Библиотека инженера

Петров И.В.

Программируемые контроллеры

Стандартные языки и приемы прикладного проектирования

- Стандарт МЭК 61131
- Инструменты программирования
- Языки МЭК
- Стандартные компоненты
- Примеры программирования

пусть эта книга принесет вам удачу

УДК 681.5
ББК 32.96
П30

Петров И. В.
П30 Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования / Под ред. проф. В. П. Дьяконова. — М.: СОЛОН-Пресс, 2010. — 256 с.: ил. — (Серия «Библиотека инженера»)

ISBN 5-98003-079-4

Описана практика применения промышленных программируемых контроллеров, широко применяющихся для автоматизации производства. Излагаются языки программирования на основе действующего стандарта МЭК 61131-3 и многочисленные примеры подготовки программ для промышленных программируемых контроллеров.

Для специалистов по автоматизации производственных процессов и производственного оборудования, а также для студентов и преподавателей высших технических заведений.

ISBN 5-98003-079-4

Оглавление

Глава 1. Программируемые контроллеры

В этой главе объясняется, что такое программируемый контроллер, как он работает и для чего вообще он нужен. Кратко будут рассмотрены принципы построения аппаратных средств и системного программного обеспечения, основы систем реального времени и важнейшие технические характеристики контроллеров.

1.1. Определение ПЛК

Любая машина, способная автоматически выполнять некоторые операции, имеет в своем составе *управляющий контроллер* — модуль, обеспечивающий логику работы устройства. Контроллер — это мозг машины. Естественно, чем сложнее логика работы машины, тем «умнее» должен быть контроллер.

Технически контроллеры реализуются по-разному. Это может быть механическое устройство, пневматический или гидравлический автомат, релейная или электронная схема или даже компьютерная программа.

В случае, когда контроллер встроен в машину массового выпуска, стоимость его проектирования распределена на большое число изделий и мала в отношении к стоимости изготовления. В случае машин, изготавливаемых в единичных экземплярах, ситуация обратная. Стоимость проектирования контроллера доминирует по отношению к стоимости его физической реализации.

При создании машин, занятых в сфере промышленного производства, как правило, приходится иметь дело не более чем с единицами однотипных устройств. Кроме того, очень существенной здесь является возможность быстрой перенастройки оборудования на выпуск другой продукции.

Контроллеры, выполненные на основе реле или микросхем с «жесткой» логикой, невозможно научить делать другую работу без существенной переделки. Очевидно, что такой возможностью обладают только *программируемые логические контроллеры* (ПЛК). Идея создания ПЛК родилась практически сразу с появлением микропроцессора, т. е. 30 лет назад.

Физически, типичный ПЛК представляет собой блок, имеющий определенный набор выходов и входов, для подключения датчиков и исполнительных механизмов (рис. 1.1). Логика управления описывается программно на основе микрокомпьютерного ядра. Абсолютно одинаковые ПЛК могут выполнять совершенно разные функции. Причем для изменения алгоритма работы не требуется каких-либо переделок аппаратной части. Аппаратная реализация входов и выходов ПЛК ориентирована на сопряжение с унифицированными приборами и мало подвержена изменениям.

Рис. 1.1. Принцип работы ПЛК

Задачей прикладного программирования ПЛК является только реализация алгоритма управления конкретной машиной. Опрос входов и выходов контроллер осуществляет автоматически, вне зависимости от способа физического соединения. Эту работу выполняет системное программное обеспечение. В идеальном случае прикладной программист совершенно не интересуется, как подсоединены и где расположены датчики и исполнительные механизмы. Мало того, его работа не зависит от того, с каким контроллером и какой фирмы он работает. Благодаря стандартизации языков программирования прикладная программа оказывается *переносимой*. Это означает, что ее можно использовать в любом ПЛК, поддерживающем данный стандарт.

