

# EXHIBIT F



04860.P1359

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor Application of:

Mark A. Dellabona, et al.

Examiner:

Serial No.: 08/381,471

Art Unit: 2609

Filed: January 31, 1995

For: METHOD AND AN APPARATUS FOR CONDUCTING A TOUCH-SENSITIVE CURSOR- CONTROLLING INPUT DEVICE TO GENERATE BUTTON VALVES SIMULATING THE BUTTON OF A MECHANICAL BUTTON SWITCH

2009  
Hej  
IBS  
M. B...  
10/15/96

INFORMATION DISCLOSURE STATEMENT

Commissioner of Patents and Trademarks Washington, D.C. 20231

Sir:

Pursuant to 37 C.F.R. 1.97, Applicant hereby provides a listing on PTO Form 1449 of references and encloses a copy of the references cited on the above referenced form. Pursuant to C.F.R. §1.97, this information disclosure statement is being submitted under 37 C.F.R. §1.97(b).

While this statement is being submitted as a means to comply with 37 CFR 1.97, it is not to be construed as an admission that the material identified above represents prior art to Applicant's invention.

Very truly yours,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Date: 7/5, 1996

J. Scheller  
James C. Scheller, Jr.  
Reg. No. 31, 195

12400 Wilshire Blvd.  
Seventh Floor  
Los Angeles, CA 90025  
(408) 720-8598

I hereby certify that this correspondence is being deposited with the United States Postal Service as first-class mail with sufficient postage in an envelope addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231

on 7/8/96  
Date of Deposit

Cheri Clark  
Name of Person Mailing Correspondence  
Cheri Clark  
Signature  
7/8/96  
Date

RECEIVED  
JUL 12 1996  
COMM-F



Form PTO-1449 (Rev. 8-83)	U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE	ATTY. DOCKET NO. 04860.P1359	SERIAL NO. 08/381,471
<b>INFORMATION DISCLOSURE CITATION</b> (Use several sheets if necessary)		APPLICANT Dellabona, et al.	
		FILING DATE 1/31/95	GROUP 2415 2609

U.S. PATENT DOCUMENTS

*EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

*EXAMINER INITIAL	DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION	
						YES	NO
JS	0490001	12/14/90	EPO	G06F	3/033		x
JS	2544103	10/12/84	FR	G06F	3/033	x	

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

JS	International Search Report, PCT/US96/00116, 7/6/96.
JS	Hamman, et al., "NOD AT YOUR COMPUTER: SWITCHLESS SELECTION TECHNIQUES USING A HEADPHONE DEVICE", Proceedings of the Annual Conference of the Engineering in Medicine

EXAMINER <i>John Surace</i>	DATE CONSIDERED <i>6/16/98</i>
--------------------------------	-----------------------------------

\*EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.



EUROPEAN PATENT APPLICATION

Application number: 90313657.0  
 Date of filing: 14.12.90

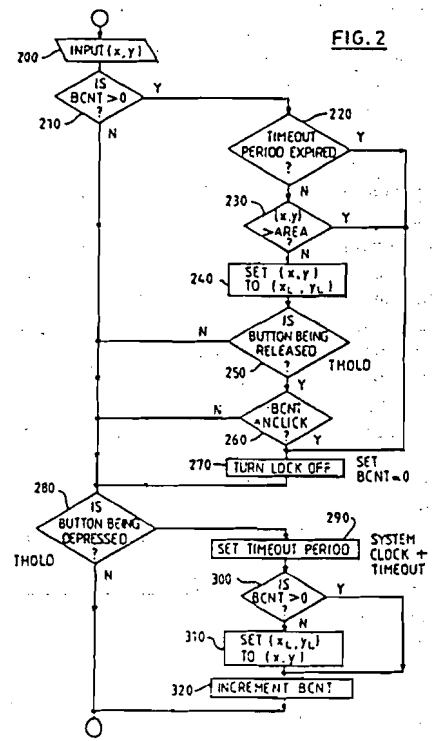
Int. Cl.<sup>5</sup>: G06F 3/033, G06F 3/023

Date of publication of application:  
 17.06.92 Bulletin 92/25  
 Designated Contracting States:  
 DE FR GB  
 Applicant: International Business Machines Corporation  
 Old Orchard Road  
 Armonk, N.Y. 10504(US)  
 Inventor: Beardall, Gavin David

1 Stable Cottages, Tichborne Down Road  
 Ailesford, Hampshire, SO24 9PA(GB)  
 Inventor: Calder, Gary James  
 2 Adamson Close  
 Eastleigh, Hampshire, SO5 1ND(GB)  
 Representative: Burt, Roger James, Dr.  
 IBM United Kingdom Limited Intellectual  
 Property Department Hursley Park  
 Winchester Hampshire SO21 2JN(GB)

Coordinate processor for a computer system having a pointing device.

A coordinate processor for a computer system having an absolute position pointing device (10) such as a touch sensitive display screen comprises stimulus detection means (200) for detecting a tactile stimulus of an absolute position pointing device (10) directed to a point within a data display area (610) of a computer system. The processor further comprises coordinate locking means (210-320) for locking a current cursor position to the point within the display area corresponding to the tactile stimulus in response to said stimulus exceeding a predetermined threshold value. The processor permits the computer system to distinguish a stimulus of the pointing device (10) for repositioning the cursor within the data display area (610) from a stimulus of the pointing device (10) for issuing a button click command to the computer system. The processor may be embodied in an electronic logic circuit within a pointing device adapter portion of the computer system. Equally, the coordinate processor may be in the form of a central processing unit operating under the control of a computer program.



EP 0 490 001 A1

The present invention generally relates to a coordinate processor for a computer system having a pointing device such as a touch sensitive display screen.

Many widely available computer systems such as the IBM PS/2 Model 70 (IBM and PS/2 are trademarks of IBM Corporation) are capable of receiving and processing data generated by a pointing device such as a mouse, tracker ball or touch sensitive display screen. The pointing device enables a user of the computer system to move, with a simple hand movement, a cursor between points within a data display area of a visual display unit.

A relative displacement pointing device such as a mouse or tracker ball provides the computer system with a vector which identifies the location within the data display area to which the cursor is to be moved relative to the current location of the cursor. The vector is generated by manipulating of the pointing device to achieve a desired cursor movement.

An absolute position pointing device such as a touch sensitive display screen provides the computer system with two dimensional coordinates identifying a point within the data display area to which the cursor is to be moved. In a touch sensitive display screen, the coordinates are generated by touching the screen at the point to which the cursor is to be moved.

In general, relative displacement pointing devices are also provided with at least one manually operable push button. The button can be operated by the user in a number of different modes and the computer system can be configured to respond differently to each mode of operation of the button. For example, the computer system may be configured to manipulate a window of displayed data within the data display area when the button is depressed as the pointing device is moved. Equally, the computer system may be programmed to perform another task when the cursor is placed on an icon representing the task within the display area and the button is depressed and released in rapid succession or "clicked". The computer system may also be configured to perform yet another task when the cursor is placed on an icon and the button is clicked twice or "double-clicked".

Touch screens are not generally provided with a manually operable push button. A button click command may however be issued via a touch screen by applying an corresponding sequence of touch stimuli to the touch screen within a predetermined time period. However, it will be appreciated that if each stimulus in the sequence is applied to a different point within a target area of the screen, then the computer system may fail to distinguish the button click command from a request to move the cursor from one point to another. In general

therefore, absolute position pointing devices have been thought of as unsuitable for issuing button click commands. Many commercially available application software packages have therefore been written with relative displacement pointing devices in mind. Such packages can therefore be wholly, or at least partially, incompatible with computer systems having absolute position pointing devices.

An aim of the present invention is therefore to provide a coordinate processor which enables a computer system comprising an absolute position pointing device to operate in the same manner as a computer system having a relative displacement pointing device.

In accordance with the present invention, there is now provided a coordinate processor comprising: stimulus detection means for detecting a tactile stimulus of an absolute position pointing device directed to a point within a data display area of a computer system; characterised in that the processor further comprises:

coordinate locking means for locking a current cursor position to the point within the display area corresponding to the tactile stimulus in response to said stimulus exceeding a predetermined threshold value.

This advantageously enables the computer system to distinguish between cursor movement commands and button click commands issued via an absolute position pointing device without requiring a separate, manually operable push button. An operator of such a computer system can therefore fully exploit application software packages designed for operation with relative displacement pointing device without any perceptible degradation in performance of the computer system. Specifically, the coordinate processor enables the computer system to lock the position of the cursor onto a particular coordinate when a stimulus which may signal a button click command is detected. If however, a button click command is not subsequently detected, the cursor is automatically unlocked.

Preferably, the coordinate locking means further comprises first reset means for releasing the cursor for movement within the data display area upon detection of a predetermined button click command.

The coordinate locking means of a preferred embodiment of the present invention further comprises second reset means for releasing the cursor for movement within the data display area upon expiry of a predetermined timeout period.

In addition, the coordinate locking means preferably further comprises third reset means for releasing the cursor for movement within the data display area upon detection of a subsequent tactile stimulus of the pointing device directed to a point outside a predetermined subarea of the data dis-

play area. Preferably, the subarea is predetermined by the computer system to be commensurate in size with an graphical icon generated within the display area by the computer system.

Viewing the present invention from a second aspect, there is now provided a coordinate processor comprising: stimulus detection means for detecting a stimulus applied to an absolute position pointing device and directed to a point within a data display area defined by a computer system; characterised in that the processor further comprises: command distinguishing means responsive to the stimulus detection means to distinguish a stimulus of the pointing device for repositioning a cursor within the data display area from a stimulus of the pointing device for issuing a button click command to the computer system.

Preferably, the command distinguishing means can be manually preset to identify either a tactile stimulus of the pointing device for issuing a single button click command to the computer system, or a tactile stimulus of the pointing device for issuing a multiple button click command to the computer system.

In a preferred embodiment of the present invention to be described later, there is provided a coordinate processor comprising: first receiver means for receiving from an absolute position pointing device an input two dimensional coordinate data value corresponding to a point within a data display area of a computer system and generated by the pointing device in response to a stimulus manually applied to the pointing device; second receiver means for receiving from the pointing device a force data value corresponding to the input coordinate value and generated by the pointing device in response to the stimulus; characterised in that the processor further comprises: coordinate locking means for setting a lock coordinate data value to the input coordinate data value in response to the force data value exceeding a predetermined threshold value; coordinate setting means for setting one or more further input coordinate force data values to the lock coordinate data value in response to any one of the further input coordinate data values falling within a predetermined range of coordinate data values during a predetermined time period; first reset means for resetting the lock coordinate data value in response to at least one discrete stimulus of the pointing device generating a force data value greater than the predetermined threshold value.

A preferred embodiment of the present invention will now be described, by way of example only with reference to the accompanying drawings in which:

Figure 1 is a block diagram of a computer system comprising an absolute position pointing

device in the form of a touch-sensitive visual display screen.

Figure 2 is a block diagram of a coordinate processor of the present invention in the form of a flow chart.

Figure 3 is a front view of a touch sensitive display screen displaying an icon within a data display area.

Figure 4 is a waveform diagram corresponding to tactile stimuli representative of a double click command.

Figure 5 is a waveform diagram corresponding to a tactile stimulus representative of a single click command.

Figure 1 illustrates an example of a computer system for processing input data from an absolute position pointing device. The system comprises a central processing unit (CPU) 20 for executing programmed instructions involving the input data. A bus architecture 30 communicates data between the CPU and other components of the computer system. A read only memory (ROM) 40 provides secure data storage. A random access system memory 50 provides temporary data storage. Data communication with other computer systems (not shown) is provided by a communications (COMM) adapter 60. An input/output (I/O) adapter 70 permits data communication between the bus architecture and a peripheral device such as a hard disk file 80. A visual output from the computer system in the form of a data display area is generated on a display device 110 by a display adapter 120. A user can operate the computer system using a keyboard 90 linked to the bus architecture via a keyboard adapter 100. By way of alternative to keyboard 90, an absolute position pointing device in the form of a touch sensitive display screen 10 is superimposed on display device 110. The touch screen 10 is responsive to a touch stimulus 130 applied by the user to issue a command to the computer system. The command may instruct the computer system to move a cursor between points within the data display area. Alternatively, the touch screen may be employed to issue a button click command instructing the computer system, to an operation corresponding to the current position of the cursor in the display area.

The touch screen 10 is resolved by digitising circuitry (not shown) in a pointing device adapter 140 into a two dimensional array of discrete coordinate points. A touch stimulus applied to any one of the coordinate points is detected by a sensor array (not shown) in the touch screen 10. The sensor array generates an analog signal proportional to the force imparted to the touch screen by the stimulus. The signal is digitised by a sampling analogue to digital convertor (ADC) circuit (not shown) in the touch screen 10 to produce an input data value. The

input data value, together with the coordinates to which it relates, are transmitted from the touch screen to the pointing device adapter 140. The input data value corresponding to each set of coordinates is typically refreshed by the ADC circuit sixty times a second. The pointing device adapter 140 connected to the bus architecture 30 passes each set of coordinates and the corresponding input data value to the bus architecture 30.

Referring now to Figure 2, a coordinate processor of the present invention distinguishes stimuli applied to the touch screen 10 to issue button click commands from stimuli to move the cursor within the display area. It will be appreciated that the coordinate processor of the present invention may be embodied in a hardwired electronic logic circuit within the pointing device adapter 140 or the touch screen 10. However, it will also be appreciated that, in other preferred embodiments of the present invention, the coordinate processor may be in the form of a processing unit such as CPU 20 operating under the control of a computer program.

The coordinate processor comprises an input stage 200. The input stage 200 sequentially reads the sampled input data value corresponding to each set of coordinates (x,y) of the touch screen 10 in turn. The coordinate processor increments a running total of button clicks BCNT each time a button click is detected. Initially BCNT is zero.

A count detect stage 210 checks BCNT for each input data value (x,y)<sub>n</sub> received from input stage 200. If BCNT corresponding to coordinates (x,y) is zero, then a depress detect stage 280 determines whether or not the corresponding input data value has increased above a predetermined button threshold value, THOLD.

If no such increase is detected, coordinates (x,y) are passed from coordinate processor to the computer system to control the positioning of the cursor within the display area. The next input data value is then received by input stage 200.

If, however, the input data value has increased over and above THOLD, then a timer stage 290 sets a timeout period, TIMEOUT which is equal to a current system clock value plus a predetermined value. In a preferred embodiment of the present invention, the timeout period can be manually adjusted about a nominal preset centre value of 500ms. Another count detect stage 300 then determines again whether or not BCNT is zero.

If BCNT is zero, then a coordinate locking stage 310 sets a pair of lock coordinates (xL,yL) to coordinates (x,y).

If BCNT is not zero, the lock coordinates (xL,yL) retain their existing values. In either case, coordinates (x,y) are then passed to the computer system to control cursor positioning. A counter 320 then increments BCNT and the next input data

value (x',y')<sub>n</sub> corresponding to coordinates (x',y') is received by input stage 200.

If count detect stage 210 determines that BCNT is greater than zero for input data value (x',y')<sub>n</sub>, timer stage 220 indicates whether or not the timeout period has been exceeded.

If the timeout period has been exceeded, then a reset stage 270 resets BCNT to zero before (x',y')<sub>n</sub> is passed to depress detect stage 280.

If the timeout period has not been exceeded, then a location check stage 230 determines whether or not coordinates (x',y') are outside a predetermined coordinate locking area, AREA, of the data display area. If coordinates (x',y') are outside AREA, then reset stage 270 resets BCNT before the (x',y')<sub>n</sub> is passed to depress detect stage 280.

If coordinates (x',y') are within AREA, then a lock stage 240 replaces coordinates (x',y') corresponding to input data value (x',y')<sub>n</sub> with the lock coordinates (xL,yL). Release detect stage 250 then detects whether or not input data value (x',y')<sub>n</sub> has decreased to below THOLD. If no such decrease is detected, then input data value (x',y')<sub>n</sub>, now corresponding to lock coordinates (xL,yL), is passed to depress detect stage 280. The cursor position is now locked to the lock coordinates (xL,yL).

If input data value (x',y')<sub>n</sub> has decreased over and below THOLD, then count detect stage 260 determines whether or not BCNT is equal to a predetermined click value, NCLICK.

In a particularly preferred embodiment of the present invention, NCLICK can be manually selected by the operator to detect different button click commands. For example, setting NCLICK to two configures the coordinate processor to detect both double and single click commands. Alternatively, setting NCLICK to one configures the coordinate processor to detect only single click commands.

If BCNT is equal to NCLICK, the button click command has been detected. BCNT is therefore reset to zero and input data value (x',y')<sub>n</sub> is passed to depress detect stage 280. The next input data value, (x'',y'')<sub>n</sub> corresponding to coordinates (x'',y'')<sub>n</sub>, is then received by input stage 200.

In a preferred embodiment of the present invention, predetermined values TIMEOUT, AREA, THOLD, and NCLICK are supplied to the coordinate processor by CPU 20 under the control of an application software program. In particular, with reference to Figure 3, AREA is preferably selected to represent an area 600 of size and location commensurate with an icon 620 representing graphically, within the data display area 610, a push button or the like. Preferably, the computer system is configured by the software so that the operator can select a particular program option by issuing a

click command via touch screen 10 at the position of the icon within the display area.

Referring now to Figure 4, a double click command can be represented in the form of a curve 400 of tactile force applied to touch screen 10 with respect to time. Button threshold THOLD is represented by reference line 420. Initially, BCNT is set to zero, NCLICK is set to 2, and TIMEOUT, AREA, and THOLD are set to appropriate values by the application software.

Initial tactile contact with touch screen 10 is made at time t0 where curve 400 is coincident with reference line 410.

At time t1, input data value (x,y)1 at coordinates (x,y) is lower than THOLD. However, at time t2, input data value (x,y)2 at coordinates (x,y) is greater than than THOLD. Depress detect stage 280 therefore indicates that the button is being depressed, and the timeout period is initialised by timer stage 290. Lock coordinates (xL,yL) are now set to coordinates (x,y). BCNT is incremented to indicate that the position of the cursor within the display area are now locked to the lock coordinates (xL,yL). Therefore, if the next force data values (x',y')3 at time t3 and (x'',y'')4 at time t4, correspond to coordinates (x',y') and (x'',y'') within the confines of AREA, then coordinates (x',y') and (x'',y'') are both replaced by lock coordinates (xL,yL).

It will be appreciated that if either (x',y') or (x'',y'') fall outside AREA then BCNT would reset to zero thereby unlocking the cursor position.

At time t5, input data value (x,y)5 is just greater than THOLD. However, at time t6, input data value (x,y)6 is lower than THOLD. Release detect stage 260 therefore indicates that the button is being released. However, BCNT does not equal NCLICK. BCNT is therefore not reset to zero. Therefore coordinates (x',y') corresponding to input data value (x',y')7 at time t7 are also replaced by lock coordinates (xL,yL). The cursor position is therefore still locked.

At time t8, input data value (x',y')8 is greater than THOLD. The timeout period is therefore initialised again by time stage 290. BCNT is incremented to indicate that a second button click has been detected. Lock coordinates (xL,yL) remain set to coordinates (x,y) originally corresponding to input data value (x,y)2. The cursor therefore remains locked to (x,y) within the display area.

At time t9, input data value (x,y)9 is lower than THOLD. Therefore release detect stage 250 indicates that the button is being released. BCNT is now equal to NCLICK indicating that a double click command has been detected. BCNT is now reset to zero to unlock the cursor position. The next touch stimulus producing an input data value greater than THOLD will therefore refresh lock coordinates

(xL,yL).

It will be appreciated that BCNT will also be reset to zero if it is maintained at a value greater than zero for a period greater than the timeout period. Similarly, BCNT will be reset to zero if any input stimulus is applied at a coordinate outside AREA.

A coordinate processor of the present invention therefore permits the operator to issue a double click command to a computer system via a touch sensitive display screen by locking the cursor position to a position within the display area at which an applied touch stimulus is of a magnitude exceeding a threshold value. The cursor position is unlocked when the prescribed number of clicks identifying the command is detected. Alternatively, the cursor position is unlocked if a subsequent stimulus is applied to the touch screen outside a predetermined area of the touch screen. Furthermore, the cursor position is also unlocked if the delivery of the command extends beyond a predetermined timeout period. It will therefore be appreciated that the coordinate processor of the present invention provides the operator of the computer system with freedom at all times to move the cursor to any point within the display area. Simultaneously however, the coordinate processor of the present invention enables the operator to issue via the touch screen a button click command to the computer system which is independent of any cursor movement command.

Referring now to Figure 5, a single click command can be represented in the form of a curve 500 of tactile force applied to the touch screen 10 with respect to time. Button threshold THOLD is represented by reference line 420. Initially, BCNT is set to zero and NCLICK is set to 1. TIMEOUT, AREA, and THOLD are set to appropriate values by an application software program.

Initial tactile contact is made with the touch screen 10 at time t0 where curve 500 is coincident with reference line 410.

At time t1, input data value (x,y)1 corresponding to coordinates (x,y) is lower than THOLD.

At time t2 however, input data value (x,y)2 corresponding to coordinates (x,y) is greater than THOLD. Depress detect stage 280 therefore indicates that the button is being depressed and the timeout period is initialised by timer stage 290. Lock coordinates (xL,yL) are now set to coordinates (x,y) and BCNT is incremented. The cursor position is now locked to coordinates (x,y).

