

12

## DEMANDE DE BREVET D'INVENTION

A1

22 Date de dépôt : 24 juillet 1987.

30 Priorité : US, 25 juillet 1986, n° 06/889,513 et 24 mars 1987, n° 07/029,772.

43 Date de la mise à disposition du public de la demande : BOPI « Brevets » n° 4 du 29 janvier 1988.

60 Références à d'autres documents nationaux apparentés :

71 Demandeur(s) : LINUS TECHNOLOGIES, société anonyme. — US.

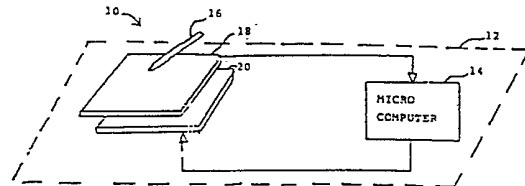
72 Inventeur(s) : Ralph Skarlew.

73 Titulaire(s) :

74 Mandataire(s) : Cabinet Benoit-Lefebvre.

54 Appareil et procédé de reconnaissance et visualisation de signes manuscrits.

57 L'appareil comprend : (a) un moyen de visualisation 20 permettant de fournir une représentation visuelle d'un caractère graphique en réponse à des signaux de sortie fournis par le dispositif d'utilisation 14, (b) un ensemble à écran d'entrée permettant de produire un train de signaux d'entrée au fur et à mesure qu'un utilisateur trace un signe ou caractère graphique, cet ensemble à écran d'entrée comprenant un écran d'entrée 18 translucide présentant des surfaces supérieure et inférieure, la surface inférieure étant disposée par rapport à la surface de visualisation d'une façon telle que la représentation visuelle produite par ledit moyen de visualisation 20 soit visible du côté de la surface supérieure de cet écran d'entrée 18, et (c) des moyens permettant de déterminer l'identité dudit caractère graphique et de fournir au moyen de visualisation 20 des signaux de sortie représentatifs de ce caractère.



La présente invention concerne d'une manière générale un système d'entrée sans clavier pour ordinateur et, lorsque celui-ci est combiné à une unité centrale, un système informatique à introduction de données sans clavier. D'une manière plus précise, l'invention concerne un dispositif de stockage, de traitement et de transfert d'informations dans lequel on introduit du texte, des données, des fonctions et des instructions informatiques en écrivant à la main des signes et des caractères de type alphanumérique ou de tout autre type, à l'aide d'un stylet analogue à un crayon et sur un écran d'entrée/sortie (E/S). Dans un mode préféré de réalisation, cet écran E/S comprend un écran à contact qui est transparent et est monté par-dessus un visuel de sortie qui est sensiblement plat. Dans ce mode préféré de réalisation, l'invention offre un système informatique qui est autonome, mais celui-ci peut également servir de périphérique pour un ordinateur central.

Les ordinateurs classiques à clavier offrent actuellement des quantités importantes d'informations et des logiciels d'application très élaborés. On pourrait accroître grandement l'utilité de ces informations et de ces logiciels d'application si l'on pouvait saisir du texte et des données et manipuler les logiciels d'application en écrivant, d'une façon normale, directement sur un visuel plat. Il existe ainsi le besoin de permettre que le caractère polyvalent de la technique informatique puisse s'étendre et la faisant utiliser par des personnes auxquelles un clavier n'est pas familier. Le besoin se fait également ressentir d'un système informatique portable qui soit léger, fiable, précis et peu onéreux et qui permette qu'on l'utilise en étant debout ou en marchant. Une façon de réduire le coût et l'encombrement et d'accroître le caractère polyvalent consiste à utiliser un système d'entrée sans clavier, tel qu'un écran à contact. Toutefois, ce type de dispositif d'entrée ne permet pas facilement une introduction détaillée précise dans le cadre d'une exploitation en temps réel à résolution élevée et d'une manière qui soit familière et naturelle à l'utilisateur.

On peut utiliser de nombreuses techniques de positionnement pour respecter les conditions requises par la technique d'introduction par détection de position. Parmi ces conditions requises, on peut essentiellement citer l'exactitude, la résolution et la vitesse. Comme techniques considérées, on peut citer les techniques mécaniques, électrostatiques électromagnétiques, acoustiques, optiques et par inertie. Ce que l'on recherche dans le présent système est que son utilisation soit aussi semblable que possible à l'écriture au stylo ou au crayon sur du papier. Un problème à résoudre réside dans le caractère de proximité : un stylo utilisé sur du papier ne

laisse une trace que lorsqu'il est effectivement en contact. Nombre des techniques considérées nécessitent un détecteur supplémentaire de « crayon posé » qui est complexe à utiliser pour de nombreux crayons du commerce. Un autre problème à résoudre réside dans l'angle d'écriture : un crayon laisse la même trace quel que soit l'angle d'écriture. Dans nombre des techniques considérées, le détecteur de position est déplacé par le bout supérieur du crayon, si bien qu'un angle de ce crayon provoque des déplacements erronés. A côté de ces problèmes généraux, chaque technique présente de nombreux avantages et inconvénients en ce qui concerne (1) le crayon : sa taille, son poids, sa forme et le fait de savoir s'il a besoin d'être alimenté en courant et/ou d'être câblé, et en ce qui concerne (2) la surface d'écriture, sa transparence, son caractère lisse, son « toucher » et le fait de savoir s'il est besoin ou non d'un contact matériel (par opposition à une pression transmise par l'intermédiaire de feuilles superposées, de papier).

On connaît un certain nombre de dispositifs autonomes permettant d'observer et de traiter des quantités importantes d'informations. La plupart utilisent, pour stocker les données, des moyens optiques, magnétiques ou électroniques à circuits intégrés de stockage. Un exemple représentatif de cette partie de la technique est constitué par le brevet US 4.159.417 (Rubincam) qui présente un livre électronique portable dont la configuration autorise l'appel sélectif page par page de quantités importantes de données numériques et les présente sur un écran plat à circuits intégrés. Le mode préféré de réalisation présenté dans ce brevet utilise comme moyen principal de stockage une carte holographique embrochable qui peut contenir plusieurs centaines de pages de texte sous forme numérique. Toutefois, le dispositif de ce brevet ne permet pas de saisir ou de traiter de nouvelles informations ou un nouveau texte.

Dans le brevet US 4.016.542 (Azure), il est présenté un système électronique de collecte de données qui utilise pour sa mémoire principale une mémoire vive (RAM) à circuits intégrés. Ce brevet, qui décrit pour la saisie des données un clavier classique, porte sur un système de stockage et de transmission de données qui est portatif, transportable à la main, ainsi qu'un visuel à diodes électroluminescentes (DEL) et divers connecteurs d'entrée/sortie (E/S).

Le brevet U.S. 3.487.731 (Franck) présente des moyens permettant de convertir des tracés manuscrits en données de caractères en utilisant un système informatique. L'invention décrite est basée sur la coïncidence de formes à disposition matricielle et n'utilise pas de technique de visualisation à coïncidence.

Le brevet U.S. 4.491.960 (Brown) présente un système de reconnaissance de signes manuscrits dans lequel un réseau de Points image, se présentant sous la forme d'un échantillonnage de lignes de trame, est converti en listes utilisables par segments qui sont filtrées et condensées de façon à donner des caractéristiques topologiques que l'on analyse alors à l'aide d'un mécanisme logique à arbre de décision.

Le brevet U.S. 4.262.281 (Buckle et autres) présente un dispositif de reconnaissance de tracés manuscrits. Le mode de réalisation décrit est destiné à être utilisé avec un ordinateur central et n'emploie pas de technique de visualisation à coïncidence.

Le brevet U.S. 4.475.239 (Van Raamsdonk) présente un appareil d'édition de textes. Ce brevet fait appel à l'utilisation de papier comme moyen intermédiaire permettant l'introduction des fonctions d'édition et nécessite un clavier classique pour la saisie des textes.

Le brevet U.S. 4.521.909 (Wang) présente un système de reconnaissance de tracés à deux niveaux. Ce système est conçu pour être utilisé avec un instrument optique.

Le brevet U.S. 4.520.357 (Castleberry et autres) présente un système électroscopique d'introduction et visualisation d'informations comportant un stylet à écrire. Le mode de réalisation décrit ne prétend pas présenter la vitesse ou la précision permettant la reconnaissance de caractères manuscrits.

Comme éléments supplémentaires de la technique antérieure qui décrivent des dispositifs électroniques portables fournissant des quantités importantes d'informations stockées de divers types, on peut citer les brevets U.S. 4.218.760 (Lévy) 4.415.486 (Laine) et 3.932.859 (Kriakides et autres). Le premier et le troisième brevets relèvent des dictionnaires électroniques, tandis que le second brevet présente un système programmable de rappel pour télévision. Aucun de ces dispositifs ne divulgue l'utilisation d'une introduction manuscrite.

Dans les brevets U.S. 4.071.691, 4.129.747, 4.198.539, 4.293.734, 4.302.011, 4.353.552, 4.371.746 et 4.430.917 (William Pepper, Jr), il est décrit divers procédés ou interfaces machine-homme utilisant le contact au doigt. Les modes préférés de réalisation de chacune de ces inventions manquent de vitesse et de résolution suffisantes pour permettre la reconnaissance de caractères manuscrits à l'aide d'un stylet et ils sont conçus à d'autres fins. Le brevet U.S. 4.318.096 (Pepper) enseigne l'utilisation d'un stylet conducteur. Ce brevet relève du domaine du dessin graphique et permet de faire varier la largeur et l'intensité du trait en appliquant une

certaine pression sur le stylet, les résultats étant présentés sur un écran cathodique classique. Le brevet U.S. 3.699.439 (Turner) et le brevet U.S. 4.055.726 (Turner et autres) présentent deux procédés de détection électronique de position utilisant une sonde.

5 L'invention a pour but de fournir un procédé et un appareil perfectionnés permettant de disposer d'un ordinateur sans clavier sur lequel les fonctions habituelles de l'ordinateur s'exécutent en écrivant d'une manière normale, à l'aide d'un stylet en forme de crayon sur un écran d'entrée placé directement par-dessus un visuel plat.

10 L'ordinateur sans clavier obtenu présente une configuration idéale pour être utilisé par des personnes qui ne sont pas spécialisées dans l'usage d'un clavier, par des personnes qui le sont et pour qui le caractère polyvalent de l'ordinateur se trouve accru, et dans divers contextes et applications dans lesquels une saisie par clavier est complexe ou impossible.

15 L'invention a également pour but de fournir des moyens à l'aide desquels on puisse charger dans un dispositif portable des informations informatisées et des logiciels d'application, afin de pouvoir, postérieurement, prendre connaissance des textes et données et les traiter, ainsi qu'ajouter de nouveaux textes et données suivant un mode normal de tracé manuscrit.  
20 Par la suite, l'utilisateur peut transmettre ces textes informatisés à un autre ordinateur, à un dispositif analogue, à un dispositif de stockage électronique extérieur, à une imprimante de recopie, ou en utilisant l'intermédiaire d'un système de télécommunications.

25 Un autre but de l'invention est de permettre de disposer d'un ordinateur capable de reconnaître des Signes Manuscrits avec un degré élevé de précision et d' « acquérir » des styles individuels d'écriture.

L'invention a encore pour but de permettre de disposer d'un ordinateur sans clavier et portable dans lequel l'introduction des données et des instructions s'effectue en utilisant un stylet.

30 L'expression « Signes Manuscrits » signifie ici tous les signes qui peuvent être tracés d'une manière manuscrite et faire l'objet d'une communication. C'est ainsi que, à titre d'exemples, mais sans caractère limitatif, les chiffres, les lettres, les signes « Kanji » (idéogrammes japonais) ou autres signes linguistiques, les signes d'édition et les signes d'ingénierie,  
35 scientifiques, d'architecture et mathématiques sont des Signes Manuscrits. D'autres exemples de Signes Manuscrits sont les tracés à main levée ou les signatures, ou tout autre information écrite de ce type dont la configuration propre est donnée par la personne particulière qui les trace. Comme Signes Manuscrits, on peut également citer les Signes d'Édition (définis ci-

dessous).

L'expression « Signes de Police de Caractères » désignera ici des signes « générés » par ordinateur qui sont visualisés sous un format préfixé de police de caractères. A titre d'exemple et sans aucun caractère limitatif, les signes alphanumériques peuvent être des Signes de Police de Caractères et être visualisés sous de nombreux formats de police. Les « idéogrammes » japonais ou chinois peuvent également être des Signes de Police de Caractères, de même que peuvent l'être les caractères d'ingénierie, scientifiques, mathématiques, ou d'architecture ou d'autres caractères de ce type. Comme autres exemples de Signes de Police de Caractères, on peut citer toute forme que l'on peut mettre en mémoire et visualiser à l'aide d'un ordinateur, par exemple le dessin d'une voiture ou d'une maison.

L'expression « Signe d'Edition » ici utilisée est un signe (tel qu'un signe de renvoi, un trait horizontal, un trait vertical court, un trait vertical allongé, etc.) qui est destiné, lorsqu'il est reconnu, à faire exécuter à l'ordinateur une Fonction particulière d'Edition (définie ci-dessous) telle que « insérer un texte » (signe de renvoi), « supprimer un texte » (trait horizontal), « supprimer une lettre » (trait vertical court) ou « déplacer une marge » (trait vertical allongé), pour ne citer que quelques exemples représentatifs.

L'expression « Fonction d'Edition » désigne toute opération d'édition de texte généré par ordinateur, comme par exemple, d'une manière non limitative, « insérer un texte », « supprimer un texte », « déplacer un texte » et « remplacer un texte ». La liste de certaines Fonctions principales d'Edition est présentée ci-dessous à la page 36.

L'invention a pour objet un dispositif d'entrée/sortie ou E/S permettant, d'une part, de fournir à un dispositif utilisateur des signaux d'entrée qui sont représentatifs d'un caractère graphique produit par un utilisateur et permettant, d'autre part, de recevoir des signaux de sortie à partir du dispositif utilisateur et de fournir une représentation visuelle de ces signaux de sortie, ce dispositif E/S étant caractérisé en ce qu'il comprend :

(a) un moyen de visualisation permettant de fournir une représentation visuelle d'un caractère graphique en réponse à des signaux de sortie fournis par le dispositif d'utilisation,

(b) un ensemble à écran d'entrée permettant de produire un train de signaux d'entrée au fur et à mesure qu'un utilisateur trace un signe ou caractère graphique, cet ensemble à écran d'entrée comprenant un écran d'entrée translucide présentant des surfaces supérieure et inférieure, la surface inférieure

rieure étant disposée par rapport à la surface de visualisation d'une façon telle que la représentation visuelle produite par ledit moyen de visualisation soit visible du côté de la surface supérieure de cet écran d'entrée, et (c) des moyens permettant de déterminer l'identité dudit caractère graphique et de fournir au moyen de visualisation des signaux de sortie représentatifs de ce caractère.

L'invention a également pour objet un appareil de reconnaissance de caractères manuscrits, caractérisé en ce qu'il comprend :

(a) un moyen de visualisation présentant une surface de visualisation permettant d'offrir une représentation visuelle de caractères graphiques en réponse à un groupe de signaux de visualisation,

(b) une surface à écrire destinée à recevoir des caractères graphiques tracés de manière manuscrite par un utilisateur, cette surface à écrire étant disposée en proximité étroite par rapport au moyen de visualisation et étant constituée d'une surface sensiblement translucide à laquelle sont associées des coordonnées de position,

(c) un moyen indicateur qu'un utilisateur fait fonctionner en liaison avec la surface à écrire de façon à écrire des signaux d'écriture en liaison avec cette surface à écrire,

(d) des moyens de détection de position permettant de convertir les signaux d'écriture en signaux de localisation représentatifs des coordonnées de position du moyen indicateur par rapport à la surface à écrire,

(e) des moyens de traitement de données destinés à recevoir et stocker les signaux de localisation au fur et à mesure qu'ils sont produits et à créer lesdits signaux de visualisation de façon que le moyen de visualisation puisse représenter graphiquement le caractère graphique après qu'il ait été produit par le moyen indicateur, ces moyens de traitement de données étant constitués par des moyens de reconnaissance de caractères permettant de comparer les signaux produits de localisation représentatifs d'un caractère graphique à un groupe de signaux de base de données mis en mémoire afin d'identifier ledit caractère graphique.

L'invention a encore pour objet un procédé de reconnaissance d'un Signe Manuscrit, caractérisé en ce qu'il consiste :

(a) à émettre un train de signaux de localisation pour des Points donnés en déplaçant, à l'endroit d'emplacements situés à proximité d'une surface à écrire, un stylet qui émet un signal représentatif de l'emplacement de ce stylet,

(b) à diviser un train de signaux de localisation en sections représentatives d'un Trait de Tracé,

- (c) à calculer des caractéristiques de Trait de Tracé,
- (d) à comparer ces caractéristiques de Trait de Tracé calculées à des caractéristiques de Trait de Tracé précédemment mises en mémoire dans une base de données, et
- 5 (e) à déterminer la meilleure comparaison et à indiquer si cette meilleure comparaison est suffisamment bonne pour être une concordance.

L'invention a aussi pour objet un appareil de reconnaissance de Signes Manuscrits et de visualisation de Signes Manuscrits et de Police de Caractères sur un écran, caractérisé en ce qu'il comprend :

- 10 (a) un écran de visualisation présentant l'aptitude graphique de représenter des Signes de Police de Caractères et d'exécuter des instructions préfixées,
- (b) des moyens que l'on tient à la main pour écrire ou tracer des Signes Manuscrits sur ou par-dessus ledit écran,
- (c) des moyens pour permettra la visualisation, sur l'écran, d'une représentation vraie des Signes Manuscrits au fur et à mesure qu'ils sont créés,
- 15 (d) des moyens de numérisation pour détecter la position des moyens tenus à la main et pour convertir celle-ci en une série de signaux électriques déterminant la position, la taille et la forme de chaque Signe Manuscrit,
- (e) des moyens pour comparer des caractéristiques préfixée de chaque Signe Manuscrit numérisé, à une base de données de caractéristiques préfixées de
- 20 Signes de Police de Caractères de façon à évaluer l'identité de chaque Signe de Police de Caractères,
- (f) des moyens pour convertir le Signe Manuscrit en un Signe de Police de Caractères préfixé ou en une instruction et pour, suivant le cas, permettre
- 25 la visualisation de ce Signe de Police de Caractères sur l'écran, à proximité étroite de la zone de cet écran sur laquelle le Signe Manuscrit a été introduit à l'origine, ou pour exécuter l'instruction.

L'invention a enfin pour objet un procédé de traitement commandé par microprocesseur et permettant la reconnaissance, le transcodage et la visualisation de Signes Manuscrits et l'exécution d'instructions, caractérisé en ce qu'il consiste :

- 30 (a) à créer une base de données personnalisée pour chaque utilisateur individuel du procédé de traitement, en faisant introduire initialement à l'utilisateur un Signe Manuscrit pour chaque caractère correspondant à une police de caractères à visualiser ou à une instruction à exécuter,
- 35 (b) à déterminer un jeu unique de caractéristiques de façon à caractériser chaque signe de ce type et à le mettre en mémoire dans la base de données ,
- (c) à écrire ou tracer des Signes Manuscrits avec un stylet sur un écran d'ordinateur,



(d) à numériser chaque Signe Manuscrit de façon à identifier les coordonnées X, Y d'une multiplicité de Points définissant le Signe,  
(e) à traiter les caractéristiques numérisées de chaque Signe Manuscrit de façon à déterminer les caractéristiques préfixées du Signe,  
5 (f) à explorer la base de données afin de trouver « un succès », une correspondance parfaite ou la correspondance la plus proche par rapport aux caractéristiques, et  
(g) à réaliser la visualisation du Signe de Police de Caractères ou à exécuter l'instruction, le ou la plus étroitement associé(e) aux caractéristiques  
10 du « succès ».

Ainsi, l'invention porte sur un système informatisé original sans clavier qui présente la capacité de reconnaître et visualiser des Signes Manuscrits et de faire visualiser à l'ordinateur des Signes de Police de Caractères, ainsi que, si on le désire, d'exécuter, rapidement, facilement et  
15 pour un coût raisonnable, des fonctions d'édition en conformité avec des Signes d'Édition.

Il peut être prévu un boîtier d'ordinateur présentant un panneau plat de visualisation sur lequel un utilisateur peut « écrire » à l'aide d'un stylet, avec la possibilité de reconnaître des Signes Manuscrits écrits sur  
20 ce panneau à l'aide de ce stylet et de les convertir en des Signes de Police de Caractères visualisés et/ou d'exécuter des Fonctions d'Édition à l'aide de ces Fonctions d'Édition, le tout avec un minimum de complexité technique pour l'utilisateur.

Egalement suivant l'invention, une fois que les informations et les logiciels d'application voulus ont été chargés dans l'ordinateur portable  
25 sans clavier, on peut utiliser ces informations et ces logiciels, et y donner suite sans que cela ne requière de compétences ou connaissances concernant les ordinateurs ou autres sources de données de l'état de la technique.

La facilité d'utilisation de la technique d'introduction de la présente invention accroît le caractère polyvalent de l'ordinateur pour les  
30 personnes auxquelles les claviers sont familiers. Le caractère portatif de la présente invention lui permet également d'être utilisée dans des applications et contextes dans lesquels des ordinateurs portatifs à clavier sont complexes, difficiles ou impossibles à utiliser. C'est ainsi par exemple que  
35 l'on peut stocker dans la mémoire de l'ordinateur portatif un très grand nombre de formulaires en blanc ou totalement ou partiellement remplis. Dans un hôpital, on peut stocker dans la mémoire de l'ordinateur portatif des « feuilles » de données concernant les malades, qu'une infirmière peut appe-

ler lorsqu'elle effectue des rondes, des informations pertinentes, telles que la pression sanguine, la température, etc., pouvant alors être introduites manuellement à l'aide d'un stylet. Ces formulaires corrigés ou complétés peuvent alors faire l'objet d'un téléchargement dans la mémoire d'un ordinateur central.

Les conditions requises en ce qui concerne la technique d'introduction par détection de position sont la précision (Point par Point), la résolution (position absolue) et la vitesse (Points par unité de temps) afin de déterminer de manière convenable le Trait de Tracé écrit en vue de l'analyse de reconnaissance. Pour l'appareil et le procédé de reconnaissance pouvant actuellement être mis en œuvre et tels que décrits ci-dessous, les conditions minimales requises actuelles sont : une précision de 0,127 mm, une résolution de 0,381 mm et une vitesse de 150 Points par seconde. Cette précision permet d'obtenir une ligne d'écriture de 6,35 mm de hauteur avec plus de 10 Points d'entrée bruts le long d'un Trait de Tracé d'une lettre minuscule. La résolution permet le positionnement du signe à deux points près sur une représentation effective à 640 points pour 22,86 cm. La vitesse permet d'obtenir environ 50 Points d'entrée bruts pour une lettre unique écrite rapidement (1/3 seconde).

Un mode de réalisation particulier de l'invention comprend un écran d'entrée transparent. Au fur et à mesure que l'utilisateur écrit sur cet écran d'entrée des signes ou caractères de type alphanumérique ou autre, le caractère est représenté sous la forme d'un train de Points simulant une introduction écrite au stylo sur du papier. Une fois que les signes ou caractères discrets de type alphanumérique ou autre sont complets, ils sont transcodés en texte d'ordinateur ou instructions d'ordinateur qui peuvent être visualisés sur un écran de visualisation en un emplacement qui est de préférence situé au-dessous de la zone de l'écran d'entrée où ils ont été introduits. Ce mode de réalisation comprend également un algorithme de reconnaissance de tracés qui permet la transduction de tout signe ou caractère écrit, par exemple des idéogrammes ou des signes scientifiques, en texte d'ordinateur.

Dans un mode particulier de réalisation, actuellement préféré, un ordinateur sans clavier conforme à la présente invention présente la configuration d'un dispositif de traitement et visualisation comprenant un écran transparent à contact et un circuit électronique associé disposés au-dessus d'un écran de visualisation de 80 colonnes sur 25 lignes ou plus, un stylet pour l'introduction des données, un microprocesseur et des moyens de mise en mémoire, un logiciel de reconnaissance de tracés/d'intelligence artifi-

cielle et un logiciel d'édition, un système d'alimentation en courant par piles, et d'autres moyens E/S.

D'autres caractéristiques et avantages de l'invention ressortiront de la description qui va suivre, à titre d'exemples non limitatifs et en regard des dessins annexés sur lesquels :

5 la Fig. 1 est un schéma-blocs du dispositif de la présente invention,

la Fig. 2 est une vue en perspective du boîtier contenant les éléments opérationnels de l'invention,

10 la Fig. 2A est un détail à plus grande échelle de la Fig. 2, des parties étant ôtées de façon à montrer la disposition relative des positions de l'écran d'entrée à contact et de l'écran de visualisation,

la Fig. 3 est une vue schématique en organigramme de l'écran d'entrée, du stylet et du circuit électronique associé,

la Fig. 4 est un schéma-blocs d'ensemble de l'appareil d'un système informatique à entrée sans clavier conforme à l'invention,

la Fig. 5 est un schéma-blocs illustrant le déplacement des données à l'intérieur du système lorsqu'elles sont modifiées à l'aide de caractères et d'instructions manuscrites,

20 la Fig. 6 est un schéma-blocs d'ensemble illustrant la hiérarchie du logiciel utilisé pour exploiter le système,

la Fig. 7 est un schéma-blocs généralisé de l'algorithme de reconnaissance de caractères et tracé.

les Fig. 8A et 8B représentent ensemble un schéma-blocs détaillé de l'algorithme de reconnaissance de tracés,

25 la Fig. 9 est un schéma-blocs du sous-programme de caractérisation des Traits de Tracé,

la Fig. 10 est une vue en plan de dessus d'un écran illustrant l'« initialisation » de la base de données prévue pour les Signes Manuscrits,

30 les Fig. 11A à 11I constituent une série de vues en plan de dessus d'écrans illustrant l'exploitation d'un système d'édition de textes,

les Fig. 12A à 12G constituent une série de vues en plan de dessus d'écrans illustrant l'exploitation d'un système de saisie de données, et

la Fig. 13 est un schéma-blocs généralisé de l'Editeur en Lignes-blocs.

