

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/44 (2006.01)
G06F 11/36 (2006.01)



[12] 发明专利说明书

专利号 ZL 200710148283.4

[45] 授权公告日 2009年12月16日

[11] 授权公告号 CN 100570556C

[22] 申请日 2007.9.4

[21] 申请号 200710148283.4

[30] 优先权

[32] 2006.9.11 [33] JP [31] 2006-246177

[73] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 伊藤祐子 青山希 伊藤隆晴

有马俊道

[56] 参考文献

CN1474312A 2004.2.11

JP2002-328921A 2002.11.15

CN1452058A 2003.10.29

审查员 李楠

[74] 专利代理机构 中国国际贸易促进委员会专利
商标事务所

代理人 李颖

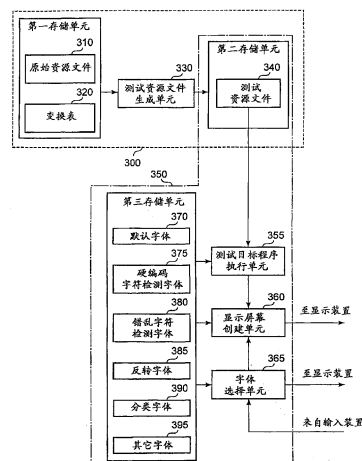
权利要求书 3 页 说明书 28 页 附图 13 页

[54] 发明名称

测试国际化软件的方法和设备

[57] 摘要

为国际化软件提供有效的测试方法。该方法包括：根据变换表，一对一地把第一语言的字符转换成第二语言的字符，模拟翻译用国际化软件的第一语言编写的具体化资源文件；利用为相应测试类别准备的多种字体之一，显示来自通过参阅模拟翻译的测试资源文件进行处理的国际化软件的输出信息。包括在输出信息中的字符代码被分成应包括在输出信息中的与第二语言的字符对应的字符代码以及不应包括在输出信息中的与第一语言的字符对应的字符代码及其它字符代码。当使字符形状与作为测试对象的一组字符代码及一组其它字符代码关联时，每种字体使用不同的关联规则，从而通过利用相关字体，在输出信息显示中，作为测试对象的字符代码和其它字符代码可相互区分。



1、一种由计算机执行的测试国际化软件的方法，所述国际化软件通过参阅具体化资源文件进行处理，所述方法包括下述步骤：

从用第一语言编写的具体化资源文件读取数据；

通过参阅具有一对一相互关联的第一语言的字符和第二语言的字符的变换表，把包括在所述数据中的第一语言的字符转换成第二语言的字符，从用第一语言编写的具体化资源文件产生用第二语言编写的测试资源文件；

执行国际化软件；和

通过使用为国际化软件的相应测试类别准备的多种字体之一，把从通过参阅作为具体化资源文件的测试资源文件进行处理的国际化软件输出的输出信息显示在屏幕上，

其中包括在输出信息中的字符代码被分成三组：与包括在变换表中的第二语言的字符对应的第二组的多个字符代码，这些字符代码应被包括在输出信息中；与第一语言的字符对应的第一组的多个字符代码，这些字符代码不应被包括在输出信息中；以及不与第一语言的字符或者包括在变换表中的第二语言的字符对应的第三组的多个字符代码，这些字符代码不应被包括在输出信息中，并且

其中当使字符形状与作为国际化软件的测试中的测试对象的一组多个字符代码以及规定字符编码方案中的一组其它字符代码相关联时，所述多种字体各使用不同的关联规则，从而在利用相关字体的输出信息的显示中，作为测试对象的字符代码和其它字符代码是相互区分的。

2、按照权利要求 1 所述的方法，其中利用多种字体之一显示输出信息的步骤包括利用用户从所述多种字体中选择的一种字体显示输出信息的步骤。

3、按照权利要求 1 所述的方法，其中利用多种字体之一显示输出信息的步骤包括通过把正在使用的字体转换成用户从所述多种字体

中选择的一种字体来显示输出信息的步骤。

4、按照权利要求 1 所述的方法，其中变换表具有与第二语言的可能导致显示问题的一些字符一对一关联的第一语言的一些字符。

5、按照权利要求 1 所述的方法，其中测试类别包括硬编码字符串的检测，并且在用于硬编码字符串的检测的字体中，第一组对应于作为测试对象的一组多个字符代码，第一组的字符代码均无修改地与对应于相关字符代码的第一语言的字符的字符形状相关联，并且规定字符编码方案中的其它字符代码均与相同的字符形状相关联。

6、按照权利要求 1 所述的方法，其中测试类别包括按照在第二语言中定义的字符的顺序的字符串分类结果的有效性的确定，并且在用于确定分类结果的有效性的字体中，第二组与作为测试对象的一组多个字符代码对应，第二组的字符代码均与指示为和相关字符代码对应的第二语言的字符定义的顺序的字符形状相关联，规定字符编码方案中的其它字符代码均与相同的字符形状相关联。

7、按照权利要求 1 所述的方法，其中测试类别包括第二语言特有的字符的可显示性的确定，在用于确定可显示性的字体中，第三组对应于作为测试对象的一组多个字符代码，并且第三组的字符代码均与相同的字符形状相关联。

8、按照权利要求 1 所述的方法，其中测试类别包括确认包括在用第一语言编写的具体化资源文件中的字符串的字符串确认，在用于字符串确认的字体中，第二组对应于作为测试对象的一组多个字符代码，并且第二组的字符代码均与第一语言的字符的字符形状相关联，第一语言的该字符具有与相关字符代码对应的第二语言的字符，作为当生成测试资源文件时的“变换为”字符。

9、一种测试通过参阅具体化资源文件进行处理的国际化软件的设备，包括：

第一存储单元，用于保存用第一语言编写的具体化资源文件以及一变换表，所述变换表具有一对一相互关联的第一语言的字符和第二语言的字符；

生成装置，用于通过参阅变换表，把第一语言的字符转换成第二语言的字符，从具体化资源文件生成用第二语言编写的测试资源文件；

第二存储单元，用于保存生成的测试资源文件；

执行国际化软件的装置；和

显示屏幕创建单元，用于创建其中利用为国际化软件的相应测试类别准备的多种字体之一，显示从通过参阅作为具体化资源文件的测试资源文件进行处理的国际化软件输出的输出信息的显示屏幕，

其中包括在输出信息中的字符代码被分成三组：与包括在变换表中的第二语言的字符对应的第二组的多个字符代码，这些字符代码应被包括在输出信息中；与第一语言的字符对应的第一组的多个字符代码，这些字符代码不应被包括在输出信息中；以及不与第一语言的字符或者包括在变换表中的第二语言的字符对应的第三组的多个字符代码，这些字符代码不应被包括在输出信息中，并且

其中当使字符形状与作为国际化软件的测试中的测试对象的一组多个字符代码以及规定字符编码方案中的一组其它字符代码相关联时，所述多种字体各使用不同的关联规则，从而在利用相关字体的输出信息的显示中，作为测试对象的字符代码和其它字符代码是相互区分的。

测试国际化软件的方法和设备

技术领域

本发明涉及测试在多种语言环境中执行的国际化软件的技术。更具体地说，本发明涉及从在与原始语言环境不同的语言环境中执行的国际化软件给用户的输出的测试技术，所述技术能够改进测试对象的可见性，从而能够实现有效测试。

背景技术

支持多种语言的软件被称为“国际化软件”。在国际化软件中，从源代码中分离出将显示在屏幕上的文本串部分，这样分离的文本串部分被保存为具体化的(externalized)资源文件。当执行国际化软件时，具体化的资源文件被参阅，并由国际化软件用于实现屏幕显示。当采用这样的配置时，国际化软件的开发人员只需要把具体化的资源文件翻译成另一种语言，以使国际化软件支持所述另一种语言。此外，一旦具体化的资源文件被翻译，即使执行码被改变，也不必重做翻译。

在这种国际化软件的开发过程中，对各种特性或方面(下面称为“测试类别”)进行测试，以了解国际化软件在不同国家的语言环境中是否正确地操作。其中，涉及给用户的输出的测试包括检测将向用户显示的消息或者其它字符串是否已从源代码中分离出来，并且已被正确地具体化为资源文件的测试。换句话说，这是一种硬编码字符串检测测试，该测试确认在源代码中没有直接描述(即，硬编码)将向用户显示的消息或其它字符串。另外还有就是分类结果(sort result)的确认测试，所述确认测试检查字符串的列表是否按照在每种语言中定义的顺序，例如字母表顺序被正确分类。另外进行的是语言特有字符的可显示性测试，所述可显示性测试检查每种语言特有的字符是否被正确显示，而不会错乱。

具体化的资源文件的翻译费时。从而，在开发过程中，不可能使用翻译后的具体化资源文件来测试上面所述的每种类别。为此，按照惯例开发了一种模拟翻译具体化资源文件，从而把其用作翻成目标语言的具体化资源文件的方法。在这种常规技术中，对于原始资源文件中的每个项目，在原始语言的字符串之前或之后添加很可能导致问题的翻译目标语言的相同字符，从而产生具体化资源文件的模拟翻译(参见专利文献1和2)。还有一种技术就是原始语言中的窄的或单宽度字母字符都被宽的或双宽度字母字符代替，从而产生模拟翻译(参见专利文献3)。在这些技术中，在通过参阅模拟翻译的资源文件进行处理的国际化软件的输出的屏幕显示上，确认对于原始语言的每个字符串，添加的字符或双宽度字母字符都被正确地显示。如果只显示原始语言的字符串，那么它是硬编码的字符串。此外，如果任意添加的字符或双宽度字母字符被另一字符替代，那么它是错乱的字符。

[专利文献1] 美国专利 No.6507812

[专利文献2] 美国专利 No.6453462

[专利文献3] 美国专利 No.6425123

发明内容

按照上面所述的常规技术的测试方法存在几个问题。问题之一是当具体化资源文件包括具有赋值部分的字符串时，不可能对赋值部分进行硬编码字符串的检测。这是因为在通过参阅模拟翻译的资源文件进行处理的国际化软件的输出的显示屏幕上，不能识别赋值部分。此外，在上面所述的常规技术中，在原始语言的字符串前/后添加翻译目标语言的相同字符。这意味着当对多个字符串分类时，理应根据原始语言的字符串部分评估分类结果，妨碍关于翻译目标语言的分类结果的有效性的确定。此外，为了对翻译目标语言特有的所有复杂字符进行语言特有字符的可显示性测试，理应必须检查至少十类字符。但是，在上述常规技术中，相同模式的字符组被添加在原始语言的每个字符串之前，从而，考虑到可见性，只能添加两类或三类的字符。为了覆

盖更多的字符类型，必须准备多个具体化资源文件。

鉴于上面所述，本申请的发明人对常规技术的模拟翻译进行了一些改进。第一种改进在于对于原始语言的资源文件中的每个项目，在原始语言的字符串的第一字符的前后，或者只在前面添加翻译目标语言的很可能导致问题的不同字符。第二种改进在于在包括在资源文件中的赋值（assignment）部分前后添加指示赋值部分的开始和结束的标记。借助这些改进，在通过参阅模拟翻译的资源文件进行处理的国际化软件的输出的显示屏幕上，在原始语言的字符串前面显示翻译目标语言的不同字符。因此，已能够利用改进的模拟翻译进行国际化软件的显示屏幕上的分类结果的确认测试，而利用上述常规技术这是不可能的。此外，借助改进的模拟翻译，可向原始语言的字符串的不同第一字符添加任意长度的翻译目标语言的不同字符。从而，在改进的模拟翻译中，已能够利用单一的资源文件对大量的语言特有字符进行可显示性测试。此外，在利用改进模拟翻译的国际化软件的屏幕显示上，指示赋值部分的开始和结束的标记被显示在赋值部分前后。从而，赋值部分变成可区别的，因此，在改进的模拟翻译中，已能够恰当地对赋值部分进行硬编码字符串的检测。

但是，常规技术的改进测试方法仍然存在几个问题。首先，在常规技术的改进测试方法中，尽管可以识别赋值部分，不过仍然存在可能忽略硬编码字符串的可能性。此外，在常规技术的改进测试方法中，尽管可以确认作为分类的结果，字符串的列表被分类，不过对翻译目标语言一无所知的测试人员不能确定分类结果是否正确。对于日语或者具有许多种字符的其它语言，即使知晓翻译目标语言的测试人员也不能立即确定分类结果是否正确。此外，就常规技术的改进方法来说，类型数目增多的翻译目标语言的字符被显示在屏幕上，使得测试人员不能立即确定在显示在屏幕上的字符中是否存在错乱字符。

