

## Logistic-Map 混沌序列的单片机实现\*

李 伟, 禹思敏

(广东工业大学 自动化学院, 广东 广州 510006)

**摘 要:** 混沌序列的产生是混沌理论应用于保密通信领域的一个重要问题。通过对 Logistic 映射进行变换处理, 利用整数运算代替小数运算, 使程序便于在 MCS-51 系列单片机平台上运行, 由此产生 Logistic-Map 混沌序列。

**关键词:** 混沌序列; Logistic-Map; 单片机

中图分类号: TN973.4

文献标识码: B

## Microcontroller implementation of Logistic-Map chaotic sequence

LI Wei, YU Si Min

(College of Automation, Guangdong University of Technology, Guangzhou 510006, China)

**Abstract:** Generation of chaotic sequence is a very important issue for the application of chaos theory in secure communication. In order to generate Logistic-Map chaos sequence, carrying on the transformation of Logistic equation, replacing decimal operation with integer operation, which makes the program run conveniently in the MCS-51 series single-chip computer.

**Key words:** chaotic sequences; Logistic-Map; single-chip computer

混沌及其应用是近年来非线性科学研究领域的一个热点课题。自 Pecora 和 Carroll 提出了混沌同步概念, 使混沌应用于保密通信技术领域成为可能。但要进行有效的混沌通信, 首先必须产生稳定的混沌序列信号。

混沌序列信号发生器最初是利用分立模拟电子元件来实现的, 这种方法能有效地产生混沌信号, 但降低了集成度, 增大了体积, 且在通信应用时, 因为元件参数较大的离散性, 容易造成收发系统之间的电路参数失配。相比而言, 用数字元件实现, 能简化系统并提高抗干扰能力, 例如, 在 DSP 和 FPGA 等数字平台上产生混沌序列, 是一种较为有效的方法。

对于产生形式简单并且应用广泛的 Logistic-Map 混沌序列<sup>[1]</sup>信号, 可在速度较慢、资源有限的 51 系列单片机硬件平台<sup>[2]</sup>上实现。由于 Logistic 映射中产生的迭代数据全是小数, 而单片机处理浮点数的能力较差, 故需对 Logistic-Map 方程进行映射变换, 以便于在单片机上运行实现。

## 1 Logistic 方程的变换处理

Logistic 方程是目前应用较为广泛的一种混沌映射,

其迭代方程的数学表达式为

$$x(n+1) = \mu x(n)[1-x(n)] \quad (1)$$

式中  $\mu \in (0, 4]$ ,  $x(n) \in (0, 1)$ 。系数  $\mu$  的选取, 将决定系统是否处于混沌状态。从图 1 所示 Logistic 方程的 Lyapunov 指数曲线可看出, 只有当系数  $\mu$  的值落在  $(3.6, 4]$  的区间上时, 该系统才会有正的 Lyapunov 指数。而由图 2 所示 Logistic 映射的分岔图可知, 当系数  $\mu$  的值在  $0 \sim 3$  之间时,  $x(n)$  只有 1 个值;  $\mu \in [3, 3.4496)$  时,  $x(n)$  有 2 个值; 当  $\mu$  的值在  $3.4496 \sim 3.56994$  之间选取时,  $x(n)$  会有 4 个值; 只有  $\mu \in (3.56994, 4]$  时, 系统才会处于混沌状态<sup>[3-4]</sup>。

由于单片机并不适合于处理小数, 故为了便于单片机的运行处理, 系数尽可能地选择整数, 这里选取参数  $\mu=4$ , 其时域仿真波形如图 3 所示。

Logistic 方程中,  $x(n)$  的值均为  $0 \sim 1$  之间的小数, 而  $x(n)$  值的精确度对系统是否处于混沌状态有一定的影响, 精度过低会将混沌系统强制性带出混沌。理论上,  $x(n)$  值的精确度越大越好, 但实际上不可能选取无穷精度。在 Matlab 仿真时, 软件采用双精度浮点的形式进行

\* 基金项目: 国家自然科学基金资助项目(60871025); 广东省自然科学基金项目(8151009001000060)

