

СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОМЫШЛЕННОСТИ И ЭКОНОМИКЕ

Якубовский А.О.

Учреждение образование «Белорусский государственный университет информатики и радиоэлектроники», Минск, Беларусь, yakubovskiy.92@bk.ru.

ЭВМ прочно вошли в производственную деятельность, и в настоящее время нет необходимости доказывать целесообразность использования вычислительной техники в системах управления технологическими процессами, проектирования, научных исследований, административного управления, в учебном процессе, банковских расчетах, здравоохранении, сфере обслуживания и т.д. Бурное развитие информационных технологий за последние десятилетия обусловлено высокой потребностью общества в них, в первую очередь потребностями производства. Многие задачи, некогда требующие монотонной и долгой работы, стало возможно решить при помощи компьютера за считанные минуты, что значительно упростило жизнь, помогло сэкономить рабочее время и успешно помогает снизить затраты разного рода на производстве. Использование современных информационных технологий становится возможным даже там, где, казалось бы, они никогда не смогут дополнить или даже полностью заменить труд специалиста.

Введение систем автоматизации в производстве помогает значительно сократить количество наемных рабочих, отдав предпочтение нескольким специалистам в области информационных технологий, которые будут способны решать большинство проблем производства. В большинстве случаев такой подход позволяет добиться существенной экономии средств, несмотря на высокий уровень зарплат подобных специалистов. По всем показателям автоматизированное производство выигрывает, так что современному специалисту важно не только знать о существовании систем автоматизации, но и уметь с ними работать в совершенстве.

Внедрение информационных технологий в сферу производства, торговли, банковского дела первоначально развивалось по пути создания замороженных информационных систем. Термин АСУП (автоматизированная система управления производством), появившийся в 60-е годы был на слуху десятки лет. Однако главная проблема комплексной автоматизации не была решена, но при этом был накоплен опыт разработок подобных систем и подготовлены специалисты, способные решать задачи внедрения информационных технологий в сферу управления бизнесом на современном уровне.

При проектировании АСУП зачастую игнорировались вопросы совместимости, стандартизации, что затрудняло внедрение современных технологий и приводило к большим затратам на модернизацию. В настоящее время, не смотря на специфику предметных областей, широкое распространение получили корпоративные информационные системы (КИС), базирующиеся на принципах корпоративных информационных технологий и современных стандартов.

Выделяют три основных класса задач, решаемых с помощью КИС.

Это задачи:

- формирование отчетных показателей (налоговые службы, статистика, инвесторы и т.д.), получаемых на основе стандартной бухгалтерской и статистической отчетности;
- выработка стратегических управленческих решений по развитию бизнеса на основе базы высокоагрегированных показателей;
- выработка тактических решений, направленных на оперативное управление и решаемых на основе базы частных, высокодетализированных показателей, отражающих различные стороны локальных характеристик функционирования структуры.

Основной трудностью при внедрении КИС является диагностика.

Здесь можно выделить три этапа:

- обследование, системный анализ и оценка существующей структуры и технологий управления;
- разработка новых вариантов организационных структур и технологий управления на основе информационных технологий;
- разработка положения по реорганизации управления, плана внедрения, регламента управленческого документооборота.

Условно выделяют тиражируемые, полузаказные и заказные КИС.

Тиражируемая КИС не требует доработки со стороны разработчика, существует сама по себе, не предоставляет возможности внесения изменений. Такие системы предназначены для малых предприятий.

Заказные системы при существующем уровне информационных технологий ушли в прошлое, они ненадежны, не соответствуют принятым стандартам и с трудом поддаются модернизации. Основная область их применения - производства с очень большой спецификой.

Полузаказные системы являются наиболее гибкими, в большей степени удовлетворяют требованиям заказчика, требуют меньших капитальных затрат. Основная область их применения - крупные предприятия (сотни документов в месяц и более пяти человек в цепочке бизнес-процессов).

В настоящее время на рынке корпоративных систем представлено большое число зарубежных разработок. Учитывая специфику принципов учета, управления, планирования, в белорусской экономике отечественные КИС занимают более прочные позиции.