此外，在常规技术的改进测试方法中，试图在通过参阅模拟翻译的资源文件进行处理的国际化软件的单个显示屏幕上检查所有的测试类别，导致测试对象的可见性较差。但是，如果准备适合于测试类别

的相应内容的不同具体化资源文件，又必须把国际化软件重启与测试类别的数目对应的次数，这阻碍了高效的测试。

鉴于上面所述，本发明的一个目的是提供一种测试对象的可见性得到改进，并且不需要重启国际化软件的有效测试国际化软件的方法、设备和程序，以及提供用于所述测试的字体的数据结构。本发明的另一目的是确保在国际化软件的测试中，无遗漏地检测硬编码字符串。本发明的另一目的是确保在国际化软件的测试中，当对多个字符串分类时，能够做出分类结果是否正确的确定，而不管是否具备翻译目标语言的知识。本发明的另一目的是在国际化软件的测试中，使关于是否存在错乱字符的确定变得容易。

为了实现上面目的，本发明提供一种如下所述由计算机执行的测试国际化软件的方法。这里，国际化软件指的是通过参阅具体化资源文件进行处理，并且在多语言环境中执行的软件。测试国际化软件的方法包括下述步骤：从用第一语言编写的具体化资源文件读取数据；通过参阅具有一对一相互关联的第一语言的字符和第二语言的字符的变换表，把包括在所述数据中的第一语言的字符转换成第二语言的字符，从用第一语言编写的具体化资源文件产生用第二语言编写的测试资源文件；执行国际化软件；并通过使用为国际化软件的相应测试类别准备的多种字体之一，把从通过参阅作为具体化资源文件的测试资源文件进行处理的国际化软件输出的输出信息显示在屏幕上。

这里，包括在输出信息中的字符代码可被分成三组：与包括在变换表中的第二语言的字符对应的第二组的多个字符代码，这些字符代码应被包括在输出信息中；与第一语言的字符对应的第一组的多个字符代码，这些字符代码不应被包括在输出信息中；以及不与第一语言的字符或者包括在变换表中的第二语言的字符对应的第三组的多个字符代码，这些字符代码不应被包括在输出信息中。此外，当使字符形状与作为国际化软件的测试中的测试对象的一组多个字符代码以及规定字符编码方案中的一组其它字符代码相关联时，采用规定编码方案的多种字体分别使用不同的关联规则，从而在利用相关字体的输出信

息的显示中，作为测试对象的字符代码和其它字符代码是可相互区分开的。

于是，按照本发明，按照可与其它字符代码区分开的方式显示作为测试对象的字符代码，这改进了测试对象的可见性。虽然对于每种测试类别来说，测试对象不同，不过按照本发明，为相应测试类别准备多种字体，以致仅仅通过利用对应的字体，就能够提供适合于每种测试类别的测试的显示屏幕，而不需要重启国际化软件。

最好，利用多种字体之一显示输出信息的步骤包括利用用户从所述多种字体中选择的一种字体显示输出信息的步骤。另一方面，上述步骤包括通过把正在使用的字体转换成用户从所述多种字体中选择的一种字体来显示输出信息的步骤。此外，变换表最好具有与第二语言的可能导致显示问题的一些字符一对一关联的第一语言的一些字符。

此外，测试类别最好包括硬编码字符串的检测。在用于硬编码字符串的检测的字体中，第一组对应于作为测试对象的一组多个字符代码，第一组的字符代码均无修改地与对应于相关字符代码的第一语言的字符的字符形状相关联。最好，上述字体使规定字符编码方案中的每个其它字符代码均与相同的字符形状相关联。相同的字符形状最好是不太经常使用的符号，或者不明显的符号，比如中空的方块符号。

此外，测试类别最好包括按照在第二语言中定义的字符的顺序的字符串的分类结果的有效性。在用于确定分类结果的有效性的字体中，第二组与作为测试对象的一组多个字符代码对应，并且第二组的字符代码均与指示关于和相关字符代码对应的第二语言的字符定义的顺序的字符形状相关联。最好，上述字体使规定字符编码方案中的每个其它字符代码均与相同的字符形状相关联。相同的字符形状最好是不太经常使用的符号，或者不明显的符号，比如中空的方块符号。

此外，测试类别最好包括第二语言特有的字符的可显示性。在用于确定可显示性的字体中，第三组对应于作为测试对象的一组多个字符代码，并且第三组的字符代码均与相同的字符形状相关联。相同的字符形状最好是不太经常使用的符号，或者明显的符号。

此外，测试类别最好包括确认包括在用第一语言编写的具体化资源文件中的字符串的字符串确认。在用于字符串确认的字体中，第二组对应于作为测试对象的一组多个字符代码，并且第二组的字符代码均与第一语言的字符的字符形状相关联，第一语言的该字符具有与相关字符代码对应的第二语言的字符，作为当生成测试资源文件时的“转换为(converted-to)”字符。在上述测试类别中，第一组还对应于作为测试对象的一组多个字符代码，并且第一组的字符代码均无修改地和与相关字符代码对应的第一语言的字符的字符形状相关联。

按照本发明的另一方面，提供一种在国际化软件的测试中使用的采用规定字符编码方案的字体的数据结构，所述国际化软件参阅通过根据变换表，把包括在用第一语言编写的具体化资源文件中的第一字符的所有字符一对一地转换成第二语言的字符产生的测试资源文件。所述数据结构包括：作为测试中的测试对象的多个第一字符代码；按照第一规则一对一地与多个第一字符代码关联的多项第一字符形状信息；作为规定字符编码方案中的其它字符代码的多个第二字符代码；和按照第二规则一对一地与多个第二字符代码关联的多项第二字符形状信息。

这里，所述多个第一字符代码对应于和第一语言的字符对应的多个字符代码，与包括在变换表中的第二语言的字符对应的多个字符代码，和与第一语言的字符或者包括在变换表中的第二语言的字符对应的多个字符代码中的一个。当从通过参阅测试资源文件进行处理的国际化软件的输出中检测出第一字符代码时，从字体数据中取回与相关第一字符代码对应的第一字符形状信息，并把其用于屏幕显示。当从输出中检测出第二字符代码时，从字体数据中取回与相关第二字符代码对应的第二字符形状信息，并把其用于屏幕显示。

虽然上面把本发明说明为测试国际化软件的方法，不过也可把本发明理解为设备、程序或存储所述程序的存储介质。此外，结合测试国际化软件的方法说明的特征适合于说明为本发明的另一方面的用于测试国际化软件的测试的字体的数据结构。此外，说明为字体的数据

结构的本发明也可被理解为存储字体的相关数据结构的记录介质。

按照本发明，可在改进测试对象的可见性，并且不需要对每种测试类别重启国际化软件的情况下，对在不同于原始语言的语言环境中执行的国际化软件给用户的输出进行有效测试。在本发明的实施例的下述描述中将说明其它优点。

附图说明

图 1A 表示测试对象程序的具体化资源文件的一个例子，图 1B 表示测试对象程序的一部分源代码的例子，图 1C 表示测试对象程序的输出屏幕的例子；

图 2A 表示基于常规技术的模拟翻译资源文件的例子，图 2B 表示测试对象程序的输出屏幕的例子；

图 3 表示按照本发明的实施例的创建测试资源文件的设备，和按照本发明的实施例的测试测试对象程序的设备的配置例子；

图 4A 表示测试对象程序的具体化资源文件的例子，图 4B 表示按照本发明的实施例的变换表的例子，图 4C 表示按照本发明的实施例的测试资源文件的例子；

图 5 表示图解说明按照本发明的实施例的创建测试资源文件的设备的处理流程的例子的流程图；

图 6 表示利用常规字体显示测试对象程序的执行结果的显示屏幕的例子；

图 7A 表示按照本发明的实施例的硬编码字符检测字体的例子，图 7B 表示利用硬编码字符检测字体显示测试对象程序的处理结果的显示屏幕的例子；

图 8A 表示按照本发明的实施例的错乱字符检测字体的例子，图 8B 表示利用错乱字符检测字体显示测试对象程序的处理结果的显示屏幕的例子；

图 9A 表示按照本发明的实施例的反转字体的例子，图 9B 表示利用反转字体显示测试对象程序的处理结果的显示屏幕的例子；

图 10A 表示按照本发明的实施例的分类字体的例子，图 10B 表示利用分类字体显示测试对象程序的处理结果的显示屏幕的例子；

图 11A 表示图解说说明按照本发明的实施例的测试测试对象程序的设备的处理流程的例子的流程图，图 11B 表示图解说说明利用指定的字体，创建处理结果的显示屏幕的处理流程的例子的流程图；

图 12 表示按照本发明的实施例的整个系统的实现例子；

图 13 表示图解说说明图 12 中所示的整个系统的处理流程的例子的流程图；

图 14 表示显示的用于选择待使用字体的对话框的例子；

图 15 表示按照本发明的实施例的创建测试资源文件的设备和按照本发明的实施例的测试测试对象程序的设备的硬件配置的例子。

具体实施方式

虽然下面将参考附图详细说明实现本发明的最佳方式，不过下面的实施例并不限制在请求专利保护的权利要求中描述的本发明。此外，在实施例中讨论的特征的所有组合不一定是本发明的解决手段所必不可少的。在实施例的说明中，相同的部件用相同的附图标记表示。

在说明本发明之前，参考图 1A、1B、1C、2A 和 2B 说明国际化软件的具体化资源文件，改进的模拟翻译方法及其问题。下面，国际化软件被称为“待测试程序”或者“测试对象程序”，具体化资源文件被简称为“资源文件”。作为资源文件的一个例子，图 1A 表示原始语言为英语的测试对象程序的资源文件 100。如附图中所示，资源文件包括多个项目，每个项目由关键字(key)和赋予该关键字的字符串组成。当测试对象程序在执行的时候遇到某个关键字时，它搜索资源文件寻找相关关键字，并读取赋予该关键字的字符串的值。下面利用图 1B 中所示的源代码 105 和图 1C 中所示的测试对象程序的输出屏幕 110 更详细地说明该操作。

包括在源代码 105 中的命令“showMessage()”是在屏幕上显示通过把第二个变元赋予包括在第一个变元的字符串中的变量而获得的字

字符串的命令。具体地说，当执行“showMessage()”命令时，测试对象程序从资源文件 100 取回作为第一个变元的关键字“MSG01”，并读取对应的字符串值“The status is {0}.”。测试对象程序随后把作为第二个变元接收的“Green”赋予读取的字符串值的{0}，从而最终在屏幕上显示“The status is Green.”，如图 1C 中在 115 中所示。注意包括在赋予关键字 MSG01 的字符串中的{0}是“赋值部分”，赋予{0}的变量的值在执行时确定。

现在，参考图 2A 和 2B 说明改进的模拟翻译方法及其问题。如上所述，在改进的模拟翻译中，对于原始语言的资源文件中的每个项目，在原始语言的字符串的第一个字符前后添加很可能导致问题的翻译目标语言的不同字符。此外，在包括在资源文件中的赋值部分前后添加指示“开始”和“结束”的标记。图 2A 中所示的模拟翻译的资源文件 200 对应于按照改进技术模拟翻译的图 1A 中所示的资源文件 100。这里，翻译目标语言是日语。

从图 2A 可看出，在本例中，作为很可能导致问题的字符，字符“竹”

和

“金”

以及字符

“講”

和

“愛”

被选择并被分别添加在字母表的第一个字符“T”和“G”的前后。对于其它项目的第一个字符，选择并在不同的字母表前后添加不同的日语字符。每个项目的第一个字符被放入“()”中，以便识别第一个字符。此外，在赋值部分{0}前后添加字符

“開始”

和

“終了”

图 2B 表示当被读取以便执行模拟翻译的测试对象程序执行图 1B 中所示的源代码时显示的屏幕 205。注视在字段 210 中显示的字符串“Green”，字符

“開始”

和

“終了”

被显示在“Green”前后，没有日语字符被添加在第一个字符“G”前后，从而知道该字符串是硬编码的赋值字符串。列表框 215 中的三个字符串的第一个字符都具有在前后添加的不同日语字符，从而知道通过执行分类命令，可能实现分类结果的确认测试。此外，在显示在按钮 225 上的字符串“Refresh the Status”的第一个字符“R”前后显示的日语字符不同于初始添加的日语字符，从而知道在相关部分中存在字符

“ソ”

和

“十”