35 Les figures, sur lesquelles des chiffres identiques indiquent des éléments identiques pour l'ensemble des diverses vues, illustrent, et plus particulièrement la Fig. 1, un schéma-blocs d'ensemble d'un système informatique portatif à entrée sans clavier, par tracé manuscrit 10. Ce système informatisé complet est enfermé dans un boîtier 12 représenté schématiquement par un trait interrompu et il comporte un micro-ordinateur numérique

universel classique 14 qui est décrit ci-dessous de manière plus détaillée. L'information d'entrée est fournie à ce micro-ordinateur 14 à l'aide d'un stylet 16 qui « écrit » sur un écran à écrire ou écran d'entrée 18. Ce stylet 16 (Fig. 2) est relié à l'ordinateur du système 10 par un câble 17 (Fig. 2). Au fur et à mesure que ce stylet 16 « écrit » sur l'écran d'entrée 18, un ensemble de signaux de localisation représentatifs d'un ensemble de coordonnées de position correspondantes sont transmis au micro-ordinateur 14. Ce micro-ordinateur a été programmé, conformément à un programme machine qui est décrit ci-dessous, de façon à reconnaître le train de signaux de localisation et à stocker ces signaux dans une mémoire interne. Ce micro-ordinateur programmé 14 fournit aussi un ensemble correspondant de signaux de visualisation à un écran de visualisation 20. L'écran d'entrée 18 et l'écran de visualisation 20 sont l'un et l'autre décrits ci-dessous de manière plus détaillée.

La Fig. 2 représente une vue en perspective d'un système informatisé sans clavier 10 mettant en oeuvre les caractéristiques de la présente invention. Ce système 10 est enfermé dans un boîtier 12 qui est une enveloppe parallélépipédique fermée présentant une surface supérieure inclinée 22 qui comporte une zone de visualisation multiligne à circuits intégrés 24. La Fig. 2A montre que l'écran d'entrée 18 est disposé par-dessus l'écran de visualisation 20. Dans le présent exemple, cet écran 20 permet la visualisation d'un groupe de lignes horizontales 25 comportant les informations suivantes :

Nom	_____
Adresse	_____
	_____

Les introductions manuscrites se font au-dessus de chaque ligne 25. L'espace ou distance séparant deux lignes 25, noté 26, est utilisé par le système pour normaliser toutes les distances, tandis que les lignes 25 elles-mêmes servent d'axe de référence ou de ligne de base.

Une section d'entrée à touches 27, située au-dessous de la zone de visualisation 24 de la surface supérieure 22, est constituée d'un groupe de « Touches Programmables » 28. L'utilisateur peut programmer ces touches 28 à tout usage voulu, par exemple pour introduire des instructions machine. Comme exemples d'instructions pour ces Touches Programmables 28, on peut citer « mémoriser », « rappeler » et « effacer ». On peut en outre utiliser ces touches 28 pour passer d'un programme à l'autre ou passer d'un mode à l'autre (par exemple du mode d'entrée de données au mode d'édition). Tou-

tefois, ces Touches Programmables 28 sont facultatives et on les utilise pour compléter l'introduction obtenue en traçant manuellement les entrées. Le stylet 16, que l'on utilise pour écrire des instructions et données d'entrée dans la zone de visualisation 24, s'utilise également pour activer la Touche Programmable sélectionnée 28. Un interrupteur MARCHÉ-ARRÊT 30 est disposé sur le côté du boîtier 12 à proximité des touches 28. Un connecteur de périphérique ou de sortie de données 31 est disposé sur le haut du côté droit du boîtier 12.

L'écran d'entrée 18 peut être un écran à contact du type résistif classique dans lequel une tension est appliquée sur les côtés de l'écran, tandis qu'un stylet détecte la tension présente à l'emplacement touché. La surface à écrire est un matériau transparent, d'une manière typique du verre, revêtu d'une couche conductrice mince et uniforme (dans le cas présent de l'oxyde d'indium et d'étain déposé par métallisation sous vide). Des bus-barres ou bandes conductrices verticales (non représentées) sont utilisées le long des deux côtés pour appliquer la tension de référence permettant de déterminer les coordonnées « X » de la position du stylet, tandis que des bus-barres ou bandes conductrices horizontales (non représentées) sont utilisées le long de l'extrémité supérieure et de l'extrémité inférieure pour appliquer la tension de référence qui permet de déterminer les coordonnées « Y » de cette position du stylet. Dans ce mode de réalisation, le stylet 16 est tout simplement une sonde électrique qui, lorsqu'elle est matériellement au contact de la couche conductrice, détecte la tension locale au Point de contact, tension qui varie avec la distance par rapport aux bandes conductrices ou bus-barres. Si l'origine est au Point d'application de la tension, les coordonnées X et Y sont inversement proportionnelles à la tension appliquée. Le stylet 16 doit réaliser un bon contact pour rendre minimale la résistance supplémentaire qui abaisserait la tension détectée et ajouterait de la sorte un incrément erroné de distance. Dans un mode de réalisation actuellement préféré, on utilise un embout en graphite doux. Un fil métallique, tel que le fil 17 de la Fig. 2, sert à conduire la tension du crayon jusqu'à un convertisseur analogique/numérique en vue d'une utilisation dans les calculs qui seront décrits ci-dessous. Le stylet peut être un « crayon » chargé comme décrit ici, un crayon lumineux comme cela est bien connu dans la technique, ou tout autre dispositif tenu à la main qui peut tracer sur un écran des Signes Manuscrits.

Un exemple d'un écran électrostatique classique est présenté dans le brevet U.S. 4.318.096 (Pepper) précité. Cet écran du type résistif a comme

avantage de rendre minimale l'interférence provoquée par un contact de la main de l'utilisateur avec l'écran.

La détection de position à la fois horizontale et verticale s'obtient en commutant de manière alternative la tension appliquée sur une couche conductrice, entre les paires de bus-barres horizontales et verticales, à l'aide d'un ensemble interface-multiplexeur commandé à l'aide d'un micro-ordinateur ou micro-contrôleur. Dans un écran d'entrée ou à contact disponible dans le commerce, les bus-barres sont divisées en une série de bandes courtes comportant des diodes, afin d'empêcher que les bandes horizontales court-circuitent les bandes verticales et vice-versa. Cette technique est utilisée dans un écran à contact disponible dans le commerce auprès de Touch Technologies de Annapolis Maryland, USA et d'Elographics de Oak Ridge, Tennessee, USA.

La Fig. 3 illustre de manière plus détaillée une variante de réalisation d'un écran d'entrée par détection de position et à faible puissance 33 qui est d'un type nouveau. Cet écran d'entrée 33 sert également à déterminer une position X, Y sur une plaque à résistivité électrique 34. On utilise, pour toucher cet écran 34, et appliquer une tension à la position touchée, un stylet 35 contenant une source de courant, telle qu'une pile 36, ou présentant une tension qui lui est transmise à partir d'une source extérieure telle que la source de courant du système. Lorsque le stylet 35 charge la position touchée suivant une tension positive par rapport à un groupe de Points de mesure 37 de la plaque, les tensions présentes en ces Points varient avec la distance jusqu'à la position du crayon, par exemple la position  $X_1, Y_1$ , indiquée en 38. Ces tensions sont mesurées de manière séquentielle suivant les directions X et Y en utilisant des moyens classiques tels que ceux décrits dans les brevets précités de la technique antérieure. Sur la Fig. 3, ces moyens sont constitués par un ensemble interface/multiplexeur 42 classique. Un convertisseur analogique/numérique 43 classique convertit les tensions détectées en un signal numérique. Un micro-contrôleur 44 reçoit ce signal numérique, exécute des contrôles standards pour s'assurer que la valeur numérique du signal est « correcte » (par exemple à l'intérieur de la gamme possible de tensions), puis convertit les tensions en distances X et Y de la manière ici décrite. Le micro-contrôleur 44 est classique, mais pourrait être remplacé par un ordinateur système. Ce micro-contrôleur 44 fournit sur la connexion de sortie 46 un signal numérique qui est représentatif des distances X et Y jusqu'au Point de mesure 38. La connexion 46 peut être une connexion RS 232 classique. En variante, le micro-contrôleur 44

pourrait transcoder le Point  $X_1, Y_1$  en n'importe quel autre Point de référence, tel qu'un Point situé sur la ligne de base 25 (Fig. 2).

5 Tant qu'il n'existe pas de contact du stylet 35 sur la position 38 ou toute autre position de la plaque 34, aucun courant ne passe et la consommation de courant est minimale. Une mesure fortuite de la tension aux Points de mesure peut survenir, si l'on utilise une tension en dent de scie et une synchronisation, lorsque la tension du Point de mesure excède une tension préfixée d'entretien.

10 Le cadre de l'invention recouvre les options suivantes pour les écrans d'entrée ou à contact 18 et 33 : la plaque résistive 34 ou son équivalent pour l'écran 18 peut être transparent ou translucide et le Point de position peut être réalisé à l'aide d'un stylet ou d'un doigt de l'utilisateur, ou encore d'un Point de connexion d'un écran conducteur de recouvrement (comme par exemple les écrans à contact disponibles dans le commerce  
15 auprès de Touch Technology, Annapolis, Maryland, USA. Les écrans d'entrée 18 et 33 peuvent être constitués par une surface matérielle solide qui est transparente ou translucide ou être du verre ou une matière plastique telle que du Mylar. Cette surface peut être revêtue d'une substance conductrice, résistante comme de l'oxyde d'indium et d'étain. D'autres surfaces matériel-  
20 les peuvent utiliser la transmission du rayonnement électromagnétique ou acoustique entre la position touchée et un ou des Points de référence, la distance étant déterminée par le temps de propagation ou le déphasage. En variante, les écrans d'entrée 18 et 33 peuvent utiliser une surface virtuelle ou géométrique déterminée par un champ électromagnétique, optique ou acous-  
25 tique.

On peut réaliser la détection de position à l'aide de la fermeture d'un contact électrique par un couplage résistant, capacitif ou inductif, une détection à distance à l'aide de champs acoustiques, électriques ou magnétiques ou à l'aide d'un balayage lumineux (UV, IR ou en microondes).

30 Les avantages de l'introduction par détection de position et à faible puissance de l'invention par rapport à d'autres écrans de ce type sont : 1) le fait que l'invention rend minimale la puissance nécessaire en attente, 2) le fait que l'invention élimine la distorsion due à des bus-barres parallèles en regard dans les écrans classiques à contact, et 3) le fait que, lorsqu'on utilise une tension en dent de scie, l'invention élimine le besoin  
35 d'une puce A/N qui est un facteur important de coût dans les écrans à contact de la technique.

Il est souhaitable que l'on choisisse le coefficient de frottement de l'écran 18 de façon qu'il soit suffisamment « rugueux » pour offrir une

certaine résistance au déplacement du stylet 16 sur cet écran. Si l'écran était trop lisse, le stylet glisserait trop facilement et serait difficile à contrôler.

La Fig. 4 illustre un schéma-blocs d'ensemble du système avec les principaux circuits électroniques utilisés dans le mode préféré de réalisation de l'invention. Le micro-ordinateur 14 comprend un microprocesseur 50 qui est interconnecté à plusieurs autres éléments électroniques à l'aide d'un chemin de données ou bus de données 52. Il est relié à ce bus 52, en vue d'une circulation de données dans les deux sens, une mémoire morte (ROM) 54 qui est programmée à l'aide de programmes d'exécution et d'application et une mémoire vive (RAM) 56 alimentée par batterie. Le microprocesseur 50 peut être un dispositif classique à huit bits ou seize bits monopuce qui fonctionne de façon à exécuter les programmes fixes de gestion résidant dans la ROM 54, et qui reçoit en outre des programmes de gestion en provenance des autres éléments électroniques par le bus 52 et fournit des signaux de commande à ces autres éléments par ce bus. Ce microprocesseur 50 peut être du type de désignation commerciale Z80 (fabriqué par Zilog Microcomputers de Cupertino, Californie, USA), d'un type de dispositif 8088 (fabriqué par Intel Corp. de Santa Clara, Californie, USA), ou tout autre microprocesseur analogue ou plus puissant. La ROM 54 peut être du type 2564 ou 4764, l'une et l'autre fabriquées par Texas Instruments de Dallas, Texas, USA). La capacité de mémoire de la RAM 56 est déterminée partiellement par la taille des programmes d'application, par le programme d'exécution et par la base de données. Comme cela sera discuté plus loin, la RAM 56 peut être du type statique SRAM ou du type dynamique DRAM. Les conditions essentielles requises de la RAM 56 sont qu'elle présente une capacité suffisante de mémoire et qu'elle demande un minimum de consommation pour l'introduction.

Une batterie 58, telle qu'une batterie au lithium, fournit la puissance nécessaire pour rendre rémanente la mémoire de la RAM 56 sur des périodes prolongées de temps. Un groupe de batteries 60 contenant les types rechargeables bien connus de batteries est utilisée pour fournir les divers niveaux de tension requis par les autres éléments électroniques du micro-ordinateur 14.

En variante, la fonction de mise en mémoire de la RAM 56 peut être prise en charge par un dispositif rémanent qui ne demande pas de puissance pour conserver la mémoire, comme par exemple une mémoire effaçable et reprogrammable électroniquement (EEPROM) ou des dispositifs utilisant les bulles magnétiques ou la capacitance. On peut également utiliser les disques ou rubans de l'état de la technique en vue d'une mise en mémoire de grande



capacité. Comme dispositifs convenables à mémoire à bulles, on peut citer les types 7110 et 7114 qui ont des capacités de mémoire de respectivement 1 mégabit et 4 mégabits (l'un et l'autre sont fabriqués par Intel Corp.). Il est par ailleurs possible d'utiliser une monopuce à circuits intégrés qui  
5 comprenne le microprocesseur 50, au moins une partie de la ROM 54 et au moins une partie de la RAM 56.

Il est également relié au bus 52 une interface 62 de série EIA RS-232 qui fournit un moyen permettant d'entrer et de sortir des données. Les données sont fournies au bus 52 (couramment à la RAM 56) en interconnectant une source extérieure de données sur la connexion RS-232 62, directement sur le microprocesseur 50 et les autres éléments du micro-ordinateur  
10 14. Le déchargement des données à partir de la RAM 56 peut également s'effectuer à l'aide du microprocesseur 50 à destination d'un ordinateur extérieur, d'un autre dispositif de collecte de données, d'un dispositif de mise en mémoire de données de grande capacité (par exemple des unités de disquette et de disque rigide), ou encore d'un système électronique de télécommunications. D'une manière analogue, à l'aide de la connexion 62, on peut  
15 communiquer des données à une imprimants (non représentée) à partir du bus d'interconnexion 52.

On utilise le stylet 16 pour écrire sur un écran d'entrée 18 et pour provoquer la création d'informations constituées par des coordonnées X, Y à l'aide d'un circuit électronique classique d'interface à écran à contact. Les informations constituées par les coordonnées sont communiquées par le bus 52 en vue d'une utilisation de vérification de la part du système 10.  
20 L'affichage à circuits intégrés 20, qui est constitué par un affichage multiligne (à titre d'exemple 80 colonnes sur 25 lignes), est interconnecté au bus 52 à l'aide de l'interface de visualisation 66. Les conditions fondamentales requises du visuel sont qu'il soit sensiblement plat et suffisamment mince pour être utilisé dans la présente invention. Ce visuel peut être  
25 de l'un des types suivants : les types à balayage tels que par exemple un type cathodique, les types à projection tels que par exemple un projecteur par l'arrière, les types à réseau de Points émettant de la lumière (par exemple électroluminescents ou à décharge plasma) et les types à réseau de Points d'arrêt de la lumière (par exemple visuels à cristaux liquides, PLIZ  
30 à circuits intégrés ou magnéto-optique). Il est par ailleurs préférable que ce visuel soit compatible avec l'écran d'entrée 18 en ce qui concerne la taille, la configuration et la transparence et que l'un et l'autre soient du type à faible consommation de courant.

Les coordonnées X, Y considérées dans l'invention servent d'entrée

pour l'ordinateur sans clavier 14 à l'aide du circuit électronique d'interface 64 constitué par un écran d'entrée et elles sont communiquées par l'intermédiaire du bus 52 au microprocesseur 50 qui exécute les programmes mis en mémoire dans la ROM 54 et la RAM 56.

5 Le nombre de Points (c'est-à-dire de jeux de coordonnées X, Y) utilisés pour définir chaque Signe Manuscrit et la vitesse à laquelle ces Points sont identifiés sont importants pour le caractère polyvalent pratique de l'invention. Il est souhaitable d'utiliser au moins environ 4 Points par centimètre et au moins environ 100 Points par seconde pour définir les Signes Manuscrits. On remarquera que plus on identifie de Points par centimètre, plus est grande l'exactitude du système dans l'identification des Signes Manuscrits, mais que par contre l'identification d'un plus grand nombre de Points ralentira la vitesse d'identification et exigera une plus grande capacité de mémoire d'ordinateur. On devra par conséquent rechercher un compromis basé sur la taille (mémoire disponible et capacité de traitement) du système informatisé et sur les conditions requises de vitesse de réponse et de précision. Pour la plupart des applications, on se satisfera de valeurs standards se trouvant dans la gamme d'environ 4 Points par centimètre et par seconde à environ 8 Points par centimètre et par seconde.

10  
15  
25 On remarquera également que plus la précision du système dans l'identification des coordonnées X, Y de chaque Point est grande, plus est faible le nombre de Points qui doivent être identifiés par centimètre et par seconde pour identifier de manière exacte des Signes Manuscrits. Inversement, plus la précision est faible, plus est important le nombre de Points dont il est besoin.

30 La résolution des Points est nécessaire pour placer des Points là où on le désire, par exemple pour écrire de manière précise un signe d'édition entre deux caractères. D'une manière idéale, il est souhaitable que la résolution porte sur un élément unique d'image de visualisation (« point de visualisation »). Toutefois, sur le plan de l'exploitation, une résolution dans la limite de deux points de visualisation est suffisante pour un visuel comportant 640 points sur une ligne horizontale de balayage de 22,9 cm.

35 Lorsque l'interrupteur 30 (Fig. 2) est placé sur « sous tension », c'est le mode de visualisation de base qui est activé et le micro-ordinateur 14 (Fig. 4), qui est programmé par le système d'exécution, provoque la visualisation d'un menu sur l'écran de visualisation 20 (Fig. 1). Ce menu présente diverses options de logiciel. Une fonction essentielle de logiciel, l'édition, fonctionne d'une manière analogue au logiciel classique de traitement de texte, la différence résidant dans le fait que des caractères, si-

gnes et instructions manuscrits sont interprétés par le système comme s'ils étaient introduits à l'aide d'un clavier classique. Le système est capable d'apprendre (acquérir) les signes d'édition utilisés dans l'écriture par une personne particulière pour des fonctions telles que rentrer, insérer, effacer, déplacer et reformater et il transcode ces signes en fonctions d'instruction numérique. D'une manière facultative, des Touches Programmables 28 (Fig. 2), activées en touchant ces zones de l'écran d'entrée à l'aide du stylet 16, fonctionnent comme des touches rigides de fonction classiques d'un clavier d'ordinateur.

La présente invention est particulièrement adaptée pour être utilisée en tant qu'éditeur ou machine de traitement de texte à écran interactif. Une fois qu'une personne désirant écrire a eu accès à un document, par exemple en touchant, à l'aide du stylet, le nom affiché d'un fichier existant ou en écrivant sur l'écran le nom de ce fichier, on peut exécuter toutes les fonctions habituelles d'édition en utilisant le stylet comme entrée. Lorsque l'utilisateur souhaite modifier un caractère ou signe visualisé, il peut tout simplement écrire par-dessus ce caractère ou signe visualisé et, comme cela sera décrit ci-dessous, l'algorithme de reconnaissance de tracés transcodera l'entrée écrite en texte machine. C'est ainsi par exemple que le logiciel d'édition permet d'éliminer du texte tout simplement en traçant une ligne à travers lui et qu'un signe classique de renvoi peut être utilisé pour passer du mode d'exécution au mode d'insertion. Dans ce mode d'insertion, l'écran de visualisation 20 offre un espace libre supplémentaire permettant l'introduction de caractères ou signes manuscrits qui se trouvent insérés dans le texte après le Point où on a écrit un signe de renvoi. On peut déplacer du texte tout simplement en plaçant des parenthèses, ou autres éléments de délimitation définis par l'utilisateur, autour d'une phrase ou d'un mot visualisé et en écrivant un signe de renvoi ou autre signe défini par l'utilisateur dans la zone du texte où l'utilisateur souhaite qu'apparaisse ce contenu. On peut fixer de nouvelles marges en traçant des lignes verticales sur le côté jusqu'au bas du texte visualisé, à l'endroit où devront apparaître les nouvelles marges.

Le logiciel éditeur de base permet également de créer de nouveaux documents tout simplement en écrivant des Signes Manuscrits sur l'écran. Tous les documents peuvent être mis en mémoire, modifiés et communiqués de la manière dont ces fonctions s'accomplissent sur un système classique de traitement de texte, à la différence du fait que ces fonctions s'accomplissent à l'aide de Signes d'Édition manuscrits tracés sur l'écran (facultatif) ou en touchant à l'aide du stylet les Touches Programmables. On peut

par la suite, par l'intermédiaire de la connexion RS 232 62 (Fig. 4), décharger le texte composite ainsi produit et mis en mémoire, à destination d'un autre ordinateur, d'un dispositif analogue, d'un dispositif extérieur de collecte de données ou d'un dispositif d'enregistrement, sur une imprimante ou par l'intermédiaire d'un système de télécommunications.

Outre ces modes principaux de fonctionnement, le caractère universel de la présente invention se trouve accru par un certain nombre d'éléments et particularités auxiliaires. Pour favoriser l'introduction de caractères alphanumériques, on peut brancher sur un interface classique de clavier (non représenté) un clavier alphanumérique classique (non représenté) contenant un jeu complet de touches. On peut également utiliser un connecteur de courant CA/CC dans les applications où le caractère portatif n'est pas nécessaire et où il est besoin de respecter des conditions déterminées de courant pour certaines techniques d'écran telles que des visuels à décharge gazeuse et des visuels électroluminescents.

Dans une utilisation pratique, l'ordinateur sans clavier peut fonctionner pour n'importe quelle application ou dans n'importe quel contexte dans laquelle ou lequel des entrées manuscrites transcodées en texte machine sont utiles ou nécessaires. A titre d'exemple, le dispositif peut servir de nouvelle génération de machine de traitement de texte ou être utilisé dans des domaines tels que la vente, les soins de santé, le contrôle de stocks, les recensements, le règlement des indemnités, pour ne mentionner que quelques-unes des nombreuses utilisations de l'invention, ou encore en tant que dispositif général d'enseignement et de contrôle des connaissances dans le domaine de l'éducation. Etant donné que le logiciel de reconnaissance de tracés peut réaliser une acquisition et un transcodage en texte machine à partir de langages qui ne sont pas constitués d'une petite série ou d'une série limitée de caractères alphanumériques (comme par exemple le japonais, le coréen et le chinois), il présente une utilité particulière pour le traitement de texte et les télécommunications dans ces langues.

Dans la mise en oeuvre de l'invention, il est particulièrement avantageux d'utiliser un écran unique d'ordinateur pour visualiser n'importe quels formulaires initiaux, n'importe quels Signes de Police de Caractères ou autres visuels à éditer, et de créer une « fenêtre » toute proche d'espace blanc où des Signes Manuscrits doivent être écrits, visualisés et identifiés et où doivent être visualisés les Signes de Police de Caractères correspondant à ces Signes Manuscrits. De cette manière, l'utilisateur peut voir le texte en cours d'édition et l'insertion ou modification proposée sans déplacement notable de la tête et des yeux (ou sans aucun déplacement). Un exem-

ple en est donné sur les Fig. 11A à 11D. Cette particularité de l'invention (proximité, sur un écran, d'un texte à éditer et de la fenêtre dans laquelle un nouveau texte doit être écrit à la main) est très importante en ce qui concerne l'utilisation simple, rapide et aisée de l'invention.

5 Dans un mode préféré de réalisation de l'invention, le système « acquiert », avant une utilisation effective, le style d'écriture d'un utilisateur particulier. C'est ainsi par exemple que, si on utilise l'alphabet ro-  
main, on devra introduire dans la base de données les vingt-six lettres de  
10 l'alphabet et les chiffres de 0 à 9. On pourrait également introduire des signes de ponctuation, comme par exemple le point, la virgule, le point d'interrogation, les deux points, le point-virgule, le trait d'union et autres signes. Il n'existe pratiquement pas de limite aux Signes Manuscrits qui  
15 peuvent être reconnus et mis en mémoire dans la base de données. Bien entendu, l'ordinateur devra mettre en mémoire un jeu convenable de Signes de Police de Caractères en vue de la conversion des Signes Manuscrits. On pourrait créer différents jeux de Signes de Police de Caractères et les ranger dans la mémoire permanente de l'ordinateur, comme par exemple dans la  
20 puce ROM 54. C'est ainsi par exemple que, dans l'utilisation en langue anglaise, une puce pourrait contenir une ou plusieurs polices de caractères pour les chiffres et les lettres, les signes convenables de ponctuation et les symboles mathématiques appropriés. Il pourrait être mis en mémoire dans  
d'autres puces des Signes de Police de Caractères pour les alphabets arabes, cyrilliques ou grecs, le japonais, le chinois ou le coréen « Konji », pour  
des symboles utilisés par les architectes ou les ingénieurs, ou pour des symboles chimiques (par exemple le noyau benzénique et autres).