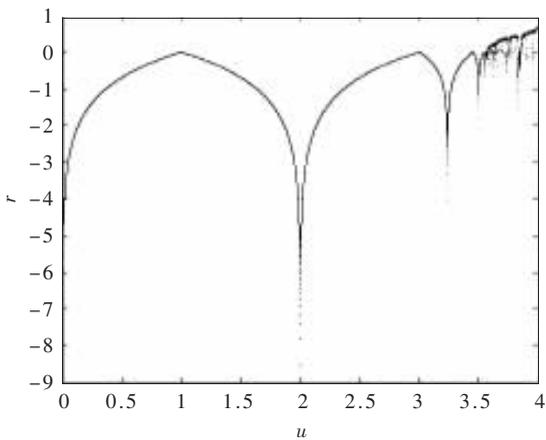


图1 Logistic 映射的 Lyapunov 指数曲线图

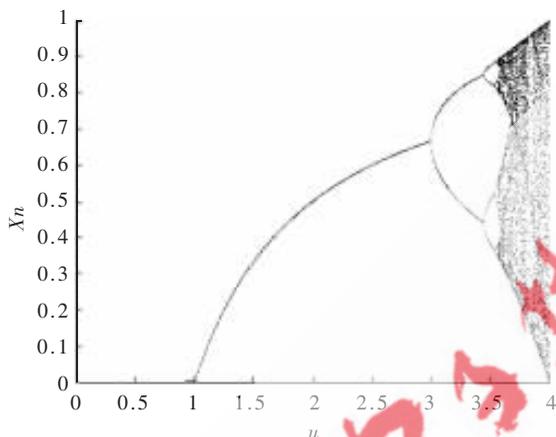


图2 Logistic 映射的分岔图

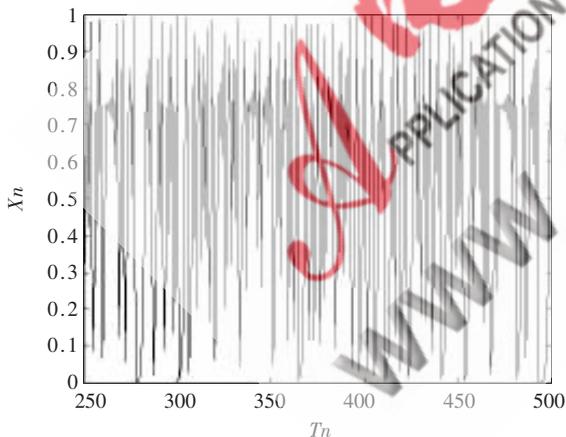


图3  $\mu=4$  时 Logistic 映射的时域仿真图

数值运算。

考虑到单片机处理浮点数的能力较差,通过线性映射:

$$X(n)=65\ 536\ x(n) \quad (2)$$

将  $x(n) \in (0, 1)$  的值映射到  $X(n) \in (0, 65\ 536)$  的区间上,而这属于一个无符号整型变量的表数范围。通过该映射,就能在保证变换前的数值精确到小数点后 4 位的情况下,用无符号整型变量代替双精度浮点型变量的运

算,并且能在一定程度上反映小数点后第 5 位的情况。经过映射变换后的 Logistic 映射方程变为:

$$X(n+1)=4X(n)[65\ 536-X(n)]/65\ 536 \quad (3)$$

式中  $X(n) \in (0, 65\ 536)$ 。数值仿真结果如图 4 所示。由此可知,变换只改变了  $x(n)$  值的幅度,并没有改变系统的性质。

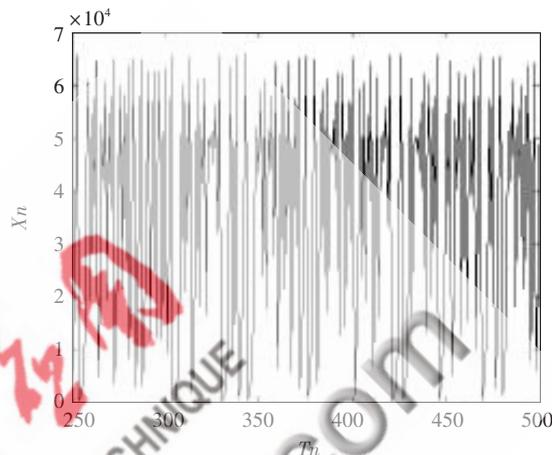


图4 变换后的 Logistic 映射仿真图

## 2 硬件设计

在硬件设计中,选用了 51 系列中的增强型单片机 AT89S52。这款单片机内置了 8 KB 的 Flash,8 个 256 bit 的片内 RAM。对于存储程序和临时数据值有足够的空间,因此可采用单片工作模式,不需扩展外围存储器。

