

# DCS 监督操作软件的设计方法研究\*

吕 东 庚

(华侨大学计算机科学系, 泉州 362011)

**摘要** 从中小型企业生产实际出发, 分析小型集散控制系统应具备的基本功能、工作特点、系统中各生产对象间的关系及对象内部的结构特点, 并以此为基础提出一种采用分层方式组织数据的 DCS 监控软件设计方法。

**关键词** 集散控制系统, 监督软件, 数据组织

**分类号** TP 317

集散型控制系统(DCS)全称为集中分散型控制系统。“集”指系统的操作、监视及管理功能集中于一台或少数几台监督机及操作站完成。“散”指系统的具体控制功能分散, 它们由分布在生产现场的一些微型机(前沿机)完成。DCS 系统以其监视及管理方便灵活、系统可靠性高而受到人们的普遍青睐。近年来我国一些新建大型企业陆续从国外引进相应产品<sup>[1]</sup>, 这些系统技术性能好、功能强、配套齐全, 但价格高且二次开发困难, 一般中小型企业难于引进使用。为满足中小型企业技术改造的需要, 我们从国内当前具体条件出发, 研制了以 8051 单片机为前沿机的小型 DCS 系统。

## 1 系统的软硬件组成

本系统仅包含直接控制级(前沿机级)及过程管理级, 并把过程管理级中的监控计算机与操作站合并为通用监控机, 其硬件结构如图 1 所示。DCS 监控站软件从功能上可分为: 组态与图形生成、系统监视与调整、以及数据通讯三大部分。对于小型 DCS, 因其信息传输量不大, 可采用 RS-232C 接口, 用定时中断方式构成主从通讯网<sup>[2]</sup>, 在传送报文格式确定后, 数据通讯功能即可用简单程序实现。因此本文仅就其他两部份的软件设计思想进行论述。

为便于操作人员对现场的监视, DCS 监控系统必须具有下述显示功能: 能以简单、规化的图形及符号反映整个系统各生产对象的现场情况, 即具有总貌显示功能; 能以某一具体对象(如窑炉、发酵罐等)为单位, 用通用工业标准图形

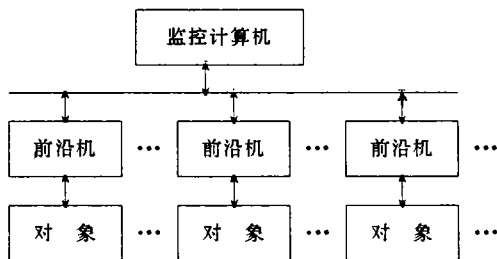


图1 监控机的硬件结构

\* 本文 1996-02-25 收到; 国务院侨办自然科学基金资助项目

显示该对象的工作流程及当前工作状态,即编组画面显示;显示任一给定对象的当前数据及历史数据的变化趋势;以报表形式显示各对象的历史数据及当前数据. 为完成上述功能,系统应具有相应的软件生成模块及显示模块. 用户可使用这些模块,根据对象的特点生成相应的显示文件,并根据监控过程的需要选择相应的显示画面.

为实现系统的通用性,使软件能适应多种对象的需要,系统必须提供多种常用的控制算法,以及常用传感器检测值的多种处理方式. 因此,系统必须具有相应的控制算法库及检测方式库,并提供一个供用户根据对象特性,从上述库中选取所需控制算法及检测方式,然后构成实用测控数据文件的回路组态模块. 根据实现上述要求而规范出的监控站软件功能模块如图 2 所示.

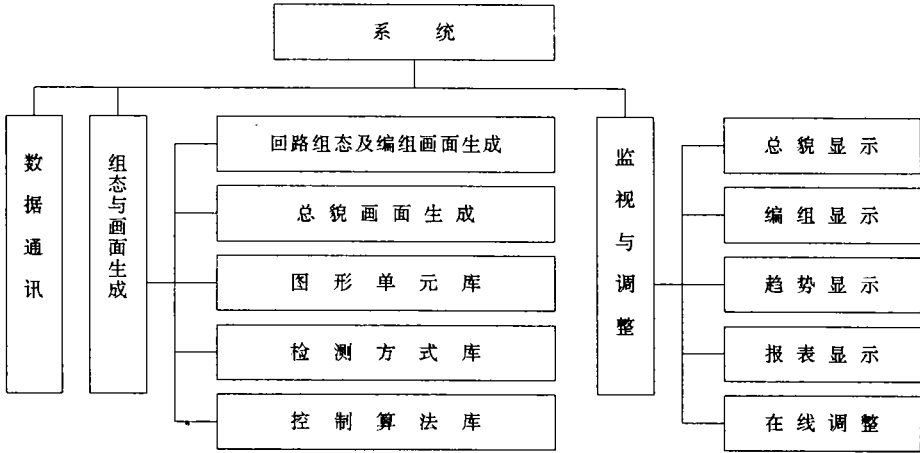


图2 监控站软件功能模块图

2 DCS 监控系统的特点

中小企业虽然因产品的不同,造成生产对象差异甚大,但其测控与监视系统以及生产对象却具有下述一些共同的特点.