的显示方面的问题。

但是，利用这种改进的模拟翻译的测试方法仍然存在如上所述的几个问题。一个问题在于尽管赋值部分可被识别，不过硬编码字符串可能被忽略。至于具有赋值部分的字符串，不仅具有赋予其的字符串的原始字符串，而且赋值部分中的字符串都必须被确认是否是硬编码字符串。如上所述根据字符串的第一个字符进行硬编码字符串的确认。这意味着当赋值部分位于字符串的中间时，图 2B 的字段 210 中的字符串的情况就是这样，对于未被置于字符串的开始的赋值部分来说，硬编码字符串的确认是不可能的。

另一个问题在于即使能够确认作为分类的结果，字符串的列表被重新排列，也难以确定分类是否正确。不仅对于对翻译目标语言一无所知的测试人员，而且对于知晓翻译目标语言的测试人员来说，都难以立即确定是否对具有大量字符的语言（例如日语）正确地进行了分类。另一个问题在于在利用改进的模拟翻译的测试方法中，翻译目标

语言的许多种字符被显示在屏幕上，妨碍立即确定在显示的字符中是否存在错乱的字符。例如，不可能通过单独查看图 2 中的显示屏幕 205，识别错乱的字符。

此外，就常规技术的改进测试方法来说，显示屏幕上测试对象的可见性较低，因为通过参考模拟翻译的具体化资源文件进行处理的国际化软件的单一显示屏幕被用于试图同时确认结果。另一方面，如果为每个测试类别准备一个适合于测试内容的具体体资源文件，那么必须把国际化软件重新启动与测试类别的数目对应的次数，阻碍了高效的测试。

鉴于上面所述，本发明通过使用新开发的测试资源文件和为相应的测试类别准备的多种字体，解决了上述问题。更具体地说，通过利用为相应的测试类别准备的多种字体，显示来自通过参考资源文件执行处理的测试对象程序的输出，在只为测试准备一个模拟翻译的资源文件的同时，本发明使得能够为每种测试类别提供适合于测试的显示屏幕。在适当考虑将与字符代码关联的字符形状的同时，通过为相应的测试类别准备多种字体，并使作为测试对象的字符代码可与不是测试对象的字符代码区分开，解决了相应的测试类别特有的问题。

首先，参考图 3、4A、4B、4C、5A 和 5B 说明按照本发明的翻译资源文件的新方法。随后，参考图 3、7A、7B、8A、8B、9A、9B、10A、10B、11A 和 11B 说明利用模拟翻译的资源文件和为相应的测试类别准备的多种字体测试测试对象程序的方法。

图 3 中用虚线划界的部分是按照本发明的实施例，创建测试资源文件的设备 300 的功能方框图。设备 300 包括：第一存储单元，用于保存原始资源文件 310 和变换表 320；测试资源文件生成单元 330；和第二存储单元，用于保存测试资源文件 340。注意第一和第二存储单元物理上可以是单个存储装置。此外，虽然在本实施例中，Unicode 被用作规定的字符编码方案，不过本发明并不局限于此。当然可以采用任何其它字符编码方案。这里使用的“字符代码”是与字符关联的代码，在 Unicode 中被称为“码点”（code point）。

原始资源文件 310 是测试对象程序在执行期间参阅的原始资源文件。作为原始资源文件 310 的一个例子，图 4A 表示原始语言为英语的资源文件 400。在本实施例中，翻译目标语言是日语。变换表 320 是其中作为第一种语言的原始语言的字符和作为第二种语言的翻译目标语言的字符一一对应关联的表格。为了能够实现这样的一对一映射，要求在翻译目标语言中使用的字符的字符代码的范围大于在原始语言中使用的字符的字符代码的范围。例如，英语(ASCII 码)使用字符代码的最小范围，从而可选择日语、汉语、韩语或法语作为翻译目标语言。相反，当法语是原始语言时，尽管可以选择日语或汉语作为翻译目标语言，不过不可能选择英语。最好，在变换表 320 中，原始语言的一些字符一对一地与翻译目标语言的很可能导致显示问题的字符关联。

这里，关于日语字符举例说明很可能导致显示问题的字符。首先，日语字符

"表",

"噂"

和

"ソ"

都具有其中第二个字节与在英语环境中用作换行等的控制代码的位模式对应的字符代码，从而在使用的某类字符编码方案中可能不被正确显示。此外，在使用的某类字符编码方案中，包括

"鯨"

在内的几个日语字符可能被转换成不同的字符。例如，虽然

"鯨"

的字符代码在 Unicode 中为 U+9BF5，不过在称为“MS932”的字符编码方案中，它被转换成 0X88B1，而在称为“CCSID 942”的字符编码方案中，它被转换成 0XE9CB。甚至除已知可能导致问题的那些字符之外的字符也可能不被正确显示。因此，在变换表 320 中，可取的是原始语言的一些字符一对一地与翻译目标语言的很可能导致问题的字符

关联，以致变换表 320 既包括正常的字符，又包括很可能导致显示问题的字符。

作为变换表 320 的一个例子，图 4B 表示了变换表 405，其中英语字符一对一地与日语字符关联。在变换表 405 中，左侧的两列中是英语字符和赋予其的字符代码，右侧的两列中是对应的日语字符和赋予其的字符代码。变换表 320 定义在原始语言中使用的字符的对应性，除了元字符之外，所述元字符是当利用规则表达式创建图案时使用的符号。从变换表 405 可看出，在 Unicode 中，除元字符之外的英语字符被赋予从 U+0020 ~ U+007E 的字符代码。这里，英语大写字母“S”、“T”和“U”分别与很可能导致问题的

“表”，

“嗜”

和

“ソ”，

关联。

测试资源文件生成单元 330 从第一存储单元读取原始资源文件 310 和变换表 320，并根据变换表 320 把第一种语言的字符转换成第二种语言的字符，从而根据用第一种语言编写的测试对象程序的资源文件 310 产生用第二种语言编写的测试资源文件 340。这样产生的测试资源文件 340 被保存在第二存储单元中，作为测试对象程序的资源文件。图 4C 表示通过根据变换表 405 把英语字符转换成日语字符，产生自原始资源文件 400 的测试资源文件 410。从该图可看出，测试资源文件 410 不包括原始语言（英语）的任何字符。

下面参考图 5A 和 5B 中所示的流程图说明按照本实施例的设备 300 的处理流程的例子。图 5A 是图解说明本实施例的设备的处理流程的概略的流程图。设备 300 首先从第一存储单元读取原始资源文件 310(步骤 500)。设备 300 随后根据变换表 320 把第一种语言的字符转换成第二种语言的字符，从而根据测试对象程序的用第一种语言编写的原始资源文件 310 生成用第二种语言编写的测试资源文件 340(步骤

505)。最后，设备 300 把生成的测试资源文件 340 保存到第二存储单元中(步骤 510)。

现在参考图 5B 详细说明步骤 505 中的处理。首先，作为预处理，设备 300 从第一存储单元读取变换表 320，并从第二存储单元读取空的测试资源文件 340。之后，开始主处理，测试资源文件生成单元 330 确定原始资源文件 310 是否包括任何可读的项目(步骤 515)。如果不存在可读的项目(步骤 515: NO)，那么处理被终止。如果确定存在可读的项目(步骤 515: YES)，那么测试资源文件生成单元 330 从原始资源文件 310 读取一个项目(步骤 520)。测试资源文件生成单元 330 从读取的项目中提取关键字，并将其赋给一个变量 Key(步骤 525)，并把变量 Key 写入测试资源文件 340 中(步骤 530)。

随后，测试资源文件生成单元 330 从读取的项目中提取字符串值，并将其赋给变量 Text(步骤 535)，另外从变量 Text 读取一个字符并将其赋给变量 X(步骤 540)。为了确认赋值部分，测试资源文件生成单元 330 确认变量 X 的值，如果是数字字符，那么它确认变量 X 前后的值是否分别对应于“{”和“}”(步骤 545)。如果变量 X 的值不是指示赋值部分的数字字符(步骤 545: NO)，那么测试资源文件生成单元 330 参阅变换表 320，以转换变量 X 的字符，并把转换后的字符赋给变量 Y(步骤 550)。如果变量 X 的值是指示赋值部分的数字字符(步骤 545: YES)，那么测试资源文件生成单元 330 把变量 X 的值原样赋给变量 Y(步骤 555)。假定变换表 320 这样定义，以致字符“{”和“}”在转换之后变成相同的字符。之后，测试资源文件生成单元 330 把变量 Y 写入测试资源文件 340(步骤 560)。随后确认先前从变量 Text 读取的字符是否是最后的字符(步骤 565)。如果不是最后的字符(步骤 565: NO)，那么处理返回步骤 540。如果是最后的字符(步骤 565: YES)，那么处理返回步骤 515。

返回图 3，图 3 中由点划线划界的部分表示按照本发明的实施例，利用模拟翻译的资源文件和为相应的测试类别准备的多种字体测试测试对象程序的设备 350 的功能方框图。这里，为了便于说明，单独说

明创建测试资源文件的设备 300 和测试测试对象程序的设备 350。实际上,可以用单台计算机实现设备 350 的功能和设备 300 的功能。设备 350 包括:保存测试资源文件 340 的第二存储单元,测试对象程序执行单元 355,显示屏幕创建单元 360,字体选择单元 365,和包括为相应的测试类别准备的多种字体的第三存储单元。包括在第三存储单元中的多个字体包括:默认字体 370,硬编码字符检测字体 375,错乱字符检测字体 380,反转字体 385,分类字体 390,和另一字体 395。注意第二存储单元和第三存储单元物理上可以是相同的存储装置。

如上所述,保存在第二存储单元中的测试资源文件 340 是用作为第二种语言的翻译目标语言编写的资源文件。测试资源文件 340 作为测试对象程序的资源文件被保存在第二存储单元中。测试对象程序执行单元 355 执行测试对象程序。当执行测试对象程序时,测试对象程序执行单元 355 首先从第二存储单元读取测试资源文件 340,之后,参阅测试资源文件 340 以执行测试对象程序的处理,并输出处理结果。

显示屏幕创建单元 360 从测试对象程序接收处理结果,并利用包括在第三存储单元中的默认字体创建处理结果的显示屏幕。此时,显示屏幕创建单元 360 把从测试对象程序接收的处理结果临时保存在缓冲器中。创建的显示屏幕被传送给显示装置以便显示。当从字体选择单元 365 收到字体的指定时,显示屏幕创建单元 360 从缓冲器读取测试对象程序的处理结果,并利用指定的字体重新创建处理结果的显示屏幕。创建的显示屏幕被再次传送给显示装置以便显示。

当通过诸如鼠标、键盘之类的输入装置从用户收到待使用的字体的选择请求时,字体选择单元 365 查阅第三存储单元,以确认可用的字体,并创建字体选择对话框。字体选择单元 365 把创建的对话框传送给显示装置。当从用户收到待使用的字体的指定时,字体选择单元 365 把指定的字体类型通知给显示屏幕创建单元 360。

现在,在说明按照本发明的实施例为相应的测试类别准备的多种字体之前,将说明利用标准或常规字体显示的测试对象程序的处理结果。当利用常规字体显示通过参阅图 4C 中所示的测试资源文件 410

进行处理的测试对象程序的处理结果时，得到如图 6 中所示的显示屏幕 660。注意图 1C 中所示的显示屏幕 110 是通过参阅原始资源文件 100 进行处理的测试对象程序的处理结果的显示屏幕。如上所述，测试资源文件 410 不包括任何英语字符，从而，推想没有任何英语字符会被显示在输出的显示屏幕上。因此，在图 6 中的字段 665 中显示的英语字符串“Green”是硬编码字符串。但是在显示屏幕 660 中，既存在日语字符又存在英语字符，使得测试人员难以立即检测硬编码的字符串。

此外，显示在图 6 中的按钮 680 上的字符串不同于测试资源文件 410 中的 MSG02 的字符串，这意味着它具有错乱的字符。但是，由于存在显示在显示屏幕 660 上的多种字符，因此仅仅通过查看显示屏幕 660，测试人员不可能确定显示在按钮 680 上的字符串包括错乱的字符。此外，显示在图 6 中的列表框 670 上的三个字符串包括各种日语字符，比如平假名字母，汉语字母，片假名字母和符号，因此，当测试人员进行分类时，测试人员不可能立即确定分类结果是否正确。如上所述，当按照本发明的实施例的测试资源文件和常规字体一起使用时，不可能为每种测试类别提供适合于测试的屏幕显示。