If coordinates (x',y') and (x'',y''), corresponding to force data values (x',y')3 at time t3 and (x'',y'')4 at time t4, are within the confines of AREA, then (x',y') and (x'',y'') are both replaced by lock coordinates (xL,yL). The cursor position is therefore locked at coordinates (x,y).



If either (x',y') or (x'',y'') are outside AREA then BCNT will be reset to zero and the next input data value to exceed THOLD will refresh lock coordinates (xL,yL) and relock the cursor position. values.

At time t5, input data value (x,y)5 is lower than THOLD. Therefore, release detect stage 260 indicates that the button is being released. BCNT now equals NCLICK indicating that the single click command has been detected. BCNT is now reset to zero. Therefore, the next input data value to exceed THOLD will refresh lock coordinates (xL,yL)

It will now be appreciated that a coordinate processor of the present invention also enables the operator of the computer system to issue a single click command to the a computer system via a touch sensitive display screen. In addition however, a coordinate processor of the present invention also enables the operator to issue cursor movement command to the computer system through the touch screen independently of button click commands.

Specifically, the coordinate processor of the present invention locks the cursor position to a point on the touch screen at which an applied touch stimulus has a magnitude exceeding a threshold value. The cursor position is unlocked when the button click command is detected, or if a subsequent stimulus is outside a predetermined area of the touch screen. The cursor position is also unlocked if the delivery of the command extends beyond a predetermined timeout period. It will therefore be appreciated that the cursor may be freely moved to any point within the display area depending on whether or not the operator wishes to complete delivery of the button click command.

Curve 510 illustrates a continuous tactile stimulus of the touch screen applied initially at time t0. BCNT is initially set to zero.

At time t1, input data value (x,y)1 corresponding to coordinates (x,y) is lower than THOLD. However, at t2, input data value (x,y)2 corresponding to coordinates (x,y) is greater than THOLD. Therefore, depress detect stage 280 indicates that the button is being depressed and the timeout period is initialised.

Lock coordinates (xL,yL) are now set to coordinates (x,y) and BCNT is incremented. The cursor position is now locked to (x,y).

At time t5, input data value (x,y)5 is not below THOLD. Therefore, if no stimulus has been applied to the touch screen outside AREA at time t5, BCNT is not reset. The cursor position remains locked to coordinates (x,y).

At time t5' however, the timeout period expires. BCNT is therefore reset to zero. The cursor position is therefore unlocked. The next touch stimulus

exceeding THOLD will thus refresh lock coordinates (xL,yL).

A coordinate processor for a computer system having touch sensitive display screen has now been described by way of example of the present invention. It will however be appreciated that the present invention is equally applicable to other absolute position pointing devices such as, for example, tablets.

#### Claims

1. A coordinate processor comprising:
  - stimulus detection means (200) for detecting a tactile stimulus of an absolute position pointing device (10) directed to a point within a data display area (610) of a computer system;
    - characterised in that the processor further comprises:
      - coordinate locking means (210-320) for locking a current cursor position to the point within the display area corresponding to the tactile stimulus in response to said stimulus exceeding a predetermined threshold value.
2. A processor as claimed in claim 1 wherein the coordinate locking means (210-320) further comprises first reset means (260,270) for releasing the cursor for movement within the data display area (610) upon detection of a predetermined button click command.
3. A processor as claimed in claim 1 or claim 2 wherein the coordinate locking means (210-320) further comprises second reset means (290,220,270)
  - for releasing the cursor for movement within the data display area (610) upon expiry of a predetermined timeout period.
4. A processor as claimed in any preceding claim wherein the coordinate locking means (210-320) further comprises third reset means (230,270) for releasing the cursor for movement within the data display area (610) upon detection of a subsequent tactile stimulus of the pointing device (10) directed to a point outside a predetermined subarea (600) of the data display area (610).
5. A processor as claimed in claim 4 wherein the subarea (600) is predetermined by the computer system to be commensurate in size with an graphical icon (620) generated within the display area (610) by the computer system.

- 6. A processor as claimed in any preceding claim, wherein the coordinate locking means (210-320) can be manually preset to identify a tactile stimulus of the pointing device (10) for issuing a single button click command to the computer system. 5
  
- 7. A coordinate processor as claimed in claim 6 wherein the coordinate locking means (210-320) can be manually preset to identify a tactile stimulus of the pointing device (10) for issue a multiple click command to the computer system. 10
  
- 8. A processor as claimed in any preceding claim wherein the coordinate locking means (210-320) is responsive to an digital input value corresponding to the force of the tactile stimulus. 15  
20
  
- 9. A processor as claimed in any preceding claim wherein the absolute pointing device (10) is a touch sensitive display screen (10). 25
  
- 10. A computer system comprising a processor as claimed in any preceding claim. 25

30

35

40

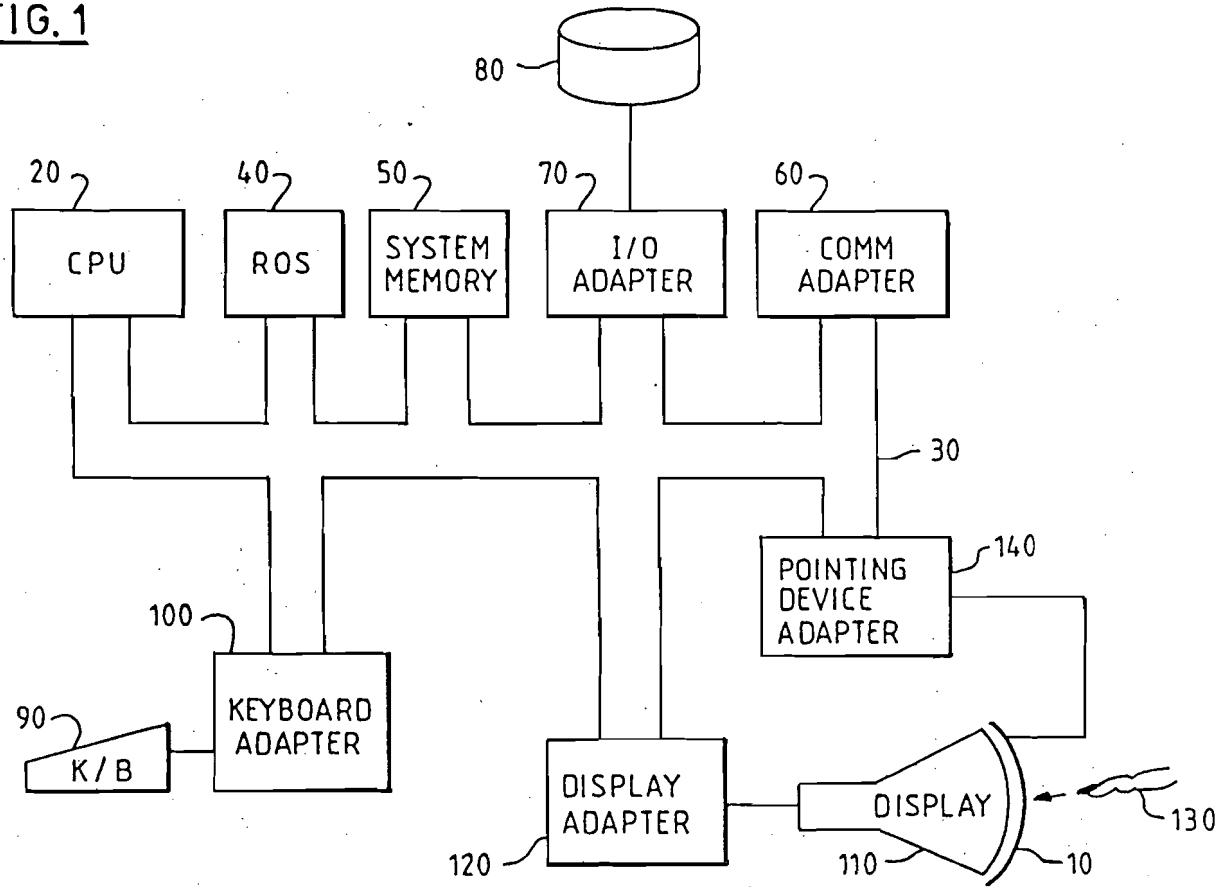
45

50

55

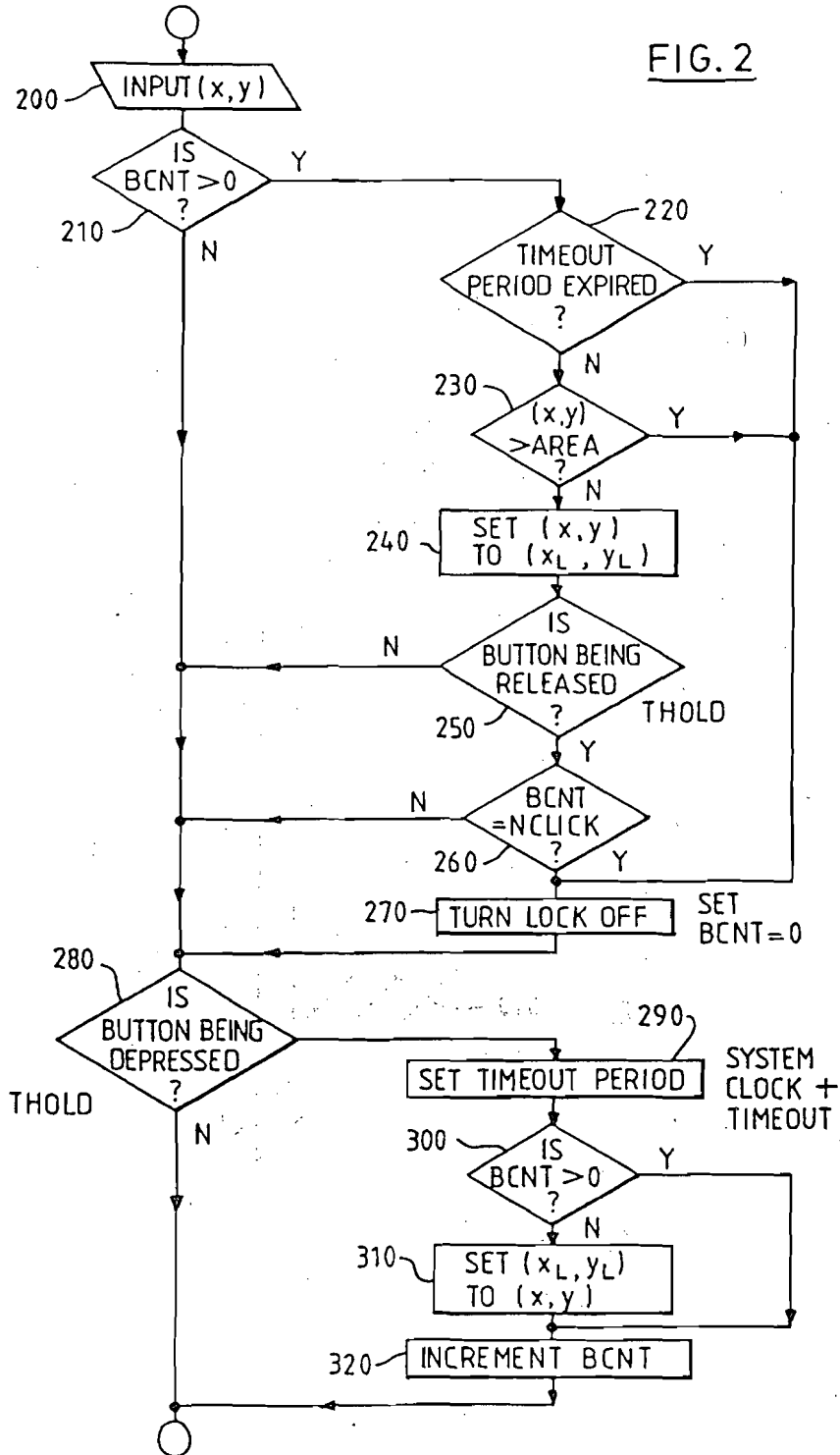
7

FIG. 1



EP 0 490 001 A1

FIG. 2



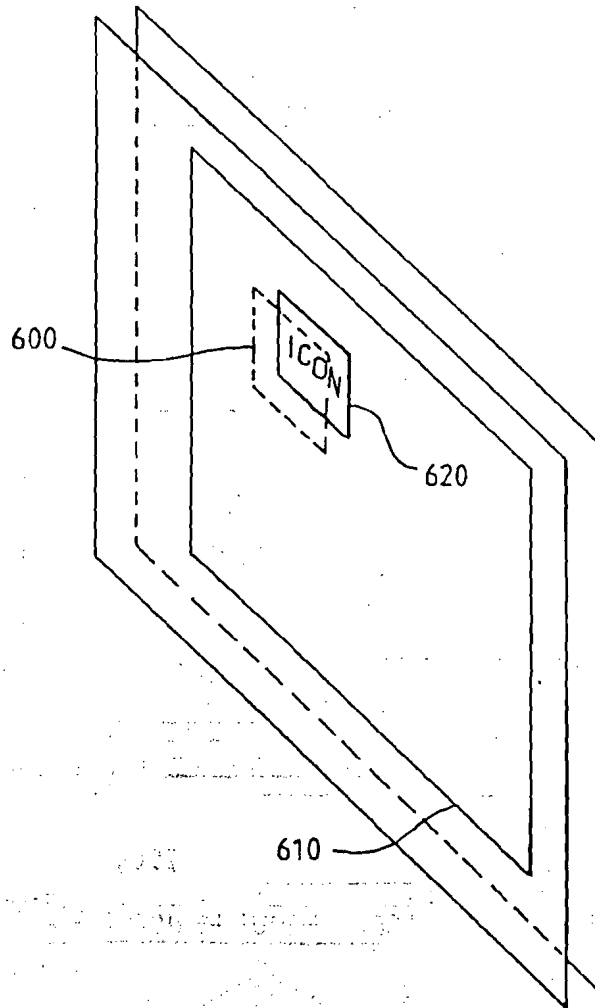


FIG. 3

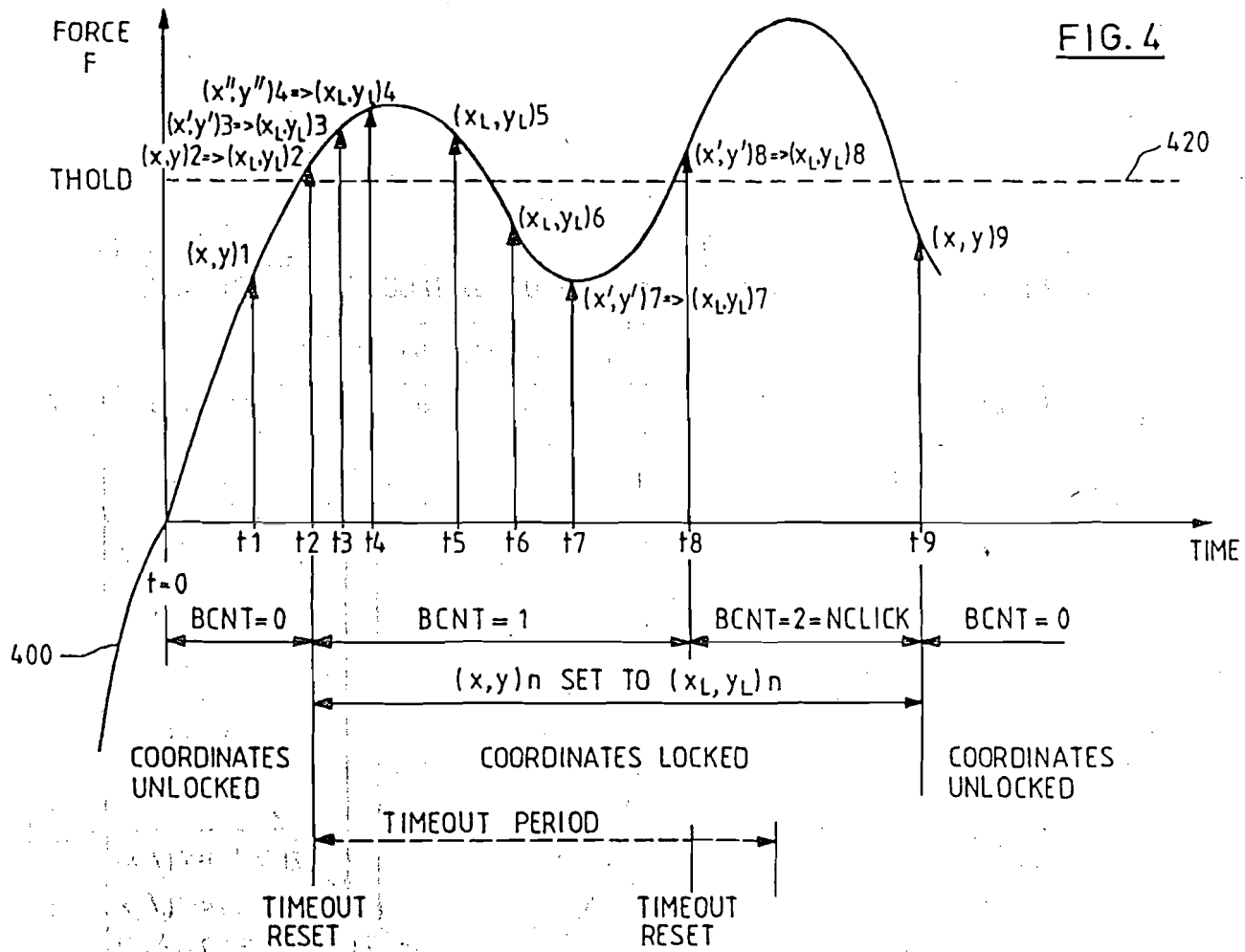


FIG. 4

EP 0 490 001 A1

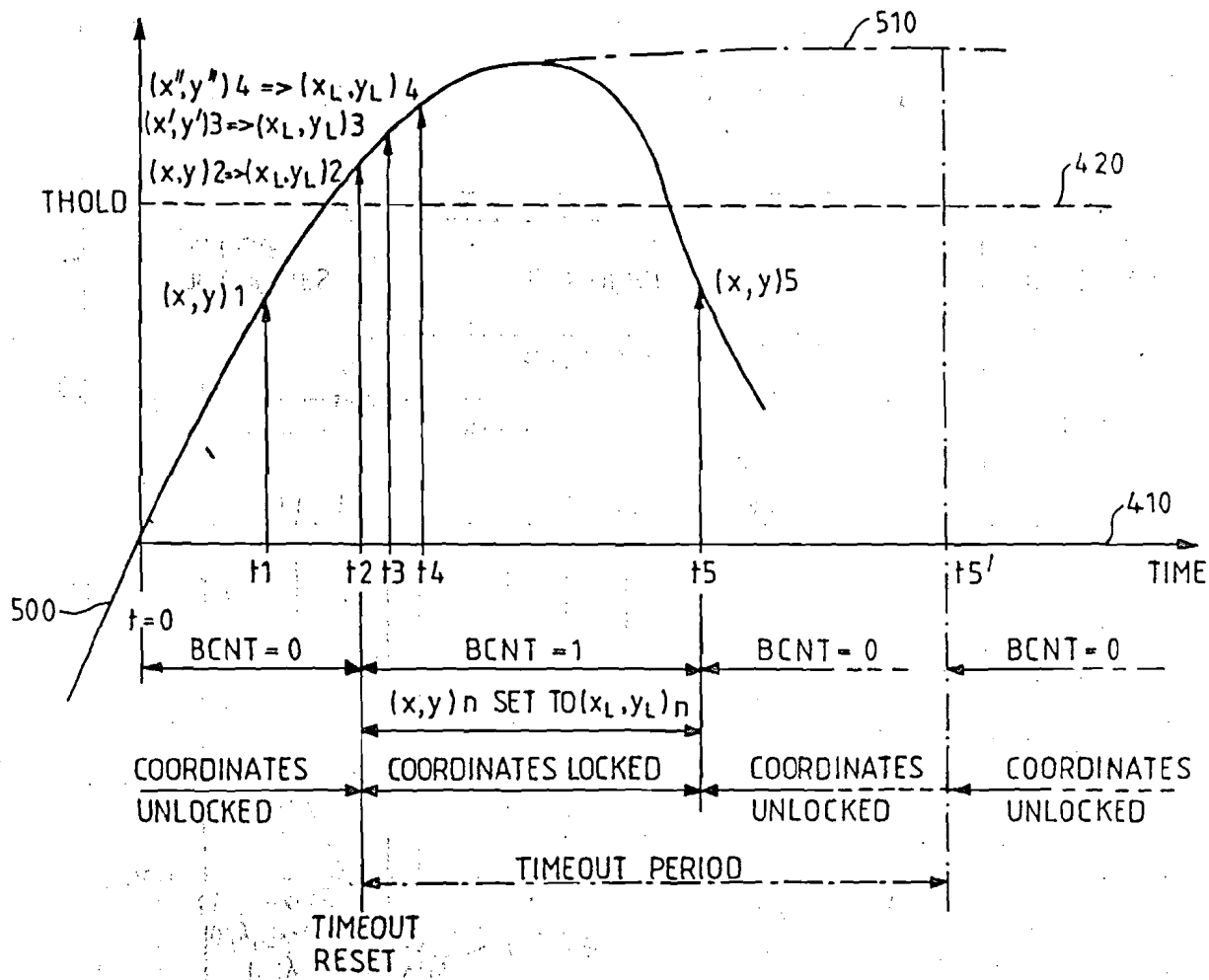


FIG. 5

EP 0 490 001 A1



European Patent  
Office

EUROPEAN SEARCH REPORT

Application Number

EP 90 31 3657

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
X	US-A-4 914 624 (D.I. DUNTHORN) * Figures 1,2; column 1, line 67 - column 3, line 44; column 5, lines 1-54; column 7, lines 23-42; claims 1,8,10,18 *	1-4, 6, 8 -10	G 06 F 3/033 G 06 F 3/023
A	GB-A-2 152 250 (MARCONI INSTRUMENTS LTD) * Column 1, lines 28-47; claims 1,4 *	1-4	
A	EP-A-0 156 593 (AMP INC.) * Page 4, lines 12-26 *	1	
			TECHNICAL FIELDS SEARCHED (Int. Cl.5)
			G 06 F 3/033 G 06 F 3/023
The present search report has been drawn up for all claims			
Place of search THE HAGUE		Date of completion of the search 19-07-1991	Examiner ALONSO Y GOICOLEA L.
CATEGORY OF CITED DOCUMENTS		T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document	
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document			

EPD FORM 1903 (01/82) (P0401)





FR 2544103 (1) FR 2544103 (1)

G06F3/033D2-G06F1/16P2-G06F3/023A4-  
G06K11/06-G06K11/12-

- 5- \*-

G06F3/033D2

(11) N° de publication :  
(à n'utiliser que pour les  
commandes de reproduction)

**2 544 103**

(21) N° d'enregistrement national :

**84 05390**

(51) Int Cl<sup>3</sup> : G 06 F 3/033.

