

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ СИСТЕМ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ СВАРКИ

Ф. Н. КИСИЛЕВСКИЙ, д-р техн. наук, В. В. ДОЛИНЕНКО, канд. техн. наук
(Ин-т электросварки им. Е. О. Патона НАН Украины)

Рассмотрено использование объектно-ориентированного программирования при разработке систем управления технологическим процессом сварки. В качестве примера сложного объекта управления выбран фрагмент электронно-лучевой сварочной установки. Выполнен объектно-ориентированный анализ, построена диаграмма отношений и синхронизации, предложено концептуальное решение для сетевой системы управления АСУ ТП сварки.

Ключевые слова: автоматизированная система управления, объектно-ориентированное программирование, декомпозиция, абстракции, иерархия, модель управления, диаграмма отношений и синхронизации, сетевая архитектура, реальный масштаб времени, марковские процессы, графы состояний, надежность программного обеспечения

Современные автоматизированные системы управления технологическим процессом (АСУ ТП) сварки представляют собой сложные информационно-управляющие системы с развитым математическим обеспечением, в которых предусмотрено взаимодействие с оператором-сварщиком или оператором-технологом. Часто в процессе сварки задействованы разнообразные агрегаты и механизмы, имеющие сложные алгоритмы функционирования. В связи с этим очень важна задача создания программного обеспечения (ПО) для АСУ ТП сварки. Разработчики современных АСУ сложными сварочными объектами должны учитывать достаточно жесткие требования, предъявляемые к ПО:

обеспечение повышенной надежности функционирования ПО совместно с аппаратными средствами АСУ ТП в условиях больших электромагнитных и других помех;

обеспечение достаточной «дружелюбности» интерфейса оператора ПО путем использования графических интерактивных объектов и средств;

минимальные сроки разработки.

В настоящее время разработчики АСУ ТП все чаще обращают внимание на объектно-ориентированное проектирование (object oriented design, далее OOD). Опыт использования такой технологии в инженерных проектах показывает ее высокую эффективность [1].

OOD включает следующие методы:

1) декомпозиция. В отличие от алгоритмической используется объектно-ориентированная декомпозиция с абстракциями объектов в предметной области сварки;

2) выделение абстракций и формирование классов объектов, имеющих сходные функции;

3) иерархия. В данном случае имеется в виду организация иерархической системы классов объектов, организация самой модели объекта управления и архитектуры АСУ ТП. Такой подход позволяет

обеспечить более устойчивую обработку программных потоков данных в очень сложных системах управления.

OOD включает несколько методологий. Во-первых, объектно-ориентированный анализ (object oriented analysis, далее OOA), который представляет собой методологию декомпозиции и создания моделей как объекта управления, так и системы управления в целом. В данном случае требования к моделям формируются на основе понятий классов и объектов из области сварки.

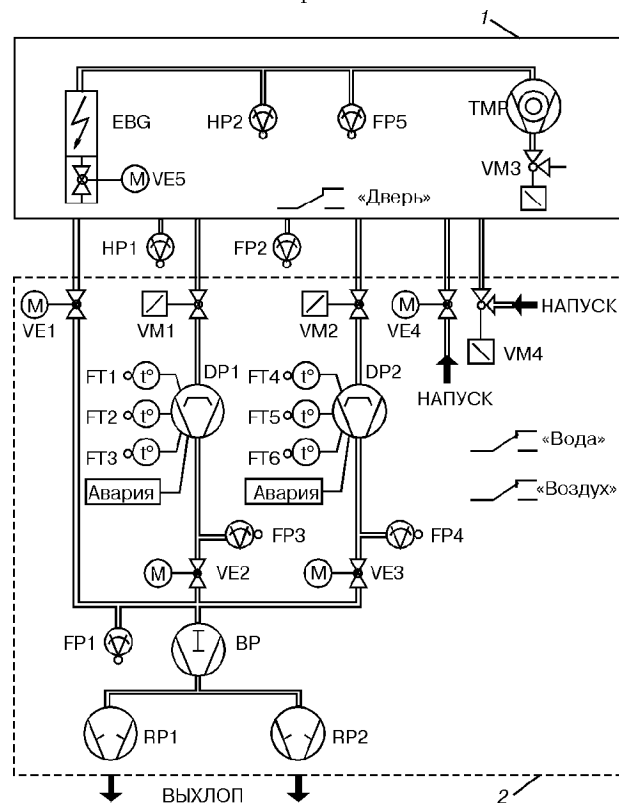


Рис. 1. Пример сложного объекта управления АСУ ТП сварки: 1 – вакуумная сварочная камера; 2 – откачная система (EBG – электронная пушка; VE5 – шибер пушки; HP1, HP2 – высоковакуумные датчики давления; FP1... FP5 – низковакуумные датчики давления; VM1... VM4 – вакуумные клапаны; VE1... VE4 – электромеханические задвижки; FT1... FT6 – низкотемпературные датчики температуры; BP – насос Рутца; DP1, DP2 – диффузионные, RP1, RP2 – форвакуумные, TMP – турбомолекулярный насосы)