下面，说明按照本发明的实施例为相应的测试类别准备的多种字体。首先，说明用于检测硬编码字符串的硬编码字符检测字体 375。在硬编码字符串的检测中，与作为第一种语言的原始语言的字符对应的多个字符代码对应于作为测试对象（或者说检测对象）的多个字符代码。由于通过根据变换表 320 把包括在原始资源文件 310 中的原始语言的所有字符转换成翻译目标语言的字符，创建按照本发明的测试资源文件 340，因此可认为包括在通过参阅测试资源文件 340 进行处理的测试对象程序的输出中的原始语言的任何字符应是用源代码硬编码的字符，或者来源于不是测试对象的另一来源。与包括在变换表 320 中的作为第二种语言的翻译目标语言的字符对应的字符代码，和不与原始语言的字符或者包括在变换表 320 中的翻译目标语言的字符对应的字符代码对应于不是测试对象的多个字符代码。

在硬编码字符检测字体 375 中，不是测试对象的所有字符代码都与相同的字符形状关联，以便提高作为测试对象的原始语言的字符在屏幕上的可见性。相同的字符形状最好是不显眼的字符形状，比如中空的方块或空白，不太经常使用的字符形状，或者既不显眼又不经常使用的字符形状。另一方面，作为测试对象的原始语言的字符代码无修改地与和相关字符代码对应的原始语言的字符的字符形状关联。

作为硬编码字符检测字体 375 的一个例子，图 7A 表示在利用图 4B 中所示的以英语作为原始语言，以日语作为翻译目标语言的变换表 405 创建测试资源文件的情况下的硬编码字符检测字体的表 700。表 700 的最左侧一列中是字符代码，最右侧一列中是赋予对应的字符代码的硬编码字符检测字体中的字形。注意字形指的是赋予字符代码的字符形状。常规字体的字形添加在表 700 的中央一列中以便参考。在表 700 中，字符代码 U+3000 ~ U+FFEA 对应于包括在变换表 405 中的日语字符。如上所述，由于它们是不作为测试对象的字符代码，因此这里它们均被赋予一个中空的方块字符形状。在表 700 中，字符代码 U+0020 ~ U+007E 对应于英语字符。如上所述，它们是作为测试对象的字符代码，从而它们都被赋予原始字符的字符形状。

表 700 中的其它字符代码对应于 Unicode 的多个其它字符代码。由于它们不是如上所述的测试对象，因此它们都被赋予一个中空的方块字符形状，和上面说明的字符代码一样。应注意表 700 中的行数并不指示字符代码的实际数目。图 7B 表示在利用硬编码字符检测字体的表 700，显示通过参阅图 4C 中所示的测试资源文件 410 进行处理测试对象程序的处理结果的情况下的显示屏幕 705。在图 7B 中，除示于字段 710 中的硬编码字符串“Green”外的字符都被显示成中空的方块符号。因而，当使用硬编码字符检测字体 375 时，不是测试对象的字符都以相同的字符形状显示，这使测试人员可以容易地检测硬编码字符，而不受不是检测对象的字符(包括翻译目标语言的那些非检测对象字符)误导。

下面说明错乱字符检测字体 380，错乱字符检测字体 380 用于确

定作为第二种语言的翻译目标语言特有的字符的可显示性。在可显示性的确定中，与原始语言的字符或包括在变换表 320 中的翻译目标语言的字符不对应的字符代码对应于作为测试对象（或者说检测对象）的多个字符代码。由于通过根据变换表 320，把包括在原始资源文件 310 中的原始语言的所有字符转换成翻译目标语言的字符来创建本发明的测试资源文件 340，因此将包括在已读取测试资源文件 340 的测试对象程序的输出中的字符应是包括在变换表 320 中的翻译目标语言的字符。然而，如上结合硬编码字符检测字体 375 所述，所述输出可能包括以源代码硬编码的、或者来源于除测试对象外的另一资源的原始语言的字符。原始语言的相关字符不是作为错乱的结果而显示的那些字符。于是，如果输出包括除包含在变换表 320 中的翻译目标语言的字符和原始语言的字符之外的任何字符，那么它必定是不应该被显示的字符，即，错乱的字符。

因而，与原始语言的字符对应的字符代码和与包括在变换表中的翻译目标语言的字符对应的字符代码对应于不是测试对象的多个字符代码。在错乱字符检测字体 380 中，作为测试对象的字符代码都与相同的字符形状关联，以便提高作为测试对象的错乱字符在屏幕上的可见性。相同的字符形状最好是明显的字符形状，比如邮政标记(postal mark)，或者不太经常使用的明显的字符形状。另一方面，不是测试对象的字符代码都与相同的字符形状关联，该相同的字符形状不同于与作为测试对象的字符代码关联的字符形状。该相同的字符形状最好是不明显的字符形状，比如中空的方块或者空白。

作为错乱字符检测字体 380 的一个例子，图 8A 表示在以英语作为原始语言，以日语作为翻译目标语言，根据图 4B 中所示的变换表 405 产生测试资源文件的情况下的错乱字符检测字体的表 800。表 800 的最左侧一列中的是错乱字符检测字体 380 的字符代码，最右侧一列中的是赋予对应的字符代码的字形。此外，在中间一列中，是添加的供参考的常规字体中的字形。在表 800 中，字符代码 U+3000 ~ U+FFEA 是与包括在变换表 405 中的日语字符对应的那些字符代码。如上所述，

它们是不作为测试对象的字符代码，从而，它们均被赋予一个中空的方块字符形状。

在表 800 中，字符代码 U+0020 ~ U+007E 是与英语字符对应的那些字符代码。如上所述，它们也是不作为测试对象的字符代码，从而，它们均被赋予一个中空的方块字符形状。表 800 中的其它字符代码是 Unicode 中的多个其它字符代码。由于如上所述，它们是作为测试对象的字符代码，因此在本例中，它们均被赋予邮政标记的字符形状。应注意表 800 的行数并不指示字符代码的实际数目。

图 8B 表示在利用错乱字符检测字体 380 显示通过参阅图 4C 中所示的测试资源文件 410 进行处理的测试对象程序的处理结果的情况下的显示屏幕 810。在图 8B 中，邮政标记被显示在按钮 830 上，使得能够立即明白错乱的字符被显示于其上。因而，当使用错乱字符检测字体 380 时，以规定的字符形状显示错乱的字符，从而使测试人员可以容易地检测错乱的字符，即使翻译目标语言具有许多种字符。此外，用相同的不明显的字符形状显示不是测试对象的字符，比如在列表框 820 中显示的那些字符，以致测试人员能够容易地检测错乱的字符，而不受除检测对象之外的字符（例如正确显示的翻译目标语言的字符）误导。

下面说明用于包括在以第一种语言编写的资源文件中的字符串的确认的反转字体 385。在原始字符串的确认中，与原始语言的字符对应的字符代码和与包括在变换表 320 中的翻译目标语言的字符对应的字符代码对应于作为测试对象（或者说检测对象）的多个字符代码。由于通过根据变换表 320，把包括在原始资源文件 310 中的原始语言的所有字符转换成翻译目标语言的字符来创建本发明的测试资源文件 340，因此不可能从通过参阅测试资源文件 340 执行处理的测试对象程序的输出屏幕获得关于原始字符串的信息。从而，为了获得关于原始字符串的信息，使用与原始语言的字符对应的字符代码和与包括在变换表 320 中的翻译目标语言的字符对应的字符代码。不与原始语言的字符对应或者包括在变换表 320 中的翻译目标语言的字符对应的字符

代码对应于不是测试对象的多个字符代码。

为了获得关于原始字符代码的信息，与包括在变换表 320 中的翻译目标语言的字符对应的字符代码均与原始语言的字符的字符形状关联，原始语言的该字符具有与相关字符代码对应的翻译目标语言的字符，作为当生成测试资源文件 340 时的“变换为”字符。这样，它可被配置成以致包括在原始资源文件 310 中的字符串都以原始语言显示在通过参阅本发明的测试资源文件 340 进行处理的测试对象程序的输出屏幕上。此外，与原始语言的字符对应的字符代码都被赋予和相关字符代码对应的字符的字符形状。这确保即使存在硬编码字符串，也能够获得关于原始字符串的完美信息。同时，不是测试对象的字符代码都与相同的字符形状关联，以方便识别。如上结合错乱字符检测字体 380 所述那样，可以认为它们是错乱字符的字符代码。即，在利用反转字体的显示屏幕中，不仅能够获得关于原始字符串的信息，而且可以检测错乱字符。在这方面，相同的字符形状最好是明显的字符形状，例如邮政标记，或者不太经常使用的明显的字符形状。

作为反转字体 385 的一个例子，图 9A 表示在以英语作为原始语言，以日语作为翻译目标语言，根据图 4B 中所示的变换表 405 产生测试资源文件 340 的情况下的反转字体的表 850。在表 850 中，为了便于理解，原始语言（英语）的字符的字符代码和对应的字符被添加到最左侧的两列中。因此，位于表 850 中心的一列表示反转字体 385 的字符代码，最右侧的一列表示赋予对应字符代码的字形。它们之间的一列表示常规字体中的字形。在表 850 中，字符代码 U+3000 ~ U+FFEA 是与包括在变换表 405 中的日语字符对应的那些字符代码。如上所述，它们是作为测试对象的字符代码，从而，在转换前，它们被赋予原始英语字符的字符形状。

参见图 9A 中由虚线划界的部分 855，字符代码 U+8868 被赋予常规字体中的

"表"

的字符形状。在反转字体 385 中，字符代码 U+8868 被赋予“S”的字符

形状，“S”是与英语的字符代码 U+0053 对应的字符，字符代码 U+0053 具有字符代码 U+8868 作为“变换为”代码。在表 850 中，字符代码 U+0020 ~ U+007E 是对应于英语字符的那些字符代码。如上所述，它们是与原始语言的字符对应的字符代码，从而，原始语言的字符的字符形状被无修改地赋予它们。表 850 中的其它字符代码对应于 Unicode 中的多个其它字符代码。如上所述，它们对应于错乱字符，从而在本例中，向它们赋予邮政标记的字符形状。应注意表 850 中的行数并不指示字符代码的实际数目。

图 9B 表示在利用反转字体 385 显示的、当启动时读取图 4C 中所示测试资源文件 410 的测试对象程序的处理结果的情况下的显示屏幕 860。如图 9B 中所示，除了具有错乱字符的那部分按钮 880 之外，字符串都以原始语言（英语）显示，并且屏幕与图 1C 中所示的原始显示屏幕 110 大体相同。从而，当使用反转字体 385 时，测试人员能够容易地检测错乱的字符，同时他/她能够确认原始字符串，而不需要打开原始资源文件。

下面说明分类字体 390，分类字体 390 被用于按照在作为第二种语言的翻译目标语言中定义的字符的顺序，确定字符串的分类结果的有效性。在分类结果的有效性的确定中，与包括在变换表 320 中的翻译目标语言的字符对应的字符代码对应于作为测试对象的多个字符代码。由于通过根据变换表 320，把包括在原始资源文件 310 中的原始语言的所有字符转换成翻译目标语言的字符来创建本发明的测试资源文件 340，因此读取测试资源文件 340 的测试对象程序的输出必定包括包含在变换表 320 中的翻译目标语言的字符。因此，与作为第一种语言的原始语言的字符对应的字符代码，以及并不与原始语言的字符或包括在变换表 320 中的翻译目标语言的字符对应的字符代码对应于不是测试对象的多个字符代码。

在分类字体 390 中，为了便于在屏幕上确定翻译目标语言的字符串是否按照在翻译目标语言中定义的字符的顺序被正确分类，作为测试对象的字符代码均与指示为与相关字符代码对应的字符定义的顺序

的字符形状相关联。同时，不是测试对象的字符代码都与相同的字符形状相关联，所述相同的字符形状最好是不太经常使用的字符形状，不明显的字符形状，或者不太经常使用并且不明显的字符形状，比如中空的方块或者空白。

作为分类字体 390 的一个例子，图 10A 表示在利用图 4B 中所示的以英语为原始语言，以日语为翻译目标语言的变换表 405，创建测试资源文件 340 的情况下的分类字体的表 900。表 900 的最左侧一列中是字符代码，最右侧一列中是赋予对应字符代码的分类字体中的字形。在表 900 的中间一列中，添加常规字体中的字形以供参考。在表 900 中，字符代码 U+3000 ~ U+FFEA 是与包括在变换表 405 中的日语字符对应的字符代码。如上所述，它们是作为测试对象的字符代码，从而，它们均被赋予指示为相关字符定义的顺序的字符形状。注意“在翻译目标语言中定义的字符的顺序”可以是任意顺序；如何定义字符的顺序取决于将进行哪种测试。这里，采用包括在变换表 405 中的日语字符的字符代码的顺序。对于在常规字体的字形中被赋予符号的字符代码，比如字符代码 U+FFEA，可被赋予诸如中空的方块之类的不明显字符形状，而不是诸如(93)之类的数字。