(12)

## DEMANDE DE BREVET D'INVENTION

A1.

(22) Date de dépôt : 5 avril 1984.

(30) Priorité : US, 8 avril 1983, n° 483.230.

(43) Date de la mise à disposition du public de la  
demande : BOPI « Brevets » n° 41 du 12 octobre 1984.

(60) Références à d'autres documents nationaux appa-  
rentés :

(71) Demandeur(s) : GAVILAN COMPUTER CORPORATION.  
— US.

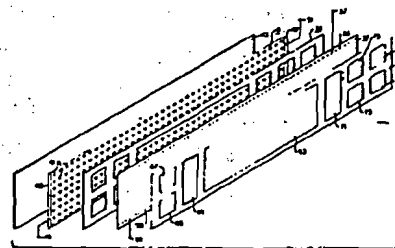
(72) Inventeur(s) : Gary J. Prosenko, Richard Tabor et Ri-  
chard B. Ravel.

(73) Titulaire(s) :

(74) Mandataire(s) : Regimbeau, Corre, Martin, Schrimpf,  
Warcoïn et Ahner.

(54) Dispositif d'entrée d'informations dans un ordinateur au moyen d'un panneau à contact.

(57) Ce panneau à contact 21 utilisable, en particulier, avec un ordinateur portable est monté dans un circuit qui permet de déterminer les coordonnées X-Y du point où l'utilisateur appuie le doigt. Le panneau est divisé en plusieurs zones de bouton-poussoir 65-79 et une zone plus grande 63 de commande de curseur. Lorsque le circuit détermine que le doigt est appuyé sur un bouton-poussoir, il transmet l'information correspondante à l'ordinateur. Lorsqu'il détermine que le doigt appuie sur la zone de commande de curseur, le circuit transmet à l'ordinateur des signaux de distance incrémentielle, fonction de la vitesse de déplacement du doigt, qui servent à déplacer un curseur sur la visu de l'ordinateur. Si l'utilisateur applique un coup bref à la surface de cette zone, le circuit le différencie d'un déplacement du doigt et transmet un signal spécial à l'ordinateur qui l'interprète, par exemple, comme un signal d'exécution.



FR 2 544 103 - A1

La présente invention se rapporte, d'une manière générale, à des dispositifs et procédés d'entrée d'informations dans un ordinateur et plus particulièrement à ceux qui utilisent une surface qui est sensible au contact du doigt d'un utilisateur pour entrer des signaux dans le système d'ordinateur afin de modifier son affichage et son fonctionnement.

La plupart des terminaux d'ordinateur utilisent un curseur ou élément de repérage qui est affiché sur un écran et que l'opérateur peut déplacer sur l'écran. Un curseur est le plus couramment utilisé pour indiquer un emplacement sur l'écran où l'opérateur désire effectuer une certaine opération. Par exemple, un terminal d'ordinateur fonctionnant en machine de traitement de textes peut remplacer ou effacer un mot ou une lettre que le curseur recouvre où qu'il désigne. L'opérateur déplace tout d'abord le curseur jusqu'à l'emplacement désiré sur l'écran d'affichage puis il donne une commande pour exécuter une opération quelconque à cet emplacement.

Il existe un certain nombre de dispositifs et techniques couramment utilisés pour entrer ces commandes de l'utilisateur dans un terminal d'ordinateur. Un clavier ordinaire est le plus courant au moins dans les applications au traitement de texte. L'emploi d'une "souris" a tendance à se généraliser. Une souris est un dispositif tenu à la main que l'on déplace sur une surface plate adjacente au terminal d'ordinateur pour provoquer le déplacement du curseur sur une distance et dans une direction proportionnelles au déplacement de la souris. Une souris porte également, en général, un ou plusieurs boutons pour transmettre des commandes à exécuter par le terminal d'ordinateur de sorte qu'un utilisateur peut indiquer qu'une action prédéterminée doit être exécutée à l'emplacement de l'écran d'affichage ou visu où le curseur a été positionné par la souris.

Des écrans d'affichage à contact sont également utilisés pour engendrer des signaux de commande et de données en vue de leur entrée dans un système d'ordinateur. D'autres dispositifs externes sont notamment les manches à balai, les moulinets, les boules roulantes et analogues. Ces disposi-

tifs sont souvent utilisés pour jouer avec l'ordinateur en commandant le mouvement d'un "curseur" qui se présente sous la forme d'un tank, d'une mitrailleuse ou analogue. Un bouton de commande de tir est également prévu, en général, avec le dispositif de commande pour provoquer l'exécution d'une action prédéterminée une fois que le "curseur" a été positionné à l'emplacement où le joueur désire qu'il se trouve.

Un autre dispositif de commande de curseur qui est de plus en plus généralement accepté est un panneau à contact ou à commande par effleurement, qui est prévu soit en tant que partie d'un terminal d'ordinateur adjacent à son clavier soit en tant que dispositif séparé utilisé à côté de lui et interconnecté avec le terminal. Un tel panneau est sensible au contact d'un doigt et il est connecté pour provoquer le déplacement de son curseur dans les deux directions X et Y correspondant au déplacement du doigt de l'opérateur sur la surface du panneau. Des panneaux à contact typiques fonctionnent à l'aide d'une surface électriquement conductrice qui a une résistance donnée par unité de longueur d'un bout à l'autre du panneau. La valeur de résistance transmise au terminal de l'ordinateur dépend de l'emplacement de la surface touché par l'utilisateur. D'autres panneaux à contact fonctionnent sur le principe de la magnétostriction et produisent un signal proportionnel à la distance de l'emplacement de la surface touché par l'utilisateur des bords du panneau. Les panneaux à contact actuellement utilisés produisent le mouvement désiré du curseur qui doit être ensuite suivi d'une autre action appropriée quelconque de l'utilisateur, qui doit, par exemple, appuyer sur une touche séparée pour provoquer l'exécution d'une fonction au nouvel emplacement du curseur sur l'écran d'affichage.

L'un des principaux buts de la présente invention est de réaliser un procédé et un dispositif de commande à panneau à contact perfectionnés qui sont plus faciles à utiliser, d'une construction plus simple et présentent, cependant, des caractéristiques d'exploitation perfectionnées.

Ces buts et d'autres buts de l'invention sont atteints

au moyen des divers aspects de l'invention suivant lesquels, sommairement, selon un aspect de l'invention, il est prévu un système et un procédé d'exploitation pour permettre au panneau à contact commandant un curseur bidimensionnel de remplir deux fonctions. Lorsqu'un utilisateur déplace le doigt sur la surface du panneau à contact, ceci provoque le déplacement normal du curseur mais lorsque le curseur est à l'emplacement désiré de l'écran, l'opérateur n'a pas besoin de chercher un autre dispositif d'entrée quelconque, tel qu'une touche séparée, pour provoquer l'exécution d'une fonction prédéterminée à cet emplacement de l'écran de la visu. L'opérateur, en utilisant le perfectionnement de la présente invention, n'a besoin que d'appliquer un petit coupe bref sur le panneau à contact et ceci est distingué par le système du panneau à contact comme une commande différente du contact plus long qui accompagne typiquement le déplacement d'un doigt sur le panneau pour provoquer le déplacement du curseur. Ceci permet à l'opérateur de conserver le regard fixé sur l'écran de la visu sans avoir à chercher un bouton de commande d'exécution séparé et ceci permet, en outre, de réaliser un système plus simple en supprimant la nécessité d'avoir à utiliser un bouton de commande d'exécution séparé.

Un autre perfectionnement apporté au système de panneau à contact selon la présente invention est le traitement des signaux provenant du panneau à contact avant qu'ils soient utilisés pour déplacer le curseur de façon que le déplacement erratique du doigt de l'utilisateur soit converti en un déplacement plus régulier du curseur sur l'écran de la visu et de façon que la distance de déplacement du curseur pour une distance donnée de déplacement du doigt sur le panneau soit rendue dépendante de la vitesse de déplacement du doigt. La régularisation est effectuée selon un mode de réalisation préféré, en faisant signaler par le système de panneau à contact au terminal d'ordinateur le déplacement incrémentiel du doigt dans les directions X et Y par unité de temps, la moyenne de plusieurs signaux de déplacement incrémentiel suc-

cessifs étant utilisée pour déplacer le curseur de façon ainsi à régulariser son mouvement. La caractéristique de changement d'échelle est fournie, selon un mode de réalisation préféré, en élevant au carré les signaux de distance de déplacement incrémentielle moyenne avant de les utiliser pour déplacer le curseur, provoquant, de ce fait, un plus grand déplacement du curseur pour un déplacement donné du doigt sur le panneau lorsque la vitesse d'un tel déplacement du doigt est plus élevée.

Un autre perfectionnement, selon un autre aspect de la présente invention, est la combinaison avec la zone du panneau à contact utilisée pour commander le déplacement du curseur dans les directions X-Y d'un certain nombre de fonctions discrètes de bouton-poussoir, l'actionnement de l'un quelconque d'un certain nombre de tels boutons-poussoirs étant transmis par le panneau à contact sur la même ligne de signalisation que le signal de déplacement X-Y du curseur et les signaux de commande d'exécution produits par des coups brefs. Les signaux des boutons-poussoirs discrets sont séparés des signaux X-Y dans le circuit de sortie commun du panneau à contact par le traitement ultérieur. Dans un mode de réalisation préféré, un unique panneau à contact est muni de zones séparées dans l'espace pour une série de boutons-poussoirs et d'une région X-Y. La signification d'un contact quelconque du doigt de l'utilisateur sur le panneau à contact est déterminée à partir de l'emplacement de ce contact, tel qu'exprimé par le signal de sortie commun du panneau à contact. Dans une forme spécifique d'un tel mode de réalisation, deux feuilles résistives sont normalement maintenues étroitement rapprochées et électriquement connectée pour détecter l'emplacement du contact du doigt d'un utilisateur lorsqu'il provoque l'entrée en contact de feuilles à un emplacement particulier.

D'autres buts, caractéristiques et avantages des divers aspects et modes de réalisation de la présente invention apparaîtront à la lecture de la description qui va suivre d'un mode de réalisation préféré de l'invention, description qui.

doit être considérée en combinaison avec les dessins annexés dans lesquels:

la Fig. 1 représente, d'une manière générale, un système d'ordinateur portatif dans lequel le système de panneau à contact perfectionné de la présente invention peut être avantageusement utilisé;

la Fig. 2 est un schéma-bloc général du système d'ordinateur de la Fig. 1;

la Fig. 3 représente une vue éclatée de l'assemblage mécanique du panneau à contact de l'ordinateur de la Fig. 1;

la Fig. 4 est un schéma du circuit électrique d'un système utilisant le panneau à contact de la Fig. 3;

la Fig. 5 est un diagramme des temps pour le circuit électronique de la Fig. 4;

la Fig. 6 représente la manipulation de données numériques par le circuit de la Fig. 4;

la Fig. 7 représente le format de signaux numériques particuliers du circuit de la Fig. 4;

la Fig. 8 est un organigramme qui représente le fonctionnement du circuit de la Fig. 4; et

la Fig. 9 est un diagramme d'états qui illustre le fonctionnement du logiciel d'ordinateur utilisé en combinaison avec le circuit de la Fig. 4 pour traiter les informations obtenues du panneau à contact.

On décrira la présente invention en se référant à son application dans un mode de réalisation d'un ordinateur portatif perfectionné, représenté d'une manière générale sur la Fig. 1. Bien que la présente invention présente une très grande utilité dans un terminal d'ordinateur quelconque, elle est particulièrement avantageuse dans un système portatif du fait du nombre réduit des éléments de circuit qui sont nécessaires pour exécuter un ensemble donné de fonctions désirées. Un carter 11 contient un clavier 13 sur le dessus de sa partie avant, ce clavier étant enfermé par un couvercle 15 lorsque l'appareil est transporté. Sur la Fig. 1, le couvercle 15 est représenté dans une position verticale et on voit qu'il contient un dispositif d'affichage ou visu 17

d'un type approprié. Le type de visu 17 qui est utilisé dans un tel ordinateur portatif est un écran d'affichage à cristaux liquides (ACL) du fait de sa faible consommation de courant. Typiquement, la visu est du type à pixels et est capable d'afficher plusieurs lignes, par exemple, huit  
5 lignes ou davantage, d'informations alphanumériques ou d'informations graphiques de dimensions semblables.

Le système est capable d'afficher un curseur 19, représenté sur la Fig. 1 sous la forme d'une flèche. Naturellement, on peut utiliser d'autres configurations de curseur,  
10 telles qu'un petit rectangle, un grand carré ou un cadre rectangulaire entourant une partie importante de l'écran et d'autres formes et dimensions variables en fonction de l'application. Dans tous les cas, le curseur peut se déplacer sous la commande de l'opérateur sur la surface de l'écran du dispositif d'affichage ou visu 17.  
15

Une feuille graphique 21 qui recouvre la partie mécanique du système de panneau à contact est disposée immédiatement adjacente au clavier 13. On décrira ci-après en se référant à la Fig. 3 les aspects mécaniques d'un ensemble  
20 de panneau à contact spécifique. L'orientation du panneau à contact, disposé immédiatement adjacent au bord supérieur du clavier 13 et à une plus faible inclinaison, le rend d'une observation et d'un accès très commodes pour l'utilisateur et fournit également un ensemble compact qui est indispensable pour un appareil portatif.  
25

Sur la Fig. 2 à laquelle on se référera, on a représenté un schéma-bloc général du système d'ordinateur 11. Un bus commun 23 du système, typique dans de tels systèmes, interconnecte le microprocesseur (UTC 8088) 25 avec les autres  
30 éléments fonctionnels principaux du système qui comprennent une mémoire morte (MM) 27, une mémoire à accès sélectif (MAS) 29, une unité à disque (Disque) 31 et un circuit 34 de commande de visu. En outre, le clavier 13 est connecté au  
35 bus du système par l'intermédiaire d'un système électronique approprié et un système 33 de panneau à contact est, de même, connecté au bus 23 dans ce système particulier. Le



système 33 de panneau à contact de la Fig. 2 comporte l'ensemble mécanique de panneau à contact représenté sur la Fig. 3 et son circuit électrique représenté sur la Fig. 4. En outre, divers circuits et dispositifs d'entrée/sortie (E/S) 5 35, tels qu'une imprimante, un modem téléphonique et des dispositifs périphériques similaires couramment utilisés, sont connectés au bus du système. Dans le système particulier décrit à titre d'exemple, le microprocesseur 25 est un microprocesseur Intel 8088.

10 On décrira maintenant, en se référant à la Fig. 3, un ensemble mécanique de panneau à contact particulier. On a représenté un type résistif de panneau à contact mais, naturellement, on peut obtenir des signaux similaires en réponse à un contact appliqué à une surface en utilisant d'autres mécanismes transducteurs, tels qu'un dispositif magnétostrictif qui fonctionne en mesurant le temps que met une impulsion pour se propager d'un point à un autre de la surface en tant qu'indication de l'emplacement du contact et analogue. Derrière la feuille de couverture protectrice 21 est disposée une première feuille résistive 37 suivie d'une feuille 20 d'espacement 39, d'une seconde feuille résistive 41 et d'un élément support 43 formant un substrat rigide. Chacune des feuilles résistives est caractérisée par le fait que sa surface qui fait face à l'autre a une conductivité qui a une 25 résistivité uniforme par unité de distance suivant sa surface.

Les feuilles 37 et 41 peuvent avoir l'une quelconque des diverses constructions spécifiques dont chacune fournit des caractéristiques électriques équivalentes. Elle peuvent être, par exemple, constituées par des feuilles de polyester 30 revêtues sur leurs surfaces en vis à vis d'une encre au carbone ou d'une matière appliquée sur elles par dépôt de vapeur, telle qu'un oxyde d'indium-étain. Suivant une variante, ces feuilles peuvent être des feuilles conductrices en matière élastomère qui sont imprégnées de matières conductrices 35 telles que du carbone. Dans tous les cas, la résistivité par unité de longueur est soigneusement réglée de façon à être uniforme sur toute la surface de chaque feuille et sa valeur

est choisie en fonction des besoins du circuit électrique particulier avec lequel fonctionne le panneau à contact.

Un potentiel électrique est appliqué à l'une des feuilles 37 et 41 entre ses bords opposés dans la direction X et un potentiel électrique est appliqué à l'autre feuille entre ses bords opposés dans la direction Y. Des bandes métalliques ou en une autre matière ayant une conductivité électrique élevée 45 et 47 sont placées sur la surface résistive de la feuille 41 le long de ses bords transversaux opposés pour provoquer l'application d'une tension appliquée à des bornes 49 et 51 dans la direction X de la feuille 41. De même, des bandes conductrices 53 et 55 sont placées sur les bords longitudinaux opposés de la feuille 37 provoquant, de ce fait, la production d'un gradient de potentiel entre ces bords, dans la direction Y, lorsqu'une tension est appliquée à des bornes 57 et 59.

Afin d'assurer que les feuilles 37 et 41 ne se touchent pas accidentellement, une grille de petites bosses en matière diélectrique est disposée par sérigraphie sur l'une des feuilles, représentée sur la Fig. 3 comme étant la feuille 41. Ces bosses sont disposées à environ 5 mm les unes des autres dans une configuration symétrique sur toute la surface de la feuille et elles ont une hauteur inférieure à 0,025 mm et un diamètre d'environ 0,1 mm. Cet espacement et ces dimensions permettent que les surfaces électriquement conductrices en vis à vis des feuilles 37 et 41 entrent en contact l'une avec l'autre lorsqu'on applique une pression du doigt, d'une force désirée, en un point quelconque de la surface de la feuille 37 par l'intermédiaire de la mince feuille indicatrice souple 21. La feuille 37 est, naturellement, elle-même flexible pour permettre un tel mouvement physique et un tel contact physique.

L'ensemble de panneau à contact de la Fig. 3 est utilisé pour engendrer un certain nombre de signaux indépendants au moyen de l'unique jeu commun des quatre bornes 49, 51, 57 et 59. La surface bidimensionnelle du panneau à contact est spatialement divisée en zones discrètes, telles que la zone 63 de commande de curseur à deux dimensions X-Y et une

série de zones de boutons ou touches discrètes 65, 67, 69, 71, 73, 75, 77 et 79. Il est possible de prévoir un nombre supérieur ou inférieur de zones discrètes selon l'application du système d'ordinateur et selon les dimensions disponibles de la surface de panneau à contact. Comme on l'expliquera ci-dessous en se référant à la Fig. 4, celle de ces zones sur laquelle l'utilisateur a posé le doigt est identifiée en déterminant les coordonnées X-Y du point de contact entre les surfaces résistives des feuilles 37 et 41.

Afin d'aider à la réalisation de cette discrimination, une mince feuille d'espacement non conductrice 39 est positionnée entre les feuilles électriquement résistives 37 et 41. Des ouvertures sont formées dans la feuille 39, ces ouvertures ayant les mêmes configurations que les indications portées par la surface de la feuille 21 pour permettre le repérage des différentes zones de bouton et de la zone de commande du curseur X-Y. La feuille d'espacement 39 n'est pas absolument indispensable mais elle facilite la distinction, par le système électronique que l'on décrira ci-après, des différentes zones discrètes.