图 5 为 Logistic 混沌信号发生器的硬件电路图。单片机的 P1.6、P1.7 分别输出转换控制信号和控制时钟信号;P1.4、P1.5 则将 Logistic 方程的迭代结果串行地送入数模转换器。

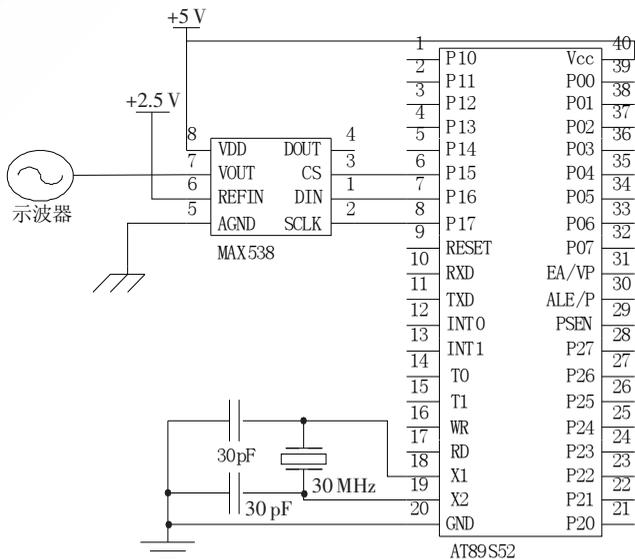


图5 Logistic 混沌信号发生器的硬件电路图

要较好地单片机计算结果还原成模拟信号并在

网络与通信 Network and Communication

示波器上显示出来,对D/A转换器的精度有一定的要求<sup>[5]</sup>。数模转换器的精度过低,会使许多不同的计算结果被转换为相同的模拟电压值,导致波形质量的严重下降。这里选用了廉价高速的12位精度数模转换器MAX538<sup>[2,6]</sup>。这款D/A转换器是电压输出型,串行输入方式,由单5V电源供电。它采用菊花链结构,1次接收16位的数据,其中只有低12位有效,而高四位只在构成菊花链时使用。图6为MAX538的时序图,当转换控制信号为高电平时,D/A转换器不接收新值,而将数字信号转换成模拟信号,输出的模拟信号电压值:

$$V_{out} = V_{REF} \times (\text{数字值} / 4096) \quad (4)$$

式中,  $V_{REF}$  为基准电压,取为2.5V;当其为低点平时,每来1个控制时钟脉冲就读入1位值。

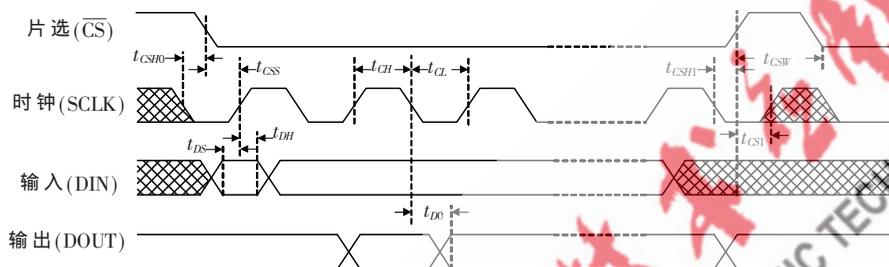


图6 MAX538的时序图

3 软件流程设计

根据(3)式可知,程序中涉及到了减法、乘法和除法运算。在单片机指令系统中,这3种运算指令均为多周期指令,会占用较多的运行时间,尤其是除法运算。相对而言,赋值指令、移位指令与逻辑运算指令则较为快速,因此,在保证准确性的前提下,可用移位运算代替乘、除运算<sup>[7]</sup>。

