

87C51 单片机在电子电话机拨号上的应用

程启明

(盐城工专高教研究室, 盐城, 224003)

摘要 讨论 87C51 单片机在电话机的脉冲/双音多频两种拨号中的应用。本系统具有设计先进、功能强、可靠性好、使用灵活方便等特点。

关键词 87C51 单片机 脉冲拨号 双音多频拨号

分类号 TP391 TN916.3

随着人们对电话机的质量要求提高,老式电话机逐步被由集成电路或单片机等构成的按键式电话机所取代。本文采用 MCS-51 系列中 87C51 单片机控制电话机拨号。

1 系统硬件结构

图 1 为由 87C51 单片机构成的电话拨号电路图。12 个按键直接输入 $P_{2.7} \sim P_{2.0}$ 和 $P_{0.7} \sim P_{0.4}$ 中。脉冲拨号通过 $P_{1.2}$ 端输出,驱动簧继电器,双音多频拨号通过软件模拟双音多频数字信号,通过 $P_{0.3} \sim P_{0.0}$ 输出至 4 位 D/A 转换电路,产生双音多频模拟信号至电话线; $P_{1.0}$ 、 $P_{1.1}$ 接 S_1 和 S_2 开关, S_1 为脉冲/双音多频拨号选择开关; S_2 为取消键,LED1 灯用来指示挂勾/摘勾状态(当用于脉冲拨号时)。此电路比较简单,既没使用 87C51 的任何中断,也没有用它的片内定时器,主要使用 87C51 的内部 SRAM 128B,内部 EPROM 2KB 和 $P_0 \sim P_2$ 并行口等。

2 系统软件结构与工作原理

该电路的程序设计主要包括主程序、脉冲拨号子程序、双音多频拨号子程序、键输入子程序和延时子程序等几部分。

2.1 主程序设计

首先对 P_0 、 P_1 、 P_2 口进行初始化;然后检验是否有键按下,若无键按下,检验 $P_{1.1}$ 位,若 $P_{1.1}$ 为 1,则重新开始,否则,将 $P_{1.2}$ 清 0,重新开始。若有键按下,检验 $P_{1.2}$ 位的状态,若 $P_{1.2}$ 为 0,则将 $P_{1.2}$ 位置 1,重新开始;若 $P_{1.2}$ 为 1,则进一步判断是进行脉冲拨号还是双音多频拨号,若是脉冲拨号,则应忽略键盘上的“*”和“#”号键,最后分别转到相应的子程序进行拨号处理。其主程序流程图如图 2 所示。

2.2 脉冲拨号原理和程序设计

脉冲拨号是最为简单的一种电话拨号。根据按下的键值,由 $P_{1.2}$ 端发出一串脉冲。每个脉冲的负脉冲宽度为 61ms,正脉冲宽度为 39ms。因此,电话线以每秒 10 个脉冲的速度发送信号。例如按下键盘的“3”号键,则要发出 3 个脉冲,注意当按下“0”号键时,将发出 10 个脉冲串。