Программируемый контроллер — это программно управляемый *дискретный автомат*, имеющий некоторое множество входов, подключенных посредством датчиков к объекту управления, и множество выходов, подключенных к исполнительным устройствам. ПЛК контролирует состояния входов и вырабатывает определенные последовательности программно заданных действий, отражающихся в изменении выходов.

ПЛК предназначен для работы в режиме реального времени в условиях промышленной среды и должен быть доступен для программирования неспециалистом в области информатики [6].

Изначально ПЛК предназначались для управления последовательными логическими процессами, что и обусловило слово «логический» в названии ПЛК. Современные ПЛК помимо простых логических операций способны выполнять цифровую обработку сигналов, управление приводами, регулирование, функции операторского управления и т. д. В стандарте МЭК и очень часто в литературе для обозначения контроллеров применяется сокращение ПК (программируемый контроллер). Поскольку в России обозначение ПК устойчиво связано с персональными компьютерами, мы будем использовать сокращение ПЛК.

Конструкция ПЛК может быть самой разнообразной — от стойки, заполненной аппаратурой, до миниатюрных ПЛК, подобных показанному на рис. 1.2.

Рис. 1.2. Миниатюрный ПЛК фирмы SIEMENS, Германия

Впервые ПЛК были применены в США для автоматизации конвейерного сборочного производства в автомобильной промышленности (фирма Модикон, 1969 г.). Сегодня ПЛК работают в энергетике, в области связи, в химической промышленности, в сфере добычи, транспортировки нефти и газа, в системах обеспечения безопасности, в коммунальном хозяйстве, используются в автоматизации складов, в производстве продуктов питания и напитков, на транспорте, в строительстве и т. д. Реально сфера применения ПЛК даже шире сферы применения персональных компьютеров. Хотя слава ПЛК значительно меньше. Их работа происходит как бы «за сценой» и незаметна для большинства людей.

1.2. Входы-выходы

На заре своего появления ПЛК имели только *бинарные входы*, т. е. входы, значения сигналов на которых способны принимать только два состояния — логического нуля и логической единицы. Так, наличие тока (или напряжения) в цепи входа считается обычно логической единицей. Отсутствие тока (напряжения) означает логический 0. Датчиками, формирующими такой сигнал, являются кнопки ручного управления, концевые датчики, датчики движения, контактные термометры и многие другие.

Бинарный выход также имеет два состояния — включен и выключен. Сфера применения бинарных выходов очевидна: электромагнитные реле, силовые пускатели, электромагнитные клапаны, световые сигнализаторы и т. д.

В современных ПЛК широко используются *аналоговые* входы и выходы. Аналоговый или *непрерывный* сигнал отражает уровень напряжения или тока, соответствующий некоторой физической величине в каждый момент времени. Этот уровень может относиться к температуре, давлению, весу, положению, скорости, частоте и т. д. Словом, к любой физической величине.

Аналоговые входы контроллеров могут иметь различные параметры и возможности. Так, к их параметрам относятся: разрядность АЦП, диапазон входного сигнала, время и метод преобразования, несимметричный или дифференциальный вход, уровень шума и нелинейность, возможность автоматической калибровки, программная или аппаратная регулировка коэффициента усиления, фильтрация. Особые классы аналоговых входов представляют входы, предназначенные для подключения термометров сопротивления и термопар. Здесь требуется применение специальной аппаратной поддержки (трехточечное включение, источники образцового тока, схемы компенсации холодного спая, схемы линеаризации и т. д.).

В сфере применения ПЛК бинарные входы и выходы называют обычно *дискретными*. Хотя, конечно, это не точно. Аналоговые сигналы в ПЛК обязательно преобразуются в цифровую, т. е. заведомо дискретную форму представления. Но в технических документах ПЛК любой фирмы вы встретите именно указание количества дискретных и аналоговых входов. Поэтому и далее в книге мы сохраним устоявшуюся здесь терминологию.

Помимо «классических» дискретных и аналоговых входов-выходов многие ПЛК имеют *специализированные входы-выходы*.