Sur la Fig. 10, c'est l'un d'une série d'écrans d'acquisition qui est visualisé et il est suggéré (« prompt » sur la figure) à l'utilisateur  
d'écrire les chiffres 0 à 4. L'ordinateur tentera de faire coïncider (par  
30 conversion) les chiffres écrits avec la base de données existante (s'il en existe une). Si la coïncidence (par conversion) ne peut pas être réalisée étant donné qu'il n'existe pas de base de donnée ou du fait que la coïnci-  
dence est médiocre avec une base de données existante, le caractère est  
ajouté à cette base de donnée. Ce procédé d'acquisition se poursuit jusqu'à  
35 ce que l'on ait introduit dans la base de données l'ensemble des signes et caractères de type alphanumérique (ou autre) à utiliser. Le système a la capacité de mettre en mémoire de multiples bases de données de caractérisa-  
tion de Traits de Tracé pour les systèmes utilisés par plus d'un utiliza-  
teur. L'existence d'une base de données de caractérisation de Traits de Tra-  
cé unique pour chaque utilisateur a l'avantage supplémentaire de ne pas fai-

re intervenir l'angle d'écriture. Il en résulte que l'invention est adaptable à tous les styles d'écriture et est utilisable par des droitiers et des gauchers. On peut souhaiter incorporer dans l'appareil de l'invention une particularité permettant d'accepter les gauchers et les droitiers. Cette particularité réside dans un logement (non représenté) qui est prévu pour le connecteur de stylet sur l'un et l'autre côté du boîtier 12, si bien que l'on peut brancher le stylet 16 sur le côté gauche pour les gauchers et sur le côté droit pour les droitiers.

La Fig. 10 fournit également un exemple de l'utilisation de « Touches Programmables ». Outre la ligne d'entrée, on y voit diverses Touches Programmables. Chaque Touche Programmable correspond à une fonction qui peut être exécutée par le système. Pour exécuter cette fonction, il suffit que l'utilisateur touche le Point indiqué avec le crayon. La Touche Programmable apparaît alors en vidéo inverse et la fonction sélectionnée est exécutée. Les Touches Programmables présentent de nombreux avantages sur les touches classiques de fonction. Parmi les avantages les plus significatifs, on peut citer le fait que l'utilisateur n'est plus astreint à mémoriser quelle touche de fonction remplit quelle fonction, le fait que le besoin de recouvrements de clavier est éliminé et le fait que l'on peut disposer de Touches Programmables différentes (pour les visualiser et les rendre opérationnelles) en différents Points à l'intérieur d'un programme.

Les figures 11A à 11I illustrent certaines des simplifications qui sont rendues possibles dans le traitement de texte en utilisant la présente invention. Sur la Fig. 11A, c'est un écran standard de texte qui est visualisé. L'utilisateur du système d'entrée sans clavier décide qu'une information supplémentaire a besoin d'être ajoutée et trace un signe d'insertion (par exemple un signe de renvoi) à l'emplacement voulu sur l'écran. Une « fenêtre » d'entrée de données apparaît alors (Fig. 11B). On y inscrit le texte sous forme de Signes Manuscrits (Fig. 11C), il fait l'objet d'une recherche de coïncidence (en étant converti en Signes de Police de Caractères) (Fig. 11D), puis est inséré (Fig. 11E). L'utilisateur revient sur sa décision concernant l'adjonction et trace une ligne horizontale d'un bout à l'autre du contenu nouvellement ajouté (Fig. 11F). Celui-ci se trouve immédiatement effacé (Fig. 11G). L'utilisateur décide ensuite qu'une marge de droite plus large conviendrait mieux pour le texte. Il trace une ligne verticale sur l'écran (Fig. 11H) et la marge se trouve automatiquement ajustée (Fig. 11I).

Un schéma-blocs généralisé du traitement d'édition est présenté sur la Fig. 13 et une description de cette figure sera donnée plus loin.

Les figures 12A à 12G illustrent la manière dont on peut utiliser un formulaire en blanc pour un malade en milieu hospitalier. L'utilisateur du système appelle tout d'abord le formulaire en blanc approprié (Fig. 12A). Cela peut par exemple se faire en touchant une Touche Programmable correspondante. On touche ensuite avec le crayon la zone où doit être insérée l'information, un relevé du pouls dans le cas présent (Fig. 12B). Après que l'emplacement voulu ait été mis en évidence, une « fenêtre » apparaît directement au-dessous de l'espace libre où le relevé doit être enregistré (Fig. 12C). L'infirmière touche alors à l'aide du crayon la case de coïncidence qui apparaît mise en évidence (Fig. 12D). Le logiciel procède alors à une recherche de coïncidence entre l'introduction manuscrite et les Signes correspondants de Police de Caractères et visualise le résultat (Fig. 12E). S'il existe une coïncidence précise, on touche le bloc « inséré » (Fig. 12F) et le nouveau relevé se trouve ajouté au dossier du malade (Fig. 12G). Il est évident que l'on peut appliquer ce mécanisme à une large variété de « formulaires en blanc » dans lesquels on introduit ou corrige des données. C'est ainsi par exemple qu'on pourrait l'utiliser pour corriger ou mettre à jour des informations financières dans un programme de calcul de tableaux financiers. Toutes les applications de ce type sont du ressort de la présente invention. On peut enregistrer de la même manière d'autres informations.

C'est pour faciliter la vérification de l'exactitude du caractère introduit que l'on utilise un fond noir et des lettres blanches pour des Signes de Police de Caractères nouvellement introduits. Bien que cela soit préférable, il ne s'agit pas là d'un élément essentiel et un fond blanc et des lettres noires sont également acceptables.

La possibilité de créer une fenêtre et des données d'entrée sur le même écran que le texte à éditer ou l'espace vide destiné aux données à introduire, et au voisinage matériel de ce texte ou cet espace vide, constitue une particularité importante de la présente invention étant donné qu'elle autorise une utilisation aisée et rapide de celle-ci. L'oeil de l'utilisateur peut réaliser sa mise au point sur l'espace vide où les données doivent être insérées, tandis que la possibilité de visualiser simultanément les Signes Manuscrits et les Signes correspondants de Police de Caractères lui permet de voir facilement les erreurs lorsque le système « commet une erreur de lecture » sur un Signe Manuscrit, et de corriger alors rapidement et facilement ces erreurs.

On va maintenant décrire, en se reportant tout d'abord à la Fig. 5, l'exécution et le fonctionnement d'ensemble du logiciel de reconnaissance de tracés. Lorsque le système d'exploitation appelle le programme de reconnais-

sance de tracés, ce programme démarre au premier pavé 75 où un certain nombre de variables et de compteurs sont initialisés. Le logiciel progresse alors jusqu'au losange de décision 76 où le programme détermine si le stylet 16 (Fig. 2) est au contact de l'écran d'entrée 18 (Fig. 2A). Le système délivre un signal « crayon posé », comme indiqué et fournit également les tensions de coordonnées X, Y en tant que signaux de localisation, comme décrit plus haut et comme représenté dans le pavé de traitement 18. En utilisant le logiciel conforme à la présente invention, le micro-ordinateur 14 (Fig. 4) convertit ces signaux de localisation de coordonnées X, Y en caractéristiques de Trait de Tracé en utilisant des programmes rangés dans la ROM 54 (Fig. 4), cette conversion pouvant par contre être effectuée par un micro-ordinateur séparé tel que le microcontrôleur 44. Si un signal de crayon posé est reçu, le logiciel progresse jusqu'au pavé de traitement 80 où les signaux individuels de localisation sont combinés en « Traits de Tracé », un Trait de Tracé se définissant comme étant l'ensemble des signaux de localisation de Point délivrés entre un signal de « crayon posé » et un signal de « crayon relevé ».

Le système calcule alors pour chaque Point une transformée telle que décrite ci-dessous et qui transforme les coordonnées de Point en les faisant passer du système de coordonnées cartésiennes X, Y à un système de coordonnées de relation. Le logiciel s'avance ensuite jusqu'au pavé de traitement 82 où il compare le Trait de Tracé aux Traits de Tracé précédemment introduits qui sont accumulés dans une base de données et il détermine si ce Trait de Tracé est représenté par un signe présent dans cette base de données. Si la coïncidence est constatée (si le Signe de Police de Caractères représenté par les Traits de Tracé est reconnu), comme indiqué dans le losange de décision 84, le microprocesseur 50 (Fig. 4) fait envoyer ce signe sur l'écran de visualisation 20 (Fig. 4), comme indiqué dans le pavé de traitement 86. Si la coïncidence n'est pas constatée, le microprocesseur 50 (Fig. 4) fait visualiser un message tel qu'indiqué dans le pavé de traitement 88 et qui requiert une nouvelle introduction à l'aide du stylet sur l'écran d'entrée 18 (Fig. 4) en faisant clignoter une entrée qui soit est voisine d'une coïncidence, soit est un signe de non-reconnaissance.

Comme indiqué plus haut, le logiciel compare les caractéristiques de Trait de Tracé de chaque Signe Manuscrit aux entrées de données précédemment mises en mémoire dans une base de données. Dans un mode préféré de réalisation, la base de donnée est agencée en sections de caractères ou signes en fonction du nombre de Traits de Tracé nécessaires pour réaliser le caractère ou signe. A l'intérieur de chaque section, les entrées sont tout



d'abord disposées de manière aléatoire, mais, après utilisation, comme expliqué par la suite, les entrées les plus fréquemment utilisées « montent » vers le haut de la base de données. On remarquera que chaque utilisateur a son propre style particulier pour écrire un Signe Manuscrit et que chaque Signe Manuscrit peut présenter un certain nombre de variations différentes.

C'est ainsi par exemple que de nombreuses personnes écrivent la lettre minuscule « h » en utilisant un Trait de Tracé unique. Elles le font en commençant à déplacer le crayon sur la tablette d'écriture en un Point où elles désirent situer l'extrémité supérieure de la lettre, en traçant une ligne verticale vers le bas jusqu'à la ligne de base, puis, sans retirer le crayon du papier, en revenant jusqu'au Point milieu de la ligne verticale précédemment tracée, en tournant vers la droite et en descendant jusqu'à la ligne de base où elles retirent alors le crayon du papier. D'autre part, ces mêmes personnes peuvent tracer la lettre majuscule « H » en utilisant deux Traits de Tracé. Elles le font en traçant la barre verticale de gauche et la barre horizontale comme on le fait pour la minuscule « h », en retirant le crayon de la tablette, puis en traçant la barre verticale de droite. L'Annexe I présente les données correspondant aux Points de données de Trait de Tracé de ces deux lettres sous la forme où ces données sont mises en mémoire après avoir été produites suivant un mode de réalisation de la présente invention.

Comme le montre cette Annexe I, la lettre « h » tracée à un moment particulier par un utilisateur présente un Trait de Tracé ( $n_s = 1$ ) comportant 20 Points ( $n_p = 20$ ) et des caractéristiques de coordonnées x et y correspondant aux valeurs normalisées de minimum, de moyenne et de maximum (1/80ème d'une largeur de ligne) qui sont les suivantes : respectivement -17 et -6, 0 et 18, et 19 et 60. Les valeurs de la première colonne verticale sont les pentes de Point à Point, normalisées à  $360^\circ/256$ . Les valeurs de la seconde colonne verticale sont les positions verticales moyennes Point par Point au-dessus de la ligne de base, normalisées à 1/80ème de la largeur de ligne. Une largeur typique de ligne est d'environ 10,16 mm.

Si l'on se reporte maintenant à la Fig. 6, on y voit illustrée une hiérarchie des programmes de logiciel. A la partie supérieure, supervisant l'exploitation entière du système informatisé 10 (Fig. 1), il est prévu un système d'exploitation indiqué par le pavé 90. Les programmes d'application indiqués dans les pavés 92 et 94, programmes résidant dans la RAM 56 (Fig. 4) et la ROM 54 (Fig. 4), peuvent être exécutés par le microprocesseur 50 (Fig. 4) sous le contrôle des systèmes d'exploitation. Lorsqu'un Caractère Manuscrit est requis ou est indiqué par une interruption, c'est le logiciel

de reconnaissance de tracés manuscrits 96 qui est appelé. Un premier sous-programme, indiqué dans le pavé 98, code les coordonnées X, Y en Traits de Tracé. Les caractéristiques des Traits de Tracé sont alors définies par un sous-programme 100 qui est suivi par une comparaison des Traits de Tracé avec une base de données qui a été chargée de la ROM 54 (Fig. 4) dans la RAM 56 (Fig. 4). Cette comparaison est réalisée par un sous-programme 102. Lorsque le système d'exploitation est réalisé dans le mode d'« acquisition », la base de données est mise à jour à l'aide des nouveaux signes et données de Trait de Tracé, comme indiqué dans le pavé 104. D'une manière analogue, un document précédemment mis en mémoire peut être édité par le programme d'application 92 en utilisant la fonction d'édition 94 qui est appelée par l'opérateur et qui fournit les instructions sous forme d'entrées, en utilisant les programmes 98, 100 et 102 du programme de reconnaissance de tracés manuscrits 92.

Si l'on se reporte maintenant également à la Fig. 7, le système d'exploitation 90 (Fig. 6) exécute le logiciel de reconnaissance de Caractères Manuscrits 96 (Fig. 6) en acceptant comme entrée les Points de coordonnées X, Y, indiqués dans le pavé 110, qui correspondent à la position du stylet 16 (Fig. 2) sur l'écran d'entrée 18 (Fig. 2) et il code ces Points sous forme de Traits de Tracé, comme indiqué dans le pavé 112. Le programme caractérise alors ces Traits de Tracé à l'aide d'un certain jeu de description, par exemple en prenant en compte la longueur, la courbure, la pente et la position du Trait de Tracé, comme indiqué dans le pavé 114. Dans le pavé 116, on trouve alors la meilleure comparaison du Trait de Tracé caractérisé ou de la séquence de Traits de Tracé caractérisée, avec ceux présents dans la base de données. Si une coïncidence étroite suffisante est trouvée, le caractère est identifié dans le pavé 118 et l'entrée de la base de données est permutée avec l'entrée située au-dessus, comme indiqué dans le pavé 120. De cette manière, les caractères les plus fréquemment identifiés vont « s'élever » jusqu'au haut de la base de données et le rendement d'ensemble du système, mesuré en temps nécessaire pour trouver une coïncidence, s'accroîtra. Si une coïncidence n'est pas trouvée, l'utilisateur peut procéder à une adjonction au bas de la base de données, comme indiqué dans le pavé 122.

Les Fig. 8 et 9 vont maintenant servir à présenter un organigramme du programme d'ordinateur permettant de reconnaître une séquence particulière de Trait de Tracé. Ce programme d'ordinateur démarre au premier pavé 150 et progresse de façon à traiter les tensions X, Y à partir du pavé de traitement 152, les tensions ayant été converties en un signal numérique. Le

programme s'avance alors jusqu'au pavé de décision 154 où le programme détermine si le crayon ou stylet 16 (Fig. 2) est hors de contact avec l'écran d'entrée 18. Cette détermination est réalisée par le fait que la tension X et la tension Y valent l'une et l'autre zéro. Si le programme détermine que le

5 crayon est relevé, le Trait de Tracé est alors déterminé comme étant achevé et le programme se branche sur le pavé de décision 156. Dans ce pavé de décision 156, le programme détermine s'il existe moins de trois Points dans le Trait de Tracé et, si tel est le cas, le programme se branche sur le pavé de décision 158. Dans ce pavé de décision 158, le programme détermine s'il

10 y a ou non zéro Point dans le Trait de Tracé. S'il y a zéro Point dans le Trait de Tracé, le programme se reboucle sur le commencement du pavé de traitement 152 où un autre jeu de Points est lu. Si le compteur de Points (qui progresse dans le pavé de traitement 164) indique qu'il y a plus de zéro Points, le programme se branche sur le pavé de traitement 172. Dans ce

15 pavé de traitement 172, le Trait de Tracé est identifié comme étant un point et sa hauteur au-dessus de la ligne de base (HALB) est calculée dans le pavé de traitement 173. A partir de ce pavé de traitement 173, le programme progresse jusqu'au pavé de traitement 171.

Par contre, si c'est le signal de crayon posé qui est reçu, le programme se branche sur le pavé de traitement 160 où les tensions sont introduites dans une fonction linéaire afin de déterminer le Point de coordonnées en utilisant les formules suivantes :

$$X = a_1 v_1 + b_1$$

$$Y = a_2 v_2 + b_2$$

15 Les constantes  $a_1$  et  $b_1$  sont des paramètres linéaires que l'on détermine en calibrant la surface d'entrée du visuel particulier considéré.

Une fois que les tensions ont été soumises à ces fonctions linéaires, le programme s'avance jusqu'au losange de décision 162 où le programme détermine s'il s'agit ou non d'un Point erroné. Cela se fait en comparant la

30 distance entre Points et en éliminant un Point si la distance est trop grande (on utilise actuellement une valeur supérieure à 2,54 mm). D'autre part, un Point est également éliminé si les Points sont trop proches l'un de l'autre. On fusionne actuellement des points s'ils sont à moins de 0,38 mm.

Le problème de comparaison qui se pose pour le premier Point est

35 résolu en déterminant si un Point est le premier Point après que le crayon se soit posé, et on n'utilise alors ce Point que pour vérifier le Point suivant qui est accepté, en supposant que ce Point se trouve à moins de la distance maximale (2,54 mm).

Si la distance entre Points est déterminée comme ne respectant pas les deux critères, le programme abandonne le Point et se reboucle au haut du pavé de traitement 152 afin de lire une autre paire de tensions de Points de coordonnées.

5 Par ailleurs, si les Points respectent les critères, le programme se poursuit vers le pavé de traitement 164 où un compteur de Points est incrémenté de façon à garder trace du nombre des Points. Ce nombre est utilisé dans le losange de décision 156, comme indiqué plus haut. Le programme progresse alors jusqu'au pavé de traitement 166 où les Points sont lissés  
10 conformément à l'une quelconque d'un certain nombre de formules. Le lissage est utilisé pour rendre minimal le bruit provenant de la numérisation, d'un déplacement aléatoire de la main et du bruit électronique. La technique la plus simple de lissage est une moyenne de Points multiple qui permet de calculer de nouveaux Points  $(x_i', y_i')$  de la manière suivante :

$$15 \quad \begin{aligned} n_2 \\ x_i' = \sum x_i / (n_2 - n_1 + 1) \\ i = n_1 \end{aligned}$$

et d'une manière analogue pour  $y_i'$  qui est lissée sur des Points  $n_1 - n_2$ .

20 Une autre méthode simple est appelée méthode des moyennes pondérées courantes et utilise la formule suivante :

$$x_i' = \alpha x_i + (1 - \alpha) x_{i-1}$$

Alpha est une constante de pondération qui, est habituellement positive (et inférieure à un) et que l'on a utilisé à 0,25. Les sommations ont été prises avec  $n_2$  moins  $n_1$  égal à un. Une troisième méthode implique ce que l'on appelle une adaptation de spline pour laquelle on utilise la formule suivante :

$$25 \quad x_i' = (x_{i-1} + 4 x_i + x_{i+1})$$

30 On peut appliquer l'une quelconque des méthodes qui précèdent, soit avant, soit après filtrage. Ce filtrage est réalisé dans le but de réduire le nombre de Points d'entrée et d'espacer les données de façon que les calculs de différence et/ou d'angle puissent être effectués entre des limites acceptables d'erreur aléatoire. Il s'est avéré qu'un filtre efficace est constitué par un procédé simple permettant de fusionner une séquence de Points et qui consiste à exclure l'acceptation de Points ultérieurs à l'intérieur d'une distance fixée par rapport aux Points précédemment acceptés.

35 A partir du pavé de traitement 166, le programme progresse jusqu'au pavé de traitement 168 où le Point est mis en mémoire dans un élément de rangement qui est incrémenté à chaque nouveau Point apparaissant depuis le dernier signal de crayon posé. De la sorte, il est créé un élément de rangement de Points qui est adressable, pour chaque séquence de Points obtenus

entre un signal de crayon posé et un signal de crayon relevé. Cette séquence de Points est appelée Trait de tracé. A partir du pavé de traitement 168, le programme se reboucle sur le haut du pavé de traitement 152 où un autre Point est obtenu jusqu'à ce qu'un signal de crayon relevé termine le Trait de Tracé.

Dans le losange de décision 156, il a été procédé à une détermination sur la question de savoir s'il y avait ou non moins de trois Points dans un Trait de Tracé. Par définition, s'il y a trois Points ou plus dans un Trait de Tracé, ce Trait de Tracé est une ligne et non un point. S'il y a trois Points ou plus dans le Trait de Tracé, le programme se branche sur le pavé de sous-programme 170. Dans ce pavé de sous-programme 170, qui est discuté ci-dessous de manière plus détaillée en regard de la Fig. 9, le Trait de Tracé est caractérisé en ce qui concerne sa pente et sa hauteur par rapport à la ligne de base.

Comme on peut le voir d'après ce qui précède, la segmentation du train de Points de coordonnées dans un Trait de Tracé est basée principalement sur la détermination du moment où le stylet 16 est « relevé » ou n'est pas en contact avec la surface de l'écran d'entrée 18. En variante, un train de Points peut être segmenté de façon à former des Traits de Tracé sur la base d'autres considérations. C'est ainsi par exemple qu'ils peuvent être segmentés sur la base de variations apparaissant dans une courbure calculée localement ou sur la base d'une courbure locale importante. Une courbure locale se calcule d'après la variation de distance le long des coordonnées d'entrée, divisée par la variation de pente. Cela donne le rayon de courbure. Lorsque le rayon de courbure varie rapidement par rapport à la distance le long des coordonnées d'entrée, ou si le rayon est trop petit, un Trait de Tracé de segmentation est alors supposé se terminer, et un nouveau Trait de Tracé commence. D'autres techniques de segmentation peuvent tenir compte du maximum et du minimum relatifs dans l'une des coordonnées ou les deux et/ou des croisements de courbe dans les coordonnées. On a toutefois établi que ces deux dernières méthodes sont moins efficaces.

La caractérisation d'un Trait de Tracé réduit la séquence de coordonnées définissant le Trait de Tracé ou segment à un jeu de caractéristiques qui sont uniques, généralisées et minimales. Le caractère d'unicité est basé sur deux facteurs, l'un selon lequel les mêmes caractéristiques sont produites par les mêmes coordonnées et l'autre selon lequel les caractéristiques sont suffisantes pour reconstituer une approximation de la séquence initiale de coordonnées. Le terme « généralisé » est utilisé pour signifier que la caractérisation est invariable sous l'effet de transformations ren-

dant les signes invariables (par exemple une translation et l'application d'une fonction linéaire, ou un étirement ou encore une légère inclinaison). L'application d'une fonction linéaire sur toutes les distances s'effectue en adoptant le rapport entre la distance et une largeur de ligne d'entrée d'écriture.

Le jeu minimal de caractéristiques d'un segment comporte les particularités suivantes :

(1) la position du Trait de Tracé : un ou plusieurs des éléments constitués par le centre de gravité/moyenne, par les Points extrêmes d'étendue ou de début et de fin déterminés par rapport à la ligne d'entrée d'écriture, à de précédents Traits de Tracé ou à un centre ou une étendue de caractère ;

(2) la forme d'un Trait de Tracé est caractérisée par un ou plusieurs des éléments constitués par la pente moyenne, la variation de pente (qui est une mesure de la courbure moyenne) et/ou une variation de courbure, par une séquence de pentes sur des segments de longueur spécifique ou sur des longueurs fractionnaires, ou encore par une description grossière de la direction linéaire ou de la finition circulaire et de la direction d'ouverture ;

(3) la longueur du Trait de Tracé, caractérisée par la distance mesurée le long de la courbe et/ou par l'extremum d'étendue le long du système de coordonnées.

Dans un mode de réalisation de la présente invention, on a utilisé avec succès un positionnement donné par le centre de gravité, un extremum d'étendue et les coordonnées de départ et de fin. La forme du Trait de Tracé est codée sous la forme d'une séquence de pentes et de positions verticales (par rapport au centre de gravité du Trait de Tracé). Une approximation de la longueur du Trait de Tracé peut être constituée par le nombre de Points filtrés. En variante, la courbure moyenne peut être codée en variation totale de pente (en même temps que la longueur), en variation entre la pente de départ et de fin, ou dans une adaptation la courbe de l'angle de pente en fonction de la longueur en ce qui concerne le taux de variation de cet angle de pente. Comme caractéristiques supplémentaires que l'on pourrait utiliser, on peut citer l'emplacement des extremum relatifs de coordonnées, les croisements de courbe, les points de rebroussement et la direction du Trait de Tracé. Une méthode particulière utilisée pour déterminer les caractéristiques uniques est indiquée ci-dessous :

1. On détermine les valeurs numériques des Critères correspondant à chaque Trait de Tracé du Signe Manuscrit.

2. On détermine les valeurs de la base de données correspondant à chaque Trait de Tracé du Signe Manuscrit précédemment « acquis » et on les soustrait respectivement des valeurs nouvellement déterminées.

5 3. On applique une fonction linéaire sur les valeurs absolues de chaque différence de façon à rendre chacune des cinq mesures raisonnablement équivalente aux autres, les longueurs étant par exemple rendues proportionnelles à la hauteur entre lignes.

4. On additionne les cinq valeurs ainsi déterminées.

10 5. On utilise un seuil préfixé en tant que test de reconnaissance de « qualité » -- si la valeur est trop élevée, les Signes de Police de Caractères sont rarement reconnus, tandis qu'une valeur trop faible provoque un défaut d'identification des Signes de Police. On utilise initialement des seuils d'approximativement 1000, puis on les fait passer à approximativement 100 pour obtenir une reconnaissance améliorée. Si le seuil est dépassé, la comparaison est rejetée et un message d'erreur est créé et visualisé.