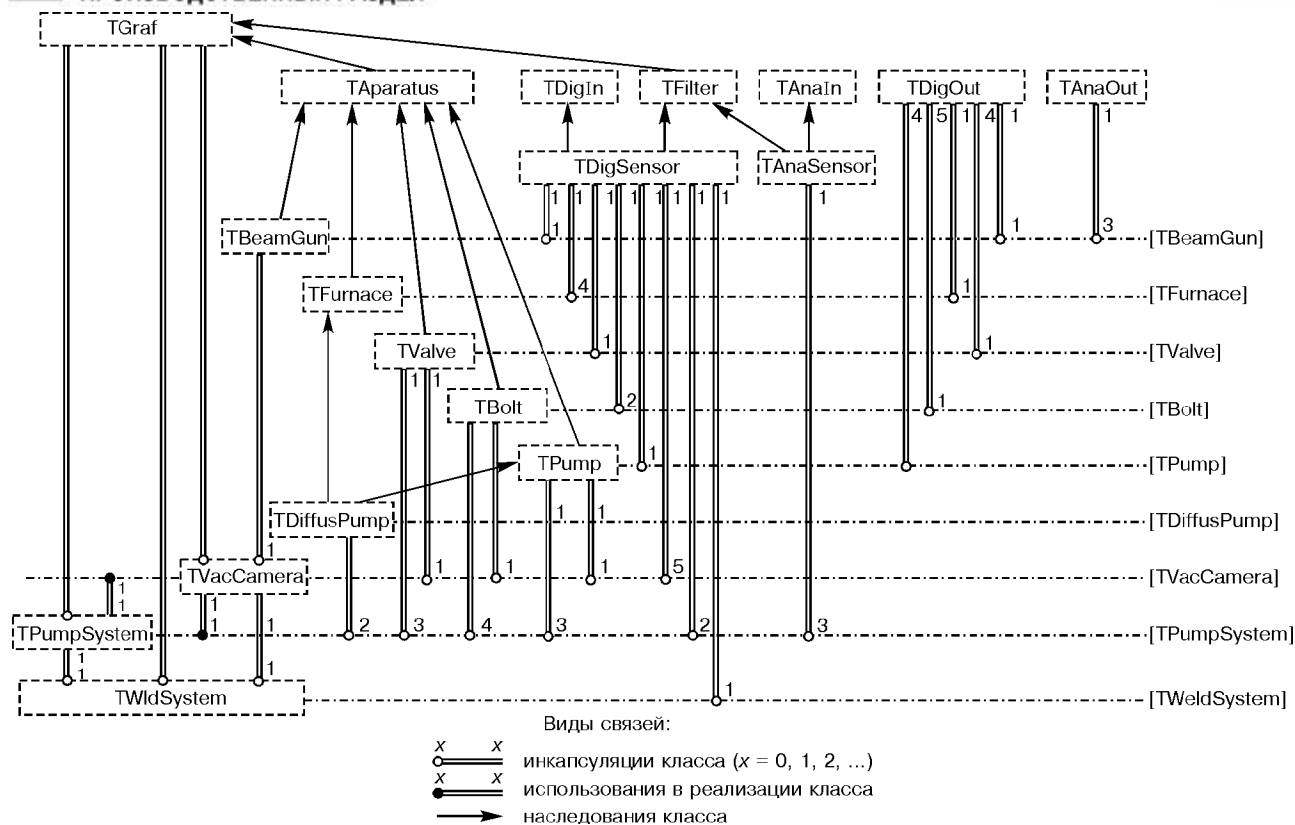


Рис. 2. Диаграмма классов объекта управления в АСУ ТП сварки

Во-вторых, объектно-ориентированное программирование (object oriented programming, далее ООР), которое представляет собой методологию программирования или реализации абстрактных моделей, т. е. программирование с использованием объектно-ориентированных языков. Таким образом, рабочая программа представляет собой систему объектов, каждый из которых является реализацией определенного класса и характеризуется собственным алгоритмом поведения и взаимодействием с другими объектами.

Как известно, основными методами ООД являются наследование, полиморфизм и инкапсуляция. Поэтому объекты могут наследовать свои свойства от объектов других классов, изменять свое поведение в зависимости от внешних условий и содержать необходимые данные, процедуры или другие объекты.

Остановимся на следующих практических аспектах применения ООД подхода при разработке ПО для АСУ ТП сварки:

- 1) использование ООА;
- 2) построение диаграммы отношений и синхронизации объектов;
- 3) использование объектно-ориентированных языков программирования таких, как С++ или Java;
- 4) использование графов состояний, описывающих поведение объектов в реальном масштабе времени;
- 5) использование результатов теории графов для автоматизированного тестирования и отладки ПО.