在表 900 中，字符代码 U+0020 ~ U+007E 是与英语字符对应的字符代码。表 900 中的其它字符代码对应于 Unicode 中的多个其它字符代码。由于如上所述，它们是不作为测试对象的字符代码，因此在本例中，它们均被赋予一个中空的方块字符形状。应注意表 900 中的行数并不指示字符代码的实际数目。图 10B 表示在利用分类字体的表 900 显示通过参阅图 4C 中所示的测试资源文件 410 执行处理的测试对象程序的处理结果的情况下的显示屏幕 905。参见图 10B，将在列表框 915 上列举的字符串的字符都被显示成数字。从而，通过检查与相应字符串的前些字符对应的数字(如果它们是相同的数字，那么检查后续字符的数字；下面同样如此)，能够容易地确定当通过点击按钮 920(参见图 1C 中的按钮 125)执行分类命令时的分类结果是否正确。从而，当使用分类字体 390 时，即使翻译目标语言具有许多种字符，也能够

容易地确定分类是否正确。

上述四种字体，即，硬编码字符检测字体 375，错乱字符检测字体 380，反转字体 385 和分类字体 390 可按照下述方式创建。复制现有的字体文件并赋予一个新的名称。利用可从市场上购得的字体编辑器等打开新创建的字体文件，酌情编辑将被分配给字符代码的字符形状，从而创建本发明的任意上述字体。当然也可通过使用字体编辑器从头开始创建本发明的字体。至于当按照本发明创建字体时指定的字符编码方案，可以选择任何字符编码方案，只要它包括原始语言的字符集和翻译目标语言的字符集。

下面，参考图 11A 中所示的流程图，说明按照本发明的一个实施例的通过利用模拟翻译的资源文件和为相应测试类别准备的多种字体，测试测试对象程序的处理流程的例子。首先，作为预处理，按照其可被测试对象程序执行单元 355，显示屏幕创建单元 360 和字体选择单元 36 查阅的方式，测试资源文件 340 被保存在第二存储单元中，为相应测试类别准备的多种字体连同常规字体一起被保存在第三存储单元中。为了实际使用字体，必须把字体安装到使用所述字体的计算机中。后面将结合一个实现例子说明安装方法。当处理开始时，首先，测试对象程序执行单元 355 从第二存储单元读取测试资源文件 340(步骤 600)。测试对象程序执行单元 355 通过查阅测试资源文件 340，执行测试对象程序的各种处理(步骤 605)，并输出处理结果(步骤 610)。

显示屏幕创建单元 360 从测试对象程序接收处理结果，并利用包括在第三存储单元中的默认字体创建处理结果的显示屏幕(步骤 615)。此时，显示屏幕创建单元 360 把从测试对象程序接收的处理结果临时保存在缓冲器中。显示屏幕创建单元 360 把创建的显示屏幕传送给显示装置(步骤 620)。字体选择单元 365 等待来自用户的输入(步骤 625)，当收到所述输入时，它确定接收的输入是否对应于字体选择请求(步骤 630)。如果不是字体选择请求(步骤 630: NO)，那么处理终止。如果是字体选择请求(步骤 630: YES)，那么字体选择单元 365 通过查阅第三存储单元，确认可用的字体，并创建字体选择显示屏幕(步骤 635)。

字体选择单元 365 随后把创建的字体选择显示屏幕传送给显示装置，以便显示在屏幕上(步骤 640)。

当字体选择单元 365 从用户收到待使用的字体的指定时(步骤 645)，它向显示屏幕创建单元 360 提供和指定字体的类型有关的信息。显示屏幕创建单元 360 从缓冲器读取测试对象程序的处理结果，并利用用户选择的字体重新创建处理结果的显示屏幕(步骤 650)。显示屏幕创建单元 360 再次把创建的显示屏幕传送给显示装置(步骤 655)。处理随后返回步骤 625。

现在参考图 11B，详细说明步骤 650 中的处理。首先，显示屏幕创建单元 360 从缓冲器读取测试对象程序的处理结果(步骤 660)。显示屏幕创建单元 360 确定读取的处理结果是否包括接下来要处理的数据(步骤 665)。如果不存在接下来要处理的数据(步骤 665: NO)，那么处理终止。如果存在接下来要处理的任何数据(步骤 665: YES)，那么显示屏幕创建单元 360 从处理结果读取接下来要处理的数据(步骤 670)，并确定所述数据是否包括任何字符代码(步骤 675)。如果不包括字符代码(步骤 675: NO)，那么显示屏幕创建单元 360 根据待处理的数据，无修改地创建显示屏幕的一部分(步骤 680)。如果在步骤 675 中确定包括至少一个字符代码，那么显示屏幕创建单元 360 从用户指定的字体中取回赋予相关字符代码的字符形状(步骤 685)，并使用取回的字符形状根据待处理的数据创建显示屏幕的一部分(步骤 690)。

按照本发明的每种字体包括将作为测试对象的多个第一字符代码，按照第一规则与第一字符代码一对一关联的多项第一字符形状信息，作为规定字符编码方案的其它字符代码的多个第二字符代码，和按照第二规则与第二字符代码一对一关联的多项第二字符形状信息。因此，在图 11A 中所示的步骤 645 中指定本发明的任意字体的情况下，如果包括在待处理数据中的字符代码是作为测试对象的第一字符代码，那么显示屏幕创建单元 360 从字体中取回按照第一规则赋予相关第一字符代码的第一字符形状信息，并把所述信息用于屏幕显示。如果包括在待处理数据中的字符代码是不作为测试对象的第二字符代

码,那么显示屏幕创建单元 360 从字体中取回按照第二规则赋予第二字符代码的第二字符形状信息,并把该信息用于屏幕显示。因此,在显示屏幕上,按照可区分的方式显示作为测试对象的第一字符代码和不作为测试对象的第二字符代码。在步骤 680 或步骤 690 之后,处理返回步骤 665。

下面参考图 12-14 说明本发明的一个实现例子。图 12 表示实现本发明的系统 950 的实现例子。系统 950 包括 Web 服务器 970,与 Web 服务器 970 连接的存储装置 955,通过网络与 Web 服务器 970 连接的客户端 975,和与客户端 975 连接的存储装置 980。存储装置 955 包括作为测试对象程序的 Web 应用程序 960,和测试资源文件 965。存储装置 980 包括 Web 浏览器 985 和为相应测试类别准备的多种字体 990。

当通过网络从客户端 975 收到处理请求时,Web 服务器 970 启动执行所述处理的 Web 应用程序 960。此外,当从 Web 应用程序 960 收到处理结果时,Web 服务器 970 通过网络把处理结果传给请求处理的客户端 975。作为测试对象程序的 Web 应用程序 960 从存储装置 955 被装入 Web 服务器 970 的 RAM 中,并由 Web 服务器 970 的中央处理器(CPU)处理,从而起图 3 中所示的测试对象程序执行单元 355 的作用。在启动的时候,Web 应用程序 960 从存储装置 955 读取测试资源文件 965,从而通过查阅相关的测试资源文件 965,执行所请求的处理。从 Web 应用程序输出的处理结果被传给 Web 服务器 970。

响应来自测试 Web 应用程序 960 的测试人员的请求,客户端 975 启动 Web 浏览器 985。Web 浏览器 985 从存储装置 980 被装入客户端 975 的 RAM 中,并由客户端 975 的 CPU 执行,从而起图 3 中所示的显示屏幕创建单元 360 和字体选择单元 365 的作用。响应测试人员的输入,Web 浏览器 985 向 Web 服务器 970 请求 Web 应用程序 960 提供的处理。当从 Web 服务器 970 收到处理结果时,充当显示屏幕创建单元 360 的 Web 浏览器 985 利用默认字体创建处理结果的显示屏幕,并把其显示在客户端 975 的显示器上。

当从测试人员收到字体选择请求时，充当字体选择单元 365 的 Web 浏览器 985 通过查阅保存在存储装置 980 中的字体，或者通过调用获得可用字体名称的函数，获得可用字体的列表，并在客户端 975 的显示器上显示字体选择对话框。当从测试人员收到待使用的字体的指定时，Web 浏览器 985 使用指定的字体在客户端 975 的显示器上再次显示处理结果。

为了使 Web 浏览器 985 可以查阅按照本发明为相应测试类别准备的多种字体 990，必须把字体 990 安装到客户端 975 中。这样的安装由在客户端 975 上运行的操作系统(OS)实现。例如，当使用 Microsoft Windows(注册商标)XP 作为 OS 时，利用控制面板中的字体来增加、修改和管理计算机的字体，安装所述字体。在字体的文件菜单中准备有“Install new font”。当使用 XP 之前的 Microsoft Windows(注册商标)版本时，可以使用称为 Adobe Type Manager(ATM)的字体管理软件。一旦 ATM 被安装到计算机中，那么只需把字体文件拖放到系统文件夹中和重启计算机。

下面参考图 13 中的流程图，说明系统 950 的处理流程。在步骤 700 中，Web 服务器 970 等待来自客户端的请求。作为客户端的 Web 浏览器 985 等待来自测试人员的输入(步骤 705)。当收到对 Web 应用程序 960 提供的处理的请求时，Web 浏览器 985 把该请求传给 Web 服务器 970(步骤 710)。响应收到该请求(步骤 715)，Web 服务器 970 启动 Web 应用程序 960(步骤 720)。当启动时，Web 应用程序 960 读取按照本发明的测试资源文件(步骤 725)，并参阅该测试资源文件，从而执行所请求的处理(步骤 730)。Web 应用程序 960 输出处理结果，并把处理结果传给 Web 服务器 970(步骤 735)。

响应收到处理结果(步骤 740)，Web 服务器 970 把处理结果传给 Web 浏览器 985(步骤 745)。当收到处理结果时(步骤 750)，Web 浏览器 985 利用默认字体创建处理结果的显示屏幕(步骤 755)，并把其显示在客户端 975 的显示器上(步骤 760)。Web 浏览器 985 等待来自测试人员的输入(步骤 765)。当收到来自测试人员的输入时，Web 浏览器

985 确定所述输入是否对应于字体选择请求(步骤 770)。如果是字体选择请求(步骤 770: YES), 那么 Web 浏览器 985 查阅存储装置 980, 以检查可用的字体, 并创建包括按照本发明为相应测试类别准备的多种字体在内的可用字体的列表。

Web 浏览器 985 使用创建的列表在客户端 975 的显示器上显示字体选择对话框(步骤 780)(参见图 14), 并等待来自测试人员的字体的指定。响应来自测试人员的字体指定的输入(步骤 785), Web 浏览器 985 使用指定的字体再次显示处理结果(步骤 790,795)。在步骤 770 中 NO 的情况下, 即, 如果来自测试人员的输入不是字体选择请求, 那么处理进入步骤 800, 在步骤 800, Web 浏览器 985 执行其它处理。在步骤 795 或步骤 800 之后, 处理返回步骤 765, 在步骤 765, Web 浏览器 985 再次等待来自测试人员的输入。在重复步骤 765~795 中的处理的同时, 当测试人员在按照本发明的为相应测试类别准备的那些字体中切换字体时, 如图 7B、8B、9B 和 10B 中所示的对应显示屏幕被依次显示在客户端的显示器上。

图 15 表示按照本发明的创建测试资源文件的设备 300 的硬件配置的例子。图 15 还表示按照本发明, 利用模拟翻译的资源文件和为相应测试类别准备的多种字体, 测试测试对象程序的设备 350 的硬件配置的例子。图 15 还可被看作当在单台计算机中实现设备 300 的功能和设备 350 的功能时的计算机的硬件配置的例子。在参考图 12-14 说明的系统 950 中, 设备 350 由两台计算机构成, 即 Web 服务器 970 和客户端 975。在这种情况下, 图 15 中所示的硬件配置可被看作每个 Web 服务器 970 和客户端 975 的硬件配置的例子。

系统 1000 包括: CPU 外围部分, 它包括由主控制器 1015 相互连接的 CPU 1010 和 RAM 1040, 图形控制器 1020 和显示装置 1025; 输入/输出部分, 它包括通过输入/输出控制器 1030 与主控制器 1015 连接的通信接口 1070, 硬盘驱动器 1050 和 CD-ROM 驱动器 1060; 和传统输入/输出部分, 它包括与输入/输出控制器 1030 连接的超级 I/O 控制器 1080, 与超级 I/O 控制器 1080 连接的软盘驱动器 1090, 闪速

