

[19]中华人民共和国专利局

[51]Int.Cl<sup>6</sup>

G06F 3/033

G06K 11/16



# [12] 发明专利申请公开说明书

[21] 申请号 97190278.X

[43]公开日 1998年6月24日

[11] 公开号 CN 1185844A

[22]申请日 97.3.26

[30]优先权

[32]96.3.28 [33]US[31]08/623,483

[86]国际申请 PCT/US97/05333 97.3.26

[87]国际公布 WO97/36225 英 97.10.2

[85]进入国家阶段日期 97.11.28

[71]申请人 辛纳普蒂克斯有限公司

地址 美国加利福尼亚州

[72]发明人 D·W·吉利斯彼 T·P·阿伦

R·沃尔夫 S·得

[74]专利代理机构 中国专利代理(香港)有限公司

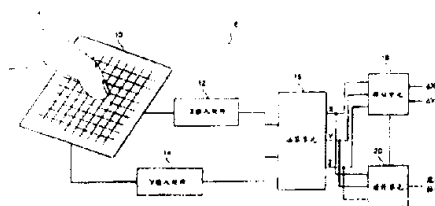
代理人 王 勇 陈景峻

权利要求书 10 页 说明书 64 页 附图页数 28 页

[54]发明名称 具有边缘移动特性和动作识别的目标位置检测器

[57]摘要

本发明公开了用于识别由一个传导性目标在一个触摸检测器垫所作的动作并用于光标移动的方法。在可疑的动作期间通过分析出该传导性目标在检测器垫上的位置，压力和移动可以识别轻击，拖动，推压，延长的拖动和可变的拖动动作，并且传送表示出现这些动作的信号给主计算机。同时也将表示传导性目标的位置的信号以及区分触摸检测器垫的周边部分和内部部分的信号送给主计算机。



## 权利要求书

1. 一种识别在一个触摸检测系统中的触摸检测器垫上所进行的一个延长的拖动动作的方法，其中该触摸检测系统向一个主计算机提供 X 和 Y 位置信息，该方法包括下列步骤：

5 检测一个传导性目标在该触摸检测器垫上的第一次出现的开始和结束之间的第一个间隔；

将所述第一个间隔与第一个时间参照量进行比较；

如果所述第一个间隔小于所述第一个时间参照量，向该主计算机传  
送一个指示出现一个动作的一个动作信号；

10 检测一个传导性目标在该触摸检测器垫上的第二次出现的开始和结束之间的第二个间隔；

将所述第一次出现的所述结束时刻与所述第二次出现的所述开始时刻之间的第一个所经过的时间与第二个时间参照量进行比较；

15 如果所述第一个所经过的时间小于所述第二个时间参照量，在所述第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息；

检测一个传导性目标在该触摸检测器垫上的第三次出现的开始和结束之间的第三个间隔；

20 将所述第二次出现的所述结束时刻与所述第三次出现的所述开始时刻之间的第二个所经过的时间与第三个时间参照量进行比较；

如果所述第二个所经过的时间小于所述第三个时间参照量，在所述第二个所经过的时间保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息。

25 2. 一种识别在一个触摸检测系统中的触摸检测器垫上所进行的一个延长的拖动动作的方法，其中该触摸检测系统向一个主计算机提供 X 和 Y 位置信息，该方法包括下列步骤：

检测一个传导性目标在该触摸检测器垫上的第一次出现的开始和结束之间的第一个间隔；

将所述第一个间隔与第一个时间参照量进行比较；

30 如果所述第一个间隔小于所述第一个时间参照量，向该主计算机传送一个指示出现一个动作的一个动作信号；

检测一个传导性目标在该触摸检测器垫上的第二次出现的开始和



结束之间的第二个间隔；

将所述第一次出现的所述结束时刻与所述第二次出现的所述开始时刻之间的第一个所经过的时间与第二个时间参照量进行比较；

5 在所述传导性目标第二次出现的所述结束时刻检测所述传导性目标的一个平均速度；

将所述传导性目标的所述第二次出现的所述结束时刻的所述传导性目标的所述平均值与第一个参照速度进行比较；

10 如果所述第一个所经过的时间小于所述第二个时间参照量，在所述第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息；

检测一个传导性目标在该触摸检测器垫上的第三次出现的开始和结束之间的第三个间隔；

15 将所述传导性目标的所述第二次出现的所述结束时刻与所述第三次出现的所述开始时刻之间的第二个所经过的时间与第三个时间参照量进行比较；

如果所述第二个所经过的时间小于所述第三个时间参照量，并且如果所述传导性目标在所述第二次出现结束时的所述平均速度大于所述第一个参照速度，在所述第二个所经过的时间和所述第三时间间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息。

20 3. 一种识别在一个触摸检测系统中的触摸检测器垫上所进行的一个延长的拖动动作的方法，其中该触摸检测系统向一个主计算机提供 X 和 Y 位置信息，该方法包括下列步骤：

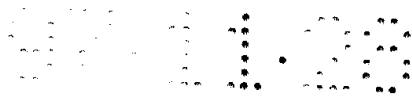
检测一个传导性目标在该触摸检测器垫上的第一次出现的开始和结束之间的第一个间隔；

25 将所述第一个间隔与第一个时间参照量进行比较；

如果所述第一个间隔小于所述第一个时间参照量，向该主计算机传送一个指示出现一个动作的一个动作信号；

检测一个传导性目标在该触摸检测器垫上的第二次出现的开始和结束之间的第二个间隔；

30 将所述传导性目标第一次出现的所述结束时刻与所述第二次出现的所述开始时刻之间的第一个所经过的时间与第二个时间参照量进行比较；



检测所述传导性目标的第二次出现在所述触摸检测器垫上所述结束时的位置；

如果所述第一个所经过的时间小于所述第二个时间参照量，在所述第一个所经过的时间保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息；

检测一个传导性目标在该触摸检测器垫上的第三次出现的开始和结束之间的第三个间隔；

将所述传导性目标第二次出现的所述结束时刻与所述第三次出现的所述开始时刻之间的第二个所经过的时间与第三个时间参照量进行比较；

检测所述传导性目标的所述第三次出现在所述触摸检测器垫上所述开始时的位置；

将所述传导性目标的所述第二次出现时的所述结束时刻的所述位置与所述第三次出现时的所述开始时刻的所述位置之间的一个距离与第一个参照距离进行比较；

如果所述第二个所经过的时间小于所述第三个时间参照量，并且所述传导性目标的所述第二次出现时的所述结束时刻的所述位置与所述第三次出现时的所述开始时刻的所述位置之间的一个距离大于所述第一个参照距离，在所述第二个所经过的时间和所述第三间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息。

4. 一种识别在一个触摸检测系统中的触摸检测器垫上所进行的一个可变的拖动动作的方法，其中该触摸检测系统向一个主计算机提供 X 和 Y 位置信息，该方法包括下列步骤：

检测一个传导性目标在该触摸检测器垫上的第一次出现的开始和结束之间的第一个间隔；

将所述第一个间隔与第一个和第二个时间参照量进行比较，所述第二个时间参照量小于所述第一个时间参照量；

如果所述第一个间隔小于所述第一个时间参照量，向该主计算机传送一个指示出现一个动作的动作信号；

检测一个传导性目标在该触摸检测器垫上的第二次出现的开始和结束之间的第二个间隔；

如果所述第一个间隔小于所述第二个时间参照量，将所述传导性目



标第一次出现的所述结束时刻与所述第二次出现的所述开始时刻之间的一个所经过的时间与第三个时间参照量进行比较；

5 如果相比于所述第三个时间参照量的所述所经过的时间小于所述第三个时间参照量，在所述的第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息；

如果所述第一个间隔大于所述第二个时间参照量并且小于所述第一个时间参照量，将所述所经过的时间与第四个时间参照量比较；

10 如果相比于所述第四个时间参照量的所述所经过的时间小于所述第四个时间参照量，在所述的第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息。

5. 一种识别在一个触摸检测系统中的触摸检测器垫上所进行的一个可变的拖动动作的方法，其中该触摸检测系统向一个主计算机提供 X 和 Y 位置信息，该方法包括下列步骤：

15 检测一个传导性目标在该触摸检测器垫上的第一次出现的开始和结束之间的第一个间隔；

将所述第一个间隔与第一个和第二个时间参照量进行比较，所述第二个时间参照量小于所述第一个时间参照量；

如果所述第一个间隔小于所述第二个时间参照量，向该主计算机传送一个指示出现一个动作的动作信号；

20 如果所述第一个间隔大于所述第二个时间参照量，但是小于所述第一个时间参照量，在一个延迟之后，向该主计算机传送一个指示出现一个动作的动作信号；

检测一个传导性目标在该触摸检测器垫上的第二次出现的开始和结束之间的第二个间隔；

25 如果所述第一个间隔小于所述第二个时间参照量，将所述传导性目标第一次出现的所述结束时刻与所述第二次出现的所述开始时刻之间的一个所经过的时间与第三个时间参照量进行比较；

30 如果相比于所述第三个时间参照量的所述所经过的时间小于所述第三个时间参照量，在所述的第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息；

如果所述第一个间隔大于所述第二个时间参照量并且小于所述第一个时间参照量，将所述所经过的时间与第四个时间参照量比较；



如果相比于所述第四个时间参照量的所述所经过的时间小于所述第四个时间参照量，在所述的第二个间隔保持所述动作信号并且重复地向所述主计算机发送 X 和 Y 位置信息。

5 6. 一种响应表示一个目标在一个二维检测平面上的位置的电信号为在与一个计算机相连的显示器屏幕上提供用于移动光标的一种电信号的方法，包括下列步骤：

提供一个检测平面，该检测平面包括安排成相隔开的多个行导电线和列导电线的导线矩阵，所述检测平面其特征在于在所述行导电线和列导电线的每个上具有一种固有电容，所述电容随着一个目标距离所述行和列导线的接近距离而改变，所述检测平面包括一个内部区域和一个外部区域，所述内部区域由所述外部区域环绕着，所述外部区域从所述检测平面的外边缘向内延伸，并且所述外部区域具有在一个 X 平面的第一部分和在一个 Y 平面的第二部分；

检测在所述行和列导线的至少所选择的一个上的电容；

15 从表示所述目标在所述检测平面上的 X 和 Y 方向的当前位置的所检测的电容产生当前位置信号；

检测所述目标是否在所述检测平面的所述外部区域中；

20 如果所述目标不在所述检测平面的所述外部区域，产生表示所述目标的所述当前位置的 X 和 Y 坐标和所述目标的一个前一位置的 X 和 Y 坐标之间的差的第一个相对位置 X 和 Y 信号，并且发送所述第一相对位置 X 和 Y 信号给所述计算机；

25 如果所述目标在所述检测平面的所述外部区域的所述第二部分，产生表示所述目标的所述当前位置的所述 X 坐标和在所述检测平面上的一个固定位置的 X 坐标之间的差的第二个相对位置 X 信号，并且只要所述目标在所述检测平面的所述外部区域的所述第二部分就发送所述第二相对位置 X 信号给所述计算机；

30 如果所述目标在所述检测平面的所述外部区域的所述第一部分，产生表示所述目标的所述当前位置的所述 Y 坐标和在所述检测平面上的一个固定位置的 Y 坐标之间的差的第二个相对位置 Y 信号，并且只要所述目标在所述检测平面的所述外部区域的所述第一部分就发送所述第二相对位置 Y 信号给所述计算机。

7. 根据权利要求 6 的方法，其特征在于所述检测平面上的所述固



定位置是所述检测平面的几何中心。

8. 根据权利要求 6 的方法, 其特征在于所述当前位置信号, 所述第一相对位置 X 和 Y 信号以及所述第二相对位置 X 和 Y 信号是数字信号。

5 9. 一种响应表示一个目标在一个两维检测平面上的位置的电信号为在与一个计算机相连的显示器屏幕上提供用于移动光标的一种电信号的方法, 包括下列步骤:

提供一个检测平面, 该检测平面包括安排成相隔开的多个行导电线和列导电线的导线矩阵, 所述检测平面其特征在于在所述行导电线和列导电线的每个上具有一种固有电容, 所述电容随着一个目标距离所述行和列导线的接近距离而改变, 所述检测平面包括一个内部区域和一个外部区域, 所述内部区域由所述外部区域环绕着, 所述外部区域从所述检测平面的外边缘向内延伸, 并且所述外部区域具有在一个 X 平面的第一部分和在一个 Y 平面的第二部分;

15 检测在所述行和列导线的至少所选择的一个上的电容;

从表示所述目标在所述检测平面上的 X 和 Y 方向的当前位置的所检测的电容产生当前位置信号;

检测所述目标是否在所述检测平面的外部区域中;

20 如果所述目标不在所述检测平面的所述外部区域, 产生表示所述目标的所述当前位置的 X 和 Y 坐标和所述目标的一个前一位置的 X 和 Y 坐标之间的差的第一个相对位置 X 和 Y 信号, 并且发送所述第一相对位置 X 和 Y 信号给所述计算机;

25 如果所述目标在所述检测平面的所述外部区域的所述第二部分, 产生一个第二相对位置 X 信号, 所述第二相对位置 X 信号包括所述增加了一个数量的所述第一个 X 信号, 该数量和与所述目标的所述 X 方向的所述当前位置与在所述检测平面的一个固定 X 位置之间在所述 X 方向上的差成比例, 并且只要所述目标在所述检测平面的所述外部区域的所述第二部分就发送所述第二相对位置数字 X 信号给所述计算机;

30 如果所述目标在所述检测平面的所述外部区域的所述第一部分, 产生一个第二相对位置 Y 信号, 所述第二相对位置 Y 信号包括所述增加了一个数量的所述第一个 Y 信号, 该数量和与所述目标的所述 Y 方向的所述当前位置与在所述检测平面的一个固定 Y 位置之间在 Y 方向上



的差成比例，并且只要所述目标在所述检测平面的所述外部区域的所述第一部分就发送所述第二相对位置数字 Y 信号给所述计算机。

10. 根据权利要求 9 的方法，其特征在于所述检测平面上的所述固定 X 位置和所述固定 Y 位置定义所述检测平面的几何中心。

5 11. 根据权利要求 9 的方法，其特征在于所述当前位置信号，所述第一相对位置 X 和 Y 信号以及所述第二相对位置 X 和 Y 信号是数字信号。

12. 根据权利要求 9 的方法，其特征在于：

10 和在所述目标的所述 X 方向的所述当前位置与在所述检测平面的一个固定 X 位置之间在所述 X 方向上的差成比例的所述数量是在所述目标的所述 X 方向的所述当前位置与在所述检测平面的一个中心 X 位置之间在 X 方向上的差的 m 倍；

15 和在所述目标的所述 Y 方向的所述当前位置与在所述检测平面的一个固定 Y 位置之间在所述 Y 方向上的差成比例的所述数量是在所述目标的所述 Y 方向的所述当前位置与在所述检测平面的一个中心 Y 位置之间在 Y 方向上的差的 n 倍；

其中所选择的 m 和 n 是对在所述显示器上的所述光标的移动产生一个所需的速度。

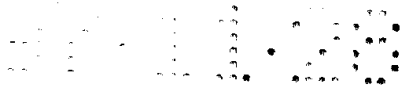
20 13. 根据权利要求 12 的方法，其特征在于：m 与 n 的比值等于所述检测平面的宽度与所述检测平面的高度之间的比值。

14. 一种响应表示一个目标在一个二维检测平面上的位置的电信号为在与一个计算机相连的显示器屏幕上提供用于移动光标的一种电信号的方法，包括下列步骤：

25 提供一个检测平面，该检测平面包括安排成相隔开的多个行导电线和列导电线的导线矩阵，所述检测平面其特征在于在所述行导电线和列导电线的每个上具有一种固有电容，所述电容随着一个目标距离所述行和列导线的接近距离而改变，所述检测平面包括一个内部区域和一个外部区域，所述内部区域由所述外部区域环绕着，所述外部区域从所述检测平面的外边缘向内延伸，并且所述外部区域具有在一个 X 平面的第一  
30 部分和在一个 Y 平面的第二部分；

同时生成第一组信号，该组信号与在没有目标接近所述检测平面时每一个所述行导电线的所述电容值成比例；





同时生成第二组信号，该组信号与在没有目标接近所述检测平面时每一个所述列导电线的所述电容值成比例；

同时生成第三组信号，该组信号与在一个目标接近所述检测平面时每一个所述行导电线的所述电容值成比例；

5 同时生成第四组信号，该组信号与在所述目标接近所述检测平面时每一个所述列导电线的所述电容值成比例；

计算所述第一组信号与所述第三组信号之间的第一个加权平均值，产生在所述检测平面的 X 方向上的一个当前位置信号；

10 计算所述第二组信号与所述第四组信号之间的第二个加权平均值，产生在所述检测平面的 Y 方向上的一个当前位置信号；

如果所述目标不在所述检测平面的所述外部区域，产生表示在 X 和 Y 方向的所述当前位置信号与在 X 和 Y 方向的前一组当前位置信号之间的差的第一个相对位置 X 和 Y 信号，并且发送所述第一相对位置 X 和 Y 信号给所述计算机；

15 如果所述目标在所述检测平面的所述外部区域的所述第二部分，产生一个第二相对位置 X 信号，所述第二相对位置 X 信号表示在所述目标的所述当前位置的所述 X 坐标与在所述检测平面的一个固定位置的 X 坐标之间的差，并且只要所述目标在所述检测平面的所述外部区域的所述第二部分就发送所述第二相对位置 X 信号给所述计算机；

20 如果所述目标在所述检测平面的所述外部区域的所述第一部分，产生一个第二相对位置 Y 信号，所述第二相对位置 Y 信号表示所述目标的所述当前位置的所述 X 坐标与在所述检测平面的一个固定位置的 Y 坐标之间的差，并且只要所述目标在所述检测平面的所述外部区域的所述第一部分就发送所述第二相对位置 Y 信号给所述计算机。

25 15. 根据权利要求 14 的方法，其特征在于所述检测平面上的所述固定位置是所述检测平面的几何中心。

16. 根据权利要求 14 的方法，其特征在于同时产生所述第一，第二，第三和第四组信号的步骤包括下列步骤：

在所述行导电线上施加上一个第一已知电压；

30 以一个固定的电流将所述行导电线放电一个固定的时间；

测量和存储在所述行导电线上的第一组行导电线结果电压；

在所述行导电线上施加上一个第二已知电压；



以所述固定的电流将所述行导电线充电所述固定的时间；  
测量和存储在所述行导电线上的第二组行导电线结果电压；  
将相应所述第一和第二组行导电线结果电压进行平均；  
在所述列导电线上施加上一个第一已知电压；

5 以一个固定的电流将所述列导电线放电一个固定的时间；  
测量和存储在所述列导电线上的第一组列导电线结果电压；  
在所述列导电线上施加上一个第二已知电压；  
以所述固定的电流将所述列导电线充电所述固定的时间；  
测量和存储在所述列导电线上的第二组列导电线结果电压；  
10 将相应所述第一和第二组列导电线结果电压进行平均。

17. 根据权利要求 14 的方法，其特征在于计算所述第一和第二加权平均的步骤包括下列步骤：

计算所述第一组数字信号的一个和与一个加权和；  
计算所述第二组数字信号的一个和与一个加权和；  
15 计算所述第三组数字信号的一个和与一个加权和；  
计算所述第四组数字信号的一个和与一个加权和；

通过从所述第三组数字信号的所述加和中减去所述第一组数字信号的所述加和计算一个行分子；

20 通过从所述第四组数字信号的所述和中减去所述第二组数字信号的所述和计算一个行分母；

用所述行分母去除所述行分子得出表示所述目标在一个行维上的位置的一个行位置信号；

通过从所述第二组数字信号的所述加和中减去所述第二组数字信号的所述加和计算一个列分子；

25 通过从所述第二组数字信号的所述和中减去所述第二组数字信号的所述和计算一个列分母；以及

用所述列分母去除所述列分子得出表示所述目标在一个列维上的位置的一个列位置信号。

18. 根据权利要求 17 的方法，还包括步骤：

30 存储所述第一和第三组信号的所述和以及所述加权和作为所述第二和第四组信号的一个存储的和以及一个存储的加权和；以及

在计算随后的所述行分子和分母以及所述列分子和分母之一中使



用所述存储的和以及所述存储的加权和；

使用所述存储的和以及一个存储的加权和以提供表示所述目标在所述二维平面上的一个随后的位置的一个电信号。

# 说明书

## 具有边缘移动特性和动作识别 的目标位置检测器

### 5 相关申请

这一申请是 1994 年 10 月 7 日提交的序列号为 08/320158 的共同未决申请的一个部分继续申请，后者又是 1994 年 9 月 2 日提交的序列号为 08/300387 的共同未决申请的一个部分继续申请，后者又是 1993 年 8 月 31 日提交的序列号为 08/115743 的一个部分继续申请（现在是  
10 5374787 号美国专利），后者又是 1992 年 6 月 8 日提交的序列号为 07/895934 的共同未决申请的一个部分继续申请。

### 背景技术

### 发明领域

本发明涉及目标位置检测传感器和系统。更具体地，本发明涉及在  
15 诸如计算装置的光标移动的应用中以及其他应用中使用的目标位置识别，尤其涉及具有增强的边缘移动和动作识别特性的光标移动。

### 现有技术

现在已有或者已经提出了用作目标位置检测器的许多装置应用于  
计算机系统和其他应用中。这种装置中的人们最熟悉的一种是计算机  
20 “鼠标器”。尽管鼠标器作为位置指示设备极其普遍，它具有机械部件，并且要求有一个平面，使得位置球能在其上面滚动。而且，为了有合理的分辨率，鼠标器通常需要滚动较长的距离。最后，鼠标器需要用户从键盘上抬起手以使得光标移动，从而干扰了人们的主要目的，因为人们的目的是通常在计算机上键入。

25 轨迹球装置类似于鼠标装置。然而其主要差别是，不同于一个鼠标装置，轨迹球不需要一个在上面滚动的平面。轨迹球装置仍然很贵，它具有移动部件，并且象鼠标装置一样要求一个相当重的接触。在规格是它们也较大，不能较好地适合体积敏感的应用中，比如膝上型计算机。

现在已有几种触摸-传感技术可以用于位置指示器。众所周知，有  
30 一种电阻膜（resistive-membrane）位置传感器，并且用于多个应用中。然而，它们总的来说分辨率较差，传感器的表面暴露给用户，因此易于磨损。另外，相对来说，电阻膜触摸传感器相对较贵。为了可靠操作，



一个单表面方案( one surface approach )要求用户接地到传感器。在便携式计算机中这是不能保证的。单表面方案的一个例子是 MA Wilmington 的 Micro Touch 的产品 UnMouse。一个两表面方案具有更差的分辨率, 并且磨损得更快。

5 授予 Yoshikawa 的美国专利( 专利号为 4680430 ) , 授予 Ellis 的美国专利( 专利号为 3497617 ) 以及其他专利揭示了电阻板技术。所有这些技术的缺点是耗电高, 以及电阻膜的高成本。

表面声波( SAW )装置可以用作位置指示器。然而, 这种传感器技术贵而且对轻的触摸不敏感。另外, SAW 装置对触摸表面的残留累积 ( residue buildup ) 敏感, 而且总的来说具有较差的分辨率。

10 拉力计或压力板方法是一种有意义的位置检测技术, 但具有几个缺点。这种方案可以使用压电传感器。其一个缺点是压电现象是一种 AC 现象, 同时对用户的移动速率敏感。另外, 由于需要特殊的传感器, 拉力计或压力板方案还是有些贵。

15 光学方案也是可能的, 但受到几个原因的限制。它们所有都需要产生光, 这就要求有外部部件, 并且增加成本和电能消耗。比如, 一种“手指-截断”红外矩阵检测器消耗高的电能, 并且具有相对较差的分辨率。

20 人们已经进行了多种尝试, 试图提供一种装置, 以检测用作指示设备的拇指或其他手指的位置, 以代替鼠标器或轨迹球。这种设备的理想特性是低功耗, 小的外观, 高的分辨率, 低成本, 快速响应, 并且在手指带有电噪声时, 或者在触摸表面沾染有污物或水分时能够可靠地工作。

25 由于电阻式装置的这些缺点, 人们已经基于电容式检测手指的位置多次尝试提供指示的能力。授予 Volpe 的美国专利 ( 专利号为 3921166 ) 教导了一种电容矩阵, 其中手指改变行和列电极的转移电容 ( transcapacitance )。授予 Bobick 的美国专利( 专利号为 4103252 ) 利用四个振荡信号在四个电容电极之间内插 x 和 y 位置。授予 Schuyler 的美国专利( 专利号为 4455452 ) 公开了一种电容板, 其中手指衰减耦合在电极之间的电容。

30 授予 Mabusth 的美国专利 ( 专利号为 4550221 ) 教导了一种电容板, 其中一个振荡信号测定对“虚地”的有效电容。连续地查询每一行或

列，并且应用插值的基本形式求出两行或列之间的位置。通过对振荡波形的多个周期进行平均试着解决电干扰的问题。通过在没有手指的时候进行检测，并且在没有手指出现期间应用一个周期校正来解决污染的问题。授予 Rympalski 的美国专利( 专利号为 4639720 ) 教导了用于检测  
5 一指点笔的位置的板。该指点笔改变耦合在行和列电极之间的转移电容，这些行和列电极是连续被扫描的。授予 Matzke 的美国专利( 专利号为 4736191 ) 教导了在键盘的空格键下的一种径向电极装置，它可以通过用拇指触摸而激活。该专利教导使用整个触摸电容作为触摸压力的一种指示，以控制光标移动的速度。利用脉冲化的连续查询解决电干扰  
10 的问题。

授予 Greanias 的美国专利( 专利号分别为 4686332 和 5149919 ) 教导了用于安装到一个 CRT 上的指点笔和手指检测系统。作为一个手指检测系统，它使用 X/Y 传感器矩阵确定传输最大值信号的两条矩阵线。利用这种编码方案，这两条线唯一地确定相对于线间隔的分辨率的手指  
15 位置的定位。对于指点笔检测系统，Greanias 首先粗略地定位它，然后通过驱动在一个方向上的目标的一侧上的所有线以及在相反方向的相对侧上的所有线产生一个虚偶极( virtual dipole )。使用不同的偶极相位和信号极性进行三次。假设一个预定的矩阵响应该目标，这三次测量提供一组联立方程，可对它求解出位置。

20 授予 Evans 的美国专利( 专利号为 4733222 ) 第一次教导了插值到高阶的电容触摸测量系统。Evans 教导了一个三极限测量系统( three terminal measurement system )，它在其矩阵中使用一个驱动，传感和电极信号集，并且基于在电极结点信号上的手指的衰减效应的测量( 使用电容分压 器现象 )。Evans 连续地扫描每一驱动集以测量电容。从  
25 三个最大的响应开始，应用一个插值例程来确定手指位置。Evans 还教导了一种归零技术，它允许"无手指"级可作为测量的部分而被取消。

授予 Gruaz 的美国专利( 专利号为 5016008 ) 教导了一种也使用插值的触摸敏感板( touch sensitive pad )。Gruaz 在触摸矩阵使用一个驱动和检测信号集，并且象 Evans 一样，取决于手指的衰减效应以调制  
30 该驱动信号。连续地扫描该触摸矩阵以读出每一矩阵行的响应信号。然后一个插值程序在两维中选择两个最大的相邻信号以确定手指的位置，并且从这 4 个数中比例度量地确定有效的位置。

Gerpheide 的 PCT 申请（申请号为 US90/04584，公开号为 WO91/03039，美国专利号为 5305017）将 Greanias 的虚偶极方法的一种变形应用到一个触摸板系统。Gerpheide 教导了将给定相位和频率的振荡电位施加到虚偶极的一侧上的所有电极，以及将一种同样频率和相反相位的振荡电位施加到其另一侧。电子电路产生一个“平衡信号”，在没有手指时它是零，如果一个手指在虚偶极的中心的一侧则其有一个极性，如果该手指在其另一侧，则具有相反的极性。为了开始获得手指的位置，连续地扫描该板上的虚偶极。一旦定位了手指，在手指已经移动超过一行或一列时将通过向手指移动虚偶极而对其进行跟踪。

10 由于虚偶极方法是通过产生在电容不随距离而变化时为零的一个平衡信号而操作的，因此它只检测手指接触区域的周边，而不是整个接触区域。由于该方法取决于该激励信号的同步检测，它必需在较长的时间周期上进行平均以排除电干扰，因此较慢。这一方法和一旦丢失了前一接触而连续地搜索一个新手指接触所需要的平均时间，将象以前的方法一样，不符合快速指示设备的要求，这种快速指示设备不受电干扰的影响。

应当指出的是，所有先前使用插值的触摸板在它们的检测板上加上了苛刻的设计要求。为了产生信号，Greanias 和 Evans 使用了一个复杂和昂贵的驱动，检测和电极行策略。Gruaz 和 Gerpheide 使用一个两信号驱动和检测集。在本发明中，在同一行中进行驱动和检测。这要求行和列的选择是对称和等价的。这又要求所有信号路径的独立校正（这使得板布局更简单和较少受到约束），以及允许更一致的传感器布局。

25 在先有技术中描述的发明和技术的缺点可以追踪到只使用一组驱动和检测电子线路，这种装置在板上的电极上连续地被进行多路复用。这种装置在分立元件时代能节约成本，并避免了在电路之间的偏移和量度差别。

以前系统的连续扫描方法也使得它们对噪音更敏感。噪音级别可在连续的测量之间改变，从而改变所测量的信号和在插值例程中使用的假设。

30 最后，所有先前的方法假设了一个手指位置相对矩阵位置的特定信号响应。由于转移曲线对多个参数是非常敏感的，并不是象 Greanias 和 Gerpheide 假设的一条平滑的线性曲线。

在先前的共同未决申请(序列号为 08/115743, 1993 年 8 月 31 日提交, 现在其美国专利号是 5734787) 中揭示了一种两维电容检测系统, 该系统配有用于电容板的每一行和每一列的一组单独的驱动/检测电子电路。所有行电极同时检测, 所有列电极同时检测。所检测的信号由模拟电路处理。

在当前可使用的触摸板设备中, 只有 Alps/Cirque GlidePoint 包括动作识别。该 GlidePoint 支持基本轻击, 双轻击和拖动动作, 以模拟一个基本的鼠标按键的动作。它不支持多手指动作, 也没有动作模拟辅助的按键撤按。现在对于在 Glide Point 中所用的实现方法还不得而知。然而已经知道 GlidePoint 对于双轻击有一定的困难, 其中一个问题已由本发明解决。GlidePoint 对在轻击动作期间可能是试图平衡光标的每一个手指移动动作表现了一些迟疑。为了允许在长距离上拖动, GlidePoint 也依赖于物理开关或极其高的增益或加速度。

有一种触摸垫(touchpad) 产品 UnMouse, 在其电阻检测器的下面安装了一个开关, 从而用户只要按压在板上就可激活该按钮。这种设备除了要求极其小心和复杂的机械安装之外, 而且用户对之十分厌倦。

在本技术领域, 除了利用手指之外, 利用按压敏感的指点笔进行操作的图形板也是众所周知的。这些设备利用类似本发明的“推压”动作的机构来模拟致动器开关。在这种指点笔操作的板中还没有看到这里所描述的其它类型的动作。

因此, 本发明的一个目标是提供一种两维电容检测系统, 该系统配有用于电容板的每一行和每一列的一组单独的驱动/检测电子电路。所有行电极同时检测, 所有列电极同时检测。

本发明的另一目标是提供一个具有电容板的一个电子系统, 该电子系统对一个手指或其它传导目标的接触的整个区域敏感, 以及提供这一接触区域的中心的一些测量的坐标作为输出, 同时保持对要检测的目标的特征外观不敏感。

本发明的另一个目标是提供一个具有一个电容板的电子系统, 该电子系统将一手指或其它传导目标的接触区域的一些测量结果输出。

本发明的另一目标是提供一个具有用于电容板的每一行和每一列的一组单独的驱动/检测电子电路的两维电容检测系统, 其中所有行电极同时检测, 所有列电极同时检测, 并且其中确定手指或其它传导目标的



位置的信息以数字形式被处理。

5 本发明的另一目标是提供一个二维电容检测系统，其中所有行电极同时检测，所有列电极同时检测，并且其中在一个检测平面的周边区域中的手指或其它传导目标的定位可以有选择地使得光标在一个显示器屏幕上光标“边缘移动”( edge motion )，使得从一个小的检测平面用一个轻微的手指动作将可控制大的光标移动。

本发明的再一目标是对手指或者其它目标在一个触摸检测器垫上所作的拖动延长动作进行识别，其方式是允许以一个动作对一个小检测平面的大的光标偏移进行控制。

10 本发明的再一目标是对手指或者其它目标在一个触摸检测器垫上所作的动作进行识别，其方式是在该动作进行期间允许对手指或其它目标的无意识的动作进行补偿。

本发明的再一目标是提供对多个手指动作的识别以及模拟辅助按钮的撤按( click )。

15 本发明的再一目标是提供对新手和熟练用户的动作之间的差别进行识别。

### 本发明的概要描述

20 随着更高集成度的集成电路的出现，已有可能将许多通道的驱动/检测电子电路以及操作这些电子电路的控制逻辑集成到一个集成电路中，并且接口电路允许指点设备与一个主微处理器直接进行通讯。本发明使用适应性的模拟技术来克服通道之间的偏移和度量差别，从而并行地检测所有板的行和列的转移电容或者固有电容( self - capacitance )。这种并行检测能力是通过每行或每列提供一组电子电路而提供的，使得检测周期极其短，从而在仍然保持不受强烈的电干扰的同时实现快速的响应。

25 本发明包括一种位置-检测技术，该技术对于在计算机“鼠标器”或轨迹球环境中需要手指位置信息的应用中尤其有用。然而，本发明的位置检测技术具有比计算机鼠标器更广泛的应用，因为如果触摸了一个或多个点其传感器能够检测和报告。另外，该检测器能够检测触摸的压力。

30 根据本发明的一个较佳实施例，这里称作“手指指示器”实施例，一个位置检测系统包括一个位置检测传感器，该传感器包括一个位于一个

基板( 比如一个印刷电路板 ) 之上的触摸- 敏感表面, 该基板又包括一个传导线矩阵。第一组传导线沿第一个方向走线, 并且与第二组传导线相绝缘, 第二组传导线沿着基本上与第一个方向垂直的第二方向走线。将一个绝缘层布置在第一和第二组传导线上。该绝缘层足够薄, 能  
5 明显提高耦合在位于其表面的一个手指与第一和第二组传导线之间的电容。

