

[12] 发明专利说明书

[21] ZL 专利号 96112344.3

[45]授权公告日 2002年5月22日

[11]授权公告号 CN 1085455C

[22]申请日 1996.9.20

[21]申请号 96112344.3

[30]优先权

[32]1995.9.20 [33]DE [31]19534940.7

[73]专利权人 西门子公司

地址 联邦德国慕尼黑

[72]发明人 P·施蒂芬斯 W·普什

[56]参考文献

US 4511762A 1985.4.16 H04M15/10

审查员 焦景梅

[74]专利代理机构 中国专利代理(香港)有限公司

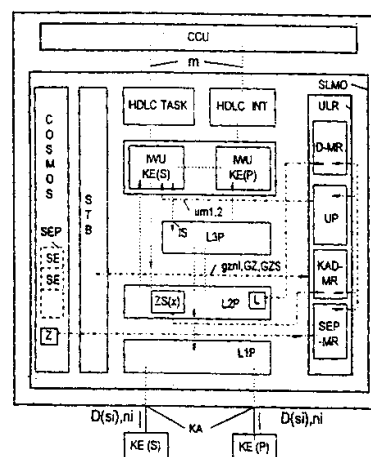
代理人 王勇 萧掬昌

权利要求书3页 说明书6页 附图页数4页

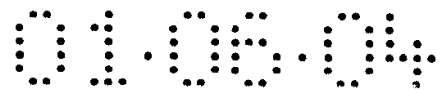
[54]发明名称 识别并减少通信系统用户接口模块中过负荷状态的方法

[57]摘要

根据本发明的方法,测量在一个通信系统(KS)的用户接口模块(SLMO)中一个微处理器系统(MPS)、所有到连接的通信终端设备(KE)的信号通道(D)和一个存储单元区(SEP)的满负荷(Z,ZS,ZS(X)、gml),并与预先给定的满负荷界限(GZ,GZS、gznl)比较,根据比较结果识别该用户接口模块(SLMO)的过负荷。通过提前识别至少两个过负荷,引入相应的或者总的快速有效的应对措施。



ISSN 1008-4274



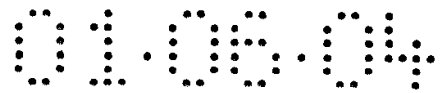
权 利 要 求 书

1. 识别用户接口模块 (SLMO) 过负荷的方法, 该用户接口模块 (SLMO) 上连接有通信终端设备 (KE) 的通信接口 (KA) 并与一个通信系统 (KS) 的中央控制单元 (CCU) 连接, 其中在用户接口模块 (SLMO) 内设计有一个微处理器系统 (MPS), 它用于处理在任
- 5 务中组织的中继和操作进程,
- 用它建立传输到所连接的通信终端设备 (KE) 上的信号消息 (si) 并通过通信接口各自的信号通道来传输这些信号消息,
 - 用它从一个由操作系统已建立的存储单元区 (SEP) 临时为进

10 程间通信和资源管理的进程分配存储单元 (SE),

 - 用它测量
 - - 微处理器系统 (MPS) 的满负荷;
 - - 所有信号通道 (D) 的满负荷; 和
 - - 存储单元区 (SEP) 的满负荷, 和

15 - 用它把测量的满负荷 (Z、ZS、ZS(X)、Gnl) 与预先给定的满负荷界限 (GZ、GZS、GZnl) 比较, 并根据比较结果识别用户接口模块 (SLMO) 的过负荷。
2. 根据权利要求 1 的方法, 其特征在于, 通过把测量的满负荷 (Z、ZS、ZS(X)、Gnl) 每次与不同的、预先给定的满负荷界限 (GZ、
- 20 GZS、GZnl) 比较来识别多个过负荷级。
3. 根据权利要求 1 或 2 的方法, 其特征在于, 在测量的满负荷 (Z、ZS、ZS(X)、Gnl) 超过预先给定的满负荷界限 (GZ、GZS、GZnl) 时, 认定用户接口模块 (SLMO) 为过负荷, 在测量的满负荷低于预先给定的满负荷界限 (GZ、GZS、GZnl) 时, 认定用户接口模块
- 25 (SLMO) 为正常负荷。
4. 根据权利要求 3 的方法, 其特征在于, 超过或低于满负荷界限 (GZ、GZS、GZnl) 是通过测量的满负荷 (Z、ZS、ZS(X)、Gnl) 在预先给定的时间区间内 (ZV1. . 3) 超过还是低于预先给定的满负荷界限 (GZ、GZS、GZnl) 而确定的。
- 30 5. 根据权利要求 1 或 2 的方法, 其特征在于, 为确定微处理器系统 (MPS) 的满负荷, 测量在一个预先给定的时间区间 (ZV1) 内的