ROM 1100 和键盘鼠标控制器 1110。

主机控制器 1015 连接 CPU 1010 与 RAM 1040，CPU 1010 以高的传送速率访问 RAM 1040。CPU 1010 根据保存在硬盘中的程序工作，从而控制各个部分。按照本发明的测试测试对象程序的程序保存在记录介质中，比如光学记录介质，例如软盘、CD-ROM、DVD 或 PD，磁光记录介质，例如 MD，或者半导体存储器，例如 IC 卡中，并被提供给用户。另一方面，测试测试对象程序的程序可通过网络从外部计算机下载。类似地，按照本发明的为相应测试类别准备的多种字体被保存在记录介质中，比如光学记录介质，例如软盘、CD-ROM、DVD 或 PD，磁光记录介质，例如 MD，或者半导体存储器，例如 IC 卡中，或者通过网络下载，并被提供给用户。

输入/输出控制器 1030 连接主控制器 1015 与通信接口 1070，硬盘驱动器 1050 和 CD-ROM 驱动器 1060，它们是相对高速的输入/输出装置。通信接口 1070 通过网络与另一计算机通信。CD-ROM 驱动器 1060 从 CD-ROM 读取程序或数据，并通过输入/输出控制器 1030 把读取的程序或数据提供给 CPU 1010。输入/输出控制器 1030 与闪速存储器 1100 连接，并与相对低速的输入/输出装置，比如软盘驱动器 1090 和键盘鼠标控制器 1110 连接。闪速 ROM 1100 保存当启动设备系统 1000 时，由 CPU 1010 执行的引导程序，取决于硬件的程序和其它程序。软盘驱动器 1090 从软盘读取程序或数据，并通过 RAM 1040 把读取的程序或数据提供给超级 I/O 控制器 1080。超级 I/O 控制器 1080 通过并行端口、串行端口、键盘端口、鼠标端口等连接软盘和各种输入/输出装置。

虽然上面结合实施例说明了本发明，不过本发明的技术范围并不局限于在实施例中说明的范围。对于本领域的技术人员来说，显然可对上面的实施例做出各种修改和改进。因此，这样的修改或改进实施例自然包括在本发明的技术范围中。

100

原始资源文件

MSG01=The status is {0}.

Color01=Green

Color02=Red

Color03=Yellow

MSG02=Refresh the Status

MSG03=Sort

图 1A

105

源代码

...

showMessage(MSG01,new Object("Green"))

...

图 1B

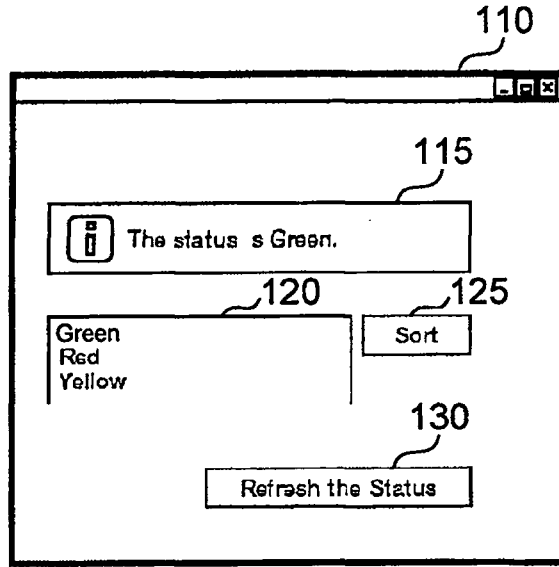


图 1C

200

模拟翻译的资源文件

MSG01=[竹(T)金he status 開始 {0}終了。----]

Color01=[講(G)愛reen----]

Color02=[ノ(R)+ed----]

Color03=[(株)(Y)綠allow----]

MSG02=[ノ(R)+efresh the Status.----]

MSG03=[豹(S)兔ort.----]

图 2A

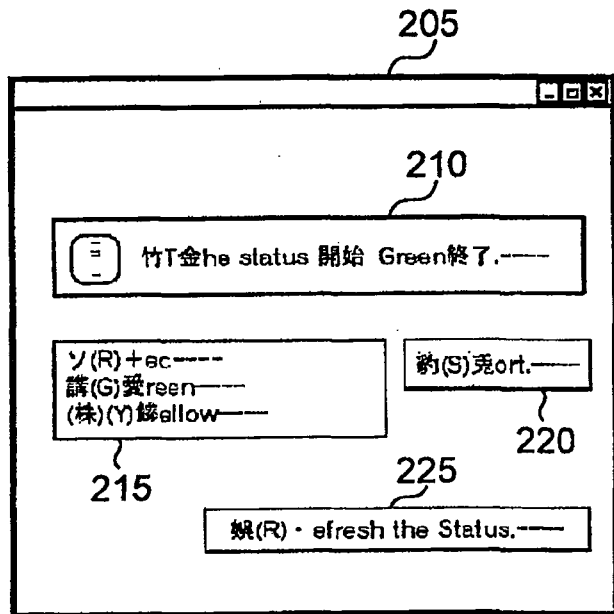


图 2B

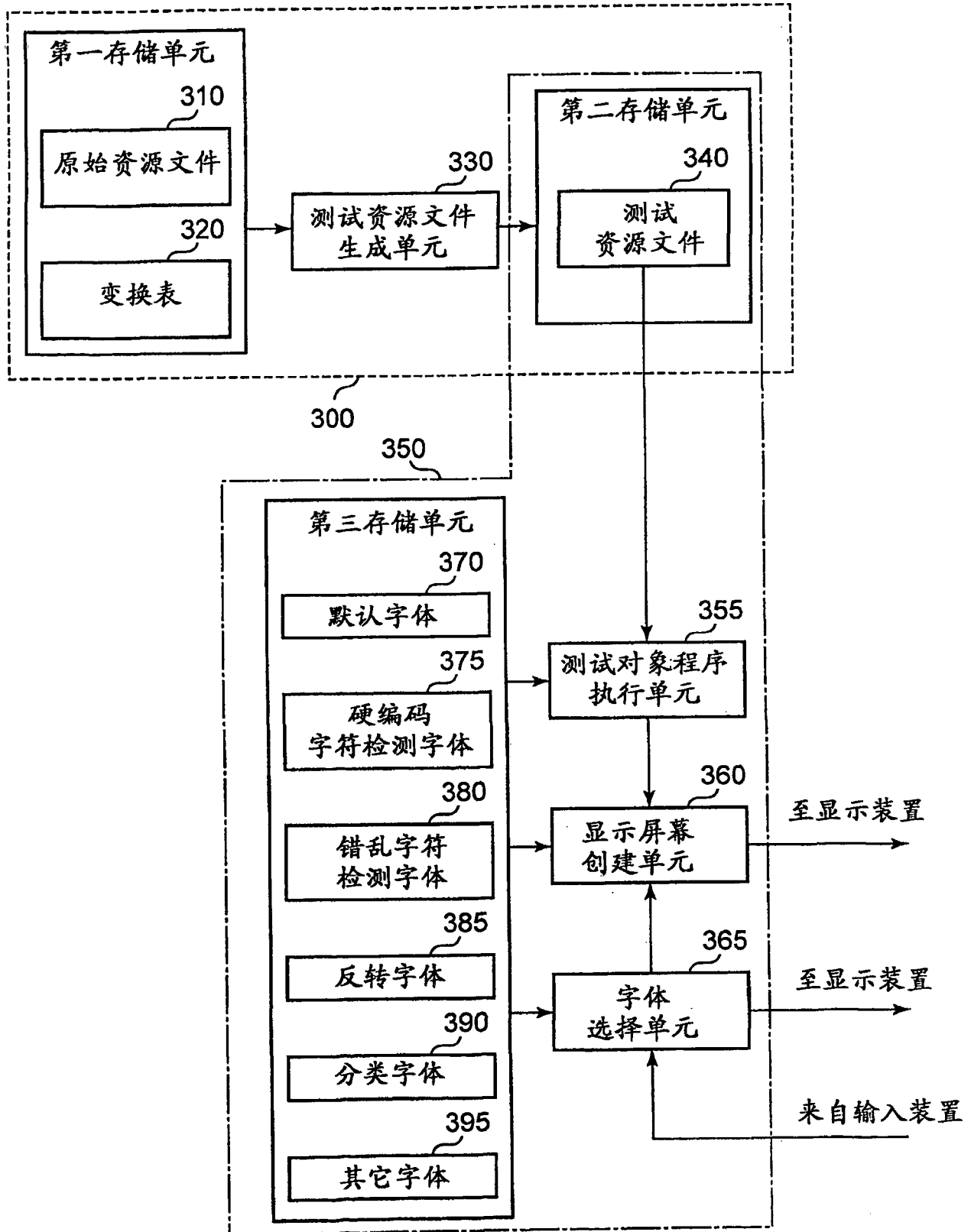


图3

400

原始资源文件

MSG01=The status is {0}.

Color01=Green

Color02=Red

Color03=Yellow

MSG02=Refresh the Status

MSG03=Sort

图 4A

405

字符代码	字符	字符代码	字符
U+0020	空白字符	U+3000	双宽度空白
...
U+0053	S	U+8868	表
U+0054	T	U+5642	樽
U+0055	U	U+30BD	ソ
...
U+007E	~	U+FFEA	↑

图 4B

410

测试资源文件

MSG01=樽愛鮭 カせあせ弓カ 金カ {0}.