检测的电子电路响应手指, 传导目标或者高介电常数( 即大约大于 5 ) 的一个目标的接近, 将目标接近所引起的导体电容的改变翻译成数字形式, 处理该数字信息以导出位置和触摸压力信息。其输出是在其表面  
10 的一个目标的简单的 X, Y 和压力值。在这里所有的描述中, 手指可被认为与传导目标和高介电常数目标互换。

不同的先有技术的板扫描技术在不同的环境中具有不同的优点。根据本发明的并行驱动/检测技术允许同时进行输入采样, 因此所有通道受到一个干扰电信号的同一相位的影响, 极大地简化了信号处理和噪声滤  
15 波。

在本发明的触摸检测技术中使用两种驱动/检测方法。根据本发明的第一和当前较佳实施例, 传感器矩阵的所有 X 线上的电压同时移动, 同时 Y 线上的电压保持在一个恒定电压上, 所取样的点的完整集合给出 X 维上的手指的轮廓。接着, 传感器矩阵的所有 Y 线上的电压同时移动,  
20 同时 X 线上的电压保持在一个恒定电压上, 所取样的点的完整集合给出另一维上的手指的轮廓。

根据第二种驱动/检测方法, 传感器矩阵的所有 X 线上的电压同时向正方向移动, 同时 Y 线上的电压向负方向移动。接着, 传感器矩阵的所有 X 线上的电压同时向负方向移动, 同时 Y 线上的电压向正方向移  
25 动。这一技术加强了两维之间的任何转移电容的效应, 或者相反地, 减少了与地的任何寄生电容的效应。在这两种方法中, 从检测过程来的电容信息提供了手指接近每一维的传感器的一个轮廓。

当前, 两个较佳实施例提取这些轮廓, 导出一个表示 X 和 Y 位置的质心( centroid ) 的数字值, 并且导出用于 Z 压力信息的第二数字值。  
30 主计算机可以直接使用该数字信息。根据本发明, 也可以使用电容信息的模拟处理。

这些实施例的位置传感器只能报告在其传感器表面的一个目标的

位置。如果存在多于一个目标，这一实施例的位置传感器计算所组合的目标集的质心位置。然而，不同于先有技术，由于要描述的是整个垫，可以利用足够的信息以分辨简单的多手指动作，以提供更强大的用户接口。

5 根据本发明的另一方面，数种节电技术已集成进该系统，它们可在测量之间关闭电路。由于根据本发明的并行测量技术比先有技术快得很多，这是可能的。

根据本发明的另一方面，减少噪声的一种变形已集成进该系统。

10 根据本发明的另一方面，使用了一种更易于校正和实现的电容测量技术。

根据本发明的两个方面，当在一个检测平面的所定义的周边区域检测手指或其它传导目标的存在时，可以改变光标移动的控制以提供"边缘移动"，以在一个小的检测平面上从一个手指的动作中控制在一个显示器屏幕上的大的光标移动。

15 根据本发明的另一方面，该主计算机识别一个拖动延长动作，并允许对显示屏上由在一个小的检测平面上进行的一个动作所引起的大的光标偏移进行控制。

20 根据本发明的再一目标，对手指或其它目标在该触摸检测器垫上所作的多个动作进行识别，并传送给一个主计算机。也对某些动作是由新手还是熟练用户产生的进行识别。对在进行动作期间手指或其它目标的无意识移动进行补偿。

### 附图的简要描述

图 1 是本发明的电容位置检测系统的一个总体框图。

25 图 2a 是根据本发明的当前较佳实施例的一个目标位置检测传感器的顶视图，它示出目标位置传感器表面层，该层包括一个顶部导电线层和连接到一个底部导电线层的导电垫。

图 2b 是图 2a 的目标位置检测传感器的底视图，示出了底部导电线层。

30 图 2c 是图 2a 的目标位置检测传感器的复合视图，示出了顶部和底部导电线层。

图 2d 是图 2a-2c 的目标位置检测传感器的剖视图。

图 3 是传感器译码电子电路的框图，它可用于根据本发明的较佳实

施例的检测器传感器中。

图 4a 是可用于本发明的一个充电集成电路的一个简化的概图。

图 4b 是图 4a 的充电集成电路的示意性概图。

图 5 是图 4a 和 4b 的充电集成电路的定时图。

5 图 6 是用于本发明的一个示意性滤波器和取样/保持电路的概图。

图 7 是用于本发明的当前最佳的 A/D 转换器的装置的更详细的框图。

图 8 是可用于本发明的示意性运算单元的框图。

图 9 是用于图 8 的运算单元的一个校正单元的框图。

10 图 10 是在本发明中使用的偏压产生电路的概图。

图 11 是显示本发明的目标位置检测器的边缘移动特性的检测平面的图。

图 12A 是第一个硬件实现的示意图, 该实现用于判断手指或者其它目标是否在检测平面的周边区域。

15 图 12B 是第一个硬件实现的示意图, 该实现用于判断手指或者其它目标是否在检测平面的周边区域。

图 13 是显示本发明的边缘移动特性的硬件实现的示意图。

图 14 是图 1 的动作单元 20 的更详细些的框图。

图 15a 至 15g 是根据本发明可以被识别的一些动作的定时图。

20 图 16A 和 16B 是根据本发明可以用于触摸垫上的两个轻击区域形状的图。

图 17A 至 17F 是展示图 14 的轻击单元的操作的流程图。

图 18A 至 18C 是展示图 14 的之字单元的操作的流程图。

图 19 是根据本发明的一个“推压”动作的定时图。

25 图 20 是图 14 的推压单元的操作流程图。

图 21 是根据本发明可以用于动作识别中的抬跳抑制器电路 ( LiftJump suppressor circuit ) 的框图。

### 较佳实施例的详细描述

30 本申请是 1994 年 9 月 2 日提交的申请号为 08/300387 的共同未决申请的一个部分继续申请, 后者是 1993 年 8 月 31 提交的申请号为 08/115743 ( 现在是美国专利 5374787 ) 的共同未决申请的一个部分继续申请, 后者是 1992 年 6 月 8 日提交的序列号为 07/895934 的共同未决

申请的部分继续申请。本发明继续了在父申请中揭示的方法，并且提供了以前未有的更一致的特性。这些改进提供了更容易集成的解决方案，增加了灵敏度，更好地排除噪声，增加了数据获取的速度，和减少了电力消耗。本发明允许连续的自校正以去掉环境改变的影响，并且增强了  
5 检测平面上的边缘移动的光标控制。

那些本领域的技术人员将会认识到本发明的下面的描述是示例性的而不是限制性，这些技术人员很容易就可想到本发明的其它实施例。

本发明融合了多个一致特性，它允许用于一些新的应用中，而这在以前是不可能的。由于本发明的目标位置检测器具有非常低的功率消耗，它尤其适于用于电池驱动或者低功率的应用中，诸如膝上型或者便携式计算机。它也是一个成本非常低的方案，没有活动部件（并且因此实际上是不需维护的），并且对于检测器使用现有的印刷电路板线迹。本发明的检测技术可以集成进计算机的母板中以进一步降低在计算机应用中的成本。类似地，在其它应用中，检测器可以是一个现有电路板  
10 的一个部分。

由于其规格和外观小，本发明的检测器技术在一个重点考虑体积的膝上型或者便携式应用中尤其有用。本发明的检测器技术对于电路板空间只需要一个检测器接口芯片（该芯片可以与一个微处理器直接接口）以及在印刷电路板上用于检测的面积。

参见图 1，展示的是本发明的电容检测系统 6 的一个简化的框图。电容性位置检测系统 6 可以精确地确定邻近或者触摸一个检测平面 10 的手指 8 或其它传导目标的位置。由 X 输入处理电路 12 检测沿第一方向（比如“X”）走线的多个导电线的电容，由 Y 输入处理电路 14 检测沿第二方向（比如“Y”）走线的多个导电线的电容。所检测到的电容值  
25 在 X 输入处理电路 12 和 Y 输入处理电路 14 进行数字化。将 X 输入处理电路 12 和 Y 输入处理电路 14 的输出送给运算单元 16，后者使用该数字信息导出表示相对于该检测平面 10 的手指 8 或其它传导目标的位置和压力的数字信息。

运算单元 16 的 X，Y 和 Z 输出送到移动单元 18，后者向主计算机提供光标移动方向信号。正如这里所使用的，本领域的普通技术人员将会认识到，所谓的“主计算机”可以是一个单独的计算机，比如 IBM 或兼容 PC，或者是 APPLE 公司的计算机，手持控制单元，个人数字助  
30

手，远程电信设备，或类似其它设备，也可以是使用一个触摸板的输出作为其输入的任何其它设备或系统。

运算单元 16 的 X，Y 和 Z 输出也被传送到动作单元 20，后者用于识别用户在检测平面 10 上所作的手指动作。另外，基于动作处理的状态动作单元 20 可以向移动单元 18 产生一个信号以允许本发明的边缘移动特性。

检测器的材料可以是任意的，只要允许建立板的导电性的 X/Y 矩阵即可。这不仅包括标准的 PC 板，也包括但不限于灵活的 PC 板，导电性弹性材料，丝-屏的导电性线（**silk-screened conductive line**）和压电 Kynar 塑料材料。这使得其在任何便携式设备的应用中以及在人机接口中 10 有用，其中需要将检测器铸模成适合人手的需要。

能够使该检测器符合任何三维表面。可以在任何制造检测器的表面轮廓中以两层方式镀敷铜。这将使得该检测器适应于任何特定应用中 15 所需要的最好的人机工程形式。它与“轻触摸”特性的结合使得它可在许多应用中不费力地使用。该检测器也可以间接形式使用，即，它可以在触摸检测平面具有由导电层覆盖的绝缘泡沫塑料，并且被用于检测挤压其表面的任何目标（不只是传导目标）。

采用小的检测器是合乎实际的，比如，当前所提出的实施例所占面积大约 1.5"x1.5"，然而，本领域的一般技术人员将会认识到对于不同的应用 20 该区域是可改变的。通过改变矩阵线的间距或者改变线的数目可改变该矩阵面积。在需要更多信息的场合，采用大的检测器也是可行的。

除了简单的 X 和 Y 位置信息外，本发明的检测器技术也提供手指压力信息。程序可以使用这一附加维的信息以控制诸如绘图程序中的“宽刷”（**brush width**）模式，特殊菜单访问等的特殊特性，允许向 25 计算机提供一个更自然的检测器输入。业已发现它在实现“鼠标揞按和拖动”模式和简单的输入动作中也是有用的。

用户甚至不需触摸该平面以产生最小反应。这一特性能极大地减少对用户的限制，使得使用更灵活。

本发明的检测系统取决于这样一种传感器设备，对于接触该传感器的目标，该传感器设备能够提供位置和压力信息。现在分别参见图 30 2a-2d 的顶部，底部，复合和剖视图，它们示出了当前的较佳检测平面 10，该检测平面 10 包括用于本发明的一个触摸检测器阵列。由于本发

明的较佳实施例利用了电容特性，将触摸检测器阵列 22 的表面设计成使耦合到手指或者其他导电目标的电容最大。

根据本发明的当前较佳的触摸检测器阵列 22 包括一个基板 24，该基板包括布置于其顶部表面 28 上的一组第一导电线 26，该第一组导电线沿第一方向走线以包括该阵列的行位置。第二组导电线 30 布置于其底部表面 32，该第二组导电线最好沿与第一方向正交的第二方向走线以形成该阵列的列位置。该顶部和底部的导电线 26 和 30 也可交替地与检测垫 ( sense pad ) 34 周期性地接触，如图 2a-2c 的菱形所示，该检测垫 34 包括放大的区域。尽管在本发明中检测垫示出为菱形，如图 2a-2c 所示，但对于本发明的目的，允许它们紧密地包在一起的诸如圆形的其他形状是等价的。如在这里任意规定的，第一导电线 26 将被称为沿“ X ”或“ 行 ”方向取向的，有时也称作“ X 线”，第二导电线 30 将被称为沿“ Y ”或“ 列 ”方向取向的，有时也称作“ Y 线”。

这些检测垫 34 的数目和间隔取决于所要的分辨率。比如，在根据本发明的原则构造的实际实施例中，导电垫使用 0.10 英寸的中心-中心菱形图案，该图案沿 15 行和 15 列的导电线的矩阵布置。在垫图案的每一方向的每隔一个检测垫 34 分别连接到基板 24 的顶部和底部表面 28 和 32 的导电线。

基板 24 可以是一个印刷电路板，灵活的电路板或者任意数目的电路互连的结构。只要可从底部导电线 30 连通到顶部表面 28 上的它们的检测垫 34，其厚度是不重要的。该包括基板 24 的印刷电路板可以使用标准的工业技术构造。电路板的厚度是不重要的。可以利用在印刷电路板技术中众所周知的标准的镀穿孔技术 ( plated-through hole techniques ) 进行从导电垫 34 到底部线 30 的连接。

在本发明的一个替换的实施例中，基板材料 24 可以具有 0.005 至 0.010 英寸量级的厚度。可以省略在上表面 28 的菱形和连接到下表面线 30 的镀穿孔，以进一步减少系统的费用。

在顶部表面 28 的检测垫 34 上放置一个绝缘层 36，以绝缘手指和其他目标。绝缘层 36 最好是一薄层 ( 即大约 5mils )，以使得电容耦合大，并且可以包括一种诸如聚酯薄膜的材料，这种选择是针对屏蔽和人机工程特点。这里所用的术语“显著的电容耦合”意味着具有大于 0.5pF 左右强度的电容耦合。

在手指接近该触摸检测器阵列 22 时，发生两种不同的电容效应。第一种电容效应是转移电容，或检测垫之间的耦合，第二种电容效应是固有电容，或耦合到虚地的。检测电路连接到本发明的检测器阵列 22，响应这些电容之一或两者的改变。这是重要的，因为根据用户的环境，两种电容的相对大小变化很大。本发明的这种检测固有电容和转移电容两者的变化的能力使得我们得到具有广泛应用的通用系统。

根据本发明的较佳实施例，位置检测器系统包括触摸检测器阵列 22 和相关的位置检测电路，该位置检测器系统通过手指接近传感器阵列 22 的电容效应检测在印刷电路板线的矩阵上的手指位置。位置检测器系统将以比行和列线 26 和 30 之间的间隔更细的分辨率报告位于检测器阵列 22 附近的手指的 X，Y 位置。根据本发明的较佳实施例的位置检测器将报告一个正比于该手指的轮廓的 Z 值，因此可指示出手指接触到检测阵列 22 上的绝缘层表面 36 的压力。

根据本发明的当前较佳实施例，使用适应性的模拟和数字 VLSI 技术提供了一种非常敏感的轻-接触的检测器电路。本发明的电路非常健壮，并且校正过程和系统性错误。本发明的检测器电路将处理该电容输入信息并且提供数字信息，该信息可直接提供给一个微处理器。

根据本发明的这一较佳实施例，在一个检测器处理器集成电路芯片中包括检测电路。该检测器处理器芯片可以具有多个 X 和 Y “矩阵”输入。X 和 Y 输入的数目并不必须相等。该集成电路具有一个数字总线，用于输出。在图 2a-2d 揭示的示意性例子中，检测器阵列在 X 和 Y 两个方向上具有 15 条线。该检测器处理器芯片因此具有 15 个 X 输入和 15 个 Y 输入。根据本发明的原则构造的一个实际的实施例中在 X 方向上利用了 18 条线，在 Y 方向上利用了 24 条线。本领域的普通技术人员将会认识到在本发明中可以利用的检测矩阵的大小是随意的，并且主要由设计选择确定。

并行驱动和检测 X 和 Y 矩阵结点，每一条线的电容信息指示手指离结点有多近。所扫描的信息提供在每一维上的手指接近的轮廓。根据本发明的这一方面，在 X 和 Y 方向上推导出轮廓质心也就是在该维中的位置。也集成接近的轮廓曲线以提供 Z 信息。

在本发明的触摸检测技术中使用两种驱动和检测方法。根据本发明的第一和当前较佳实施例，传感器矩阵的所有 X 线上的电压同时移动，



同时 Y 线上的电压保持在一个恒定电压上。接着，传感器矩阵的所有 Y 线上的电压同时移动，同时 X 线上的电压保持在一个恒定电压上。这一扫描方法强调了对由手指提供的对虚地的电容的测量。本领域的普通技术人员将会认识到这两个步骤的顺序是任意的，可以颠倒过来。

5 根据第二种驱动/检测方法，传感器矩阵的所有 X 线上的电压同时向正方向移动，同时 Y 线上的电压向负方向移动。接着，传感器矩阵的所有 X 线上的电压同时向负方向移动，同时 Y 线上的电压向正方向移动。这一第二驱动/检测方法强调了转移电容而不强调虚地电容。如第一种驱动/检测方法，本领域的普通技术人员将会认识到这两个步骤的顺序是任意的，可以颠倒过来。

10 现在参见图 3，提供了根据本发明的所使用的当前较佳检测电路 40 的框图。这一框图以及所附带的说明只与一维（X）中的检测电路有关，并且包括图 1 的 X 输入处理电路 12。本领域的普通技术人员将会认识到可使用一个同样的电路用于检测相反的维（Y），并且包括图 1 15 的 Y 输入处理电路 14。这样的技术人员还会注意到不需要两维互相正交。比如，它们可以是辐射状的，或者与该触摸检测器阵列和系统的其他所需的轮廓相一致的其他特征。本领域的普通技术人员将会认识到这里所揭示的技术也可应用于二维的情况，其中只使用一组导电线。

20 由等价的电容器 42 - 1 至 42 - n 表示在每一检测器矩阵结点的电容。电容器 42 - 1 至 42 - n 的电容包括矩阵导线的电容，并且在没有目标（比如手指）接近检测器矩阵的检测平面时具有一个特征背景值。当一个目标接近该检测平面时，电容器 42 - 1 至 42 - n 的电容正比于目标的大小和接近程度而增加。

25 根据本发明，使用充电积分器电路 44 - 1 至 44 - n 同时测量在每一检测器矩阵的电容。充电积分器电路 44 - 1 至 44 - n 分别起到将电荷注入电容 42 - 1 至 42 - n 的作用，并产生一个正比于在对应的 X 矩阵线上检测的电容的输出值。因此充电积分器电路 44 - 1 至 44n 示出为双向放大器符号。通过偏电压发生电路 46 向每一充电积分器电路 44 - 1 至 44 - n 提供一个工作偏电压。

30 这里所使用的短语“正比于电容”意味着所产生的电压信号是所检测的电容的单调函数。在这里所描述的实施例，电压直接并线性地正比于所检测的电容。本领域的普通技术人员将会认识到其他单调函数（包

括但不限于反比例 ) 以及非线性比例函数( 比如对数或指数函数 ) 也可用于本发明中而不偏离这里所揭示的原则。另外可使用电流检测以及电压检测技术。

5 根据在本发明中使用的当前较佳驱动/检测的方法,在一维上同时对所有的输入进行电容测量以克服在现有技术中扫描单个输入的方法中所固有的问题。现有技术的方法所具有的问题是它对高频和大幅度的噪声( 大的  $dv/dt$  噪声 ) 敏感,这种噪声是通过触摸目标耦合到电路上的。这种噪声可能扭曲手指的轮廓,这是因为噪声程度的改变,噪声出现在  
10 一个较迟的扫描周期而不是在一个较早的扫描周期。

本发明通过同时沿 X 方向然后沿 Y 方向对所有输入“ 进行一个快照” ( 或反过来 ) 克服了这一问题。由于注入的噪声与在所有输入上的手指信号强度成正比,因此它围绕手指的质心对称。由于它围绕手指的质心成正比,所以它不影响手指的位置。另外,充电放大器执行一个  
15 差分测量功能以进一步排除共有模式的噪声。

由于充电积分器电路 44 - 1 至 44 - n 的特性,它们的输出将随时间而改变,并且只对于一个较短的时间具有所需的电压输出。在当前的较佳实施例中,将滤波器电路 48 - 1 至 48 - n 实现为采样和保持切换的电容器滤波器。

20 滤波器电路 48 - 1 至 48 - n 捕获所需的电压。就如控制电路 56 所控制的,滤波器电路 48 - 1 至 48 - n 将从所检测的信号中滤除任何高频噪声。这是通过选择用于滤波器的电容器的电容较大于充电积分器电路 44 - 1 至 44 - n 的输出电容而实现的。另外,本领域的普通技术人员将会认识到所切换的电容器滤波器电路 48 - 1 至 48-n 将捕获所需  
25 的电压并且存储它们。

根据本发明,对从电容测量中以电压形式获得的电容信息进行数字化并以数字形式进行处理。相应地,将滤波器电路 48 - 1 至 48 - n 存储的电压存储在取样/保持电路 50 - 1 至 50 - n 中从而电路的其余部分处理同时取得的输入数据。可以将取样/保持电路 50 - 1 至 50 - n 配置  
30 为普通的取样/保持电路,这是众所周知的。

模数( A/D )转换器 52 对取样/保持电路 50 - 1 至 50 - n 的输出的取样的模拟电压进行数字化。正如在当前较佳实施例中, A/D 转换器

52 将输入电压转化为宽度为 10 位的数字信号( 其分辨率为 1024 ) , 尽管在本领域的普通技术人员也可以实现为其他的分辨率。 A/D 转换器 52 可以是本领域中众所周知的普通的连续逼近型的转换器 ( successive approximation type converter ) 。

5 如果在本发明中使用充电积分器电路, 充电积分器的输出的背景电平( 没有目标出现 ) 大约将是 1 伏。由于手指或者其它目标出现而产生的  $\Delta V$  一般大约 0.4 伏。因此 A/D 转换器 52 的电压范围在大约 1 - 2 伏的范围之间。

10 一个重要的考虑是 A/D 转换器的最大和最小电压参考点(  $V_{min}$  和  $V_{max}$  )。已经发现, 如果这些参考电压是一个固定点, 噪声将造成位置的颤动。在本发明中所使用的对于这一问题的解决方案是从参考电容 42 -  $V_{min}$  和 42 -  $V_{max}$  动态生成  $V_{min}$  和  $V_{max}$  参考电压, 这些参考电容由充电积分器电路 44 -  $V_{min}$  和 44 -  $V_{max}$  检测, 由滤波器电路 48 -  $V_{min}$  和 48 -  $V_{max}$  处理, 并且存储在取样/保持电路 50 -  $V_{min}$  和 50 -  $V_{max}$  中。以此方式, 在从检测器阵列对信号进行取样时所出现的任何共有模式噪声也将出现在  $V_{min}$  和  $V_{max}$  参考电压值中, 因此趋于取消。本领域的普通技术人员将认识到参考电容 44 -  $V_{min}$  和 44 -  $V_{max}$  可以是分立电容器或检测器阵列中的额外的线。

20 根据本发明, 从这样一个电容器中产生  $V_{min}$  参考电压, 该电容器具有等于一个最低电容的值, 该最低电容是在没有目标出现时预期在检测器阵列中出现的电容( 对于 2 平方英寸的检测器阵列大约为 12pF )。从这样一个电容器中产生  $V_{max}$  参考电压, 该电容器具有等于一个最高电容的值, 该最高电容是在没有目标出现时预期在检测器阵列中出现的电容( 对于 2 平方英寸的检测器阵列大约为 16pF )。

25 A/D 转换器 52 的输出向运算单元 16 提供输入。将如参照附图 8 所详细描述, 运算单元 16 的功能是对触摸检测器阵列 22 中的 X 和 Y 方向的各个检测线上的信号进行加权平均计算。因此如图 1 所示, X 输入处理电路 12 和 Y 输入处理电路 14 共享运算单元 16。

30 图 3 的控制电路 56 协调电路的其余部分的操作。由于系统是离散地取样, 并且流水处理的, 因此提供控制电路 56 以管理该信号流。控制电路 56 所执行的功能一般可以通过本领域中众所周知的状态机或微控制器进行开发。

下面将揭示图 3 的各个模块的结构和操作。现在参照图 4a, 4b 和 5 描述一个典型的充电积分器电路。充电积分器电路 44 示出为图 4a 和图 4b 所简化的示意性框图。图 5 示出充电积分器电路 44 的操作定时。控制电路 56 提供这些定时信号。

5 充电积分器电路 44 是基于使用电流向电容器充电这一基本的物理现象。如果以一恒定电流对电容器充电一恒定时间, 电容器上产生的电压将反比于其电容。要充电的电容是与一个内部电容器相并联的检测器矩阵线电容 42。这一内部电容将包括所关心的电压。

10 现在参见图 4a, 示出了示意性充电积分器电路 44 的一个简化的框图。充电积分器电路输入结点 60 连接到检测器矩阵的一个 X ( 或 Y ) 线。一个第一个短路开关 62 连接在充电积分器电路输入结点 60 和电源正极  $V_{DD}$  之间。一个第二个短路开关 64 连接在充电积分器电路输入结点 60 和电源负极地之间。一个正的恒定电流源 66 连接到电源正极  $V_{DD}$  和充电积分器电路输入结点 60, 并且经过第一电流源开关 68。一个负的恒定电流源 70 连接到地和充电积分器电路输入结点 60, 并且经过第二电流源开关 72。很显然可以使用其它高的和低的电压线取代  $V_{DD}$  和地。

20 一个第一内部电容器 74 连接在  $V_{DD}$  和充电积分器电路 44 的输出结点 76 之间。一个正的电压储能开关 78 连接在输出结点 76 和输入结点 60 之间。一个第二内部电容器 80 的一个极板经过开关 82 连接到地, 经过开关 84 连接到充电积分器电路 44 的输出结点 76, 其另一个极板经过负电压储能开关 86 连接到输入结点 60, 经过开关 88 连接到  $V_{DD}$ 。第一和第二内部电容 74 和 80 的电容应该为各个检测器矩阵线的电容的一个很小的比例 ( 比如大约 10 % )。在典型的实施例中, 检测器矩阵线的电容大约是 10pF, 而电容器 74 和 80 的电容大约为 1pF。

30 根据本发明的当前的较佳实施例, 所使用的方法是差分测量, 以增加对噪声干扰的排除, 其好处是去除任何低频率公共模式噪声。为了下面的描述, 假设所有开关是开的, 除非明确指出它们是闭合的。首先经过开关 62 将检测器矩阵线瞬时短接到  $V_{DD}$ , 闭合开关 78 将电容器 74 与检测器线的电容并联。然后通过开关 72 使用来自电流源 70 的一个恒定电流对并联电容器组合进行放电一个固定的时间周期。在固定的时间期间的末尾, 打开开关 78, 从而将检测器矩阵线的电压存储到电容器

74。

然后经过开关 64 将检测器矩阵线瞬时短接到地，闭合开关 82 和 86 将电容器 80 与检测器线的电容并联。闭合开关 68，使用来自电流源 66 的一个恒定电流对并联电容器组合进行充电一个固定的时间周期，该时间  
5 与第一个循环的固定时间周期相等。在固定的时间期间的末尾，打开开关 86，从而将检测器矩阵线的电压存储到电容器 80。

然后将第一个和第二个所测量的电压进行平均。这是通过打开开关 82 和闭合开关 88 和 84 而完成的，这将电容器 80 与电容器 74 并联。由于电容器 74 和 80 具有同样的电容，在它们上面所得到的电压等于在各  
10 个上的电压的平均。这一所得到的值将传送到滤波器电路 48 - 1 至 48 - n 中适当的一个。

低频噪声(众所周知为 50/60Hz)和它们的谐波表现为一个 DC 电流分量，它在一个测量中是加上，在另一个测量中是减去。当将两个结果相加时，将噪声分量平均为零。噪声排除的量是进行两个相反的充电  
15 和放电循环连续继续的快速程度的一个函数，正如这里将要描述的。选择这种充电积分器电路的一个理由是它允许快速地进行测量。

现在参见图 4b，示出了图 4a 的简化图的充电积分器电路 44 的一个示意性实施例的一个更加完整的概图。输入结点 60 通过传输门( pass gate ) 90 和 92 连接到  $V_{DD}$  和地，传输门取代了图 4a 的开关 62 和 64。  
20 传输门 90 由提供给它控制端的信号 ResetUp 控制，传输门 92 由提供给它控制端的信号 ResetDn 控制。本领域的普通技术人员将会认识到传输门 90 和 92 以及其它由图 4b 中的同一符号表示的传输门可以是普通的 CMOS 传输门，这在本领域是众所周知的。所公知的是，在传输门的控制端为低时，它断开，而在其控制端为高时它表现为低阻抗连接。

25 将 P - 沟道 MOS 晶体管 94 和 96 配置成一个电流镜( current mirror )。P - 沟道 MOS 晶体管 94 起到电流源 66 的作用，传输门 98 起到图 4a 开关 68 的作用。信号 StepUp 控制传输门 98 的控制输入。

将 N - 沟道 MOS 晶体管 100 和 102 也配置成一个电流镜。N - 沟道 MOS 晶体管 100 起到电流源 70 的作用，传输门 104 起到图 4a 开关  
30 72 的作用。信号 StepDn 控制传输门 104 的控制输入。P - 沟道 MOS 晶体管 106 和 N - 沟道 MOS 晶体管 108 与 P - 沟道 MOS 电流镜晶体管 96 和 N - 沟道电流镜晶体管 102 串联。P - 沟道 MOS 晶体管 106

的控制栅由使能信号 EN 驱动，它开启 P - 沟道 MOS 晶体管 106 以给电流镜提供能量。这一设备用作一个节能设备，从而在充电积分器电路 44 不使用时可以关闭它。

5 N - 沟道 MOS 晶体管 108 的栅极由一个参考电压 Vbias 驱动，它设置流经电流镜晶体管 96 和 108 的电流。电压 Vbias 由一个伺服反馈电路设置，下面将参照图 10 对之详细描述。本领域的普通技术人员将会认识到本实施例允许实时地（通过长时间的固定反馈）进行校正，从而消除了由于检测器环境的改变的任何长期效应。在本发明的当前实施例中，Vbias 对于所有充电积分器电路 44 - 1 至 44 - n 和 44 - Vmax  
10 和 44 - Vmin 是共同的。

注：适当设置 MOS 晶体管 102 和 108 的大小可以提供温度补偿。这是利用下面的事实而完成的，即 N - 沟道 MOS 晶体管 108 的阈值随温度而降低，而同时 N - 沟道 MOS 晶体管 102 和 108 的迁移率随温度降低。阈值的降低具有增加电流的效应，而迁移率的降低具有减少电流  
15 的效应。通过适当地确定器件的大小这些效应可以在工作范围的一个重要的部分上相互抵销。

电容器 74 的一个极板连接到 V<sub>DD</sub>，另一极板连接到输出结点 76，并通过传输门 110（图 4a 中示为开关 78）连接到输入结点 60。传输门 110 的控制输入由控制信号 SUp 驱动。电容器 80 的一个极板通过传  
20 输门 112（图 4 的开关 86）连接到输入结点 60，通过传输门 114（图 4a 的开关 82）连接到 V<sub>DD</sub>。传输门 112 的控制输入由控制信号 SDn 驱动，传输门 114 的控制输入由控制信号 ChUp 驱动。电容器 80 的另一个极板通过 N - 沟道 MOS 晶体管 116（图 4 的开关 82）连接到地，通过传输门 118（图 4a 的开关 84）连接到输出结点 118。传输门 118  
25 的控制输入由控制信号 Share 驱动。

现在参见图 4a,4b,和图 5 的定时图，可以观察到在一个扫描循环中充电积分器电路 44 的操作。首先，EN（使能）控制信号通过变为 0v 而激活。它开启电流镜和向充电和放电电流源，MOS 晶体管 94 和 100 供能。这时，ResetUp 控制信号变为高，从而将输入结点 60（以及它所连接到的检测器线）短接到 V<sub>DD</sub>。这时 SUp 控制信号也是高，它将  
30 电容器 74 和输出结点 76 连接到输入结点 60。这种安排保证了工作周期后面的放电部分总是从一个已知的平衡状态开始。

在 ResetUp 控制信号不再有效时，开始放电过程。 StepDn 控制信号变为有效，将 MOS 晶体管 100，放电电流源连接到输入结点 60 和其相连的检测器线。 StepDn 有效并持续一个所设置的时间长度，负的恒定电流源将检测器线和电容器 74 的组合的电容进行放电，从而在该时间周期将其电压降低。接着 StepDn 断开。在一个短时间之后， SUp 控制信号变为不再有效，将所测量的电压存储在电容器 74 中。这就结束了放电周期。

接下来， ResetDn 控制信号变为有效，将检测器线与地短接。同时， SDn 和 ChDn 控制信号变为有效，将电容器 80 连接在地和检测器线之间。将电容器 80 对地放电，确保下面的充电周期总是从一个已知的状态开始。

在 ResetDn 控制信号变为不再有效和 StepUp 控制信号变为有效之后，开始充电周期。在该点上，将电流充电电源， MOS 晶体管 94 连接到检测器线，并通过增加其上的电压提供一个恒定的电流以对检测器线充电。 StepUp 控制信号变为有效并持续一个所设置的时间长度（最好等于上述周期的时间长度），以对电容充电，然后将之断开。然后 SDn 控制信号变为不再有效，在电容器 80 上留下了所测量的电压。

现在开始平均周期。首先对电容器 80 上的电压进行电位移动。这可以通过将 ChDn 控制信号变为不再有效，使得电容器 80 的一个极板浮动而进行。接着， ChUp 控制信号变为有效，将电容器的第二极板连接到  $V_{DD}$ 。然后， Share 控制信号变为有效，将电容器 80 的第一极板连接到输出结点 76，从而使得电容器 74 和 80 并联。这具有对两个电容上的电压进行平均的作用，从而去除掉前面所述的公共模式噪声。然后在输出结点 76 上也得到这一平均电压。

本领域的普通技术人员将会认识到，在放电和充电周期在时间上很接近地进行时，在两个周期中所获得的电压的平均过程中所带来的环境的交变电流和其它低频噪声取消特性是最有效的。根据本发明，在这样一个时间期间中 ChDn 和 ChUp 信号应当相互确立，该时间期间比要取消的噪声的周期的 1/4 还小，从而利用了本发明的这一特性。

根据本发明，揭示了两个不同的驱动/检测方法。本领域的普通技术人员很容易就会认识到，参照图 4a,4b 和 5 所说明的充电积分器电路 44 可以适应根据这里所揭示的扫描方法之一而操作。