由该微处理器系统 (MPS) 处理的空负荷任务 (ID-T) 数; 满负荷界限 (GZS) 由在预先给定的时间区间 (ZV1) 上一个预先给定的空负荷任务数 (ID-T) 确定; 以及超过和低于满负荷界限 (GZS) 是由空负荷任务 (ID-T) 的测量数 (GZS) 是超过还是低于一个预先给定的空负荷任务 (ID-T) 数 (GZS) 确定。

5

6. 根据权利要求 5 的方法, 其特征在于, 在处理完一个空负荷任务 (ID-T) 时, 计数器 (Z) 增量, 在这种情况下在每一个时间区间 (ZV1) 后存储该计数器 (Z) 的计数状态, 而该计数器 (Z) 向一个确定的计数器状态 (ZS) 复位。

10

7. 根据权利要求 1 或 2 的方法, 其特征在于, 为确定引向通信终端设备 (KE) 的信号通道 (D) 的满负荷, 采集在一个预先给定的时间区间 (ZV2) 内的所有信号消息 (si) 的长度 (L), 并且加在一个总的消息长度 (Gnl) 上; 满负荷界限由一个与时间区间 (ZV2) 有关的总消息长度 (Gnl) 确定; 以及超过或低于满负荷界限是由测量的总长度 (Gnl) 是超过还是低于一个预先给定的极限消息长度 (GZnl) 而确定。

15

8. 根据权利要求 5 的方法, 其特征在于, 在一个过负荷区间 (ZV1、2) 内当持续超过或者低于满负荷界限 (GZS、GZnl) 时, 建立一个指示第一过负荷状态的消息 (um1)。

20

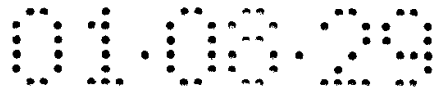
9. 根据权利要求 1 或 2 的方法, 其特征在于, 为确定存储单元区 (SEP) 的满负荷, 测量尚未占用的存储单元 (SE) 数目; 满负荷界限由一个预先给定的尚未占用的存储单元 (SE) 的数 (GZ) 确定; 以及超过或低于满负荷界限是由尚未占用的存储单元 (SE) 的测量数 (Z) 低于还是超过预先给定数目 (GZ) 的尚未占用的存储单元 (SE) 而确定。

25

10. 根据权利要求 9 的方法, 其特征在于, 在每次对连接的通信终端设备 (KE) 发送一个信号消息 (si) 后, 要测量尚未占用的存储单元 (SE) 的数目 (Z)。

30

11. 根据权利要求 1 或 2 的方法, 其特征在于, 附带地在预先给定的时间区间 (ZV3) 内, 对通过通向一个通信终端设备 (KE) 的通信接口 (KA) 的一条信号通道 (D) 传输的信号消息 (si) 的每一数



目 (ZS(X)) 进行计数, 并且存储当时具有最高数目 (ZS(X)) 的通信接口 (KA)。

12. 识别并减少用户接口模块 (SLMO) 过负荷的方法, 该用户接口模块 (SLMO) 上连接有通信终端设备 (KE) 的通信接口 (KA) 并与一个通信系统 (KS) 的中央控制单元 (CCU) 连接, 其中在用户接口模块 (SLMO) 内设计有一个微处理器系统 (MPS), 它用于处理在任务中组织的中继和操作进程,

- 用它建立传输到所连接的通信终端设备 (KE) 上的信号消息 (si) 并通过通信接口各自的信号通道来传输这些信号消息,

10 - 用它从一个由操作系统已建立的存储单元区 (SEP) 临时为进程间通信和资源管理的进程分配存储单元 (SE),

- 用它测量

- - 微处理器系统 (MPS) 的满负荷;

- - 所有信号通道 (D) 的满负荷; 和

15 - - 存储单元区 (SEP) 的满负荷, 和

- 用它把测量的满负荷 (Z、ZS、ZS(X)、Gnl) 与预先给定的满负荷界限 (GZ、GZS、GZnl) 比较, 并根据比较结果识别用户接口模块 (SLMO) 的过负荷,

在该微处理系统中, 其特征在于

20 在确认有关微处理器系统 (MPS) 和信号通道 (D) 存在过负荷状态后, 引入第一级减小过负荷措施; 在确认有关存储单元区 (SEP) 的尚未占用的存储单元 (SE) 过负荷后, 引入第二级减小过负荷措施。