(1) 对象运行的独立性. 系统中各个生产对象(如炉、罐、机床等)通常具有较强的独立性,即具有独立的加工对象、独立的工艺流程、独立的启动及停止时间. 因此,每个生产对象可作为一个独立目标处理. 它可拥有属于自己的前沿机,在屏幕上占有独立的显示画面.

(2) 对象结构的组合性. 从测控角度分析,任何一个生产对象,总是由有限个被检测的过程变量和若干个与这些变量直接或间接相关的控制算法组成. 它的流程显示画面同样可由一些简单的图形单元或符号组合而成. 所以,对任一特定的对象,它的测控处理文件及流程画面显示文件,都将是一些常用的检测处理方法、通用控制算法及简单的图元与工程符号的组合.

(3) 画面显示的实时性. 画面显示是系统关键的人机接口,监督操作人员通过画面的切换,可了解系统中整体或各个对象在生产过程中的变化情况. 因此,画面中用于表征过程变化特性的图元及数值,必须随现场数据的变化而周期性地快速刷新. 它意味着在系统中的画面显示文件与现场的实时数据之间应设立相应的快速数据查寻结构.

(4) 在线可操作性. 生产过程的应急处理是所有过程控制所必备的基本功能. 它决定了系统对环境条件的偶然变化或系统中某些难以预计因素出现时的适应能力. 为此, DCS 系统必须容许操作人员在生产过程不间断的前提下, 通过监督站对某些执行机构进行在线调节, 或对某些回路的控制算法进行调整, 以完善地处理相应的突发事件. 这就要求系统必须具备在线的应急信息传递渠道, 有相应的数据组织措施以及相应的执行功能模块.

(5) 联机可维护性. 企业在投产后, 有时需增加若干个测控对象, 或改变某一特定对象中回路间的相互关系. 这在逻辑上就涉及到对显示画面的修改与扩充和对测控回路的再组态, 但在多数情况下 DCS 系统却又必须处于运行状态. 因此, 要求系统必须具备在联机条件下实现回路维护及扩展的功能.

### 3 数据结构设计

根据对象及系统的上述特性, 在本设计中把数据的组织分为 3 层, 即基本单元层、运行参数层及对象层. 对于对象层上的任一生产对象, 可采用有限个给定类型的参数表给予表征, 而每一个运行参数表又用有限个给定类型的单元表征. 基于这一观点, 对一个 DCS 监控软件, 在定义了各种常用的显示图元、检测及控制算法的基本单元、各类运行参数表的结构格式以及对对象的表征格式之后, 就可方便地描述系统中各对象在各种情况下的状态.

#### 3.1 基本单元的数据组织

从监控的角度分析, 对系统中任一对象的描述必须包含 3 类最基本的特征: 外部形态、检测方式及控制算法. 在工业上由于企业的产品不同, 其对象的特性可有很大的差别. 因此, 一个通用 DCS 系统中上述 3 类基本特性中的每一类, 必然含有大量与不同工程对象相对应的具体单元. 它们分别被组织成图形单元库、检测方式库及控制算法库.

(1) 图形单元库. 它包括常见工业流程显示的图元(如方形、矩形、三角形、园形等); 基本符号如开关、阀门、管道等; 常用的部件图符如马达、泵、风机等. 它们均为一个具有特定功能的绘图函数, 每个函数均有一个统一格式的专用名称. 用户可直接使用该名称填写画面组态文件, 系统在进行画面处理时, 则通过组态文件直接调用该函数.

(2) 检测方式库. 它包括铂电阻传感器、铂铑热电偶传感器、电位器式压力传感器等多种传感器参数的修正算法(或修正折线), 以及多种实用的数字滤波算法(如平均值滤波、中值滤波、惯性滤波等). 此外, 用户可根据系统的实际需要, 在必要时对库进行扩充. 库中的每一个算法(或折线), 均有一个统一格式的标识名及序号, 它直接表明该算法属于哪一类传感器. 在系统组态时, 开发人员可根据需要直接从系统提供的菜单中选取相应的检测方式构成组态文件.