Sur la Fig. 4 à laquelle on se référera maintenant, on a représenté un schéma électrique dans lequel le panneau à contact de la Figure 3 est représenté électriquement par des résistances tracées à l'intérieur d'un cadre 81 en traits interrompus. Une résistance 83 représente la résistance de contact produite lorsque les feuilles 37 et 41 sont poussées l'une contre l'autre à un point particulier par la pression du doigt d'un utilisateur. L'emplacement d'un tel contact est représenté schématiquement sur la Fig. 4 par une connexion électrique à des emplacements particuliers de la résistance de la feuille 41 dans la direction X et de la résistance de la feuille 37 dans la direction Y. L'emplacement de la connexion entre les deux résistances représentatives des surfaces 37 et 41 varie tant dans la direction X que dans la direction Y lorsque l'emplacement de la pression de contact appliquée à la surface bidimensionnelle du panneau à contact est changé.

Une tension d'alimentation +V est sélectivement appliquée aux surfaces des feuilles résistives 37 et 41 sous la commande de quatre transistors Q1, Q2, Q3 et Q4. Ces transistors fonctionnent en interrupteurs. Les transistors Q1 est connecté entre la borne 49 et le potentiel de la masse et il est maintenu dans un état conducteur ou non conducteur en fonction du niveau de tension d'un conducteur 85 qui est connecté par l'intermédiaire d'une résistance à la borne de base du transistor Q1. De même, le transistor Q2 est connecté entre la borne 59 et le potentiel de la masse tandis que sa base est connectée par l'intermédiaire d'une résistance à une ligne de commande 87. Le transistor Q3 est connecté entre la source de tension positive +V et la borne 51, tandis que sa base est connectée par l'intermédiaire d'une résistance à la ligne 87. Le transistor Q4 est connecté entre la source d'alimentation positive +V et la borne 57 tandis que sa base est connectée à un circuit 89 par l'intermédiaire d'une résistance série.

Les signaux de commande servant à mettre ces quatre transistors à l'état conducteur et à l'état non conducteur sont émis par une unité de traitement centrale (UTC) 91 sous la forme d'un signal de commande TPD sur une ligne de sortie 93 et d'un signal de CONTROLE DU PANNEAU sur une ligne 95. Dans le cas du système particulier qui est décrit, l'unité de traitement 91 est, de préférence, un microprocesseur Intel 8051 qui comporte une certaine quantité de mémoire morte (MM) et de mémoire à accès sélectif (MAS). Un signal de commande TPD à deux niveaux appliqué sur la ligne 93 est transmis par l'intermédiaire d'un inverseur 97 dont la sortie produit un signal de commande sur la ligne 87. Le signal de commande de la ligne 87 est également transmis par l'intermédiaire d'un inverseur 99 qui produit un signal de commande sur la ligne 89. Les inverseurs 97 à 99 sont, de préférence, constitués par des éléments de circuits intégrés à MOS complémentaires. Les transistors Q3 et Q4 sont du type PNP et les transistors Q1 et Q2 sont du type NPN. Le signal de CONTROLE DU PANNEAU à deux niveaux transmis sous la ligne 95

est appliqué à une entrée d'une porte NI 101 et à la base, par l'intermédiaire d'une résistance série, d'un transistor Q5. Une seconde entrée de la porte NI 101 est la ligne 87. Le transistor Q5 est utilement connecté entre la borne 59 et le potentiel de la masse par l'intermédiaire d'une résistance série et il est commandé dans ses états conducteur et non conducteur par un signal appliqué sur la ligne 95. La sortie de la porte NI 101 fournit un signal de sortie sur la ligne 85.

10 Comme on l'expliquera plus complètement ci-après, les coordonnées X et Y du point de contact d'un doigt sur les feuilles 37 et 41 sont déterminées au cours d'étapes chronologiquement séparées. Au cours d'une première période de temps, les transistors Q1 et Q2 sont rendus conducteurs pour  
15 appliquer une tension aux bornes de la résistance X 41, tandis que la résistance Y 37 sert de détecteur pour transmettre la tension qui existe à l'emplacement de la feuille résistive où le contact s'est produit. Ce signal est transmis par la borne 59 qui est connectée à l'entrée non inverseuse d'un  
20 amplificateur opérationnel 103 dont la sortie 105 est directement connectée à son entrée inverseuse. L'amplificateur 103 sert d'étage séparateur afin que la résistance 37 de l'axe des Y ne soit pas elle-même modifiée par la charge car, autrement, elle ne pourrait pas servir à indiquer la  
25 tension au point de sa connexion avec la résistance 41 de l'axe des X.

De même, la lecture de la position du point de contact suivant l'axe des Y est effectuée en rendant les transistors Q2 et Q3 conducteurs pour appliquer une tension aux bornes  
30 de la résistance 37. La résistance 41 de l'axe des X sert de mécanisme pour transmettre la tension au point de contact avec la résistance Y, cette transmission s'effectuant par l'intermédiaire de la borne 49 qui est connectée à l'entrée non inverseuse d'un second amplificateur opérationnel 107.  
35 La sortie 109 de l'amplificateur est connectée directement en retour à son entrée inverseuse.

Les tensions analogiques produites sur les lignes 105

et 109 sont proportionnelles respectivement à la coordonnée X et à la coordonnée Y de l'interconnexion entre les feuilles résistives X et Y, 41 et 37. Ces signaux sont appliqués l'un après l'autre à un convertisseur d'analogique en numérique (A/N) 111 par l'intermédiaire d'un multiplexeur 113 (commutateur) suivi d'un amplificateur séparateur et d'un circuit résistance-condensateur, comme représenté. Le multiplexeur 113 reçoit sur une ligne 115 provenant de l'unité de traitement 91 un signal de commande indiquant si le signal de la ligne 105 ou celui de la ligne 109 doit être transmis au convertisseur A/N 111. Un signal appliqué sur une ligne 117 qui provient également de l'unité de traitement 91 indique également au multiplexeur 113 le moment où il doit choisir l'un ou l'autre des signaux. Un signal de commande émanant de l'unité de traitement 91 et appliqué sur une ligne 119 est transmis au convertisseur A/N 111 pour déterminer le moment où il doit commencer sa conversion. De même, un autre signal appliqué sur une ligne 121 détermine le moment où la valeur de conversion du convertisseur doit être lue. Une version numérique des tensions analogiques des lignes 105 et 109 est ainsi présentée séquentiellement à un bus de données du système 123 de l'unité de traitement 91. Ce bus de système est connecté, par l'intermédiaire d'une mémoire-tampon appropriée 125, au bus 23 du système de l'unité de traitement principale 25 (représentée sur la Fig. 2).

On décrira maintenant, à l'aide du diagramme des temps de la Fig. 5 et de l'organigramme de la Fig. 8, le fonctionnement du circuit de la Fig. 4 pour obtenir les coordonnées X et Y d'une connexion particulière entre les résistances X et Y 37 et 41. La Fig. 5 représente la relation chronologique des signaux de sept lignes différentes de la Fig. 4, ces lignes étant identifiées sur la Fig. 5. A un temps initial  $t_0$ , le signal CONTROLE DU PANNEAU de la ligne 95 passe à un haut niveau (de son état 0 à son état 1) de même que le signal de commande TPD de la ligne 93. Comme on peut le voir sur le schéma logique de la Fig. 4, cette condition a pour effet que les transistors Q1, Q2 et Q4 sont à l'état

bloqué (non conducteur) tandis que les transistors Q3 et Q5 sont dans leur état passant (conducteur). Ceci a pour effet d'appliquer une tension à l'entrée non inverseuse de l'amplificateur 103 si les feuilles résistives X et Y se touchent à un point quelconque. Ceci produit une tension positive sur la ligne 127 de sortie de l'amplificateur 103 transmettant un signal SUR PANNEAU à l'unité de traitement 91. On suppose aux fins de cet exemple qu'il y a une fermeture du circuit entre les deux résistances 37 et 41 à un certain point et qu'ainsi le signal SUR PANNEAU (courbe (B) de la Fig. 5) suit la même forme que le signal CONTROLE DU PANNEAU (courbe (A)).

Au temps  $t_1$ , le signal CONTROLE DU PANNEAU de l'unité de traitement 91 est amené à retourner à son état de bas niveau ou état 0 et le multiplexeur 113 est connecté pour recevoir le signal analogique de la position X de la ligne 105 par les signaux appropriés qui lui sont appliqués sur les lignes 115 et 117 par l'unité de traitement 91. A cet instant, la combinaison d'un signal de bas niveau CONTROLE DU PANNEAU et du haut niveau qui est maintenu du signal TPD de la Fig. 5(C) a pour effet, comme on peut le voir en examinant les circuits logiques de la Fig. 4, de faire passer les transistors Q1 et Q2 à l'état conducteur tandis que les autres transistors Q2, Q3 et Q5 passent à l'état bloqué ou non conducteur. La tension au point de la résistance 41 qui est en contact avec la résistance 37 est ainsi transmise par l'intermédiaire de l'amplificateur opérationnel 103 et du multiplexeur 113 à l'entrée du convertisseur A/N 111. Au temps  $t_2$ , l'instruction est donnée au convertisseur 111 de commencer sa conversion et à un temps  $t_3$ , il reçoit l'instruction de transmettre la valeur de la tension de la ligne 105 au bus de données 123 sous une forme numérique qui représente la coordonnée X du point de contact.

Egalement au temps  $t_3$ , le signal TPD retombe à un bas niveau et le multiplexeur 113 est commuté pour recevoir le signal analogique de la position Y transmis sur la ligne 109. Les circuits logiques de la Fig. 4 ont pour effet, dans ces conditions, de faire passer les transistors Q1 et Q3 à l'état

non conducteur tandis que le transistor Q5 reste non conducteur et que les transistors Q2 et Q4 passent à l'état conducteur. La résistance X est alors utilisée pour appliquer à l'amplificateur 107 la valeur de la tension au point de la résistance 37 où un contact a été établi par l'utilisateur en appuyant à un emplacement de la surface du panneau à contact. Au temps  $t_4$ , le convertisseur A/N 111 reçoit l'instruction de commencer sa conversion et, au temps  $t_5$ , il reçoit l'instruction de transférer la valeur numérique de la tension analogique de la position Y de la ligne 109 au bus de données 123.

A ce stade, les deux coordonnées X et Y d'un point de contact sur le panneau à contact ont été déterminées et sont disponibles pour être transmises à l'unité de traitement hôte 25 (Fig. 2) par l'intermédiaire du bus 23 du système. Cependant, comme représenté sur les Fig. 5 et 8, il est préférable d'effectuer un contrôle final pour s'assurer qu'il existe une fermeture de circuit entre les résistances 37 et 41 à la fin du cycle. Le cycle dure environ 3 millisecondes dans cet exemple particulier et il est préférable de n'accepter les valeurs de coordonnées X et Y que si le contact a duré au moins pendant cette période. Dans la négative il est peu probable que les données soient des données valides. Par conséquent, au temps  $t_6$ , le signal CONTROLE DU PANNEAU et le signal TPD repassent à un haut niveau et si le signal SUR PANNEAU passe également à un haut niveau, comme indiqué sur la Fig. 5, le système sait alors que le contact détecté sur le panneau à contact était un contact effectif et que les valeurs X et Y peuvent être transmises au système pour être exploitées.

La forme de l'information numérique engendrée par l'unité de traitement 91 et transmise au bus 23 du système a été représentée sur la Fig. 7. Deux multiplets 131 et 133 sont transmis immédiatement l'un après l'autre par l'unité de traitement 91. Le multiplet (octet) de données 133 contient une représentation numérique des données tandis que le multiplet (octet) d'état 131 immédiatement précédent



identifie les données. Il y a trois types de données transmises de cette manière. Le premier type sert à indiquer qu'une des touches ou boutons 65 à 79 à contact de la Fig. 3 a été touché et à identifier le bouton actionné. L'unité de traitement 91 compare les informations de coordonnées engendrées de la manière décrite ci-dessus aux limites X et Y connues de chacune de ces zones de bouton-poussoir. Elle engendre alors dans le multiplet d'état 131 un code indiquant qu'une zone de bouton quelconque a été touchée et elle engendre dans le multiplet de données 133 pour ce type d'informations l'identité du bouton particulier qui a été actionné.

Si l'unité de traitement 91 détermine que les coordonnées X-Y reçues se trouvent dans la zone 63 de commande de curseur du panneau à contact de la Fig. 3, l'existence d'informations de commande de curseur est indiquée dans le multiplet d'état 131 de la Fig. 7 et les informations relatives aux coordonnées X-Y exactes du point de contact au cours d'un cycle donné sont incluses dans le multiplet de données 133. Etant donné que les coordonnées X et Y sont calculées toutes les 30 à 40 millisecondes, dans un mode de réalisation particulier, le déplacement d'un doigt sur la zone 63 du panneau à contact a pour effet que l'unité de traitement 91 transmet périodiquement de nouvelles informations de coordonnées X et Y en tant que partie du multiplet 133. A un premier instant, le multiplet 131 indiquera qu'une valeur de coordonnée X est indiquée dans le multiplet 133 et l'instant suivant que la valeur de coordonnées indiquée est dans la direction Y et ainsi de suite. Cette information est ensuite utilisée par l'unité de traitement hôte 25 avec le matériel ou le logiciel habituel de commande de curseur pour provoquer le déplacement du curseur sur la visu 17 sur une distance et dans une direction qui sont fonction du déplacement du doigt sur la zone 63 du panneau à contact. Ceci constitue le second des trois types de données engendrées par l'unité de traitement 91 à partir des informations X-Y ainsi obtenues du panneau à contact.

Dans un mode de réalisation préféré de ce second type

d'informations, l'unité de traitement 91 transmet dans le multiplet de données 133 le déplacement différentiel plutôt que la position X-Y absolue, c'est-à-dire que la distance de déplacement de la fermeture d'un circuit entre les  
5 feuilles résistives 37 et 41 depuis que la dernière lecture a été effectuée est indiquée. En outre, il est souhaitable de ne pas appliquer les signaux différentiels directement au matériel ou logiciel commandant le curseur étant donné que ceci aurait pour effet de faire suivre au curseur un trajet  
10 erratique dans le cas où le doigt de l'utilisateur ne suivrait pas un trajet régulier. Par conséquent, l'unité de traitement 91 effectue une certaine égalisation des valeurs de déplacement X et Y différentiel successivement acquises en transmettant au bus 31 du système une moyenne de plusieurs lectures différentielles.  
15

La Fig. 6 représente les éléments 135, 137 et 139 d'un registre interne à l'unité de traitement 91 ainsi que les éléments 141, 143 et 145 d'un autre registre interne. Chaque nouvelle valeur de déplacement X différentiel est introduite  
20 dans le premier étage 135 du premier registre tandis que chaque nouvelle valeur de déplacement Y incrémentiel est introduite dans le premier étage du second registre. Au cours de chaque cycle de curseur successif, ces valeurs sont décalées dans les registres, de gauche à droite puis détruites.  
25 La valeur de déplacement X différentiel transmise par l'unité de traitement 91 au bus 23 du système au moyen du multiplet 133 est la moyenne des trois valeurs contenues dans les éléments 135, 137 et 139 du registre. De même, la valeur du déplacement Y incrémentiel est déterminée en établissant la  
30 moyenne des trois dernières valeurs incrémentielles enregistrées dans les éléments 141, 143 et 145 du registre.  
Il est également souhaitable, dans la plupart des applications, de prévoir une caractéristique de changement d'échelle des déplacements du curseur en réponse au déplacement  
35 du doigt dans la zone X-Y 63 du panneau à contact. Ainsi, l'unité de traitement hôte 25 avant d'appliquer les valeurs X et Y incrémentielles moyennes au logiciel ou au matériel

habituel de commande de curseur manipule ces valeurs à l'aide d'une grandeur proportionnelle à la vitesse de déplacement du contact. Une technique préférée consiste à élever ces valeurs mathématiquement au carré puis à multiplier ce résultat par une constante appropriée (qui peut être l'unité). Ceci a pour effet que le curseur se déplace plus loin sur l'écran 17 de la visu pour une distance donnée de déplacement du point de contact sur le panneau à contact lorsque la vitesse de ce déplacement s'accroît. On peut voir que, dû fait que chaque valeur X et Y incrémentielle représente le déplacement du point de contact pendant une période de temps fixe qui est le temps du cycle de mesure des valeurs X et Y, l'élévation au carré de ces valeurs différentielles avant de les appliquer au matériel ou au logiciel habituel de commande de curseur déplace le curseur sur une distance qui dépend de la vitesse de déplacement du point de contact. Si l'utilisateur désire que le curseur se déplace sur une courte distance, il déplace le doigt lentement sur la surface 63 du panneau à contact. Un déplacement rapide du doigt a pour effet que le curseur se déplace plus loin pour la même distance de déplacement du doigt.

Nous référant à nouveau au format de données représenté sur la Fig. 7, le troisième et dernier type d'informations transmis de cette manière par l'unité de traitement 91 au bus 23 du système est une information indiquant qu'un coup bref a été appliqué à la zone 63 X-Y du panneau à contact. En d'autres termes, une fois que l'unité de traitement 91 a déterminé à partir des coordonnées X et Y qui lui sont présentées qu'une fermeture de circuit particulière entre les surfaces résistives 37 et 41 se trouve à l'intérieur de la zone 63 de commande du curseur X-Y, elle effectue alors d'autres investigations en effectuant des interrogations successives pour déterminer (1) si le temps de fermeture est supérieur ou inférieur à un temps fixe particulier, habituellement de l'ordre de quelques centaines de millisecondes et (2) si la fermeture de circuit a été déplacée sur une distance plus ou moins supérieure à une certaine dis-

tance définie dans la direction X ou à une autre certaine distance définie dans la direction Y pendant cet intervalle de temps fixe particulier. Si le temps de fermeture est supérieur à cette période de temps fixe particulière ou si le déplacement du contact est supérieur à l'une de ces certaines distances X et Y définies, le contact est alors interprété comme une commande de déplacement du curseur et le second type de mot de données décrit ci-dessus en se référant à la Fig. 7 est transmis.

10       Cependant, si la fermeture de circuit à une durée inférieure au temps prédéterminé et si le déplacement du contact est inférieur aux certaines distances X et Y définies, un coup bref est détecté et une forme différente de données selon la Fig. 7 est transmise au bus 23 du système. Le multiplet d'état 131 indique alors une autre touche à contact (bouton poussoir); c'est-à-dire que le coup tapé est traité simplement comme une autre zone de bouton-poussoir mais un bouton-poussoir séparé n'a pas été nécessaire pour engendrer ce signal. Le multiplet de données 133 indique par un code propre qu'il s'agit d'un coup tapé. Bien que l'emploi à la fois du temps et de la distance de déplacement du contact pour différencier un coup bref d'un signal de déplacement du curseur soit préféré pour la plupart des applications, l'emploi d'un seul de ces facteurs ou de l'autre est satisfaisant pour certaines applications.

25       Ainsi, la zone 63 de déplacement du curseur X-Y remplit la fonction supplémentaire d'engendrer un signal du type bouton-poussoir qui est très avantageusement utilisé par l'unité de traitement hôte 25 comme signal d'exécution du type qui suit normalement le repositionnement du curseur sur un écran de visu d'ordinateur. Ainsi, l'utilisateur n'a pas besoin de quitter l'écran des yeux pour trouver un autre bouton qu'il doit pousser pour provoquer une exécution désirée à l'emplacement du curseur mais il lui suffit d'appliquer un coup bref sur la zone X-Y 63 où son doigt est déjà placé:

35       Un autre avantage que présente la génération du signal

tapé de cette manière est qu'elle simplifie le circuit électrique. La simplification entraînant une réduction du nombre des éléments de circuit présente un très grand intérêt dans le cas d'un ordinateur portatif et, dans d'autres domaines d'application, elle entraîne des économies financières.

Une annexe jointe donne une liste en code d'origine d'un programme de commande de l'unité de traitement 91 qui lui permet d'exécuter toutes les fonctions de commande du système et de traitement des données que l'on a décrites ci-dessus comme étant exécutées par cette unité de traitement. Cette annexe est écrite en langage d'assemblage Intel pour le microprocesseur 8051. Le programme de commande pour certaines des étapes d'exécution représentées dans l'organigramme de la Fig. 8 est donné dans les parties de l'annexe désignées états 80, 81, 90 et 91 ainsi que dans certaines des sous-routines.

Les calculs décrits ci-dessus comme étant exécutés par l'unité de traitement 91 sont représentés dans le diagramme d'états de la Fig. 9 qui montre le fonctionnement de l'ensemble pris globalement du logiciel de commande de l'annexe. Les opérations exécutées par le logiciel de l'annexe dans les parties désignées états 10 et 11 sont représentées par un cercle 151 sur la Fig. 9. Dans cet état, le système est inactif pendant qu'il exécute ses cycles répétitifs d'interrogation du panneau à contact pour déterminer si une fermeture de circuit entre les feuilles résistives 37 et 41 s'est produite. Lorsqu'une fermeture de circuit est détectée du fait que SUR PANNEAU est retourné, le système suit un trajet 153 jusqu'aux états 20 et 21 indiqués dans un cercle 155, si cette fermeture de circuit se trouve dans la zone X-Y 63 du panneau à contact. Si la fermeture de circuit est détectée, au cours des états 10 et 11, comme étant située dans l'une des zones de bouton-poussoir 65 à 79, le système suit alors un trajet 157 jusqu'à un état 30 représenté par un cercle 159. Lorsqu'il est dans l'état 30, le système indique simplement en sortie, sous un format que l'on a décrit en se référant à la Fig. 7, la zone de bouton-poussoir parti-

culière qui a été pressée tant que le signal SUR PANNEAU reste à un haut niveau.