13. 根据权利要求 12 的方法, 其特征在于, 在通信终端设备 (KE) 中处于第一级减小过负荷措施时, 光学的或者语音输出的信号消息 (si) 不再向单个或多个强过负荷通信终端设备 (KE) 发送。

14. 根据权利要求 13 的方法, 其特征在于, 抑制当前传输最大数目 (ZS(X)) 信号消息 (si) 的通信接口 (KA) 的信号消息 (si)。

15. 根据权利要求 12 的方法, 其特征在于, 第二级减小过负荷措施是通过把单个或多个通信接口 (KA) 停止运行而实现的。



说明书

识别并减少通信系统用户接口模块中过负荷状态的方法

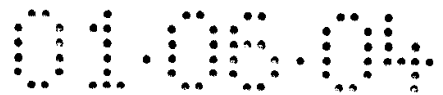
5 本发明涉及一种识别并减少通信系统用户接口模块中过负荷状态的方法，其中通信终端设备连接在用户接口模块的通信接口上。

程序控制的通信系统装备有一个中央控制单元和与之相连的用户接口模块，通过这些接口模块实现为接入通信终端设备而做的通信连接。例如有带 24 个通信接口的用户一个接口模块，可连接至少 24
10 个通信终端设备。通过每一个这样的通信接口引入一个具有例如 16 千位/秒的标准化传输容量的信号通道 - 在专业圈内称为 D 信道。通过该信号通道既向通信终端设备，也向通信系统传输信号消息。通过对通信终端设备的操作器表层的中继技术调整 - 特别是在激励通信终端设备的场合 - 通过该通信系统可出现从通信系统到通信终端设
15 备极高的信号消息通量。这种情况特别可能在通信终端设备一定的应用中导致到通信终端设备过高的信号消息通量，从而使得用户接口模块过负荷。

另外每一用户接口模块装备一个微处理器系统，用它来处理由在用户接口模块中执行的进程传输来的处理请求。在这种情况下这些进
20 程作为独立的任务被组织起来，它们在消息处理方面有不同的优先级。在一个用户接口模块内，为进程间通信的消息处理和为用户接口模块上的其它资源的资源管理而提供了微处理器系统的资源处理能力。微处理器系统的过负荷既可由在通信终端设备的方向上极高的消息潮引起，也可由要传输到中央控制单元的消息引起。

25 一个已知的识别用户接口模块过负荷的方法是考虑由操作系统管理的存储单元区尚未占用的存储单元数目。存储单元被临时配给每一进程亦即任务为中间存储信息，也就是说每一进程可以把它的数据或者说信息存储在分配给它的存储单元中。

为判断过负荷而专门监视存储单元区是有缺点的，因为在这种方
30 法中过负荷状态识别太晚，因此不能引入快速有效的相应的措施。用



户接口模块在超过一个过负荷门限后在故障状态下被控制，这只能通过重新起动，也就是说通过对用户接口模块的复位才能消除。

5 本发明的任务在于提出一种识别用户接口模块过负荷的方法，使得在发觉过负荷时通过相应的措施避免用户接口模块一般的故障状态或停机。

10 根据本发明的方法的重要方面在于，为识别过负荷，连续测量微处理器系统、引向通信终端设备的信号通道和存储单元区的满负荷，并通过把测量的满负荷和预先给定的满负荷界限比较来识别用户接口模块的过负荷。在发现至少两个或三个过负荷后，可以引入针对性的或者总体的、快速有效的对应措施。一个这样的措施例如是，限制极大地影响处理器处理 - 资源的信号消息的产生和向引起信号消息潮的通信设备的传输。在这种情况下可以查明当前向其传输很大数目信号消息的那个通信终端设备并且引入相应的措施 - 也就是说不再传输所有的信号消息。特别适用的场合是这些信号消息在通信终端设备上以光学的、或者语音输出的。作为另外的应对措施可以使单个的或者全部通信接口停止工作，以避免每一用户接口模块的当前待处理的过负荷。

20 通过把每次测量的满负荷与不同的预先给定的满负荷界限比较而有利的区分多个过负荷级。这种对付过负荷策略在一个多级应对措施策略的情况下很有意义。这里单个的应对措施代表一个整体策略的一部分或几部分。

25 根据一个有利的按照本发明的方法的实施例，用户接口模块测量的满负荷在超过预先给定的满负荷界限时可认定为过负荷。因此，低于预先给定的过负荷界限时表示正常负荷。为抑制极短时期的过负荷和测量错误，仅在预先给定的时间区间之内测量时，才确定是超过还是低于过负荷界限。