正如对充电积分器电路 44 的操作的清晰理解，该电路的输出电压只能在一个很短的时间周期中获得，因此易于受到环境噪声的影响。为了减少噪声的这种影响，使用一个切换的电容器滤波器电路 48。现在参见图 6，示出了可以在本发明中使用的一个示意性切换的电容器滤波器电路 48 的概图。本领域的普通技术人员将会认识到，这一切换的电容器滤波器电路包括一个输入结点 120，具有由 Sample 控制信号驱动的一个控制输入的传输门 122，连接在传输门 126 的输出和一个固定的电压（诸如地）之间的一个电容器 124，以及一个输出结点（它包括连接在电容器 124 和传输门 126 的输出之间的公共连接）。在一个典型的实施例中，电容器 116 具有大约 10pF 的电容。

正如本领域的普通技术人员所理解的，切换的电容器滤波器 48 在取样/保持电路部分，其滤波器时间常数是取样周期的 K 倍，这里 K 是电容 124 与其所连接的图 4a 和 4b 的充电积分器电路 44 的电容器 74 和 80 的电容之和的比值。该切换的电容器滤波器电路 48 还减少了系统中的噪声的注入。在较佳实施例中， $K=10/2=5$ 。本领域的普通技术人员将会认识到，其它类型的滤波器电路，比如 RC 滤波器可以用于本发明。

现在参见图 7，示出了图 3 的当前的 A/D 转换器 52 的较佳装置的一个更详细的框图。A/D 转换器的个数小于触摸检测器阵列的线的个数，对 A/D 转换器的输入进行多路复用以在触摸检测器阵列的多条线之间共享多个 A/D 转换器的每一个。将图 7 的装置用于集成电路布局区域将比对每个输入线提供单个 A/D 转换器更为有效。

在图 7 的实施例中，对于图 2a-2d 的检测器阵列 10，假设有 24 条导线。正如图 7 所示，将取样/保持电路 50 - 1 至 50 - 24 的输出馈送到模拟多路转换器 130 的模拟的数据输入。模拟的多路转换器 130 具有六个输出，每一个驱动一个 A/D 转换器 50 - 1 至 52 - 6 的输入。模拟的多路转换器 130 的内部安排是这样的，四个不同的输入多路复用到一个输出。模拟的多路转换器 130 已被概念性地示为 6 个内部多路转换器模块 132 - 1 至 132 - 6。

在图 7 所示的例子中，从取样/保持电路 50 - 1 至 50 - 4 所取得的输入被多路转换到内部多路转换器 132 - 1 的输出，后者驱动 A/D 转换器 52 - 1；类似地，从取样/保持电路 50 - 5 至 50 - 8 所取得的输入被多路转换到内部多路转换器 132 - 2 的输出，后者驱动 A/D 转换器 52



- 2；从取样/保持电路 50 - 9 至 50 - 12 所取得的输入被多路转换到内部多路转换器 132 - 3 的输出，后者驱动 A/D 转换器 52 - 3；从取样/保持电路 50 - 13 至 50 - 16 所取得的输入被多路转换到内部多路转换器 132 - 4 的输出，后者驱动 A/D 转换器 52 - 4；从取样/保持电路 50 - 17 至 50 - 20 所取得的输入被多路转换到内部多路转换器 132 - 5 的输出，后者驱动 A/D 转换器 52 - 5；以及从取样/保持电路 50 - 21 至 50 - 24 所取得的输入被多路转换到内部多路转换器 132 - 6 的输出，后者驱动 A/D 转换器 52 - 6。

模拟的多路转换器 130 具有一组由总线 134 示意性表示的控制输入。在图所示的示意性实施例中，内部多路转换器 132 - 1 至 132 - 6 的每一个是 4 输入多路转换器，因此控制总线 134 可以包括一个 2 位总线用于一个 4 选 1 的选择。本领域的普通技术人员将会认识到，图 7 的安排只是从 24 通道进行 A/D 转换的任务的许多特定方案中的一个，也可以采用其它等同的方案。

在一个直接的译码策略中，多路转换器 132 - 1 至 132 - 6 顺序地将出现在它们第一到第四个输入上的模拟电压分别传送到 A/D 转换器 52 - 1 至 52 - 6 的输入上。在 A/D 转换器 52 - 1 至 52 - 6 的输入上确定了模拟值之后，就在共同的 A/D 控制线 136 上确立一个 CONVERT 命令以开始 A/D 转换过程。

在 A/D 转换过程完成时，表示输入电压的数字值被存储在寄存器 138 - 1 至 138 - 6 中。正如在本发明的较佳实施例中，寄存器 138 - 1 至 138 - 6 每一个可以包括一个 2 - 字寄存器，从而在将第二个字写入该寄存器的同时能够从该寄存器中读出另一个字到运算单元 54，使得系统的速度最块。这种寄存器的设计在本领域中是普通的。

现在参见图 8，提供了运算单元 16 的一个更详细的框图。本领域的普通技术人员将会理解，运算单元 16 处理来自 X 维和 Y 维的信息，即来自图 1 的 X 输入处理电路 12 和 Y 输入处理电路 14 的信息。

在揭示运算单元 16 的结构安排之前，理解根据本发明确定一个目标邻近检测器阵列 22 的质心位置较佳的方法将是有帮助的。

根据本发明的较佳实施例，通过对在检测器阵列 10 上的各个检测线所测量的电容值进行加权平均可以确定在任一方向上的目标位置。在下面的描述中，使用了 X 方向，但是本领域的普通技术人员将会认识

到，这种描述也可用于确定 Y 方向的加权平均。众所周知，可以如下确定加权平均：

$$X_{position} = \frac{\sum_{i=0}^n i \times \Delta C_i}{\sum_{i=0}^n \Delta C_i} \quad (\text{公式 1})$$

5 其中  $\Delta C_i = C_i - C_{0i}$ 。  $C_i$  是在当前第  $i$  条线上测得的电容，  $C_{0i}$  是在过了一些时间没有目标存在时在同一个线上测得的值。考虑到这些过去和当前的电容测量值，该位置可以表示为

$$X_{position} = \frac{\sum_{i=0}^n i \times (C_i - C_{0i})}{\sum_{i=0}^n (C_i - C_{0i})} \quad (\text{公式 2})$$

10 利用乘法对加法的分配性，可以看出这一表示式等价于

$$X_{position} = \frac{-\sum_{i=0}^n (i \times C_{0i}) + \sum_{i=0}^n (i \times C_i)}{-\sum_{i=0}^n (C_{0i}) + \sum_{i=0}^n (C_i)} \quad (\text{公式 3})$$

15 在分子和分母中的负项是偏移值，表示在没有目标出现时的背景值。如果用项  $O_N$  表示分子偏移值，用  $O_D$  表示分母偏移值，则公式 3 可表示为：

$$X_{position} = \frac{-O_N + \sum_{i=0}^n (i \times C_i)}{-O_D + \sum_{i=0}^n (C_i)} \quad (\text{公式 4})$$

20 现在参见图 8，可以看出，运算单元 16 包括 X 分子和分母累加器 150 和 152 和 Y 分子和分母累加器 154 和 156。X 分子和分母累加器 150 和 152 以及 Y 分子和分母累加器 154 和 156 的操作数的来源是图 1 的检

测器阵列 22 的每一方向( X 和 Y )的寄存器 138 - 1 至 138 - 6。X 和 Y 分母累加器 152 和 156 将 A/D 转换的结果相加。X 和 Y 分子累加器 150 和 154 计算输入数据的加权和而不是直接的和。累加器 150, 152, 154 和 156 可以设计成一个硬件单元或者是在一个微处理器上的软件, 这在本领域中很容易理解。

从图 8 中可以看出, 分子累加器 150 和 154 计算公式 4:

$$\sum_{i=0}^n i \times C_i \quad (\text{公式 5})$$

分母累加器 152 和 156 计算公式 4:

$$\sum_{i=0}^n C_i \quad (\text{公式 6})$$

从存储在加法器 166, 168, 170 和 172 中的累加器 150, 152, 154 和 156 的结果中减去 X 和 Y 分子和分母偏移寄存器 158, 160, 162 和 164 中的内容。加法器 166 减去 X 分子偏移寄存器 158 中的偏移值  $O_{NX}$ 。加法器 168 减去 X 分母偏移寄存器 160 中的偏移值  $O_{DX}$ 。加法器 170 减去 Y 分子偏移寄存器 162 中的偏移值  $O_{NY}$ 。加法器 172 减去 Y 分母偏移寄存器 164 中的偏移值  $O_{DY}$ 。分子分母对被除法部件 174 和 176 相除以产生 X 和 Y 位置数据, 框 178 使用 X 和 Y 分母对以产生 Z 轴( 压力 ) 数据。稍后将描述框 178 所执行的功能。在校正单元 180 的指挥下, 从累加器的内容中取样偏移值  $O_{DX}$ ,  $O_{NX}$ ,  $O_{DY}$ , 和  $O_{NY}$ 。

本领域的普通技术人员很容易理解, 本发明的系统的结构可以以多种方式分布, 其中几种包括使用微处理器, 它可以在一个主计算机中, 本发明的系统与之相连, 或者在这里所描述的集成电路和主计算机之间。本发明的实施例也可考虑为其中表示和项的累加的分子和分母值与  $O_N$  和  $O_D$  偏移值一起分送到这一处理器中供处理, 或者所有的处理由本领域所公知的所编程的微处理器完成。

开始, 在系统启动期间, 将分子和分母累加器 150, 152, 154 和 156 设置为 0。如果使用图 7 所示的多路复用的 A/D 转换器, 将寄存器 138 - 1 的第一个字中的数字化的电压数据表示取样/保持电路 50 - 1

的输出电压) 加到累加器的和中, 并将其结果存储在该累加器中。连续地, 将存贮在寄存器 138 - 2 至 138 - 6 的第一个字中的数字化的电压数据( 分别表示取样/保持电路 50 - 5, 50 - 9, 50 - 17 和 50 - 21 的输出电压 ) 加到累加器的和中, 并将其结果存储在该累加器中。

5 如前所示, 在这时, A/D 转换器 52 - 1 至 52 - 6 可以将取样/保持电路 50 - 2, 50 - 6, 50 - 10, 50 - 14, 50 - 18 和 50 - 22 的输出电压进行转换并将其数字化值分别存储在寄存器 138 - 1 至 138 - 6 的第二个字中。

10 接下来, 连续地将存贮在寄存器 138 - 1 至 138 - 6 的第二个字中的数字化的电压数据( 分别表示取样/保持电路 50 - 2, 50 - 6, 50 - 10, 50 - 14, 50 - 18 和 50 - 22 的输出电压 ) 加到累加器的和中, 并将其结果存储在该累加器中。

15 接下来, 连续地将存贮在寄存器 138 - 1 至 138 - 6 的第一个字中的数字化的电压数据( 分别表示取样/保持电路 50 - 3, 50 - 7, 50 - 11, 50 - 15, 50 - 19 和 50 - 23 的输出电压 ) 加到累加器的和中, 并将其结果存储在该累加器中, 接下来将数字化的电压值存储在寄存器 138 - 1 至 138 - 6 的第二个字中( 分别表示取样/保持电路 50 - 4, 50 - 8, 50 - 12, 50 - 16, 50 - 20 和 50 - 24 的输出电压 )。

20 在这一时间点上, 累加器保持所有数字化的电压值的和。现在分别从存储在分子和分母累加器中的值中减去存储在  $O_N$  和  $O_D$  偏移寄存器 158 和 164 中的数字值。接着在除法器 174 和 176 中的除法操作完成加权平均计算。

25 也可以由外部微处理器进行除法运算, 该外部微处理器可以取得存储在累加器中的值或者其自身进行累加。当前由于  $O_N$  和  $O_D$  偏移值是由一个外部微处理器导出的, 由这一除法运算给该外部微处理器所带来的额外处理开销是最小的。另外, 可以在一个芯片中包括一个专用微处理器, 以处理这些任务而不偏离这里所揭示的本发明。

30 上述的处理在 1 毫秒中发生, 并且可以重复进行。当前的鼠标标准是每秒更新位置信息 40 次, 因此本发明的装置可以容易地以这种重复的速率工作。

由于在本发明中所使用的方法的自然属性, 因此在本发明的系统中, 存在这样一个时机, 它能提供额外的噪声免疫而又不要求额外增加

硬件。尽管很显然，在上述序列执行之后，可以清除累加器和重复该过程，也可以允许该这些值留在累加器中。如果这样的话，可以实现一个平均函数以进一步滤除噪声。根据本发明的这一方面，进行多次取样，并通过累加器执行，在处理序列的末尾不清除它们。正如在本发明的  
5 5 较佳实施例中，在系统中获得一个除法结果之前处理 25 个样值，从而极大地减少了瞬间系统噪声尖峰的影响。本领域普通技术人员将会认识到，在清除累加器之前所取的采样的个数受设计时的一些选择的影响，诸如数据获得速率，数据处理速率等。

最好对图 8 的运算单元 16 的除法模块 174 和 176 产生的 X 和 Y 位  
10 10 置进行另外的过滤。这一过滤最好在图 1 的运算单元 16 和移动与动作单元 18 与 20 之间进行。该 X 和 Y 坐标作为单独的数字被分别过滤。每一个过滤器是一个计算“运行平均值”（running average）的平均寄存器，这在本领域中众所周知。在第一次检测到一个手指时，用当前的系数初始化该过滤寄存器。在随后的示例中，这一新的系数被过滤寄存器的  
15 15 值平均，以产生一个新的过滤寄存器值。在当前的较佳实施例中，这些值在平均中权重相同，虽然可以使用不同的权重以提供较强或较弱的过滤。在过滤寄存器中的值的序列用作 X 和 Y 坐标，图 1 的移动和动作单元 18 和 20 可以使用这些坐标。

本发明的系统适应于改变的条件，比如部件老化，由于湿度引起的  
20 20 电容改变，触摸表面的污染等。另外，本发明有效地减少了环境噪声。根据本发明，以三种方式考虑这些效应。首先动态地更新偏移值  $O_N$  和  $O_D$  以适应变化的情况。其次提供一个伺服反馈电路以确定用于设置充电积分器电路 44 - 1 至 44 - n 的偏置的偏置电压。第三，如前所揭示的，也动态地改变 A/D 转换器的  $V_{max}$  和  $V_{min}$  的参考电压点，以提高信噪  
25 25 比的极限。

现在参见图 9，提供了校正单元 150 的一个框图，该校正单元可以与图 8 的运算单元一起使用。校正单元 150 执行一个算法通过试探地确定何时没有手指或其它传导目标邻近触摸检测器阵列 22 来建立分子和分母偏移值。

30 30 如前所揭示的， $O_N$  和  $O_D$  偏移值表示没有目标存在时的阵列电容的基线值。根据本发明这些值也要更新，因为太高或者太低的基线值电平根据误差的符号具有移动目标的外观位置的效应。在没有目标出现在检

测器阵列 22 上时通过选择所读取的值建立这些值。由于没有一个外部方式来“知道”何时没有目标出现在检测器阵列 22 上，使用根据本发明的另一方面一个算法来建立和动态地更新这些偏移值。在校正单元观察到一个 Z 值时，该值是在没有手指出现时的典型的 Z 值，它指示偏移寄存器（图 8 的 158，160，162，和 164）从累加器的当前值进行再加载。根据本发明的当前较佳实施例，对更新偏移值的判断是基于 X 或 Y 方向的一个上的检测器阵列 22 的行为，但是在进行该判断时，更新所有 4 个偏移值（ $O_{NX}$ ， $O_{DX}$ ， $O_{NY}$ ， $O_{DY}$ ）。在本发明的其它实施例中，更新的判断可以是根据上述的准则独立地对于每一方向进行的。

校正算法通过监视分母累加器值的所选择的一个的变化进行工作。根据本发明，在触摸检测器阵列 22 的一组导电线的电容变化的灵敏度大于在触摸检测器阵列 22 的其它一组导电线的电容变化的灵敏度。经验告诉我们，该组对电容变化具有较高灵敏度的导电线是物理地位于在另一方向上的导电线上面的那一组，因此距离该检测器阵列 22 的触摸表面最近。较上的导电线组趋向于部分地屏蔽较下的导电线组在检测器阵列 22 的表面上出现的电容的变化。

通过将检测线上的所测得的电容相加获得手指压力。在减去偏移  $O_D$  之后该值已经出现在分母累加器中。如果该压力超过一个适当的阈值，表示出现一个手指。可通过试验确定这一阈值，它是表面材料和电路定时的一个函数。可以调整该阈值以适应不同用户的要求。

该装置所报告的压力是 X 和 Y 方向的分母的一个简单的函数  $f(X_D, Y_D)$ ，正如在图 8 的框 178 所实现的。可能的函数包括选择一个较佳的分母值，或者将分母相加。在当前的较佳实施例中，选择两个分母中较小的一个。如果手指轻轻地滑离垫的边缘，这一选择具有使得该压力低于阈值的理想效果，其中 X 检测器产生有效值，而 Y 检测器不产生，或者相反。这起到一个电子档板（bezel）的作用，它取代了检测器区域的周边的机械档板。

在图 8 的例子中，选择 Y 分母用于监视，因为它最敏感。为了校正算法的目的，将所选择的分母称作 Z。这一分母的当前所保存的偏移值称作  $O_Z$ 。

校正算法的目的是跟踪静止 Z 电平的逐渐变化，同时保证既不校正

手指也不校正从噪声引起的瞬间尖峰。正如本领域普通技术人员从下面的描述所能理解的，校正算法能够以数字或模拟硬件的方式或者以软件的方式实现。在由本发明人所试验的当前实际的实施例中，是以软件方式实现的。

- 5 在 Z 值到达校正单元时，将它们通过过滤器 182 传送。与过滤器 182 共同工作的历史缓冲区 184 保持最近 Z 值一个“运行平均值”。在一个新的 Z 值到达时，根据下面的公式更新当前的运行平均值  $F_z$ ：

$$\text{新 } F_z = \alpha (\text{旧 } F_z) + (1 - \alpha) Z \quad (\text{公式 7})$$

10

其中  $\alpha$  是在 0 和 1 之间的一个常数因子，一般接近 1，Z 是当前 Z 值。在较佳实施例中， $\alpha$  大约是 0.95。其目的是使  $F_z$  足够慢地改变以跟随逐渐的变化，不会受到 Z 的短的扰动的大的影响。

- 15 过滤器 182 从控制单元 186 中接收一个信号 ENABLE。只有在 ENABLE 确立时基于新的 Z 值更新运行平均值  $F_z$ 。如果撤销 ENABLE， $F_z$  保持常数，并且不受当前 Z 的影响。

- 20 历史缓冲区 184 记录  $F_z$  的几个最近值。在当前的实施例中，历史缓冲区记录两个先前的  $F_z$  值。历史缓冲区可以以一个移位寄存器，循环队列或模拟延迟线实现。在历史缓冲区从控制单元 186 接收到一个 REWIND 信号时，它将当前运行平均值  $F_z$  恢复为最老的所保存的值。它就好象过滤器 182 倒回禁止了一段时间，该时间对应于历史缓冲区的深度。历史缓冲区的目的是允许这种回溯禁止。

- 25 由绝对差值单元 188 和 190 以及比较器 192 将当前运行平均值  $F_z$  与当前的 Z 值和当前的偏移值  $O_z$  进行比较。绝对差值单元 188 将 Z 和  $F_z$  相减，输出它们的差值的绝对值。绝对差值单元 190 将  $O_z$  和  $F_z$  相减，输出它们的差值的绝对值。如果绝对差值单元 188 的输出小于绝对差值单元 190 的输出，即  $F_z$  离 Z 比它离  $O_z$  更近，则比较器 192 确立 UPDATE 信号。在 Z 的平均值移动到一个新的静止电平时，UPDATE 信号总是趋于被确立。在 Z 从它的正常静止电平进行一个短暂的偏移时它总是趋于不被确立。滤波器常数确定一个偏移的长度，为此目的，它将被认为是“短暂的”。

- 30 减法器单元 194 是一个简单的减法器，它计算 Z 和  $O_z$  的带符号的

差值。该减法器实际上与图 8 的减法器 172 重复，因此在实际的实现上，可以合并在一起。这一减法器的输出  $C_Z$  是所校正的  $Z$  值，是手指压力的一个估计值。比较器 196 和 198 将这一压力值与一个正的和负的阈值进行相比。这些阈值示为  $Z_{TH}$  和  $-Z_{TH}$ ，虽然在实际上不要求它们在幅度上相等。

5 如果压力信号  $C_Z$  大于  $Z_{TH}$ ，确立信号 FINGER，表示可能出现一个手指。校正单元所使用的  $Z_{TH}$  阈值类似于本系统的其余部分检测手指的存在所使用的阈值，或它可以具有一个不同的值。在当前的实施例中，将校正信号  $Z_{TH}$  设置得略小于主  $Z_{TH}$ ，以使校正单元能够对于手指的出现作出一个稳健的选择。

10 如果压力信号  $C_Z$  小于  $-Z_{TH}$ ，确立信号 FORCE。由于有意使  $O_Z$  等于没有手指出现的  $Z$  的静止值，并且手指只能增加检测器的电容，从而  $Z$  的值，主要是负的  $C_Z$  意味着该装置自身对于手指具有不正确的校正，而手指已经移走了。既然手指不再存在，校正逻辑 200 使用这一事实以强制进行再校正。

15 控制逻辑 186 负责防止运行平均值  $F_Z$  受到在手指出现时产生的  $X$  值的影响。在 FINGER 信号为真时，输出 ENABLE 信号总是 off，而在 FINGER 信号为假时，总是 on。然而在 FINGER 从假向真过渡时，控制逻辑也产生 REWIND 信号脉冲。在 FINGER 从真向假过渡时，控制逻辑在确立 ENABLE 之前等待一个短的时间（相比于历史缓冲区的深度）。从而，防止在手指出现时运行平均值接着  $Z$ ，以及在手指出现之前和之后保持一个较短的时间。

20 校正逻辑 200 从三个比较器 192，196 和 198 的输出中产生信号 RECAL。在确立 RECAL 时，将从当前的累加器值中重新装入偏移寄存器  $O_N$  和  $O_D$ 。从下面的逻辑表达式中产生 RECAL：

$$\text{RECAL} = \text{FORCE 或 ( UPDATE 与 ( 非 FINGER ) )} \quad (\text{公式 8})$$

30 另外，在每次系统被首先初始化时，也有可能在一个较短的周期之后等待充电积分器电路和其它电路稳定，校正逻辑 200 将确立 RECAL。

从控制逻辑 186 和校正逻辑 200 的描述中，本领域普通技术人员很显然能够使用普通的逻辑以简单的和常规的逻辑设计配置这些模块。



很显然，对于本领域的任何技术人员来说，所描述的校正算法并不特定于当前发明的电荷积分器和累加器的具体系统。而是它能实现于任一触摸检测器中，该检测器产生接近或压力数据，其中在没有手指或尖刺噪声出现时它最好能维护反映检测器的状态的一个校正点。

5 现在参见图 10，以框图形式显示了在本发明中所使用的一个偏压产生电路 46。根据本发明的当前较佳实施例，尽管本领域的普通技术人员可认识到其它装置也是可行的，充电积分器电路 44 - 1 至 44 - n 的所有偏置晶体管 108（图 4b）的栅极连接到一个偏置电压源。还有许多方法可以产生充电积分器电路 44 - 1 至 44 - n 所需要的偏置电压。

10 观察图 10 可以看出，偏置电压产生电路 46 是一个强衰减（overdamped）伺服系统。一个接近一个典型的充电积分器电路 44 - 1 至 44 - n 的电流源功能的参考电源包括其一个极板接地的电容器 202。其另一个极板经过一个第一传输门 204 连接到电源  $V_{DD}$ ，经过一个第二传输门 208 连接到一个电流源晶体管 206。滤波器电路 210 与滤波器电路 48 - 1 至 48 - n 相同并受到与滤波器电路 48 - 1 至 48 - n 相同的信号控制，将滤波器电路 210 连接成取样电容器 202 的电压，其方式与滤波器和取样/保持电路 48 - 1 至 48 - n 对检测器阵列 22 中的检测器导体电容的电压进行取样相同。

20 将滤波器电路 210 的输出送到一个微弱互导放大器 212 的非反向输入端，后者具有一个范围在  $0.1 - 0.2 \mu A$  的偏置电流。互导放大器 212 的反向输入端被连接到大约为 1 伏的一个固定电压，该电压例如由二极管 214 和电阻 216 产生。互导放大器 212 的输出被电容器 218 分流，同时也经过传输门 222 被电容器 220 分流。将电容器 220 选择得比电容器 25 218 大很多。在本发明的典型的实施例中，电容器 218 可以大约  $0.2pF$ ，电容器 220 大约  $10pF$ 。

电容器 220 连接到 N - 沟道 MOS 晶体管 224 的栅极，晶体管 224 的漏极连接到 P - 沟道 MOS 晶体管 226 的漏极和栅极，其源极连接到 N - 沟道 MOS 晶体管 228 的漏极和栅极。P - 沟道 MOS 晶体管 226 的源极连接到  $V_{DD}$ ，N - 沟道 MOS 晶体管 228 的源极连接到地。晶体管 30 224 和 228 的共同漏极连接是偏置电压输出结点。

在一个固定电压源（也就是大约 2 伏）和电容器 220 之间可以选

择连接一个传输门 230。传输门 230 可以用于在启动时通过将电容器 220 充电到该固定电压而初始化偏置产生电路 46。

5 在每一取样周期，滤波器电路 210 进行一次新的采样。如该新的采样不同于前一采样，互导放大器 212 的输出电压将改变，并且开始充电或放电电容器 218 到一个新的电压。传输门 222 短间接通（也就是大约 1 微秒），电容器 218 和 220 的电压试着将它们自己进行平均。由于电容器 218 和 220 之间的较大规格差别，在传输门 222 打开时，电容器 218 在该期间不能够提供足够的电荷以平衡该电压。这一安排防止了从一个周期到另一个周期偏置电压的大的改变。

10 电容器 202 看上去应该象一个检测器阵列通道，其值等于一个典型的检测器线的背景电容（也就是没有目标接近或出现在电容部件时的电容）。可以以几种方式形成电容器 202。电容器 202 可以在检测器阵列的一部分中包括一个额外的检测器线，并且配置成逼近一个活跃的检测器线但由一个地平面等与手指电容相屏蔽。另外，电容器 202 可以是在集成电路中形成或者连接在其上的电容器，并且其所选的值与一个典型的检测器线的相匹配。在这一方面，包括电容器 210 和滤波器电路 210 的信号源有些类似产生  $V_{MAX}$  和  $V_{MIN}$  参考电压的电路，这时它模仿一个典型的检测器线。

20 另一个可替代的方案是利用一条实际的检测器线设置偏置电压。将在两个端点（end-point）检测器线上所测量的电压进行比较，在理论上，如果一个手指或其它目标接近该检测器阵列，可以选择具有最小值的那一个，它不会出现在位于该阵列的相对边缘上的检测器线上。

25 根据本发明的另一个方面，在将本发明的目标位置检测器用作计算机光标控制设备以取代鼠标器时，可以实现“边缘移动”特性。在试着在一个计算机屏幕上在一个较大距离上移动一个目标时，使用计算机鼠标器或其它光标控制设备会产生一个实际的问题。这一问题是在将一个小的鼠标垫用于一个计算机鼠标器时，或者在这里所描述的目标位置检测传感器具有一个较小的触摸检测器区域时碰到。

30 在触摸检测器应用中，在“拖动”动作期间这一问题尤其尖锐。如果用户抬起手指以开始第二个笔画时，则该拖动效果在屏幕上过早结束。本发明的边缘移动效应有助于避免使用手指的“划动（rowing）”或多笔划以在屏幕上移动较大的距离。

对于长距离拖动问题的以前的解决方案是提供一种加速特性，即“弹道”曲线，其中其增益以手指速度的函数变化，尽管笨拙，但允许用户使用一种重复的手指快速划动动作以移动较长的距离。可将这种技术用于任一速度可变的指点设备中，比如在有限大小的鼠标垫中的鼠标器中。典型的鼠标驱动程序软件包括一个可调节的加速特性（有时以一种易使人误解的名字“鼠标速度”）。

根据本发明的一个当前较佳实施例，由图 1 的移动单元 18 实现目标位置检测器的边缘移动特性，并且通过在包含触摸检测器阵列 22 的检测表面 10 中定义两个区域而工作。如图 11 所示，最好将检测表面 10 分成一个内部区域 240 和一个外部区域 242，该内部区域 240 包括该检测平面 10 表面的中央部分的大部分，而该外部区域 242 典型地包括在检测器阵列的周边的细的边缘区域。可以将检测平面 10 的中心描述为在笛卡尔坐标系中的原点（ $X_{center}$  和  $Y_{center}$ ）。然而本领域的普通技术人员将会认识到内部和外部区域可以是任何形状。

因此，在图 11 中，由上虚线  $Y_0$ ，右虚线  $X_0$ ，下虚线  $-Y_0$ ，和左虚线  $-X_0$  定义内部区域 240。而外部区域 242 是由  $Y_{max}$ ， $-Y_{max}$ ， $X_{max}$  和  $-X_{max}$  定义的检测平面 10 的外边缘与由  $Y_0$ ， $X_0$ ， $-Y_0$  和  $-X_0$  定义的内部区域 240 的外边界之间的区域。

根据本发明的这一方面，以标准的方式将内部区域 240 的手指移动翻译成移动事件，送给主计算机。正如在本领域所理解的，将鼠标移动传送给主计算机的标准方式可以用于在本发明中将手指移动传送给主计算机。在如此所揭示的建立了手指位置之后，传送给主计算机的信息是

$$\Delta X = A ( X_{cur} - X_{old} ) \quad ( \text{公式 9} )$$

$$\Delta Y = A ( Y_{cur} - Y_{old} ) \quad ( \text{公式 10} )$$

其中  $\Delta X$  是手指在 X 位置的改变， $\Delta Y$  是手指在 Y 位置的改变， $X_{cur}$  是当前手指的 X 位置， $X_{old}$  是上一次报告的手指的 X 位置， $Y_{cur}$  是当前手指的 Y 位置， $Y_{old}$  是上一次报告的手指的 Y 位置；A 是“增益因子”，它通常在鼠标器光标控制应用中使用。

一般地，主计算机获取（ $\Delta X, \Delta Y$ ）事件，并且在每一个轴上将

光标移动一个所指示的量，从而在累积连续的  $\Delta X$  和  $\Delta Y$  的值时重建屏幕上的手指的位置。到目前，这还是没有考虑边缘移动特性的标准的光标控制动作。

5 根据本发明，在报告手指在外部区域 242 时，可以允许本发明的边缘移动特性。确定手指是否在外部区域就是简单地判定：

$$[-X_0 < X_{cur} < X_0] \text{是假, 或者} [-Y_0 < Y_{cur} < Y_0] \text{是假} \quad (\text{公式 11})$$

10 现在参见图 12A，它以概图形式示出了判定手指是否在外部区域 242 的电路 244。图 12A 示出了判定手指是否在外部区域 244 的硬件实施例，但是本领域的普通技术人员很容易就会认识到这些判定可以容易地以一种等价的软件例程实现。从这里所描述的功能中，这种软件例程是明显的，和直截了当的。

15 电路 244 包括数字比较器 246，248，250 和 252，它们可以由通用逻辑直接实现。在一个输入端上的量  $X_{cur}$  大于在其另一个输入端上的固定量  $X_0$  时比较器 246 产生一个真信号。在一个输入端上的量  $X_{cur}$  小于在其另一个输入端上的固定量  $-X_0$  时比较器 248 产生一个真信号。在一个输入端上的量  $Y_{cur}$  大于在其另一个输入端上的固定量  $Y_0$  时比较器 250 产生一个真信号。在一个输入端上的量  $Y_{cur}$  小于在其另一个输入端上的固定量  $-Y_0$  时比较器 252 产生一个真信号。

20 比较器 246，248，250 和 252 的输出由或门 254 进行或运算。正如本领域的普通技术人员所认识到的，只有在满足公式 11 的必要条件时，或门 254 的 FingerOuter 信号输出才是真。

25 当前最好是由用户选择允许或禁止本发明的边缘移动特性。在允许边缘移动特性以及报告手指位于上述的外部区域时，将第二个分量加到所报告的  $(\Delta X, \Delta Y)$  事件上。

$$\Delta X = A ( X_{cur} - X_{old} ) + S ( X_{cur} - X_{center} ) \quad (\text{公式 12})$$

30 
$$\Delta Y = A ( Y_{cur} - Y_{old} ) + S ( Y_{cur} - Y_{center} ) \quad (\text{公式 13})$$

其中  $X_{center}$  是该垫板的中心的 X 坐标， $Y_{center}$  是该垫板的中心的 Y

坐标，S是用于速度的一个乘法因子。S的选择应当使得光标以一个适当的速度在显示屏幕上移动。

比如，如果将手指保持在距离右边一个适当的距离（从而  $X_{cur} > X_0$ ），然后光标将趋向于以公式 12 和 13 乘数速度因子 S 设置的固定速度向右边“滑动”。这一因子可以根据用户的个人爱好调节。

如果检测器阵列在 X 和 Y 具有不同的维数，将 X 和 Y 方向的乘数速度因子 S 参数设置得与垫板的维数的比率不同是有用的，从而保持在检测器阵列的左边或右边缘的手指将产生与保持在顶部或底部边缘的手指同样的光标速度。在检测器阵列的当前较佳实施例中，有 24 个 X 线和 18 个 Y 线。因此，由于 X 是 Y 的 4/3（24 线比 16 线），将 X 乘数速度因子  $S_x$  设置为乘数速度因子  $S_y$  的 3/4。

在边缘移动期间光标的滑动速度显然是手指距离垫板的中心的距离的一个直接函数，其滑动方向等于从中心到手指的方向。如果外部区域如图 11 所示具有较佳的“边缘边际”形状，不论何时激活边缘移动，则手指总是大致距离中心同一距离（假设一个方垫板，在 2 的平方根的系数=1.41 内）。从而其心理效应是边缘移动涉及一个固定的滑动速度，其中方向由在检测器阵列周围的位置设置。

通过归一化下列形式的系数以公式（12 和 13）将边缘移动项分解可以取消 2 的平方根变化，但是这是一个计算密集型步骤，它是用于克服普通用户就可注意到的一个问题，因此可以省略。

$$\sqrt{(X_{cur} - X_{center})^2 + (Y_{cur} - Y_{center})^2} \quad (\text{公式 14})$$