Они ориентированы на работу с конкретными специфическими датчиками, требующими определенных уровней сигналов, питания и специальной обработки. Например, квадратурные шифраторы, блоки управления шаговыми двигателями, интерфейсы дисплейных модулей и т. д.

Входы-выходы ПЛК не обязательно должны быть физически сосредоточены в общем корпусе с процессорным ядром. В последние годы все большую популярность приобретают технические решения, позволяющие полностью отказаться от прокладки кабелей для аналоговых цепей. Входы-выходы выполняются в виде миниатюрных модулей, расположенных в непосредственной близости от датчиков и исполнительных механизмов. Соединение подсистемы ввода-вывода с ПЛК выполняется посредством одного общего цифрового кабеля. Например, в интерфейсе Actuators/Sensors interface применяется плоский профилированный кабель («желтый кабель») для передачи данных и питания всего по двум проводам.

1.3. Режим реального времени и ограничения на применение ПЛК

Для математических систем характеристикой качества работы является правильность найденного решения. В системах реального времени помимо правильности решения определяющую роль играет *время реакции*. Логически верное решение, полученное с задержкой более допустимой, не является приемлемым.

Принято различать системы жесткого и мягкого реального времени. В системах *жесткого реального времени* существует выраженный временной порог. При его превышении наступают необратимые катастрофические последствия. В системах *мягкого реального времени* характеристики системы ухудшаются с увеличением времени управляющей реакции. Система может работать плохо или еще хуже, но ничего катастрофического при этом не происходит.

Классический подход для задач жесткого реального времени требует построения *событийно управляемой системы*. Для каждого события в системе устанавливается четко определенное время реакции и определенный приоритет. Практическая реализация таких систем сложна и всегда требует тщательной проработки и моделирования.

Для ПЛК существенное значение имеет не только быстродействие самой системы, но и время проектирования, внедрения и возможной оперативной переналадки.

Абсолютное большинство ПЛК работают по методу *периодического опроса* входных данных (сканирования). ПЛК опрашивает входы, выполняет пользовательскую программу и устанавливает необходимые значения выходов. Специфика применения ПЛК обусловливает необходимость одновременного решения нескольких задач. Прикладная программа может быть реализована в виде множества логически независимых задач, которые должны работать одновременно.

На самом деле ПЛК имеет обычно один процессор и выполняет несколько задач псевдопараллельно, последовательными порциями. Время реакции на событие оказывается зависящим от числа одновременно обрабатываемых событий. Рассчитать минимальное и максимальное значения времени реакции, конечно, можно, но добавление новых задач или увеличение объема программы приведет к увеличению времени реакции. Такая модель более подходит для систем мягкого реального времени. Современные ПЛК имеют типовое значение времени рабочего цикла, измеряемое единицами миллисекунд и менее. Поскольку время реакции большинства исполнительных устройств значительно выше, с реальными ограничениями возможности использования ПЛК по времени приходится сталкиваться редко.

В некоторых случаях ограничением служит не время реакции на событие, а обязательность его фиксации, например работа с датчиками, формирующими импульсы малой длительности. Это ограничение преодолевается специальной конструкцией входов. Так, счетный вход позволяет фиксировать и подсчитывать импульсы с периодом во много раз меньшим времени рабочего цикла ПЛК. Специализированные интеллектуальные модули в составе ПЛК позволяют автономно отрабатывать заданные функции, например модули управления сервоприводом.

1.4. Условия работы ПЛК

К негативным факторам, определяющим промышленную среду, относятся: температура и влажность, удары и вибрация, коррозионно-активная газовая среда, минеральная и металлическая пыль, электромагнитные помехи (рис. 1.3). Перечисленные факторы, весьма характерные для производственных условий, обу-

Рис. 1.3. Настройка оборудования (РААЗ, г. Рославль, Смоленская обл.)

словливают жесткие требования, определяющие схемотехнические решения, элементную и конструктивную базу ПЛК. В процессе серийного производства ПЛК обязательным является технический прогон готовых изделий, включающий климатические, вибрационные и другие испытания.