30 以有利的方式通过在一个预先给定的时间区间内的被处理的无负荷任务数而确定微处理器系统的满负荷。如果测量的数低于一个代表满负荷界限的预先给定的空负荷任务数，则可认定该微处理器系统过负荷。按优选方案通过一个计数器采集空负荷任务数。

5 信号通道的满负荷特别有利地通过确定一个总消息长度测量。在这种情况下为一个信号通道采集单个信号消息的消息长度，并且加在一个总消息长度上。如果在一个时间区间内的总消息长度超过一个代表满负荷界限的界限消息长度，则可认定通过信号通道到通信终端设备过负荷。

于是，当在一个过负荷区间内持续超过满负荷界限时，则可确定用户接口模块为第一过负荷状态。在这种情况下，建立一个指示第一过负荷状态的消息。这种附带的措施避免在短时超过单个满负荷界限时引入相应的措施，而在确认过负荷后可以引入相应的措施。

10 以有利的方式通过测量尚未占用的存储单元确定存储单元区的过负荷。在这种情况下，测得的未占用存储单元与一个代表满负荷界限的、预先给定的数比较，在低于预先给定的数时，确定存储单元区过负荷。优选在每一次给连接的通信终端设备发送一个信号消息后测量存储单元区的满负荷。这一时点是有利的，因为在一次可能发生的
15 超过满负荷界限时可避免继续发送消息。

下面按照一幅框图和五幅流程图来说明根据本发明的方法。其中：

- 图 1 表示在一个通信系统中的用户接口模块的程序结构；
- 图 2 表示微处理器系统进程满负荷测量的流程图；
- 20 图 3 表示信号通道满负荷测量的流程图；
- 图 4 表示存储单元区满负荷测量的流程图；
- 图 5 表示特定通信接口信号通道满负荷测量的流程图；和
- 图 6 表示过载识别的流程图。

25 图 1 表示一个通信系统 KS，其中作为示例绘出一个用户接口模块 SLMO。在这个通信系统上，或者更确切地说在用户接口模块 SLMO 上可连接通信终端设备 KE。这些通信终端设备 KE 可以以激励法 (S) 或者按照一种协议法 (P) 运行，前者是通信终端设备 KE 的所有调整由通信系统 KS 进行，后者是在传输一个信号消息 s_i 后通信终端设备独立地调整。信号消息 s_i 通过一个由每一通信接口 KA 导向的信号通道 D 与使用信息 n_i ，例如数字化的语音信息，一起传输。
30

在通信系统 KS 中，用户接口模块 SLMO 与一个通信系统 KS 协调，并与实行控制的控制设备 CCU 连接。图 1 另外还示出以示例表示的用户接口模块 SLMO 的程序结构，其中另外的未表示出的用户接口模块 SLMO 是以同样的程序结构实现的。该程序结构以后面叙述的不同的进程构建。这些进程作为独立的任务来组织，它们分配关于消息处理的不同的优先级，这些消息是由中央控制设备 CCU 或者通信终端设备 KE 来的或者到这些设备去的。

在中央控制单元 CCU 和用户接口模块 SLMO 之间的通信通过两个 HDLC 传输子例程 HDLC TASK、HDLC INT 实现。两个 HDLC 传输子例程 HDLC TASK、HDLC INT 都包含通信驱动子例程，其中第一个 HDLC 传输子例程 HDLCTASK 用于发送消息 m 到中央控制单元 CCU，而第二个 HDLC 传输子例程 HDLC INT 负责接收从中央控制单元 CCU 来的消息 m。在用户接口模块 SLMO 中对 HDLC 传输子例程 HDLC TASK、HDLC INT 的请求以最高的优先级处理。两个 HDLC 传输子例程 HDLC TASK、HDLC INT 通过一个前端设备处理器 VDH 通信，其中实现对由通信终端设备 KE 传输的信号信息 si 的中继技术的前处理（呼叫处理）。在前端设备处理器 VDH 中设计了一个由程序模块 IWU-S 实现的进程，它用于处理激励通信终端设备 KE (S) 的中继技术进程；另外还设计了一个由程序模块 IWU-S0 实现的进程，它用于处理协议通信终端设备 KE (P) - 例如一个带 S0 接口的 ISDN 通信终端设备 - 的中继技术消息。另外还特别包含为建立用于激励通信终端设备 KE (S) 的显示单元的特定信息的第三层程序 L3P，其专门通过一个确定的接口 IS 与程序 IWU-S 通信。第三层程序 L3P 主要设计用于建立和管理所有连接在每一用户接口模块 SLMO 上的激励通信终端设备 KE (S) 的显示信息，这里这些在每一激励通信终端设备 KE (S) 上的显示信息的信息内容在一个显示设备上光学显示。