15 6. La base de données est explorée pour trouver une différence numérique minimale. Si cette différence minimale est inférieure au seuil acceptable pour la reconnaissance, le Signe de Police de Caractères correspondant est visualisé sur l'écran ou l'instruction est exécutée, suivant le cas.

20 Il s'est également avéré que la classification préférée pour un Trait de Tracé est une classification continue plutôt qu'une classification grossièrement discrète. C'est ainsi par exemple qu'il est préférable de déterminer une pente par un angle mesuré suivant 256 directions plutôt que suivant 8. Comme autres classifications non continues, on peut citer les  
25 barres/arcs/crochets, le nombre et la fermeture des points de rebroussement, ou encore les Traits de Tracé horizontaux ou verticaux.

A partir du sous-programme 170, le programme progresse jusqu'au pavé de traitement 171 où à la fois un Trait de Tracé individuel et un ou plusieurs Traits de Tracé précédents sont comparés à une entrée de base de  
30 données qui est en mémoire dans la RAM 56 (Fig. 4).

35 Cette comparaison commence initialement par trois questions éliminatoires qui sont posées par le programme dans les losanges de décision 174, 176 et 178. Dans chaque cas, si l'entrée de la base de données est éliminée, le programme progresse jusqu'à un pavé de traitement 180 où l'adresse de la nouvelle entrée de données est reçue et à partir duquel le programme se reboucle sur le haut du pavé de traitement 171. Dans le losange de décision 174, la première question éliminatoire est posée en recherchant si le nombre des Traits de Tracé est différent. Si ce nombre des Traits de Tracé est le même, le programme progresse jusqu'au losange de décision 176 où la

Hauteur Au-dessus de la Ligne de Base (HALB) moyenne est calculée et comparée à la HALB de l'entrée de données. Cette entrée est éliminée si la différence dans les HALB moyennes est supérieure à la moitié de la hauteur de la ligne d'entrée. A partir d'une détermination négative dans le losange de décision 176, le programme progresse jusqu'au losange de décision 178 où le nombre de Points par Trait de Tracé est comparé, l'entrée de la base de données étant éliminée si la différence sur le nombre de Points est supérieure à 10. Cette détermination diffère de celle effectuée dans le losange de décision 174 dans la mesure où elle ne concerne que le nombre de Points correspondant à chaque Trait de Tracé. Par contre, dans le losange de décision 174, certaines lettres, telles que les lettres majuscules « E » et « A », comportent plus d'un Trait de Tracé par lettre.

Si une entrée de données n'est pas éliminée par le losange de décision 178, le programme progresse alors jusqu'au pavé de traitement 182 où ce programme calcule un étalon à utiliser pour déterminer la finesse de la coïncidence existant entre l'entrée sélectionnée dans la base de données et le Trait de Tracé tiré. Un étalon actuellement préféré est constitué par la somme des valeurs absolues des différences existant entre les valeurs de Trait de Tracé et les valeurs d'entrée de base de données :

- a) pour les distances ou longueurs, en unités de 1/80ème de la hauteur de ligne (par exemple de l'espace 26 de la Fig. 2), et
- b) pour les pentes, en unités de 1/256ème de 360° pour tous les Points existant le long de la diagonale de la matrice de comparaison.

En variante, on peut utiliser des Techniques de Programmation Dynamique pour rendre la comparaison optimale en utilisant également des éléments hors diagonale.

A partir du pavé de traitement 182, le programme progresse jusqu'au losange de décision 186 où est déterminée une coïncidence. D'une manière pratique, une coïncidence se détermine à l'aide (de l'application) d'un étalon (arbitraire) (variance maximale autorisable) qui est la somme des valeurs absolues des différences existant entre la caractérisation de Trait de Tracé introduite et celle de l'entrée de base de données en mémoire. Dans le pavé de traitement 183, c'est celui qui est le plus faible parmi l'étalon actuel et l'étalon le plus faible précédent qui est sauvegardé comme constituant la meilleure coïncidence. Le programme progresse alors jusqu'au losange de décision 184 où une détermination est effectuée pour savoir si l'entrée actuelle est ou non la dernière entrée de la base de données. Si elle ne l'est pas, le programme se branche sur le pavé de traitement 180 où est sélectionnée l'entrée suivante. Si c'est la dernière entrée, le programme



progresses jusqu'au losange de décision 186 où une détermination d'une coïncidence est réalisée sur la base de l'étalon se trouvant au-dessous d'un seuil préfixé. Ce seuil est fixé par l'utilisateur en se basant sur son expérience du système.

5 Si aucune coïncidence n'est obtenue, le programme se branche sur le losange de décision 188 où est réalisée une détermination sur la question de savoir si tous les Traits de Tracé ont été contrôlés ou non. Si le dernier Trait de Tracé a été contrôlé, le Trait de Tracé actuel est alors comparé de manière séquentielle à un Trait de Tracé précédent pour toutes les  
10 entrées à deux Traits de Tracé. De même que dans la comparaison avec toutes les entrées de dictionnaire à un Trait de Tracé, la meilleure comparaison de concordance pour tous les Traits de Tracé introduits constitue le signe ou séquence de signes, reconnu(e).

Si par contre le dernier Trait de Tracé a été lu et qu'il n'existe pas encore de coïncidence, le programme progresse alors jusqu'au pavé de traitement  
15 190 où une question est affichée sur l'écran de visualisation 20, demandant à l'utilisateur si un nouveau Signe de Police de Caractères doit être ajouté à la base de données. L'utilisateur répond et cette réponse est utilisée dans le losange de décision 192. Ou bien la séquence de Trait de Tracé est  
20 ajoutée à la base de données dans le pavé de traitement 194 et le programme se rebranche sur le haut du pavé de traitement 152, ou bien le programme se branche immédiatement sur ce haut du pavé de traitement 152.

Si par ailleurs une coïncidence est déterminée dans le losange de décision 186, le programme se branche sur le pavé de traitement 195 où le  
25 programme brasse la base de données en intervertissant l'emplacement de série de l'entrée faisant l'objet de la coïncidence avec l'entrée située au-dessus d'elle. Le programme progresse alors jusqu'au pavé de traitement 196 où ce programme met à zéro le compteur de Points et le compteur d'incrémentations. Le programme progresse ensuite jusqu'au pavé de traitement 198 où le  
30 ou les Traits de Tracé qui font l'objet d'une coïncidence et d'une caractérisation sont visualisés par l'ordinateur comme constituant le Signe de Police de Caractères identifié. Cette visualisation est située à l'emplacement où l'introduction a été effectuée sur l'écran d'entrée 18 (Fig. 2).

A partir du pavé de traitement 198, le programme progresse jusqu'au  
35 pavé de traitement 200 où ce programme peut effectuer n'importe quelle instruction qui a été interprétée. Une variante de caractérisation du Trait de Tracé utilise les Points eux-mêmes à la place de la longueur, de l'extension, de la courbure, et de la position.

La caractérisation des Traits de Tracé présentement considérée est

illustrée de manière plus détaillée sur la Fig. 9. Le sous-programme de caractérisation des Traits de Tracé 170 exécute essentiellement une transformation mathématique de chaque Point, sur une base de travail Point par Point, de façon à transformer les Points en les faisant passer d'un système de coordonnées cartésiennes X, Y à un système dans lequel les coordonnées sont la pente normalisée de chaque Point et la hauteur normalisée de ce Point au-dessus de la ligne de base (HALB).

Le sous-programme 170 calcule tout d'abord la pente Point par Point dans le pavé de traitement 220, puis calcule la hauteur de chaque Point au-dessus de la ligne de base dans le pavé de traitement 222. La pente et la HALB de chaque Point sont alors normalisées, respectivement à  $1/256$ ème de  $2\pi$  et à  $1/80$ ème de la largeur de la ligne d'entrée, dans le pavé de traitement 224. A partir de ce pavé de traitement 224, le système progresse jusqu'au pavé de traitement 226 où les valeurs normalisées calculées pour chaque Point sont mises en mémoire dans un élément de rangement adressable. Le sous-programme retourne alors au programme par l'intermédiaire du pavé de sortie 228.

Lorsque la comparaison s'effectue entre chaque Trait de Tracé et les valeurs mises en mémoire, cette comparaison est effectuée pour la pente normalisée du Point et la hauteur normalisée du Point au-dessus de la ligne de base. Comme indiqué ci-dessus, une coïncidence se détermine à l'aide d'un étalon arbitraire qui est la somme des valeurs absolues des différences existant entre le Trait de Tracé écrit et le Trait de Tracé de dictionnaire ou Trait de Tracé en mémoire. Le système procède à une « acquisition » en ajoutant de nouveaux Traits de Tracé à la base de données de dictionnaire. Une fois cette base de données remplie, les Signes de Police de Caractères qui sont utilisés rarement sont remplacés par de nouvelles entrées.

Dans un mode pratique de réalisation de la présente invention, l'algorithme a identifié avec succès des lettres majuscules et minuscules et des chiffres écrits séparément l'un de l'autre. Pour des Signes Manuscrits qui sont tracés de façon à être continus, une extrapolation directe exigerait l'exploration d'une base de données de manière séquentielle pour des signes à un, deux, trois, etc. Traits de Tracé et la recherche de la meilleure concordance. Lors de l'identification d'une concordance de Trait de Tracé, une lettre « nouvelle » fait l'objet d'une reconnaissance à titre d'essai, à la seule différence du fait que ce sont les quelques Traits de Tracé suivants qui sont analysés pour vérifier s'ils modifient le signe précédent en permettant une meilleure concordance. C'est ainsi par exemple que deux Traits de Tracé qui ont été identifiés comme étant des « uns » seraient combinés

et modifiés pour donner la lettre majuscule « H » une fois qu'une barre transversale a été identifiée.

5 La conception de système présentée par les Fig. 7 à 9 pourraient facilement être codée dans pratiquement n'importe quel langage informatique par un spécialiste ordinaire dans la technique de programmation des ordina-  
10 teurs. Les listings en langage source prévus pour un programme d'application utilisant l'invention ici présentée font l'objet de l'Annexe II. Le logiciel de cette Annexe II est écrit en Microsoft Basic, langage informatique courant qui est disponible pour pratiquement tous les micro-ordinateurs et  
15 systèmes d'exploitation. Ce programme est une démonstration complète d'édition de texte qui tire avantage de nombreuses particularités-clés de l'invention et prouve les perfectionnements qui peuvent être réalisés par rapport aux systèmes traditionnels de traitement de texte en utilisant cette invention.

15 Les lignes de programme 2600 à 4000 contiennent le sous-programme de reconnaissance des caractères qui comprend le logiciel de codage nécessaire pour obtenir les coordonnées X et Y. Cette section du programme correspond à la Fig. 8.

20 Les lignes de programme 2600 à 2699 constituent un sous-programme conçu pour obtenir les coordonnées X et Y d'un Point donné. Ce codage correspond aux pavés 152, 154, 160, 162, 164, 166 et 168 de la Fig. 8.

Les lignes de programme 3000 à 3339 constituent un sous-programme d'analyse et de caractérisation des Points et des Traits de Tracé formant une application des pavés 156, 158, 170, 172 et 173 de la Fig. 8.

25 Les lignes de programme 3700 à 3790 constituent un sous-programme conçu pour comparer les Traits de Tracé analysés à une base de données de Traits de Tracé. Ces lignes de programme forment une application des pavés 171 à 184 de la Fig. 8.

30 Les lignes de programme 3810 à 3980 constituent un sous-programme qui est conçu pour acquérir un nouveau caractère. Ce codage forme une application des pavés 186, 188, 190, 192 et 194 de la Fig. 8.

Les lignes de programme 3060 à 3273 constituent un sous-programme conçu à des fins de caractérisation des Traits de Tracé. Cette section du codage correspond à la Fig. 9.

35 Les lignes de programme 3060 à 3095 sont utilisées pour calculer des pentes Point à Point et forment une application du pavé 220.

Les lignes de programme 3058, 3241 et 3262 sont utilisées pour calculer une hauteur au-dessus de la ligne de base (HALB) et correspondent au pavé 222 de la Fig. 9.

Les lignes de programme 3253 et 3270 à 3273 sont utilisées pour normaliser la hauteur et la pente des points et correspondent au pavé 224 de la Fig. 9.

5 La ligne de programme 3253 est utilisée pour mettre en mémoire la hauteur au-dessus de la ligne de base et forme une application du pavé 226 de la Fig. 9.

10 Le programme précédent peut être rangé dans la mémoire d'un micro-ordinateur ou microprocesseur en exigeant approximativement 25K de mémoire machine, de sorte que l'on peut constater que l'utilisation du programme n'emploie pas une grande quantité de mémoire coûteuse et est relativement rapide pour réaliser l'exécution de ce programme. Si ce programme est écrit en langage autre que le Basic, exigeant moins de mémoire, par exemple le langage d'assemblage, la taille du programme peut être rendue plus petite.

15 Les pavés 195 à 200 de la Fig. 8 apparaissent à certains emplacements logiques au cours du codage.

Un dictionnaire des variables de la section pertinente de codage fait l'objet de l'Annexe III.

20 La Fig. 13 représente un organigramme utilisé pour le logiciel d'édition ( « Editeur » ) présenté sur les Fig. 11A à 11I et décrit ci-dessus. Une fois l'Editeur chargé dans le système (pavé 229), le contrôle de l'écran est rendu au système. Ce système progresse alors de la manière normale décrite ci-dessus de façon à acquérir des Points et à les visualiser (pavé 230), convertir ces Points en Traits de Tracé (pavé 231), caractériser chaque Trait de Tracé (pavé 232) et tenter de faire coïncider le ou les Traits de Tracé avec la base de données (pavé 233). Dans le pavé de traitement 234, le système envoie chaque Signe Manuscrit à l'Editeur en vue d'interpréter et exécuter si nécessaire une instruction. Au losange de décision 235, l'Editeur détermine si le Signe Manuscrit est ou non un Signe d'Edition ou un Signe de Police de Caractères. Si la détermination indique que le caractère est un Signe d'Edition, l'Editeur progresse jusqu'au pavé de traitement 236 où il détermine quel Signe d'Edition a été introduit et exécute la fonction d'Edition. Si la détermination indique que le caractère n'est pas un Signe d'Edition, c'est alors le caractère alphanumérique correspondant à l'entrée manuscrite qui est visualisé au pavé de traitement 30 237. Dans une variante de configuration de l'Editeur, des Signes de Police de Caractères ne seront acceptés que lorsque l'Editeur est dans le « Mode d'Insertion ». Cette structure donne l'assurance que chaque Signe de Police de Caractères est vérifié avant d'être ajouté à un document.

35

L'Editeur utilise divers signes conçus pour rendre l'édition sur ce système semblable à l'édition traditionnelle à l'aide d'un crayon et papier, mais beaucoup plus efficace que celle-ci. Parmi ces fonctions, on peut citer, sans qu'elles y soient limitées :

5 Le signe EFFACER - « \_\_\_\_\_ », c'est-à-dire une ligne horizontale tracée d'un bout à l'autre d'un ou plusieurs caractères. L'Editeur retirera les caractères recouverts et reformatera le texte.

10 Le signe AJUSTER LES MARGES - « | », c'est-à-dire une ligne verticale plus longue que la hauteur d'une ligne de la visualisation. L'Editeur ajustera la marge à la position indiquée et reformatera le texte.

Le signe INSERER - « A », c'est-à-dire un signe de renvoi tracé au Point où le texte doit être ajouté. L'Editeur visualise une ligne pour écrire l'entrée (Fig. 11B) et, une fois l'entrée reconnue, l'introduit dans le texte.

15 Les signes MARQUER LE TEXTE - « < » et « > », c'est-à-dire des signes « plus petit que » et « plus grand que » tracés au début et à la fin d'un bloc de texte. Le texte marqué est visualisé en vidéo inverse et des fonctions spéciales peuvent être exécutées sur ce bloc.

EFFACER LE TEXTE MARQUE - Un signe effacer tracé à l'intérieur du texte marqué supprimera le texte marqué et reformatera.

20 DEPLACER LE TEXTE MARQUE - Un signe insérer tracé à un endroit quelconque à l'intérieur du texte déplace le texte marqué vers la position indiquée, l'efface de sa position d'origine et reformate le texte.

25 REMPLACER LE TEXTE MARQUE - Un signe insérer tracé à l'intérieur du texte marqué visualise une ligne d'entrée et remplace le texte marqué par le texte introduit.

Les Signes d'Edition décrits ci-dessus peuvent être remplacés par les Signes particuliers d'Edition préférés par chaque utilisateur, ce qui personnalise ainsi l'Editeur et empêche de nouveaux utilisateurs d'avoir à apprendre des Signes d'Edition peu familiers.

30 D'autres modifications et perfectionnements apportés à la présente invention seront évidents pour les spécialistes en la matière. C'est ainsi par exemple que les caractéristiques courantes de chaque Signe de Police de Caractères pourraient être extraites et organisées sous forme d'un signe de synthèse. Les caractéristiques de ce signe de synthèse pourraient alors être  
35 exagérées de façon à rendre maximale leur divergence par rapport à d'autres signes de synthèse. Cela permettrait de créer une base de données optimale, très compacte. Par ailleurs, à titre d'exemple, une base de données créée à l'aide du mode préféré de réalisation de l'invention ici décrit donnerait habituellement deux ou trois caractérisations différentes pour chaque signe.

L'invention présente de nombreuses applications utiles, pratiquement sans limitation. Les applications les plus évidentes sont l'édition de textes et le remplissage et la modification de formulaires. Certaines des nombreuses autres applications qui ne pourraient pas venir aussi facilement à l'esprit sont l'écriture dans des langues utilisant de grands nombres de signes comme le Japonais ou le Chinois, l'écriture en Arabe ou langues analogues constituées d'un nombre limités de signes complexes, l'écriture des équations chimiques, y compris celles impliquant des composés organiques, l'écriture de la musique (une « fenêtre » comportant cinq lignes parallèles peut être prévue pour ces applications musicales), l'écriture de signes et codes en vue d'une manipulation graphique de données, y compris le transfert de données graphiques sur un programme de calcul de tableaux financiers, le domaine de l'éducation, où des questions fixées à l'avance sont présentées sur des écrans et où les réponses sont présentées en écriture cursive, le domaine d'enseignement des mathématiques, où des nombres sont introduits manuellement dans des équations et où les équations sont analysées en utilisant ces nombres pour en déterminer le résultat, les applications de conception et fabrication assistées par ordinateur impliquant des signes, des formes géométriques et des éléments analogues.

38  
ANNEXE I

'h' (1) ns = 1 x = (-17..0..19) y = (-6..18..60)  
 1) np = 20 x = (-17..0..19) y = (-6..18..60)

	5,	42
5	-57,	34
	-85,	26
	-71,	19
	-76,	10
	-64,	3
	-89,	-3
10	-72,	-9
	-85,	-17
	-72,	-24
	47,	-17
	56,	-11
15	43,	-3
	39,	3
	11,	7
	-9,	5
	-57,	-3
20	-71,	-11
	-71,	-18

'H' (1) ns = 2 x = (-27..0..37) y = (-20..18..61)  
 1) np = 14 x = (-27..-12..-2) y = (-20..22..61)

25	-9,	37
	4,	38
	-39,	32
	-48,	25
	-58,	16
	-71,	8
30	-72,	2
	-64,	-4
	-89,	-11
	-72,	-17
	-80,	-24
35	-89,	-30
	-80,	-36

2) np = 20 x = (-21..10..37) y = (-15..16..59)

	27,	43
	-80,	37
40	-71,	29
	-98,	23
	-80,	16
	-78,	8
	-85,	1
45	-89,	-6
	-80,	-12
	-80,	-18
	-64,	-25
	-64,	-31
50	48,	-25
	56,	-18
	48,	-12
	64,	-6
	89,	1
55	-124,	0
	-124,	-2

## ANNEXE II

```

1 REM Z SERIES COPYWRITE: LINUS TECHNOLOGIES INCORPORATED, RESTON, VA., 7/30/85
10 DEFINIT A-Y : REM all integers except z
20 DIM X(2000),Y(2000),D(9,800),DN(300),DC(300),SX(80),SY(80),SL(80),SC(80)
,SA(80),SXX(80),SXI(80),SYX(80),SYN(80),SNO(80),MTITL$(20),AS(1500),LL(24),
5 ASS(200),SLET(80),DBC(9,10),ASTRT(80),AEND(80)
25 PRINT FRE(0)
30 PRINT "Z SERIES COPYWRITE: LINUS TECHNOLOGIES INCORPORATED, RESTON, VA., 7/30
/85"
40 IMARGN=5 : IMARGX=62 : JTOP=1 : JBTM=15 : JMENU=160 : CRS=""
10 50 KYBD=0: LHT=16 :SXTT=4: JTX=0: ACPT=1000
60 LHT4=LHT\4 :LHT3=LHT\3 :LHT5=LHT\5 :LHT2=LHT\2
70 PRINT "PLEASE INPUT PEN SWITCH...-1 TO RERUN, 0 TO RUN, 1 TO RUN & STORE": IN
PUT SWTPN
15 75 IF SWTPN > -1 THEN GOSUB 10700 :REM TOUCH TECH SCREEN IN ASCII @ 9600
80 PRINT "PLEASE INPUT DEBUG ...-5 TO 5 ": INPUT DBUG
85 PRINT "PLEASE INPUT ACCEPTANCE CRITERIA...10 TO 1000 ": INPUT ACPT
90 PRINT "PLEASE INPUT EDIT/LEARN SWITCH...1 TO EDIT": INPUT EDLN
92 PRINT "PLEASE INPUT KEYBOARD SWITCH...0 FOR KEYS, 1 FOR MENU ": INPUT KYBD
20 95 PRINT "PLEASE INPUT RESTART SWITCH...1 TO RESTART WITH STORED DATA BASE": INF
UT RESTRT
97 IFAVE=0 :IF SWTPN=-1 THEN IFAVE=0 :REM PRINT "PLEASE INPUT AVERAGING SWITCH..
.1 TO READ & AVERAGE, 2 TO AVERAGE INPUT": INPUT IFAVE
100 KEY OFF :SCREEN 2 :CLS
25 120 IF RESTRT = 1 THEN GOSUB 12700
130 IF EDLN = 1 THEN GOSUB 12100 : REM READ IN TEXT
140 IF EDLN = 1 THEN GOSUB 600 : REM PRINT TEXT
150 IF SWTPN = 1 THEN GOSUB 7200
151 IF SWTPN = -1 THEN GOSUB 7000
30 170 IF EDLN <> 1 THEN GOSUB 6100
180 IF EDLN <> 1 THEN GOSUB 12500
200 REM EDIT
205 LINY=0
210 IF SWTPN = -1 THEN GOSUB 7100 ELSE GOSUB 2600 :REM TOUCH TECH SCREEN IN ASC
II @ 9600
35 211 IF SWTPN =1 THEN GOSUB 7300
220 REM RECOGNIZE SYMBOLS
221 IF . > 2000 THEN LOCATE 23,1: PRINT "TOO MANY POINTS" : STOP
222 IF . < 4 THEN GOTO 200
230 GOSUB 3000 :GOSUB 3400 :REM -, <>, |, ^, \, /, P, E
240 ON NCHAR GOTO 1200,1300,1700,6200,6700,6500,6600,12500,6900,200 :REM DELETE,
MOVE,FORMAT,INSERT,WRITE OVER,REMOVE,PARAGRAPH,INSERT CHARACTER,END
250 GOTO 200
500 END
600 REM DISPLAY TEXT FROM SINGLE CHARACTER ARRAY
45 605 LOCATE 23,20: PRINT " "
610 I=II : J=JJ : REM INITIAL DISPLAY POINT FOR NC1
620 FOR K=NC1 TO NC
630 IF AS(K) <> CRS THEN GOTO 660
640 LOCATE J,I : PRINT " " : I=I+1 : IF I < 81 GOTO 640
50 650 J=J+1 : I=IMARGN : LL(J)=K : GOTO 740
660 IF I < IMARGX THEN GOTO 670 ELSE IF AS(K)=" " GOTO 690
670 IF J > JBTM THEN LOCATE 21,20: PRINT "READY", :RETURN
680 LOCATE J,I : IF I=IN THEN GOTO 710 ELSE PRINT AS(K) : I=I+1 : GOTO 740
690 LOCATE J,I : PRINT " " : I=I+1: IF I < 81 GOTO 690
55 700 J=J+1 : I=IMARGN : LL(J)=K+1 : GOTO 740
710 IF AS(K) <> " " THEN GOTO 730
720 IF LL(J)=K GOTO 740
730 I=I+1 : PRINT AS(K)
740 IF I > IMARGX+10 THEN GOTO 690 ELSE NEXT K
60 750 IF J > JTX THEN GOTO 790
760 JS=J
765 LOCATE J,I : PRINT " " : I=I+1: IF I < 81 GOTO 765
770 J=J+1 : I=IMARGN: IF J > JTX THEN GOTO 780 ELSE GOTO 765