ПЛК — это конструктивно законченное изделие, физическое исполнение которого определяется требуемой степенью защиты, начиная от контроллеров в легких пластиковых корпусах, предназначенных для монтажа в шкафу (степень защиты IP20), и до герметичных устройств в литых металлических корпусах, предназначенных для работы в особо жестких условиях.

Правильно подобранный по условиям эксплуатации контроллер нельзя повредить извне без применения экстремальных методов. Штатными для ПЛК являются такие аппаратные решения, как полная гальваническая развязка входов-выходов, защита по току и напряжению, зеркальные выходные каналы, сторожевой таймер задач и микропроцессорного ядра.

1.5. Интеграция ПЛК в систему управления предприятием

Контроллеры традиционно работают в нижнем звене *автоматизированных систем управления предприятием* (АСУ) — систем, непосредственно связанных с технологией производства (ТП). ПЛК обычно являются первым шагом при построении сис-

тем АСУ. Это объясняется тем, что необходимость автоматизации отдельного механизма или установки всегда наиболее очевидна. Она дает быстрый экономический эффект, улучшает качество производства, позволяет избежать физически тяжелой и рутинной работы. Контроллеры по определению созданы именно для такой работы.

Далеко не всегда удается создать полностью автоматическую систему. Часто «общее руководство» со стороны квалифицированного человека — диспетчера необходимо. В отличие от автоматических систем управления такие системы называют *автоматизированными*. Еще 10 — 15 лет назад диспетчерский пульт управления представлял собой табло с множеством кнопок и световых индикаторов. В настоящее время подобные пульты применяются только в очень простых случаях, когда можно обойтись несколькими кнопками и индикаторами. В более «серьезных» системах применяются ПК.

Появился целый класс программного обеспечения реализующего интерфейс *человек—машина* (MMI). Это так называемые системы сбора данных и оперативного диспетчерского управления (Supervisory Control And Data Acquision System — SCADA). Современные SCADA-системы выполняются с обязательным применением средств мультимедиа. Помимо живого отображения процесса производства, хорошие диспетчерские системы позволяют накапливать полученные данные, проводят их хранение и анализ, определяют критические ситуации и производят оповещение персонала по каналам телефонной и радиосети, позволяют создавать сценарии управления (как правило, Visual Basic), формируют данные для анализа экономических характеристик производства.

Создание систем диспетчерского управления является отдельным видом бизнеса. Разделение производства ПЛК, средств программирования и диспетчерских систем привело к появлению *стандартных протоколов обмена данными*. Наибольшую известность получила технология OPC (OLE for Process Control), базирующаяся на механизме DCOM Microsoft Windows. Механизм динамического обмена данными (DDE) применяется пока еще достаточно широко, несмотря на то что требованиям систем реального времени не удовлетворяет.

Все это «многоэтажное» объяснение призвано подчеркнуть еще одно немаловажное преимущество ПЛК — средства системной интеграции являются составной частью базового программного обеспечения современного ПЛК (рис. 1.4). Допустим, вы

написали и отладили автономный проект на контроллере при помощи системы подготовки программ CoDeSys. Как теперь нужно доработать программу, чтобы связать ПЛК с системой диспетчерского управления, базой данных или Интернет-сервером? Ответ: никак. Никакого программирования далее вообще не потребуется. В комплекс программирования ПЛК входит OPC-сервер. Он умеет получать доступ к данным ПЛК также прозрачно, как и отладчик. Достаточно обеспечить канал передачи данных ПЛК — OPC-сервер. Обычно такой канал уже существует, вы использовали его при отладке. Вся дальнейшая работа сводится к определению списка доступных переменных, правильной настройке сети, конфигурированию OPC-сервера и SCADA-системы. В целом, операция очень напоминает настройку общедоступных устройств локальной сети ПК.