在一个另外的第二层程序 L2P 中，既在中央控制单元 CCU 的方向上也在通信终端设备 KE 的方向上实现了按照为用户 - 网络 - 信号和信息传输的 OSI 参考模型的第二层。该第二层为第三层亦即传输层

接管在两个方向上的信号信息 s_i 的保密传输。

第二层程序 L2P 与第一层程序 L1P 通信, L1P 对每一个连接的通信终端设备 KE 起到与交换技术传输设备和供电设备的连接环节的作用。通过该第一层程序 L1P 实现通信接口的接通和拆除、控制测试帮助以及实现电源管理。

为管理和操作微处理器系统资源, 实现了一个操作系统 COSMOS。该操作系统 COSMOS 也管理存储单元区 SEP 的存储单元 SE, 并且根据相应的请求 - 特别是来自消息处理的请求 - 临时分配给每一进程或者更确切地说每一程序模块, 带存储单元 SE 的存储单元区 SEP 在操作系统块中用虚线矩形框表示。在操作系统 COSMOS 中另外还设计了一个计数器 Z, 用它计数存储单元区 SEP 的尚未占用的存储单元 SE。

为对用户接口模块 SLMO 的操作技术控制亦即操作技术调整, 设计了一个安全和操作技术程序模块 STB, 通过这个操作技术程序模块 STB 可以使用所有关于为过负荷识别需要的满负荷界限的操作参数。这些操作参数的特别是关于微处理器系统满负荷界限 GZS、界限完全信息长度 GZ_{nl} 和区间时间 IV 以及存储单元区满负荷界限 GZ 的参数。

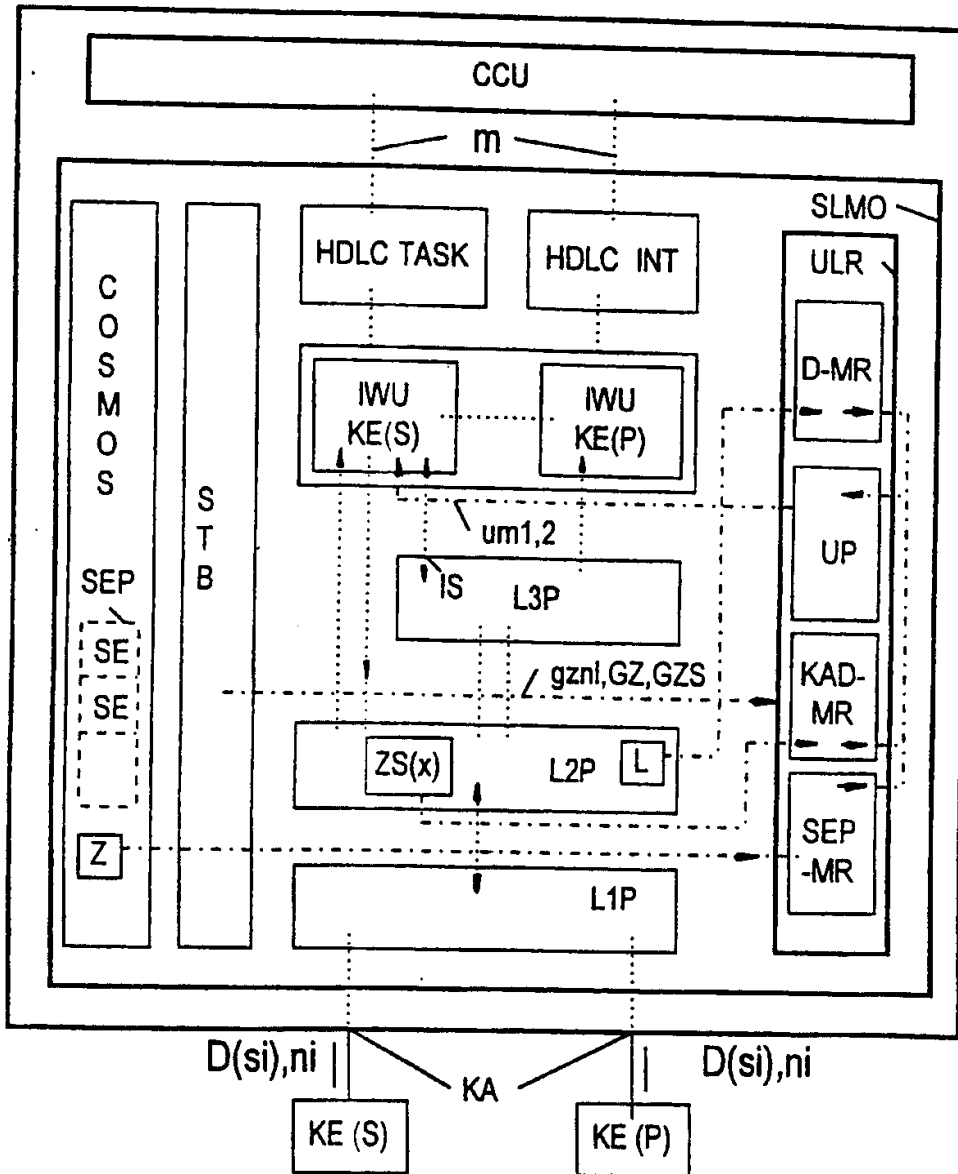
另外在第二层程序 L2P 中确定每一个在通信终端设备 KE 的方向上要传输的信号消息 s_i 的消息长度 L, 并通知一个过负荷子例程 URL。为识别过负荷状态, 过负荷子例程 ULR 配备一个存储单元区测量子例程 SEP-MR、一个信号通道测量子例程 D-MR、一个过负荷程序 UP 以及一个与通信接口有关的信号通道测量子例程 KAD-MR。过负荷状态通过第一级和第二级过负荷消息 $um_{1, 2}$ 特别通知由程序 IWU-S 实现的进程以控制激励通信终端设备 KE(S)。通过传输过负荷消息 $um_{1, 2}$ 将禁止显示信息向单个的、特别是过负荷的通信接口 KA 或者所有的通信接口发送, 或者禁止信号消息 s_i 向单个的或者特别是过负荷的通信接口 KA 发送。