正如上面所讨论的，当一个手指位于外部区域中时，作为 X 和 Y 轴的一个全局信号的 FingerOuter 信号为真，并且根据公式 12 和 13 将增量加到  $(\Delta X, \Delta Y)$  事件上。对于  $(\Delta X, \Delta Y)$  事件，由于在 X 和 Y 方向分别加上对应于  $S(X_{cur} - X_{center})$  和  $S(Y_{cur} - Y_{center})$  的增量，因此光标移动的方向将沿着从垫板的中心至手指位置的向量。在图形环境中，有许多垂直和水平目标，并且使用边缘移动可能引起一些意外的结果。比如，如果一个图形环境中的用户下拉一个大的弹出菜单，该用户可能需要借助边缘移动特性到达该菜单的底部。然而在这种情况下，当用户实际上想让光标沿着 Y 轴在垂直方向上移动时，光标的移动方向

可能使得光标滑落该弹出菜单。

在本发明的边缘移动特性的另一个实施例中，光标的移动方向将与手指在移动到外部区域所交叉的边缘移动边界正交。

比如，在手指与对应于 X 轴的边缘移动边界（触摸垫的右和左边缘）相交叉，并且移动到外部区域时，由于该正交的边缘移动特性，光标移动的方向将只沿着 X 轴。当手指位于对应于 X 轴的外部区域中时，光标在 Y 方向上的任何移动将以正常的方式发生，即不根据边缘移动特性发生。类似地，在手指与对应于 Y 轴的边缘移动边界（触摸垫的顶和底边缘）相交叉，并且移动到外部区域时，由于该边缘移动特性，光标移动的方向将只沿着 Y 轴。当手指位于对应于 Y 轴的外部区域中时，光标在 X 方向上的任何移动将以正常的方式发生，即不根据边缘移动特性发生。然而，应该理解，在手指进入该垫的四个角落的任一个，从而与 X 和 Y 边缘移动边界相交叉时，边缘移动实际上是沿着从垫的中央到在该角落的手指的向量。

与使用全局 FingerOuter 信号表示手指位于外部区域的边缘移动特性不同的是，该正交的边缘移动特性有两个信号，一个是 X FingerOuter，而另一个是 Y FingerOuter。当手指越过对应于 X 轴的边界之一时，在较佳实施例中，这时手指位于触摸垫的右和左边缘，X FingerOuter 为真；当手指越过对应于 Y 轴的边界之一时，在较佳实施例中，这时手指位于触摸垫的顶和底边缘，Y FingerOuter 为真；

在图 12B 中，示出了用于判断手指是否位于对应于该正交边缘移动特性的外部区域的硬件示意图。对于本领域的普通技术人员来说，可以理解图 12B 所示的电路可以用一个等价的软件程序实现。在适当的地方，图 12B 使用图 12A 所示的同样的参照数字。

现在参见图 12B，电路 256 包括数字比较器 246，248，250 和 252，这些比较器可以用常规逻辑实现。当在比较器 246 的一个输入端上的量  $X_{cur}$  大于在其另一输入端上的一个固定的量  $X_0$  时，比较器 246 产生一个真信号。当在比较器 248 的一个输入端上的量  $X_{cur}$  小于在其另一输入端上的一个固定的量  $-X_0$  时，比较器 248 产生一个真信号。当在比较器 250 的一个输入端上的量  $Y_{cur}$  大于在其另一输入端上的一个固定的量  $Y_0$  时，比较器 250 产生一个真信号。当在比较器 252 的一个输入端上的量  $Y_{cur}$  小于在其另一输入端上的一个固定的量  $-Y_0$  时，比较

器 252 产生一个真信号。

OR ( 或 ) 门 258 将比较器 246 和 248 的输出进行或运算。当  $-X_0 < X_{cur} < X_0$  为假时, OR 门 258 输出一个真信号, 即 X FingerOuter。

5 OR ( 或 ) 门 260 将比较器 250 和 252 的输出进行或运算。当  $-Y_0 < Y_{cur} < Y_0$  为假时, OR 门 260 输出一个真信号, 即 Y FingerOuter。

相应地, 一个 X FingerOuter 信号将产生一个值, 该值的大小由公式 12 的  $S ( X_{cur} - X_{center} )$  确定, 该值要被加到  $\Delta X$  事件上。然而, 由于只有在已经越过 X 轴的边缘移动边界的时刻才加上这一分量, 也就是在仅仅已经越过一个 Y 轴的边缘移动边界时并不将由  $S ( X_{cur} - X_{center} )$  确定的额外分量加到  $\Delta X$  事件上, 由于  $X_{cur} - X_{center}$  近似于为常量, 因此加到该  $\Delta X$  事件的分量也近似于为常量。

15 一个 Y FingerOuter 信号将产生一个值, 该值的大小由公式 13 的  $S ( Y_{cur} - Y_{center} )$  确定, 该值要被加到  $\Delta Y$  事件上。然而, 由于只有在已经越过 Y 轴的边缘移动边界的时刻才加上这一分量, 即在仅仅已经越过一个 X 轴的边缘移动边界时并不将由  $S ( Y_{cur} - Y_{center} )$  确定的额外分量加到  $\Delta Y$  事件上, 由于  $Y_{cur} - Y_{center}$  近似于为常量, 因此加到该  $\Delta Y$  事件的分量也近似于为常量。

相应地, 正交的边缘移动特性允许光标仅在八个“圆弧”方向移动。比如, 如果手指位于触摸垫的左边缘, 该正交的边缘移动特性使得光标沿着 X 轴相左滑动, 因为 X FingerOuter 是真, 同时 Y FingerOuter 为假。对于垫的右边缘也是如此, 只是光标将沿着 X 轴向右滑动。本领域的普通技术人员将会认识到, 对于垫的上和下边缘将产生相应的移动。在手指在垫的任一个角落上时, X FingerOuter 和 Y FingerOuter 都为真, 光标将沿着从垫的角落上到手指所处位置的 45 度的方向 ( 假设是一个方形触摸垫 ) 移动。

25 如果用户不期望这一边缘移动特性, 则本发明的边缘移动特性可能使人迷惑。由于边缘移动在拖动操作中最有用, 最好将其安排成只在拖动期间发生, 即只在动作逻辑实际上处于“保持鼠标按钮按下”期间发生。使用图 1 的动作单元 20 实现拖动动作和其它动作。

30 在不需要边缘移动功能时, 外部区域 242“被丢在一边”(即忽略), 将内部区域 240 有效地延展到整个检测平面 10。已经发现, 实际上这较少有迷惑性, 可能是因为在拖动动作期间比单一的光标移动期间用户

更容易感知光标控制设备。

假设图 11 的较佳区域边界形状，利用下列算法实现本发明的边缘移动特性：

```
IF NOT (  $-X_0 < X_{cur} < X_0$  AND  $-Y_0 < Y_{cur} < Y_0$  )
5  AND ( 可选地 ) 在进行一个拖动动作, THEN
    使  $eX = S_X ( X_{cur} - X_{center} )$ 
    使  $eY = S_Y ( Y_{cur} - Y_{center} )$ 
    ELSE
    使  $eX = eY = 0$ .
10  END IF
    另外, 对于正交的边缘移动, 其算法如下:
    IF NOT (  $-X_0 < X_{cur} < X_0$  ) AND ( 可选地 ) 在进行一个拖动动作,
    THEN
    使  $eX = S_X ( X_{cur} - X_{center} )$ 
15  ELSE
    使  $eX = 0$ 
    END IF
    IF NOT (  $-Y_0 < Y_{cur} < Y_0$  ) AND ( 可选地 ) 在进行一个拖动动作,
    THEN
20  使  $eY = S_Y ( Y_{cur} - Y_{center} )$ 
    ELSE
    使  $eY = 0$ 
    END IF
```

接下来, 从下列规则算法计算  $dX$  和  $dY$  移动项:

```
25  即, 让  $dX = A ( X_{cur} - X_{old} )$ 
    让  $dY = A ( Y_{cur} - Y_{old} )$ 
```

最后将结果的分组 (  $\Delta X = dX + eX$ ,  $\Delta Y = dY + eY$  ) 传送到主计算机。本领域的普通技术人员将认识到由上述公式描述线性比例。正如这里所使用的, “比例”意味着所产生的信号是一个单调函数。但是本领域的普通技术人员将会认识到在本发明中可以使用其它单调函数, 包  
30 括但不限于反比例函数和诸如对数或指数函数的非线性比例函数, 而不偏离这里所揭示的原则。



在图 13 中以概图形式示出这一算法的一个硬件实现。尽管所示出的电路 262 只在 X 方向实现，本领域的普通技术人员也可以在 Y 方向实现同样的电路。这些熟练的人员也将立即认识到可用一个等价的软件例程实现图 13 的硬件方案。

5 边缘移动电路 262 包括一个减法器电路 264，其中从  $X_{cur}$  的当前值中减去存储在延迟单元 266 中的  $X_{cur}$  的前一值。将减法器电路 264 的输出提供给乘法器 268，其将该结果乘以增益因子“ A ”。乘法器 268 的输出是项  $dX$ 。

10 也将项  $X_{cur}$  提供给减法器电路 270，其中将  $X_{cur}$  的当前值减去  $X_{center}$  的值。将减法器电路 270 的输出提供给乘法器 272，其将该结果乘以增益因子“ S ”以得到项  $eX$ 。可以理解，利用正交边缘移动将不需要减法器电路 270 和乘法器 272，并且  $eX$  是一个固定值，如果  $X_{cur} > X_{center}$ ，则其符号为正，如果  $X_{cur} < X_{center}$ ，则其符号为负。

15 两输入与门 274 输入项是来自图 12A 的电路的值 FingerOuter 和值 MotionEnable，后者是用于本发明的边缘移动特性的一个来回切换的 on/off 允许信号。如果 FingerOuter 和 MotionEnable 都是真，将开关 276 配置成传送乘法器 272 的输出送到加法器电路 278。如果 FingerOuter 或 MotionEnable 任一是假，则，将开关 276 配置成将值 0 送到加法器电路 278。开关 276 的输出是  $eX$  项。将加法器电路 278 的输出作为  $\Delta X$  20 送到主计算机。用户比如可以通过控制面板控制 MotionEnable 信号。另外也可以下面将要描述的动作单元对之控制。

应该理解，在允许正交边缘移动而不是边缘移动时，两输入 AND (与) 门 274 两个输入项是来自图 12B 的电路的 X FingerOuter 值 (或者是来自图 12B 的电路的 Y 方向的 Y FingerOuter 值) 以及 25 OrthoMotionEnable 值，后者是用于本发明的边缘移动特性的来回切换的 on/off 允许信号。如果 X Finger 和 OrthoMotionEnable 都是真，开关 276 将乘法器 272 的输出传送到加法器电路 278。如果 X FingerOuter 或 OrthoMotionEnable 之一为假，则开关 276 将 0 值送到加法器 278。开关 276 的输出是  $eX$  项。将加法器 278 的输出作为  $\Delta X$  送到主计算机。用户比如通过控制面板控制 OrthoMotionEnable 信号。另外，也可以通过 30 动作单元对它进行控制，下面对之将更全面描述。

在一个替代的形式中，在手指处于“外部”区域时，可以用  $eX$  项

代替  $dX$  项，对于  $dY$  和  $eY$  也同样，而不是在该区域中增加这两项。这造成一个更“纯”的边缘移动，用户更难导向。用户测试已经显示出上述的  $dX+eX$  形式感觉更好而且更易于使用。

5 另外一个功能性的但不太理想的替代方案利用一个稍宽的外部区域。正比于手指进入外部区域的距离而不是距离垫板中心的距离改变滑动速度。从而，在手指进入外部区域时，滑动速度从 0 开始，并且在手指到达垫板的边缘时增加到一个合理的限值。其结果是在边缘移动和正常行为之间有更平滑的过渡。不难以修改上述公式以产生这一可替代的行为。本发明人已经试过这一变例，因为过渡到边缘移动模式似乎过于突然；测试已经表明在一般使用中这种突然性实际上是方便的。这种平滑过渡更难以“感觉”，其结果是更多地而不是较少地突然抓住用户。本领域的普通技术人员将会认识到，利用在这两种所述方法之间的中间解决方案以产生一个较少突然的过渡。

15 对于长距离拖动问题的另一个方案是提供“锁定”拖动或拖动“延长”，下面将对之详细描述。

有利的是，利用一个或多个手指动作使用本发明的边缘移动特性，用户可以在检测器阵列上进行这些手指动作，并由系统识别。其中尤其使人感兴趣的是基本的轻击和拖动动作。轻击动作类似揿按普通的鼠标设备上的鼠标按钮，所有的鼠标器用户都熟悉拖动目标的概念。

20 诸如鼠标器的指示设备一般包括一个或多个鼠标按钮。用户可以指示屏幕上的一个目标，并且揿按一个按钮对之进行选择，或者保持鼠标按钮按下，并移动鼠标，以在屏幕上拖动一个目标。触摸检测器指示设备可以提供各种“动作”，这些动作是特殊的手指动作，它们能模拟鼠标按钮的动作而不需要物理开关。（由于新手或有残疾的用户作出这些动作可能困难，最好也提供物理开关。）在下面的讨论中，单词“手指”可以解释为前面描述的指点笔或其它传导目标。

25 现在回头参见图 1，根据本发明的另一特征，动作单元 20 检查运算单元 16 产生的  $(X, Y, Z)$  数据，以产生一个或多个“虚拟鼠标按钮”信号，连同  $(\Delta X, \Delta Y)$  信号一起送到主计算机。

30 图 14 是图 1 的动作单元 20 的更详细的框图。根据本发明，本发明的动作单元 20 能够支持多种动作。动作单元 20 包括轻击单元 280，之字单元 284，推压单元 284 以及按钮控制单元 286。

动作单元 20 可以支持一些物理开关，在图 14 的示例性例子中，按钮控制单元 286 的两个输入来自物理开关。这种开关可以安装在触摸垫模块上，或者外部提供。可以提供任意数目的开关，或者一个也不提供。输入 A 和 B 具有两个状态，逻辑“0”和逻辑“1”。本领域的普通技术人员将会认识到可以利用特殊的触摸检测器而不是机械开关实现这些开关信号，由类似与单元 44 的充电积分器对这些传感器进行操作，将开关信号送入阈值比较器以形成数字信号。

轻击单元 20，之字单元 282 和推压单元 284 检查 ( X,Y,Z ) 样值序列以寻找各种动作类型。所有这些单元的输出以及开关信号在按钮控制单元 286 中组合，产生实际的按钮按下信号送给主机。在这里所揭示的示意性例子中，触摸垫模拟一个三按钮指示设备 ( 左，中，右 )。可以清除地扩充图 14 的系统，以支持这里所描述之外的其它动作，或者为了简单考虑，支持更少的动作。

按钮控制单元 286 可以使用任意众所周知的方法来组合多个信号。比如，可以在各种信号源之间建立一种优先排序，或者在任一信号源指示该按钮时确立每一按钮输出 ( 左，中，右 ) ( “ 撇按 ” ， “ 按下 ” ，或 “ 按下保持 ” )。组合这些信号的任一方法都是依赖于一种特定的系统配置的常规性设计细节，本领域的普通技术人员可以十分容易地实现。

在当前的较佳实施例中，按钮控制单元 286 将开关和动作映射到通常经常用到的虚拟按钮，给予用户以最大的灵活性。在另一个实施例中，将开关和动作映射到不同的虚拟按钮，从而不需要使用奇怪的动作就可覆盖许多虚拟按钮。或者，让用户选择映射。

在本领域众所周知的是，允许将另外的按钮开关作为特殊的命令处理，比如双击，选择最常使用的菜单项等，而不是作为通常的鼠标按钮。类似地，按钮控制单元 285 或主计算机软件可以将这里所描述的一些动作映射到软件命令，而不是模拟鼠标按钮。这种处理和映射都在本领域的普通技术范围之内。

轻击单元 280 对大部分基本动作进行译码，其中包括轻击，拖动，跳跃和轻击区域。这些动作作为定时图示出在图 15A 至 15E。在图 15A 至 15E 中的每一个中，示出了相对于时间的两个信号的图，一个是模拟 “ Z ” ( 手指压力 ) 信号，另一个是数字 “ 输出 ” ( 虚拟按钮按下 )

信号。利用标号“ t1 ”至“ t21 ”示出了各种与时间有关的变化范围。

基本的“轻击”动作是手指在垫上的快速轻打。这种轻击是短时间的，而且在轻击期间手指在 X 或 Y 方向上没有或几乎没有移动，将这种动作提供给主计算机作为鼠标按钮的短暂揿按。如果模拟一个多按钮鼠标，轻击动作可以模拟“主”鼠标按钮的一次揿按，或者，用户可以使用换档键，控制面板或其它已知的装置选择要模拟的按钮。迅速连续的两次轻击作为按钮的双击提供给主计算机。总之，以该显然的和自然的方式将多次轻击翻译为多次揿按。

由于在手指仍然按下时不可能判断出该手指动作是一次有效的轻击（与光标移动不同），因此在手指抬起之后，当前较佳实施例的装置才报告一次鼠标揿按。因为所定义的轻击是非常短的动作，因此用户通常注意不到这种延迟。

在轻击动作期间可能出现少量的移动，这是因为在压力之下手指尖的自然变形等因素造成的。这就使得由轻击动作产生的虚拟揿按选择屏幕上的错误的项或位置。为避免这种情况，或者是必须抑制这种动作，直到这种动作足够大或者时间足够长以能够取消一次轻击，或者是允许这种动作，但是一旦识别出该轻击动作就回溯取消它。后一方案是较佳的，因为即使是很少量的抑制动作用户也会注意到。

根据本发明的当前较佳实施例，通常将移动事件送往主计算机，并且也记录在寄存器或队列中。当识别出轻击动作时，迅速地重新移动一个相应的负的移动量以“取消”已经报告的移动，并且将光标位置恢复到第一次检测到手指的瞬间光标所在的原始位置。在该动作期间该动作可能已经以数个包序列的形式送给主计算机。为了实现最精确，可以保存这一序列，并且反方向进行重放。然而，如果主计算机的移动处理是线性的，在该动作期间累积移动的总量并且在一个包中发送一个补偿动作将是足够的。由于一个典型的鼠标驱动程序的“累积”特性只操作在高速上，因此这一线性的假设在这一意义上通常是安全的。

由轻击单元 280 所使用的输入是 CurPos，即来自运算单元的当前（X，Y）手指位置；Z，即当前压力值；和 CurTime，即在一些适当的时间单位内的当前时间（比如毫秒或所处理的样值的数目）。

在轻击单元 280 中有 9 个状态变量。如果没有动作在进行，TapState 是 NONE（空）；如果有一个轻击或拖动动作在进行，TapState 是 TAP

( 轻击 ) ; 如果有一个锁定拖动或拖动延长在进行, 则为 LOCKED ( 锁定 ) 。如果 Z 值足够高, 在当前的划动中已被看作满足为一个轻击动作, 则 TapOkay 是 TRUE ( 真 ) 。DownPos 是手指上一次触摸到垫上的 ( X , Y ) 位置。DownTime 是手指上一次触摸的时间。UpPos 和 UpTime 记录下手指上一次从垫上抬起时的位置和时间。TapButton 是 LEFT ( 左 ) , MIDDLE ( 中 ) 或 RIGHT ( 右 ) 之一, 表示当前的动作是在分别模拟左, 中或右虚拟鼠标按钮中的哪一个。如果对于一个双击撤按正在抑制该虚拟按钮, 则 Suppress 是 TRUE ( 真 ) 。最后, Out 代表轻击单元的输出, 是 NONE , LEFT , MIDDLE , 或 RIGHT 之一。

使用几个参数来控制该轻击单元。TapTime 是满足为一个轻击动作的一个动作的最大时间期间。DragTime 是形成一个拖动动作的最初轻击和手指的收回之间的最大的间隔。ExtendTime 是拖动动作将要结束之前在一个拖动延长动作期间手指可以离开触摸垫的最大时间量。HopTime 是先于看作轻击而看作一个跳跃的最大抬起时间。TapRadius 是在一个轻击期间可以出现的最大移动量。DragRadius 是对于一个拖动的最初轻击和手指的收回之间的最大距离。DragExtendRadius 是为满足为一次拖动延长在手指抬起和手指触摸之间的最小距离。HopDistance 是满足为一次跳跃所移动的最小距离。Zthresh 是检测一个手指的最小压力 ( Z ) 。DragExtendSpeed 是在手指抬起期间以满足为一次拖动延长所要求的最小的平滑的速度。这里所要声明的是, 所述的“检测到手指或其它目标的存在” ( 或者是这种效果的其它字面表示 ) 的步骤将假设已经检测到压力大于 Zthresh 。最后, Ztap 是检测到一个轻击手指的最小 Z 。

图 15A 示出了一个基本的轻击动作的定时。首先, 表示了一个成功的轻击, 而后所跟的手指动作由于其时间太长而不被看作一次轻击。在第一个动作中, 手指落下的时间是“ t1 ”, 它小于 TapTime 。同样, ( 在图 15A 上没有示出 ) , 在时间“ t1 ”期间的 ( X , Y ) 移动小于 TapRadius 。最后, 该动作的至少某些部分的 Z 信号超过阈值 Ztap 。从而, 该动作被看作一次轻击。Out 信号 ( 图 15A 的较低的一条线 ) 在时间“ t2 ”期间成为真, 然后成为假。正如后面将要讨论的是, 时间量“ t2 ”等于 DragTime 。在随后所讨论的流程图中描述的设备中, 变

量 TapState 在整个“ t2 ”期间等于 TAP。在当前， TapTime 最好是大约 400 毫秒， TapRadius 最好是大约触摸垫的宽度的 2%， Ztap 略大于 Zthresh，这个值可由用户调节。

5 在图 15A 的右半部，手指按下的时间大于参数 TapTime，在图中示为“ t3”。从而，它将不被看作是一次轻击动作，从这一动作不会产生 Out 信号。

10 在该基本的拖动动作中，用户轻击一次，迅速地返回与垫接触，然后在垫的 XY 平面的一个希望的方向上移动手指。在拖动动作的开始按下这种模拟的鼠标按钮，只在手指再次从垫上抬起时才将之释放。动作的逻辑是将拖动动作的最初轻击组合进所延长的拖动，而不是向主机提供一个另外不同的按钮的撤按动作。

15 在该拖动动作的一种变形中，上述动作开始一个拖动，该拖动甚至持续到当手指抬起时。在手指再次轻击到检测垫上时该拖动动作结束（即释放所模拟的鼠标按钮）。这一特征称为“锁定拖动”。在一个小的垫上，锁定拖动允许拖动的距离大于一个手指移动所能覆盖的距离，但在这一特征被错误地激活时，就很容易使人迷惑。这一锁定拖动在用户界面的研究中成为一种隐藏模式，是一种众所周知的不理想的选项。因此，在该较佳实施例中，它是作为一个选项提供给用户，在缺省情况下是禁止的。

20 在拖动动作的另一个实施例中，上述的拖动动作将持续，即使手指已经抬起，如果手指在一个称为拖动超时（ drag timeout ）的指定的时间期间内回到触摸垫上的话。这一特征称为拖动“延长”。当前这一拖动超时最好为 500 毫秒，但是可以由用户对之优化，本领域的普通技术人员对此已经了解。当然，如果手指从触摸垫上移开，在该拖动超时期间没有回到触摸板上，这一拖动动作将结束。

25 相应地，在拖动延长被允许的情况下，在手指从垫上抬起的时间小于拖动超时时，该拖动动作将持续，在手指离开触摸垫的时间大于该超时时间时，该拖动动作结束。这使得用户可以重复地“划动”或“挥动”以拖动较长的距离。不同于锁定拖动的是，拖动延长对于用户不是表现为一种隐藏模式，因为如果手指没有及时回到触摸垫，在拖动超时期间之后就会产生拖动结束，这一超时时间在人类的感觉中是一个非常短的时间期间。

然而，拖动延长也产生一个问题，因为拖动在整个超时时间期间内持续，甚至在拖动结束时。在用户想立即结束拖动时可能出现这种情况，例如，在使用拖动动作按下滚动条箭头时。通常这些箭头是自动重复的，直到用户释放鼠标按钮。在拖动超时期间拖动动作的持续将使得这一卷动特性卷过所希望停下的位置。

相应地，该拖动动作实际上可以表示两种不同的动作。在光标到处移动而虚拟按钮被持续按下的情况下，表示一种真正的拖动；在光标保持静止而虚拟按钮被持续按下的情况下，表示一次按压。只有对于真正的拖动，该拖动延长才是理想的。有几种方式可以区别真正的拖动和按压。如果在手指抬起之前手指的移动速度大于一个小的阈值，可以识别为一次真正的拖动。如果在整个动作期间手指是静止的，有可能忽略小的不连续的移动，或者在手指抬起的时刻手指是静止的，则可以识别为一次按压。在本发明的拖动延长动作的较佳实施例中，利用在抬起时手指的速度大于一个特定的阈值来区别一次真正的特定和一次按压。将抬起时手指的速度作为移动平均过滤器（*running average filter*）的输出。如果该速度低于该特定的阈值，该拖动结束，而不会被延长。在另一个实施例中，可以用抬起时手指的位置识别一次真正的拖动和一次按压。如果在抬起时手指距离垫的边缘在一个所选择的距离内，可以将之看作一次真正的拖动。

在利用拖动延长的同时，如果在 *ExtendTime* 的期间用户开始一个新的无关联的手指动作，有可能出现第二个问题。正如以上所讨论的，在启动拖动延长时，即使手指已经从触摸垫上抬起，如果手指在拖动超时期间内返回到触摸垫上，该拖动将持续。在用户抬起手指时可能用户实际上想结束这次拖动，而将手指放回到触摸垫上可能要开始一个新的动作。确定拖动动作是在继续还是要被结束和开始一个新的手指动作的方法是比較抬起的手指位置和触摸下的手指位置。通常，一个所延长的拖动的随后的动作不是开始于前一动作的结束处。从而，如果手指落下的位置与抬起的位置在一个指定的距离之内（在指定的拖动超时之内），该拖动延长特性允许拖动继续，否则，该拖动立即结束。然而，本领域的普通技术人员可以理解的是，可以不将触摸时手指的位置与抬起时手指位置相比较，而且进一步不必立即结束该拖动，而实现这一拖动延长特性，虽然这不是较佳的。

先前描述的“边缘移动特性”作为实现长距离拖动的一种替代方案。

以以下方式实现拖动动作。在识别到一次轻击时，如前面所描述的，该虚拟鼠标按钮被按下。然而，直到手指从垫上移开了一个足够长的时间以使之不被认为是一个拖动动作时才释放该虚拟按钮。这一时间量  $DragTime$  应被选择得足够长以允许进行一个适当的拖动动作，但也要足够短，以允许由轻击动作产生的撤按仍然合理地短。这里较佳的是，所使用的这一时间量大约 200 毫秒。

正如图 15B 所示，拖动动作以上述的轻击开始，该轻击的时间期间“ $t_4$ ”小于  $TapTime$ 。响应这一轻击，Out 信号变成高。手指离开垫的时间持续“ $t_5$ ”，其小于  $DragTime$ ，然后手指返回到垫上，并保持“ $t_6$ ”，“ $t_6$ ”大于  $TapTime$ 。从而这一动作就作为一次拖动。Out 信号保持为高，并持续到在时刻“ $t_7$ ”手指最后释放为止。在图 15B 的实现中，手指的撤去和虚拟鼠标按钮的释放之间的时间“ $t_7$ ”为 0；在其它类似的实现中，它也可以为非 0，但是很小，比如等于  $DragTime$ 。注意，在从“ $t_5$ ”到“ $t_7$ ”的整个时间间隔  $TapState$  将等于  $TAP$ 。

对于  $DragTime$  的定时，有许多替代方案可以考虑。图 15A 所示出的间隔“ $t_2$ ”也是间隔“ $t_6$ ”的上限，它正好等于参数  $DragTime$ 。在另一个替代方案中，相对于  $DownTime$  而不是  $UpTime$  测量  $DragTime$ ，也就是说必须将间隔“ $t_1$ ”和“ $t_2$ ”（分别是“ $t_5$ ”和“ $t_6$ ”）加到  $DragTime$ 。这一方法的一个后果就是在基本的轻击动作中，一个更长的更慢的轻击产生一个更短的虚拟按钮撤按。相对于图 15A - B 的方案，这种矛盾使得用户对这种方法不太满意。

在另一个替代方案中，使得  $DragTime$  正比于间隔“ $t_1$ ”（也是“ $t_5$ ”）的长度，使得一次短的轻击产生一个短的虚拟按钮撤按，而且一次较长的轻击（最大至极限  $TapTime$ ）产生一个较长的撤按。这一替代使得用户对于模拟的按钮撤按有了更多的控制，但是，作为该轻击动作的一个不重要的特征，它使得该行为依赖于用户的感知。

有数种方法使得  $DragTime$  的大小正比于间隔“ $t_1$ ”的长度。在一种情况中，该虚拟按钮撤按的长度或  $DragTime$  是用户轻击的长度的一个直接函数。正如在上一段所描述的，一次短的轻击产生一次短的虚拟按钮撤按，较长的轻击产生较长的撤按。这一方法对于那些通常轻击得



较慢以及需要一个较长的时间期间( 较长的 DragTime ) 来将手指放回以开始一个拖动动作的新用户来说提供了一种好处。不幸的是, 较长的 DragTime 也产生较长的虚拟按钮撤按( OUT 信号 ), 这可能具有不太理想的负效应, 包括无意识地使滚动条自动重复卷动或“一阵一阵卷动”。

5 在新手和熟练人员的轻击有差别时, 一种较佳的方法是识别不同长度的轻击, 但是对于不同长度的轻击使该虚拟按钮撤按或 OUT 信号长度相同。然而, 在识别一个新手的轻击时, OUT 信号的定时将被延迟, 从而新用户有一个较长的 DragTime 来开始一个拖动动作。本领域的普通技术人员所理解的是, 用于区别新手和熟练用户的轻击的长度在用户研究之后将被优化。应该认识到, 可以有其它方法来区别新手和熟练用户。比如, 一个新手轻击的压力经常大于一个熟练人员的轻击的压力。另外, 利用轻击长度的历史也是有益的, 比如, 利用几个先前轻击的平均长度。本领域的普通技术人员也将会认识到, 可以由用户在控制面板进行新手和熟练用户的判断。

10 现在参见图 15C, 它示出了变量 DragTime 的较佳实施例, 该变量是轻击长度的一个函数。一方面, 正如当前所希望的, 一个熟练用户的轻击具有间隔“ t1 ”的期间, 小于 200 毫秒。一旦手指离开垫, 在间隔“ t2 ” 200 毫秒的虚拟按钮撤按或 OUT 信号就产生, 从而为熟练用户提供最可能快的响应。为了开始一个拖动动作, 在 200 毫秒虚拟按钮撤按的时间到时之前手指要返回到触摸垫, 因此, 在较佳实施例中, 将变量 DragTime 选为 200 毫秒, 从而将熟练用户的快速的手指动作错误地解释成一个拖动动作的机会减少了。

20 另一方面, 将新手轻击的时间是 200 毫秒和 500 毫秒之间的一个间隔“ t1 ”( 在较佳实施例中, 超过 500 毫秒的动作将不被看作轻击 )。在 300 毫秒的延迟“ t2a ”之后, 在间隔“ t2b ”中产生 200 毫秒的虚拟按钮撤按或 OUT 信号, 因此, 用户将有 500 毫秒的较长的 DragTime 以开始一个拖动动作。本领域的普通技术人员将会认识到, 可以用几种方法选择延迟的长度, 其中包括作为轻击长度的一个函数。类似地, 其它与时间有关的动作识别的参数, 诸如 HopTime 和 ExtendTime, 在涉  
30 及到新手轻击时, 可以进行调整。如果在延迟的撤按开始之前( 例如在“ t2a ”间隔期间 ) 手指回到垫上开始一个拖动动作, 则在手指一落下

时必须立即产生虚拟按钮撤按。否则，如果这一新的手指动作也被证实是一次轻击，可以将得到的双击的第一次撤按包含到“ t2a ”间隔中。

图 15D 示出锁定的拖动动作。该拖动动作以一个标准的拖动动作开始，包括间隔“ t8 ”至“ t10 ”。然而，在手指抬起时， Out 信号保持高。（在流程图中，在这一时刻 TapState 将从 TAP 变为 LOCKED ）。该图示出了长度大于 TapTime 的第二个拖动动作（示为“ t11 ”）（它并没有结束该锁定拖动），后面跟有长度“ t12 ”小于 TapTime 的另一动作。由于这一最后的动作满足为一次轻击，在时刻“ t13 ”结束该锁定拖动。在流程图中，在这一时刻， TapState 变回 TAP；然后处理正规的轻击，在时间“ t13 ”将 Out 保持为高，“ t13 ”象通常一样等于 DragTime。一个合理的替换的实现可以在一个不同的间隔“ t13 ”之后，比如 0 结束这一拖动。

图 15E 示出拖动延长动作。该拖动延长开始于一个标准的拖动，包括间隔“ t14 ”至“ t16 ”。在间隔“ t17 ”抬起手指，但是由于手指离开触摸垫的时间小于拖动超时参数 ExtendTime，OUT 信号保持高。同样，（图 5E 中未示出），“ t17 ”期间（ X ， Y ）移动大于 DragExtendRadius，并且在间隔“ t17 ”的开始在从垫上抬起的时候所平滑的平均手指速度大于 DragExtendSpeed。该图示出手指抬起了一个第二间隔“ t18 ”。由于在间隔“ t18 ”期间手指从触摸垫抬起的时间长度大于 ExtendTime，在手指从垫上抬起之后 OUT 信号在等于 ExtendTime 的时间期间为低。对于新手或熟练用户最好可以调节 ExtendTime，正如前述对于 DragTime 所做的。

图 15F 示出双轻击动作。该双轻击的进行与拖动动作的区别不明显。然而，第二个动作“ t21 ”小于 TapTime，因此作为第二次轻击而不是一次拖动。正规轻击的处理使得 Out 在长度为 DragTime 的另一间隔“ t23 ”保持为高；然而，在识别到该轻击之后，在流程图中所示出的特殊双轻击处理抑制虚拟鼠标按钮一个短的“ t22 ”时间。从而，主计算机察觉到两个不同的撤按而不是看作没有这一特殊处理的一个长的连在一起撤按。

可以使用其它动作模拟一个多按钮鼠标。在这样一个方法中，该基本动作被一个“跳跃”动作扩大了，其中手指从垫上的某个位置的休止处抬起，并轻击在距离其休止处一个相当长的地方。如果该距离足够大

