ICS 35.060
L 74

中华人民共和国国家标准

GB/T 30997—2014/ISO/IEC TR 18037:2008

编程语言　C　支持嵌入式处理器的扩展

Programming languages—C—Extensions to support embedded processors

(ISO/IEC TR 18037:2008,IDT)

2014-09-03 发布　　2015-02-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会　发布

目　　次

前　言

本标准按照 GB/T 1.1—2009 给出的规则起草。

本标准使用翻译法等同采用 ISO/IEC 技术报告 ISO/IEC TR 18037:2008《编程语言　C　支持嵌入式处理器的扩展》,做了如下编辑性修改:

——增加对标准适用范围的描述;

——对原文的符合性要求做了以下编辑性修改:原文中表述"因为这是一份技术报告,所以不存在符合性要求,实现者可以自由选择他们需要的那些规范。(As this is a Technical Report there are no conformance requirements and implementers are free to select those specifications that they need.)"转化为我国国家标准后,不宜有此表述,故将其删除;

——原文中列举出对 ISO/IEC 9899:1999 编程语言 C 部分条目的编辑性修改,为与本标准自身条目相区别,将 ISO/IEC 9899:1999 编程语言 C 的条目加实线框以区分;

——删除了资料性附录 G 对技术报告 18037 第二版的更新和变更。

本标准由全国信息技术标准化委员会(SAC/TC 28)提出并归口。

本标准起草单位:中国电子技术标准化研究院、复旦大学、上海计算机软件评测重点实验室。

本标准主要起草人:李海波、杨丽蕴、丛培勇、贺红卫、苗宗利、王雷、钱乐秋、蔡立志。

引　言

在快速增长的嵌入式系统市场，使用诸如C语言之类高级语言编写应用程序的需求在不断增加。基本上，造成这种趋势的原因有两种：嵌入式系统的程序变得更加复杂（使用汇编语言会难以维护）和嵌入式系统处理器模型的生命周期在变短（这隐含了应用程序要更频繁地重新适应新的指令集）。C语言级别上的编程所能获得的代码重用性被认为是解决上述问题的一个重要前进步骤。

很多技术领域都定义了由处理器提供的功能（例如DSP），这些在嵌入式系统中使用的功能不容易被C语言编写的应用程序采用。比如，定点操作、不同内存空间的用法和底层I/O操作等。目前的提案仅仅只能解决这些技术领域中的一小部分问题。

嵌入式处理器经常被用来分析模拟信号，以及通过对接收到的数据运用滤波算法来处理这些信号。典型的应用程序可以在所有的无线设备中找到。滤波算法中使用的通用数据类型是定点数据类型，并且，为了达到必要的速度，嵌入式处理器经常会配备特殊的定点数据设备。由于C语言（根据ISO/IEC 9899:1999中的定义）不提供对定点运算操作的支持，导致目前程序员因无法选择而不得不使用汇编语言手工编写其大多数算法。本标准为C语言指定了一个在一定精度和饱和度范围内定义的定点数据类型。优化C编译器能像对整型和浮点型数据一样容易地为定点数据类型生成高效率的代码。

许多嵌入式处理器拥有多个不同的内存库并且要求将数据按不同的库分组以达到最大的性能。例如，确保针对FIR过滤设计出进入处理器的乘法器/累加器的并发数据流和协同数据，对处理器的操作至关重要。为了允许程序员声明那个必须从中取出某特定数据对象的内存空间，本标准规定了对多个地址空间的基本支持。因而，优化编译器能利用那些支持多个地址空间的处理器的能力，例如，在一个周期内从两个分离的内存中读取数据，以获得最大执行速度。

随着C语言这些年的逐渐成熟，语言中加入了多种访问基本I/O硬件（iohw）寄存器的扩展来克服语言的缺陷。如今，几乎所有的独立式环境和嵌入式系统的C编译器都支持一些从C源码级别上直接访问iohw寄存器的方法。然而，这些扩展在不同的C语言“方言”之间仍然是不一致的。

本标准提供了一种方法，用于针对基本iohw寄存器寻址编纂通用实践和提供单一的统一语法。

建议本标准与ISO/IEC 9899:1999结合使用。

编程语言　C　支持嵌入式处理器的扩展

1　范围

本标准规定了 ISO/IEC 9899:1999 编程语言 C 的一系列扩展。这些扩展支持嵌入式处理器。

本标准的每章都处理一个特定的主题。第 4、5、6 章的第一条都包含了对该主题的特征的技术描述。这些条文提供了概览但不包含所有的细节。每一章的最后一条都包含对 ISO/IEC 9899:1999 编程语言 C 的编辑性修改，这些修改对完整地规定 ISO/IEC 9899:1999 中相关主题是必要的，并提供一个完整的定义。附加的解释和原理列在附录中。

本标准适用于使用 C 语言对嵌入式处理器进行程序开发的过程。

2　规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本(包括所有的修改单)适用于本文件。

ISO/IEC 9899:1999 编程语言 C(ISO/IEC 9899:1999—Programming languages—C)

3　符合性

本标准分 3 章表述了基本独立的三个功能集合的规范(第 4 章定点运算、第 5 章命名地址空间和命名寄存器存储类，以及第 6 章基本输入/输出硬件寻址)。如果实现某一章的功能，建议实现者实现该章的全部功能。

4　定点运算

4.1　定点数据类型概述和原则

4.1.1　数据类型

本标准中，定点数据值是指小数值(介于－1.0～＋1.0 之间的值)或者由整数部分和小数部分组成的值。由于小数点的位置是已知的，因此对这些数据类型值的操作，能以与整数值操作同样的效率实现。对定点数据值及其操作的典型用法，可以在将模拟值转换成数字表示后的一些过滤算法的应用程序中发现。更多关于定点数据类型的信息，参见 A.1.1。

本标准中，对 C 语言增加了两组定点数据类型：fract 类型和 accum 类型。fract 类型的数据值没有整数部分，因此 fract 类型的值介于－1.0～＋1.0 之间。accum 类型的取值范围取决于该数据类型中整数位的个数。

定点数据类型由相应的新的关键字和类型区分符_Fract 和_Accum 来指定。这些类型区分符与已有的类型区分符 short、long、signed、unsigned 组合起来，能用于指定下列 12 种定点类型：

unsigned short_Fract　　unsigned short_Accum

unsigned_Fract　　unsigned_Accum

unsigned long_Fract　　unsigned long_Accum