Второй часто возникающей задачей является интеграция нескольких ПЛК с целью синхронизации их работы. Здесь появляются сети, обладающие рядом специфических требований. В це-

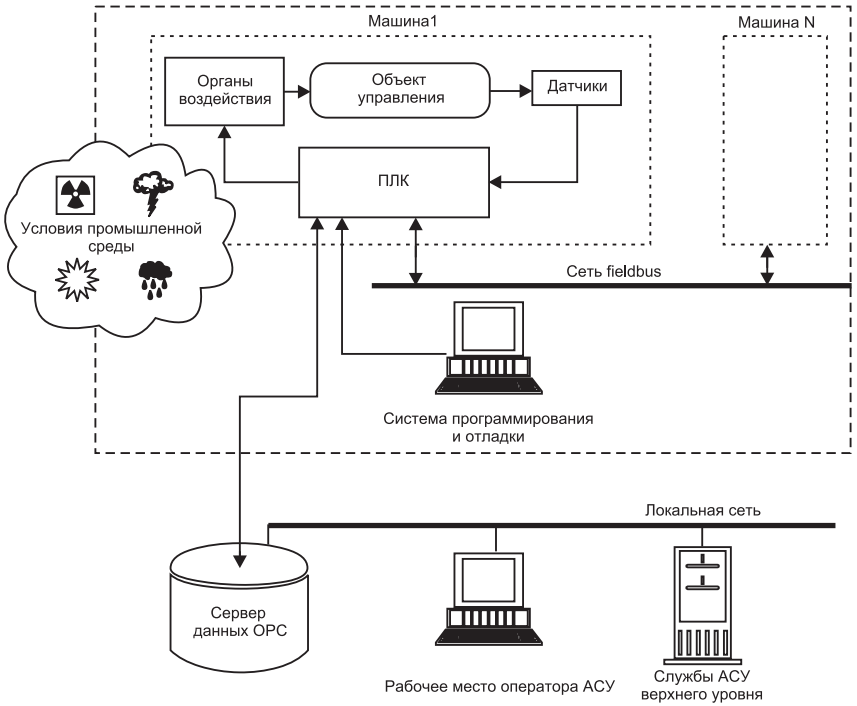

Рис. 1.4. Место ПЛК в АСУ ТП

лом это требования, аналогичные требованиям к ПЛК: режим реального времени, надежность в условиях промышленной среды, ремонтопригодность, простота программирования. Такой класс сетей получил название *промышленных сетей* (fieldbus). Существует масса фирменных реализаций и достаточно много стандартов таких сетей (Bitbus, Modbus, Profibus, CANopen, DeviceNet), позволяющих интегрировать аппаратуру различных фирм, но ни один из них нельзя признать доминирующим.

Благодаря продуктивному развитию средств сетевой интеграции появилась возможность создания *распределенных систем управления*. В 80-х гг. XX в. доминировали ПЛК с числом входов-выходов несколько сотен. В настоящее время большим спросом пользуются микроПЛК с количеством входов-выходов до 64. В распределенных системах каждый ПЛК решает локальную задачу. Задача синхронизации управления выполняется компьютерами среднего звена АСУ. Распределенные системы выигрывают по надежности, гибкости монтажа и простоте обслуживания.

1.6. Доступность программирования

Главным требованием к ПЛК всегда была и остается возможность его эксплуатации существующим техническим персоналом и возможность быстрой замены старого оборудования. Поэтому языки программирования компьютеров и встраиваемых микропроцессорных систем управления плохо подходят для программирования ПЛК. Здесь нужны более простые и наглядные языки, позволяющие излагать задачу в близких к применяемым технологиям категориях. Привлечение же к программированию специализированной фирмы неизбежно порождает зависимость, если реализация не является достаточно прозрачной. Сложный язык программирования ПЛК снижает его шансы на конкурентном рынке существенно больше, чем массогабаритные показатели.

1.7. Программный ПЛК

Программные приложения, имитирующие технологию ПЛК на компьютере (оснащенном платами ввода-вывода), получили название *программный ПЛК* (soft PLC). Программная эмуляция ПЛК удобна тем, что благодаря наличию многозадачной операционной системы можно совместить в одном месте контроллер, среду программирования и систему диспетчерского управления.

