
11760 DATA 43         'INC BX
11770 DATA 43         'INC BX
11780 ' -----
11790 DATA 2e         'CS:
11800 DATA 89,1e,00,00 'MOV [0000],BX
11810 'DATA 81,fb,00,20 'CMP BX,2000
11820 'DATA 81,fb,8c,04 'CMP BX,048c   '291 dots/ch 291*2 data
11830 'DATA 81,fb,00,04 'CMP BX,0400   '256 dots/ch 256*2 data
11840 DATA 75,16      'JNZ 008C
11850 DATA e4,02      'IN AL,02      '8259 interrupt control
11860 DATA 0c,01      'OR AL,01
11870 DATA e6,02      'OUT 02,AL
11880 DATA 31,c0      'XOR AX,AX
11890 DATA 8e,d8      'MOV DS,AX
11900 'DATA c7,06,20,00,98,17 'MOV WORD PTR[0020],1798
11910 'DATA c7,06,22,00,80,fd 'MOV WORD PTR[0022],FD80
11920 'DATA c7,06,20,00,8f,02 'MOV WORD PTR[0020],028F
11930 'DATA c7,06,22,00,73,54 'MOV WORD PTR[0022],5473
11940 DATA 5e         'POP SI 008C:
11950 DATA 5b         'POP BX
11960 DATA 1f         'POP DS
11970 DATA 58         'POP AX
11980 DATA cf         'IRET
11990 ' -----
12000 'PROG22
12010 DATA 50         'PUSH AX
12020 DATA 53         'PUSH BX
12030 DATA 51         'PUSH CX
12040 DATA 52         'PUSH DX
12050 DATA 1e         'PUSH DS
12060 DATA 56         'PUSH SI
12070 DATA 57         'PUSH DI
12080 DATA b8,00,b0   'MOV AX,B000
12090 DATA 8e,d8      'MOV DS,AX
12100 DATA 33,db      'XOR BX,BX
12110 '
12120 DATA be,20,03   'MOV SI,0320   '0320H=800 Y-offset
12130 DATA b9,08,00   'MOV CX,0008
12140 DATA 51         'L1 PUSH CX
12150 DATA b4,1d      'MOV AH,1D
12160 DATA e8,be,00   'CALL PRN
12170 'DATA b9,00,10    'MOV CX,0800   '0800H=2048
12180 DATA b9,00,02   'MOV CX,0200   '0200H=512
12190 DATA 33,ff      'XOR DI,DI
12200 DATA eb,06,90   'JMP L3
12210 DATA b4,3b      'L2 MOV AH,3B   '3BH=";"
12220 DATA e8,b1,00   'CALL PRN
12230 DATA 51         'L3 PUSH CX
12240 '
12250 DATA 8b,c7      'MOV AX,DI
12260 DATA d1,e8      'SHR AX,1
12270 DATA 90,90      'SHR AX,1
12280 DATA 90,90      'SHR AX,1
12290 DATA 05,dc,05   'ADD AX,05DC   '05DCH=1500 X-offset
12300 DATA e8,4b,00   'CALL ASCPRN
12310 DATA b4,3b      'MOV AH,3B
12320 DATA e8,9d,00   'CALL PRN
12330 DATA 8b,07      'MOV AX,[BX]
12340 '
12350 DATA f6,c4,80   'TEST AH,80
12360 DATA 75,06      'JNZ L4
12370 DATA 80,cc,80   'OR AH,80
12380 DATA eb,04,90   'JMP L5
12390 DATA 80,e4,7f   'L4 AND AL,7F
12400 '                L5
12410 DATA f7,d0      'NOT AX
12420 DATA b1,04      'MOV CL,04
12430 DATA d3,e8      'SHR AX,CL
12440 DATA b9,68,10   'MOV CX,1068   '1068H=4200=YDIV*10
12450 DATA f7,e1      'MUL CX
12460 DATA b9,00,10   'MOV CX,1000   'DX*256*2+AX <- AX*DX
12470 DATA f7,f1      'DIV CX
12480 DATA 03,c6      'ADD AX,SI
12490 DATA e8,21,00   'CALL ASCPRN
12500 '
12510 'DATA 83,c7,09    'ADD DI,+09
12520 'DATA 83,c7,48    'ADD DI,+48
12530 DATA 43         'INC BX
12540 DATA 43         'INC BX
12550 DATA 59         'POP CX
12560 DATA e2,ba      'LOOP L2
12570 '
12580 DATA b4,6c      'MOV AH,6C
12590 DATA e8,6b,00   'CALL PRN