图 2 到图 6 表示在测量子例程 D-MR、SEP-MR 和在过负荷程序 UP 中实现的方法。每一个流程图基本上自身已阐明, 并特别显示出为识别

- 过负荷要进行的关于测量微处理器进程满负荷 MPS、测量信号通道满负荷、测量存储单元区 SEP 的满负荷、测量与通信接口特定的信号通道的满负荷和一个为识别包括不同过负荷等级的过负荷的方法步骤。在图 6 中未表示用户接口模块 SLMO 向不满负荷状态的转变或恢复, 这里当微
- 5 处理器系统 MPS 和信号通道 D 在过负荷区间 ZVU 中并不固定为过负荷情况下建立一个由复位第一过负荷级产生的复位消息 rsm, 并将其通知由程序模块 IWU-S 实现的进程。与此类似, 在不存在存储单元区过负荷消息 SEPum 时, 建立一个由复位第二过负荷级产生的复位消息 rsm, 并将其通知所涉及的单元。如果不存在任何过负荷指示消息 um1、2,
- 10 因此也就没有任何用户接口模块的过负荷状态报告, 也就是说用户接口模块 SLMO 出于正常的操作状态。在图 6 中通过 rsm 的符号表示出这种关系。不过这里必须注意不同的处理过程(未示出)。