Кроме КИС следует отметить программные системы, реализующие отдельные функции управления:

- Бухгалтерские программы: 1С: Бухгалтерия, БЭСТ, Турбо-бухгалтер, Парус, Инфо-бухгалтер;
- Системы автоматизации торговли: 1С: Торговля, Парус, БЭСТ 4, Фолио;
- Информационно-справочные системы: Гарант, Консультант Плюс, Кодекс;
- Программы для бизнес-планирования: Project Expert, Microsoft Project, Триумф-аналитик;
- Системы автоматизации складского учета: 1С: Склад, Фолио, БЭСТ, Парус;
- Системы автоматизации документооборота: Дело, Lotus Notus, 1С: Документооборот.

Отдельно от проблем построения КИС рассматривается направление создания автоматизированных систем управления технологическими процессами (АСУ ТП). Актуальность этой проблемы объясняется тем, что в старых системах зачастую выбранные элементы не стыкуются между собой, не удовлетворяют предъявляемым требованиям и нет средств и возможностей для исправления сложившейся ситуации. В настоящее время в области АСУ ТП господствующей является концепция открытых систем на основе системной интеграции, базирующаяся на следующих принципах:

- совместимость программно-аппаратных средств различных фирм-производителей снизу-вверх;
- комплексная проверка и отладка всей системы на стенде фирмы интегратора на основе спецификации заказчика.

В большинстве случаев АСУ ТП представляют двухуровневую систему управления. Нижний уровень включает контроллеры, обеспечивающие первичную обработку информации, поступающей непосредственно с объекта управления. Программное обеспечение контроллеров обычно реализуется на технологических языках типа языка релейно-контактных схем.

Верхний уровень АСУ ТП составляют мощные компьютеры, выполняющие функции серверов баз данных и рабочих станций, обеспечивающих хранение, анализ и обработку всей поступающей информации, а также взаимодействие с оператором. Основой программного

обеспечения верхнего уровня являются пакеты SCADA (Supervision Control and DATA Acquisition).

Наиболее ярко концепция открытых систем прослеживается в открытой модульной архитектуре контроллеров - OMAC (Open Module Architecture Controls), разработанной фирмой General Motors. Близкие к ним концепции предложены европейскими (European Open System Architecture for Control within Automation Systems - OSACA), японскими (Japan International Robotics and Factory - IFORA, Japan Open System Environment for Controller Architecture - OSEC) и американскими (Technologies Enabling Agile Manufacturing - TEAM Projects) организациями. Содержание OMAC-требований заключается в основных терминах:

- Open - открытая архитектура, обеспечивающая интеграцию аппаратного и программного обеспечения;

- Modular - модульная архитектура, позволяющая использовать компоненты в режиме Plug and Play.

- Scaleable - масштабируемая архитектура, позволяющая легко изменять конфигурацию для конкретных задач;

- Economical - экономичная архитектура;

- Maintainable - легко обслуживаемая архитектура.

Аппаратная платформа контроллеров базируется на миниатюрных PC-совместимых компьютерах, обладающих высокой надежностью, быстродействием, совместимостью в силу «родственности» с компьютерами верхнего уровня. Операционная среда PC-контроллеров также должна удовлетворять требованиям открытости.

Здесь наиболее распространенной является операционная система QNX (фирма QSSL, Канада). Архитектура QNX является открытой, модульной, легко модифицируемой. Спецификой работы с контроллерами является использование языков технологического программирования, описывающих сам технологический процесс и ориентированных на работу не программистов, а технологов. Накопленный опыт работы с подобными языками обобщен в стандарте IEC 1131-3, где определены пять основных языковых средств:

- SFG - язык последовательных функциональных схем;

- LD - язык релейных диаграмм;

- FDB - язык функциональных блоковых диаграмм;

- ST - язык структурированного текста;

- IL - язык инструкций.

Основной целью стандарта было повышение скорости и качества разработки программ для программируемых логических контроллеров (ПЛК), а также создание языков программирования, ориентированных на технологов, обеспечение соответствия ПЛК идеологии открытых систем, исключение этапа дополнительного обучения при смене типа ПЛК.

Системы программирования, основанные на МЭК 61131-3, характеризуются следующими показателями:

- надежностью создаваемого программного обеспечения. Надежность обеспечивается тем, что программы для ПЛК создаются с помощью специально предназначенной для этого среды разработки, которая содержит все необходимые средства для написания, тестирования и отладки программ с помощью эмуляторов и реальных ПЛК, а также множество готовых фрагментов программного кода;

- возможностью простой модификации программы и наращивания ее функциональности;

- переносимостью проекта с одного ПЛК на другой;

- возможностью повторного использования отработанных фрагментов программы;

- простотой языка и ограничением количества его элементов.