```



```

780 J=JS
790 JTX=J : LOCATE 23,20: PRINT "READY", :RETURN
1200 REM DELETE
5 1205 XX=SXX(1): XN=SXX(1): YX=SYX(1): YN=SYN(1)
1206 IF NSTRK=1 GOTO 1220
1210 FOR L=2 TO NSTRK
1212 IF XX < SXX(L) THEN XX=SXX(L)
1214 IF XN > SXN(L) THEN XN=SXX(L)
10 1216 IF YX < SYX(L) THEN YX=SYX(L)
1218 IF YN > SYN(L) THEN YN=SYN(L)
1219 NEXT L
1220 IN=IMARGN
1225 I=INT(XX/8)+1 : IF I < IN THEN I=IN
15 1230 J=INT(YX/8)+1 : ML=LL(J) : M2=ML+I-IN : IF AS(ML)=" " THEN M2=M2+1
1231 IF AS(ML)=CRS THEN M2=M2+1
1232 I=INT(XN/8)+1 : IF I < IN THEN I=IN
1234 J=INT(YN/8)+1 : ML=LL(J) : M1=ML+I-IN : IF AS(ML)=" " THEN M1=M1+1
1240 IF AS(ML)=CRS THEN M1=M1+1
20 1250 MD=M2-M1+1 : IF MD < 0 THEN MD=M1-M2+1 : M1=M2
1260 FOR M=M1 TO NC : AS(M)=AS(M+MD) : IF M+MD=NC THEN GOTO 1270 ELSE NEXT M
1270 FOR M=NC-MD+1 TO NC : AS(M)=" " : NEXT M
1280 II=I: JJ=J: NC1=M1 : GOSUB 600
1290 NC=NC-MD : GOTO 200
25 1300 REM MOVE INDICATED BY OPENING BRACKET
1305 IN=IMARGN
1310 XX=SXX(1): XN=SXX(1): YX=SYX(1): YN=SYN(1)
1320 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE BRACKET POSITION
1330 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
1340 ML=LL(J): M1=ML+I-IN :REM CALC A ARRAY INDEX OF LETTER AT I,J
1350 IF AS(ML)=" " THEN M1=M1+1
1360 IF AS(ML)=CRS THEN M1=M1+1
1390 XX=SXX(2): XN=SXX(2): YX=SYX(2): YN=SYN(2)
1400 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE BRACKET POSITION
35 1410 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
1420 ML=LL(J): M2=ML+I-IN :REM CALC A ARRAY INDEX OF LETTER AT I,J
1430 IF AS(ML)=" " THEN M2=M2+1
1440 IF AS(ML)=CRS THEN M2=M2+1
1450 M3=0:REM INITIALIZE COUNTER
1460 FOR M=M1 TO M2 :REM LOOP OVER PORTION CUT
40 1470 M3=M3+1: REM INCREMENT COUNTER
1480 ASS(M3)=AS(M) :NEXT M :REM COPY CUT TEXT
1490 XX=SXX(3): XN=SXX(3): YX=SYX(3): YN=SYN(3)
1500 REM INSERT WITH M3 LETTERS STORED IN ASS & xx,xn,yx,yn
45 1520 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE BRACKET POSITION
1530 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
1540 ML=LL(J): M3=ML+I-IMARGN :REM CALC A ARRAY INDEX OF LETTER AT I,J
1550 IF AS(ML)=" " THEN M3=M3+1
1560 IF AS(ML)=CRS THEN M3=M3+1
50 1570 MD=M2-M1+1 : IF MD > 0 THEN GOTO 1580 ELSE GOTO 200
1580 FOR M=M1 TO NC : REM LOOP OVER AS ARRAY POSITIONS TO BE SHIFTED
1590 AS(M)=AS(M+MD) : REM SHIFT AS ARRAY LETTERS
1600 IF M+MD = NC THEN GOTO 1610 ELSE NEXT M : REM STOP AT END OF ARRAY=NC
1610 IF M3 > M2 THEN M3=M3-MD :REM OFF SET INSERTION POINT TO ACCOUNT FOR DELETION
55 1620 FOR M=NC-MD TO M3 STEP -1 : REM SPRED ARRAY TO MAKE ROOM FOR INSERTION
1630 AS(M+MD)=AS(M) :REM MOVE LETTERS
1640 NEXT M
1650 FOR M=1 TO MD :REM LOOP OVER CHARACTERS
60 1660 AS(M3+M-1)=AS(M) :REM INSERT LETTERS
1670 NEXT M
1680 CLS: II=IMARGN: JJ=JTOP: NC1=1 : GOSUB 600
1690 GOTO 200
1700 REM REFORMAT MARGINS

```

```

1720 XP=(SXN(L)+SXX(L))\2 :REM AVE BRACKET POSITION
1730 I=XP\8+1 :REM CONVERT X TO I
1740 IF ABS(I-IMARGN) < ABS(I-IMARGX) THEN IMARGN=I ELSE IMARGX=I
1750 CLS: II=IMARGN: JJ=JTOP: NC1=1 : GOSUB 600
5 1760 GOTO 200
2600 REM TT ACQUIRE POINTS IN ASCII @ 9600 WITH TIME OUT
2603 M=20
2605 IF LOC(2) > 0 THEN INPUT$=INPUT$(1,2) : GOTO 2605
2606 PRINT #2,"G" :N=0
0 2610 IF LOC(2) < 40 THEN GOTO 2610
2620 NU=0: INPUT #2,XP,YP :INPUT #2,XP,YP :INPUT #2,XPA,YPA :INPUT #2,XP0,YP0
2630 N=N+1: XPA=(XPA+XP0)\2: YPA=(YPA+YP0)\2: XP0=XPA: YP0=YPA: X(N)=XPA: Y(N)=Y
PA
2640 INPUT #2,XP,YP
5 2653 XPA=(XP+3*XPA)\4: YPA=(YP+3*YPA)\4
2657 IF ABS(XPA-XP0)+ABS(YPA-YP0) < 3 THEN GOTO 2672
2660 I=32*XPA\AX+BX:J=32*YPA\AY+BY:PSET(I,J)
2670 N=N+1 : X(N)=XPA :Y(N)=YPA: XP0=XPA: YP0=YPA
2672 IF LOC(2) > 9 THEN GOTO 2640
10 2675 MM=0
2680 MM=MM+1: IF LOC(2) > 9 THEN GOTO 2640 ELSE IF MM < M THEN GOTO 2680
2685 X(N)=-X(N): IF X(N) = 0 THEN X(N)=-1
2690 IF LOC(2) > 39 THEN GOTO 2620 ELSE NU=NU+1 :IF NU < 4000 THEN GOTO 2690
2695 PRINT #2,"S" : NP=N
5 2696 IF DEBUG <> 0 THEN LOCATE 20,2 :PRINT "NP=",NP
2697 IF IFAVE=2 THEN GOSUB 4200
2698 IF DEBUG <> 0 THEN LOCATE 20,12 :PRINT NP
2699 RETURN
10 3000 REM CALCULATE CHARACTERIZATION OF STROKES
3005 REM GOSUB 4200 :REM 2 POINT AVERAGE POINTS
3010 A1=0: A2=0: DA1=0: DA2=0: SEND=0: ZX=0: ZY=0: ZA=0: ZDA=0: ZL=0: NSP=0
3015 NSTRK=0: XX=ABS(X(1)): XN=XX: YX=Y(1): YN=YX: M1=1: M2=1: M3=2: NLONG=0
3020 FOR M=2 TO NP
3025 IF X(M) < 0 THEN SEND=1 :X(M)=-X(M)
3030 IF M > M1 GOTO 3035 ELSE 3391
3035 IF X(M1) < 0 THEN M1=M GOTO 3391
3040 DX=X(M)-X(M1): DY=Y(M)-Y(M1): ADX=ABS(DX): ADY=ABS(DY): DS=ADX+ADY
3043 IF AY > 0 THEN DY=-DY
10 3044 IF DS < 3 THEN GOTO 3391
3052 IF X(M) < XN THEN XN=X(M)
3053 IF X(M) > XX THEN XX=X(M)
3054 IF Y(M) < YN THEN YN=Y(M)
3056 IF Y(M) > YX THEN YX=Y(M)
3058 ZX=ZX+X(M)*DS: ZY=ZY+Y(M)*DS: ZL=ZL-DS: NSP=NSP-1
15 3060 IF ADX < ADY THEN GOTO 3080 :REM ENCODE ANGLE 0-255 +-n256
3070 IF DX > 0 THEN A1=64-(32*DY)\DX ELSE A1=192-(32*DY)\DX
3075 GOTO 3085
3080 IF DY > 0 THEN A1=(32*DX)\DY ELSE A1=128+(32*DX)\DY
3085 IF NSP>M2 THEN GOTO 3090
10 3086 IF A1<0 THEN A1=A1+256
3087 ZA=ZA+A1*DS
3089 GOTO 3190
3090 IF A1-A2 < -127 THEN A1=A1+256 : GOTO 3090
3095 IF A1-A2 > 128 THEN A1=A1-256 : GOTO 3095
15 3100 ZA=ZA+A1*DS
3110 DA1=A1-A2
3115 ZDA=ZDA+DA1
3120 IF NSP > M3 THEN ADA=(DA2+ADA)/2 ELSE ADA=DA1 : ASTRT(NSTRK+1)=(A1+A2)\2
10 3130 IF NSP = M3+1 THEN ASTRT(NSTRK+1)=(A1+A2)\2
3140 IF ABS(DA1-ADA) > 80 THEN SEND=1
3150 IF ABS(DA1) > 64 THEN SEND=1
3190 IF SEND=0 THEN GOTO 3300
3200 REM END OF STROKE

```

```

3205 IF SEND = 1 THEN GOTO 3220
3206 X(M)=-X(M)
3208 SEND=0: SZL=ZL: SZL=32*SZL\AY: IF SZL < 0 THEN SZL=-SZL
3209 IF NSP < 3 THEN GOTO 3210 ELSE IF SZL > 2 THEN M1=M+1: GOTO 3250
5 3210 IF NSTRK > 0 THEN IF X(SNO(NSTRK)) > 0 THEN X(SNO(NSTRK))=-X(SNO(NSTRK))
3212 M1=M+1
3215 GOTO 3280
3220 SEND=0: SZL=ZL-DS: SZL=32*SZL\AY: IF SZL < 0 THEN SZL=-SZL
10 3226 IF NSP < 4 THEN GOTO 3300 ELSE IF SZL < LHT5 THEN GOTO 3300
3230 M1=M
3240 ZX=ZX-X(M)*DS
3241 ZY=ZY-Y(M)*DS
3242 ZL=ZL-DS: ZA=ZA-A1*DS: ZDA=ZDA-DA1
3243 NSP=NSP-1
15 3250 NSTRK=NSTRK+1
3253 SL(NSTRK)=50*SZL\LHT: AEND(NSTRK)=(A3+A4)\2: SC(NSTRK)=ZDA: AZA=ZA/ZL
3254 IF AZA > 255 THEN AZA=AZA-256: GOTO 3254: REM USE 128 HERE FOR DIRECTION I
NDEPENDENCE
3256 IF AZA < 0 THEN AZA=AZA+256: GOTO 3256
20 3260 SX(NSTRK)=ZX/ZL: SX(NSTRK)=32*SX(NSTRK)\AX+BX
3262 SY(NSTRK)=ZY/ZL: SY(NSTRK)=32*SY(NSTRK)\AY+BY
3264 SA(NSTRK)=AZA
3270 SXX(NSTRK)=32*XX\AX+BX: SXN(NSTRK)=32*XN\AX+BX: SNO(NSTRK)=M
3272 IF AY < 0 THEN SYN(NSTRK)=32*YX\AY+BY: SYX(NSTRK)=32*YN\AY+BY: GOTO 3280
3273 SYX(NSTRK)=32*YX\AY+BY: SYN(NSTRK)=32*YN\AY+BY
3280 IF DEBUG > 3 THEN LPRINT USING "#####";M,NSP,X(M),Y(M),ZL,ZA,A1,DA1,ADA,ZDA
,(A1+A2+A3+A4)\4 ELSE IF DEBUG < -3 THEN PRINT USING "#####";M,NSP,X(M),Y(M),ZL,
ZA,A1,DA1,ADA,ZDA,(A1+A2+A3+A4)\4
3281 IF DEBUG > 2 THEN LPRINT "SL,SA,SC,AS,AE",SL(NSTRK) SA(NSTRK) SC(NSTRK) ASTR
T(NSTRK) AEND(NSTRK) ELSE IF DEBUG < -2 THEN PRINT "SL,SA,SC,AS,AE",SL(NSTRK) SA(
30 NSTRK) SC(NSTRK) ASTRT(NSTRK) AEND(NSTRK)
3285 A1=0: A2=0: DA1=0: DA2=0: DS2=0: DS1=0: SEND=0: ZX=0: ZY=0: ZA=0: ZDA=0: ZL=0
3290 XX=ABS(X(M1)): XN=XX: YX=Y(M1): YN=YX: NLONG=0: NSP=0
3295 GOTO 3391
35 3300 REM FINISH LOOP
3310 M1=M
3320 DA3=DA2: DA2=DA1: DS2=DS1: DS1=DS: A4=A3: A3=A2: A2=A1
3390 IF DEBUG > 3 THEN LPRINT USING "## ==";M,NSP,X(M),Y(M),ZL,ZA,A1,DA1,ADA,ZDA
,(A1+A2+A3+A4)\4 ELSE IF DEBUG < -3 THEN PRINT USING "#####";M,NSP,X(M),Y(M),ZL,
40 ZA,A1,DA1,ADA,ZDA,(A1+A2+A3+A4)\4
3391 NEXT M
3392 IF DEBUG > 1 THEN LPRINT " I NO X Y L A C XX XN Y
X YN ST ND"
3393 IF DEBUG < -1 THEN LOCATE 9,1:PRINT " I NO X Y L A C
45 XX XN YX YN ST ND"
3394 IF DEBUG < -1 THEN LOCATE 10,1:FOR I=1 TO NSTRK:PRINT USING "#####";I,SNO(I
),SX(I),SY(I),SL(I),SA(I),SC(I),SXX(I),SXN(I),SYX(I),SYN(I),ASTRT(I),AEND(I):NE
XT I
3395 IF DEBUG > 1 THEN FOR I=1 TO NSTRK: LPRINT USING "#####";I,SNO(I),SX(I),SY(
50 I),SL(I),SA(I),SC(I),SXX(I),SXN(I),SYX(I),SYN(I),ASTRT(I),AEND(I):NEXT I
3396 IF NSTRK > 1 THEN GOSUB 4000 ELSE GOTO 3399
3397 IF DEBUG < -1 THEN FOR I=1 TO NSTRK:PRINT USING "#####";I,SNO(I),SX(I),SY(I)
,SL(I),SA(I),SC(I),SXX(I),SXN(I),SYX(I),SYN(I),ASTRT(I),AEND(I):NEXT I
3398 IF DEBUG > 1 THEN FOR I=1 TO NSTRK: LPRINT USING "#####";I,SNO(I),SX(I),SY(
I),SL(I),SA(I),SC(I),SXX(I),SXN(I),SYX(I),SYN(I),ASTRT(I),AEND(I):NEXT I
3399 RETURN
3400 REM EDITING SYMBOL RECOGNITION
3410 FOR L=1 TO NSTRK
3412 A1=(SA(L)+32)\64: IF A1 > 3 THEN A1=0
60 3416 IF A1 = 0 THEN IF SL(L) > 100 THEN NCHAR=3: RETURN: REM VERTICAL
3418 IF A1 = 1 THEN IF SL(L) > 60 THEN NCHAR=1: RETURN: REM HORIZONTAL
3420 IF A1 = 2 THEN IF SL(L) > 100 THEN NCHAR=3: RETURN: REM VERTICAL
3422 IF A1 = 3 THEN IF SL(L) > 60 THEN NCHAR=1: RETURN: REM HORIZONTAL

```

```

3430 NEXT L
3500 REM CONVERT LETTER TO EDITTING CODE
3510 ED=1 :GOSUB 3600 :ED =0
3515 IF NLET > 0 THEN GOTO 3530
5 3520 NCHAR=10: LOCATE 20,1: PRINT "TRY AGAIN" :RETURN
3530 IF CSS = "-" THEN NCHAR = 1 :RETURN
3532 IF NLET = 3 THEN IF CSS = "0" THEN NCHAR=2 :RETURN
3536 IF CSS = "." THEN NCHAR=4 :RETURN
0 3538 IF CSS = "\" THEN NCHAR=9 :RETURN
3540 IF CSS = "P" THEN NCHAR=7 :RETURN
3542 IF CSS = "E" THEN NCHAR=8 :RETURN
3544 IF CSS = "/" THEN NCHAR=6 :RETURN
3546 IF NLET= 1 THEN NCHAR=5 :RETURN
3550 GOTO 3520
3590 RETURN
3600 REM PARCEL STROKES INTO CHARACTERS
3602 IF DEBUG > 4 THEN LPRINT USING "#####"; SXX(1),SXN(1),SYX(1),SYN(1),LHT4
3603 SLET(1)=1: NLET=1: IF NSTRK=1 THEN GOTO 3650
3605 FOR L=2 TO NSTRK
0 3606 IF DEBUG > 4 THEN LPRINT USING "#####"; SXX(L),SXN(L),SYX(L),SYN(L)
3607 ICONT=0
3610 FOR L1=1 TO 2 : IF L > 2 THEN GOTO 3615 ELSE IF L1=2 GOTO 3635
3615 IF SXX(L-L1) > SXX(L) THEN GOTO 3630 ELSE IF (SXX(L-L1)+LHT4) < SXN(L) THEN
GOTO 3635
5 3620 IF ABS((SYX(L)+SYN(L))-((SYX(L-L1)+SYN(L-L1)))) > LHT THEN GOTO 3635
3629 ICONT=L1 :GOTO 3635
3630 IF SXN(L-L1) > SXX(L)+LHT4 THEN GOTO 3635 ELSE GOTO 3620
3635 NEXT L1
3640 IF ICONT=0 THEN NLET=NLET-1 :SLET(L)=NLET: GOTO 3649
0 3645 NLET=SLET(L-ICONT) : SLET(L)=NLET :SLET(L-1)=NLET
3646 IF DEBUG > 3 THEN LPRINT "L,NLET,SLET",L,NLET,SLET(L-1),SLET(L-2)
3649 NEXT L
3650 REM LOOP OVER LETTERS
3652 IF ED=1 THEN ZLINY=0: FOR LMN=1 TO NSTRK : ZLINY=ZLINY+SYX(LMN)+SYN(LMN): N
EXT LMN : LINY=ZLINY/NSTRK :LINY=(1-LINY\16)*8-4 :IF DEBUG > 0 THEN LPRINT "LINY=
",LINY ELSE IF DEBUG < 0 THEN LOCATE 22,1 :PRINT "YINY=",LINY
3655 MCHR=1: NST=0: XX=-1: XN=1000: XX1=-100 :YX=-1 :YN=1000 :YX1=-100
3656 FOR LSTRK=1 TO NSTRK-1
0 3657 LST=LSTRK :IF LSTRK = NSTRK-1 THEN GOTO 3659
3658 IF SLET(LSTRK) = MCHR THEN NST=NST+1: GOTO 3659
3659 IF NST > 10 THEN LOCATE 23,1 : PRINT "TOO MANY STROKES", :STOP
3660 FOR M=1 TO NST :LMN=LST-NST+M-1
3661 DBC(1,M)=SL(LMN): DBC(2,M)=SA(LMN): DBC(5,M)=SC(LMN)
3662 DBC(6,M)=(100*(SX(LMN)-SX(LST-NST)))\LHT :DBC(3,M)=ASTRT(LMN) :DBC(4,M)=AEN
D(LMN)
5 3663 DBC(7,M)=(100*(SY(LMN)-LINY))\LHT : DBC(9,M)=( 50*(32*Y(SNO(LMN)-1)\AY+BY-L
INY))\LHT: IF LMN = 1 THEN DBC(8,M)=( 50*(32*Y(2)\AY+BY-LINY))\LHT ELSE DBC(8,M)=
50*(32*Y(SNO(LMN)-1)+2)\AY+BY-LINY)\LHT
3664 IF YX < SYX(LMN) THEN YX=SYX(LMN)
3665 IF YN > SYN(LMN) THEN YN=SYN(LMN)
3666 IF XX < SXX(LMN) THEN XX=SXX(LMN)
3667 IF XN > SXN(LMN) THEN XN=SXN(LMN)
3668 IF DEBUG > 0 THEN LPRINT USING "#####";MCHR,DBC(1,M),DBC(2,M),DBC(3,M),DBC(4
,M),DBC(5,M),DBC(6,M),DBC(7,M),DBC(8,M),DBC(9,M)
5 3669 IF DEBUG < 0 THEN LOCATE 10+M,27: PRINT USING "#####";MCHR,DBC(1,M),DBC(2,M)
,DBC(3,M),DBC(4,M),DBC(5,M),DBC(6,M),DBC(7,M),DBC(8,M),DBC(9,M)
3670 NEXT M :GOSUB 3700
3671 NPOS=(2+(XX+XN)\16-IMARGN)\2
3672 REM PRINT "MCHR,NPOS,CHAR,ED" ,MCHR,NPOS,CSS,ED :INPUT IJK
3673 IF ED = 1 THEN NPOS=0: ASS(MCHR)=CSS
3674 IF NPOS > 0 THEN IF NPOS < 40 THEN ASS(NPOS)=CSS
3675 REM
3676 IF ED (<) 1 THEN LOCATE (LINY-LHT+1)\8,(XX+XN)\16+1 :PRINT CSS

```

```

3677 REM
3680 SXX(MCHR)=XX : SXN(MCHR)=XN
3682 SYX(MCHR)=YX : SYN(MCHR)=YN
3688 MCHR=MCHR+1: NST=1: XX1=XX: XX=-1: XN=1000: YX1=YX: YX=-1: YN=1000
5 3690 NEXT LSTRK
3699 RETURN
3700 REM COMPARE TO DATABASE
3710 R1=16000 :M3=0 :IF NDET=0 THEN GOTO 3900
3715 FOR M=1 TO NDBT
10 3720 IF DN(M) <> NST THEN GOTO 3790 :REM CHECK NUMBER OF STROKES
3725 M2E=9 :IF LINY = 0 THEN M2E=6
3730 R=0 : FOR M1 =1 TO NST
3735 FOR M2=1 TO M2E
15 3738 DR=ABS(D(M2,M3+M1))-DBC(M2,M1))
3740 IF M2 < 2 THEN GOTO 3744 ELSE IF M2 > 4 THEN GOTO 3744
3741 IF DR > 128 THEN DR=ABS(DR-256): GOTO 3741 :REM USE 64 FOR DIRECTION INDEPE
NDENCE
3742 IF ABS(D(5,M3+M1)) > 20 THEN DR=20*DR/(ABS(D(5,M3+M1)))
3744 IF DR > 300 GOTO 3790
20 3748 R=R+DR
3749 IF R > R1 GOTO 3790
3750 NEXT M2 : NEXT M1
3780 MS1=M : R1=R
3790 M3=M3+ DN(M) :NEXT M
25 3800 REM IDENTIFY CHARACTER
3803 CSS=CHRS( DC(MS1))
3804 REM
3805 IF DEBUG<> 0 THEN LOCATE 19,1 :PRINT R1,MS1,CSS
3806 REM
30 3810 IF R1\NST > ACPT GOTO 3900 :REM CHARACTER NOT FOUND
3830 RETURN
3900 REM LEARN NEW CHARACTER
3902 IF LERN = 0 THEN CSS=CHRS(2): RETURN
3904 IF KYBD=0 THEN LOCATE 23,1 :INPUT "CHARACTER NOT UNDERSTOOD, PLEASE TYPE IT
IN: ",CSS
35 3906 IF KYBD = 0 THEN LOCATE 23,1 :PRINT "
"
3910 IF KYBD=1 THEN GOSUB 11400
3912 IF KYBD =1 THEN GOSUB 9600
40 3914 IF KYBD =1 THEN GOSUB 11600 : CSS=CHRS(MSEL)
3915 IF CSS="" THEN GOTO 3960
3917 IF ED=1 THEN RETURN
3920 IF NDET = 300 THEN PRINT "Too Many Symbols": STOP
3926 NDBT=NDBT-1 : DN(NDBT)=NST : DC(NDBT)=ASC(CSS)
45 3930 IF M3+NST > 800 THEN PRINT "Too Many Strokes" :STOP
3936 FOR M1=1 TO NST : FOR M2=1 TO 9
3940 D(M2,M1+M3)=DBC(M2,M1) : NEXT M2 : NEXT M1
3950 RETURN
50 3960 CSS=""
3980 RETURN
4000 REM CONCATENATE STROKES
4010 FOR L=2 TO NSTRK
4014 IF X(SNO(L-1)) < 0 THEN GOTO 4090
4016 IF SL(L-1) < 4 THEN GOTO 4100
4020 ALNO=(ABS(ABS(X(SNO(L)))-ABS(X(SNO(L-1))))+ABS(Y(SNO(L))-Y(SNO(L-1))))
4021 REM LOCATE 1,1: PRINT "L,ALNO",L,ALNO
4022 IF ALNO > 100 THEN GOTO 4090 : REM TOO FAR AWAY
4024 ALNO= 8*ALNO/(SNO(L)-SNO(L-1))
60 4026 DD =ABS(ABS(X(SNO(L-1)))-ABS(X(SNO(L-1)+2)))+ABS(Y(SNO(L-1))-Y(SNO(L-1)+2))
4027 REM PRINT "L,DD,ALNO",L,DD,ALNO
4028 IF DD > ALNO THEN GOTO 4090 : REM TOO FAR AWAY
4030 IR1=80*SC(L)\SL(L) : IR2=80*SC(L-1)\SL(L-1)
4031 REM PRINT "L,IR1,IR2",L,IR1,IR2