12600 DATA b4,47      'MOV AH,47
12610 DATA e8,66,00   'CALL PRN
12620 'DATA 81,c6,20,0d 'ADD SI,0D20   '0D20H=YDIV*8=420*8
12630 DATA 59         'POP CX
12640 DATA e2,9b      'LOOP L1
12650 '
12660 DATA 5f         'POP DI
12670 DATA 5e         'POP SI
12680 DATA 1f         'POP DS
12690 DATA 5a         'POP DX
12700 DATA 59         'POP CX
12710 DATA 5b         'POP BX
12720 DATA 58         'POP AX
12730 DATA cf         'IRET
12740 ' -----
12750 ' ASCPRN
12760 'DATA 3d,10,27    'CMP AX,2710   'IF AX>=10000 THEN A1
12770 'DATA 73,0d      'JAE A1
12780 'DATA 3d,e8,03    'CMP AX,03E8   'IF AX>=1000 THEN A2
12790 'DATA 73,18      'JAE A2
12800 'DATA 3d,64,00   'CMP AX,0064   'IF AX>=100 THEN A3
12810 'DATA 73,23      'JAE A3
12820 'DATA eb,32,90    'JMP A4
12830 '                'ELSE GOTO A4
12840 'DATA 33,d2      'A1 XOR DX,DX
12850 'DATA b9,10,27    'MOV CX,2710   '2710H=10000
12860 'DATA f7,f1      'DIV CX
12870 'DATA 04,30      'ADD AL,30
12880 'DATA 8a,e0      'MOV AH,AL
12890 'DATA e8,37,00   'CALL PRN
12900 '                'position of 10,000

```

```

12900 '
12910 DATA 8b,c2 ' MOV AX,DX ' AX <- DX(remainder)
12920 DATA 33,d2 'A2 XOR DX,DX '
12930 DATA b9,e8,03 ' MOV CX,03E8 ' 03E8H=1000
12940 DATA f7,f1 ' DIV CX ' AX <- (DX*256^2+AX) \ CX
12950 DATA 04,30 ' ADD AL,30 ' ASCII <- binary
12960 DATA 8a,e0 ' MOV AH,AL '
12970 DATA e8,27,00 ' CALL PRN ' position of 1,000
12980 '
12990 DATA 8b,c2 ' MOV AX,DX ' AX <- DX(remainder)
13000 DATA b1,64 'A3 MOV CL,64 ' 64H=100
13010 DATA f6,f1 ' DIV CL ' AX <- AX \ CL
13020 DATA 04,30 ' ADD AL,30 ' ASCII <- binary
13030 DATA 50 ' PUSH AX ' ----->
13040 DATA 8a,e0 ' MOV AH,AL '
13050 DATA e8,19,00 ' CALL PRN ' 100
13060 DATA 58 ' POP AX ' <-----
13070 '
13080 DATA 8a,c4 ' MOV AL,AH '
13090 DATA 32,e4 ' XOR AH,AH '
13100 DATA b1,0a 'A4 MOV CL,0A ' 0AH=10
13110 DATA f6,f1 ' DIV CL ' AX <- AX \ CL
13120 DATA 04,30 ' ADD AL,30 ' AH <- AX MOD CL
13130 DATA 50 ' PUSH AX ' ----->
13140 DATA 8a,e0 ' MOV AH,AL '
13150 DATA e8,08,00 ' CALL PRN ' 10
13160 DATA 58 ' POP AX ' <-----
13170 DATA 80,c4,30 ' ADD AH,30 ' ASCII <- binary
13180 DATA e8,01,00 ' CALL PRN ' 1
13190 DATA c3 ' RET
13200 '-----
13210 ' PRN
13220 DATA e4,42 '0000 IN AL,42
13230 DATA a8,04 '0002 TEST AL,04
13240 DATA 74,fa '0004 JE 0000 ; inverted by hard =JNE
13250 DATA 8a,c4 '0006 MOV AL,AH
13260 DATA e6,40 '0008 OUT 40,AL
13270 DATA b0,0e '000A MOV AL,0E
13280 DATA e6,46 '000C OUT 46,AL
13290 DATA b0,0f '000E MOV AL,0F
13300 DATA e6,46 '0010 OUT 46,AL
13310 DATA c3 '0012 RET
13320 '
13330 CLS
13340 CONSOLE,,0
13350 DEF SEG=&HA800
13360 ' horizontal lines
13370 FOR K=0 TO 26
13380 FOR J=0 TO 1
13390 FOR I=0 TO 36
13400 IF K MOD 6=4 THEN POKE 3+80*4+80*15*K+40*J+I,&H88 ELSE POKE 3+80*4+80*15*K+40*
J+I,&H80
13410 NEXT I
13420 NEXT J
13430 NEXT K
13440 ' vertical lines
13450 FOR K=0 TO 1
13460 FOR J=0 TO 4
13470 FOR I=0 TO 25