(3) 控制算法库. 该库中包含加、减、乘、除、开方根、P、PI、PID 及迭通等常用的控制算法. 用户在必要时还可按照系统规定的函数格式, 自行定义其他控制算法. 系统组态时, 开发人员则根据生产对象所需的控制算法, 通过系统提供的菜单, 从库中选择相应的算法名称, 并填写对应的参数表, 以构成组态文件.

上述 3 类库的建立, 为系统适应多种企业的生产过程控制提供了必需的基本显示、监测及控制手段.

#### 3.2 运行参数表的数据组织

逻辑上的一个运行参数表, 在屏幕上对应一个图元, 物理上对应一个部件(如开关、阀门)

或部件中的某一部分(如一段窑炉、发酵罐的顶部或底部). 在本软件系统中一个运行参数表对应一个结构<sup>[3]</sup>. 参数表依据其所表征物理部件的不同而分为 2 类, 即“图元显示特性表”及“图元动态参数表”. 显示特性表中除包含该图元在图形单元库中的函数名称外, 还包含该图元被显示时的屏幕坐标、相对尺寸、颜色特征等. 动态参数表为显示特性表的附属表. 仅有特性参数随时间变化的图元, 其特性表才有相应的动态参数表, 并在特性表中给出对应的动态参数表指针(Struct dgraphics \* dgram), 其结构如下

```

struct  sgraphica {                                /* 图元显示特性表      */
    int sgtype;                                     /* 图元名称            */
    int firstx,firsty,;                             /* 屏幕坐标 1          */
    int secondx,secondy;                             /* 屏幕坐标 2          */
    int sgsize;                                     /* 相对尺寸            */
    int sgcolor;                                    /* 颜色或动态标志      */
    struct dgraphics * dynam;                       /* 图元动态表指针      */
    struct sgraphics * prev;                         /* 前趋指针            */
    struct sgraphics * next;                         /* 后继指针            */
}

```

图元动态参数表包含该图元所表示的工作部件中动态参数在系统中的逻辑地址、对该参数数据处理方法、数据存贮起始地址、报警状态以及是否处于在线修改状态等, 其结构如下

```

struct  dgraphics {                                /* 图元动态参数表      */
    float spltime;                                  /* 刷新速率系数        */
    int objnum;                                     /* 对应的前沿机号      */
    char opstate;                                   /* 启动状态 ON 或 OFF  */
    char iotype;                                    /* 输入或输出          */
    char datatype;                                  /* 数据类型            */
    char testtype;                                  /* 检测器的类型序号    */
    char testproce;                                 /* 输入变量处理方式    */
    int optrul;                                     /* 控制算法序号        */
    int optpara;                                    /* 算法参变量组号      */
    int standpara;                                  /* 标准工艺曲线组号    */
    float instdata;                                 /* 当前数据            */
    float data[DN];                                 /* 历史数据缓冲区      */
    int ccolor;                                     /* 当前显示颜色        */
    char falarm;                                    /* 报警标志            */
    char fchange;                                   /* 在线修改标志        */
    struct sgraphics * stat;                         /* 图元显示表指针      */
    struct dgraphics * ptev;                        /* 前趋指针            */
    struct dgraphics * next;                        /* 后继指针            */
}

```

采用上述填表方式定义系统中各部件的特性及数据处理方式,并分配数据存贮区,这将有利于用户根据需要在合适的检测器及控制执行机构,同时可根据生产实际的需要选取相应的信号处理算法及控制算法.它不仅提高了监控软件的适应性,而且为今后系统的修改、扩充、在线调整及监控机与前沿机的通讯提供了良好的数据组织基础.

3.3 对象的数据组织

一个生产对象通常包含一系列构成对象实体的物理部件及若干检测与控制回路.为了完整表达一个对象的工作状况,就需要若干个可直接表示上述部件及回路的运行参数表.鉴于不同对象所需的参数表个数差别甚大,本设计采用双向动态链表的结构形式来描述对象.在设计中每个生产对象(如一座辊道窑、一个发酵罐)对应两个链表.一个称为“图元显示特性链(SGLINK)”,链中各节点分别为组成该对象的物理部件的“图元显示特性表”;另一个称为“图元动态参数链(DGLINK)”,链中各节点即为相应测控部件的“图元动态参数表”.它们的结构如图 3 所示,图中的显示特性链仅画出与动态参数有关的节点,其他节点则用“...”表示.上述