```

```

4032 IF ABS(IR1) > 350 THEN GOTO 4040 :REM SMALL WIGGLE
4034 IF ABS(IR1) < 80 THEN GOTO 4038 :REM STRAIGHT LINE
4036 IF ABS(IR1-IR2) > ABS(IR1+IR2)\3 THEN GOTO 4090 ELSE GOTO 4050 : REM TOO DI
FFERENT CURVATURE
5 4038 IF ABS(IR2) > 80 THEN GOTO 4036 ELSE IF IR1*IR2 < -2000 THEN GOTO 4090 :REM
TOO DIFFERENT CURVATURE
4039 GOTO 4050
4040 IF ABS(IR2) < 350 THEN GOTO 4036 ELSE IF IR1*IR2 < 0 THEN GOTO 4090
4050 ASTND=ABS(ASTRT(L)-AEND(L-1))
10 4051 REM PRINT "ASTND",ASTND
4052 IF ASTND > 128 THEN ASTND=ABS(ASTND-256): GOTO 4052
4054 IF ASTND > 64 THEN GOTO 4090 ELSE IF ASTND > DD*(ABS(IR1)+ABS(IR2))\40 THEN
GOTO 4090
4056 GOTO 4100
15 4090 NEXT L
4099 RETURN
4100 REM COMBINE STROKES
4101 REM PRINT "COMBINING",L,L-1
4110 L1=L-1 :SLL=SL(L1)+SL(L)
20 4115 IF SLL = 0 THEN LOCATE 23,1 :PRINT "TWO STROKES WITH NO LENGTH" :STOP
4120 IF SXX(L) > SXX(L1) THEN SXX(L1)=SXX(L)
4122 IF SXN(L) < SXN(L1) THEN SXN(L1)=SXN(L)
4124 IF SYX(L) > SYX(L1) THEN SYX(L1)=SYX(L)
4126 IF SYN(L) < SYN(L1) THEN SYN(L1)=SYN(L)
25 4130 SX(L1)=(SL(L1)*SX(L)+SL(L)*SX(L))/SLL
4133 SY(L1)=(SL(L1)*SY(L)+SL(L)*SY(L))/SLL
4137 SA(L1)=(SL(L1)*SA(L)+SL(L)*SA(L))/SLL
4140 SL(L1)=SLL +DD
4150 SC(L1)=SC(L1)+SC(L)+((SC(L1)+SC(L))*DD)\SLL
30 4160 SNO(L1)=SNO(L) :AEND(L1)=AEND(L)
4170 NSTRK=NSTRK-1 : IF NSTRK < L THEN RETURN
4180 FOR LL=L TO NSTRK
4185 L1=LL+1 :SXX(LL)=SXX(L1) :SXN(LL)=SXN(L1) :SYX(LL)=SYX(L1) :SYN(LL)=SYN(L1)
4187 SX(LL)=SX(L1) :SY(LL)=SY(L1) :SA(LL)=SA(L1) :SC(LL)=SC(L1) :SNO(LL)=SNO(L1)
4189 SL(LL)=SL(L1) :ASTRT(LL)=ASTRT(L1) :AEND(LL)=AEND(L1)
4190 NEXT LL
4195 GOTO 4010
4200 REM 2 POINT AVERAGE POINTS
4210 M2=0
40 4220 FOR M=2 TO NP STEP 2 :M1=M-1
4230 IF X(M1) > 0 GOTO 4240
4235 M2=M2+1 :X(M2)=X(M1) :Y(M2)=Y(M1) :GOTO 4280
4240 IF X(M) > 0 GOTO 4250
4245 M2=M2+1 :X(M2)=X(M) :Y(M2)=Y(M) :GOTO 4280
45 4250 IF ABS(X(M)-X(M1)) > 30 THEN GOTO 4280
4260 IF ABS(Y(M)-Y(M1)) > 30 THEN GOTO 4280
4270 M2=M2+1 :X(M2)=(X(M)+X(M1))\2 :Y(M2)=(Y(M)+Y(M1))\2
4280 NEXT M
4283 X(M2)=X(NP)
50 4285 NP=M2
4290 RETURN
4340 REM ORDER STROKES BY SXN
4342 XCH=0 :IF NSTRK < 2 THEN RETURN
4345 FOR L=2 TO NSTRK
55 4350 IF SXN(L) > SXN(L-1) THEN GOTO 4380
4355 XCH=1 :L1=L-1
4360 ST=SXX(L1) :SXX(L1)=SXX(L) :SXX(L)=ST
4361 ST=SXN(L1) :SXN(L1)=SXN(L) :SXN(L)=ST
4362 ST=SYN(L1) :SYN(L1)=SYN(L) :SYN(L)=ST
60 4363 ST=SYX(L1) :SYX(L1)=SYX(L) :SYX(L)=ST
4364 ST=SX(L1) :SX(L1)=SX(L) :SX(L)=ST
4365 ST=SY(L1) :SY(L1)=SY(L) :SY(L)=ST
4366 ST=SA(L1) :SA(L1)=SA(L) :SA(L)=ST

```

```

4367 ST=SC(L1) :SC(L1)=SC(L) :SC(L)=ST
4368 ST=SL(L1) :SL(L1)=SL(L) :SL(L)=ST
4369 ST=SNO(L1) :SNO(L1)=SNO(L) :SNO(L)=ST
4380 NEXT L
5 4390 IF XCH = 0 THEN RETURN ELSE XCH=0 :GOTO 4345
6100 REM LEARN
6105 LERN=1 :J=3: JJ=J : FOR I=1 TO 40 :ASS(I)=" " :NEXT I
6110 LINY=8*(JJ+3)-1
10 6120 LINE (0,LINY-LHT-8)-(639,LINY+LHT\2),0,BF
6130 LINE (8*IMARGN-1,LINY)-(639,LINY-LHT),1,B
6140 LINE (607,LINY)-(639,LINY-LHT),1,BF
6150 IF SWTPN = -1 THEN GOSUB 7100 ELSE GOSUB 8600 :REM TOUCH TECH SCREEN IN AS
CII @ 9600
6151 IF SWTPN = 1 THEN GOSUB 7300
15 6155 IF NP < 8 THEN GOTO 6190
6160 GOSUB 3000
6170 GOSUB 3600
6180 LOCATE 18,IMARGN: FOR I=1 TO 40: PRINT ASS(I);: NEXT I
6186 GOTO 6150
20 6190 FOR I=1 TO 4
6192 IF ASS(I) = "@" THEN RETURN
6194 NEXT I
6195 GOTO 6105
6200 REM INSERT
25 6201 XX=SXX(1): XN= SXN(1): YX=SYX(1): YN=SYN(1)
6202 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE CHARACTER POSITION
6203 I=INT(XP/8)+1: J=INT(YP/8)+1 :ISAV=I: JSAV=J: REM CONVERT X,Y TO I,J
6204 ML=LL(J): M1=ML+I-IMARGN :REM CALC A ARRAY INDEX OF LETTER AT I,J
6205 IF AS(ML)=" " THEN M1=M1+1
30 6206 IF AS(ML)=CRS THEN M1=M1+1
6207 MSAV=M1
6210 JJ=JSAV : FOR I=1 TO 40 :ASS(I)=" " :NEXT I
6215 LINY=8*(JJ+3)-1
6220 LINE (0,LINY-LHT-8)-(639,LINY-LHT\2),0,BF
6220 LINE (8*IMARGN-1,LINY)-(639,LINY-LHT),1,B
6240 LINE (607,LINY)-(639,LINY-LHT),1,BF
6250 IF SWTPN = -1 THEN GOSUB 7100 ELSE GOSUB 8600 :REM TOUCH TECH SCREEN IN AS
CII @ 9600
6251 IF SWTPN = 1 THEN GOSUB 7300
40 6255 IF NP < 8 THEN GOTO 6270
6256 GOSUB 3000
6257 GOSUB 3600
6258 IF DBUG (>) 0 THEN LOCATE 22,IMARGN: FOR I=1 TO 40: PRINT ASS(I);: NEXT I
6259 GOTO 6250
45 6270 JB=0: J=1
6271 REM LOCATE 16,1: FOR I=1 TO 40: PRINT ASS(I);: NEXT I
6272 IE=2 :FOR I=40 TO 2 STEP -1 : IF ASS(I) (>) " " THEN IE=I+1 : GOTO 6274
6273 NEXT I
6274 FOR I=2 TO IE
50 6275 IF ASS(I) (>) " " THEN GOTO 6280
6276 JB=JB+1
6278 IF JB < 3 THEN GOTO 6288
6280 J=J+1: JB=0
6282 ASS(J)=ASS(I)
55 6288 NEXT I
6289 REM LOCATE 17,1: FOR I=1 TO 40: PRINT ASS(I);: NEXT I
6290 ASS(J+1)=" "
6295 IF J < 2 THEN IF ASS(1) = " " THEN LOCATE 18,1 : PRINT "NO LETTERS, TRY AGAI
N" :GOTO 6210
60 6300 MD=J+1 : M3=MSAV : I=ISAV : J=JSAV
6310 REM LOCATE 20,1: FOR I=1 TO 40: PRINT ASS(I);: NEXT I
6315 REM INPUT IJK
6320 GOTO 1620

```

```

6500 REM REMOVE
6510 XX=SXX(1): XN=SXN(1): YX=SYX(1): YN=SYN(1)
6520 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE CHARACTER POSITION
6530 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
5 6540 ML=LL(J): M1=ML-I-IMARGN :REM CALC A ARRAY INDEX OF LETTER AT I,J
6550 IF AS(ML)=" " THEN M1=M1+1
6560 IF AS(ML)=CRS THEN M1=M1+1
6570 M2=M1
6590 GOTO 1250
10 6600 REM PARAGRAPH
6690 GOTO 200
6700 REM REPLACE A CHARACTER WHEN WRITTEN OVER
6710 XX=SXX(1): XN=SXN(1): YX=SYX(1): YN=SYN(1)
6720 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE CHARACTER POSITION
15 6730 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
6740 ML=LL(J): M1=ML-I-IMARGN :REM CALC A ARRAY INDEX OF LETTER AT I,J
6750 IF AS(ML)=" " THEN M1=M1+1
6760 IF AS(ML)=CRS THEN M1=M1+1
6770 AS(M1)=ASS(1) :LOCATE J,I :PRINT ASS(1)
20 6790 GOTO 200
6900 REM INSERT A LETTER
6910 XX=SXX(1): XN=SXN(1): YX=SYX(1): YN=SYN(1)
6920 XP=(XN+XX)/2: YP=(YN+YX)/2 :REM AVE CHARACTER POSITION
6930 I=INT(XP/8)+1: J=INT(YP/8)+1:REM CONVERT X,Y TO I,J
25 6940 ML=LL(J): M1=ML-I-IMARGN :REM CALC A ARRAY INDEX OF LETTER AT I,J
6950 IF AS(ML)=" " THEN M1=M1+1
6960 IF AS(ML)=CRS THEN M1=M1+1
6970 ASS(2)=ASS(1)
6980 M3=M1 :MD=1
6990 GOTO 1570
7000 REM SET UP RERUN
7010 OPEN "I",#1,"POINTS.DAT"
7020 INPUT #1,AX,BX,AY,BY
7030 RETURN
35 7100 REM READ POINTS
7110 INPUT #1,NP
7120 FOR N=1 TO NP
7130 INPUT #1,X(N),Y(N)
7132 NEXT N
7133 IF IFAVE = 1 THEN GOSUB 4200
7134 FOR N=1 TO NP
7135 IF X(N) > 0 THEN I=32*X(N)/AX+BX ELSE I=-32*X(N)/AX+BX
7136 J=32*Y(N)/AY+BY
7137 PSET(I,J)
45 7140 NEXT N
7148 IF DEBUG > 5 THEN FOR N=1 TO NP: LPRINT N,X(N),Y(N):NEXT N
7149 IF DEBUG <> 0 THEN LOCATE 20,2 :PRINT "NP=",NP
7150 RETURN
7200 REM SET UP STORE POINTS
50 7210 OPEN "C",#1,"POINTS.DAT"
7220 PRINT #1,AX,BX,AY,BY
7230 RETURN
7300 REM STORE POINTS
7310 PRINT #1,NP
7320 FOR N=1 TO NP
7330 PRINT #1, USING "#####"; X(N),Y(N)
7340 NEXT N
7350 RETURN
60 8600 REM TT ACQUIRE POINTS IN ASCII @ 9600 WITH INSERT
8603 M=20
8605 IF LOC(2) > 0 THEN INPUT$=INPUT$(1,2) : GOTO 8605
8606 PRINT #2,"G" :N=0
8610 IF LOC(2) < 40 THEN GOTO 8610

```



```

8620 NU=0: INPUT #2,XP,YP :INPUT #2,XP,YP :INPUT #2,XPA,YPA :INPUT #2,XPO,YPO
8630 N=N+1: XPA=(XPA+XPO)\2: YPA=(YPA+YPO)\2: XPO=XPA: YPO=YPA: X(N)=XPA: Y(N)=Y
PA
5 8635 IF 32*XPA\AX+BK > 607 THEN GOTO 8695
8640 INPUT #2,XP,YP
8653 XPA=(XP+3*XPA)\4: YPA=(YP+3*YPA)\4
8657 IF ABS(XPA-XPO)+ABS(YPA-YPO) < 3 THEN GOTO 8672
8660 I=32*XPA\AX+BK:J=32*YPA\AY+BY:PSET(I,J)
8665 IF I > 607 THEN GOTO 8695
10 8670 N=N+1 : X(N)=XPA :Y(N)=YPA: XPO=XPA: YPO=YPA
8672 IF LOC(2) > 9 THEN GOTO 8640
8675 MM=0
8680 MM=MM+1: IF LOC(2) > 9 THEN GOTO 8640 ELSE IF MM < M THEN GOTO 8680
8685 X(N)=-X(N): IF X(N) = 0 THEN X(N)=-1
15 8690 IF LOC(2) > 39 THEN GOTO 8620 ELSE NU=NU+1 :IF NU < 25000 THEN GOTO 8690
8695 PRINT #2,"S": NP=N
8696 IF DEBUG <> 0 THEN LOCATE 20,2 :PRINT "NP=",NP
8697 IF IFAVE = 2 THEN GOSUB 4200
8698 IF DEBUG <> 0 THEN LOCATE 20,12 :PRINT NP
20 8699 RETURN
9600 REM TT ACQUIRE POINTS IN ASCII @ 9600 WITH MENU
9603 M=20
9605 IF LOC(2) > 0 THEN IPUTS=INPUTS(1,2) : GOTO 9605
9606 PRINT #2,"G"
25 9610 II=0 :JJ=0 :N=0
9640 INPUT #2,XP,YP
9660 II=32*XP\AX+BK+II: JJ=32*YP\AY+BY-JJ: N=N+1
9670 IF N < 10 THEN GOTO 9674
9672 I=II\10: J=JJ\10: N=0 :II=0: JJ=0
30 9673 LINE (I+1,J+1)-(I-1,J-1),1,B
9674 IF LOC(2) > 9 GOTO 9640
9675 MM=0
9680 MM=MM+1: IF LOC(2) > 9 THEN GOTO 9640 ELSE IF MM < M THEN GOTO 9680
9690 IF LOC(2) > 9 THEN GOTO 9610 ELSE NU=NU+1 :IF NU < 1500 THEN GOTO 9690
35 9695 PRINT #2,"S"
9699 RETURN
10700 REM SET UP TOUCH SCREEN FOR ASCII @ 9600
10710 OPEN "I",#2,"CAL.DAT"
10720 INPUT #2,AX,BK,AY,BY
40 10730 CLOSE #2
10740 OPEN "COM1:9600,N,8,1,CS0,DS0" AS #2
10750 PRINT #2,"S"
10760 RETURN
11000 REM UTILITIES
45 11100 REM input selection menus
11101 REM mnum= no. of items
11102 REM mtitls=titles
11103 REM mi0,mie,mj0,mje=start & end points for boxes
11105 IF KYBD=1 GOTO 11400
50 11110 LINE(MI0,MJ0)-(MIE,MJE),0,BF :LINE(MI0,MJ0)-(MIE,MJE),1,B
11115 XX=MI0 : MRATO=(MIE-MI0)\MNUM
11120 FOR K=1 TO MNUM : LINE (XX,MJ0)-(XX,MJE)
11130 MJA=(MJ0+MJE+8)\16+1 : LOCATE MJA,(XX+20)\8 : PRINT MTITLS(K)
11140 XX=XX+MRATO :NEXT K
55 11160 RETURN
11300 REM DETERMINE MENU ITEM SELECTED
11310 IF KYBD=1 GOTO 11600
11320 XX=0 : FOR K=1 TO MNUM : XX=XX+MRATO : IF I < XX THEN GOTO 11340
11330 NEXT K
60 11340 MSEL=K
11350 RETURN
11400 REM DISPLAY KEYBOARD
11405 JMENU=163

```

```

11410 LINE(1,160)-(639,199),0,BF
11440 FOR I= 1 TO 26 :LOCATE 22,2*I : PRINT CHR$(I+64), : NEXT I
11450 FOR I= 1 TO 26 :LOCATE 24,2*I : PRINT CHR$(I+96), : NEXT I
11460 FOR I= 1 TO 10 :LOCATE 22,2*I+52 : PRINT CHR$(I+47), : NEXT I
5 11470 LOCATE 22,74 : PRINT ". ^ ? ",
11480 LOCATE 24,54 : PRINT "\ / * + - ( ) , ; : " :CHR$(34);" $ &";
11482 FOR J=163 TO 195 STEP 16 : LINE(3,J)-(629,J) :NEXT J
11484 FOR I= 3 TO 635 STEP 16 : LINE(I,163)-(I,195) :NEXT I
11490 LOCATE 1 , 1 :RETURN
10 11600 REM keyboard select from menu
11620 MSEL=(I\8+1)\2
11640 IF J > 179 THEN GOTO 11700
11650 IF MSEL < 27 THEN MSEL=MSEL+64 : RETURN
11660 IF MSEL < 37 THEN MSEL=MSEL+21 : RETURN
15 11670 IF MSEL=37 THEN MSEL=46 : RETURN
11680 IF MSEL=38 THEN MSEL=94 : RETURN
11690 MSEL=63 : RETURN
11700 IF MSEL < 27 THEN MSEL=MSEL+96 : RETURN
11710 IF MSEL=27 THEN MSEL=92 : RETURN
20 11720 IF MSEL=28 THEN MSEL=47 : RETURN
11730 IF MSEL=29 THEN MSEL=42 : RETURN
11740 IF MSEL=30 THEN MSEL=43 : RETURN
11750 IF MSEL=31 THEN MSEL=45 : RETURN
11760 IF MSEL=32 THEN MSEL=40 : RETURN
25 11770 IF MSEL=33 THEN MSEL=41 : RETURN
11780 IF MSEL=34 THEN MSEL=44 : RETURN
11790 IF MSEL=35 THEN MSEL=59 : RETURN
11800 IF MSEL=36 THEN MSEL=58 : RETURN
11810 IF MSEL=37 THEN MSEL=34 : RETURN
30 11820 IF MSEL=38 THEN MSEL=36 : RETURN
11830 MSEL=38 : RETURN
11850 RETURN
12000 REM input output
12100 REM read in text
12110 OPEN "I", #1,"TEXT" :NC=0
12120 IF EOF(1) THEN CLOSE #1 : GOTO 12160
12130 INPUT #1, A1$ :KK=0
12140 FOR K=1 TO LEN(A1$) : KK=KK+1 :AS(NC-KK)=MIDS(A1$,K,1)
12143 IF LEN(AS(NC-KK))=0 THEN KK=KK-1
40 12145 NEXT K
12150 NC=NC+KK:IF NC < 1500 THEN GOTO 12120
12155 PRINT "TOO MUCH TEXT" : STOP
12160 REM TEXT EDITOR INITIALIZATION
12170 II=IMARGN : JJ=JTOP : NC=1 : REM DISPLAY ALL TEXT
45 12175 CLOSE #1
12180 RETURN
12500 REM SAVE SYMBOL DATA BASE
12505 IF SWTPN (<) 0 THEN CLOSE #1
12510 OPEN "C",#1,"SYM.DAT"
50 12520 PRINT #1,NDBT
12530 M2=0 :FOR M=1 TO NDBT
12540 PRINT #1, DN(M) DC(M)
12550 FOR M1=M2+1 TO M2+DN(M)
12560 PRINT #1, D(1,M1) D(2,M1) D(3,M1) D(4,M1) D(5,M1) D(6,M1) D(7,M1)
55 12570 NEXT M1 :M2=M2+DN(M) : NEXT M
12580 CLOSE #1
12590 END
12700 REM RESTART READ IN DATA BASE
12710 OPEN "I",#1,"SYM.DAT"
60 12720 INPUT #1,NDBT
12730 M2=0 :FOR M=1 TO NDBT
12740 INPUT #1, DN(M),DC(M)
12750 FOR M1=M2+1 TO M2+ DN(M)

```

```
12760 INPUT #1,D(1,M1),D(2,M1),D(3,M1),D(4,M1),D(5,M1),D(6,M1),D(7,M1)
12770 NEXT M1 :M2=M2+DN(M) : NEXT M
12780 CLOSE #1
12790 RETURN
5 30000 FOR I=1 TO NSTRK:PRINT I,SNO(I),SX(I),SY(I),SL(I),SA(I),SC(I):NEXT I
30010 FOR I=1 TO NP:PRINT I,X(I),Y(I):NEXT I
30020 FOR I=1 TO NSTRK:PRINT USING "#####":I,SNO(I),SX(I),SY(I),SL(I),SA(I),SC(I)
):NEXT I
10 30030 FOR M1=1 TO NST :FOR M2=1 TO 5 :PRINT M1,M2,D(M2,M1),DBC(M2,M1):NEXT M2 :N
EXT M1
```

ANNEXE III

- A1 - angle de pente entre les Points M et M1  
 A2 - angle précédent  
 DA1 - variation d'angle (A1-A2)  
 SEND - indicateur de fin d'un Trait de Tracé
- 5 ZX - sommation des positions X à double précision par pondération des longueurs d'arc  
 ZY - sommation des positions Y à double précision par pondération des longueurs d'arc  
 ZA - sommation des angles à double précision par pondération des longueurs
- 10 d'arc  
 ZDA - variation de la sommation des angles à double précision  
 ZL - longueur à double précision  
 NSP - nombre de Points dans un Trait de Tracé, compteur  
 NSTRK - nombre de Traits de Tracé
- 15 XX - X maximum  
 XN - X minimum  
 YX - Y maximum  
 YN - Y minimum  
 M1 - Pointeur pour le Point précédent
- 20 M2 - (1) nombre minimum de Points pour lequel il faut ajuster pour un angle de  $2\pi$   
 M3 - (2) nombre minimum de Points pour lequel il faut calculer la variation de la variation d'angle  
 NLONG - compteur (non utilisé)
- 25 M - Pointeur pour le Point pris en considération  
 NP - Nombre de Points  
 X - élément de rangement des coordonnées X  
 Y - élément de rangement des coordonnées Y  
 DX - variation de X
- 30 DY - variation de Y  
 ADX - valeur absolue de DX  
 ADY - valeur absolue de DY  
 DS - pseudo longueur d'arc  $ADX + ADY$   
 AY - multiplicateur d'étalonnage pour la direction verticale (Y), par exemple  $AY > 0$  signifie système de coordonnées vrai vers le bas
- 35 ADA - variation moyenne courante d'angle  
 DA2 - variation précédente d'angle

- ASTRT - élément de rangement de l'angle de début du Trait de Tracé  
SZL - longueur de Trait de Tracé à simple précision  
SNO - élément de rangement du numéro du premier Point pour chaque Trait de Tracé
- 5 LHT5 - hauteur entre lignes divisée par 5 en coordonnées de visualisation utilisée comme mesure  
SL - élément de rangement des longueurs de Trait de Tracé rapportées à la hauteur entre lignes  
AEND - élément de rangement des angles à la fin des Traits de Tracé  
SC - élément de rangement des variations d'angle pour chaque Trait de Tracé  
AZA - angle moyen d'un bout à l'autre d'un Trait de Tracé  
SX - élément de rangement des coordonnées X moyennes du centre de gravité pour chaque Trait de Tracé  
SY - élément de rangement des coordonnées Y moyennes du centre de gravité
- 15 pour chaque Trait de Tracé  
SA - élément de rangement des coordonnées moyennes d'angle (pente) pour chaque Trait de Tracé  
SXX - élément de rangement de la coordonnée X maximale pour chaque Trait de Tracé
- 20 SXN - élément de rangement de la coordonnée X minimale pour chaque Trait de Tracé  
SYN - élément de rangement de la coordonnée Y minimale pour chaque Trait de Tracé  
SYX - élément de rangement de la coordonnée Y maximale pour chaque Trait de Tracé
- 25 DEBUG - indicateur de mise au point pour l'impression  
INPUT - chaîne d'octets d'entrée à rejeter  
XP, YP - coordonnées d'entrée  
XPA - moyenne de deux Points suivant la direction X  
YPA - moyenne de deux Points suivant la direction Y
- 30 XPO - moyenne X initiale pour le crayon posé  
YPO - moyenne Y initiale pour le crayon posé  
I - point de visualisation correspondant à l'entrée de coordonnée X  
J - point de visualisation correspondant à l'entrée de coordonnée Y
- 35 N - compteur de Points d'entrée  
MM - compteur pour mesurer la durée avant relèvement du crayon  
NU - compteur pour mesurer la durée avant relèvement du crayon.

REVENDICATIONS

1. Dispositif d'entrée/sortie ou E/S permettant, d'une part, de fournir à un dispositif utilisateur (14, 44) des signaux d'entrée qui sont représentatifs d'un caractère graphique produit par un utilisateur et permettant, d'autre part de recevoir des signaux de sortie à partir du dispositif utilisateur (14, 44) et de fournir une représentation visuelle de ces signaux de sortie, ce dispositif E/S comprenant :

- (a) un moyen de visualisation (20) permettant de fournir une représentation visuelle d'un caractère graphique en réponse à des signaux de sortie fournis par le dispositif d'utilisation (14, 44),
- 10 (b) un ensemble à écran d'entrée permettant de produire un train de signaux d'entrée au fur et à mesure qu'un utilisateur trace un signe ou caractère graphique, cet ensemble à écran d'entrée comprenant un écran d'entrée (18, 34) translucide présentant des surfaces supérieure et inférieure, la surface inférieure étant disposée par rapport à la surface de visualisation d'une façon telle que la représentation visuelle produite par ledit moyen de visualisation (20) soit visible du côté de la surface supérieure de cet écran d'entrée (18, 34), et
- 15 (c) des moyens permettant de déterminer l'identité dudit caractère graphique et de fournir au moyen de visualisation (20) des signaux de sortie représentatifs de ce caractère.
- 20

2. Dispositif E/S selon la revendication 1, caractérisé en ce que le moyen de visualisation (20) comporte une surface de visualisation sur laquelle est produit le caractère graphique, et en ce que la surface inférieure de l'ensemble à écran d'entrée est disposée en proximité étroite au-dessus de ladite surface de visualisation de façon que la représentation visuelle produite par le moyen de visualisation (20) soit visible à partir du côté supérieur de l'écran d'entrée (18, 34).