( HopDistance , 一般是触摸垫的宽度的一个小部分; 当前最好是大约 25 % ) , 并且抬起和随后的轻击的时间间隔小于一个适当的阈值 ( HopTime , 一般是一秒的一个小部分; 当前最好大约是 0.5 秒 ) , 然后在一个不同的鼠标按钮上模拟由该轻击所开始的撤按或拖动动作。这一不同的按钮可以是一个固定的“辅助”按钮, 或者它可以由用户通过控制面板或其它装置选择, 或者手指跳跃的方向的一个函数( 比如向右或向左 )。根据本发明的当前较佳实施例, 该跳跃动作可作为一个选项使用, 其缺省是禁止( off )。

注意, 尽管一些用户在跳跃动作中喜欢用第二个手指轻击, 但这一动作从不涉及在任一时刻在垫上有多于一个手指。这里描述了一个类似的动作“之字”( zigzag ) , 它确实涉及同时使用两个以上的手指。

图 15G 示出一个“跳跃”动作。这一动作以手指已经在垫上开始。然后将手指抬起“ t24 ”间隔, 该时间小于 HopTime ; 然后手指落下, 用于正规的轻击“ t25 ”。同样在图中未示出, 在间隔“ t24 ”期间, 该手指必须从其前一位置移动至少某一距离 HopDistance 。当这一情况方式时, 将这一动作处理为“跳跃”而不是通常的轻击, 在右按钮 Out ( R ) 出现该虚拟按钮按压, 而不是在通常的左按钮 Out ( L ) 上。很容易看到, 轻击“ t25 ”是如何后跟进一步的手指动作以形成一次拖动或在右按钮上的双轻击的。

另一多按钮动作使用“轻击区”, 垫的表面被分成两个或多个区。在一个给定的区中启动的轻击或拖动模拟对应于该区的按钮上的一个事件。即使在拖动期间该手指在区之间移动, 也在对应于启动该拖动动作的最初轻击的区域的按钮上模拟整个拖动。

图 16A 和 16B 显示了两个轻击区的形状。在图 16A 中, 该垫被分成三个垂直条 288 , 290 和 292 , 分别对应于左, 中和右鼠标键。在图 16B 中, 该垫被分成一个模拟左鼠标键的主区域 294 和模拟右鼠标键的小角落区域 296 。如果一个按钮在一般的应用中比另一按钮频繁地使用, 图 16B 的实现就更容易理解了。

最好是使得这些区域对应于该垫的表面上清楚标记的区。对于本领域的普通技术人员来说, 使用其它的区域形状, 比如多角落或水平条, 同样是显而易见的。

在轻击区和边缘移动特性之间有一个相互影响, 这一点需要加以考

虑。特别是对于图 16B 的角落区域 296，轻击区鼓励用户轻击在垫的边缘附近。如果在轻击和拖动期间边缘移动是活跃的，或者一直是活跃的，则边缘移动将趋向于干扰角落轻击的适当行为。为了防止这种情况，可以稍微修改图 13 的边缘移动允许逻辑。在一个给定的动作中，  
5 如果在该动作期间手指已经位于内部区域至少一次，则边缘移动才工作。从而，如果手指触摸在外部区域，边缘移动将不激活，直到手指离开垫的边缘并且然后返回。

上述的所有动作都是基本的轻击和拖动动作的变形。在这里所描述的系统中，轻击单元 280 识别所有这些动作。很容易以流程图的形式将  
10 轻击单元 280 的工作描述为一个算法。从本公开中，本领域的普通技术人员将会认识到这里所描述的轻击单元可以以比如软件程序，硬件状态机或其它已知的和显然的等价物实现。所有这些实现都落在本发明的范围之内。

图 17A 至 17F 包括了轻击单元 280 工作的流程图。轻击单元 280 实  
15 现这里所描述的轻击，拖动，锁定的拖动，拖动延长，角落轻击和跳跃动作。在这里所描述的动作识别操作中，使用角落轻击来模拟右虚拟鼠标键。使用向左和向右的跳跃来模拟中和右虚拟鼠标键。简单的轻击模拟左（主）虚拟鼠标键。

当每一个（ $X$ ， $Y$ ， $Z$ ）样值从图 1 的运算单元 16 到来时，在步  
20 骤 300 开始处理。在本发明的当前较佳实施例中，这种数据每秒钟到达 40 次。每次一个样值到达时，图 17A 至 17F 的算法将从开始（步骤 300）运行到结束（步骤 392）。

通过比较  $Z$ （压力）与  $Z_{thres}$  来判断手指是出现（“落下”）  
25 还是没出现（“抬起”），步骤 302 判断手指是抬起还是落下。可以使用两个阈值，而不是一个简单的阈值比较来提供本领域中众所周知的迟滞（hysteresis）。在图 17A 中未示出迟滞，但类似的迟滞稍后将示出在“推压”动作的图 20 中。

在步骤 304，已知手指按下。检查前一  $Z$ ，看看该手指是先前按下的还是刚刚触摸到垫上。

30 在步骤 306，已经检测到手指按下转换。这可以指示一个拖动动作或在一个拖动延长中的一个连续的行的开始。对于一个拖动或拖动延长动作，分别检查在 DragTime 和 ExtendTime 期间手指位置从从触摸垫上

的前一手指位置的改变。

在图 15B 示出的拖动动作中，在时间“ t5 ”期间检查手指还没有移动一个很长的距离是有利的，该间隔“ t5 ”是开始轻击和手指返回到垫上的间隔。如果所计算的在“ t5 ”期间的距离表示手指已返回到垫上的一个不同的位置，则该拖动动作可能不是所希望的。

在图 15E 中所示出的拖动延长动作中，有必要检查在间隔“ t17 ”在一个延长的拖动动作的连续行之间手指已经移动一个很长的距离。如果该距离不是足够长，该拖动动作结束。

由于在拖动动作期间 TapState 是 TAP ，同时在拖动延长期间 TapState 是 LOCKED ，步骤 306 确定该 TapState 。在步骤 306 ，如果 TapState 是 TAP ，则步骤 308 计算在当前位置 CurPos ( 所过滤的和平滑的 X 和 Y 位置数据 ) 和所保留的前一轻击的位置 DownPos 之间的距离。如果该距离大于某一阈值 DragRadius ，则过程前进到步骤 310 。否则，进行到步骤 312 。阈值 DragRadius 应该是垫宽的小部分，最好大于在基本的轻击检测中所使用的 TapRadius ( 或更大 ) 。

在步骤 314 ，判断 DragLock 是否允许。如果 DragLock 被允许，该执行前进到步骤 312 。否则，该执行前进到步骤 316 。

如果步骤 306 的 TapState 是 LOCKED ，并且 DragLock 没有被允许，则一个拖动延长一定在继续。步骤 316 计算 CurPos 和所保留的前一动作的结束位置 UpPos 之间的距离。如果该距离大于某个阈值 DragExtRadius ，则该执行继续到步骤 312 ，否则它前进到步骤 310 。阈值 DragExtRadius 应该是垫宽的一小部分，正如用户测试所确定的。( 某些用户可能喜欢 DragExtRadius 为 0 ，因此，有效地禁止步骤 316 。 )

本领域的普通技术人员将认识到，几个可能的距离测量方法适合于步骤 308 和 314 。一个实欧几立德距离测量方法是合理的，但难以计算；一个较简单的测量是取在 X 和 Y 的距离的绝对值的和或者最大值。该和或者最大值将围绕最初的轻击产生一个“拖动区域”，它是分别是菱形或正方形，而不是欧几立德距离测量所产生的圆形区域。试验揭示出用户不能感觉到这些区域形状之间的差别，因此，哪一测量最容易计算，哪一测量就是最好的。同样，手指和触摸垫的几何形状可能使得显著的移动常常位于一个方向上，比如 X 方向，在这种情况下，最好简单

地使用 X 坐标的绝对差。

在一个较佳实施例中，用户使用一个控制面板或其它装置能够改变动作识别的级别。如果用户选择允许轻击而不允许拖动，可以将步骤 308 编程为直接前进到步骤 310，从而所有的轻击不会被看作拖动。

5 在步骤 310，拖动动作已经不被考虑。TapState 从 TAP 变为 NONE；其效果是简单的轻击动作，后跟光标移动，没有虚拟按钮被按下。

步骤 312 记录下手指触摸下的位置和时间。

10 步骤 318 将 TapOkay 初始化为 FALSE（假）。它也清除抑制标志，该标志用于延迟由于“新手”的轻击产生的虚拟按钮撤按。如果手指返回接触触摸垫，步骤 318 提前结束该延迟。如果该新的手指动作是一个双轻击动作的第二次轻击，步骤 318 保证由第一次轻击产生的虚拟撤按完全不是偶然被抑制的。

15 对手指按下的所有样值执行步骤 320，该步骤将 Z 与 Ztap 阈值进行比较；如果 Z 大于 Ztap 阈值，步骤 322 将 TapOkay 置为 TRUE（真）。因此，当手指抬起时，如果在作为一个轻击动作的备选的该简短动作期间 Z 曾经超过轻击阈值，则 TapOkay 将为 TRUE。

现在参见图 17B，在步骤 324，已知手指离开垫。检查前一 Z 看看该手指是否先前已抬起或正在从垫上抬起。

20 在步骤 326，已经检测到手指抬起转换。对大多数最近的动作进行各种测试（在手指按下期间），看看它是否为一次轻击。要作为一次轻击，该动作必须时间上很短（CurTime 减去 DownTime 必须小于 TapTime），很少或没有移动（从 CurPos 到 DownPos 的距离必须小于 TapRadius），并且有足够的峰值手指压力（TapOkay 必须为 TRUE）。

25 在步骤 328，为了“取消”已经报告的移动和恢复到第一次检测到手指出现时刻的最初光标位置，通过快速地从寄存器或队列向主计算机发送一个对应的负的移动量，从而反向取消已经出现的任一手指移动。如果在该动作期间已经以几个包的序列的形式向主计算机报告该移动，可以保存该序列，并且反向重放。如果该主计算机的移动处理是线性的，在该动作期间累积移动的总量和在一个包中发送一个补偿的移动是足够的。由于一个普通的鼠标驱动程序的“加速”特性只在高速度上激活，在这一意义上，这一线性假设通常是安全的。

30

步骤 330 基于当前的 TapState 取几个动作中的一个。首先，如果 TapState 是 NONE（没有动作在进行），执行简单地前进到步骤 32。在步骤 332，计算该轻击动作的时间长度，即 CurTime 减去 DownTime，以区别短的熟练用户的轻击和长的新手的轻击。对于熟练用户，执行简单地前进到图 17C 的步骤 338。对于新手的轻击，执行前进到步骤 334，对于当前的动作，它使用较长的 DragTime 值。这些步骤简单地将该轻击期间与一个固定的阈值比较，以在两个固定的 DragTime 值之间选择，或者它们可以使用该轻击期间来平滑地调节 DragTime。

步骤 334 也将 Suppress 标志置为 TRUE，使得该虚拟按钮信号为低保持一个短的期间。在较佳实施例中，将这一期间选择为新手和熟练用户的 DragTime 值的差值，使得结果的虚拟撤按的非抑制部分在所有情况下具有同样的期间，如图 15C 所示。

其次，如果 TapState 是 TAP（一个最近的轻击仍在进行），则已经检测到一个双轻击。步骤 334 将 Suppress 标志置为 TRUE，使得对于一个样值该虚拟按钮信号为低。这对应于图 15F 的时间“t22”。在一个替换的方案中，可以将表示该虚拟按钮的释放的一个或多个另外的包插入到正规的包流中，而不是使用这里所示的 Suppress 标志。在另一个替换的方案中，为了延迟新手的撤按和处理双轻击的双重目的，可以使用不同的 Suppress 标志；比如，用于延迟新手的撤按 300 毫秒的抑制可能是较佳的，但是对于双轻击，在反映虚拟按钮的抑制的至少一个包被发送给主计算机之后它可能足以结束该抑制。

最后，如果 TapState 是 LOCKED，这是结束一个锁定的拖动的轻击。步骤 336 将 TapState 设置回 TAP，然后直接跳到图 17E 的步骤 370，跳过判断模拟三个鼠标键中的哪一个的步骤。从而，锁定拖动改变回在同一虚拟鼠标键上的轻击。在正常的短期间之后（图 15D 的“t13”），释放该虚拟按钮。

重要的是，在 LOCKED 情况下，忽略了按钮选择（图 17c）。如果，比如由在垫上的角落上的轻击启动一个右按钮的锁定拖动，则应有可能通过轻击垫上的任何地方终止该拖动，而不是仅仅轻击在该角落上。同样重要的是，在双轻击情况中，包括按钮选择。否则，将不可能比如通过交替在垫上的两个不同的位置轻击在迅速的交替中执行左和右按钮的撤按。

在另一个可替换的实施例中,如果 TapState 是 LOCKED,将 TapState 置为 NONE,并且接着执行步骤 386。这使得图 15D 的“ t13 ”等于 0。由于在拖动延长期间 TapState 可能也是 LOCKED,这里所示出的实现也允许用户轻击以缩短拖动延长时间 DragExtTime。在另一个可替换的  
5 实施例中,在拖动延长期间可以忽略轻击。然而,如果 DragExtTime 很短,这一意义就不重要了。

现在参见图 17C,步骤 338 检查当前的轻击是否满足为一个“跳跃”动作。这一检查包括几个测试。首先,该跳跃动作必须由用户启动。其次,在当前轻击和手指上一次在垫上之间的十分短的时间内  
10 ( DownTime 减去 UpTime 必须小于 HopTime )手指必须已经抬起。最后,这一轻击的位置必须十分显著地远离前一位置(从 DownPos 到 UpPos 之间的距离必须大于 HopDistance)。再一次,可以使用多种距离测量方法。图 17A 至 17F 的操作支持向左和向右的跳跃;一个合理的距离测量是 DownPos 和 UpPos 之间在 X 坐标上的绝对差值。

15 在一种很容易看成是最等价的变形中,在步骤 338 中利用 CurTime 和 CurPos 代替 DownTime 和 DownPos。

如果该轻击满足为一次跳跃,执行前进到步骤 340。由于这一示例的系统支持两种不同的跳跃动作,检查跳跃的方向以确定动作的类型。如果 DownPos (或 CurPos) 的 X 坐标小于 UpPos 的 X 坐标,出现的是向左的跳跃(假设 X 向右是增加的)。如果 DownPos (或 CurPos) 的 X 坐标大于 UpPos 的 X 坐标,出现的是向右的跳跃。注意,由于步骤 336 的检查,此刻 DownPos 将显著地到 UpPos 的左侧或显著地到其  
20 右侧。

在步骤 342,一个向左的跳跃使得 TapButton 被置为符号 MIDDLE,从而,该轻击动作将产生一个虚拟的鼠标中键的揞按。  
25

在步骤 244,向右的跳跃使得 TapButton 被置为 RIGHT 启动一个虚拟的鼠标右键的揞按。

如果没有检测到跳跃,执行步骤 346。对其它所支持的替换动作-角落轻击进行检查。角落轻击是出现在一个小的角落区域的轻击,如图  
30 16B 所示。如果用户已经启动角落轻击,则出现角落轻击; DownPos (或 CurPos) 的 X 坐标大于某一坐标 CornerX; 并且 Y 坐标大于某一坐标 CornerY。CornerX 和 CornerY 示出在图 16B。



对本领域的普通技术人员来说，很显然通过检查轻击位置的 X 和 Y 坐标以一种完全类似的方式对其它轻击区域，诸如图 16A 所示出的，或多角落区域进行解码。

5 在当前的较佳实施例中，作为模拟替换的按钮撤按的一种机制，用户可以选择跳跃动作，角落轻击或都不选择。没有任何困难阻止一种实现同时提供跳跃和角落轻击，除了给用户带来的益处还不如给用户带来的困难多。

在步骤 348，没有检测到角落轻击，从而将 TapButton 置为 LEFT，模拟左虚拟鼠标键的撤按。

10 在步骤 350，检测到一个角落轻击，从而将 TapButton 置为 RJGHT，模拟右虚拟鼠标键的撤按。

步骤 352 记录下当前位置作为新的 UpPos，该抬起位置用于以后的跳跃译码。总的来说，每次看到手指从垫上抬起，就更新 UpPos。然而，对这一规则，有两个例外。首先，如果该手指抬起是一个跳跃的轻击动作的部分，不更新 UpPos。可以在图 17C 中的流程图的左分支看到这一点。需要这一例外支持比如虚拟右按钮的双击。抬起该手指，基本上向右移动，然后两次轻击。这两个轻击大体上出现在同一位置。如果由第一次轻击更新 UpPos，则第二次的轻击被译码为一个左按钮的撤按。

其次，在图 17A 至 17F 的流程图中，对于结束一个锁定拖动的轻击并不更新 UpPos。用户测试表明用户所察觉到的上一次抬起的位置通常是在该锁定拖动期间的最后一次抬起位置，同时该终止的轻击是一个下意识的动作，其位置不被意识为相关的。因此，对于一个锁定拖动的终止轻击更有意识地忽略 UpPos 的更新。

25 在任一轻击，角落轻击或跳跃动作之后，步骤 354 将 TapState 设置为 TAP，从而记录一个动作正在进行中。

现在参见图 17D，在手指从垫上抬起，其动作不被视为一个轻击的时候，执行步骤 356。这一步骤检查 TapState 是否为 TAP；如果是，手指必须已经从一次拖动动作的长时间动作中抬起，即在图 15B 的时间“t7”。根据用户的偏好，或者由手指的抬起终止该拖动动作，或者将之锁定而成为一个锁定拖动。

步骤 358 检查锁定拖动是否已被用户启动。对于拖动总是锁定或总是不锁定的系统来说，可以在设计时进行这种判断，或者基于运行时的

选项，诸如控制面板，进行判断。

如果该锁定拖动还没有被启动，则步骤 360 检查拖动延长是否已被启动。对于拖动总是被延长或总是不被延长的系统来说，可以在设计时进行这种判断，或者基于运行时的选项，诸如控制面板，进行这种判断。

5 在步骤 362，如果启动拖动延长，则检查抬起时手指的速度是否高于 DragExtSpeed。这允许在上述的真正的拖动和按压之间产生区别。

在步骤 364，将拖动的 TAPSTATE 改变为 LOCKED。

在步骤 366，手指的抬起终止拖动。

10 在手指一被抬起并且不被当作一次轻击时就执行步骤 368，就如上所述将 UpPos 更新为当前位置。

现在参见图 17E，在手指一从垫上抬起时就执行步骤 370。更新变量 UpTime 以记录手指从垫上抬起的时刻。

步骤 372 确定每一样值的 TapState，其中手指仍然离开垫。

15 如果 TapState 是 TAP，则步骤 374 将 CurTime 减去 UpTime 的差与 DragTime 相比较，看看在一次轻击之后手指是否已经离开垫很长时间，该轻击是一次拖动的开始。应该理解，如果变量 DragTime 在使用中，用于比较的 DragTime 将是已经进行一次短的熟练的轻击或者已经进行了一次长的新手的轻击的函数。如果超过该时间限制，并且 TapState 等于 TAP，则该执行前进到步骤 376。否则，执行前进到步骤 378。

20 如果 TapState 是 LOCKED，步骤 380 确定是否已经启动 DragLock 模式。如果还没有启动 DragLock，则执行前进到步骤 382。如果启动 DragLock，则执行前进到步骤 378，该拖动继续。

25 步骤 382 确定手指离开触摸垫的时间是否已经超过 ExtendTime。如果不是，则拖动继续，并且执行前进到步骤 378。否则，执行前进到步骤 376，其中 TapState 成为 NONE，因为手指已经离开触摸垫很长时间，而不继续该拖动。

步骤 378 检查手指已经离开垫的时间是否足够长以结束由步骤 334 开始的抑制期间。如果是，则执行前进到步骤 384，其中将 Suppress 标志置为 False（假）。

30 步骤 376 将 TapState 从 TAP 改变为 NONE，结束该轻击，并且从而防止该轻击延长为一次拖动，或者防止现有的拖动被进一步延长。

现在参见图 17F，所有路径会聚到步骤 386，其对每一样值执行，

而不管手指的状态。这一步骤开始一系列的检查，以确定对这一样值轻击单元 280 的输出。首先，在步骤 386，如果 Suppress 标志是 TRUE，抑制虚拟按钮，从而在步骤 388 将输出设置为 NONE。

5 如果 Suppress 标志为 FALSE，并且没有按钮抑制，在步骤 390 检查 TapState。如果 TapState 是 TAP 或 LOCKED，则在步骤 392 将 TapButton 表示的按钮输出。

如果 TapState 是 NONE，没有轻击，拖动或跳跃动作在继续，在这种情况下，步骤 394 将输出置为 NONE。

10 该处理在步骤 396（结束）结束。在下一（X,Y,Z）样值从运算单元到来时，轻击单元将从步骤 300 开始工作。

图 13 的边缘移动特性在拖动动作期间是最有用的。从而，最好是允许移动单元 18 的 MotionEnable 输入由动作单元 20 的状态产生。特别是，通过 MotionEnable=( TapState=TAP ) 或 ( TapState=LOCKED ) 获得图 13 的输入进 AND 门 268 的“ MotionEnable ”信号。

15 图 14 的“之字单元” 282 译码两手指动作，其中一个手指一直放置在垫上，同时另一手指轻击在该主手指的一侧。对于由基本装置产生的（X，Y，Z）信息，这一动作将有效地增加 Z 值，同时快速地移动 X 和/或 Y 值一个显著的距离。（在两个手指在垫上时，所报告的位置在两个手指的中间。）如果检测到这种改变，而后是迅速地回  
20 到最初的 X，Y 和 Z 值，则识别到一个第二手指的轻击。

由于只有到该第二个手指抬起时才能可靠地识别第二个手指的轻击，首先到一侧然后又回来的突然的光标移动不可避免地送给主计算机。名称“之字”指的就是这些特性的光标移动。可以利用类似于在轻击单元 280 中使用的反向移动机制，以确保该虚拟鼠标揞按发生在最初  
25 的非之字的位置。其唯一困难是，在这种情况下，所涉及的移动可以是足够大以触发主计算机的加速特性，这意味着必须存储该反向移动并且一个包一个包地重放，或者之字单元和主计算机软件必须相互协作以确保实际上光标确实返回到所希望的位置。

30 有可能只使用来自标准的运算单元 16 的（X，Y，Z）信息识别第二手指的轻击，正如这里所描述的。然而，很显然可以修改运算单元 16 以产生附加的信息，诸如传感器线轮廓的宽度或形状以有助于这一动作的精确识别。

图 18A 至 18C 是描述之字单元 282 的算法的流程图。类似于轻击单元 280 的情况，最好以流程图描述之字单元 282。然而，硬件状态机是一种已知的等价实现，并且也是之字单元 282 的一种合理的实现。不同于图 17A 至 17F 的轻击单元 280 的流程图，之字单元 282 流程图每一动作执行一次。在检测到手指的出现时 ( $Z > Z_{thresh}$ )，在步骤 386 开始执行操作。如果手指在执行结束之前离开该垫，之字单元 282 放弃其计算，并且在步骤 386 开始对下一动作的处理。

图 18A 至 18C 示出了另外的特性，其中向左的之字模拟左按钮的揿按，而向右的之字模拟右按钮的揿按。

10 之字单元 282 需要与轻击单元 280 同样的位置  $Z$  和时间输入。它也需要一个速度的量度  $S$ ，它是作为在任一时刻从前一手指位置到当前手指位置的距离而计算的。如果如前所述，对运算单元 16 的正常的 ( $X$ ,  $Y$ ) 输出进行任何过滤或平滑，最好从未过滤的 ( $X$ ,  $Y$ ) 值计算速度  $S$ 。

15 之字单元 282 的状态变量包括： $ZigZ$  和  $ZigZ'$ ，它们记录下  $Z$  的两个最近的值； $ZigPos$  和  $ZigPos'$ ，它们记录下两个最近的位置； $ZigTime$ ，它记录下检测到第二个手指出现的时间； $ZigLeft$  和  $ZigRight$ ，如果已经检测到向左或向右的之字，它们相应为 TRUE；和  $Out$ ，它表示之字单元 282 的输出，是 LEFT, RIGHT 或 NONE 之一。

20 之字单元 282 使用几个参数： $ZigDistance$ ，它是手指位置的移动被作为这一动作的最小距离； $ZigMaxTime$ ，它是将第二个手指作为一个动作的最大时间量； $Szig$ ，它是开始动作的检测所需要的瞬时手指的速度，它可以根据样值速率，检测器的大小，和在充电积分器中的模拟过滤的量以试验方式确定； $ZigRadius$  和  $ZigLimit$ ，它们分别指示在第二个手指抬起后该位置和  $Z$  值如何靠近它们必须返回到的最初的之字之前的值。在当前的较佳实施例， $ZigRadius$  可相比于  $TapRadius$ ，并且  $ZigLimit$  大约是  $Z_{thresh}$  的 30%。

现在参见图 18A，在检测到手指的出现时，在步骤 400 开始执行。

30 在步骤 402，之字单元 282 等待来自运算单元 16 的三个 ( $X$ ,  $Y$ ,  $Z$ ) 样值。在当前的较佳实施例中，这些样值以每秒 40 个的速率到达。这一最初的延迟用于防止在动作的开始时  $Z$  的波动被误认为第二个手指。

在步骤 404，初始化 ZigZ 和 ZigZ' 为标为 NONE 的保留值。

在步骤 406，之字单元 282 等待下一 ( X, Y, Z ) 样值的到来。

5 步骤 408 检查一个“弯折” ( zig ) 的开始，它是之字动作的第一个一半，其中明显地手指延长并且跳到一侧。将当前样值的速度 S 与阈值 Szig 相比较。如果 Z 较大，并且 ZigZ 包含有效的数据 ( 不是所保存的 NONE 值 )，该执行前进到图 18B 以进行进一步的验证。

10 在步骤 410，没有看到最初的“弯折”，因此更新 ZigPos' 以反映最近的手指位置，并且更新 ZigPos 以反映第二个最近的手指位置。如果对图 1 和 8 的运算单元 16 的输出进行平滑或者过滤，则不同于较早描述的 S 的计算，应该从所过滤的或平滑的位置数据更新 ZigPos。换句话说，应该用所处理的位置数据进行更新，该位置数据被用于更新在主计算机上的光标位置。

15 在步骤 412，类似地更新 ZigZ' 和 ZigZ 以反映两个最近的 Z 值。在典型的使用模式中，第二个手指的轻击典型地发生于左或右，即 X 不同，但不需要 Y 的不同。因此，在第二个手指出现时，X 分母 ( 图 8 的减法器 168 的输出 ) 趋向于提高整两倍，然而 Y 分母 ( 减法器 172 的输出 ) 可能或不可能相应地提高。这取决于充电积分器 44 的线性特征。因此，为了之字单元 282 的目的，最好直接使用 X 分母输出作为 Z，而不是使用通常从图 8 的方框 178 得到的所组合的和所处理的值。

20 在步骤 412 之后，执行返回到步骤 406，等待下一样值。

现在参见图 18B，步骤 414 记录下最初的“弯折”被检测到的时间。

步骤 416 初始化 ZigLeft 和 ZigRight 标志。如果看到手指从开始位置显著地向左或向右移动，该标志将为 TRUE。

25 在第二个手指落在垫上时，( X, Y, Z ) 值通常取两个或三个样值以会聚到它们的新值，该新值反映了两个手指的出现。步骤 418 等待一个或两个更多的样值的到来，在此之后 ( X, Y, Z ) 应该被确定。选择一个，两个或更多个样值取决于下列因素：基本的样值速率和在该设备的模拟输入部分出现的过滤量。

30 在步骤 418 之后，CurPos 反映之字表示的手指位置，ZigPos 反映在速度超过 Szig 阈值之前两个样值的位置。两个样值的历史是重要的，因为由于第二个手指的靠近，在手指触摸和产生超过 Szig 的较大移动之前可能已经出现一个小的移动量。在步骤 418 之后，ZigPos 包含在第二

个手指有可能产生效果之前的一个时刻所保存的当前位置。类似地，ZigZ 包含在第二个手指到来之前的 Z 值。

步骤 420 检查 Z 是否已经基本上增加到超过保存的 Z 值 ZigZ。在当前的较佳实施例中，与一个阈值相比，Z 比 ZigZ 大 30%。如果 Z 太小，该“弯折”就不被看作为弯折，执行返回到步骤 404。

步骤 422 检查当前位置是否在本位位置 ZigPos 的左边很远。由于之字单元 282 正在寻找位置的一个急剧的“不自然”的变化，在对位置数据进行通常的过滤或平滑之前，步骤 422 最好使用直接来自运算单元 16 的除法器 174 和/或 176 的位置数据。这里将这一数据称为 RawPos 以区别于所过滤的和平滑的值 CurPos。然而，如果需要的话，可以使用值 CurPos，其结果不是最优。

在这一实现中，步骤 422 比较 RawPos 的 X 坐标与 ZigPos 减去 ZigDistance 的 X 坐标。在一个手指按下另一个手指自然轻击时可以根据它们在垫上所观察到的两个手指之间的间隔用试验方式选择参数 ZigDistance。

如果检测到一个适当的向左的弯折，步骤 424 将 ZigLeft 置为 TRUE。

类似地，步骤 426 检查当前位置是否在本位位置的右边很远。如果是，步骤 428 将 ZigRight 置为 TRUE。

然后步骤 439 等待下一 (X, Y, Z) 样值的到来。

步骤 432 通过比较 Z 与第二个“弯折” (zag) 阈值 (它略小于步骤 420 的“弯折”阈值) 检查第二个手指是否已经从垫上抬起。(在当前的系统中，这一阈值大约比 ZigZ 大 20%)。为了提供简单的迟滞，将第二个“弯折”阈值设置得比第一个“弯折”阈值小。

如果第二个手指还没有抬起，执行返回到步骤 422 继续等待。如果第二个手指已经抬起，执行前进到图 18C 的步骤 434。

现在参见图 18C，步骤 434 等待一个或两个 (X, Y, Z) 数据的样值，以在第二个手指抬起时进行稳定；这一步类似于步骤 418。

步骤 436 进行完成之字动作的最后检查。到目前为止，已经看到伴随着 Z 的增加，然后是 Z 的突然减小的一个突然的移动。另外步骤 436 检查该位置已经返回到其之字之前的值 (从 RawPos 到 ZigPos 的距离小于 ZigRadius)；类似地 Z 值返回到正常 (Z 减去 ZigZ 的绝对值小于

ZigLimit ) ; 并且 ZigLeft 或 ZigRight 之一为 TRUE , 但不能同时为 TRUE 。另外,第二个手指轻击必须是短时间的,即 CurTime 减去 ZigTime 小于 ZigMaxTime.