13480 POKE 3+80*9+I*80*15+J*9+K*40,&H80
13490 POKE 3+80*14+I*80*15+J*9+K*40,&H80
13500 NEXT I
13510 NEXT J
13520 NEXT K
13530 ' channel number
13540 FOR J=0 TO 1
13550 FOR I=0 TO 3
13560 PUT(J*320+8,I*90+50),KANJI(&H31+I+J*4)
13570 NEXT I
13580 NEXT J
13590 CLS:LOCATE 28,12
13600 COLOR 6: PRINT " Push [start] to measure. ": COLOR 0
13610 'INPUT "",F$
13620 IF (INP(&HD2) AND &HF)<>&HE THEN 13620 ' check SW0
13630 CLS
13640 '
13650 OUT &HD6,&H80: OUT &HD6,&H0 ' output signal
13660 DEF SEG=SEGPtr(2)
13670 S=2:CALL S
13680 DEF SEG=0
13690 IF PEEK(&H23)<>&H54 THEN 13690
13700 OUT &HD6,&H40 ' PL2,PL3 ON
13710 CLS:LOCATE 28,12
13720 COLOR 6: PRINT " Select [print] or [cancel]. ": COLOR 0
13730 IF (INP(&HD2) AND &HF)=&H7 THEN CLS 3: LOCATE ,,1: CONSOLE ,,1: GOTO *START
13740 IF (INP(&HD2) AND &HF)<>&HB THEN 13730 ' check SW3
13750 CLS
13760 CLS:LOCATE 33,12
13770 COLOR 6: PRINT " printing ": COLOR 0
13780 '
13790 '----- plot figure with ESC/Page (BASIC) -----
13800 '
13810 DEF SEG=&HA800
13820 GS$=CHR$(29)
13830 OPEN "LPT1:" AS #1
13840 '
13850 PRINT #1,GS$;"rhE"; ' hard reset
13860 PRINT #1,GS$;"2;0.01muE"; ' unit 0.01 mm
13870 PRINT #1,GS$;"2.5;0;3lwG"; ' line width 2.5 connection type=2
13880 PRINT #1,GS$;"0;0lpG"; ' normal line
13890 PRINT #1,GS$;"0;20;25;2dlG"; ' register dotted line
13900 PRINT #1,GS$;"1;15;75;2dlG"; ' register dotted line
13910 '
13920 'XOFF=1500: YOFF=800: XDIV=450: YDIV=420
13930 XOFF=1500: YOFF=800: XDIV=180: YDIV=420
13940 PRINT #1,GS$;"12.5wcf"; GS$;"7.68hpF"; GS$;"0;-20pP"; ' CHR size *.8
13950 '
13960 PRINT #1,GS$;XOFF;"X"; GS$;YOFF-200;"Y";
13970 'PRINT #1,"200ms/DIV 1V/DIV"
13980 PRINT #1,"10ms/DIV 1V/DIV"
13990 PRINT #1,GS$;XOFF+14000;"X"; GS$;YOFF-200;"Y";
14000 PRINT #1,DATES;" ";TIMES$
14010 PRINT #1,GS$;"1;1lpG"; ' registered line
14020 'PRINT #1,GS$;XOFF+XDIV*11;"";YOFF-200;"";XOFF+XDIV*29;"";YOFF-200;"IG";
14030 PRINT #1,GS$;XOFF+XDIV*11*2.5;"";YOFF-200;"";XOFF+XDIV*29*2.5;"";YOFF-200;"IG";
14040 '
14050 ' label CH
14060 FOR I=0 TO 7
14070 PRINT #1,GS$;XOFF-700;"X"; GS$;YOFF+YDIV*5+100+YDIV*8*I;"Y";
14080 PRINT #1,I+1
14090 NEXT I
14100 '
14110 ' horizontal lines
14120 FOR I=0 TO 66
14130 IF I MOD 8=5 THEN PRINT #1,GS$;"1;0lpG"; ELSE PRINT #1,GS$;"1;1lpG";
14140 PRINT #1,GS$;XOFF;"";YOFF+YDIV*I;"";XOFF+XDIV*106;"";YOFF+YDIV*I;"IG";
14150 NEXT I
14160 '
14170 ' vertical lines
14180 FOR I=0 TO 106
14190 IF I MOD 10=0 THEN PRINT #1,GS$;"1;0lpG"; ELSE PRINT #1,GS$;"1;1lpG";
14200 PRINT #1,GS$;XOFF+XDIV*I;"";YOFF;"";XOFF+XDIV*I;"";YOFF+YDIV*66;"IG";
14210 NEXT I
14220 '
14230 ' draw CH 1 to 8
14240 PRINT #1,GS$;"0;0lpG"; ' normal line
14250 '
14260 DEF SEG=SEGPtr(2)
14270 S2=287:CALL S2
14280 '
14290 PRINT #1,GS$;"10wcf"; GS$;"9.6hpF"; GS$;"0;-25pP"; ' CHR size normal
14300 PRINT #1,CHR$(&HC);
14310 PRINT #1,GS$;"rhE"; ' hard reset
14320 CLOSE
14330 CLS 3
14340 LOCATE ,,1
14350 CONSOLE,,1
14360 GOTO *START

```