说明书附图

图 1



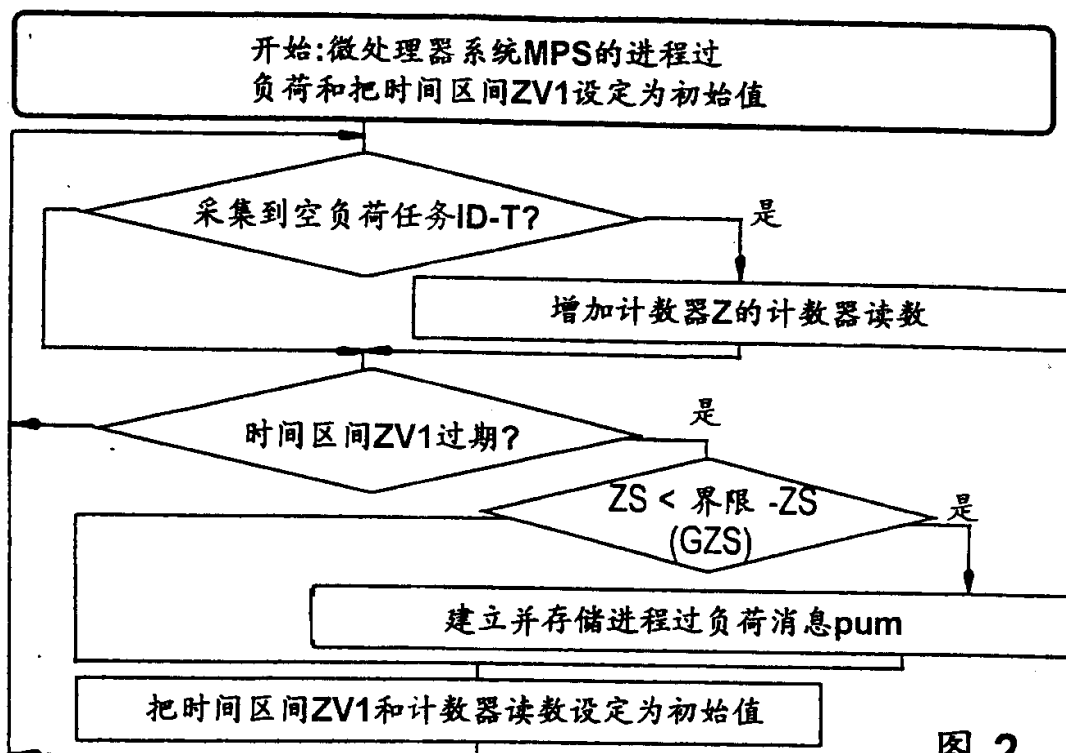


图 2

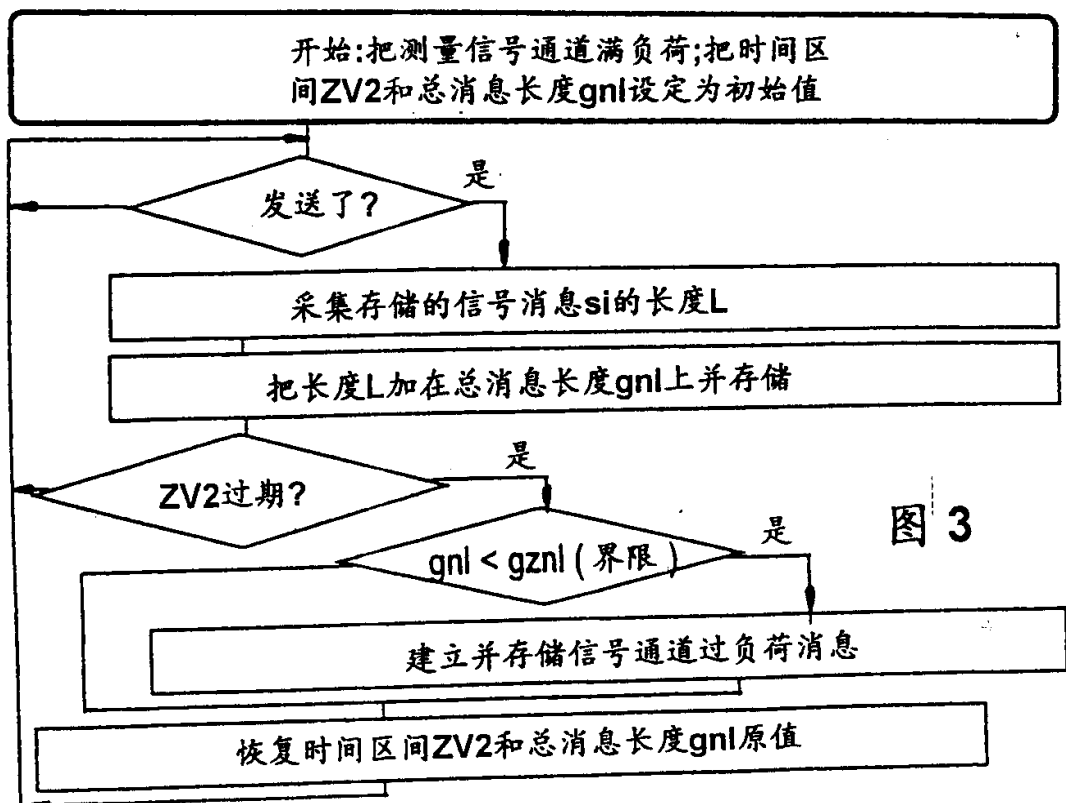