话线的输出电平还可以通过 470Ω 电阻来调节。在 R_x 点上还可以接收音频信号,所以还可以用于双路通信中。

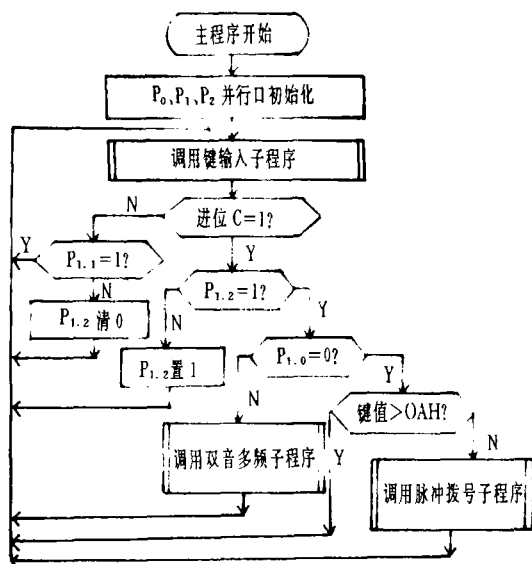


图2 电话机拨号机子程序流程图

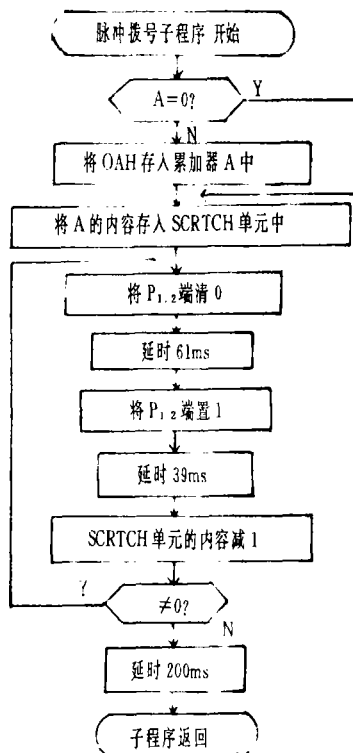


图3 脉冲拨号子程序流程图

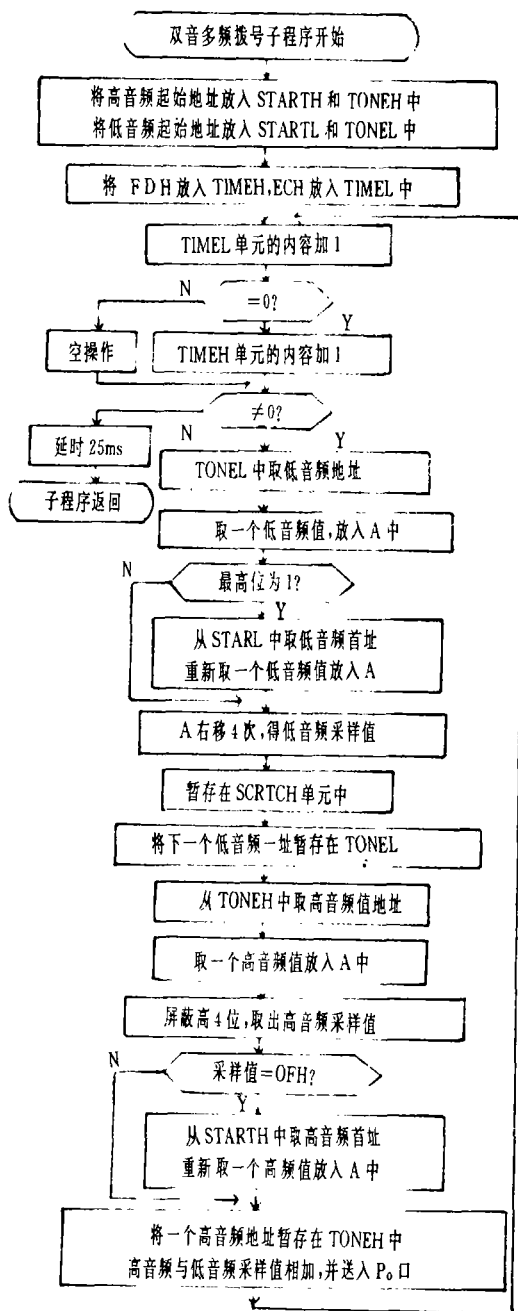


图 4 双音多频拨号子程序流程图

在设计双音多频拨号程序时,需要设置两个表,一个是双音频地址查询表 TFAB,它存放着每个键值对应的低音频和高音频采样值的起始地址;另一个是音频发生器表,在该表中,低音频的采样值存在每个字节的高 4 位中,高音频的采样值存在每个字节的低 4 位中,每种频率的采样值均以 0FH 结束。根据键值从双音频地址查询表中查出存放两种正弦波采样值的起始地址,由此可取出两种正弦波的 3 位采样值,然后相加,得到对应的输出值,将该输出再送入 $P_{0.3} \sim P_{0.0}$ 中。将逐点采样值进行相同的计算,顺序送入 $P_{0.3} \sim P_{0.0}$ 中。若取出的采样值为 0FH,则重新回到起始地址处,循环重新开始,这样可产生一个连续的输出波形。对应于每一个键值,将产生 65ms 的双音多频输出信号和 25ms 的静音,最后返回主程序,等待下一个按键。编程时设计了几个变量:STARTR 和 STARTL 分别存放高音频和低音频的起始地址;TONEH 和 TONEL 分别存放高音频和低音频的当前地址;TIMEH 和 TIMEL 用于存放循环产生输出值的计数器的高 8 位和低 8 位。由于要产生 65ms 的输出信号,而产生某一点输出值需要 $122\mu s$,故需要进行 531 次循环,即需将 TIMEH 和 TIMEL 分别置初值 0FDH 和 0ECH。双音多频拨号程序流程图如图 4 所示。

2.4 键输入与键扫描子程序

该子程序对键盘进行两遍扫描,判断是否有键按下,若有键按下,则进位 C 为 1,键值在累加器 A 中。在这个子程序中,又调用了—个键盘扫描子程序,它用于扫描键盘。首先判断 P_2 口各位对应的键是否按下,再判断 P_0 口的 $P_{0.7} \sim P_{0.4}$ 位对应的键是否按下,若有键按下,则进位 C 置 1,键值在累加器 A 中。在键输入子程序中,使用了一个工作变量 LSTKEY。如果两次扫描均无键按下,则将 0AH 存入 LSTKEY 中;若第一次扫描后无键按下,并且 LSTKEY 中已为 0AH,则不进行第二次键盘扫描;若键盘扫描后有键按下,且键值与 LSTKEY 中的相同,则也认为是无键按下;否则将新键值存入 LSTKEY 中。两次键扫描间隔为 25ms。键输入与键扫描子程序流程图略去。

参考文献

- 1 齐刚编著. 汇编程序设计. 北京理工大学出版社. 1993
- 2 夏栋等编著. 电子电话机原理使用与检修. 人民邮电出版社. 1992
- 3 程启明. 集散型麦芽生产微机控制系统. 工业仪表与自动化装置. 1993, (3): 42~47