25

3. Dispositif E/S selon la revendication 1, caractérisé en ce que l'ensemble à écran d'entrée comprend une plaque sensiblement plate (18, 34), et en ce que la surface supérieure de l'écran d'entrée est la surface supérieure de cette plaque (18, 34).

30

4. Dispositif E/S selon la revendication 3, caractérisé en ce qu'il comprend en outre un moyen indicateur (16, 35) manoeuvré par un utilisateur de façon qu'il coopère avec la surface supérieure de l'écran d'entrée (18, 34) d'une manière telle que l'ensemble écran d'entrée produit ledit train de signaux d'entrée lorsque ce moyen indicateur (16,35) est dans une position préfixée par rapport à la surface supérieure de la plaque (18,34).

35

5. Dispositif E/S selon la revendication 4, caractérisé en ce que l'ensemble à écran d'entrée produit un signal d'entrée lorsque le moyen indicateur (16,35) est au contact de la surface supérieure de l'écran d'entrée (18, 34).

5 6. Dispositif E/S selon la revendication 5, caractérisé en ce que :  
(a) la plaque (18,34) présente des caractéristiques électriques telles que la tension en un Point quelconque situé sur celle-ci varie d'une manière connue avec la distance par rapport à l'emplacement (37) où une tension lui est appliquée,

10 (b) l'ensemble à écran d'entrée comporte en outre des moyens permettant d'appliquer une tension, de façon sélective et de manière séquentielle, le long d'un axe cartésien de la plaque (18, 34), puis le long de l'autre axe cartésien de cette plaque (18,34), et

15 (c) le moyen indicateur (16, 35) est allongé et présente à l'une de ses extrémités un embout (38), tandis qu'il comporte un moyen de détection de tension qui fournit un signal représentatif de la tension présente à l'endroit où ledit embout (38) vient au contact de la plaque (18, 34) formant l'écran d'entrée.

7. Dispositif E/S selon la revendication 5, caractérisé en ce que :

20 (a) la plaque (18, 34) présente des caractéristiques électriques telles que la tension en un Point quelconque situé sur celle-ci varie d'une manière connue avec la distance à partir de l'emplacement (37) où une tension lui est appliquée,

25 (b) l'ensemble à écran d'entrée comporte en outre des moyens permettant de réaliser un échantillonnage, à l'endroit d'une origine située sur lui, de façon sélective et de manière séquentielle, pour la tension existant le long d'un axe cartésien de la plaque (18, 34), puis le long de l'autre axe cartésien de cette plaque (18, 34), et

30 (c) le moyen indicateur (16) est allongé et comporte à l'une de ses extrémités un embout (38), tandis qu'il comprend un moyen (36) permettant d'appliquer une tension connue sur cet embout (38), lesdits moyens d'échantillonnage détectant les tensions présentes à l'endroit de ladite origine lorsque l'embout (38) vient au contact de la plaque (18, 34) formant l'écran d'entrée en fournissant un signal représentatif des tensions détectées.

35 8. Dispositif E/S selon la revendication 2, caractérisé en ce que l'écran de visualisation (20) présente une zone de visualisation (24) sur laquelle des signes graphiques peuvent être produits et une zone d'introduction du type touches (28) à l'aide de laquelle des signaux d'instruction destinés au dispositif d'utilisation (14, 44) peuvent être produits lorsqu'un

contact est réalisé avec la zone correspondante située au-dessus d'elle sur l'écran d'entrée (18,34).

9. Appareil de reconnaissance de caractères manuscrits, caractérisé en ce qu'il comprend :

- 5 (a) un moyen de visualisation (20) présentant une surface de visualisation permettant d'offrir une représentation visuelle de caractères graphiques en réponse à un groupe de signaux de visualisation,
- (b) une surface à écrire (18, 34) destinée à recevoir des caractères graphiques tracés de manière manuscrite par un utilisateur, cette surface à écrire  
10 (18, 34) étant disposée en proximité étroite par rapport au moyen de visualisation (20) et étant constituée d'une surface sensiblement translucide à laquelle sont associées des coordonnées de position (X, Y),
- (c) un moyen indicateur (16, 35) qu'un utilisateur fait fonctionner en liaison avec la surface à écrire (18, 34) de façon à écrire des signaux d'écriture en liaison avec cette surface à écrire (18, 34),  
15 (d) des moyens de détection de position (37-38) permettant de convertir les signaux d'écriture en signaux de localisation représentatifs des coordonnées de position (X, Y) du moyen indicateur (16, 35) par rapport à la surface à écrire (18, 34),
- 20 (e) des moyens de traitement de données (14, 44) destinés à recevoir et stocker les signaux de localisation au fur et à mesure qu'ils sont produits et à créer lesdits signaux de visualisation de façon que le moyen de visualisation (20) puisse représenter graphiquement le caractère graphique après qu'il ait été produit par le moyen indicateur (16, 35), ces moyens de traitement de données (14, 44) étant constitués par des moyens de reconnaissance  
25 de caractères permettant de comparer les signaux de localisation produits représentatifs d'un caractère graphique, à un groupe de signaux de base de données en mémoire, afin d'identifier ledit caractère graphique.

10. Appareil de reconnaissance de caractères manuscrits selon la revendication 9, caractérisé en ce que la surface à écrire (18, 34) comprend  
30 une plaque sensiblement plate qui produit un signal lorsque le moyen indicateur (16, 35) est à son contact.

11. Appareil de reconnaissance de caractères manuscrits selon la revendication 9, caractérisé en ce qu'il comporte en outre des moyens permettant de convertir les coordonnées de position (X, Y) en coordonnées de relation.  
35

12. Appareil de reconnaissance de caractères manuscrits selon la revendication 11, caractérisé en ce qu'il comprend en outre des moyens permettant de diviser un train de signaux de localisation en sections repré-



sentatives d'un Trait de Tracé, et des moyens permettant de calculer des caractéristiques de Trait de Tracé qui peuvent être comparées à des caractéristiques de Trait de Tracé appartenant aux signaux de la base de données.

5           13. Appareil de reconnaissance de caractères manuscrits selon la revendication 9, caractérisé en ce que les moyens de traitement de données (14, 34) comprennent en outre des moyens permettant de mettre à jour les signaux de base de données en mémoire, à l'aide desdits signaux de localisation produits, lorsqu'aucun caractère graphique correct n'est identifié.

10           14. Procédé de reconnaissance d'un Signe Manuscrit, caractérisé en ce qu'il consiste :

(a) à émettre un train de signaux de localisation pour des Points obtenus en déplaçant, à l'endroit d'emplacements situés à proximité d'une surface à écrire (18, 34), un stylet (16, 35) qui émet un signal représentatif de l'emplacement de ce stylet,

(b) à diviser un train de signaux de localisation en sections représentatives d'un Trait de Tracé,

(c) à calculer des caractéristiques de Trait de Tracé,

(d) à comparer ces caractéristiques de Trait de Tracé calculées à des caractéristiques de Trait de Tracé précédemment mises en mémoire dans une base de données, et

(e) à déterminer la meilleure comparaison et à indiquer si cette meilleure comparaison est suffisamment bonne pour être une concordance.

25           15. Appareil de reconnaissance de Signes Manuscrits et de visualisation de Signes Manuscrits et de Signes de Police de Caractères sur un écran, caractérisé en ce qu'il comprend :

(a) un écran de visualisation (20) présentant l'aptitude graphique de représenter des Signes de Police de Caractères et d'exécuter des instructions préfixées,

30           (b) des moyens (16, 35) que l'on tient à la main pour écrire ou tracer des Signes Manuscrits sur ou par-dessus ledit écran (20),

(c) des moyens (66) pour permettre la visualisation, sur l'écran (20), d'une représentation vraie des Signes Manuscrits au fur et à mesure qu'ils sont créés,

35           (d) des moyens de numérisation (43) pour détecter la position des moyens (16, 35) tenus à la main et pour convertir celle-ci en une série de signaux électriques déterminant la position, la taille et la forme de chaque Signe Manuscrit,

(e) des moyens (82) pour comparer des caractéristiques préfixées de chaque Signe Manuscrit numérisé, à une base de données de caractéristiques préfixées de Signes de Police de Caractères de façon à évaluer l'identité de chaque Signe de Police de Caractères, et

5 (f) des moyens (200) pour convertir le Signe Manuscrit en un Signe de Police de Caractères préfixé ou en une instruction et pour, suivant le cas, permettre la visualisation de ce Signe de Police de Caractères sur l'écran (20), à proximité étroite de la zone de cet écran sur laquelle le Signe Manuscrit a été introduit à l'origine, ou pour exécuter l'instruction.

10 16. Appareil selon la revendication 15, caractérisé en ce qu'il comprend des moyens permettant de visualiser des Signes de Police de Caractères préfixés, sous la forme de texte ou de formulaires, sur ledit écran (20), de sorte que les Signes Manuscrits peuvent être utilisés pour introduire des informations afin de remplir un formulaire ou d'éditer un texte  
15 préfixé.

17. Appareil selon la revendication 15, caractérisé en ce qu'il comprend une unité portative présentant des dimensions atteignant environ 40,64 cm x 40,64 cm x 10,16 cm, un poids atteignant environ 0,453 kg et en ce qu'il est agencé de façon à comporter une source autonome de courant  
20 (36).

18. Appareil selon la revendication 15, caractérisé en ce qu'il comprend des moyens permettant de visualiser sur l'écran (20) des signes préfixés faisant partie de Signes de Police de Caractères et de signes représentant des instructions, de façon à permettre à des signes tracés à la  
25 main représentant chaque signe préfixé d'être définis manuellement sur l'écran, et des moyens permettant d'identifier le signe défini manuellement au signe préfixé correspondant situé dans la base de données.

19. Appareil selon la revendication 18, caractérisé en ce qu'il comprend des moyens permettant de modifier la base de données lorsqu'un Signe Manuscrit provoque la visualisation sur l'écran d'un Signe de Police de Caractère erroné ou l'exécution d'une instruction erronée suivant le cas.  
30

20. Appareil selon la revendication 16, caractérisé en ce qu'il comprend des moyens permettant de créer une fenêtre d'espace libre sur l'écran (20) à l'occasion d'une instruction préfixée et en vue d'introduire des Signes Manuscrits dans cette fenêtre.  
35

21. Appareil selon la revendication 15, caractérisé en ce qu'il comprend des zones (28) situées sur l'écran et sensibles au toucher de façon à provoquer l'exécution de fonctions préfixées.

22. Appareil selon la revendication 15, caractérisé en ce que

l'écran (18) est sensiblement plat et agencé de façon à être utilisé suivant une position sensiblement horizontale.

23. Appareil selon la revendication 15, caractérisé en ce qu'il comprend des Touches Programmables (28) situées sur l'écran (18, 34) et des moyens permettant d'effectuer des instructions d'exécution lorsque l'une quelconque de ces Touches Programmables (28) a été touchée.

24. Procédé de traitement commandé par microprocesseur et permettant la reconnaissance, le transcodage et la visualisation de Signes Manuscrits et l'exécution d'instructions, caractérisé en ce qu'il consiste :

- 10 (a) à créer une base de données personnalisée pour chaque utilisateur individuel du procédé de traitement, en faisant introduire initialement à l'utilisateur un Signe Manuscrit pour chaque caractère correspondant à une police de caractères à visualiser ou à une instruction à exécuter,
- 15 (b) à déterminer un jeu unique de caractéristiques de façon à caractériser chaque signe de ce type et à le mettre en mémoire dans la base de données ,
- (c) à écrire ou tracer des Signes Manuscrits avec un stylet (16, 35) sur un écran d'ordinateur (18, 34)
- (d) à numériser (43, 78) chaque Signe Manuscrit de façon identifier les coordonnées (X, Y) d'une multiplicité de Points définissant le Signe,
- 20 (e) à traiter les caractéristiques numérisées de chaque Signe Manuscrit de façon à déterminer les caractéristiques préfixées du Signe,
- (f) à explorer la base de données afin de trouver « un succès », une correspondance parfaite ou la correspondance la plus proche par rapport aux caractéristiques, et
- 25 (g) à réaliser la visualisation du Signe de Police de Caractères ou à exécuter l'instruction, le ou la plus étroitement associé(e) aux caractéristiques du « succès ».

25. Procédé selon la revendication 24, caractérisé en ce qu'il consiste en outre :

- 30 (a) à déterminer si le Signe de Police de Caractères visualisé ou l'instruction exécutée est une erreur, et,
- (b) s'il existe une erreur, à réintroduire un Signe Manuscrit à la place du Signe de Police de Caractères ou de l'instruction voulu(e) de façon à modifier la base de données.

35 26. Procédé selon la revendication 24, caractérisé en ce qu'il consiste en outre :

- (a) à visualiser le Signe Manuscrit sur l'écran (20) de manière sensiblement simultanée à son écriture ou son tracé.

27. Procédé selon la revendication 24, caractérisé en ce que les opérations (b) et (c) impliquent une détermination de (i) la longueur, (ii) la pente moyenne, (iii) la hauteur du centre de gravité au-dessus de la ligne de base, (iv) la courbure (taux de variation de la pente) et (v) la comparaison de la localisation du centre de gravité de chaque Trait de Tracé avec le centre de gravité du Signe Manuscrit.

28. Procédé selon la revendication 24, caractérisé en ce que l'opération (f) consiste :

(a) à déterminer la valeur de chacune des cinq caractéristiques,

(b) à soustraire de la valeur nouvellement mesurée la valeur de la caractéristique située dans la base de données,

(c) à appliquer une fonction linéaire aux valeurs absolues de chaque mesure de façon à rendre chacune des cinq mesures raisonnablement équivalente aux autres,

(d) à additionner l'ensemble des cinq valeurs,

(e) si un seuil préfixé est dépassé, à rejeter la comparaison et à créer un message d'erreur, et

(f) à explorer la base de données pour trouver un « succès » numérique ou, si cela n'est pas faisable, la coïncidence numérique la plus proche entre la valeur de la base de données et la valeur nouvellement introduite.

29. Procédé selon la revendication 24, caractérisé en ce que le taux d'erreur est de 5 % ou moins.

30. Procédé selon la revendication 28, caractérisé en ce que le programme de l'opération (f) exige moins de 6K de mémoire machine.

31. Procédé selon la revendication 24, caractérisé en ce que les Signes Manuscrits comprennent des Signes d'Edition et en ce que ce procédé comporte la reconnaissance de ces Signes d'Edition tracés sur l'écran (18, 34) et l'exécution des Fonctions d'Edition représentées.

32. Procédé selon la revendication 24, caractérisé en ce qu'un texte en format de Signes de Police de Caractères est visualisé sur l'écran (20) et en ce que le procédé consiste :

(a) à créer sur l'écran (20) une fenêtre située à proximité d'une zone dudit texte à éditer, mais sans y être superposée,

(b) à introduire et visualiser lesdits Signes Manuscrits dans ladite fenêtre,

(c) à visualiser en proximité étroite par rapport aux Signes Manuscrits les Signes de Police de Caractères correspondant à ces Signes Manuscrits.

33. Procédé selon la revendication 24, caractérisé en ce qu'il consiste (a) à créer une ou plusieurs Touches Programmables (28) sur l'écran

(20) de façon à faire exécuter à l'ordinateur (14, 44) des fonctions d'exploitation, et

(b) à toucher une ou plusieurs de ces Touches Programmables (28) de façon à exécuter les fonctions correspondantes.

5

34. Procédé selon la revendication 24, caractérisé en ce qu'il consiste en outre à déterminer les coordonnées (X, Y) d'environ 39,37 à 78,74 Points par centimètre et d'environ 100 à 200 points par seconde pour caractériser les Signes Manuscrits.

TRADUCTION DES MENTIONS DES DESSINS

- Fig. 1 : micro-ordinateur
- Fig. 2 : nom - adresse
- Fig. 3 : écran d'entrée - interface et multiplexeur - convertisseur analogique/numérique - microcontrôleur - connexion
- Fig. 4 : RAM - microprocesseur - ROM - interface RS 232 - circuit d'interface d'écran d'entrée (microcontrôleur) - interface de visualisation - écran d'entrée - écran de visualisation - batterie au lithium - stylet - groupe de batteries
- 10 Fig. 5 : départ - crayon posé - non - oui - convertir tensions X,Y en signaux de localisation X, Y numériques - convertir signaux de localisation X, Y en caractéristiques de Trait de Tracé - comparer caractéristiques de Trait de Tracé à base de données - symbole reconnu - non - oui - requérir plus d'informations - visualiser signe - fin
- 15 Fig. 6 : système d'exploitation ordinateur - programme d'application (p.e. éditeur) - fonctions d'édition - programme reconnaissance de tracés - sous-programme codage Points X, Y en Traits de Tracé - sous-programme caractérisation de Traits de Tracé - sous-programme comparaison de Traits de Tracé - sous-programme mise à jour comparaison Traits de Tracé
- 20 Fig. 7 - Points X, Y - incorporer dans Traits de Tracé - caractériser Traits de Tracé (longueur, pente, courbure, etc.) - reconnaître caractères - trouvé - comparer Trait de Tracé et séquence Traits de Tracé avec base de données - non trouvé - brasser base de données pour donner priorité usage signe - demander utilisateur ajouter à base de données
- 25 Fig. 8 A : départ - lire tensions X, Y - crayon relevé - oui - non - (Trait de Tracé effectué) - < 3 Points dans Trait de Tracé - appliquer fonction linéaire à tension - oui - zéro Points - non - sous-programme caractérisation Traits de Tracé - filtrer Points erronés - oui - non - identifier Trait de Tracé comme point - incrémenter compteur Points - calculer hauteur moyenne - lisser Points - comparer Trait(s) de Tracé avec entrée base de données - mettre Point en mémoire - oui - non-Traits de Tracé # différent
- 30 - oui - HALB myn Point > ½ ligne - prendre Point suivant - non - sélectionner nouvelle entrée base de données - oui - # PTS. diffère > ID - non
- 35 Fig. 8B : calculer étalon - sauvegarder comparaison minimale - non - dernière entrée base de données - oui - non - coïncidence - oui - non - dernier Trait de Tracé - oui - brasser base de données - demander utilisateur si Trait de Tracé doit être ajouté à base de données - zéro Point et in-

crémenter compteur - non - ajouter à base de données - oui - visualiser caractère à position Trait de Tracé - ajouter au bas de base de données - exécuter instructions.

5 Fig. 9 : sous-programme caractérisation Trait de Tracé - calculer pente Point par Point - calculer hauteur Point au-dessus ligne de base (HALB) - normaliser hauteur et pente Point - mettre en mémoire HALB et pente de chaque Point dans élément de rangement - retour.

10 Fig. 13 : départ - acquérir Points et simuler sur écran - convertir en Trait de Tracé - caractériser Trait de Tracé - consulter/reconnaître caractère - envoyer caractère à logiciel éditeur pour interpréter et exécuter instructions - oui - signe ou fonction d'édition - non - déterminer quelle instruction et exécuter - visualiser caractère -