Color01=ひと鮭鮭¥

Color02=☆鮭エ

Color03=~鮭構構No.留

MSG02=☆鮭梅ど鮭カ愛 せ愛鮭 カせあせ弓カ

MSG03=表No.どせ

图 4C

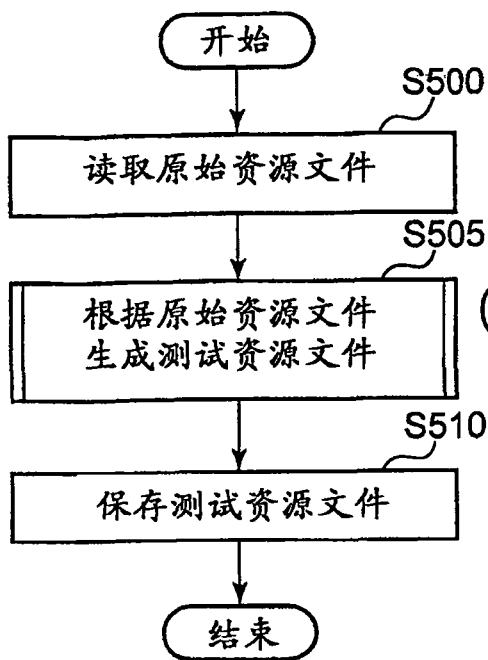


图 5A

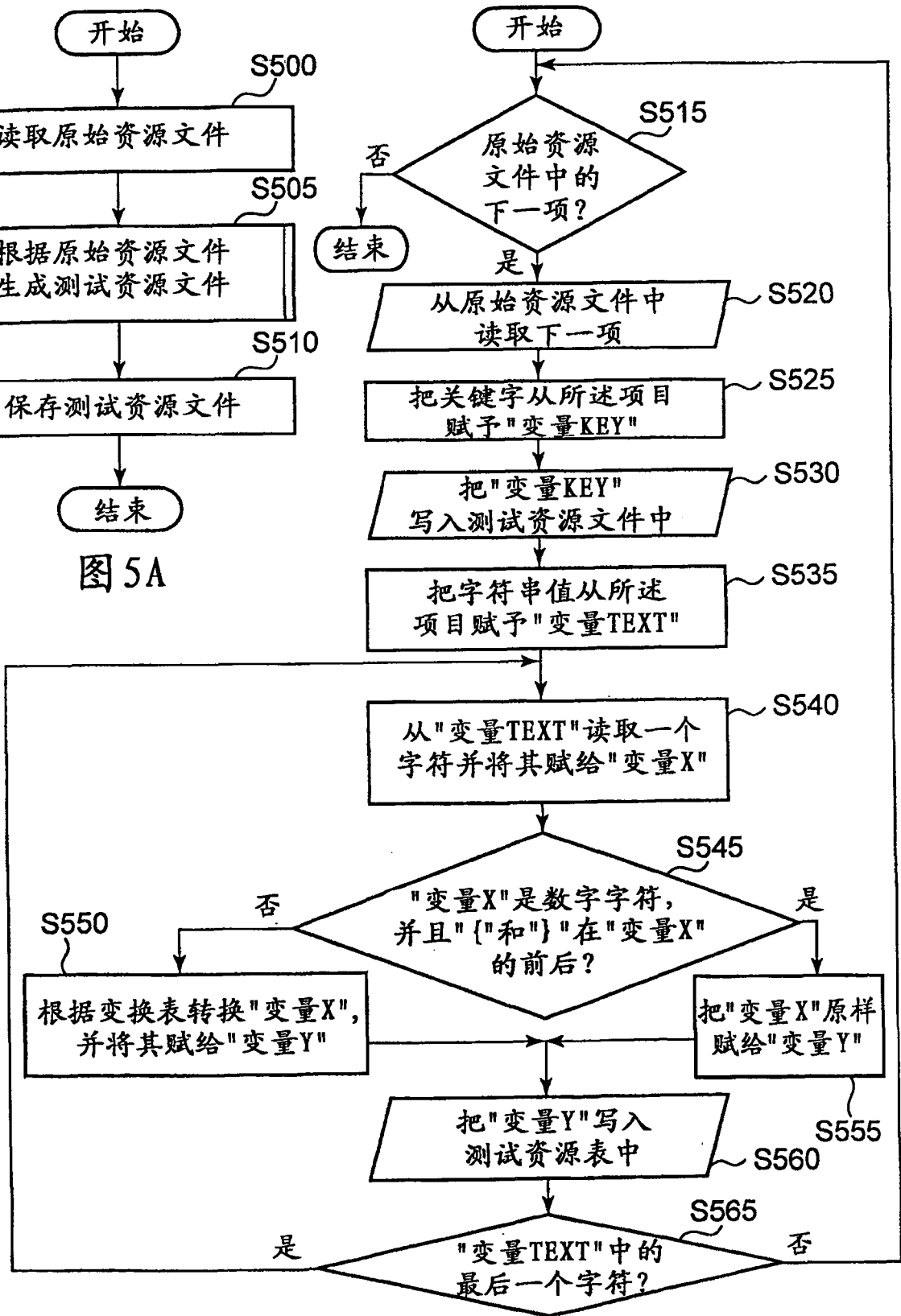


图 5B

硬编码字符检测字体

字符代码	常规字体中的字形	硬编码字符检测字体中的字形
U+3000	空白	□
...
U+FFEA	↑	□
U+0020	空白字符	空白字符
...
U+0053	S	S
...
U+007E	~	~
其它字符代码	□	□

700

不是测试对象的字符代码
(与包括在变换表中的翻译目标语言的字符对应的字符代码)

作为测试对象的字符代码
(与原始语言的字符对应的字符代码)

不是测试对象的字符代码
(不和原始语言的字符或者包括在变换表中的翻译目标语言的字符对应的字符代码)

图 7A

利用硬编码字符检测字体的屏幕显示

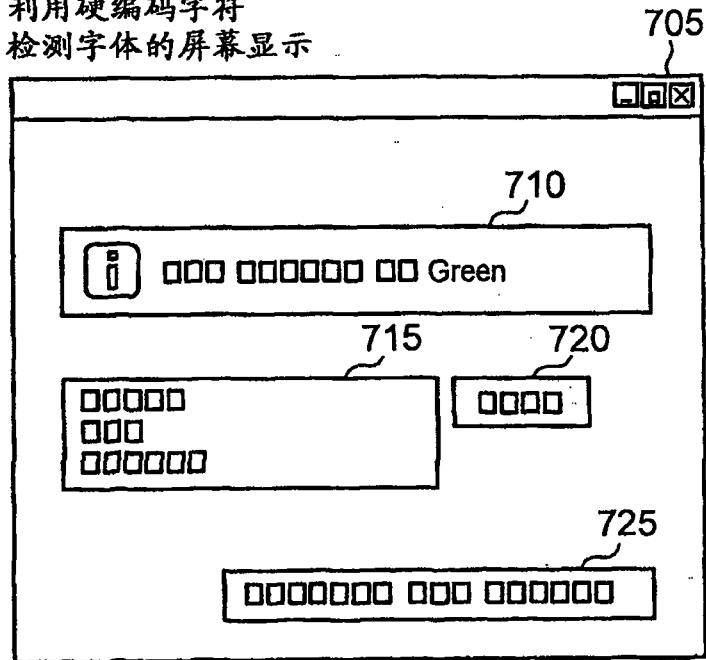


图 7B

错乱字符检测字体 800

字符代码 (日语)	常规字体中的字形	错乱字符检测字体中的字形
U+3000	双空度空白	□
...
U+FFEA	↑	□
U+0020	空白字符	□
...
U+007E	~	□
其它字符代码	□	〒

(与包括在变换表中的翻译目标语言的字符对应的字符代码) 不是测试对象的字符代码

(与原始语言的字符对应的字符代码)

作为测试对象的字符代码 (不和原始语言的字符或者包括在变换表中的翻译目标语言的字符对应的字符代码)

图 8A

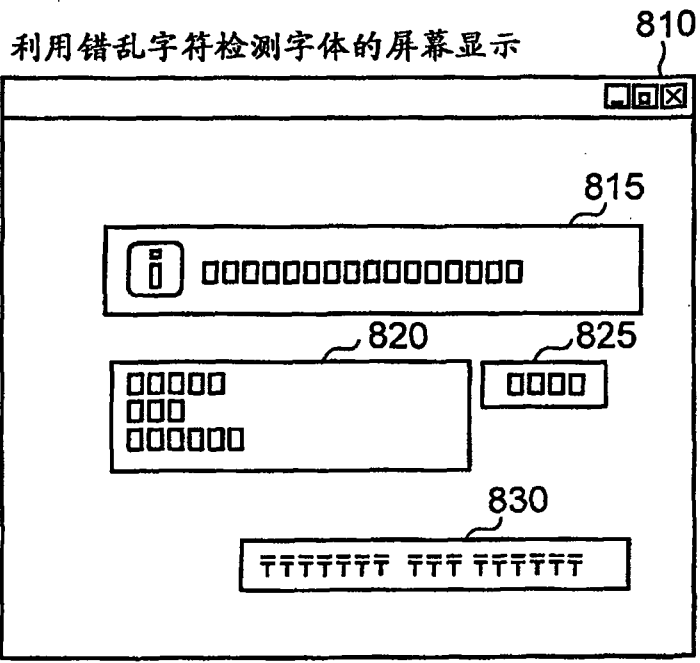


图 8B

反转字体

字符代码 (英语)	对应 字符	字符代码 (日语)	常规字体 中的字形	反转字体 中的字形
U+0020	空白 字符	U+3000	双宽度 空白	空白 字符
...
U+0053	S	U+8868	表	S
U+0054	T	U+5642	噂	T
...
...
U+007E		U+FFEA	↑	~
N/A	N/A	U+0020	空白字符	空白字符
N/A	N/A
N/A	N/A	U+007E	~	~
N/A	N/A	其它代码	□	□/〒

850

855

作为测试对象的
字符代码
(与包括在变换表
中的翻译目标
语言的字符
对应的字符代码)

(与原始语言
的字符对应
的字符代码)

不是测试对象的字符代码
(不和原始语言的字符或者
包括在变换表中的翻译目标
语言的字符对应的字符代码)