在设计程序时,利用循环迭代,使单片机不断地计算出 Logistic 方程的当前值。程序流程图如图7所示。在整个循环体内,通过以下几个步骤完成1次迭代运算:

(1)将公式(3)中的减法运算 $[65536 - X(n)]$ 用“异或”运算 $[X(n) \oplus 0x\text{FFFF}]$ 代替,所得值送入变量  $y$  所指向的存储空间;

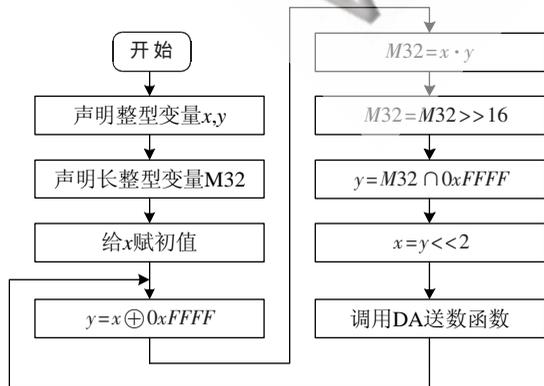


图7 Logistic混沌信号发生器的单片机程序流程图

(2)将  $y$  的值与  $x$  相乘,实现 $[X(n)[65536 - X(n)]]$ ,而这个乘法运算的结果是一个32位的值,因此赋予长整型变量  $M32$ ;

(3)乘法运算结果要除以65536,该值是2的16次方( $65536 = 2^{16}$ ),故可通过将  $M32$  的值右移16位来替换除法运算;

(4)在单片机指令系统中,除法运算的结果是个16位的值,但移位运算并不会改变变量的数据类型,故仍为32位的数据。所以,要令  $M32$  和  $0x\text{FFFF}$  相“与”,取出  $M32$  的低16位值存入与整型变量相对应的存储空间;

(5)用  $y$  的值左移2位替代乘以系数  $\mu = 4$  的运算,计算出 Logistic 方程的当前值;

(6)将计算结果送入数模转换器。

数模转换芯片 MAX538 是串行输入的,上述步骤计算的结果是一个整型数值,不能1次送入 D/A 芯片,要将数值按权位的高低逐个取出并传送。并且 MAX538 是12位的 D/A 芯片,只有低12数据有效。计算结果的16位数据中,对信号幅度影响较大的是高12位数据,故要将  $x(n)$  的值右移4位,将高4位无效位清零,而把有效值保存在低12位。根据 MAX538 的信号时序,可得如图8所示的流程图。



图8 向DA传送数据的控制子函数流程图

混沌系统方程中,均为小数形式的运算,增加了产生混沌信号时的硬件复杂程度。因此,通过对 Logistic 方程进行线性变换,采用整数运算替代浮点数运算,可以在保证一定数值精度的情况下,提高程序的运行速度,

为利用 51 系列单片机产生混沌序列信号提供了一种实际可行的实现方法。

参考文献

[1] 吕金虎,陆君安,陈士华.混沌时间序列分析及其应用[M].武汉:武汉大学出版社,2002.  
[2] 李群芳,张士军,黄健.单片微型计算机与接口技术(第2版)[M].北京:电子工业出版社,2005.  
[3] 赵艳红,张春,吴楚.扩频通信中数字混沌序列的产生[J].信息工程大学学报,2000(3):40-43.

[4] 刘文波.Logistic 映射的电路实现及应用[J].数据采集与处理,2001(1):129-132.  
[5] 赵耿,郑德玲,董翼媛.Logistic 映射数字混沌产生器[J].北京科技大学学报,2001(2):173-176.  
[6] 王福瑞.单片微机测控系统设计大全[M].北京:北京航空航天大学出版社,1998.  
[7] 唐秋玲,覃团发,姚海涛,等.数字语音混沌保密通信系统及硬件实现[J].电子技术应用,2000(2):58-60.

(收稿日期:2009-04-13)