Si le système se trouve dans l'état 155 par suite d'une fermeture de circuit dans la zone 63 de commande de curseur X-Y, il détermine si la fermeture de circuit présente les caractéristiques d'un coup bref d'exécution ou, au contraire, d'une commande de déplacement du curseur, conformément aux critères examinés ci-dessus. S'il est déterminé qu'il s'agit d'un coup bref, le système suit un trajet 161 jusqu'à des états 50 et 51 représentés par un cercle 163. D'autre part, s'il est déterminé que la fermeture de circuit est une instruction destinée à provoquer un mouvement X-Y, le système suit un trajet 165 jusqu'aux états 40 et 41 représentés sur la Fig. 9 par un cercle 167. Lorsqu'il est dans les états 40 et 41, le système continue à produire en sortie des valeurs de déplacement X et Y différentiel sous la forme de signaux numériques dans le format que l'on a précédemment décrit en se référant à la Fig. 7.

Lorsque le système est dans les états 40 et 41, un cycle d'exécution au cours duquel a été détectée la disparition du signal SUR PANNEAU a pour effet que le système suit un trajet 169 jusqu'aux états 50 et 51. Lorsque le système est dans ces deux derniers états, il retourne aux états 40 et 41 suivant un trajet 171 si le signal SUR PANNEAU ne reste absent que pendant un temps inférieur à un temps prédéterminé (compte M). Cette caractéristique fournit la capacité d'accepter le "patinage"; en d'autres termes, le doigt de l'opérateur en se déplaçant sur la zone X-Y 63 peut s'écarter temporairement de la surface ou n'appliquer pendant une courte période de temps qu'une pression trop faible pour provoquer une fermeture de circuit sans que ceci provoque le passage du système à un autre état. Cependant, dès que le système détecte que le signal SUR PANNEAU a été absent pendant une période supérieure à un temps prédéterminé, le système, fonctionnant dans les états 50 et 51, retourne par un trajet 173 aux états initiaux 10 et 11.

Une autre caractéristique du logiciel de l'annexe est,

21.

comme représenté sur son diagramme d'états de la Fig. 9, le passage des états 50 et 51, par un trajet 175, à l'état 30, lorsque le signal SUR PANNEAU est rétabli en un temps inférieur au temps prédéterminé mais, par suite du déplacement du point de contact de la zone X-Y 63 à l'une des zones 65-79 (Fig. 3), dans l'une de ces zones de bouton.

Bien que l'on ait décrit les divers aspects de la présente invention en se référant à un mode de réalisation préféré, il est bien entendu que la présente invention et ses divers aspects ne sont limités que dans la mesure définie par les revendications annexées.

## ANNEXE

Mar 31 13:11 1983 tpad.a51 Page 1

```

; $$$
;
5 ;
; GAVILAN 21/3/83
; Richard B. Ravel
; copyright 1983 gavilan computer corporation
;-----
;-----
10 ;-----
;-----
; p2.3 - 'sur panneau' (détecter contact sur panneau)
; p2.4 - 'tpd' (mise en fonction axes des X) (1=x, 0=y)
;-----
15 ; Entrer ici pour la tâche de panneau à contact puis effectuer un
; branchement à l'état approprié
;
do_touchpad:
;
20 mov psw,n°rb1 ; choisir bloc de registres 1
mov a,tpd_state ; mettre état dans 'a'
clr c ; effacer report
subb a,n°tpd_state80 ; comp.état en cours avec n°tpd_
; état80
25 ; NOTER que n°tpd_état80 <> 0x80
jnc do_tpd1 ; branchement si état80 ou sup. puis-
; qu'on NE VEUT PAS contrôler sur
; panneau
call pad_check ; mettre indicateur sur panneau
30 do_tpd1:
mov a,tpd_state ; remettre état dans 'a'
mov dptr,n°tpd_jump_tbl ; préparer pour branchement à table
jmp @a+dptr ; effectuer branchement au moyen de
; la table ci-dessous
35 ;
tpd_jump_tbl:
ajmp tpd_s10 ; état inactif & 1ère détection sur
; panneau

```



Mar 31 13:11 1983 tpad.a51 Page 2

```

ajmp tpd_s11      ; déterminer touche ou panneau-xy
ajmp tpd_s20      ; déterminer si toujours sur panneau
ajmp tpd_s21      ; signaler coup bref ou mouvement-xy
5  ajmp tpd_s30      ; état relâchement touche
ajmp tpd_s40      ; lectures mouvement-xy
ajmp tpd_s41      ; signaler mouvement-xy ou sortir
ajmp tpd_s50      ; effectuer lecture xy
ajmp tpd_s51      ; mouvement-xy ou état relâchement
10 'coup bref'
ajmp tpd_s80      ; commencer conversion ::
ajmp tpd_s81      ; lire conversion x;préparer pour y
ajmp tpd_s90      ; commencer conversion y
ajmp tpd_s91      ; lire conversion y
15 ;-----
;-----
;-----
; ETAT 10
;-----
20 ;-----
;
tpd_s10:
    jb status.onpad, tpd_s10a ; branchement si on est sur panneau
    mov pad_count, n°2 ; on doit obtenir 2 valeurs sur pan-
25 ; neau avant d'agir sur la valeur
; du panneau à contact
;-----
; TOUTES LES SORTIES NORMALES SE FONT PAR CE POINT
;-----
30 tpd_exit:
    mov tpd_timer, n°tpd_sample_rate ; préparer temps de retour
    jmp main ; sortir et revenir ultérieurement
;-----
; Venir ici si on est sur panneau. On
35 ; doit venir ici deux fois avant d'agir.
;
tpd_s10a:

```

Mar 31 13:11 1983 tpad.a51 Page 3

```

        jnb status.adc_active, tpd_s10b ; branchement si convertisseur A/N
                                         ; (adc) n'est pas en service
5  -----
        ; TOUTES LES SORTIES POUR CONV.A/N (ADC) OCCUPE S'EFFECTUENT PAR CE POINT
        ; -----
adc_busy_exit:
        mov tpd_timer, n°adc_wait ; ré-essayer dans lms (conv.A/N(adc)
10                                     ; occupé)
        jmp main ; revenir bientôt
tpd_s10b:
        djnz pad_count, tpd_exit ; on doit obtenir 2 valeurs sur panneau
                                         avant d'agir sur la valeur
15        mov tpd_ret_state, n°tpd_state 11; préparer retour à état 11
        ; -----
        ; COMMENCER LA CONVERSION X-Y S'EFFECTUE ICI
        ; -----
start_xy_conv:
20        setb status.adc_active ; exclure toute autre activité du conv.
                                         ; A/N
        mov r0, n°padon_loc ; préparer emplacement de 'padon' (con-
                                         ; trôle du panneau)
        mov a, n°padon_0 ; préparer arrêt détection sur panneau
25        movx @r0, a ; arrêt détection sur panneau; maintenant
                                         ; prêt pour préparer conversion x
        mov r0, n°x_channel ; préparer enregistrement canal x dans
                                         ; conv.A/N
        movx @r0, a ; enregistrer canal x; données sans si-
30                                     ; gnification
        mov tpd_state, n°tpd_state80; revenir et commencer conversion x
        ; -----
        ; TOUTES LES PREPARATIONS_XY ATTENDENT SORTIE A CE POINT
        ; -----
35 xy_setup_exit:
        mov tpd_timer, n°xy_setup ; commencer conversion dans 1 ms
        jmp main ; effectuer autre chose en attendant
        ; -----

```

Mar 31 13:11 1983 tpad.a51 Page 4

```

-----
;
;
5 ;                               ETAT 11
;
-----
; Revenir ici après avoir effectué une lecture x et y
;
10 tpd_s11:
    jb  status.onpad, tpd_s11a ; branchement si on est encore sur
                                ; panneau
    ;
    mov  tpd_state, n°tpd_state10; ce doit être des parasites, recom-
15                                ; mencer
    jmp  tpd_exit          ; sortir et revenir ultérieurement
    ;
    tpd_s11a:
        call where_am_i      ; déterminer si touche ou panneau-xy
20 ;-----
; S'il y a un retour à partir de ce point, ce doit être un mouvement-xy.
; Sinon where_am_i signale une touche et retourne à l'état de relâche-
; ment de touche (30) et la détermination de CETTE adresse de retour aura
; été effectuée.
25 ;
    tpd_s11b:
        call flush_deltas    ; mettre à zéro tous les deltas et
                                ; initialiser les récepteurs de x et y
        mov  pad_count, n°tap_count; préparer comptage délai de coup bref
30 tpd_s11c:
        mov  tpd_state, n°tpd_state20; préparer retour à état 20
        jmp  tpd_exit        ; sortir et revenir ultérieurement
;-----
;
35 ;
; $$$
;

```

Mar 31 13:11 1983 tpad.a51 Page 5

```

; Richard B. Ravel      GAVILAN      21/3/83
;
5 ; -----
; -----
;          ETAT 20
; -----
; -----
10 ;
; Déterminer s'il s'agit d'un coup bref ou d'un mouvement-xy
;
tpd_s20:
    jb  status.onpad,tpd_s20c ; branchement si on est sur panneau
15 ; -----
;   SIGNALER UN COUP BREF
; -----
; Passer ici en séquence si non sur panneau==>signaler un 'coup bref'
;
20 report_tap:
    mov  b,n°xy_tap      ; indiquer un 'coup bref'
    mov  a,n°touchkey    ; le genre est une touche
    call to_8088         ; transmettre
    mov  psw,n°rbi      ; restaurer bloc de registres 1
25 ; -----
; Faire produire un 'clic' par l'avertisseur sonore
;
    call click          ; faire produire un clic par l'aver-
; tisseur sonore pour signaler un
30 ; 'coup bref.'
; -----
tpd_s20b:
    mov  pad_count,n°release_count; préparer pour état relâchement xy
35 ; ou coup bref
    mov  tpd_state,n°tpd_state50 ; retourner à l'état de relâchement
; pour un coup bref

```

Mar 31 13:11 1983 tpad.a51 Page 6

```

        jmp tpd_exit          ; sortir et revenir ultérieurement
;-----
5   ; Aboutir ici si l'on n'est pas sûr d'un 'coup bref' ou d'un mouvement
; xy et effectuer une autre lecture
;
tpd_s20c:
        jnb status.adc_active,adc_busy_exit
10          ; branchement si convertisseur A/N
          ; occupé
          ; et retourner état 20

tpd_s20d:
        mov tpd_ret_state,n°tpd_state21; retourner à état 21
15        jmp start_xy_conv    ; commencer une conversion
;
;-----
;
;
20          ;
;-----
;
;   ; contrôler si non sur panneau; si non sur panneau signaler un coup
;   ; bref; sinon, contrôler déplacement ou expiration délai
;
25 tpd_s21:
        jnb status.onpad,report_tap; branchement si on N'est PAS sur
          ; panneau
;
        call calc_deltas      ; calculer deltas et décaler dans
30          ; canaux
;-----
;   ; contrôler déplacement x
;
        mov a,latest_x        ; déplacer x le plus récent dans 'a'
35        clr c                ; mettre report pour arith. à complé-
          ; ment à 2
        subb a,first_x        ; lire '1er delta'
        jnc tpd_s21b          ; branchement si delta est positif

```

Mar 31 13:11 1983 tpad.a51 Page 7

```

tpd_s21a:
5   cpl a           ; complémenter 'a' (complément à 1)
   inc a           ; complément à 2
   clr c           ; effacer report

tpd_s21b:
   subb a,n°min_dx ; comparer 1er_dx à dx_min
10  jnc tpd_s21g   ; branchement si 1er_dx >= dx_min
   ; (ceci signifie un déplacement)
;-----
; Passer ici en séquence pour contrôler le mouvement y
;
tpd_s21c:
15  mov a,latest_y ; déplacer y le plus récent dans 'a'
   clr c           ; mettre report pour arith à complément
   ; à 2
   subb a,first_y  ; lire '1er delta'
   jnc tpd_s21e   ; branchement si delta est positif
20  tpd_s21d:
   cpl a           ; complémenter 'a' (complément à 1)
   inc a           ; complément à 2
   clr c           ; effacer le report

tpd_s21e:
25  subb a,n°min_dy ; comparer 1er_dy à dy_min
   jnc tpd_s21g   ; branchement si 1er_dy >= dy_min
   ; (ceci signifie un déplacement)
;-----
; passer en séquence et contrôler pour déterminer s'il est temps de
30 ; considérer que ce ne peut pas être un coup bref
;-----
tpd_s21f:
   djnz pad_count,tpd_s11c ; branchement si on N'EST PAS resté ici
   ; trop longtemps
35 ;-----
   ; RETOURNER A ETAT 20 ET EFFECTUER UNE
   ; NOUVELLE LECTURE
   ; aller à sortie_tpd
;-----

```

Mar 31 13:11 tpad.a51 Pge 8

```

; Aller ici si déplacement ou si temps trop
; long pour un coup bref
5 -----
; SIGNALER DOIGT APPUYE
-----
tpd_s21g:
    clr a ;
10    mov b, a ; 0 chargé dans 'b' ==> doigt appuyé
; 'sur panneau'
    mov a, n°delta_x ; indiquer delta_x
    call to_8088 ; signaler 8088 doigt appuyé
    mov psw, n°rbl ; restaurer bloc de registres i
15 -----
;
; SIGNALER DELTAS
-----
; Il faut établir la moyenne de deltas avant de les transmettre
20 ;
; signaler_deltas:
;
;
; MOYENNE DES DELTAS
25 -----
;
; Cette routine établit la moyenne des 3 deltas x et des 3 deltas y.
; Les restes respectifs sont également propagés.
;
30 avg_deltas:
    mov average_dx, n°0 ; commencer avec moyenne 0
    mov r0, n°dx1 ; adresse de dx1 dans r0
;
;
35 avg_x1:
    mov a, r0 ; 1er delta dans 'a'
    add a, n°excess_128 ; créer excès de 128
    mov b, n°3 ; préparer pour division

```

Mar 31 13:11 1983 tpad.a51 Page 9

```

    div ab                ; diviser ==> a/b
    add a,average_dx     ; accumuler moyenne
5   mov average_dx,a     ; accumuler moyenne
    mov a,b              ; reste dans 'a'
    add a,remainder_x    ; accumuler reste (toujours positif)
    mov remainder_x,a    ; sauvegarder reste accumulé
    inc r0               ; préparer pour delta suivant
10  cjne r0,n°dx3+1,avg_x1 ; branchement si pas encore 3 deltas
;-----
; Passer en séquence lorsque la moyenne des deltas x a été effectuée
; et établir la moyenne des restes de l'ajouter au delta.
; NOTE: 'a' contient déjà le reste_x
15  ;
    avg_x2:
        mov b,n°3        ; préparer pour division
        div ab           ; diviser ==> a/b
        mov remainder_x,b ; sauvegarder nouveau reste
20  add a,average_dx     ; ajouter au reste dont la moyenne a
                                ; déjà été établie
        clr c            ; préparer pour arith à complément à 2
        subb a,n°excess_128 ; faire de delta un nombre en complément à 2
25  mov average_dx,a     ; sauvegarder la nouvelle moyenné
;-----
; Il est temps maintenant d'effectuer la moyenne des deltas y
====> NOTE: r0 contient l'adresse de dy1 à la suite de la routine ci-dessus
;
30  avg_y1:
        mov average_dy,n°0 ; commencer avec moyenne 0
;
    avg_y2:
        mov a,r0          ; 1er delta chargé dans 'a'
35  add a,n°excess_128    ; créer un excès de 128
        mov b,n°3        ; préparer pour division
        div ab           ; diviser ==> a/b

```



Mar 31 13:11 1983 tpad.a51 Page 10

```

        add a,average_dy      ; accumuler moyenne
        mov average_dy,a      ; accumuler moyenne
5       mov a,b                ; reste chargé dans 'a'
        add a,remainder_y     ; accumuler reste (toujours positif)
        mov remainder_y,a     ; sauvegarder reste accumulé
        inc r0                 ; préparer pour delta suivant
        cjne r0,n°dy3+1,avg_y2 ; branchement si pas encore 3 deltas
10      ;-----
        ; passer en séquence lorsque la moyenne des deltas y a été effectuée
        ; et établir la moyenne des restes et l'ajouter au delta
        NOTE: 'a' contient déjà le reste_y
        ;
15      avg_y3:
        mov b,n°3              ; préparer pour division
        div ab                  ; diviser ==> a/b
        mov remainder_y,b      ; sauvegarder le nouveau reste
        add a,average_dy       ; ajouter la moyenne du reste
20      clr c                   ; préparer pour arithm en complément
        ; à 2
        subb a,n°excess_128    ; faire de delta un nombre en complément à 2
        ;-----
25      ; SIGNALER DELTA_Y SI NON=0
        ;-----
        ; NOTE: 'a' contient déjà la moyenne_dy
        ;
        tpd_s21h:
30      jz tpd_s21i             ; branchement si delta est 0(pas de
        ; signalisation)
        mov b,a                 ; données dans 'b'
        mov a,n°delta_y        ; indiquer valeur delta_y
        call to_8088            ; transmettre la moyenne de delta y
35      mov psw,n°rbl          ; restaurer bloc de registres 1
        ;-----
        ; SIGNALER DELTA_X SI NON=0
        ;-----

```

Mar 31 13:11 1983 tpad.a51 Page 11

```

tpd_s21i:
    mov     a,average_dx      ; moyenne de la valeur x dans 'a'
5         jz     tpd_s21j      ; branchement si delta est 0(pas de
                                ; signalisation)
    mov     b,a              ; données dans 'b'
    mov     a,n°delta_x      ; indiquer delta_x
    call    to_8088          ; transmettre la moyenne de delta x
10        mov     psw,n°rbl    ; restaurer bloc de registres 1
tpd_s21j:
    mov     tpd,state,n°tpd_state40; retourner à l'état de signalisation
                                ; du mouvement-xy

tpd_s21k:
15        jmp     tpd_exit     ; sortir et revenir ultérieurement
;-----
;
;
; $W$
20        ;
; Richard B. Ravel          GAVILAN          21/3/83
;
;-----
;-----
25        ;                   ETAT 30
;-----
;-----
tpd_s30:
30        jnb    status.onpad,tpd_s30a ; branchement si ON N'EST PAS sur panneau
        jmp     tpd_exit     ; sortir et revenir ultérieurement
;-----
; Venir ici si non sur panneau. Il est temps
; de tout recommencer à l'état 10
35        ;
tpd_s30a:
    mov     tpd_state,n°tpd_state10; préparer retour à état 10

```

Mar 31 13:11 tpad.a51 Page 12

```

      mov  pad_count,n°2      ; on doit détecter 2 valeurs sur pan-
5      ; neau avant de traiter la valeur du
      ; panneau à contact
      ;-----
      ;
      tpd_a30b:
      jmp  tpd_exit          ; sortir et revenir ultérieurement
10     ;
      ;-----
      ;
      ;
      ;
15     ;
      ;
      ;
      ;
      ;
20     ; $W$
      ;
      ; Richard B. Ravel          GAVILAN          21/3/83
      ;
      ;-----
25     ;-----
      ; ETAT 40
      ;-----
      ;-----
30     ; Si non sur panneau, signaler doigt soulevé (non sur panneau) aller
      ; à l'état 50 et attendre expiration du délai de relâchement. Si sur
      ; panneau effectuer une autre lecture et transmettre les deltas
      tpd_s40:
      jb  status.onpad,tpd_s40b ; branchement si sur panneau et contrôler
35     ; si conv.A/N (adc) est occupé
      ;-----
      ; SIGNALER DOIGT SOULEVE
      ;-----
```

Mar 31 13:11 tpad.a51 Page 13

```

tpd_s40a:
    clr  a                ;
5     mov  b,a            ; 0 chargé dans 'b' ==> doigt soulevé
                                ; (non sur panneau)
    mov  a,n°delta_y      ; indiquer delta_y
    call to_8088          ; informer 8088 doigt soulevé
    mov  psw,n°rbl       ; restaurer bloc de registres 1
10    jmp  tpd_s20b       ; charger compte de relâchement puis
                                ; ALLER A ETAT 50
;-----
tpd_s40b:
    jnb  status.adc_active,tpd_s40c
15                                ; branchement si conv A/N (adc)est
                                ; occupé
    jmp  adc_busy_exit    ; revenir prochainement (1ms)
;-----
tpd_s40c:
20    mov  tpd_ret_state,n°tpd_state41
                                ; retourner à état 41 après lecture xy
    jmp  start_xy_conv    ; retourner exécuter une conversion xy
;
;-----
25    ;-----
;                                     ETAT 41
;-----
;
30    ; Si sur panneau calculer les deltas, établir leurs moyennes et les
; transmettre. Si non sur panneau signaler doigt soulevé (non sur pan-
; neau) et aller à l'état 50
;
tpd_s41:
35    jnb  status.onpad,tpd_s40a ; branchement si non sur panneau et
                                ; signaler doigt soulevé puis
                                ; ALLER A ETAT 50
;-----