5 如果该移动并不作为一个之字,执行返回到步骤 404 , 等待检测一个之字动作。如果该移动为一个之字,步骤 438 提供反向的移动以将光标恢复到对应于 ZigPos 的准确位置,如果需要的话。这一步骤类似于图 17B 的步骤 328 。

10 在步骤 440 , 已经检测一个完整的之字。如果 ZigLeft 是 TRUE , 该移动是一个向左的之字。否则, ZigRight 必须为 TRUE , 并且该移动是一个向右的之字。相应地,或者步骤 442 模拟一个向左之字的左按钮的按压,或者步骤 444 模拟一个向右之字的一个右按钮的按压。

步骤 446 暂停某一时间量。比如,这一步骤可能等待一个或几个更多的样值的到来,或它可能等待要送到主计算机的一个或几个数据包。(通常在样值和数据包之间有一个一一对应)。

15 最后,通过将 Out 置为 NONE , 步骤 448 结束所模拟的按钮按压。在这一例子中,之字动作只模拟撤按,不模拟拖动。该之字动作不能以通常的单指轻击的同样的方式简单地延伸到拖动,否则就隐含着用两个手指在垫上笨拙地保持着进行整个拖动移动。一种替代是模拟一个锁定按钮,在本领域中经常这样使用轨迹球按钮,其中连续的之字交替地按压和释放虚拟按钮。另一种替代是用之字动作按压虚拟按钮和在主手指也从垫上移开时释放该虚拟按钮。

在步骤 448 , 执行返回到标志 04 等待检测进一步的之字动作。

25 在特殊的应用中有用的另一动作是“推压”动作,它简单地将 Z(压力)信息与第二个 Z 阈值 ZpushDown (它比基本的手指检测阈值相当高)进行比较,并且 Z 一旦超过这一阈值,就模拟一个鼠标按钮的动作。这一“推压”动作类似于基于笔的指点设备通常操作的方式;然而,用它作为主撤按或拖动动作太不精确而且使得手指十分疲劳。该“推压”动作在特殊的情况下特别有用,比如手写绘图程序。

30 图 19 是描述一个“推压”动作的定时图。为了执行这一动作,首先将手指足够靠近,使得光标移动而不产生虚拟按钮的按压。接下来,增加手指的压力超过阈值 ZpushDown , 使得按下虚拟按钮。然后,将压力减小到低于阈值 ZpushUp , 使得虚拟按钮被释放。如果 ZpushUp

略低于  $Z_{pushDown}$ ，如果手指的压力在“推压”阈值的周围稍稍变化，其得到的迟滞能防止虚拟按钮的不希望有的振荡。

在可能是较佳的一种变形中， $Z_{pushUp}$  被置为  $Z_{thresh}$ ，使得一旦开始一次推压，为了释放所模拟的按钮，手指必须完全从垫上抬起。其它用户可能喜欢使  $Z_{pushUp}$  离  $Z_{pushDown}$  比离  $Z_{thresh}$  更近，产生一个更灵敏的感觉。

图 14 的推压单元 284 识别该推压动作。图 20 的流程图示出这一动作的实现。用于识别这一动作的一个等价的硬件电路的对应的图示是十分显然的。

10 每次在一个新的  $(X, Y, Z)$  样值从运算单元 16 来到时，步骤 450 开始执行。注意，推压单元 284 只检查每一样值的  $Z$  值。

步骤 452 检查一个“推压”动作是否已经正在进行。如果没有“推压”动作正在进行，步骤 454 开始执行这一步骤检查一个“推压”是否应当开始。首先“推压”动作必须由用户启动。其次，当前的  $Z$  值必须  
15 大于阈值  $Z_{pushDown}$ 。

如果  $Z$  足够开始一个推压动作，步骤 456 将  $Out$  置为 LEFT，以指示现在按下左按钮。

步骤 458 检查当前推压动作是否应当结束。这一检查涉及简单地比较  $Z$  与  $Z_{pushUp}$ 。如果  $Z$  小于  $Z_{pushUp}$ ，在步骤 460 终止该推压动作。

20 执行在步骤 462 结束。如果既不执行步骤 456 也不执行步骤 460， $Out$  保持不变，从而提供了上述的迟滞。在开始时，状态变量  $Out$  应该被初始化为 NONE。

本领域的普通技术人员会注意到，轻击单元 280 适于与提供  $(X, Y)$  和手指压力信息的触摸垫一起使用，推压单元 284 适于与产生  $Z$ （压力）信息的触摸垫一起使用。只有之字单元 282 取决于这里所揭示的特定触摸垫的特殊特性，即两个手指可靠地报告一个平均的手指位置这一情况。

可以使用并非完全是动作处理的一部分的两个以上的算法解决手指轻击在垫上时出现的一些小问题。特别是，有时手指位置在手指抬起时在一个方向上急剧地切断。这是由于在这一动作期间手指的自然滑动，在手指保持在一个小角度时这一情况更严重。一个“反向的移动”算法可以处理这一问题的一部分，但是如果表现的手指位置跳跃很大，  
30



以致 TapRadius 测试失败，反向算法无能为力。

如果看到 Z 在当前和前一样值之间迅速地改变(即如果当前的和前一样 Z 值的绝对差值小于实验所确定的阈值)，则可以增加运算单元 16 的输出的 (X, Y) 过滤的时间常量。通常，老的过滤器值和新的系数用大体上相同的权重进行平均，产生一个新的过滤器值。如果 Z 迅速地改变，相对于新的系数增加老的滤波器的值的权重(比如一个数量级)。这就使得在大的 Z 改变的瞬间发生的任何移动被严重地衰减了。

在 Z 减小到低于手指落下阈值 Zthresh 之前的最后一个样值经常出现由于手指抬起而产生的虚假移动。解决这一虚假的手指抬起移动的另一方案是“抬起跳跃机制”，它试图抑制这一最后的虚假移动事件。图 21 示出了执行这一抬起跳跃抑制功能的示意性电路。

图 21 的电路执行抬起跳跃抑制。它检查由图 8 的除法器 174 和 176 到来的 (X, Y) 位置样值的序列，以产生一个速度 S，该速度 S 经过进一步的处理以获得一个移动抑制信号。正如先前所描述的，最好在任一平滑或过滤阶段之前，在计算速度 S 时直接使用系数值。

参见图 21，将 X 坐标存储在延迟单元 470 中。减法器 472 计算当前的 X 值和存储在延迟单元 470 中的前一样值之间的差的绝对值。同样地，延迟单元 474 和减法器 476 计算 Y 的绝对变化。加法器计算这些绝对差值的和产生速度 S，即当前的和前一样值的距离。正如前面所讨论的，很显然，可以使用其它距离量度用于这一计算。注意，除了图 21 的电路之外，之字单元 282 也使用前面所揭示的速度值。

延迟单元 480 和 482 记录下 S 的前一个值和第二个先前值，分别称为 S' 和 S''。除法器计算 S 的一半，记为 S/2。抬起跳跃抑制单元寻找 S, S', S'' 和 S/2 之间的特性关系，以识别出一个虚假的抬起跳跃事件。本领域的实践认识到直到一个给定的手指动作的第四个样值出现时 S'' 才有效；从而对于每一个动作的前三个样值将抬起跳跃抑制单元禁止。该抬起跳跃抑制单元也利用参数 LiftJump，即一个由实验所确定的速度阈值，其受到检测器垫的取样速率和灵敏度的影响。

比较器 486 检查速度 S 是否大于阈值 LiftJump。比较器 488 检查前一速度 S' 是否小于 LiftJump，比较器 490 检查 S' 是否小于 S/2。类似地，比较器 492 检查第二个前一速度 S'' 是否小于 LiftJump，并且比较器 494 检查 S'' 是否小于 S/2。如果所有五个条件都满足，与门 496 输出一个“抑

制移动“信号，对于这一样值它抑制移动单元 18 的动作。在抑制移动单元 18 时，对于当前的样值，不产生输出(  $\Delta X$  ,  $\Delta Y$  )，并且延迟单元 260 不被同步。

5 在手指抬起之前在上一虚假的移动样值期间通常产生由抬起跳跃抑制单元检测到的轮廓。由于在最近下一样值上 Z 落在 Zthresh 之下，当前的样值从不产生任何移动事件送给主计算机。该算法在设计时保证在一行中不会抑制多于一个样值。从而，如果该算法“猜测错误”，并且 Z 并没有落在 Zthresh 之下，所跳过的的手指动作将被代入下一样值产生的(  $\Delta X$  ,  $\Delta Y$  ) 包，观察到光标的移动只有很小的迟疑。

10 本发明的触摸检测器系统的所提高的灵敏度允许更轻的输入手指触摸，使得人们更易使用。所提高的灵敏度也使得使用其他输入设备更方便，例如指点笔等。另外这一灵敏度也可以相对于较厚的保护层，或不同的材料进行折衷，它们使得制造成本更低。

15 更好的噪声排除造成使用中的更好的灵活性，并且降低了对假噪声问题的灵敏度。使用两种技术，它们产生最佳的噪声排除的好处。

由于在本发明中采用了驱动和检测技术，数据获取速度比以前的技术提高了 30 倍。这带来了几个明显的辅助效应。首先，对于同一级别的信号处理，电路在大部分时间可以是断开的，降低了设计中的模拟部分的大约 30 倍的电力消耗。其次，由于可以利用更多的数据，可以进行诸如滤波和动作识别的更多的信号处理。

20 本发明所使用的检测器电子电路非常健壮，并且校正过程和系统中的误差。它将来自检测器的电容信息进行处理，产生数字信息提供给一个外部设备，比如一个微处理器。

25 由于本发明的独特的物理特性，其具有人们比较感兴趣的人机工程应用，这些应用在以前是不可能的。当前的鼠标器或轨迹球在便携式计算机上使用时在物理上不太方便。而本发明提供了一种非常方便的和易于使用的光标位置方案，它可以代替那些设备。

30 在鼠标器型的应用中，本发明的检测器可以位于通常的位置，比如在便携式计算机的“空格键”的下面。当处于这一位置时，用户的大拇指可以用作检测器上的位置指示器，以控制光标在计算机屏幕上的位置。用户的手指可以不需要离开键盘就可以移动光标。从人机工程学上，这类似于带有轨迹球的 MacIntosh Power Book 的概念，然而本发明

在尺寸上比轨迹球更有利。可以将这一基本思想扩展成在“空格键”下放置两个检测器用于更有特色的控制。

带有光标反馈的计算机显示器是通用应用领域的一个很小的例子，这里显示器可以是光的或 LED 的，一个 LCD 显示器或一个 CRT。例子还包括在实验室设备上的触摸控制，其中这些设备使用旋钮/按钮/触摸屏的组合。由于这一界面的连接的能力，这些输入中的一个或多个可以结合这里参照本发明所描述的一个输入中去。

消费者电子设备（立体声音响，图形均衡器，混合器）应用常常利用用于滑动电压计的前面板平面区域，因为需要可变的控制。本发明可以在一个小的触摸垫板位置提供这种控制。随着电子家庭系统的普及，越来越需要更紧密和更强大的人机界面。本发明的检测器技术允许一种非常紧密的控制面板。如果使用这种检测器技术，可以形成符合人机工程学上的手持 TV/VCR/立体声控制，并且允许更强大的特性。

本发明的检测器可以适合任意表面，并且可以制成检测多个触摸点，使得有可能用于一个更强大的游戏杆。本发明的检测器技术的独特的压力检测能力是这一应用的关键。计算机游戏，遥控（业余爱好者的电子飞机）和机器加工工具的控制是应用的几个例子，这些应用从本发明的检测器技术中获益。

音乐键盘（合成器，电子钢琴）要求对速度敏感的键，它们可以由这种检测器的压力检测能力提供。也具有变调控制和其他滑动开关，它们都可由这一技术替代。一种甚至更独特的应用包括一种乐器，它能作为在一种联合的 3 维界面上手和手指的位置和压力的函数产生音调。

本发明的检测器技术最好能够检测压在其上面的传导材料。通过在本发明的检测器的表面加上一层覆盖了一层传导材料的紧密绝缘层也可以间接地检测来自所操作的任何目标的压力，而不管其导电性。

由于从这一检测器中可获得大量的信息，这一检测器可以很好地用作虚拟现实机器的一种输入设备。很容易就可想象出这样一种结构，它允许在三维空间中的位置监视，并且对于动作的响应程度（压力）。

尽管已经显示和描述了这一发明的多个实施例和应用，很显然，对于本领域的普通技术人员，他们可作出比上述更多的修改而不偏离这里所揭示的创造性的概念。因此，本发明只受所附的权利要求的限制。