图 3

图 4

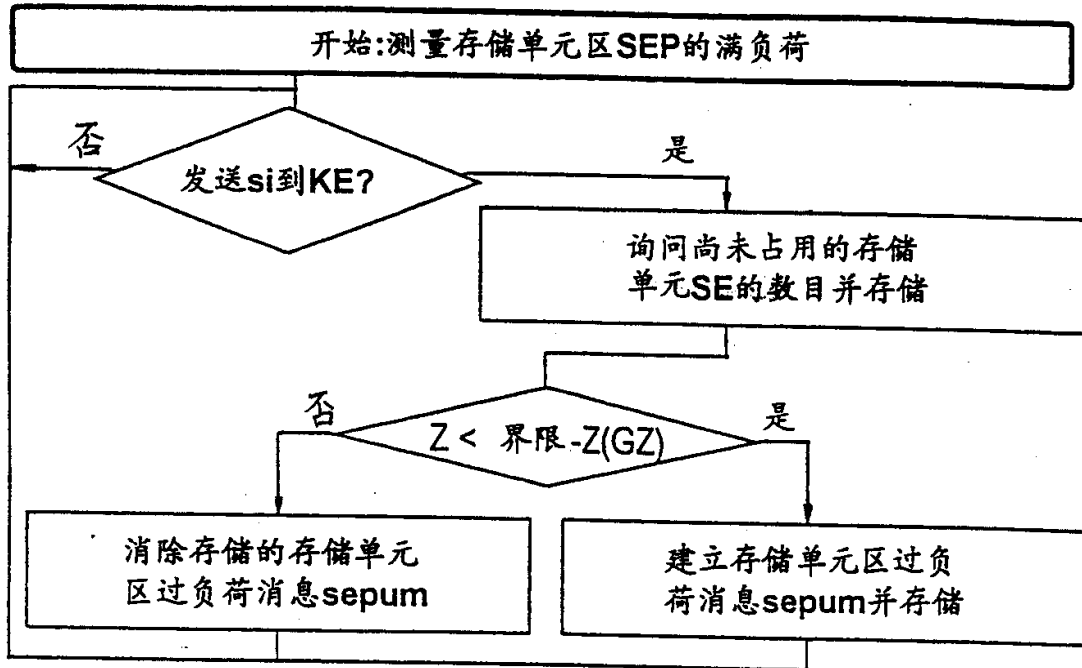


图 5

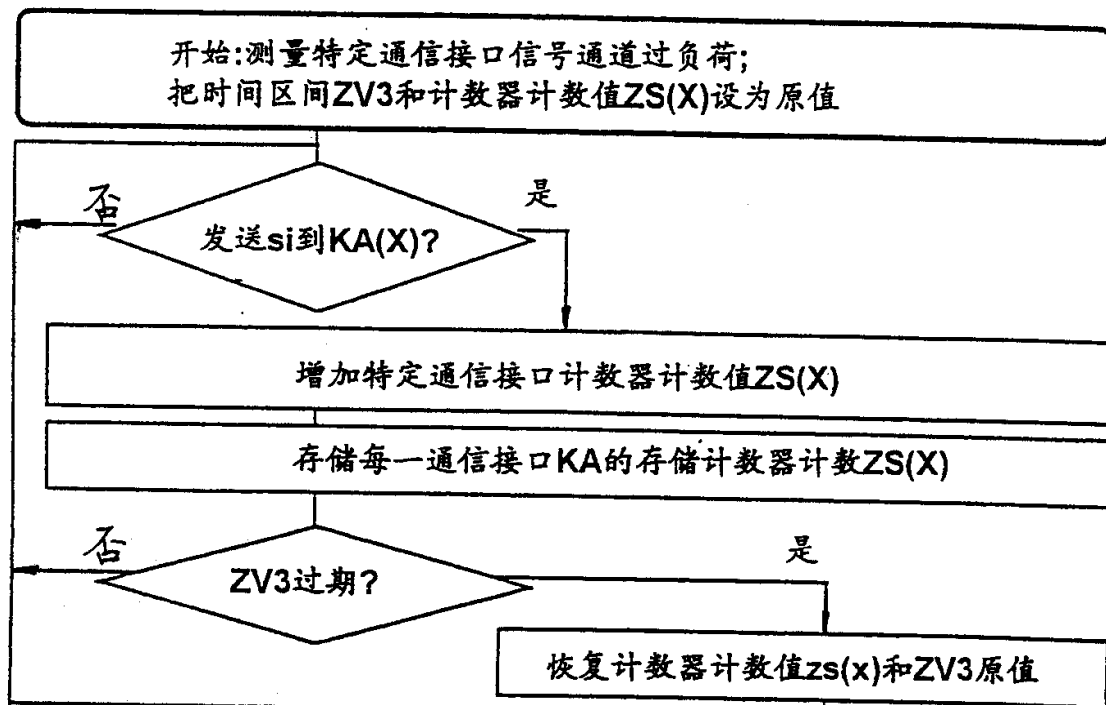


图 6