```

Mar 31 13:11 1983 tpad.a51 Page 14

```

; Passer en séquence ici si encore sur panneau,calculer les deltas,
; établir leurs moyennes et les transmettre si elles ne sont pas nulles et
5 ; RETOURNER A ETAT 40
;
tpd_s41a:
      call  calc_deltas      ; calculer les deltas et décaler dans
                               ; canaux
10      jmp   report_deltas  ; établir les moyennes de deltas et les
                               ; transmettre. RETOURNER A ETAT 40
;
;
; $$$
15 ;
; Richard B. Ravel          GAVILAN          21/3/83
;
; -----
;                               ETAT 50
; -----
;
; Etat de relâchement de mouvement_xy et de coup bref
25 ; On doit détecter un 'compte_de_relâchement' de valeurs non sur panneau
; consécutives avant de retourner à l'état 10. S'il se produit un retour
; sur le panneau, il faut alors déterminer s'il est sur le panneau_xy
; ou sur une touche. S'il est sur une touche, le signaler et aller à
; l'état 30. S'il est sur le panneau_xy,effacer les deltas,signaler.
30 ; doigt appuyé et aller à état 40 pour transmettre de nouveaux deltas.
;
tpd_s50:
      jnb  status.onpas,tpd_s50c ; branchement si non sur panneau
; -----
35 ; Passer en séquence ici si sur panneau. Préparer
; l'exécution d'une lecture xy.
;

```

Mar 31 13:11 1983 tpad.a51 Page 15

```

tpd_s50a:
    jnb status_adc_active, tpd_s50b
5      ; branchement si conv.A/N (adc) est
    ; occupé et REVENIR A ETAT 50
    jmp adc_busy_exit ; revenir dans lms (conv.A/N occupé)
tpd_s50b:
    mov tpd_ret_state, n*tpd_state51; REVENIR A ETAT 51 APRES LECTURE
10     jmp start_xy_conv ; commencer une conversion xy
    ;-----
    ; Venir ici si non sur panneau au moment de
    ; l'entrée dans cet état
    ;
15     tpd_s50c:
    djnz pad_count, tpd_s30b ; branchement si délai NON expiré et
    ; retourner d'ici à état 50 (ceci
    ; est une sortie normale)
    ;-----
20     ; Passer en séquence ici si délai expiré et si on est prêt à
    ; revenir au début du temps de comptage de sorte qu'on
    ; peut détecter un nouveau coup bref.
    ;
    tpd_s50d:
25     jmp tpd_s30a ; RETOURNER A ETAT 10
    ;-----
    ;-----
    ;-----
    ;-----
    ;-----
30     ;-----
    ;-----
    ; Contrôler si sur panneau. Si encore sur panneau déterminer s'il
    ; s'agit du panneau_xy ou d'une touche.
    ;
35     tpd_s51:
    jb status.onpad, tpd_s51b ; branchement si encore sur panneau
    ;-----

```

Mar 31 13:11 1983 tpad.a51 Page 16

```

; Passer en séquence ici si non sur panneau et
; vérifier s'il est déjà temps de revenir à état 10
5 ;
tpd_s51a:
    mov tpd_state,n°tpd_state50; préparer retour à état 50
    jmp tpd_s50c                ; contrôler expiration délai d'attente
;
10 tpd_s51b:
    call where_am_i            ; déterminer s'il s'agit d'une touche
                                ; ou du panneau_xy
;-----
; S'il se produit un retour à partir de ce point, ce doit être
15 ; un mouvement_xy. Sinon where_am_i signale une touche et retourne
; à l'état de relâchement de touche (30) et la détermination de
; CETTE adresse de retour aura été effectuée.
;
tpd_s51c:
20     call flush_deltas        ; mettre à zéro tous les deltas
                                ; et initialiser les récepteurs de x et y
;-----
; SIGNALER DOIGT APPUYE
;-----
25     clr a                    ;
    mov b,a                    ; 0 chargé dans 'b' ==> doigt appuyé
                                ; (sur panneau)
    mov a,n°delta_x            ; indiquer delta_x
    call to_8088                ; informer 8088 doigt appuyé
30     mov psw,n°rbi            ; restaurer bloc de registres 1
    jmp tpd_s21j                ; RETOURNER A ETAT 40
;-----
;
;
35 ; $W$
;
; Richard B. Ravel            GAVILAN            21/3/83

```

Mar 31 13:11 1983 tpad.a51 Page 17

```

-----
;
;
5  :                               ETAT 80
;
-----
;
;
tpd_s80:
10  mov  r0,n°start_conv      ; préparer commencement conversion x
    movx gr0,a                ; commencer conversion (données sont xxx)
    mov  tpd_state,n°tpd_state81; retourner à état 81
;
-----
;  TOUTES LES CONVERSIONS XY ATTENDENT ICI LA SORTIE
15  ;-----
xy_conv_exit:
    mov  tpd_timer,n°xy_conversion; retour dans approximativement 0,5ms
    jmp  main                  ; sortir pendant quelque temps
;
-----
20  ;
;                               ETAT 81
;
-----
;
25  ; Retourner ici lorsque la conversion x a été faite. La lire et
; préparer la conversion y.
;
tpd_s81:
    mov  r0,n°read_conv      ; préparer lecture valeur x
30  movx a,gr0                ; lire valeur x
    mov  latest_x,a          ; sauvegarder cette nouvelle valeur x
;
-----
; Préparer la conversion y
;
35  tpd_s81a:
    clr  p2.4                ; changer panneau à contact à axe y
    mov  r0,n°y_channel      ; préparer enregistrement canal y dans
; conv.A/N

```



mar 31 13:11 1983 tpad.e51 Page 18

```

movx r0,a           ; enregistrer canal y dans conv.A/N
                    ; (données sont xxx)
5   mov tpd_state,n°TPD_state90; revenir et commencer la conversion y
    jmp xy_setup_exit ; sortir pour quelque temps
;-----
;
;
10  ; $W$
;
; Richard B. Ravel           GAVILAN           21/3/83
;-----
;-----
15  ;                               ETAT 90
;-----
;-----
; Commencer la conversion y
;
20  tpd_s90
    mov r0,n°start_conv ; préparer commencement conversion y
    movx @r0,a           ; commencer conversion y(données sont
                    ; xxx)
    mov tpd_state,n°tpd_state 91; retourner à l'état 91 et lire y
25  jmp xy_conv_exit    ; sortir pour un bref instant
;-----
;-----
;                               ETAT 91
;-----
;-----
30  ; Lire la conversion y
;-----
;-----
tpd_s91:
    clr status.upper_key ; admettre touche inférieure
    mov r0,n°read_conv   ; préparer lecture conversion y
35  movx a,@r0           ; lire valeur y
    jb acc.7,tpd_s91a    ; branchement si touche inférieure
;
    setb status.upper_key ; préparer état touche supérieure
;-----

```



Mar 31 13:11 1983 tpad.a51 Page 20

```

; état dans lequel l'axe des x est choisi et le signal de contrôle du
; panneau 'padon' est à un haut niveau. Avant une conversion, le signal
5 ; 'padon' doit être ramené à zéro.
;
pad_check:
    mov r0,n°padon_loc      ; préparer emplacement 'padon'
    mov a,n°padon_1        ; préparer condition 'padon'
10    movx &r0,a             ; porter 'padon' à haut niveau
    setb p2.4              ; axe des x sur panneau à contact
;
    clr status.onpad       ; admettre non sur panneau
    mov b,n°080h          ; préparer pour une boucle d'attente
15 pad_wait1:
    djnz b,pad_wait1      ; attendre pendant environ 275 µs
;
    jnb p2.3,off_pad      ; saut si non sur panneau
;
20 on_pad
    setb status.onpad     ; indiquer sur panneau pour échantil-
;                               ; lonnage du courant
;
off_pad
25    ret
;-----
;                               CALC_DELTAS
;-----
;                               &
30 ;-----
;                               DECALER_DANS_CANAL
;-----
;
; Cette routine calcule les deltas pour x et y et décale les deltas
35 ; dans le canal d'une position à chaque fois et elle ajoute la nouvelle
; valeur. Le delta le plus ancien est éliminé par décalage en sortie.
;

```

Mar 31 13:11 1983 pad.a51 Page 21

```

; le dernier_x et le dernier_y sont mis à jour avec les valeurs les
; plus récentes. Les deltas sont des entiers de 6 bits signés en com-
5 ; plément à 2.
;
calc_deltas:
    mov     a,latest_y      ; obtenir valeur y la plus récemment dé-
                           ; calée du conv. A/N
10    clr    c              ; préparer pour arithm à complément à 2
        subb  a,last_y     ; calculer nouveau delta_y
    calc_delt1:
        mov   b,a          ; laisser delta_y dans 'b' pour décalage
        mov   last_y,latest_y ; mettre à jour dernier_y
15    ;-----
        ; Calculer maintenant delta_x
        ;
    calc_delt2:
        mov   a,latest_x   ; obtenir valeur x brute la plus récente
                           ; du conv.A/N
20    clr    c              ; préparer pour arithm.à complément à 2
        subb  a,last_x     ; calculer nouveau delta_x
                           ; laisser delta_x dans 'a' pour décalage
    calc_delt3:
25    mov   last_x,latest_x ; mettre à jour dernier_x
        ;-----
        ; 'a' doit contenir le nouveau delta_x
        ; 'b' doit contenir le nouveau delta_y
        ;-----
30    ; Décaler les valeurs delta x
        ;
    shift_pipe:
        mov   dx1,dx2      ; décaler le plus ancien en sortie et
                           ; l'intermédiaire dans le plus ancien
35    mov   dx2,dx3        ; décaler premier delta dans delta
                           ; intermédiaire
        mov   dx3,a        ; décaler nouveau delta en entrée
        ;-----

```

Mar 31 13:11 1983 pad.a51 Page 22

```

; décaler les valeurs delta y
;
5      mov   dy1,dy2      ; décaler le plus ancien en sortie et
                               ; l'intermédiaire dans le plus ancien
      mov   dy2,dy3      ; décaler premier delta dans delta
                               ; intermédiaire
      mov   dy3,b        ; décaler nouveau delta en entrée
10     ret
;
-----
;                               EFFACER DELTAS
-----
;
15     ;
; Cette routine vide les mémoires-tampons delta et initialise les
; dernières valeurs x et y et les valeurs x et y en cours (qui sont
; les mêmes pour la première lecture)
;
20     flush-deltas:
;
      mov   r0,n*dx1-1    ; préparer premier delta-1
flush_1:
      inc   r0
25     mov   @r0,n*0      ; 0 chargé dans delta
      cjne  r0,n*dy3,flush_1 ; branchement si non exécuté
;
      mov   first_x,latest_x ; maintenir 1ère valeur x
      mov   first_y,latest_y ; maintenir 1ère valeur y
30     mov   last_x,latest_x ; ceci est la première valeur prise
      mov   last_y,latest_y ; ceci est la première valeur prise
      mov   remainder_x,80   ; aucun reste x encore pour la moyenne
      mov   remainder_y,80   ; aucun reste y encore pour la moyenne
      ret                    ; retour après exécution
35     ;
;
-----
;                               WHERE AM I
-----

```

Mar 31 13:11 1983 tpad.a51 Page 23

```

;
; Cette routine détermine si le doigt est appuyé sur une touche ou sur
5 ; le panneau_xy. S'il est sur une touche, elle signale la touche, re-
; monte la valeur de retour hors de la pile et passe à l'état 30. Si
; le doigt n'est pas sur une touche, elle retourne à la routine appe-
; lante par l'intermédiaire de la variable 'tpd_ret_state'.
;
;
10 ; désignation des touches
;
;-----
; haut: 0 2 panneau-xy 4 6 8 (extrême droite)
; bas: 1 3 panneau-xy 5 7 9
15 ;-----
;
; ==> NOTE: les touches de la rangée supérieure sont PAIRES
;           les touches de la rangée inférieure sont IMPAIRES
;
20 ;-----
; Commencer contrôle au bord droit de la colonne 0
;
where_am_i:
;
25     mov  r0,n°0           ; admettre comme étant dans colonne 0
        mov  b,latest_x     ; contrôler coordonnée x
        mov  a,n°tpd_col0_max_x ; à gauche de max col.0?
        clr  c               ; préparer pour soustraction
        subb a,b             ; max_col-x_le + récent
30     jnc  where6          ; branchement si dans colonne 0
; report est mis si x_le + récent > max_
; col
;
;-----
35 ; Passer en séquence si n'est pas dans colonne 0
; b contient encore x_le + récent
;
where1:

```

Mar 31 13:11 1983 tpad.a51 Page 24

```

inc r0 ; admettre comme étant dans colonne 1
mov a,n*tpd_col1_max_x ; à gauche de max_col.1?
5 clr c ; préparer pour soustraction
subb a,b ; max_col-x_le + récent
jnc where6 ; branchement si dans colonne 1
; report est mis si x_le + récent>max_col
;-----
10 ; Passer en séquence si n'est pas dans colonne 1
; 'b' contient encore x_le + récent
;
where2:
mov a,n*tpd_max_xy ; à gauche de max_panneau-xy?
15 clr c ; préparer pour soustraction
subb a,b ; max_panneau-xy-x_le + récent
jc where3 ; branchement si NON dans panneau-xy
; report est mis si x_le + récent>max_
; col
20 ret ; retour à routine appelante si on EST
; sur panneau-xy
;-----
; Passer en séquence si n'est pas dans panneau-xy
; 'b' contient encore x_le + récent
25 ;
where3:
inc r0 ; admettre comme étant dans colonne 2
mov a,n*tpd_col2_max_x ; à gauche de max_col 2?
clr c ; préparer pour soustraction
30 subb a,b ; max_col-x_le + récent
jnc where6 ; branchement si dans colonne 2
; report est mis si x_le + récent>max_
; col
;-----
35 ; Passer en séquence si n'est pas dans colonne 2
; 'b' contient encore x_le + récent
;

```

Mar 31 13:11 1983 tpad.a51 Page 25

```

where4:
    inc r0 ; admettre comme étant dans colonne 3
5    mov a,n°tpd_col3_max_x ; à gauche de max_col 3?
    clr c ; préparer pour soustraction
    subb a,b ; max_col-x_le + récent
    inc where6 ; branchement si dans colonne 3
    ; report est mis si x_le + récent>max_
10    ; col
    ;-----
    ; Passer en séquence si n'est pas dans colonne 3
    ; 'b' contient encore x_le plus récent
    ;
15    where5:
        inc r0 ; ON DOIT ETRE dans la colonne 4
        ;
        ; Etant donné que nous savons que le doigt est dans une zone de touche,
        ; il doit se trouver dans l'une de ces colonnes. Doubler le numéro
20    ; de colonne pour obtenir des nombres pairs.
        ;
        where6:
            mov a,r0 ; colonne 0 à 4
            rl a ; multiplier par 2(on admet que les bits
25    ; les plus élevés sont des 0)
            jb status.upper_key,where7 ; branchement si une touche supé-
            ; rieure (PAIRE)
            ;-----
            ; Passer en séquence ici si touche inférieure et
30    ; incrémenter pour obtenir la valeur IMPAIRE
            ;-----
            inc a ; les touches inférieures sont impaires
            ;
            ;-----
35    ; SIGNALER UNE TOUCHE
            ;-----
        where7:

```



Mar 31 13:11 1983 tpad.a51 Page 26

```

mov    b,a                ; ce sont les données de touche
mov    a,n°touchkey      ; c'est l'état de touche
5  ;-----
; remonter l'adresse de retour de la pile étant donné qu'on retourne
; à l'état de relâchement de touche (30) et non à la routine appelante
;
; cette adresse doit être remontée hors de la pile avant d'appeler la
10 ; '8088' étant donné qu'on n'utilise qu'un seul niveau d'appels de sous-
; routine (pour la conservation de la pile).
;
pop    first_x            ; cet emplacement de mas n'est pas
;                               ; utilisé à ce moment dans la machine
15 ;                               ; à l'état tpd
pop    first_x            ; cet emplacement de mas n'est pas u-
;                               ; tilisé à ce moment dans la machine
;                               ; à l'état tpd
;-----
20 ;
call   to_8088            ; transmettre touche à 8088
mov    psw,n°rbi         ; restaurer le bloc de registres 1
;
;-----
25 ; faire produire un 'clic' pour l'avertisseur sonore
;
call   click              ; faire produire un clic par l'avertis-
;                               ; seur sonore pour un 'coup bref'
;-----
30 ;-----
mov    tpd_state,n°tpd_state30 ; RETOURNER A ETAT DE RELACHEMENT DE
;                               ; TQUCHE (30)
jmp    tpd_exit           ; sortir/revenir ultérieurement
;-----
35 ;-----
;                               CLIC
;-----
; Cette routine sert seulement à préparer l'avertisseur sonore pour

```

## REVENDEICATIONS

- 1 - Un dispositif d'entrée et de commande conçu pour être utilisé avec un système d'ordinateur numérique (11) ayant une visu (17) et un système capable d'engendrer un curseur (19) et de le déplacer sur l'écran de la visu en réponse à des signaux d'entrée, ce dispositif constituant un dispositif auxiliaire à un clavier quelconque (13) et étant caractérisé en ce qu'il comporte: une surface (21) sensible au contact qui produit un unique signal électrique indiquant la position dans deux dimensions d'un contact appliqué à cette surface; des moyens (63-79) prévus en combinaison avec cette surface pour définir plusieurs zones séparées dans l'espace sur cette surface; des moyens (151) recevant ce signal de position de contact pour engendrer un signal propre à chacune des zones d'une série de diverses zones séparées (65-79) lorsqu'un contact est appliqué à un quelconque emplacement de la zone, de sorte que chacune des diverses zones joue le rôle d'un interrupteur unipolaire à bouton-poussoir et engendre un unique signal en fonction/hors fonction qui sert à commander le système d'ordinateur; et des moyens (91) recevant ce signal de position de contact pour engendrer chronologiquement des signaux de position représentatifs d'un changement résultant du déplacement d'un contact sur la surface à l'intérieur d'une autre (63) des dites zones séparées dans l'espace de façon ainsi à produire un signal qui peut être utilisé par le système de déplacement du curseur du système d'ordinateur pour provoquer le déplacement du curseur sur une distance proportionnelle au déplacement du contact sur ladite autre zone (63) de la surface.
- 2 - Dispositif d'entrée et de commande auxiliaire selon la revendication 1, caractérisé en ce qu'il comporte, en outre, des moyens (91) qui reçoivent le signal de position de contact lorsqu'il est engendré par un contact à l'intérieur de ladite autre zone (63) des dites zones séparées dans l'espace pour engendrer encore un autre signal particulier lorsqu'un contact est appliqué à cette autre zone pendant un temps bref inférieur à un temps donné, les moyens généra-

Mar 31 13:11 1983 tpad.a51 Page 27

```
; qu'il produise un 'clic'
; Ce clic est destiné à indiquer un 'coup bref' ou l'actionnement
5 ; d'une touche
;
;
click:
      mov cycle_1sb,n°0d0h      ; valeur d'interruption pour minuteur 'à
10                               ; comptage progressif'
      mov tone_lenght,n°2      ; unetrès courte durée du son
      mov s_count,n°20         ; un son raisonnablement bas
      mov cycle_msb,n°20       ; retenir le son raisonnablement bas
      mov spkr_state,n°2       ; préparer tâche du haut-parleur pour
15                               ; qu'elle fonctionne jusqu'à achèvement
      setb et1                  ; mettre en fonction l'interruption du
                               ; 'minuteur 1'
;
      ret                       ; retour à la routine apelante! et
20                               ; l'avertisseur doit maintenant émettre
                               ; un clic
```

contact qui produit un signal électrique indiquant la position dans deux dimensions d'un contact appliqué à cette surface; des moyens (91) recevant ce signal de position de contact pour engendrer chronologiquement des signaux de position représentatifs d'un changement résultant du déplacement d'un contact sur la surface lorsque le contact reste appliqué à la surface pendant un temps supérieur à un temps prédéterminé; et des moyens (91) recevant le signal de position de contact pour n'engendrer un signal de coup bref indépendant des signaux de position chronologiques que lorsque le contact n'est appliqué à la surface que pendant un temps inférieur au temps prédéterminé de sorte que le signal de coup bref peut être utilisé par le système d'ordinateur pour exécuter une réponse sélectionnée par un déplacement précédent du curseur sur l'écran de la visu (17) sans qu'il soit nécessaire d'utiliser un interrupteur séparé.

9 - Dispositif d'entrée et de commande auxiliaire selon la revendication 8, caractérisé en ce que les moyens générateurs de signaux chronologiques comprennent des moyens pour engendrer des signaux de distance de déplacement incrémentielle par unité de temps destinés à être appliqués au système de commande du curseur du système d'ordinateur.

10 - Dispositif d'entrée et de commande auxiliaire selon la revendication 9, caractérisé en ce que les moyens générateurs de signaux de distance de déplacement incrémentielle comprennent des moyens (135-145) pour établir la moyenne de plusieurs lectures de distance de déplacement incrémentielle successives avant d'appliquer un signal incrémentiel au système de commande du curseur de sorte que le mouvement éventuellement erratique du contact sur la surface est régularisé avant que le mouvement soit communiqué au curseur.