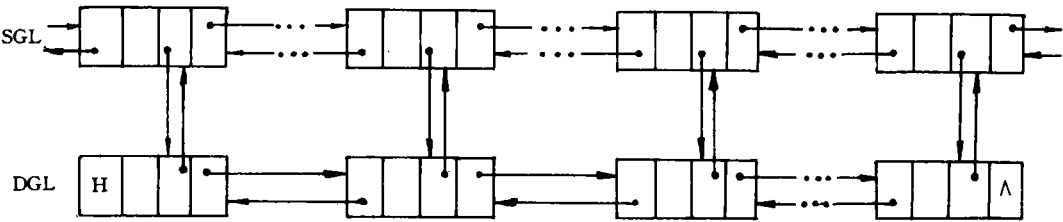


图 3 双向动态链表结构

链表中各节点在系统进行组态编辑时,由用户通过屏幕菜单以交互方式填写,而后由系统依序构成链表.这一结构具有下述优点:画面编辑与测控回路的组态可以同时完成;采用动态内存分配方式,提高了软件的环境适应力;容许系统在线增加或减少测控对象的数量;可分别使用 SGLINK 及 DGLINK 实现静态画面显示及动态画面刷新,从而提高屏幕显示的实时性.

3.4 生产系统的组织

软件中所涉及的生产系统在物理上对应着所有生产对象的集合,其数据组织原则是确保系统在执行前台实时显示、人机交互、以及后台与前沿机进行实时通讯时数据搜索的可靠性与速度要求.为此,本设计使用一个二维指针数组  $A[2][n]$  来描述系统中所有对象.数组的列数  $n$  由用户根据生产系统将实际投入的对象个数,在系统初始化时键入.数组中第 1 行各元素分别为各对象图元显示特性链的链首指针.第 2 行各元素则为相应图元动态参数链的链首指针,其结构见图 4.此外,系统还设置了 3 个显示标志字和 2 个通讯标志字.通过这些标志字,程序

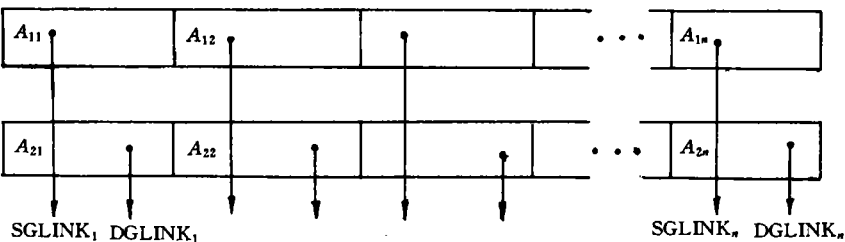


图 4 生产系统的数据组织

可利用上述数组,方便地实现不同屏幕显示格式下的快速显示刷新,及后台与前沿机的正常通讯或在线调整时所需的紧急通讯。

## 4 结束语

本设计从中小型企业生产对象运行的独立性,及其测控回路与外部形态的可组合性出发,首先建立各类通用单元库,而后对生产对象内部的数据采用分层组织方式,统一对象中各层次的数据结构,使得 DCS 系统中的大量静态及动态数据的组织变得规范、简明。由于各对象均使用独立的动态链表,因此系统对生产对象的在线修改、删除及扩充十分灵活,内存利用也更加合理。对象内部的数据组织,采用将静态参数及动态参数分别组成相关链表的方式,不仅使画面组态和回路组态可通过简单的屏幕编辑同时完成,而且提高了显示画面的刷新速度和减少屏幕切换过程的建立时间,从而明显提高了系统的实时性。使用二维指针数组实现系统对所有对象的统一管理,为监控机与前沿机的实时通讯及在线调整提供了快速可靠的查寻路径。

## 参 考 文 献

- 1 王常力,廖道文.集散型控制系统的设计与应用.北京:清华大学出版社,1993.15~17
- 2 吕东庚,吕俊白.集散控制系统中实时通讯功能的实现.计算机应用,1995,(1):46~48
- 3 潘金贵,沈默君,袁峰等.TURBO C 程序设计技术.南京:南京大学出版社,1990.97~99

## A Study on the Design Method of DCS Monitor Operating Software

Lu Donggeng

(Dept. of Computer Science., Huaqiao Univ., 362011, Quanzhou)

**Abstract** Starting from the reality of production in small and medium enterprises, a study is made on the design method of DCS monitor operating software for a small size control system of centralized and decentralized type. The author analyses its envisaged basic function, working characteristic, relation between subjects of labour, and their internal architectural characteristic; and on this basis, puts forward a design method of DCS monitor operating software which organizes data in hierarchic mode.

**Keywords** control system of centralized and decentralized type, monitor software, data organization