Языки МЭК 61131-3 появились не как теоретическая разработка, а как результат анализа множества языков, уже используемых на практике и предлагаемых рынку

производителями ПЛК. Стандарт устанавливает пять языков программирования со следующими названиями:

- структурированный текст (ST - Structured Text) рисунок 1;
- последовательные функциональные схемы (SFC - "Sequential Function Chart") рисунок 1;

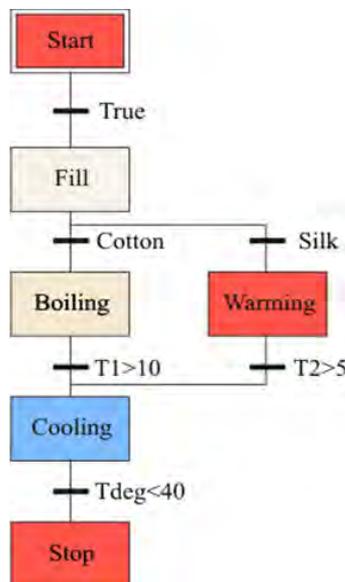


Рисунок 1 - Пример программы на языке SFC.

- диаграммы функциональных блоков (FBD - Function Block Diagram);
- релейно-контактные схемы, или релейные диаграммы (LD - Ladder Diagram);
- список инструкций (IL - Instruction List).

Графическими языками являются SFC, FBD, LD. Языки IL и ST являются текстовыми.

В стандарт были введены несколько языков (а не один) для того, чтобы каждый пользователь мог применить наиболее понятный ему язык. Программисты чаще выбирают язык IL (похожий на ассемблер) или ST, похожий на язык высокого уровня Паскаль; специалисты, имеющие опыт работы с релейной логикой, выбирают язык LD, специалисты по системам автоматического управления (САУ) и схемотехники выбирают привычный для них язык FBD.

Выбор одного из пяти языков определяется не только предпочтениями пользователя, но и смыслом решаемой задачи. Если исходная задача формулируется в терминах последовательной обработки и передачи сигналов, то для нее проще и нагляднее использовать язык FBD. Если задача описывается как последовательность срабатываний некоторых ключей и реле, то для нее нагляднее всего будет язык LD. Для задач, которые изначально формулируются в виде сложного разветвленного алгоритма, удобнее будет язык ST.

Языки МЭК 61131-3 базируются на следующих принципах [Lewis]:

- вся программа разбивается на множество функциональных элементов - Program Organization Units (POU), каждый из которых может состоять из функций, функциональных блоков и программ. Любой элемент МЭК-программы может быть сконструирован иерархически из более простых элементов;
- стандарт требует строгой типизации данных. Указание типов данных позволяет легко обнаруживать большинство ошибок в программе до ее исполнения;
- имеются средства для исполнения разных фрагментов программы в разное время, с разной скоростью, а также параллельно. Например, один фрагмент программы может

сканировать концевой датчик с частотой 100 раз в секунду, в то время как второй фрагмент будет сканировать датчик температуры с частотой один раз в 10 сек;

- для выполнения операций в определенной последовательности, которая задается моментами времени или событиями, используется специальный язык последовательных функциональных схем (SFC);

- стандарт поддерживает структуры для описания разнородных данных. Например, температуру подшипников насоса, давление и состояние "включено-выключено" можно описать с помощью единой структуры "Pomp" и передавать ее внутри программы как единый элемент данных;

- стандарт обеспечивает совместное использование всех пяти языков, поэтому для каждого фрагмента задачи может быть выбран любой, наиболее удобный, язык;

- программа, написанная для одного контроллера, может быть перенесена на любой контроллер, совместимый со стандартом МЭК 61131-3.

Любой ПЛК работает в циклическом режиме. Цикл начинается со сбора данных с модулей ввода, затем исполняется программа ПЛК и оканчивается цикл выводом данных в устройства вывода. Поэтому величина контроллерного цикла зависит от времени исполнения программы и быстродействия процессорного модуля.

Список использованных источников:

1. [Электронный ресурс]. – Режим доступа: http://bookasutp.ru/Chapter9_3.aspx. – (Дата обращения 11.11.2016).

2. [Электронный ресурс]. – Режим доступа: <http://technologies.su> (Дата обращения 12.11.2016).