# 说明书附图

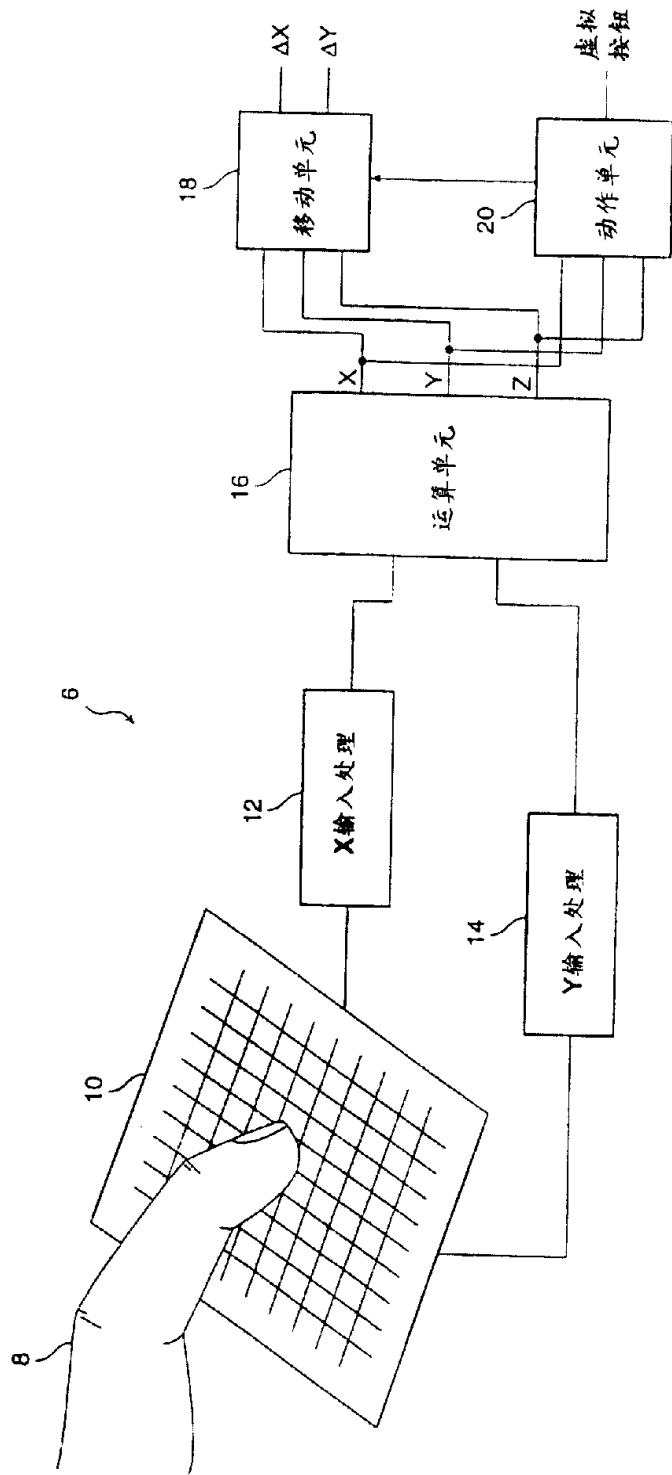


图 1

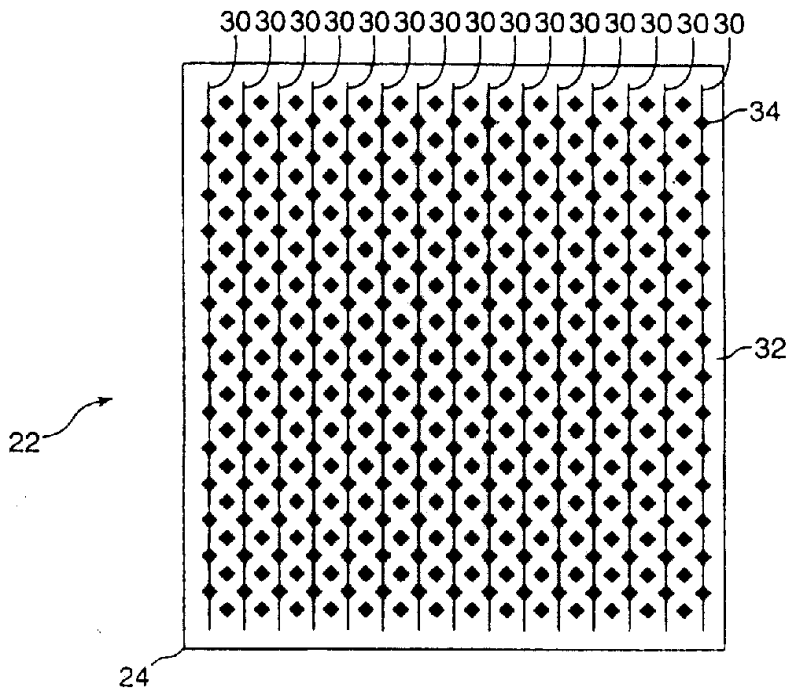


图 2 A

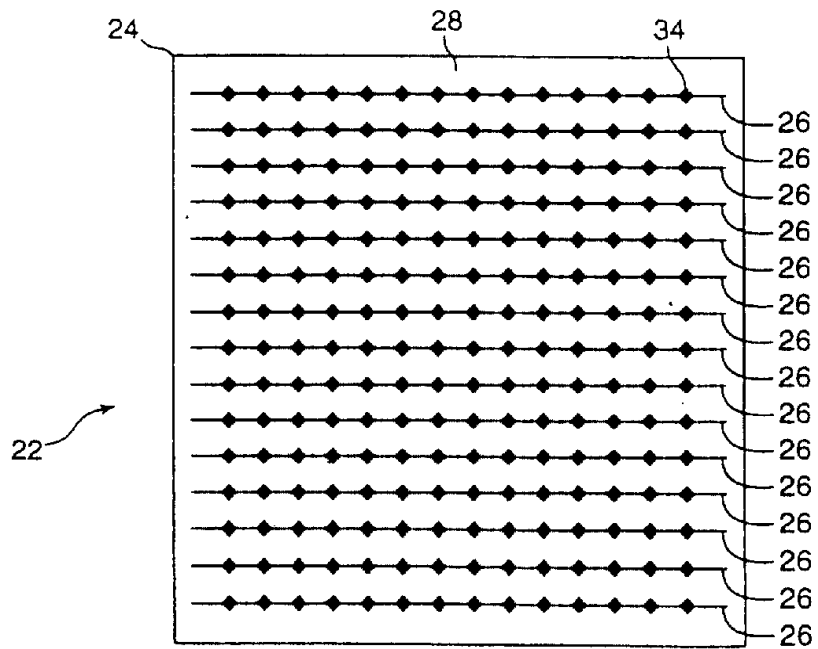


图 2 B

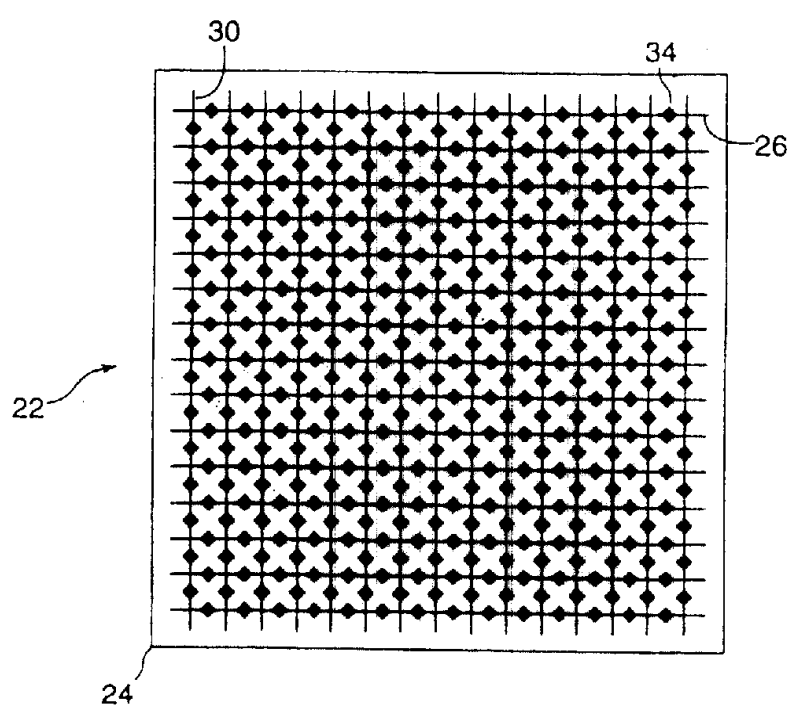


图 2 C

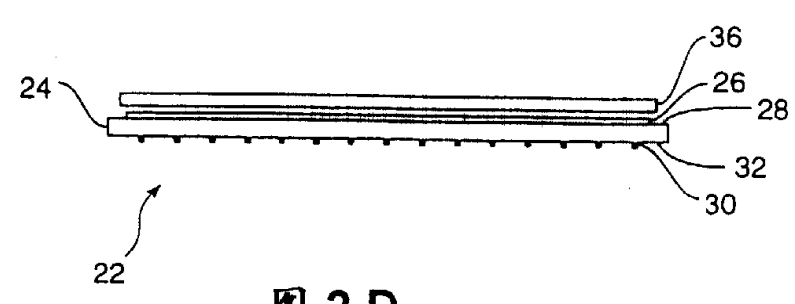


图 2 D

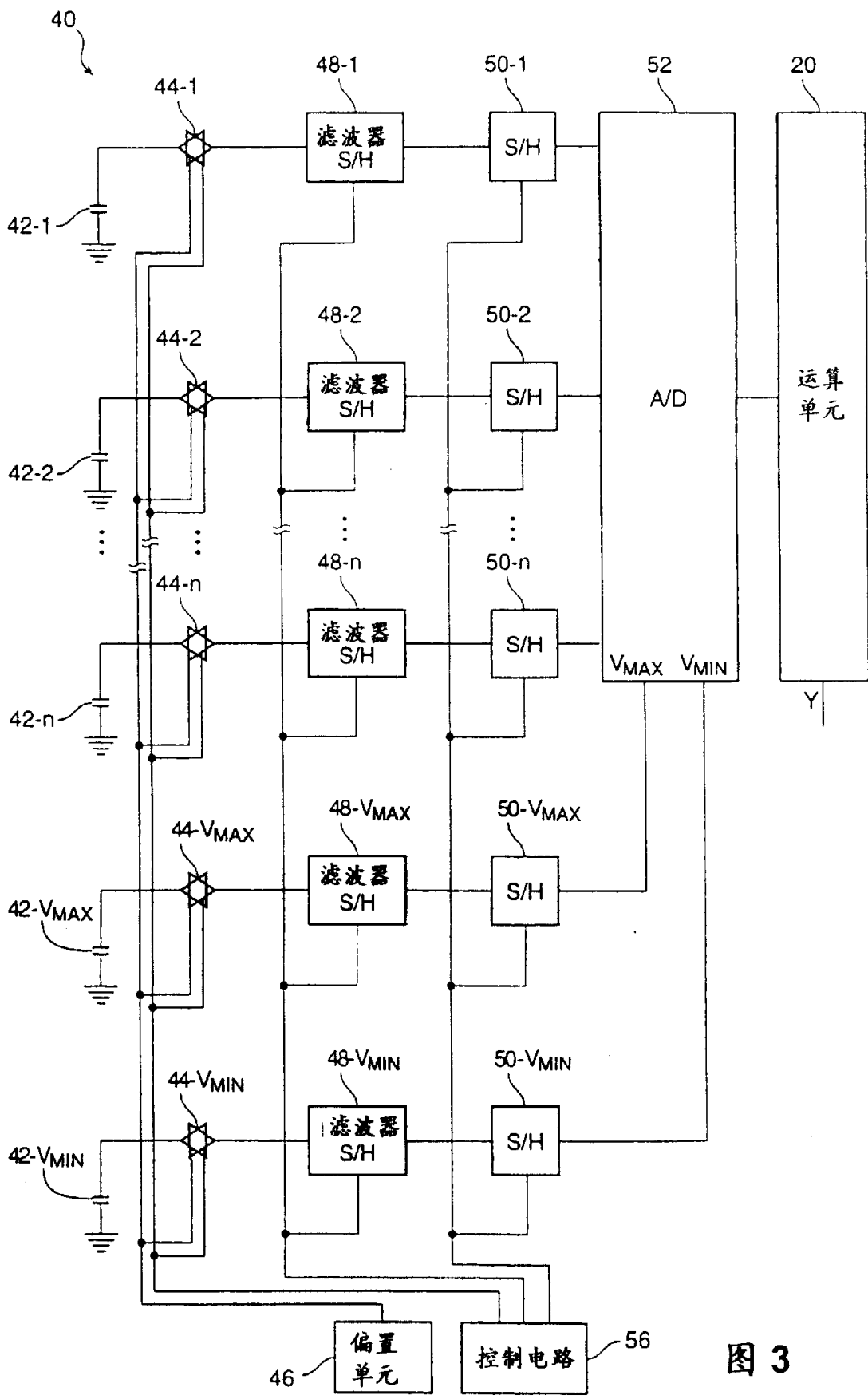


图 3

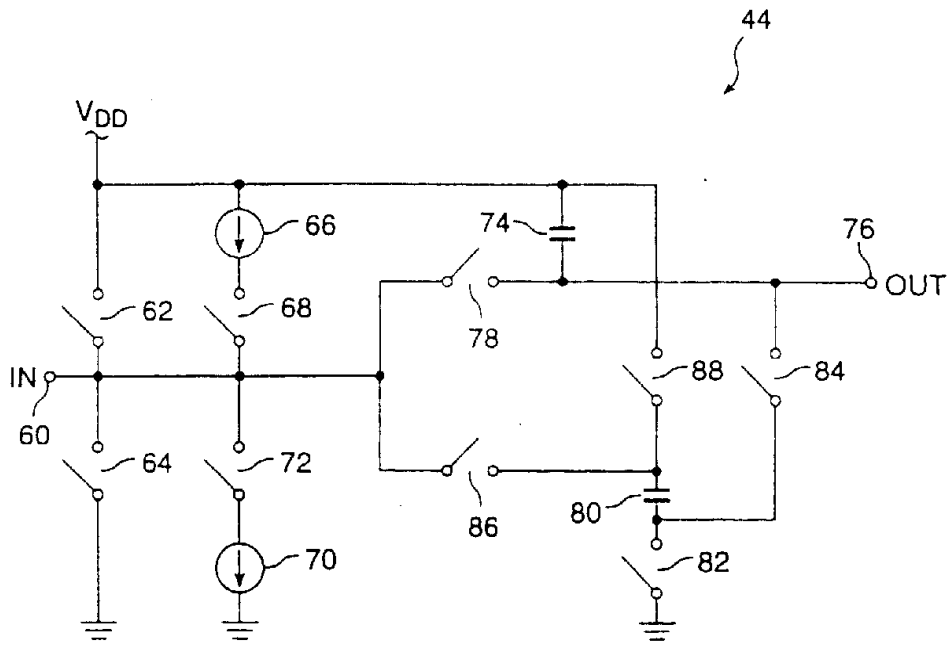


图 4 A

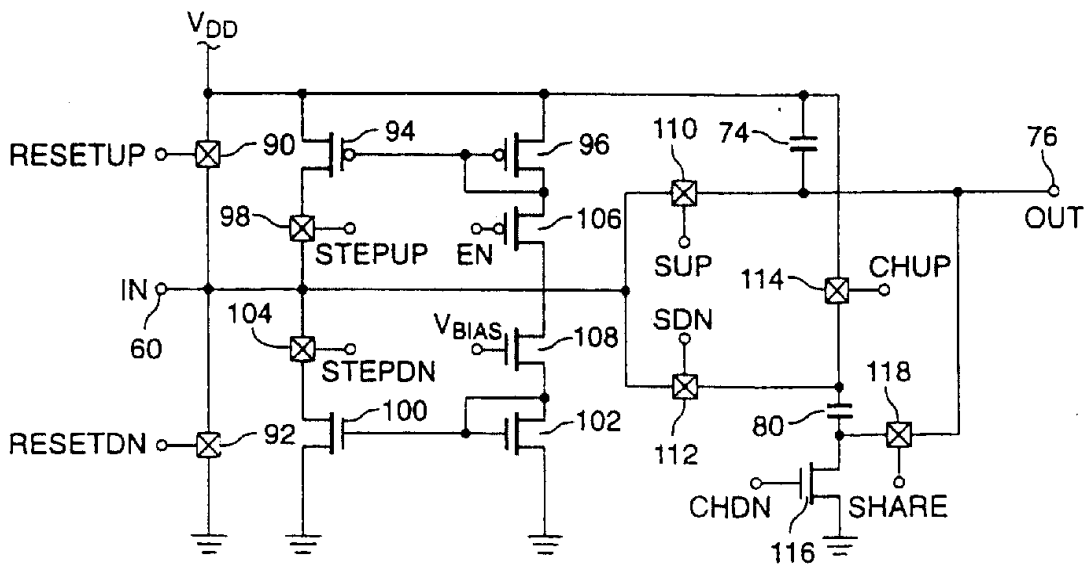


图 4 B



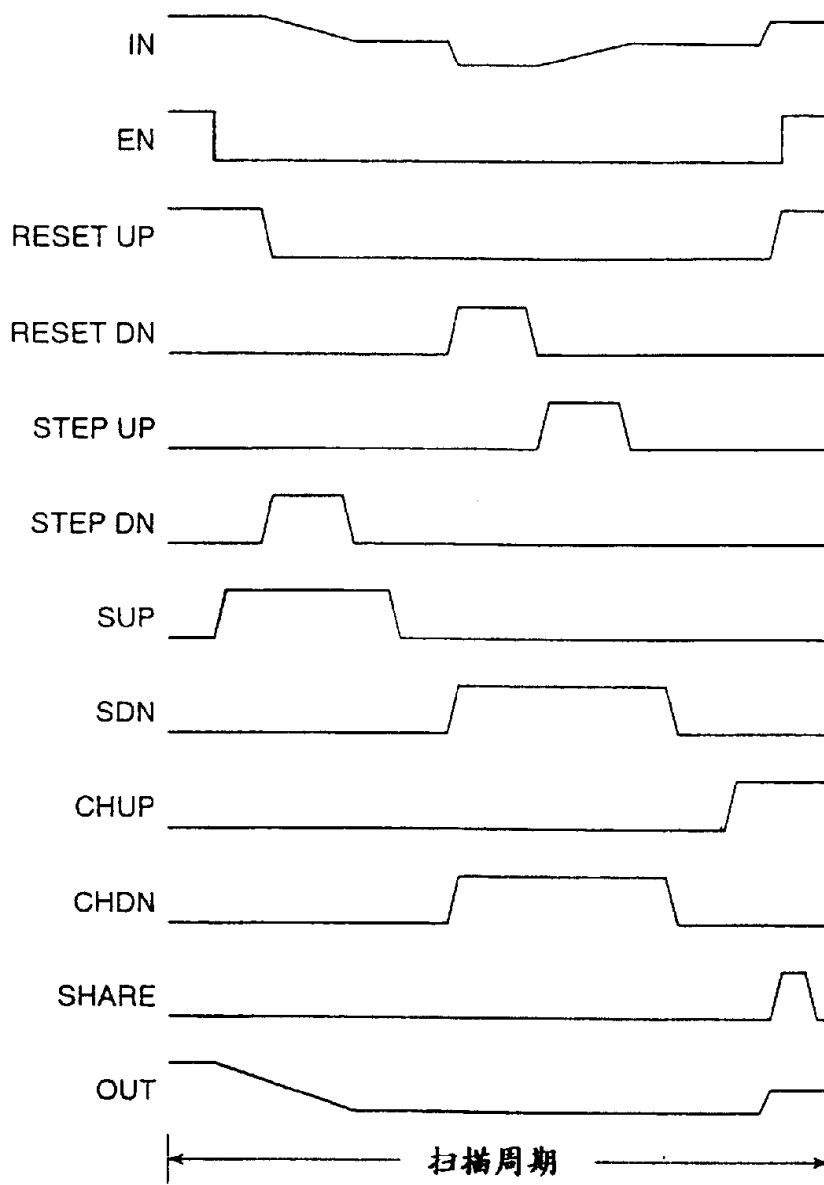


图 5

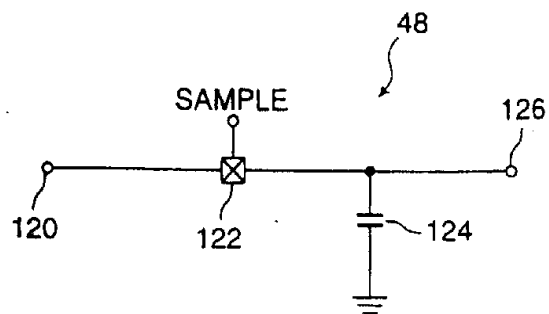


图 6

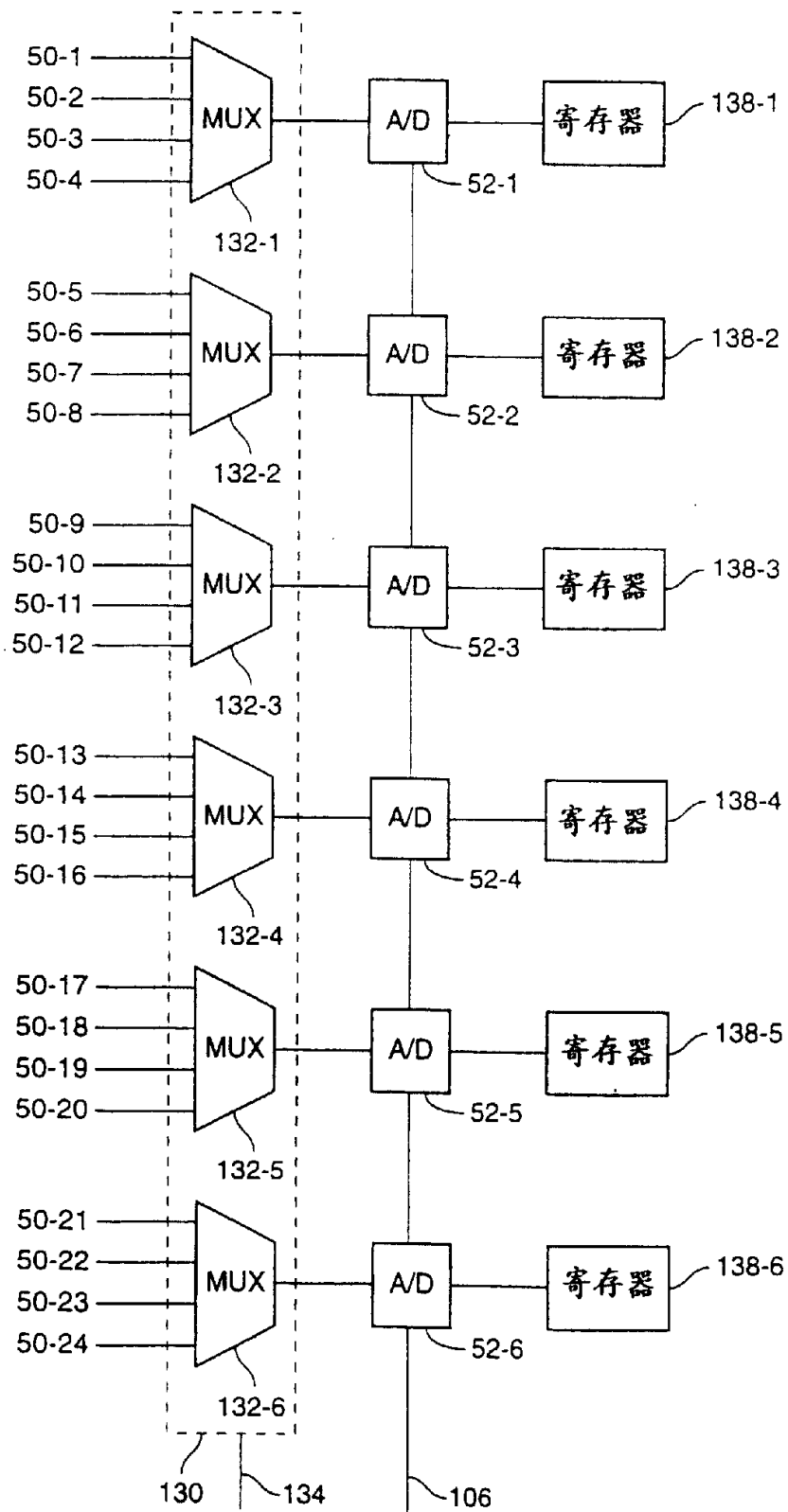


图 7

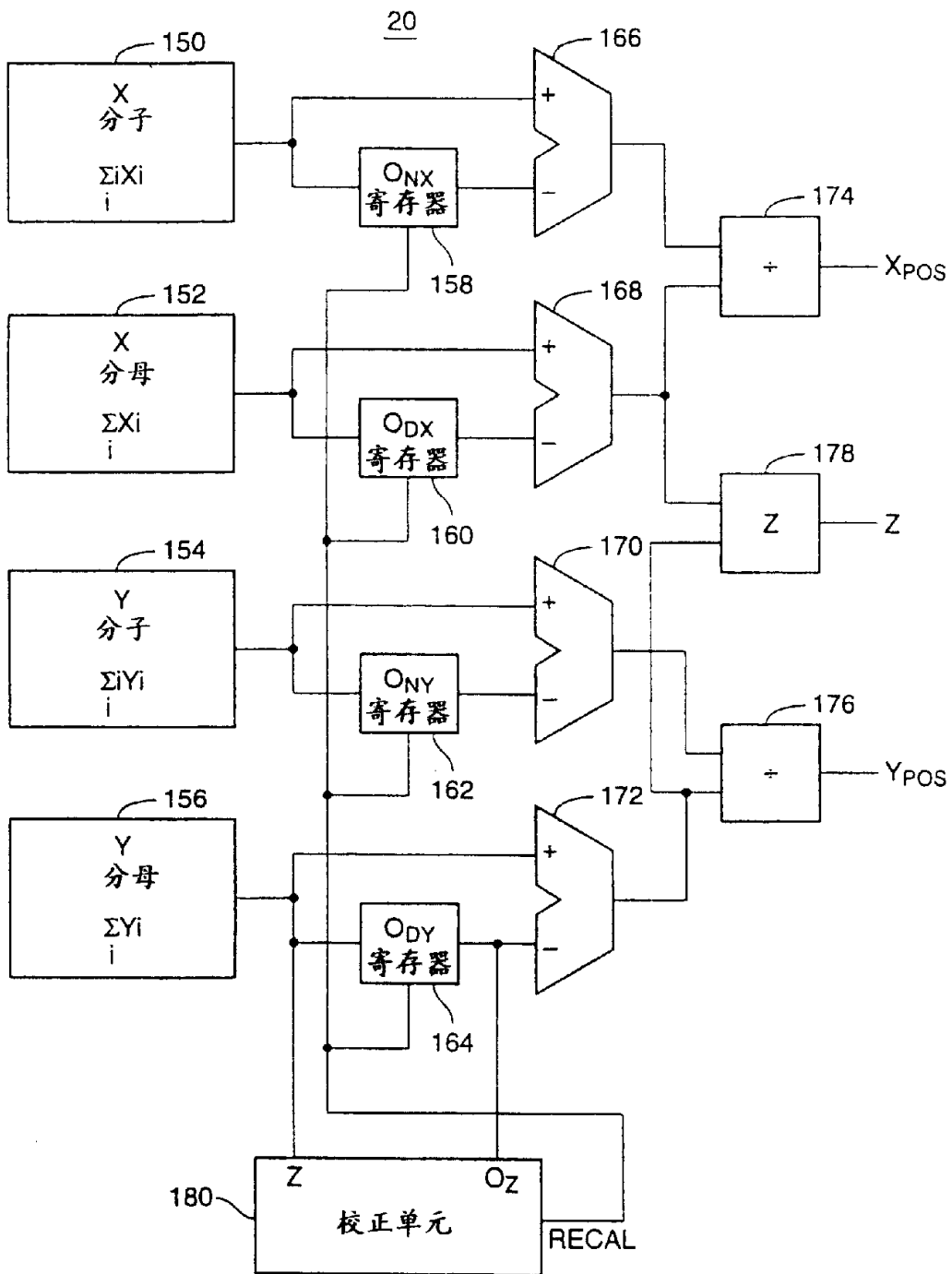


图 8

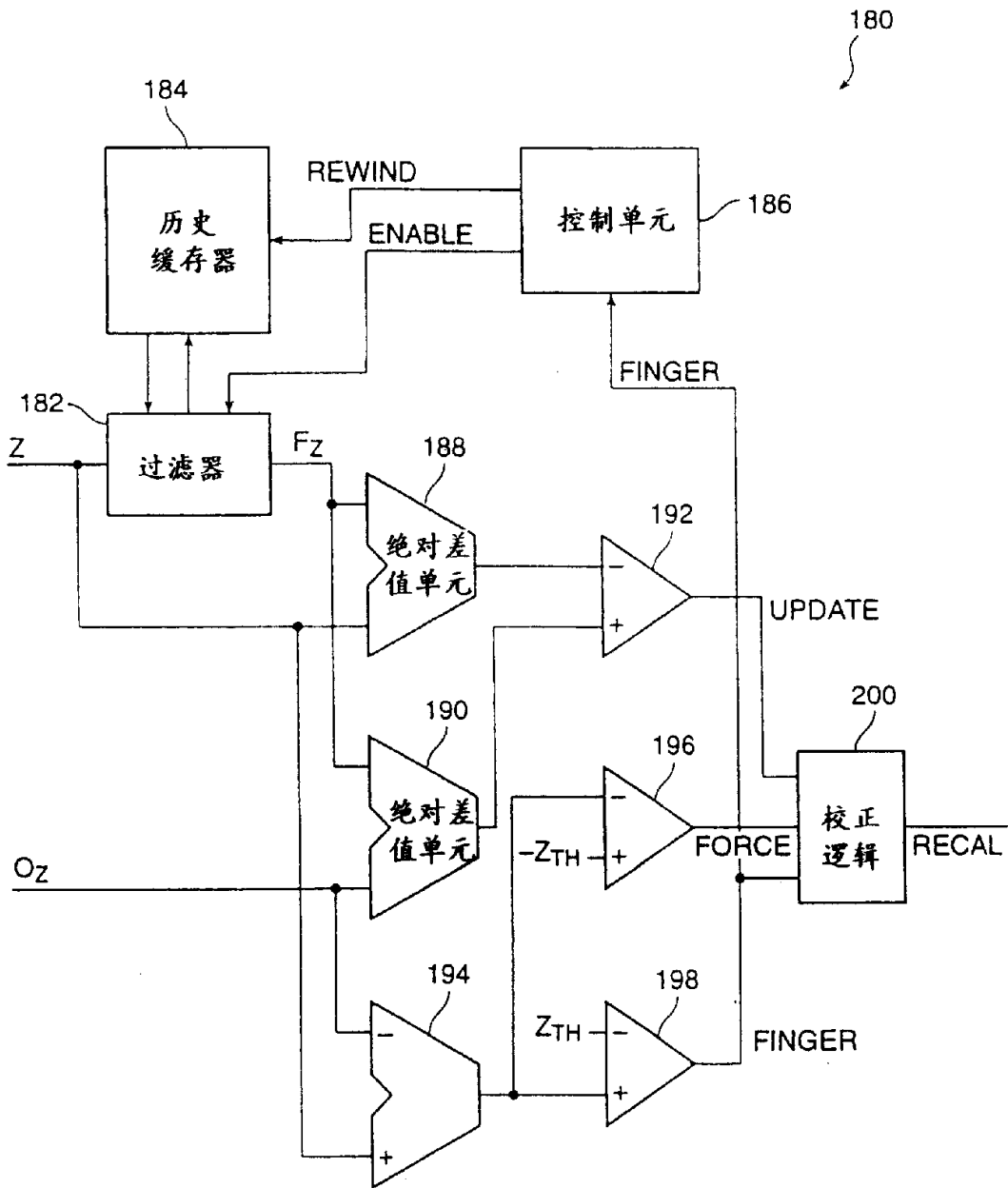


图 9

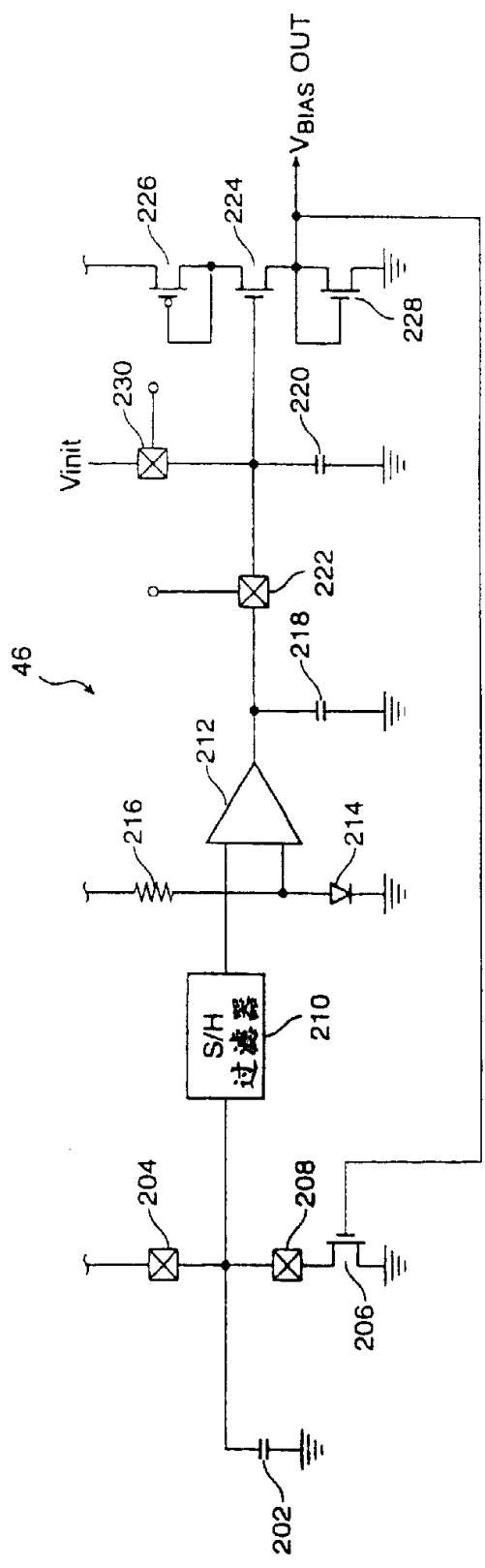


图 10

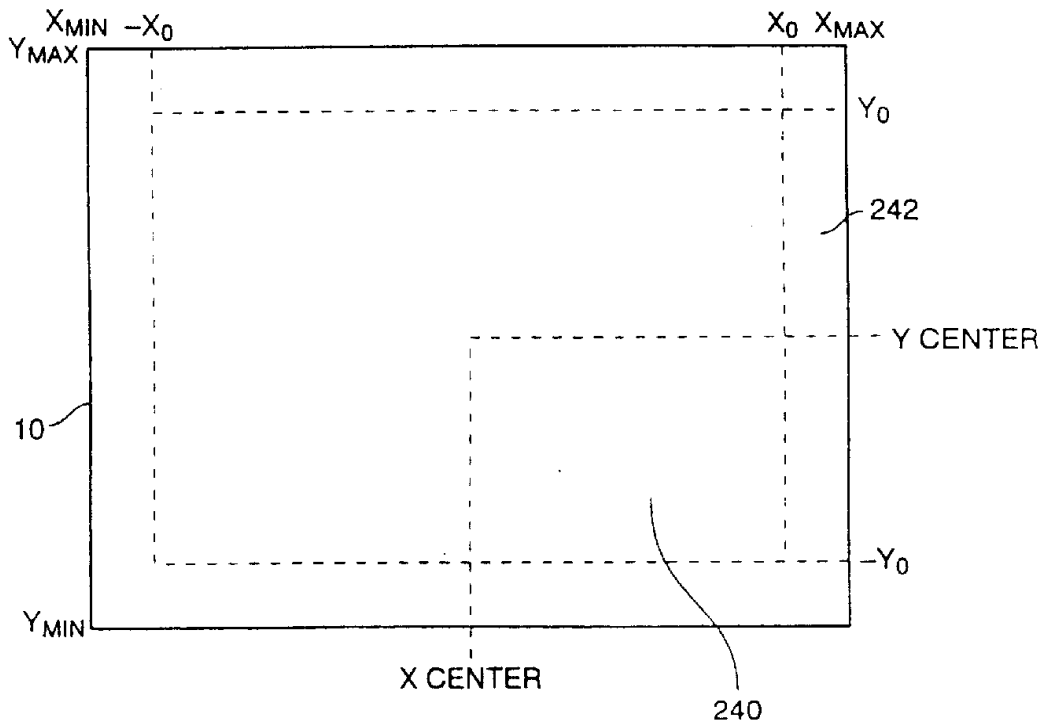


图 11

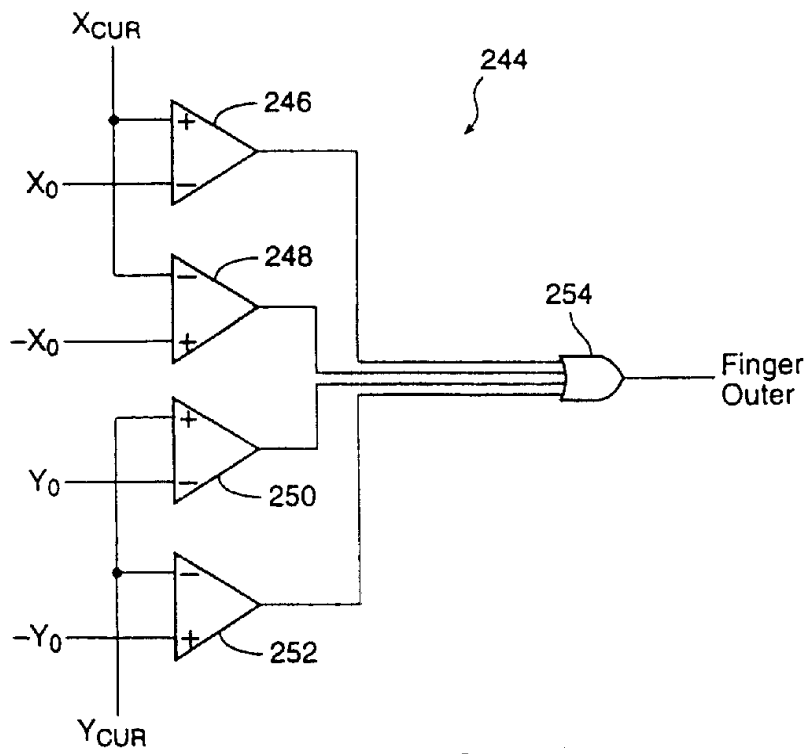


图 12 A

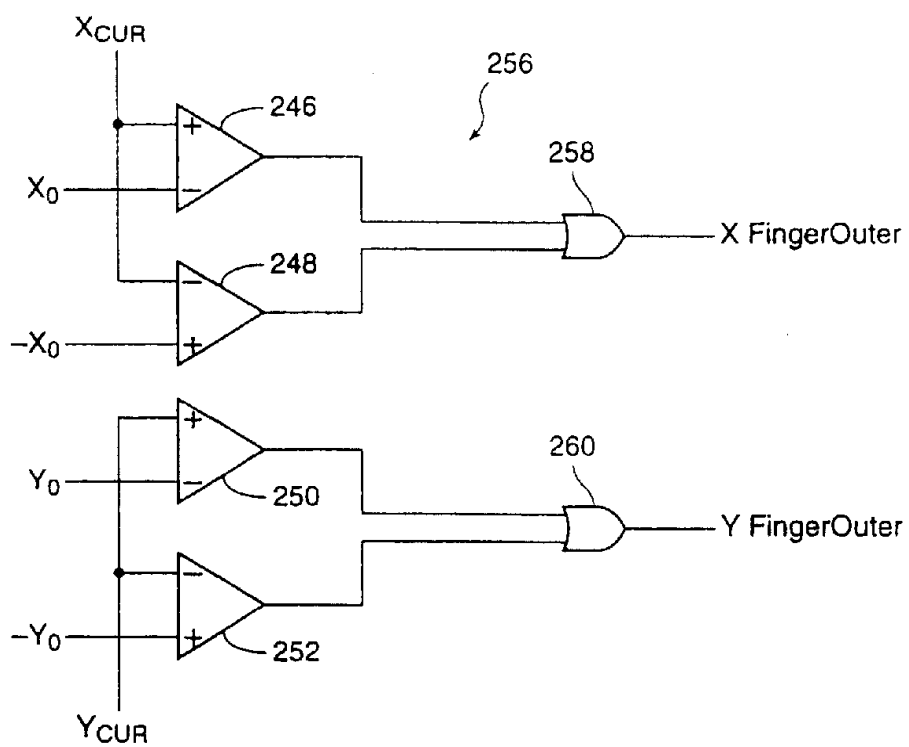


图 12 B

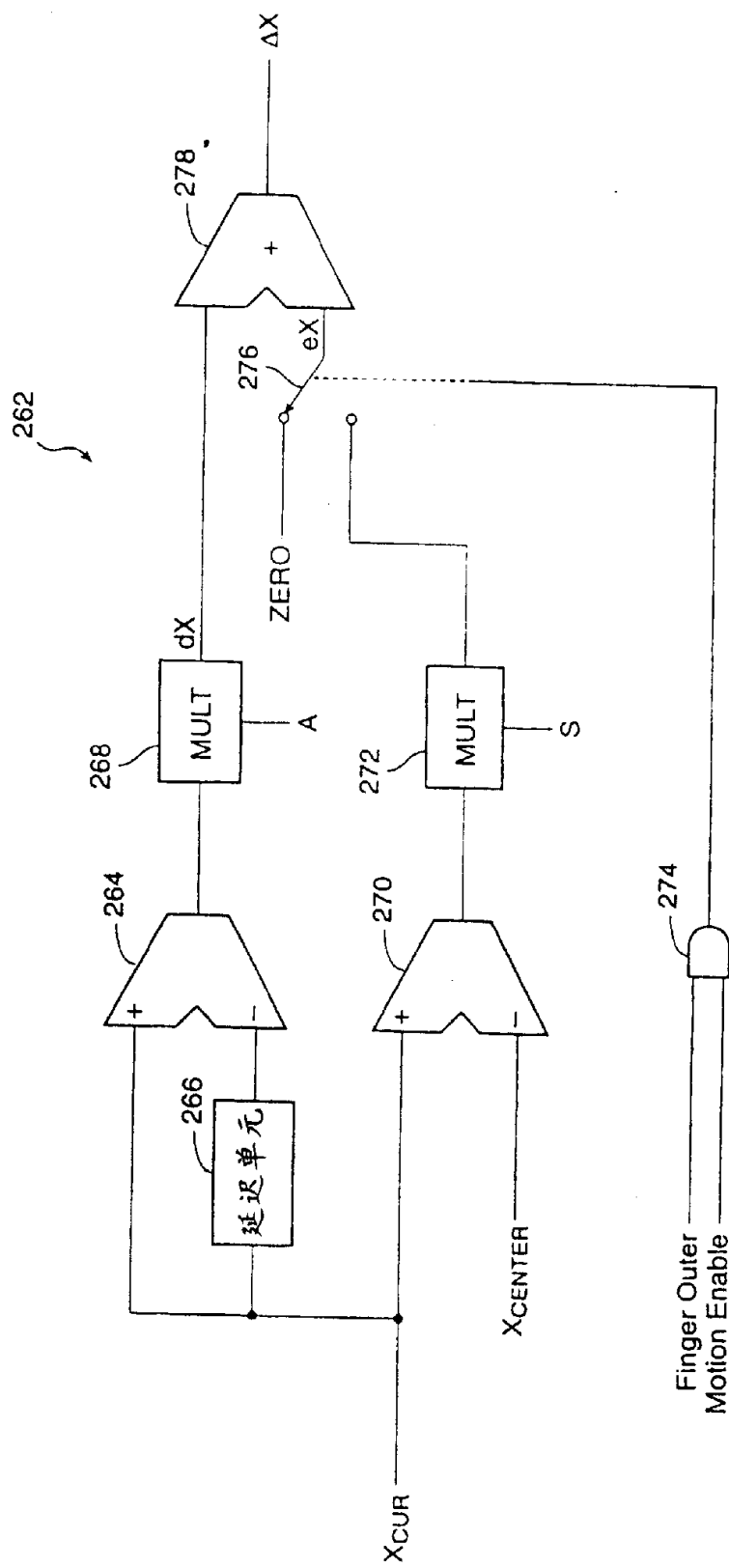


图 13



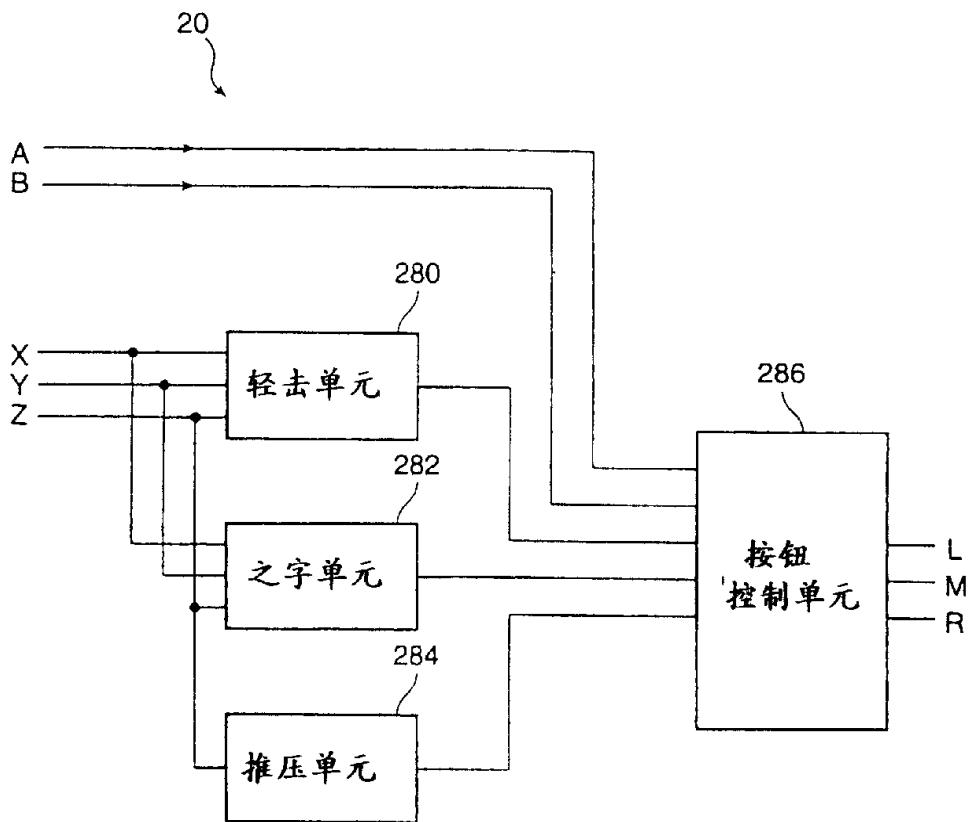


图 14

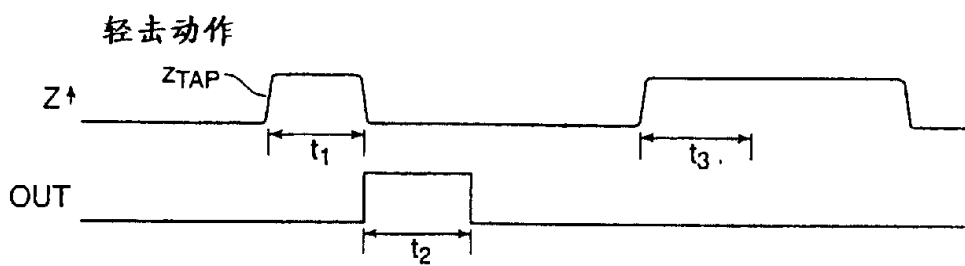


图 15 A

