

原创文章，转载请注明出处。

更多实用资料请登录方正智芯官网：[www.founderchip.com](http://www.founderchip.com)

作者：北岛李工

前几天有网友询问关于问静态变量的问题，考虑到该问题的普遍性，今天这篇文章我们就来了解下什么是静态变量？什么是临时变量？二者之间有什么区别？



我最早接触“静态变量”的概念是在计算机C语言的编程中，清楚的记得它需要用"static"关键字来声明。静态变量（Static Variable）其实也是一种变量（Variable），因此在介绍静态变量（Static Variable）之前，我们先来介绍下在计算机和PLC的编程中“**变量（Variable）**”的概念。

所谓“**变量**”，是在程序的运行过程中，其值可以根据某些条件而发生变化的量。与“变量”相对的是“**常量（Constant）**”，顾名思义，常量的值在整个程序的运行中保持不变。

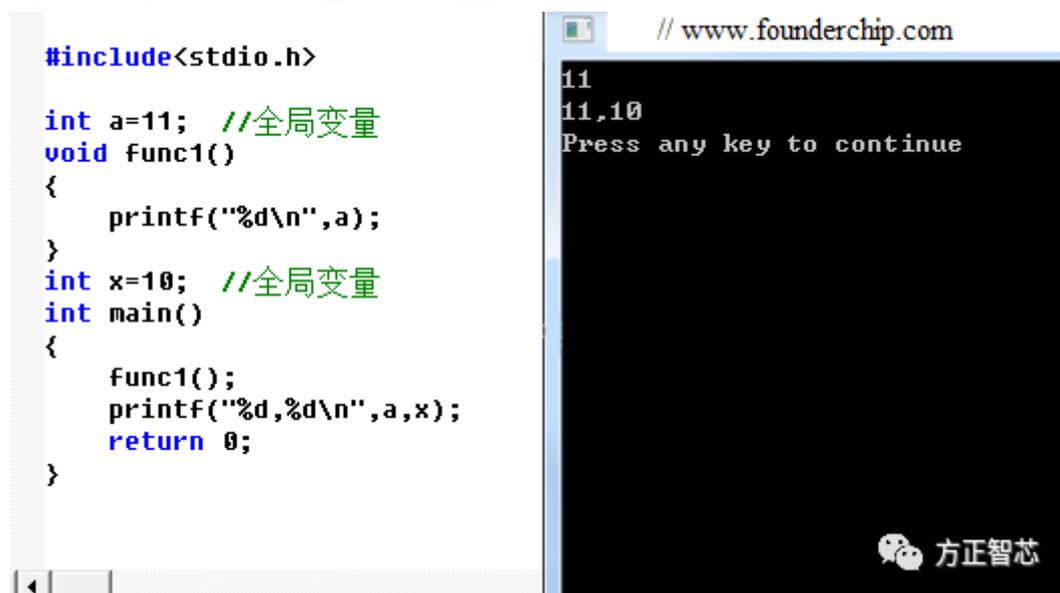
变量的名字称为“变量名”，变量名通常是一个助记符，用来表示一个物理地址。在这个地址上存放着变量的值。

变量有两个特性：作用域和生命周期。

变量的作用域是指变量的作用范围，也就是在哪些范围内该变量是有效的。根据作用域的不同，变量可以分为全局变量（Global Variable）和局部变量（Local Variable）。

全局变量在整个程序范围内都有效。

在C语言中，全局变量是在函数外部声明的变量。比如下面C语言代码中的“a”和“x”都是全局变量：

The image shows a code editor window on the left and a terminal window on the right. The code editor contains the following C code:

```
#include<stdio.h>

int a=11; //全局变量
void func1()
{
    printf("%d\n",a);
}
int x=10; //全局变量
int main()
{
    func1();
    printf("%d,%d\n",a,x);
    return 0;
}
```

The terminal window shows the output of the program:

```
11
11,10
Press any key to continue
```

The terminal window also has a URL bar at the top that says "// www.founderchip.com" and a logo at the bottom right that says "方正智芯".

*注：由于C语言从上而下执行，全局变量x并不能在函数func1中访问。*

在西门子S7系列PLC的编程中，那些存放在公共存储区的变量都属于全局变量。

公共存储区包括输入缓存区（I）、输出缓存区（Q）、位存储区（M）、全局数据块（Global DB）、定时器（T）、计数器（C）等等。比如，位存储区的M0.0，你可以在主程序（OB）或子程序（FB、FC）中访问它，它是全局的。

局部变量（Local Variable）是只能在某个子程序（FB或FC）中访问的变量。也就是说，局部变量是私有的，子程序（FB或FC）只能访问其自身内部定义的变量，而不能访问其它子程序内部定义的变量。

从生命周期来看，全局变量的生命周期是整个程序的运行周期。也就是说，只要PLC还运行用户程序，全局变量就一直有效。

而局部变量的生命周期则取决于其存放的位置。

对于S7系列PLC而言，局部变量有两个存放的位置：临时变量存储区（堆栈）和背景数据块（Instance DB）。

临时变量存储区（堆栈），也称为局部变量存储区（Local Memory），用来存放FB或FC在运行过程中的临时变量。

在S7系列PLC中，FB、FC或者OB都被称为“程序组织单元（POU，program organizational unit）”。PLC为每个POU分配一定数量的临时变量存储区，在S7-300/400 PLC中，这个区域的大小为256字节；在S7-200 Smart PLC中，这个区域的大小为64字节。

临时变量存储区中的变量称为“**临时变量**”，它们只在当前POU运行时有效。当POU运行结束后，该变量将被释放（无法保存）。

*注：当POU由于中断而被挂起时，其临时变量会被保存，以便下次中断现场恢复后继续执行。*

如果想保存程序运行的中间值怎么办呢？答案是使用静态变量（Static Variable）。

在博途（TIA Portal）环境或Step7开发环境中，功能块（FB）接口有“Static(STAT)”一项。所有在“STAT”栏内定义的变量都将会被存放到背景数据块中。在FB运行结束后，“STAT”中定义的变量不会被释放。这种变量称为“**静态变量**”。如下图中的变量“STAT\_test1”和“STAT\_test2”：

FB_Test							
	Name	Data type	Default value	Retain	Accessible f...	Visible in ...	Setpoint
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	▼ InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	▼ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	STAT_test1	Bool	false	Set in IDB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	STAT_test2	Bool	false	Set in IDB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	▼ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	▼ Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

*注：FB中的IN、OUT参数也会保存到背景数据块中。背景数据块中的数据只能被拥有它的功能块（FB）访问。*

说到这里，你应该明白了吧？静态变量从作用域上分属于局部变量；从生命周期上来看，它与用户程序的生命周期相同。临时变量也属于局部变量，不过其生命周期短，仅在POU运行的过程中有效。

好了，关于静态变量和临时变量就先介绍到这里。如果你喜欢这篇文章，可以去官网（[www.founderchip.com](http://www.founderchip.com)）下载本文PDF版本。

小程序【李工谈工控】提供方便的文章检索功能，欢迎体验：



扫码关注小程序