11 - Dispositif d'entrée et de commande auxiliaire selon la revendication 9, caractérisé en ce que les moyens générateurs de signaux de distance de déplacement incrémentielle comprennent des moyens pour élever au carré chaque valeur incrémentielle avant de l'appliquer au système de commande du curseur, de sorte que la distance dont le curseur se déplace sur

l'écran de la visu (17) pour une distance de déplacement donnée du contact sur ladite surface dépend de la vitesse de ce déplacement.

5 12 - Un dispositif d'entrée et de commande conçu pour être  
utilisé avec un système d'ordinateur numérique (11) ayant  
une visu (17) et un système capable d'engendrer un curseur  
(19) et de le déplacer sur l'écran de la visu en réponse à  
des signaux d'entrée, ce dispositif constituant un disposi-  
10 tif auxiliaire à un clavier quelconque (13) et étant caracté-  
risé en ce qu'il comporte: une surface (21) sensible au  
contact qui produit un signal électrique indiquant la posi-  
tion dans deux dimensions d'un contact appliqué à cette sur-  
face; des moyens (91) recevant ce signal de position de con-  
15 tact pour engendrer chronologiquement des signaux de posi-  
tion représentatifs d'un changement résultant du déplacement  
d'un contact sur la surface lorsque le contact reste appli-  
qué à la surface pendant un temps supérieur à un temps prédé-  
terminé ou se déplace sur une distance supérieure à une cer-  
taine distance pendant ce temps prédéterminé; et des moyens  
20 (91) recevant le signal de position de contact pour engendrer  
un signal de coup bref indépendant des moyens de position  
chronologiques lorsque ledit contact existe sur la surface  
pendant un temps inférieur au temps prédéterminé et se dé-  
place sur une distance inférieure à la certaines distance  
25 pendant ce temps prédéterminé de sorte que le signal de coup  
bref peut être utilisé par le système d'ordinateur pour exé-  
cuter une réponse sélectionnée par un déplacement précédent  
du curseur sur l'écran de la visu (17) sans qu'il soit né-  
cessaire d'utiliser un interrupteur séparé.

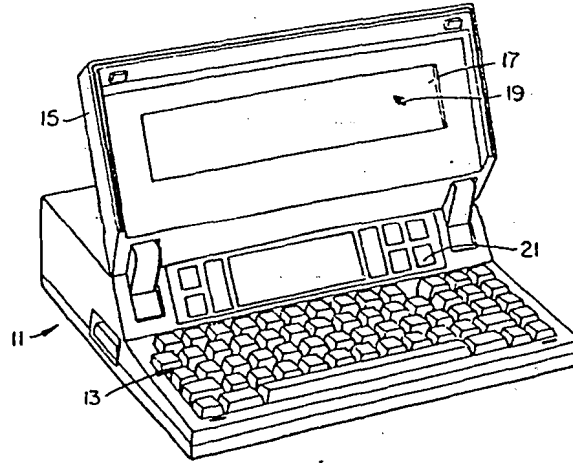


FIG. 1.

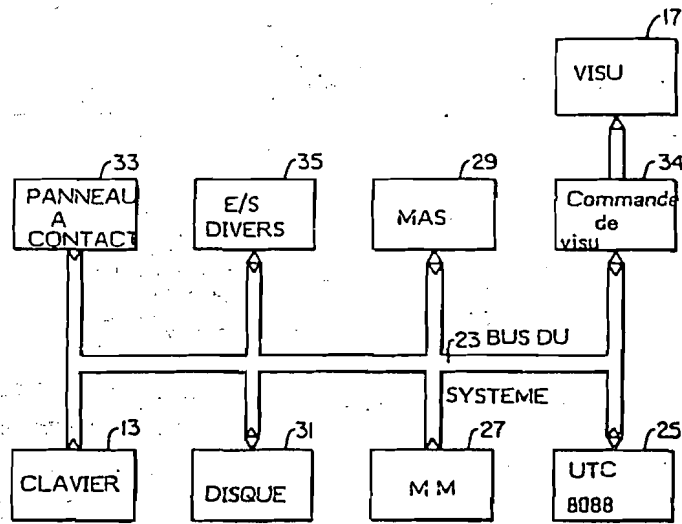


FIG. 2.

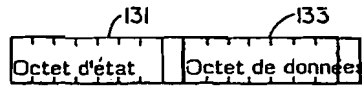


FIG. 7.

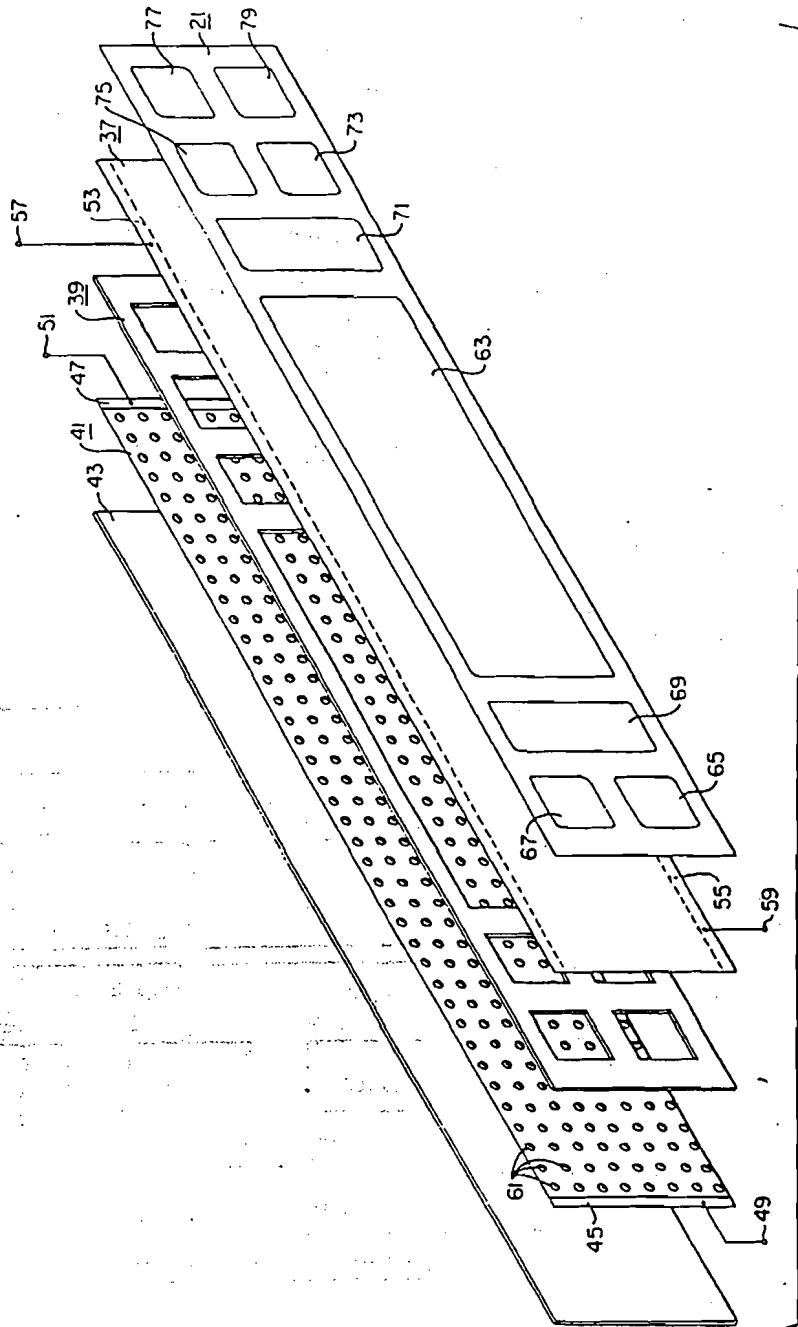


FIG. 3.

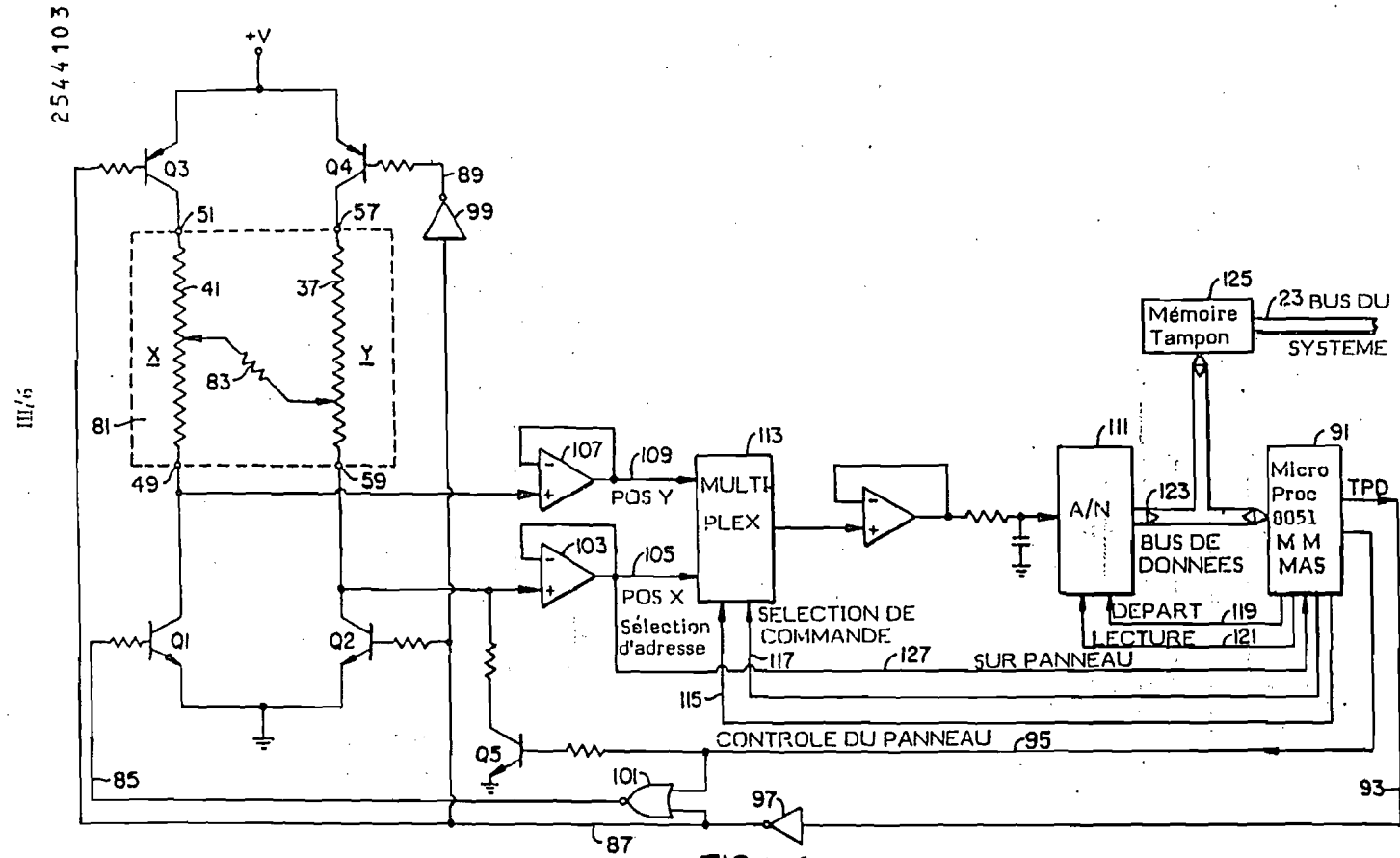


FIG. 4.



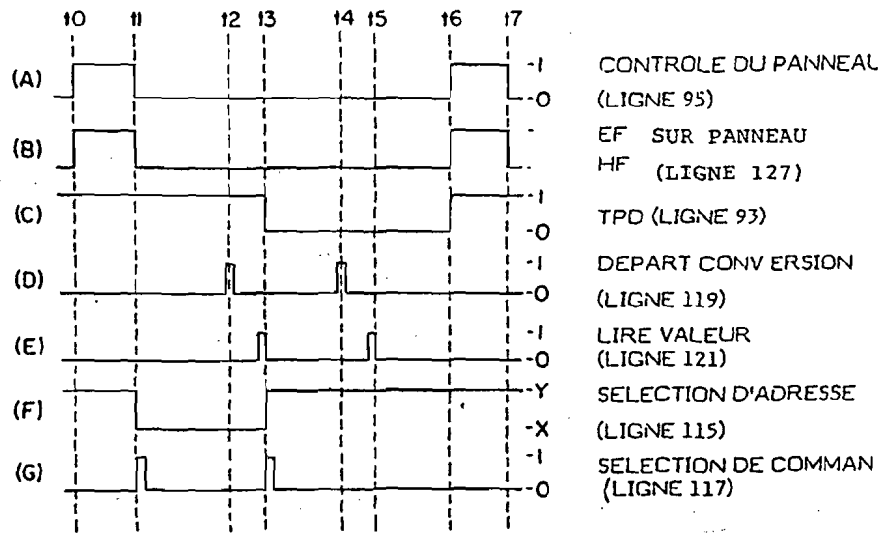


FIG. 5.

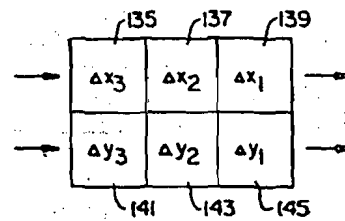


FIG. 6.

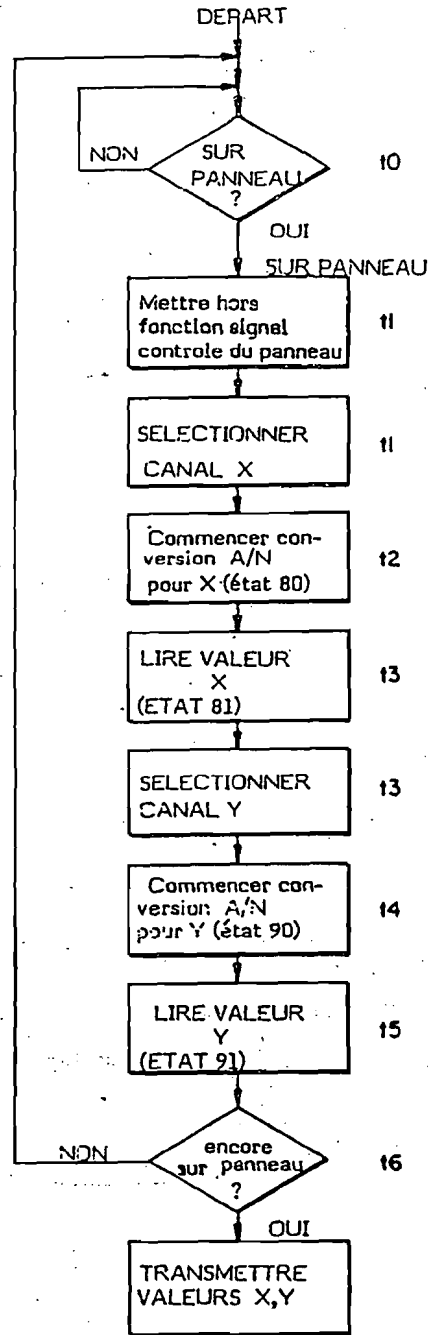


FIG. 8.

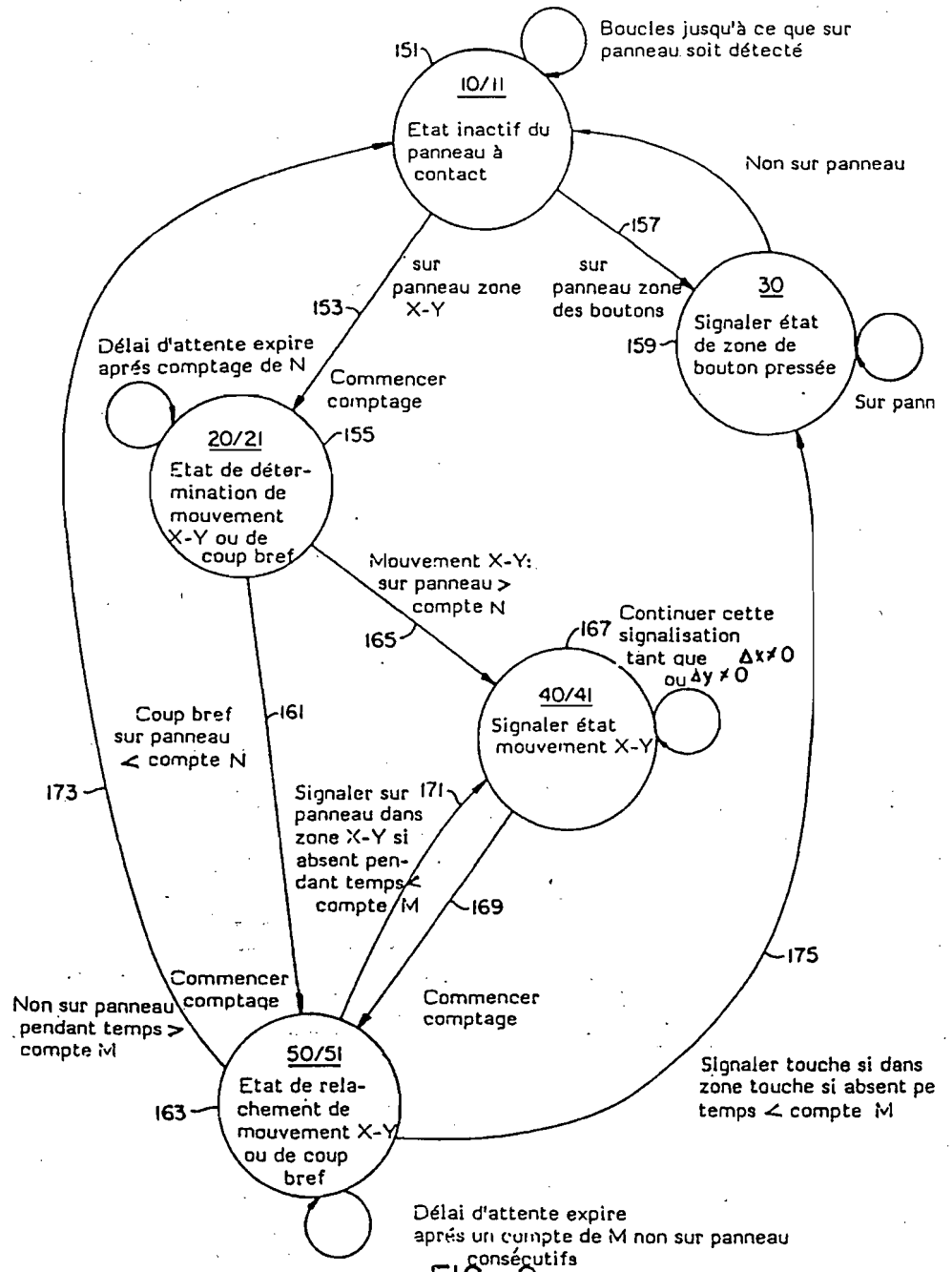


FIG. 9.

PATENT COOPERATION TREATY

From the INTERNATIONAL SEARCHING AUTHORITY

PCT

To:  
 BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN  
 Attn. SCHELLER, James C.  
 12400 Wilshire Boulevard  
 7th Floor  
 LOS ANGELES, CA 90025 1026  
 UNITED STATES OF AMERICA

NOTIFICATION OF TRANSMITTAL OF  
 THE INTERNATIONAL SEARCH REPORT  
 OR THE DECLARATION

RECEIVED  
 JUN 14 1996

(PCT Rule 44.1)

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN  
 LOS ANGELES

Date of mailing  
 (day/month/year) 07/06/96

Applicant's or agent's file reference  
 04860.P1359

FOR FURTHER ACTION See paragraphs 1 and 4 below

International application No.  
 PCT/US96/00116

International filing date  
 (day/month/year) 11/01/96

Applicant

APPLE COMPUTER, INC.

1.  The applicant is hereby notified that the international search report has been established and is transmitted herewith.

Filing of amendments and statement under Article 19:

The applicant is entitled, if he so wishes, to amend the claims of the international application (see Rule 46):

When? The time limit for filing such amendments is normally 2 months from the date of transmittal of the international search report; however, for more details, see the notes on the accompanying sheet.

Where? To the International Bureau of WIPO  
 34, chemin des Colombettes  
 1211 Geneva 20, Switzerland  
 Facsimile No.: (41-22) 740.14.35

For more detailed instructions, see the notes on the accompanying sheet.

2.  The applicant is hereby notified that no international search report will be established and that the declaration under Article 17(2)(a) to that effect is transmitted herewith.

3.  With regard to the protest against payment of (an) additional fee(s) under Rule 40.2; the applicant is notified that:

the protest together with the decision thereon has been transmitted to the International Bureau together with the applicants's request to forward the texts of both the protest and the decision thereon to the designated Offices.

no decision has been made yet on the protest; the applicant will be notified as soon as a decision is made.

4. Further action(s): The applicant is reminded of the following:

Shortly after 18 months from the priority date, the international application will be published by the International Bureau. If the applicant wishes to avoid or postpone publication, a notice of withdrawal of the international application, or of the priority claim, must reach the International Bureau as provided in Rules 90bis.1 and 90bis.3, respectively, before the completion of the technical preparations for international publication.