Существенный минус такого решения — большое время выхода на рабочий режим после включения питания или зависания компьютера. Особенно опасно, если перезапуск произвел «сторожевой таймер» в автоматическом режиме, в то время как состояние исполнительных механизмов не соответствует исходным позициям. Загрузка операционной системы может отнимать несколько минут, все это время система оказывается неуправляемой. Для ПЛК время «холодного» запуска измеряется миллисекундами.

Для достижения сравнимых с ПЛК технических показателей по надежности компьютер, конечно, должен быть промышленного исполнения (на базе магистралей PC/104 или VME), а не дешевый офисный «no name».

1.8. Рабочий цикл

Задачи управления требуют непрерывного циклического контроля. В любых цифровых устройствах непрерывность достигается за счет применения дискретных алгоритмов, повторяющихся через достаточно малые промежутки времени. Таким образом, вычисления в ПЛК всегда повторяются циклически. Одна итерация, включающая замер, обсчет и выработку воздействия, называется *рабочим циклом* ПЛК. Выполняемые действия зависят от значения входов контроллера, предыдущего состояния и определяются пользовательской программой.

По включению питания ПЛК выполняет самотестирование и настройку аппаратных ресурсов, очистку оперативной памяти данных (ОЗУ), контроль целостности прикладной программы пользователя. Если прикладная программа сохранена в памяти, ПЛК переходит к основной работе, которая состоит из постоянного повторения последовательности действий, входящих в *рабочий цикл*.

Рабочий цикл ПЛК состоит из нескольких фаз.

1. Начало цикла.
2. Чтение состояния входов.
3. Выполнение кода программы пользователя.
4. Запись состояния выходов.
5. Обслуживание аппаратных ресурсов ПЛК.
6. Монитор системы исполнения.
7. Контроль времени цикла.
8. Переход на начало цикла.

В самом начале цикла ПЛК производит физическое чтение входов. Считанные значения размещаются в области памяти входов. Таким образом, создается полная одномоментная зеркальная копия значений входов.

Далее выполняется код *пользовательской программы*. Пользовательская программа работает с копией значений входов и выходов, размещенной в оперативной памяти. Если прикладная программа не загружена или остановлена, то данная фаза рабочего цикла, естественно, не выполняется. Отладчик системы программирования имеет доступ к образу входов-выходов, что позволяет управлять выходами вручную и проводить исследования работы датчиков.

После выполнения пользовательского кода физические выходы ПЛК приводятся в соответствие с расчетными значениями (фаза 4).

Обслуживание аппаратных ресурсов подразумевает обеспечение работы системных таймеров, часов реального времени, оперативное самотестирование, индикацию состояния и другие аппаратно-зависимые задачи.

Монитор *системы исполнения* включает большое число функций, необходимых при отладке программы и обеспечении взаимодействия с системой программирования, сервером данных и сетью. В функции системы исполнения обычно включается: загрузка кода программы в оперативную и электрически перепрограммируемую память, управление последовательностью выполнения задач, отображение процесса выполнения программ, пошаговое выполнение, обеспечение просмотра и редактирования значений переменных, фиксация и трассировка значений переменных, контроль времени цикла и т. д.

Пользовательская программа работает только с мгновенной копией входов. Таким образом, значения входов в процессе выполнения пользовательской программы не изменяются в пределах одного рабочего цикла. Это фундаментальный принцип построения *ПЛК сканирующего типа*. Такой подход исключает неоднозначность алгоритма обработки данных в различных его ветвях. Кроме того, чтение копии значения входа из ОЗУ выполняется значительно быстрее, чем прямое чтение входа. Аппаратно чтение входа может быть связано с формированием определенных временных интервалов, передачей последовательности команд для конкретной микросхемы или даже запросом по сети.

Если заглянуть глубже, то нужно отметить, что не всегда работа по чтению входов полностью локализована в фазе чтения