图 9A

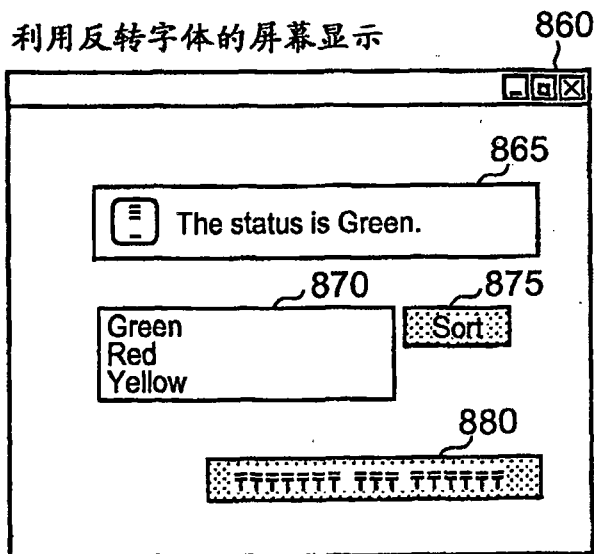


图 9B

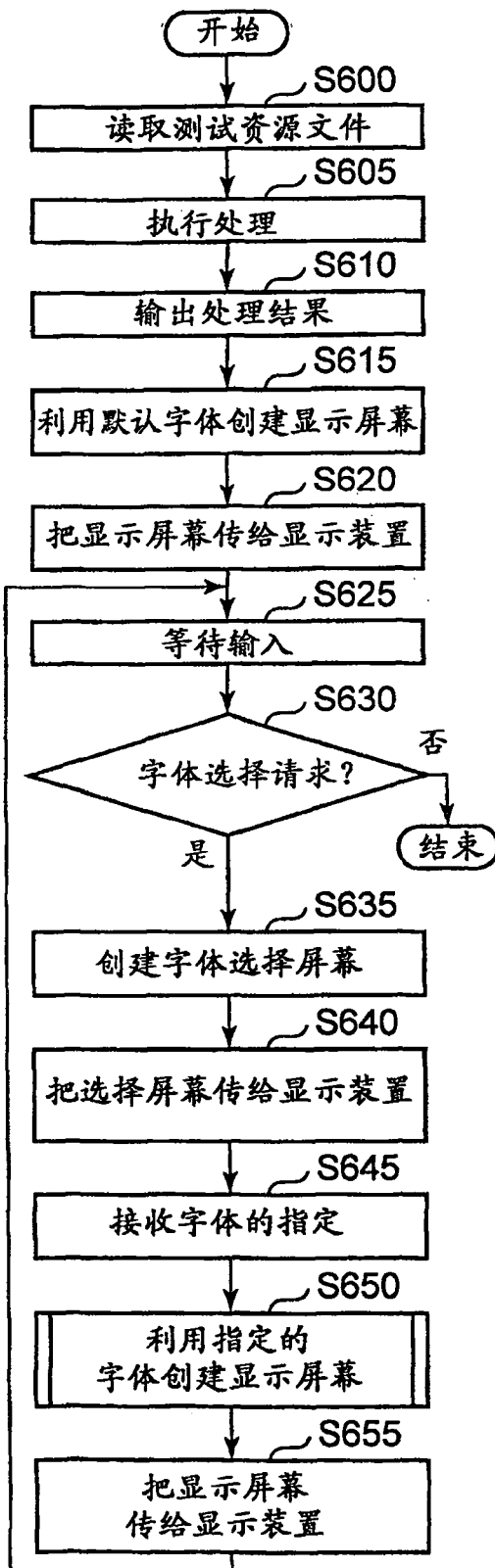


图11A

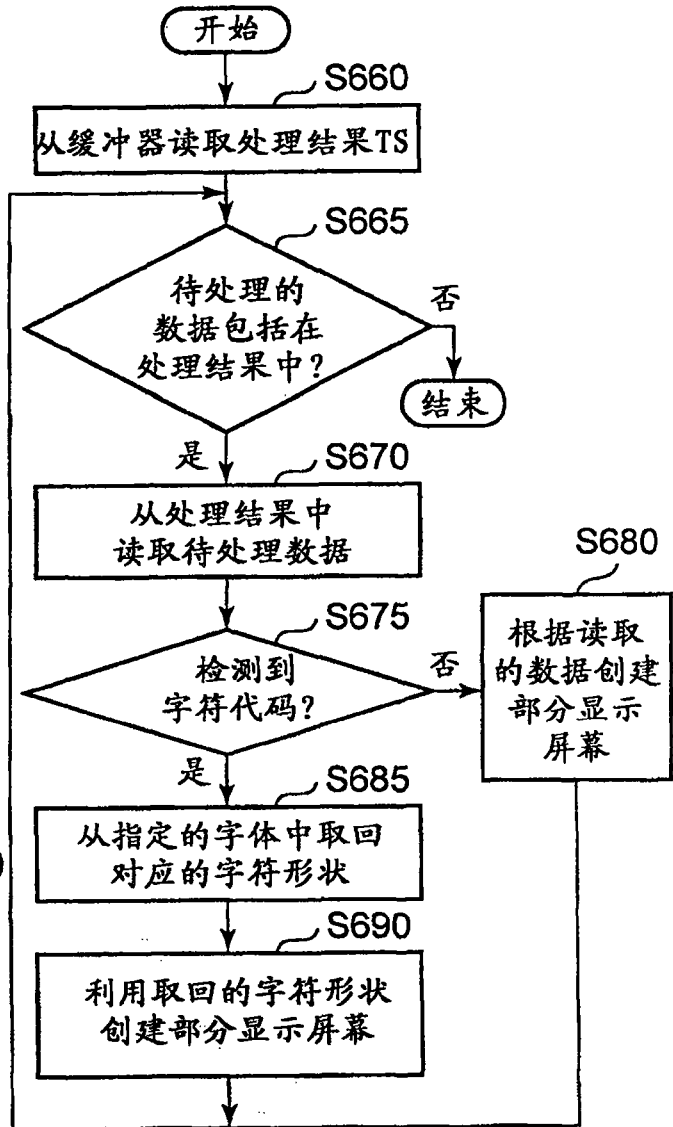


图11B

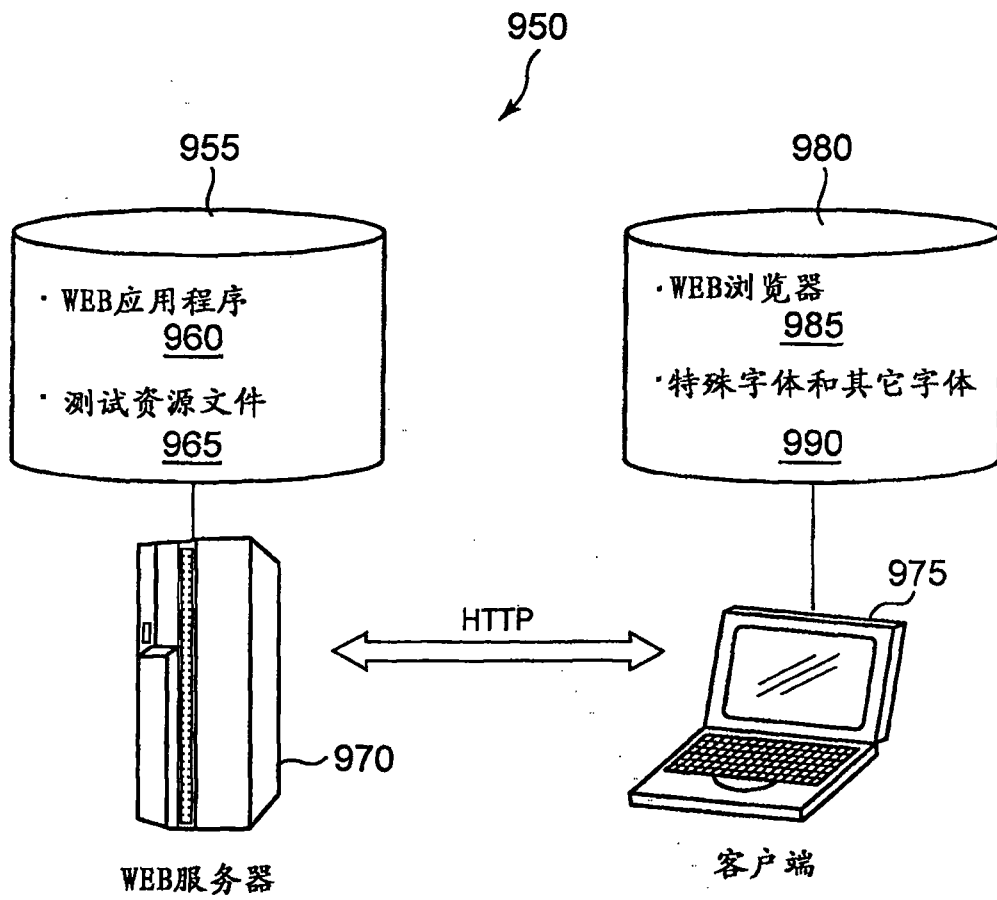


图12

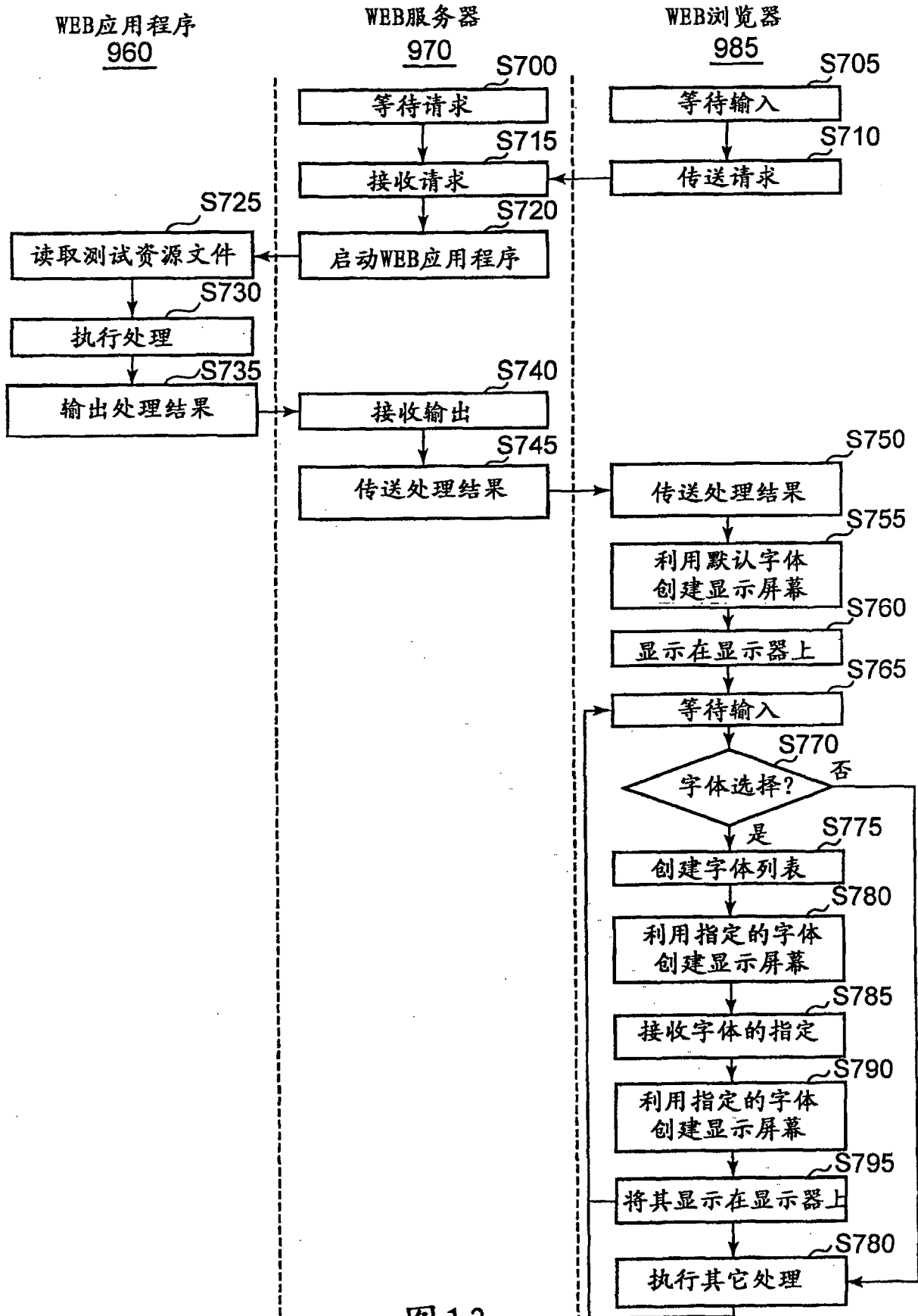


图 13

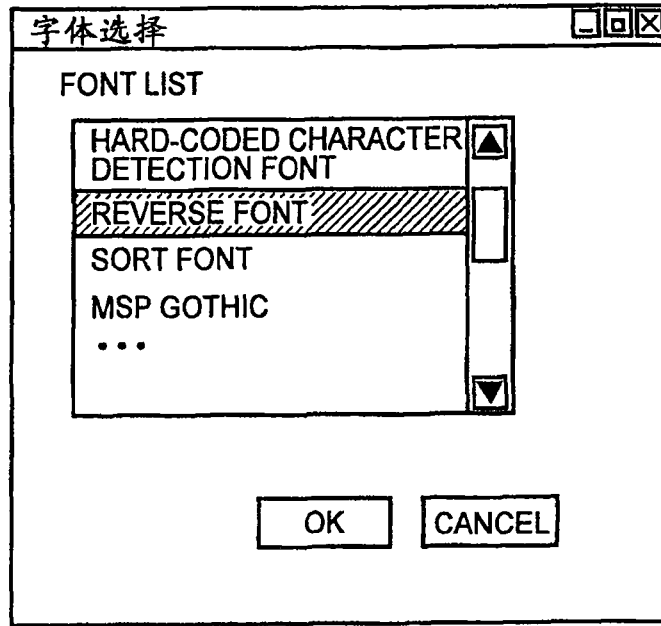


图 14

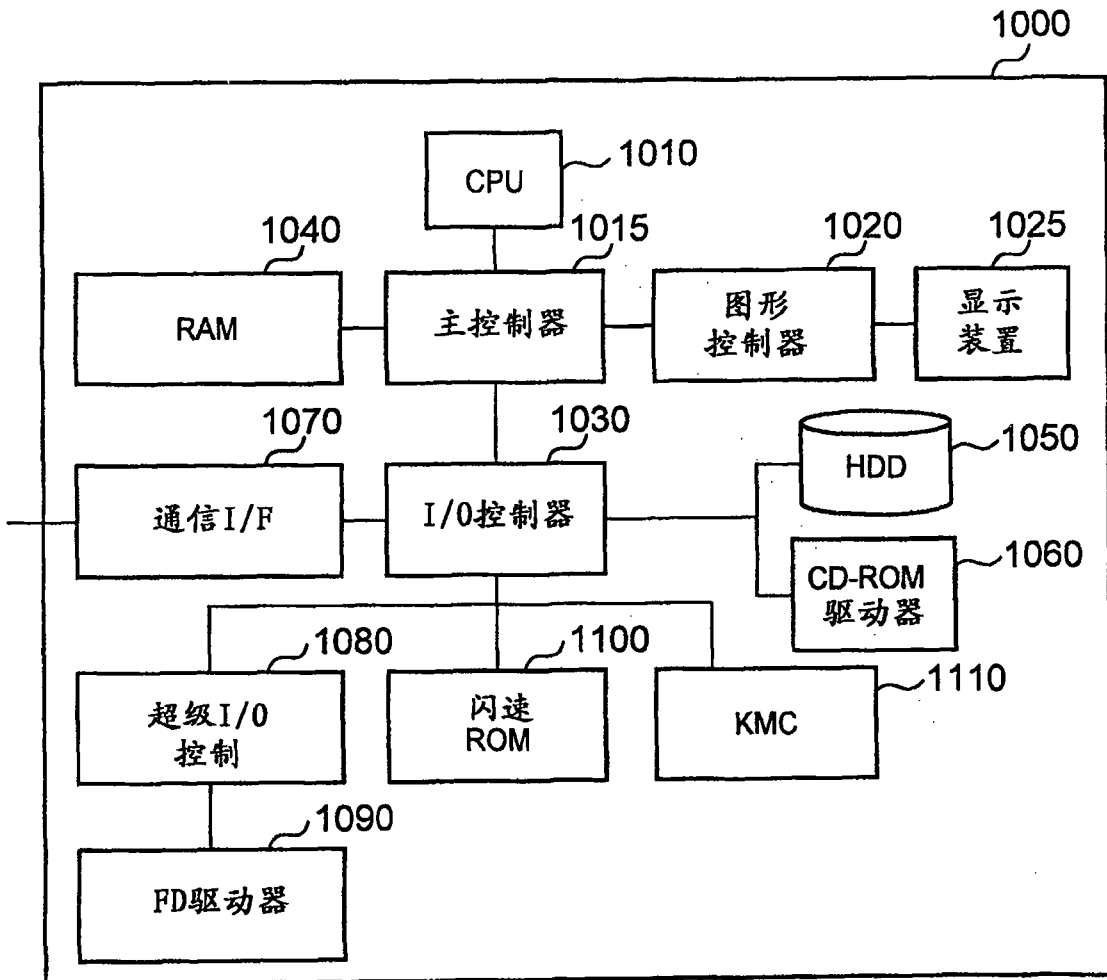


图 15