Within 19 months from the priority date, a demand for international preliminary examination must be filed if the applicant wishes to postpone the entry into the national phase until 30 months from the priority date (in some Offices even later).

Within 20 months from the priority date, the applicant must perform the prescribed acts for entry into the national phase before all designated Offices which have not been elected within 19 months from the priority date or could not be elected because they are not bound by Chapter II.

Name and mailing address of the International Searching Authority



European Patent Office, P.B. 5818 Patentlaan 2  
 NL-2280 HV Rijswijk  
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
 Fax: (+31-70) 340-3016

Authorized officer

Trees van Deursen

## NOTES TO FORM PCT/ISA/220

These notes are intended to give the basic instructions concerning the filing of amendments under article 19. The Notes are based on the requirements of the Patent Cooperation Treaty and of the Regulations and the Administrative Instructions under that Treaty. In case of discrepancy between these Notes and those requirements, the latter are applicable. For more detailed information, see also the PCT Applicant's Guide, a publication of WIPO.

In these notes, "Article", "Rule", and "Section" refer to the provisions of the PCT, the PCT Regulations and the PCT administrative Instructions respectively.

### INSTRUCTIONS CONCERNING AMENDMENTS UNDER ARTICLE 19

The applicant has, after having received the international search report, one opportunity to amend the claims of the international application. It should however be emphasized that, since all parts of the international application (claims, description and drawings) may be amended during the international preliminary examination procedure, there is usually no need to file amendments of the claims under Article 19 except where, e.g. the applicant wants the latter to be published for the purposes of provisional protection or has another reason for amending the claims before international publication. Furthermore, it should be emphasized that provisional protection is available in some States only.

#### What parts of the international application may be amended?

The claims only.

The description and the drawings may only be amended during international preliminary examination under Chapter II.

**When?** Within 2 months from the date of transmittal of the international search report or 16 months from the priority date, whichever time limit expires later. It should be noted, however, that the amendments will be considered as having been received on time if they are received by the International Bureau after the expiration of the applicable time limit but before the completion of the technical preparations for international publication (Rule 46.1).

#### Where not to file the amendments?

The amendments may only be filed with the International Bureau and not with the receiving Office or the International Searching Authority (Rule 46.2).

Where a demand for international preliminary examination has been/is filed, see below.

**How?** Either by cancelling one or more entire claims, by adding one or more new claims or by amending the text of one or more of the claims as filed.

A replacement sheet must be submitted for each sheet of the claims which, on account of an amendment or amendments, differs from the sheet originally filed.

All the claims appearing on a replacement sheet must be numbered in Arabic numerals. Where a claim is cancelled, no renumbering of the other claims is required. In all cases where claims are renumbered, they must be renumbered consecutively (Administrative Instructions, Section 205(b)).

#### What documents must/may accompany the amendments?

##### Letter (Section 205(b)):

The amendments must be submitted with a letter.

The letter will not be published with the international application and the amended claims. It should not be confounded with the "Statement under Article 19(1)" (see below, under "Statement under Article 19(1)").

The letter must indicate the differences between the claims as filed and the claims as amended. It must, in particular, indicate, in connection with each claim appearing in the international application (it being understood that identical indications concerning several claims may be grouped), whether

- (i) the claim is unchanged;
- (ii) the claim is cancelled;
- (iii) the claim is new;
- (iv) the claim replaces one or more claims as filed;
- (v) the claim is the result of the division of a claim as filed.

## NOTES TO FORM PCT/ISA/220 (continued)

The following examples illustrate the manner in which amendments must be explained in the accompanying letter:

1. [Where originally there were 48 claims and after amendment of some claims there are 51]:  
"Claims 1 to 29, 31, 32, 34, 35, 37 to 48 replaced by amended claims bearing the same numbers; Claims 30, 33 and 36 unchanged; new claims 49 to 51 added."
2. [Where originally there were 15 claims and after amendment of all claims there are 11]:  
"Claims 1 to 15 replaced by amended claims 1 to 11."
3. [Where originally there were 14 claims and the amendments consist in cancelling some claims and in adding new claims]:  
"Claims 1 to 6 and 14 unchanged; claims 7 to 13 cancelled; new claims 15, 16 and 17 added." or  
"Claims 7 to 13 cancelled; new claims 15, 16 and 17 added; all other claims unchanged."
4. [Where various kinds of amendments are made]:  
"Claims 1-10 unchanged; claims 11 TO 13, 18 and 19 cancelled; claims 14, 15 and 16 replaced by amended claim 14; claim 17 subdivided into amended claims 15, 16 and 17; new claims 20 and 21 added."

### "Statement under article 19(1)" (Rule 46.4)

The amendments may be accompanied by a statement explaining the amendments and indicating any impact that such amendments might have on the description and the drawings which cannot be amended under Article 19(1).

The statement will be published with the international application and the amended claims.

The statement should be brief, it should not exceed 500 words if in English or if translated into English.

It should not be confounded with and does not replace the letter indicating the differences between the claims as filed and as amended. It must be filed on a separate sheet and must be identified as such by a heading, preferably by using the words "Statement under Article 19(1)."

It should not contain any disparaging comments on the international search report or the relevance of citations contained in that report. Reference to citations, relevant to a given claim, contained in the international search report may be made only in connection with an amendment of that claim.

### In what language?

The amendments must be made in the language in which the international application is published. The letter and any statement accompanying the amendments must be in the same language as the international application if that language is English or French; otherwise, it must be in English or French, at the choice of the applicant.

### Consequence if a demand for international preliminary examination has already been filed?

If, at the time of filing any amendments under Article 19, a demand for international preliminary examination has already been submitted, the applicant must preferably, at the same time of filing the amendments with the International Bureau, also file a copy of such amendments with the International Preliminary Examining Authority (see Rule 62.2(a), first sentence).

### Consequence with regard to translation of the international application for entry into the national phase?

The applicant's attention is drawn to the fact that, where upon entry into the national phase, a translation of the claims as amended under Article 19 may have to be furnished to the designated/elected Offices, instead of, or in addition to, the translation of the claims as filed.

For further details on the requirements of each designated/elected Office, see Volume II of the PCT Applicant's Guide.

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference 04860.P1359	<b>FOR FURTHER ACTION</b> see Notification of Transmittal of International Search Report (Form PCT/ISA/220) as well as, where applicable, item 5 below.	
International application No. PCT/US96/00116	International filing date (day/month/year) 11/01/96	(Earliest) Priority Date (day/month/year) 31/01/95
Applicant APPLE COMPUTER, INC.		

This international search report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This international search report consists of a total of 3 sheets.

It is also accompanied by a copy of each prior art document cited in this report.

1.  Certain claims were found unsearchable (see Box I).

2.  Unity of invention is lacking (see Box II).

3.  The international application contains disclosure of a nucleotide and/or amino acid sequence listing and the international search was carried out on the basis of the sequence listing

filed with the international application.

furnished by the applicant separately from the international application,

but not accompanied by a statement to the effect that it did not include matter going beyond the disclosure in the international application as filed.

Transcribed by this Authority

4. With regard to the title,  the text is approved as submitted by the applicant.

the text has been established by this Authority to read as follows:

**METHOD AND AN APPARATUS FOR SIMULATING THE STATES OF A MECHANICAL BUTTON ON A TOUCH-SENSITIVE INPUT DEVICE**

5. With regard to the abstract,

the text is approved as submitted by the applicant.

the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box III. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority.

6. The figure of the drawings to be published with the abstract is:

Figure No. 2  as suggested by the applicant.

because the applicant failed to suggest a figure.

because this figure better characterizes the invention.

None of the figures.

INTERNATIONAL SEARCH REPORT

International Application No  
PCT/US 96/00116

A. CLASSIFICATION OF SUBJECT MATTER IPC 6 G06F3/033		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) IPC 6 G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used)		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	PROCEEDINGS OF THE ANNUAL CONFERENCE OF THE ENGINEERING IN MEDICINE AND BIOLOGY SOCIETY, PHILADELPHIA, NOV. 1 - 4, 1990, vol. 12: 1990, 1 November 1990, PEDERSEN P C; BANU ONARAL, pages 2296-2297, XP000239010 HAMANN G E ET AL: "NOD AT YOUR COMPUTER: SWITCHLESS SELECTION TECHNIQUES USING A HEADPOINTING DEVICE" see page 2296, right-hand column, line 14 - line 34; figure 2 ---	1-20
A	FR,A,2 544 103 (GAVILAN COMPUTER CORP) 12 October 1984 see page 19, line 17 - page 20, line 36; figure 9 --- -/--	1,2,12, 21
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C.		
<input checked="" type="checkbox"/> Patent family members are listed in annex.		
* Special categories of cited documents :		
*A* document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) *O* document referring to an oral disclosure, use, exhibition or other means *P* document published prior to the international filing date but later than the priority date claimed *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. *&* document member of the same patent family		
Date of the actual completion of the international search 20 May 1996		Date of mailing of the international search report 07. 06. 96
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl, Fax (+ 31-70) 340-3016		Authorized officer Bailas, A

Form PCT/ISA/210 (second sheet) (July 1992)



INTERNATIONAL SEARCH REPORT

International Application No  
PCT/US 96/00116

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP,A,0 490 001 (IBM) 17 June 1992 see column 5,- line 10 - column 8, line 1 -----	1,2,12, 21

2

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No  
PCT/US 96/00116

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
FR-A-2544103	12-10-84	DE-A- 3413114	25-10-84
		GB-A- 2139762	14-11-84
		JP-A- 59200336	13-11-84
-----			
EP-A-0490001	17-06-92	JP-A- 4268619	24-09-92
		US-A- 5432531	11-07-95
-----			

G06F 15/20

**Proceedings of the Twelfth Annual  
International Conference of the IEEE  
Engineering in Medicine and Biology Society**

Volume 12: 1990

Philadelphia, Pennsylvania, USA • November 1 - 4, 1990

EDITED BY

Peder C. Pedersen  
Banu Onaral

90CH2936-3



B0178250  
O.E.B. Doc. LK.  
27 MEI 1991

2080/91

Part 5/5

*Clinical, Therapeutic and Rehabilitation  
Aspects of Biomedical Engineering*

THIS PART CONTAINS:

- Track 8: Biorobotics
- Track 11: Clinical Engineering
- Track 13: Critical-Care Monitoring and Control
- Track 28: Sleep and Respiratory Control Dynamics
- Track 2: Bioengineering in Dentistry
- Track 4: Biomaterials
- Track 5: Biomechanics
- Track 24: Neuromuscular Systems
- Track 27: Rehabilitation Engineering
- Track 3: Bioengineering Education

G06F3/0088



XP 000239010

p. 2296-7

Pour le titre du livre  
voir en seconde page

## NOD AT YOUR COMPUTER: SWITCHLESS SELECTION TECHNIQUES USING A HEADPOINTING DEVICE

G.E. Hamann, G.F. Shein\*, M. Milner

Institute of Biomedical Engineering  
\*Department of Industrial Engineering  
University of Toronto, Toronto, Ontario

The Hugh MacMillan Rehabilitation Centre  
Toronto, Ontario

### ABSTRACT

Current graphical user interfaces often require very fine hand control in order to manipulate a typical pointing device such as a mouse or a trackball. This can be a barrier to many people with physical disabilities who do not have such fine control with their hands. In this paper, the emulation of a mouse by a headpointing device is described. In particular, two switchless techniques are described in which head gestures are used to emulate the mouse buttons.

### INTRODUCTION

A wide range of devices and programs have been developed and modified to allow people with disabilities access to computers. Until recently, these efforts have concentrated on text-based applications and keyboard emulators. Today, graphical user interfaces (GUIs) such as Microsoft Windows™ and IBM OS/2 Presentation Manager™ are quickly replacing the conversational style of human-computer interaction. The fine hand control needed to manipulate a typical pointing device that goes along with a GUI, such as a mouse or a trackball, presents a new barrier to many people with physical disabilities [1].

### BACKGROUND

An example of a keyboard emulator is the Long Range Optical Pointer (LROP), which was originally developed at the Trace Research and Development Center and is now available from Words+, Inc. [2,3]. It is a headpointer which detects the raster scan of the computer display and calculates the position at which the user is pointing, similar to a lightpen. Currently it only operates with software which displays a visual representation of a keyboard on the screen from which a user can select keystrokes.

People with good head control can use a headpointing device to emulate the mouse and move a cursor about the screen. Mouse button selections can be made with the aid of external switches such as sip-and-puff or eyebrow switches. As the user tries to click a button while pointing at a particular location on the screen, however, the extra effort often causes the head to move, resulting in an erroneous selection.

A method is required to emulate the mouse button actions without external, adapted switches. The solution proposed here is to use a natural head gesture: the nod. Using head gestures to implement switchless selection techniques is a novel approach, although gestural hand input has been explored in the field of human computer interaction [4]. Through a combination of pauses with head nods and head shakes, it is possible to differentiate among several different intended selections. The measurable parameters include: (1) pause time, (2) direction, (3) duration, (4) speed, and (5) distance of movement. These have been used to formulate several selection techniques, two of which are described in the following section.

### IMPLEMENTATION

#### Mapping

Application programs which use graphical user interfaces require the user to perform various low level tasks with the mouse and its buttons. Tasks such as selecting a character or a file are often done by clicking a mouse button once. Double clicks can be used to select a word or to start an ap-

plication. Highlighting a phrase or moving a file is accomplished by dragging: pushing the mouse button and holding it down while moving the mouse to a new position. Thus the user's actions are performed with the input device, which then sends messages to the application. The application translates these messages into the required tasks. Figure 1 illustrates this flow of information.

Since most commercial applications only understand mouse button messages such as single and double clicks or drags, the output from alternative devices must be mapped or translated into the standard mouse button messages, as shown in the lower part of Figure 1. The user's actions, the gesture recognizer, and the mapping function combine to form a selection technique for the headpointer.

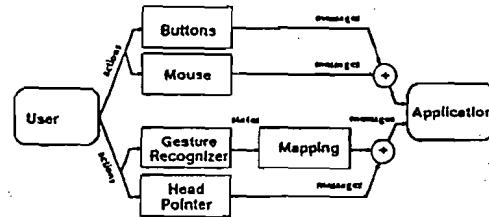


Figure 1. Mapping user actions to mouse button messages

#### Multi-Level Pause Technique

Using a simple pause or dwell time has been a common technique for implementing a single mouse button click in order to select characters from a visual keyboard. The person holds the pointer steady over the intended target for a predetermined amount of time before the system sends the single click message and the selection is made. This technique can be expanded to multiple levels to allow a greater variety of mouse button messages to be sent. The state transition diagram in Figure 2 illustrates this.

In state 0, the system waits for the user to hold the cursor steady within a given area. When the first pause level has been completed, a warning click is heard and the system transfers to state 1. Continuing to hold completes a longer pause, causing the system to beep, send a mouse button down message, and transfer to state 2. In this state, the user can drag a selected object on the screen to a new location. A scratching sound is heard to give the impression of dragging an object over a surface. Pausing longer sounds another beep as the emulated mouse button is released and the system transfers to state 3. A further pause causes a double click mouse button message to be sent along with a final beep. The system then resets and returns to state 0. The user can bypass or disable the dragging action by using a separate configuration program, as indicated by the direct branch from state 1 to state 3.

#### The Nod and Shake Technique

The technique of using multi-level pause times works well, but it is limited to only one mouse button. It also becomes increasingly difficult to hold the cursor steady for the extended pause levels. These difficulties can be overcome by using natural head gestures to accelerate the selections.

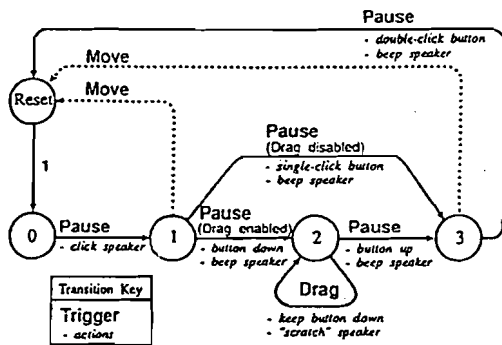


Figure 2. State transition diagram for the "Multi-Level Pause" technique.

Figure 3 illustrates the Nod and Shake technique of emulating the mouse button methods. As with the multi-level pause technique, an initial pause causes the transition from state 0 to state 1. In addition, a "cursor clutch" is set, which locks the cursor in position on the screen. From state 1, each gesture takes two stages: an initial movement, followed by a reversal of the movement direction. For instance, a gesture in the downwards direction (nod down) causes a transition from state 1 to state 2 accompanied by a speaker click. Reversing directions to return to the original position completes the nod and sends a single click of the left mouse button, releasing the cursor clutch and sounding a beep. The system is then reset and returns to state 0. Similarly, an upward gesture immediately followed by a downward return emulates a double click of the left mouse button and a gesture to the right and back emulates a single click of the right button. If the appropriate gesture is not completed within a specified time, the system is reset and no mouse button is emulated. Dragging is not done directly with the Nod and Shake technique. Instead, the user pauses long enough for the Multi-Level Pause technique to begin.

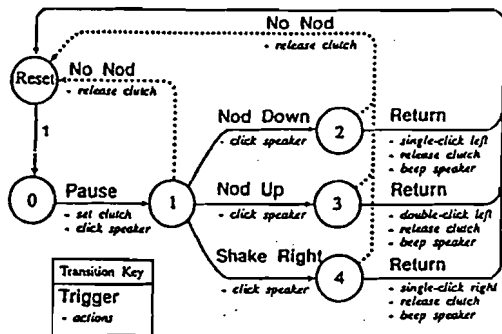


Figure 3. State transition diagram for the "Nod and Shake" technique.

**Software**

To investigate the switchless selection techniques, a Long Range Optical Pointer was modified to provide a signal with acceptable resolution and stability. The two techniques described above were implemented as a device driver under Microsoft Windows 2.1 in order to access a large number of existing and future commercially available applications. This has significant financial and vocational benefits in the rehabilitation field.

A configuration utility was written as a standard Windows application to allow the user to customize the device driver and test changes interactively. Each of the two selection techniques can be independently disabled or enabled, the dragging action can be disabled for the multi-level pause technique, and the cursor clutch can be disabled for the nod and shake

technique. The program also allows adjustment of the pause interval, the pause target size, the threshold for the nod and shake distance, and the maximum time for the completion of a gesture.

**PRELIMINARY RESULTS**

The modified headpointer has been field-tested with seven clients. Three of the clients had spinal cord injuries resulting in quadriplegia, three had athetoid cerebral palsy, and one had a peripheral nerve syndrome. They have provided valuable feedback for the iterative development of the techniques. A painting program and a file manager were used in the field-tests.

Initially, fatigue was high as the users concentrated on holding the cursor steady by tensing their neck muscles. This was counter-productive as it caused more tremor. Learning to relax reduced the users' tremor, making it easier to pause in a given location. As they learned the techniques, the users also became confused by the profusion of clicks and beeps produced by the computer. It was often necessary to explain to them what state the system was in.

In spite of these minor difficulties, the techniques worked well, providing the users with control over a graphical user interface which they had not had before.

**FUTURE WORK**

Further research will be concentrated in several areas. To give the user more confidence in the state of the system, better visual and audio feedback will be investigated. This should decrease the concentration required, lowering fatigue and reducing confusion.

In addition, some basic research questions will be investigated. Jagacinski and Monk suggest that the information processing rate of the head and neck in positioning and tracking tasks is approximately 5 bits per second [5]. Does the addition of the new selection techniques increase or decrease this rate?

A variety of parameters for the techniques can be adjusted. Although the usable extremes of these settings can be found by trial and error, further research will be conducted to investigate criteria and find tactics to adjust for the optimal settings.

**ACKNOWLEDGEMENTS**

The University of Toronto has provided support for this project through a Mary H. Beatty Fellowship (1989) awarded to G.E. Hamann. IBM Corporation, IBM Canada Ltd., TRACE Research and Development Center, and Words+, Inc. have generously donated equipment and software.

Nicholas Brownlow and Rod McGill at the Hugh MacMillan Rehabilitation Centre provided valuable advice, suggestions and comments.

**REFERENCES**

- [1] Brownlow, N., Shein, F., Thomas, D., Milner, M., and Parnes, P. "Direct Manipulation: Its Significance for People with Disabilities", *Proceedings of the RESNA 12th Annual Conference*, New Orleans, Louisiana, 1989, 244-245.
- [2] Gunderson, J., and Vanderheiden, G. "One Screen Multiplexed Keyboard for Transparent Access to Standard IBM Software", *Proceedings of ICAART 88*, Montreal, Quebec, 1988, 378-379.
- [3] Words+, Inc., P.O. Box 1229, Lancaster, California, 93534.
- [4] Buxton, W., Fiume, E., Hill, R., Lee, A., and Woo, C., "Continuous Hand-Gesture Driven Input", *Proceedings of Graphics Interface '83*, Edmonton, Alberta, 1982, 191-195.
- [5] Jagacinski, R.J., and Monk, D.L., "Fitts' Law in Two Dimensions With Hand and Head Movements", *Journal of Motor Behavior*, 1985, 77-95.

**ADDRESS**

Microcomputer Applications Programme  
The Hugh MacMillan Rehabilitation Centre  
350 Rumsey Road  
Toronto, Ontario M4G 1R8  
CANADA

(416) 762-8759 ext. 534