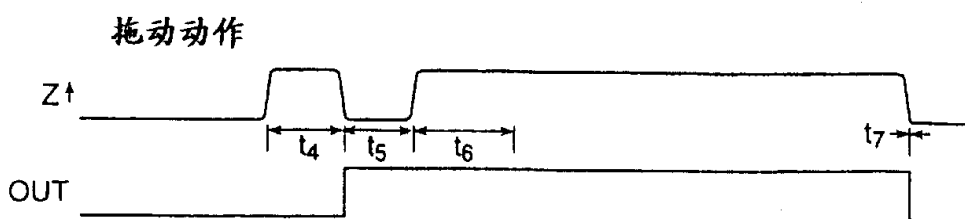


图 15 B

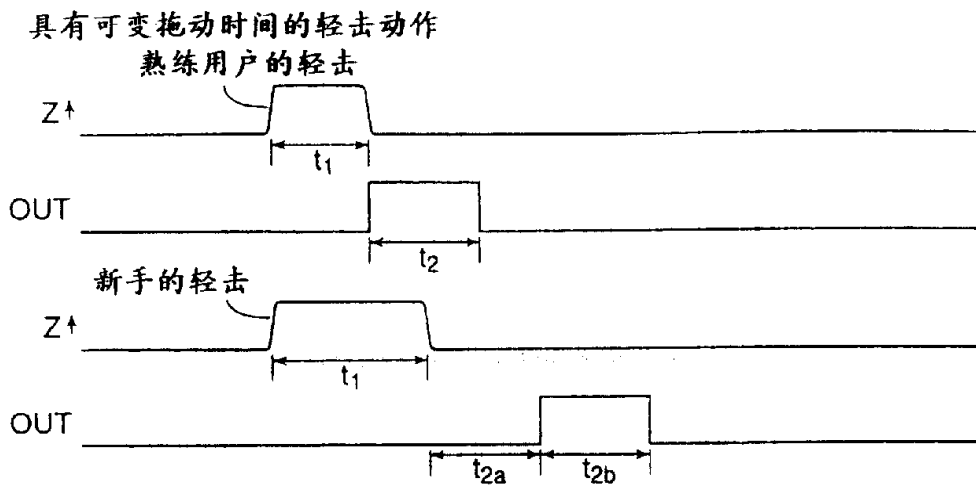


图 15 C

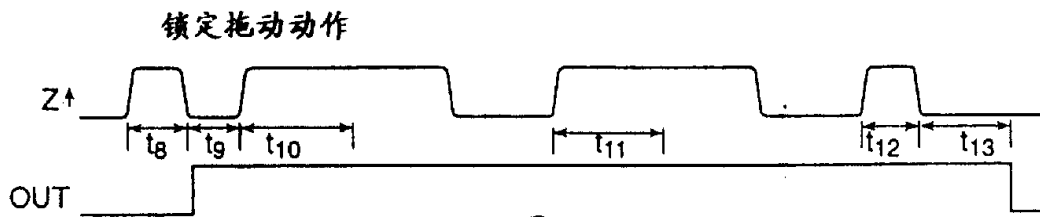


图 15 D

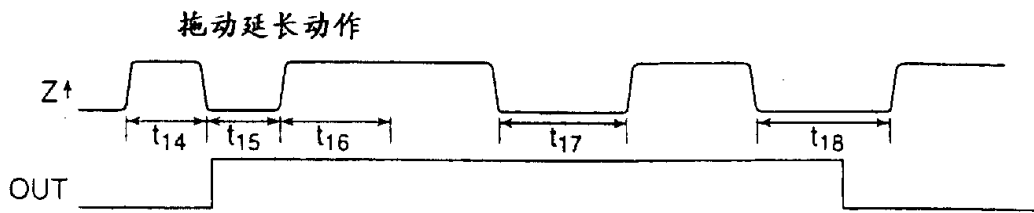


图 15 E

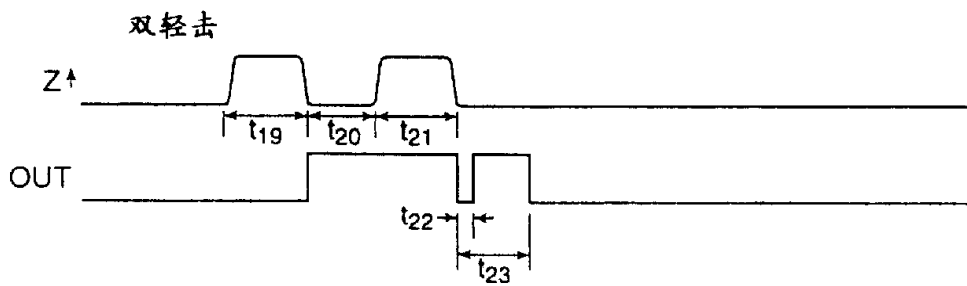


图 15 F

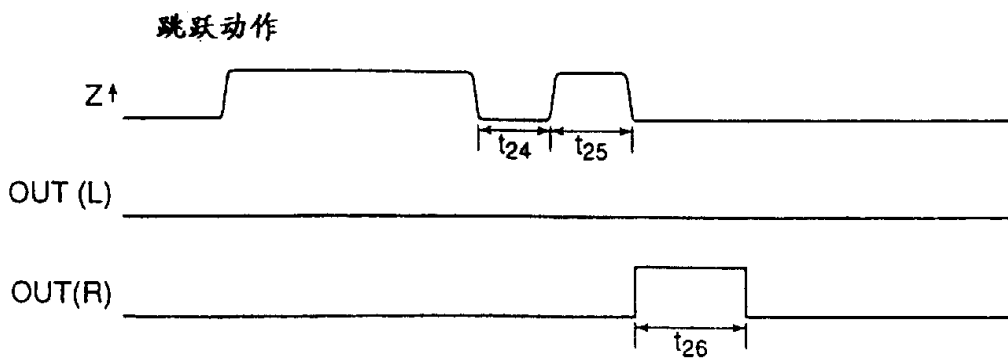


图 15 G

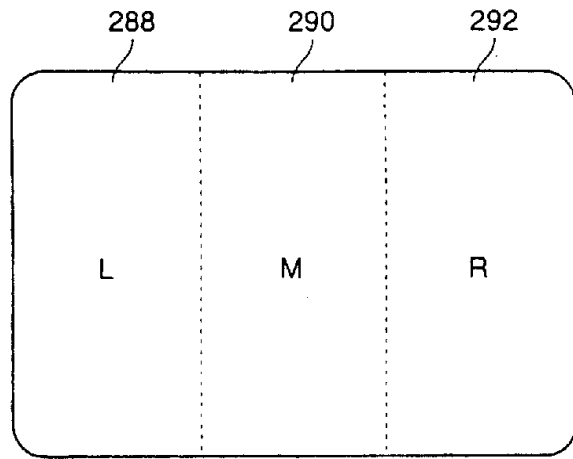


图 16 A

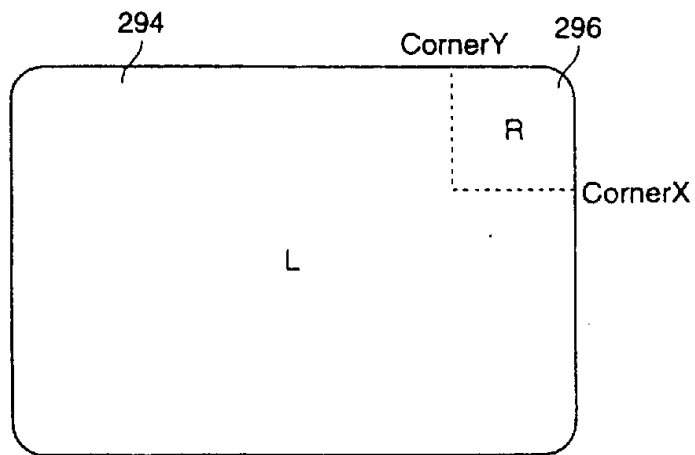


图 16 B

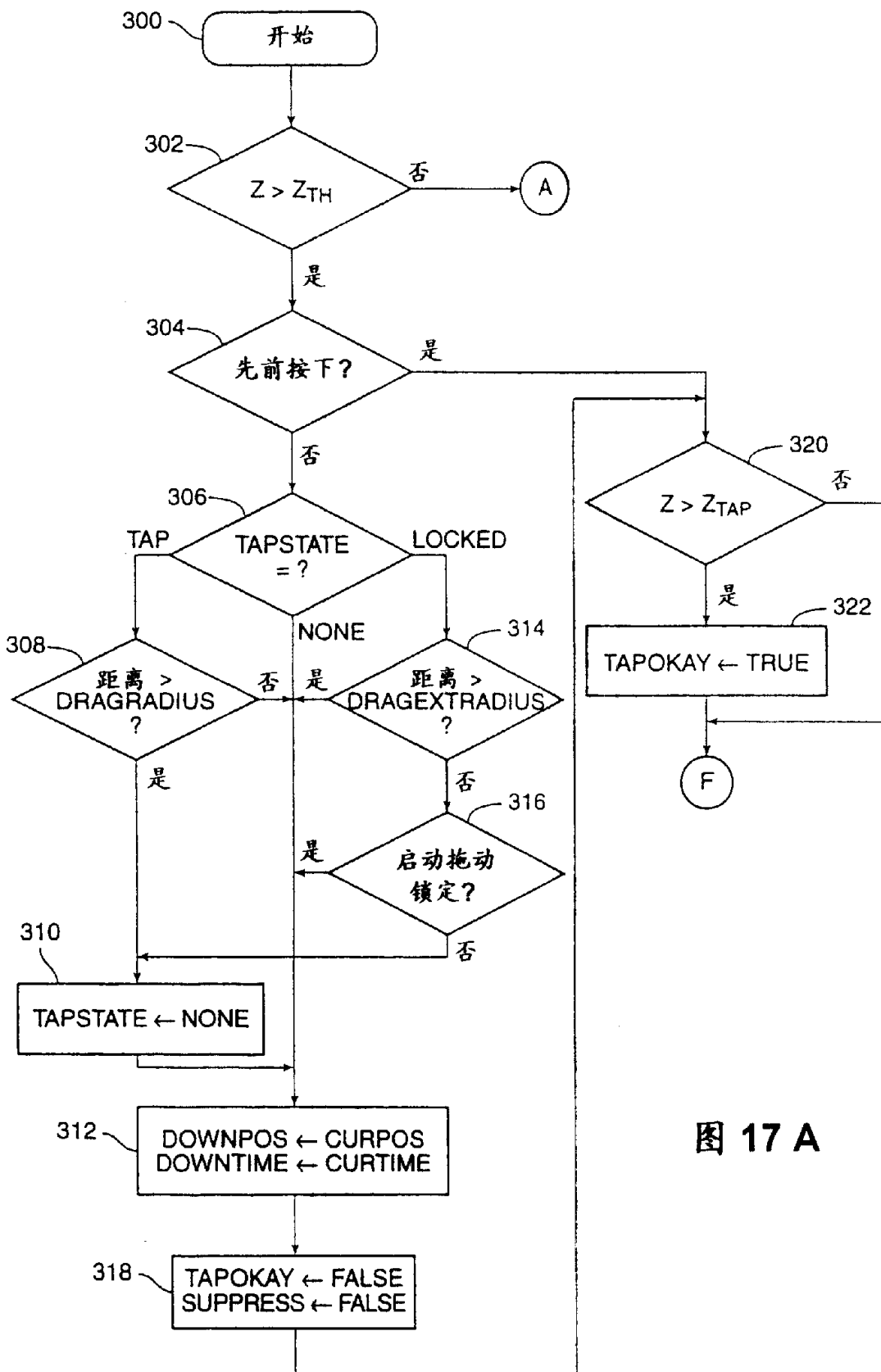


图 17 A

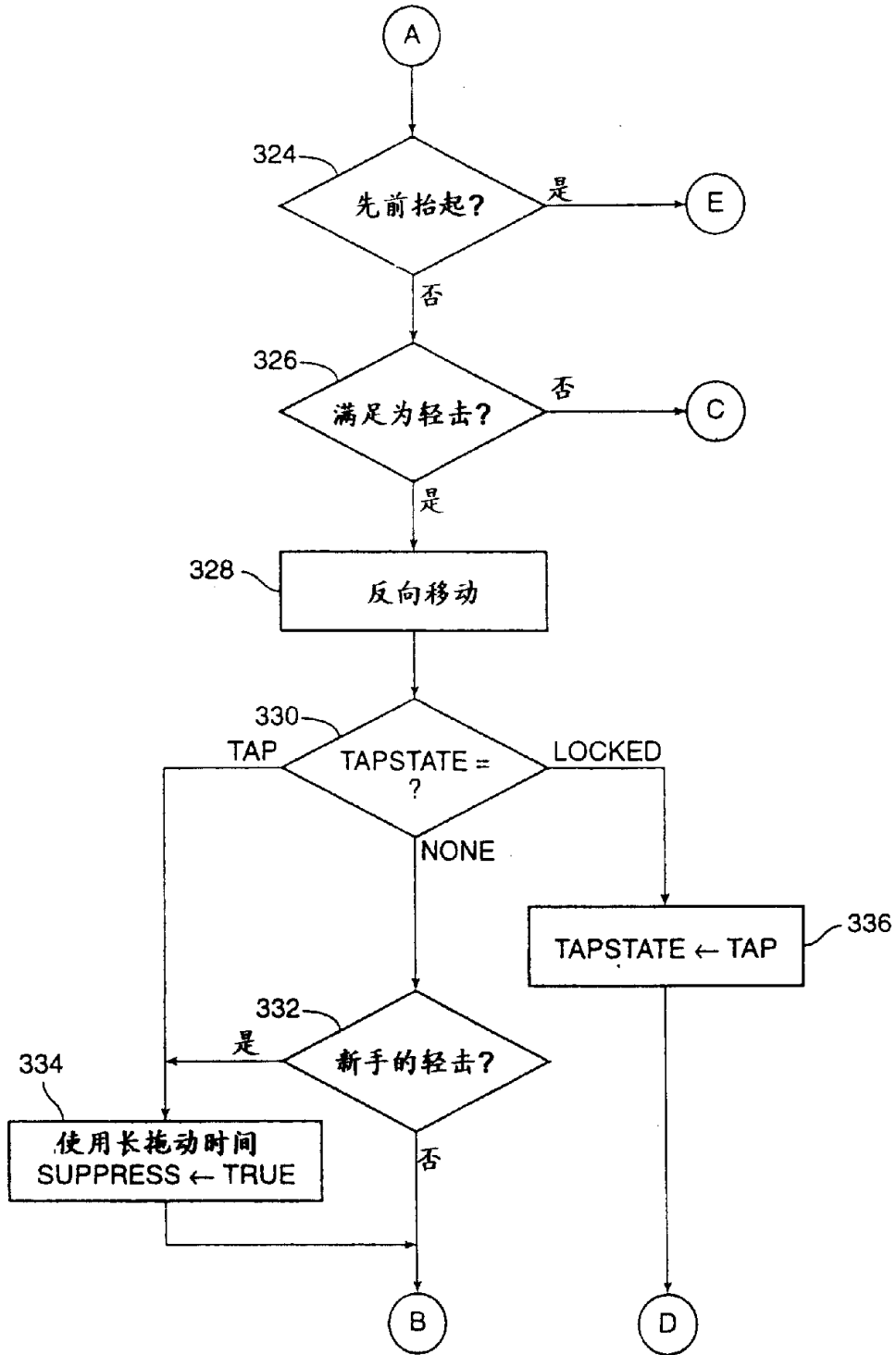


图 17 B

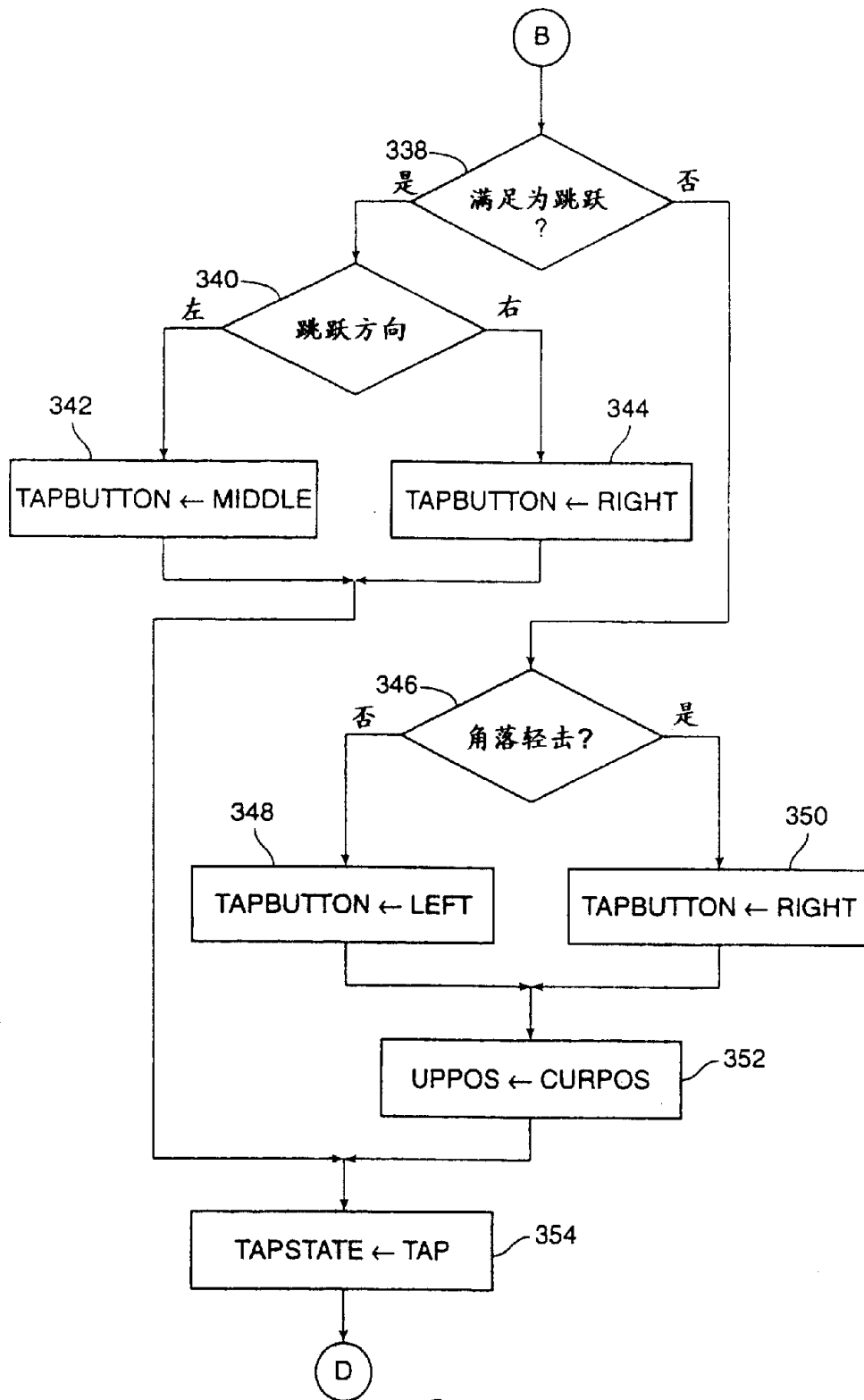


图 17 C

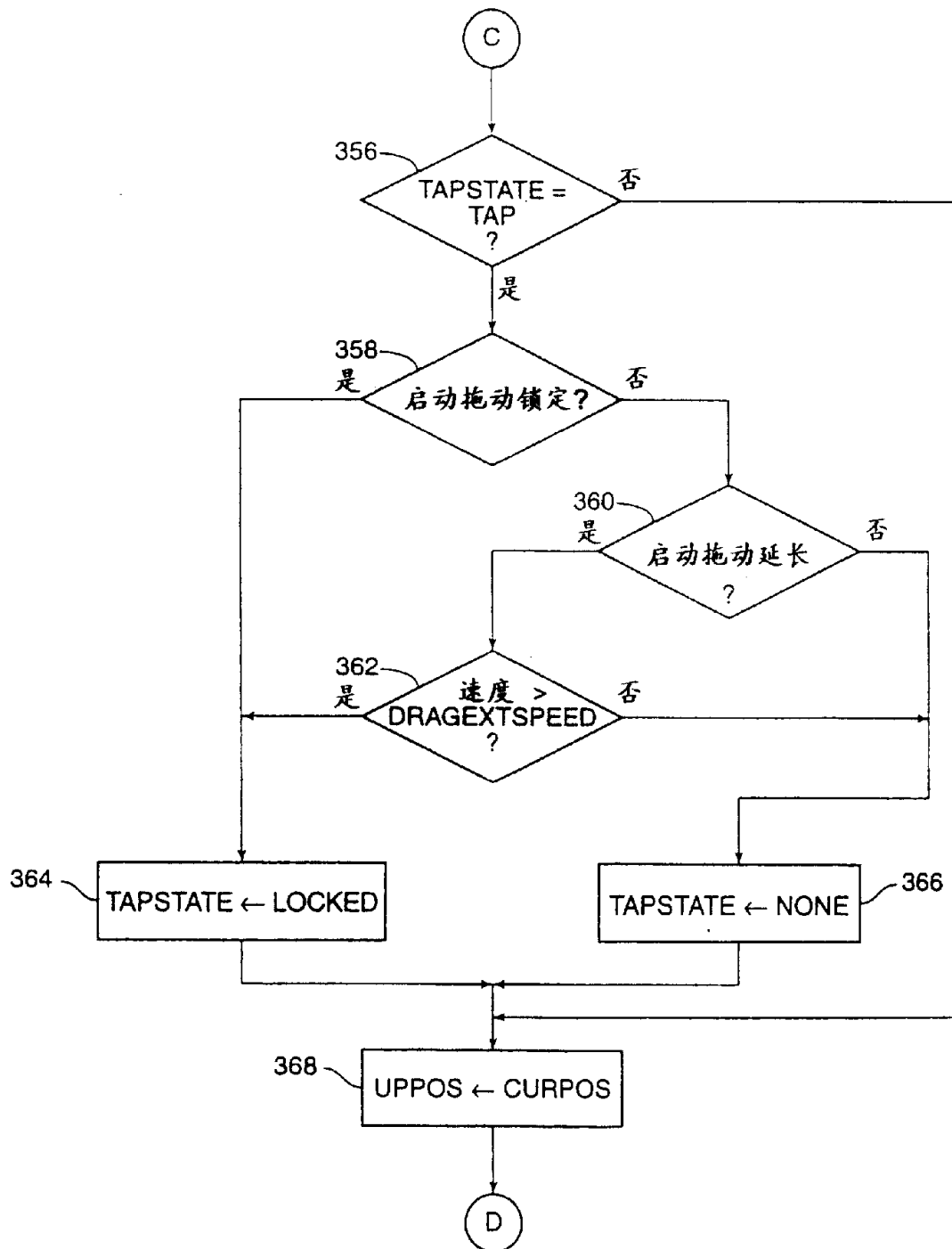


图 17 D



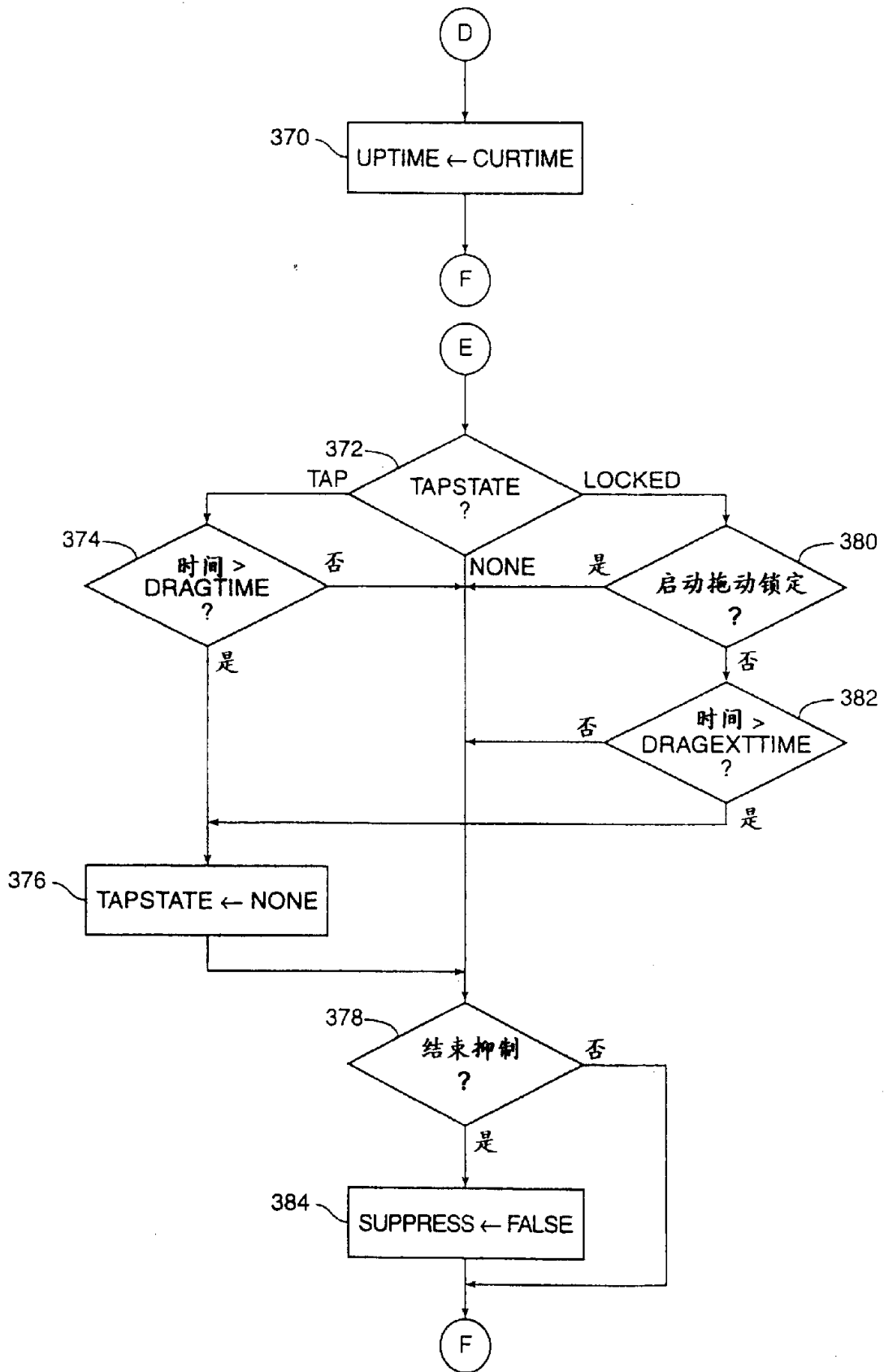


图 17 E

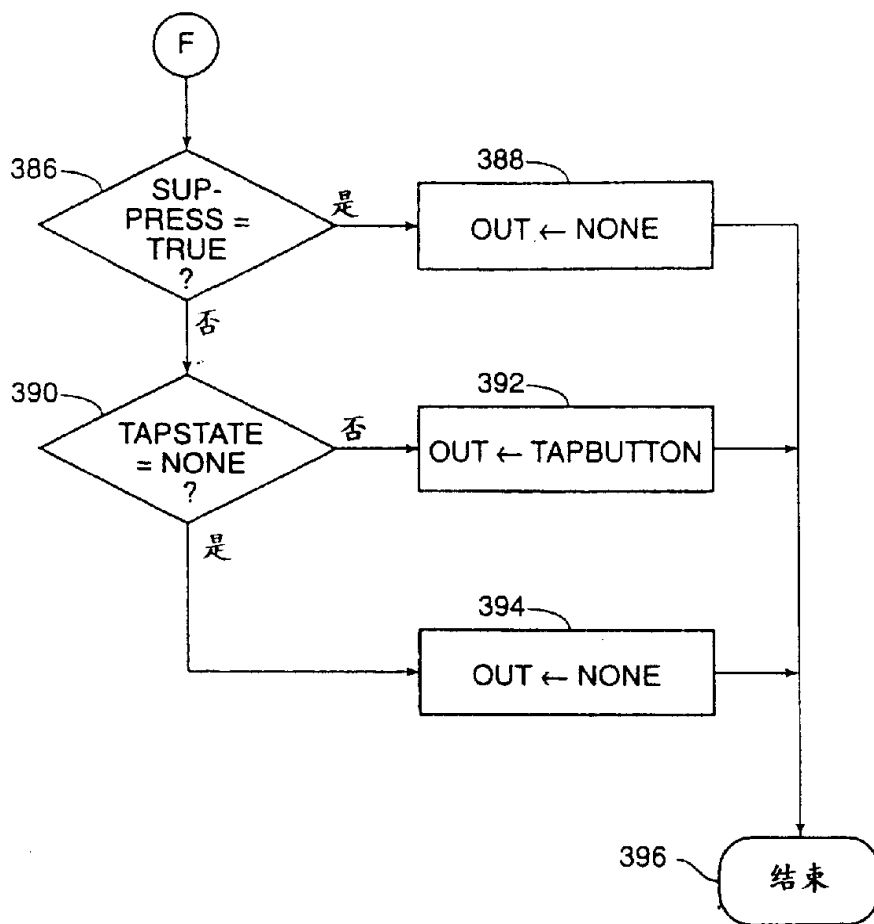


图 17 F

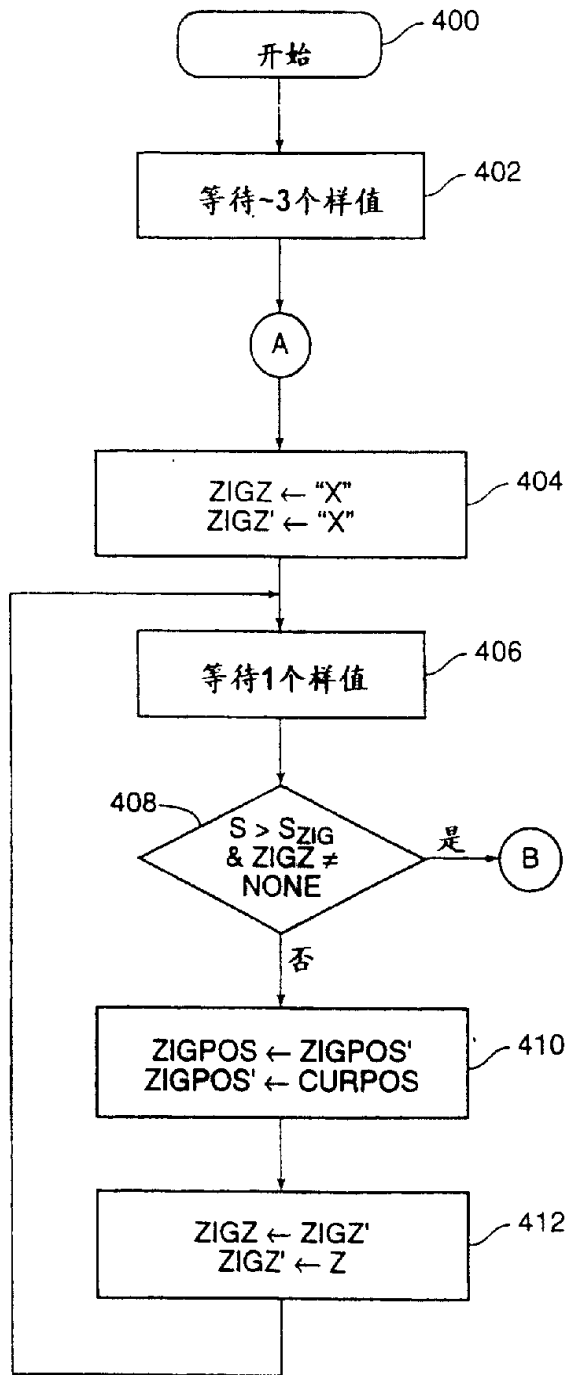


图 18 A

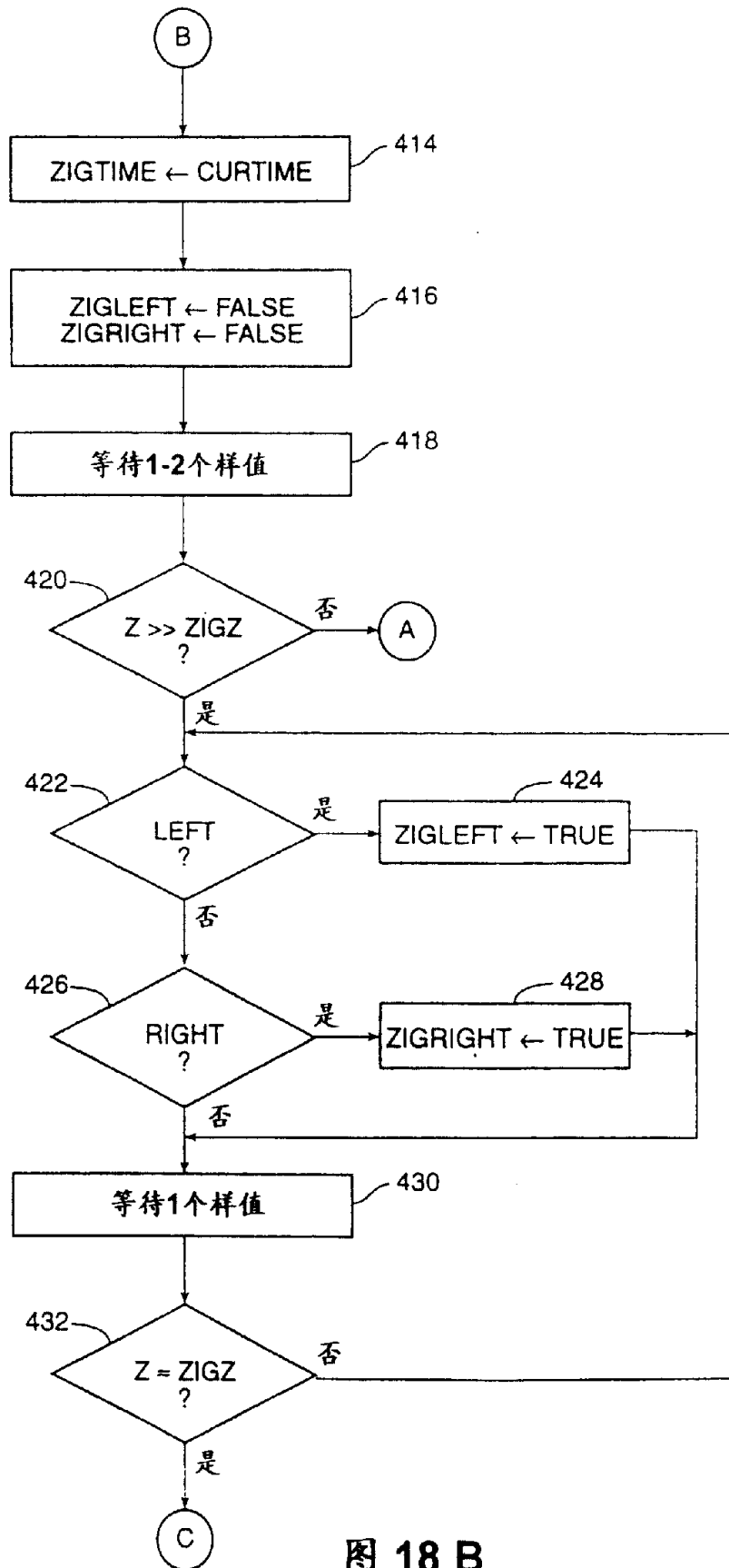


图 18 B

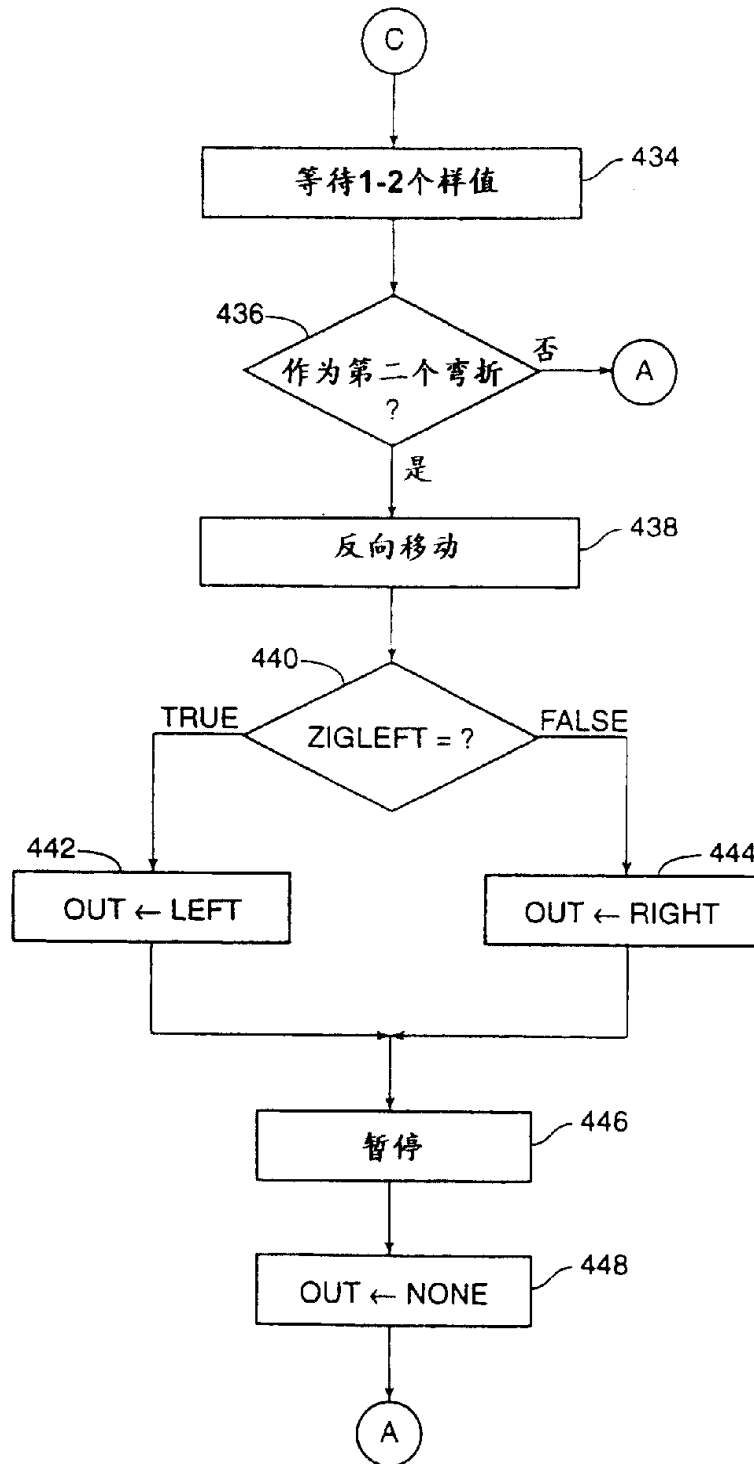


图 18 C

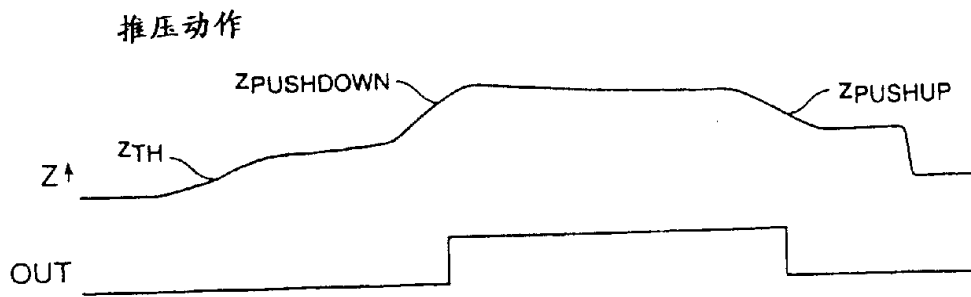


图 19

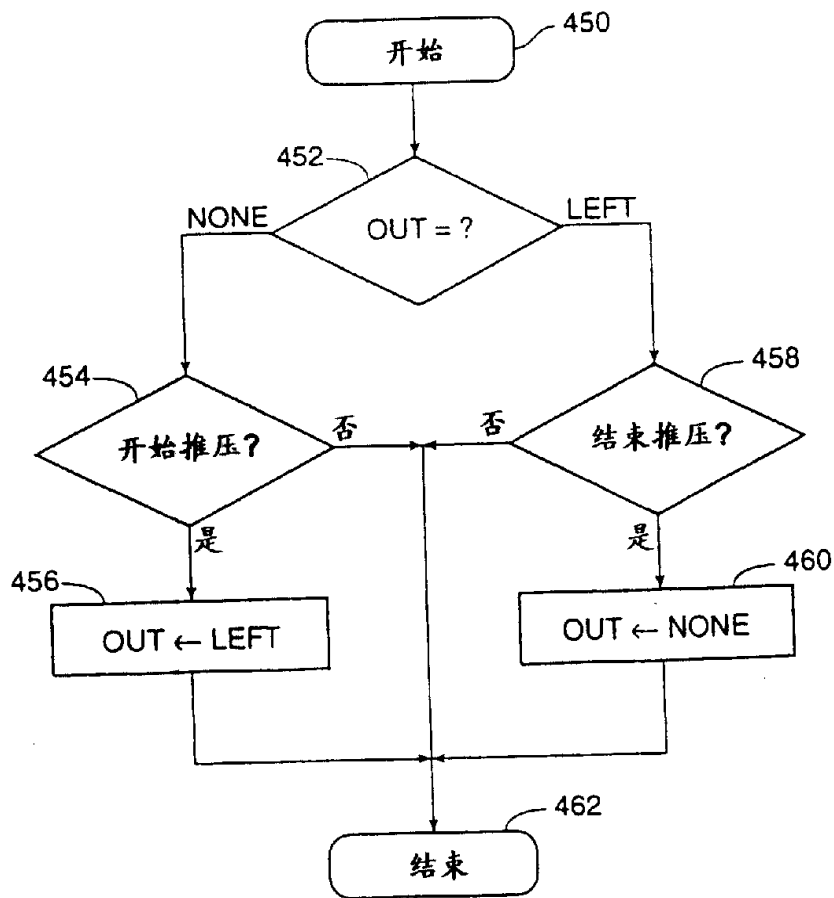


图 20

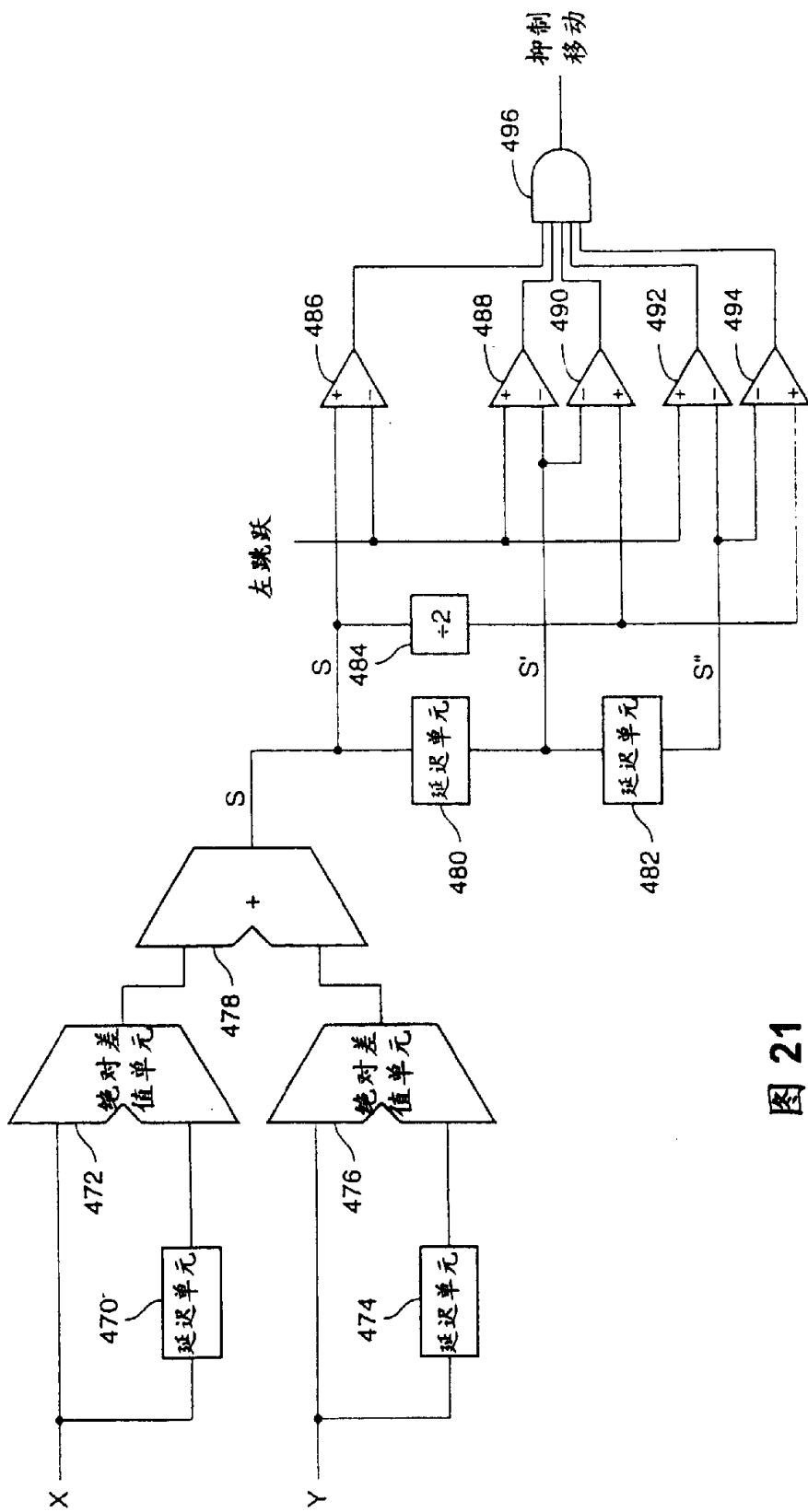


图 21