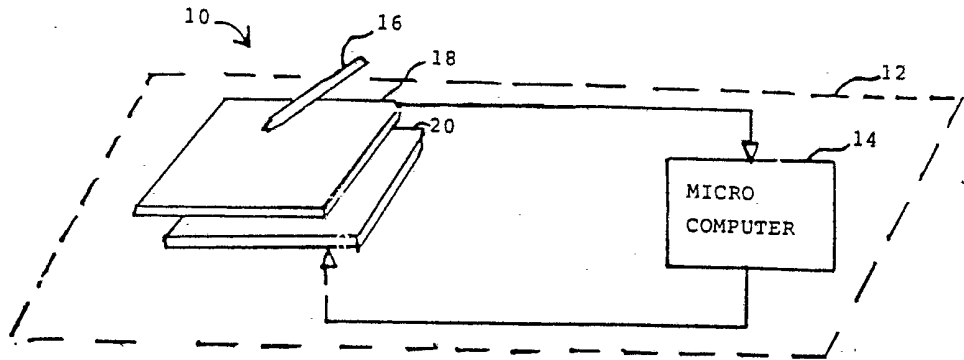


FIG. 1

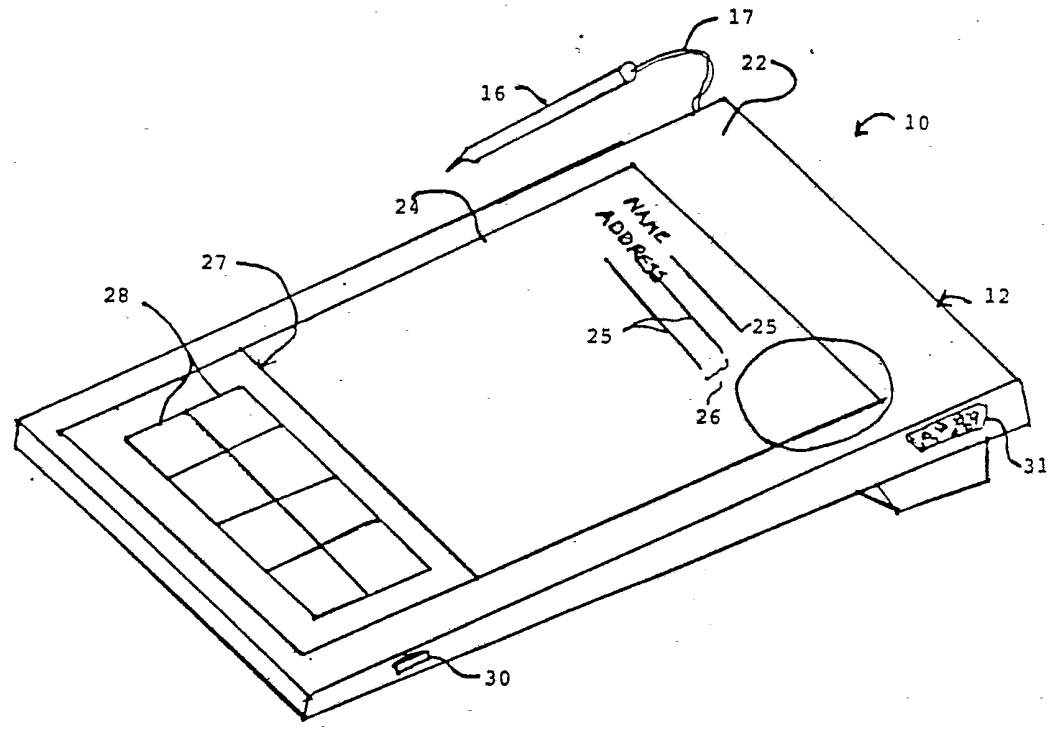


FIG. 2

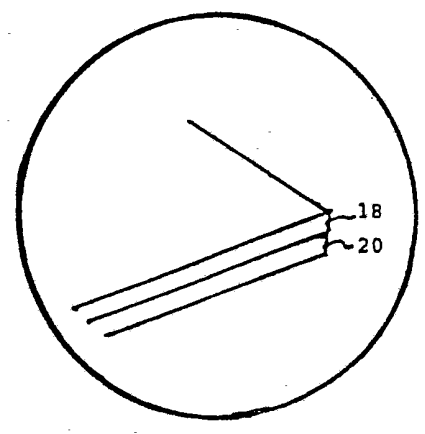


FIG. 2A



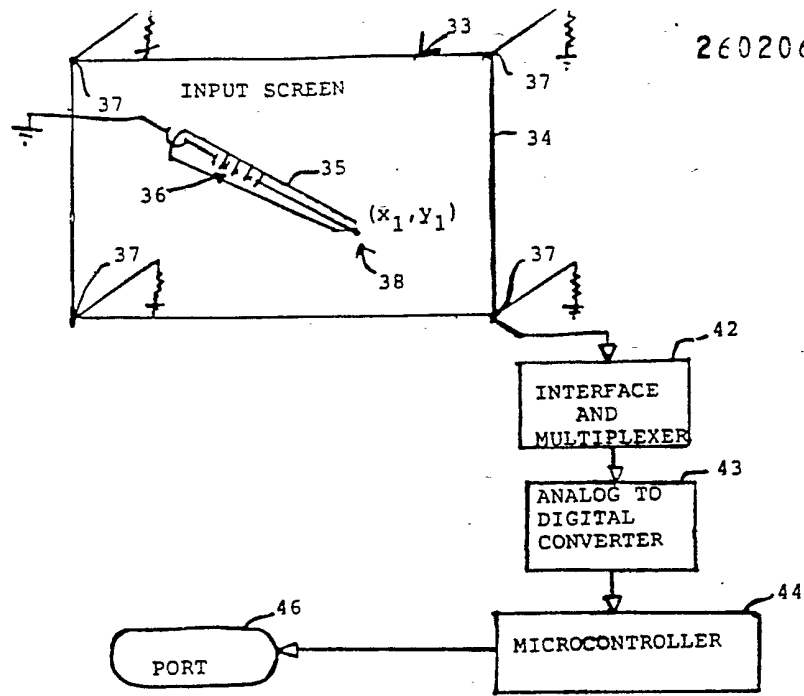


FIG. 3

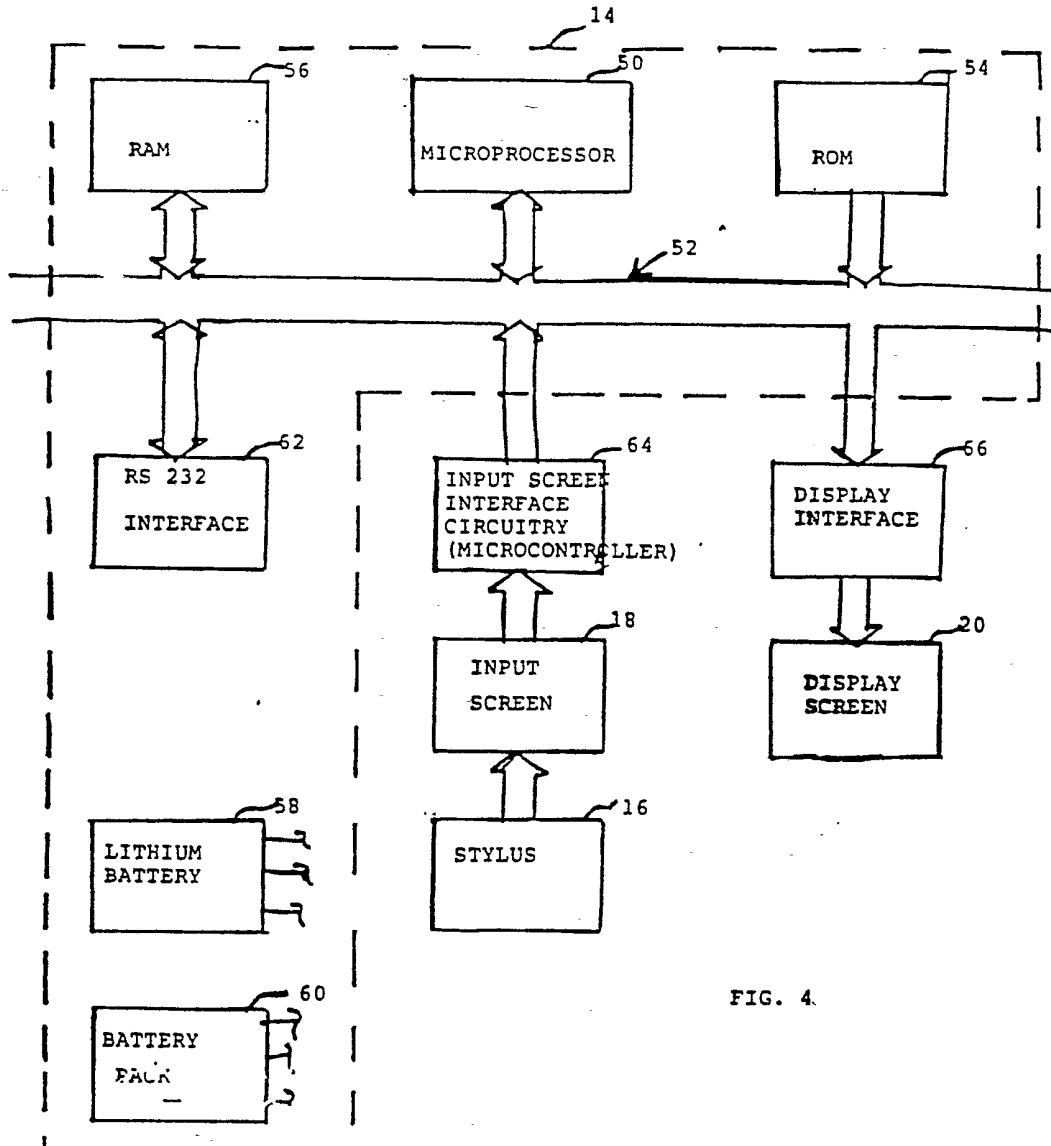


FIG. 4

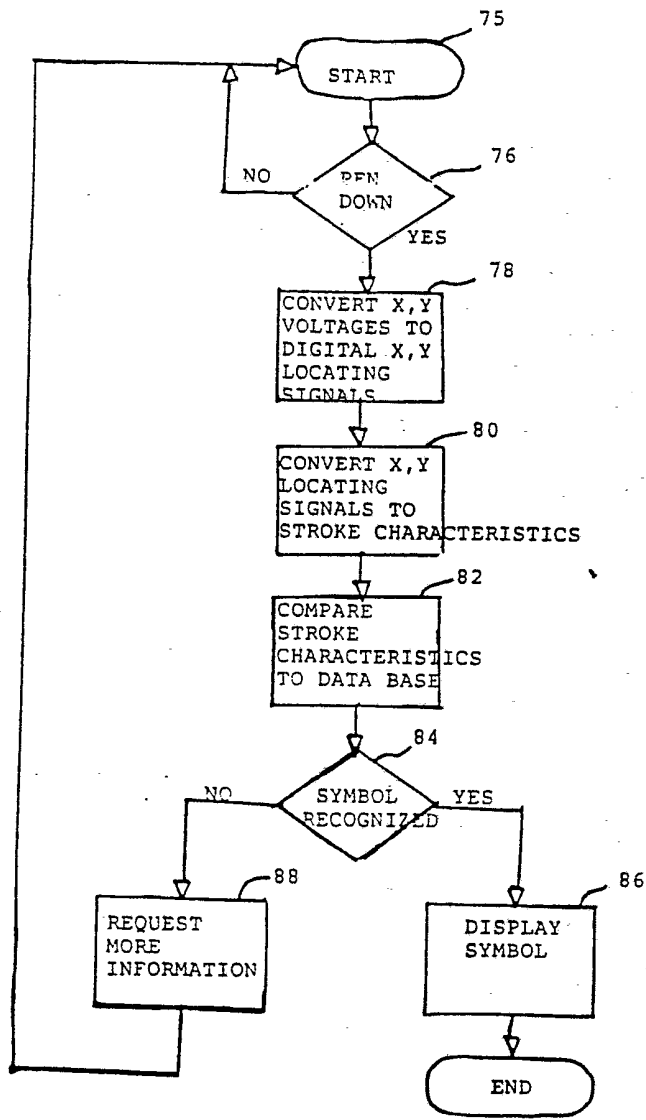
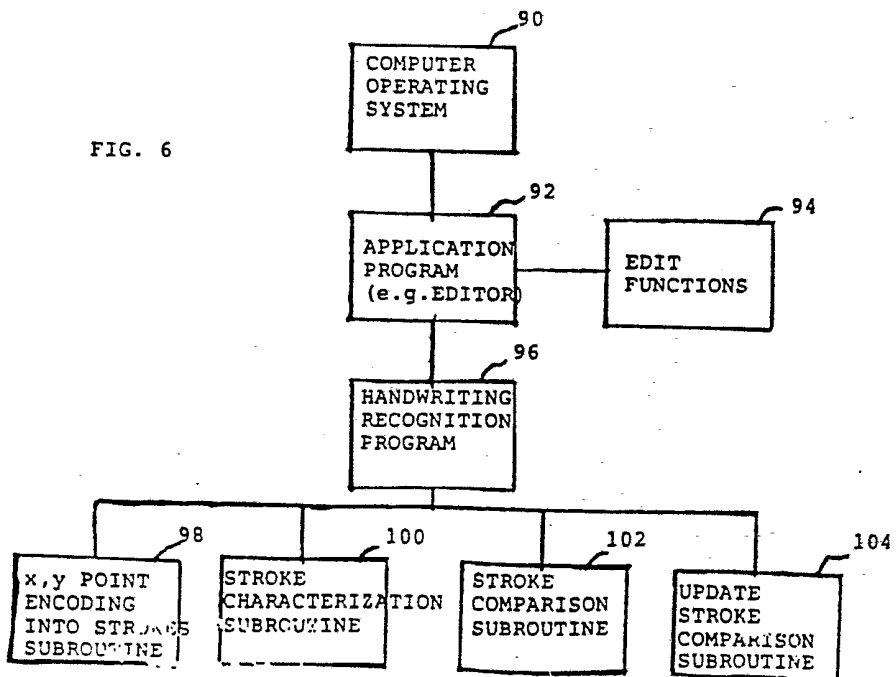


FIG. 5

FIG. 6



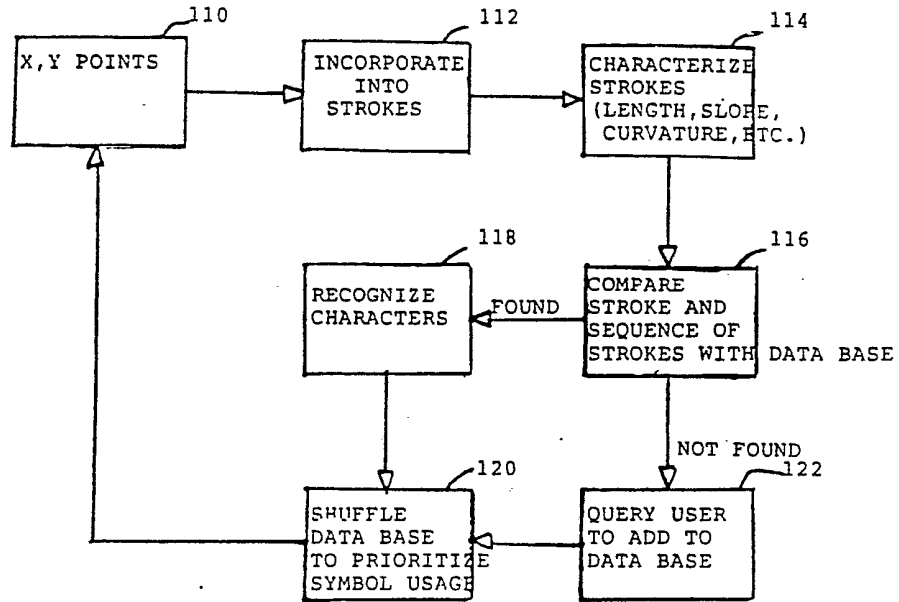


FIG. 7

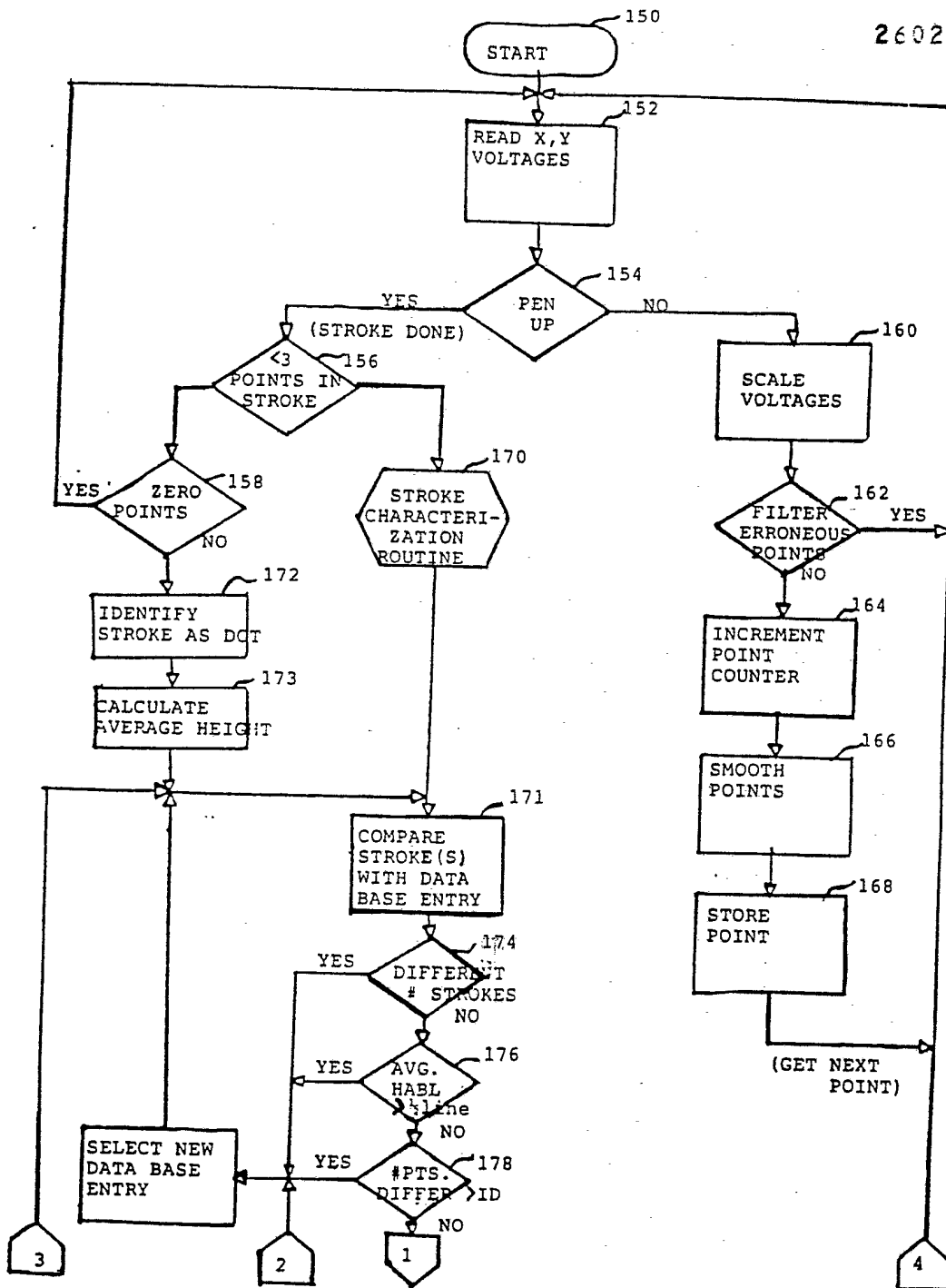


FIG. 8A

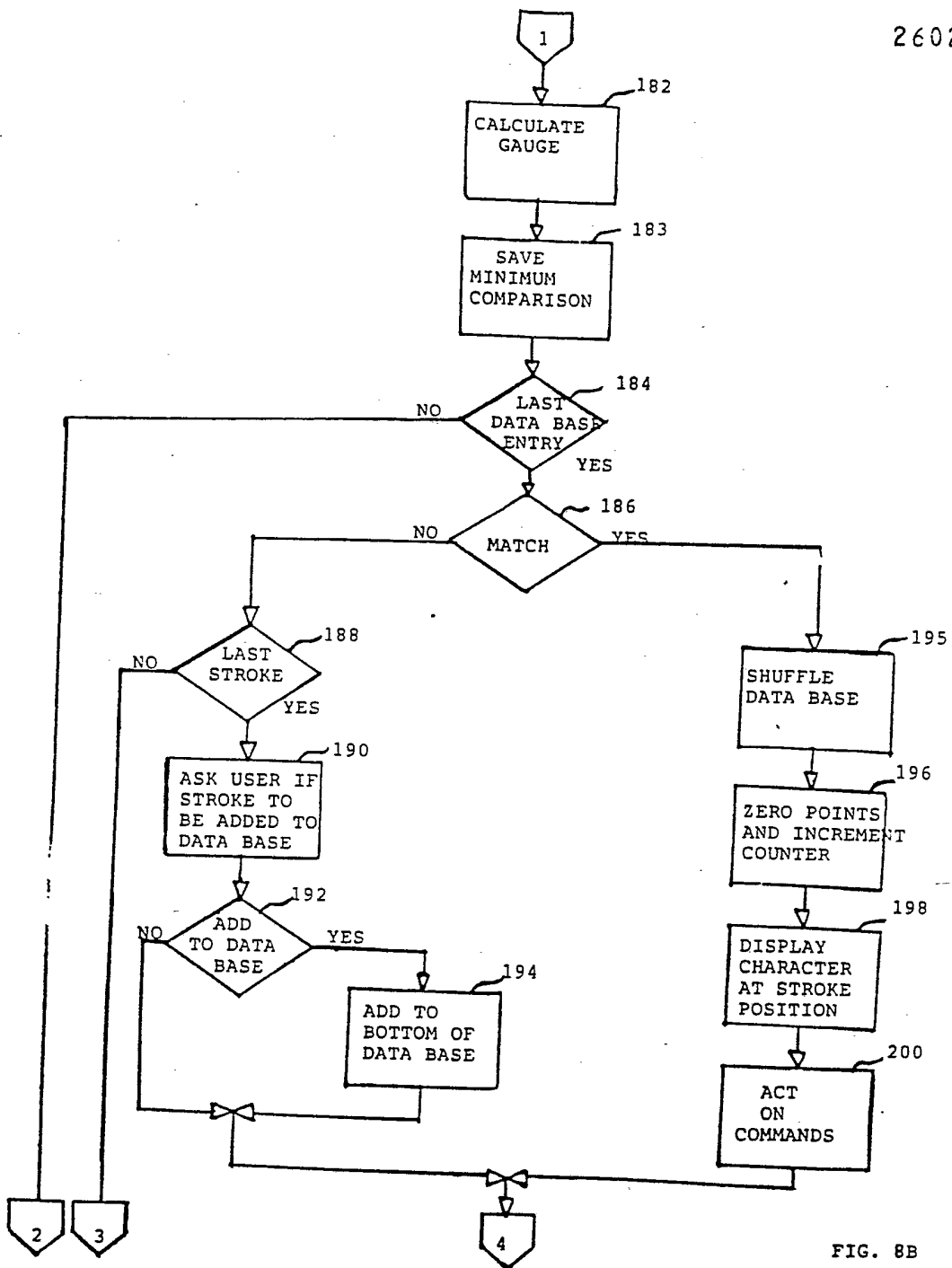


FIG. 8B

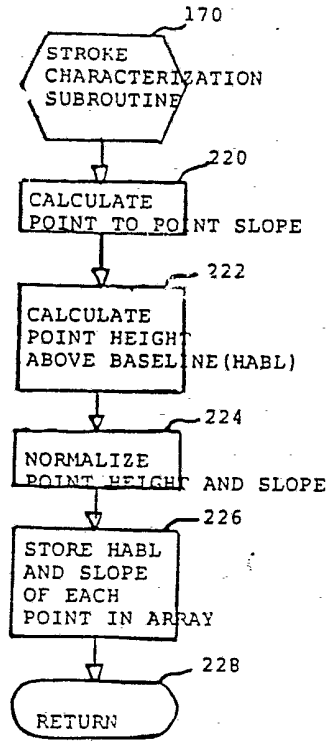


FIG. 9

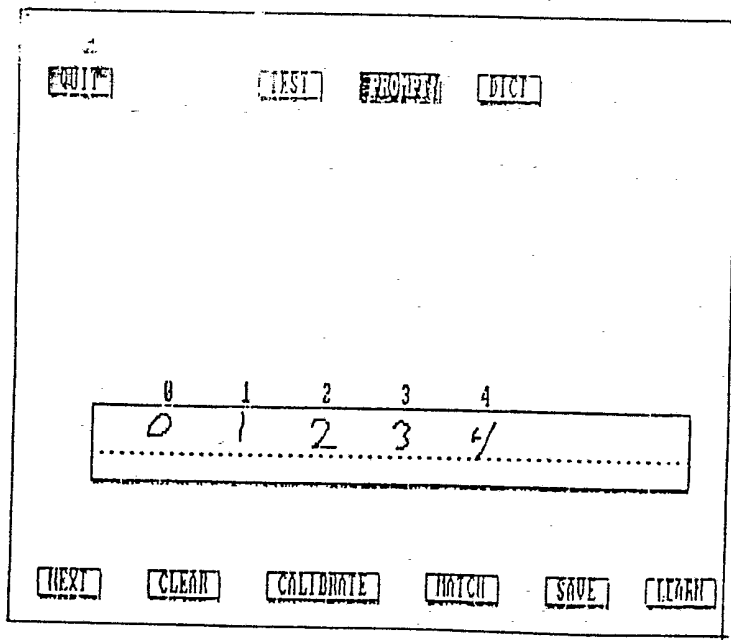
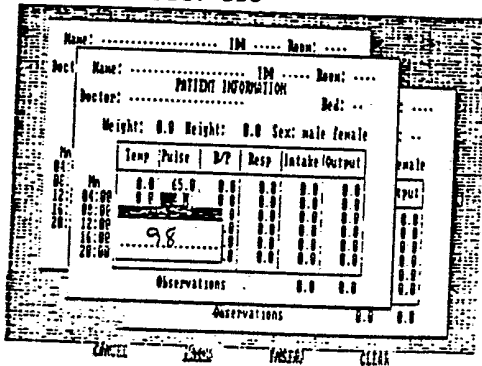
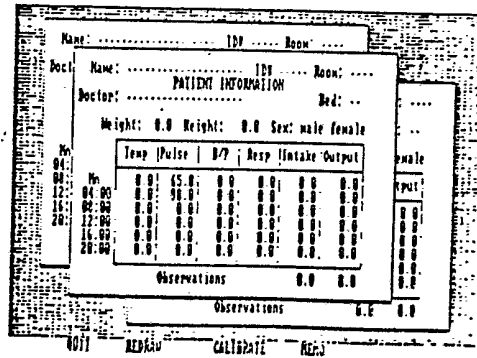
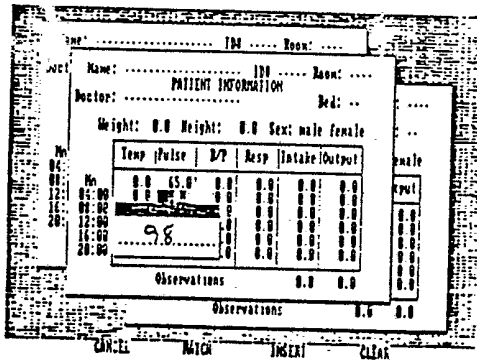
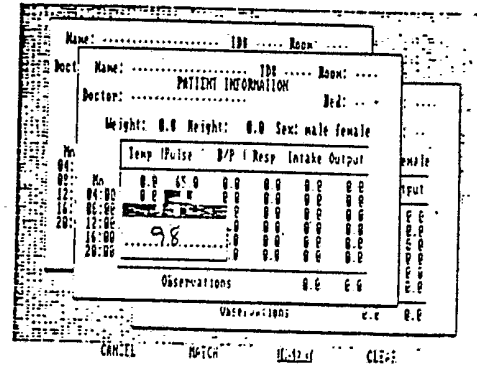
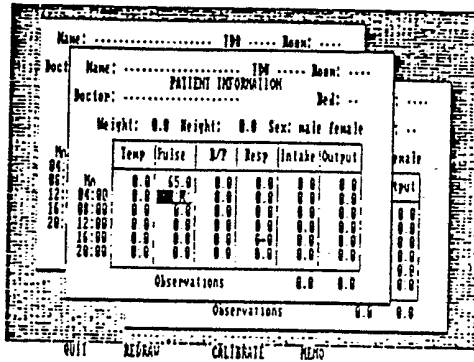
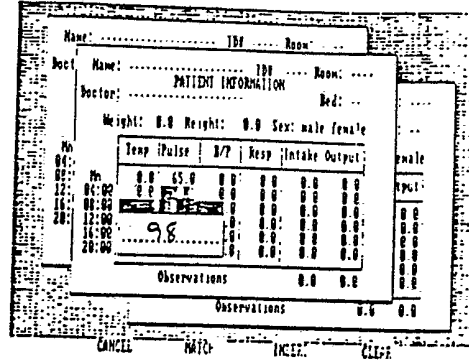
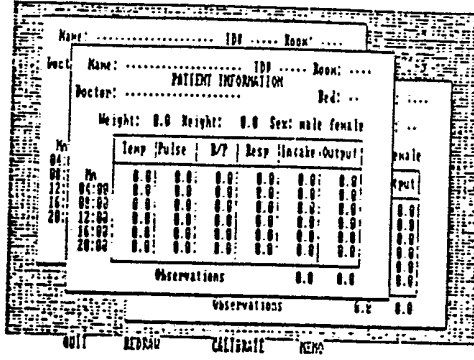


FIG. 10







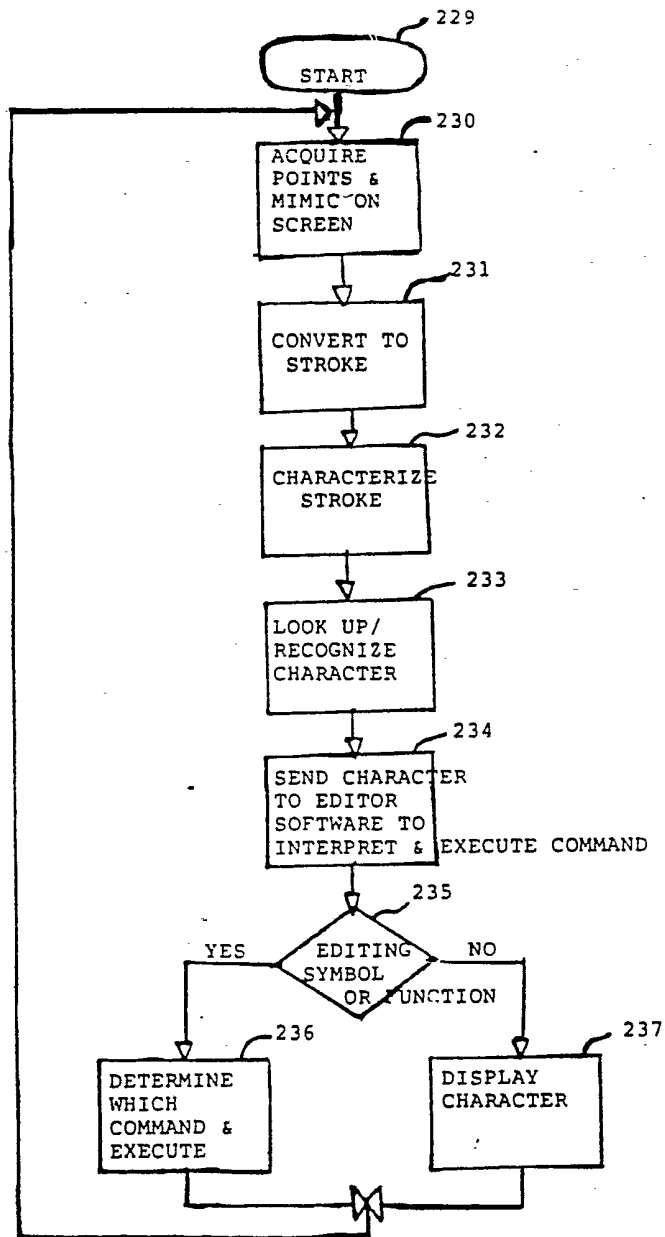


FIG. 13