Для того чтобы проиллюстрировать возможности использования ООД подхода в задачах автоматизации сварочных процессов, рассмотрим конкретный случай автоматизации сложного объекта управления при электронно-лучевой сварке (ЭЛС). Ни

в коей мере не претендуя на полноту обсуждения проблемы автоматизации в ЭЛС, рассмотрим лишь фрагмент сложного объекта управления (рис. 1) — вакуумную сварочную камеру и откачную систему [2]. Конечно, в данном случае мы вынуждены учитывать и то, что эти две ключевые подсистемы невозможно рассматривать изолированно от других подсистем электронно-лучевой установки. Поэтому при рассмотрении вопроса синхронизации объектов будут упоминаться также подсистемы приводов перемещения электронной пушки и манипулятора изделия, а также электронно-лучевая аппаратура (ЭЛА). К сожалению, ввиду ограниченного объема статьи нет возможности детально рассмотреть эти подсистемы, впрочем, как и другие, которые могут входить в состав электронно-лучевой установки: телевизионная следящая система [3], подсистема мониторинга процесса сварки и др.

Несмотря на это, выбранный в качестве демонстрационной модели фрагмент объекта управления ЭЛС дает нам представление об уровне сложности и разнообразии абстракций, которыми оперируют разработчики ПО.

В данной статье делается попытка показать, что ООД подход позволяет выполнить более полную и детальную постановку задачи разработки ПО системы управления АСУ ТП сварки. При этом существенно ограничивается субъективный аспект, свойственный эвристическим, интуитивным решениям при выборе архитектуры и декомпозиции системы управления, расширяется база формальных правил, что в итоге гарантирует получение устойчивых непротиворечивых результатов на каждом этапе реализации проекта.

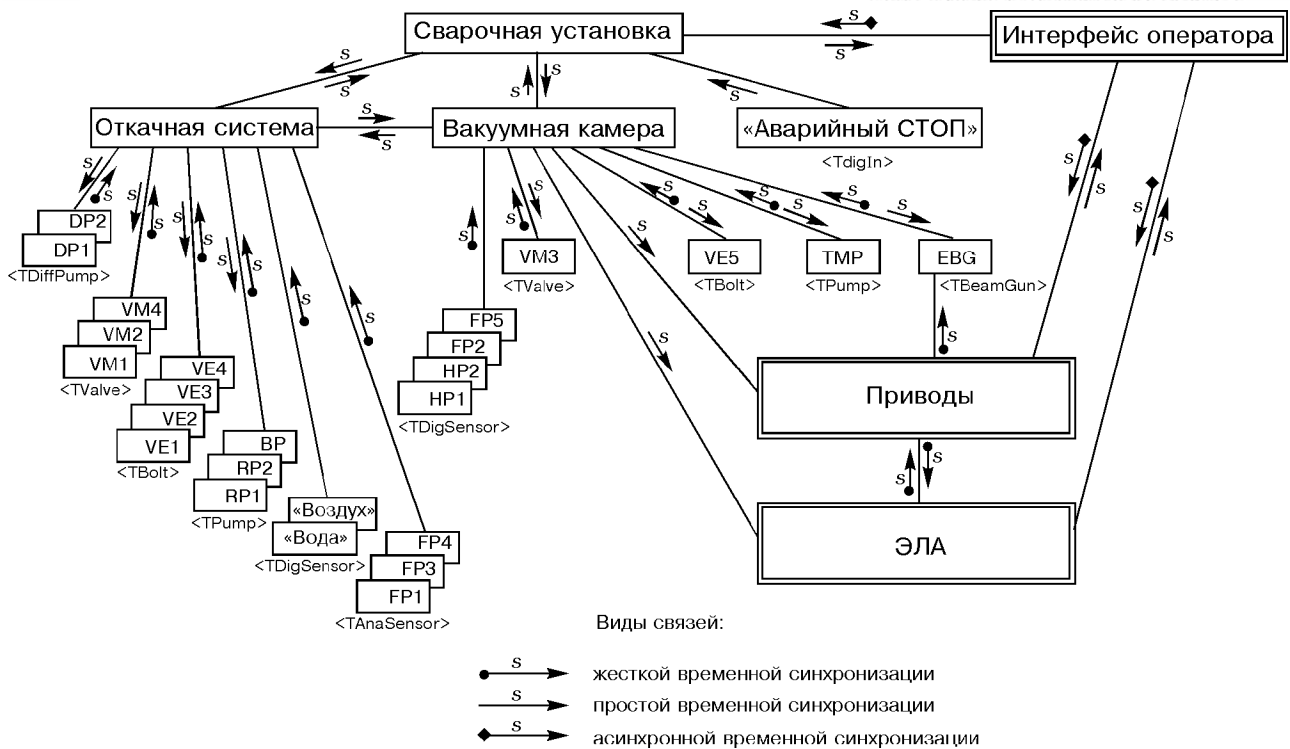


Рис. 3. Диаграмма отношений и синхронизации объектов ПО АСУ ТП сварки

Опуская этапы анализа задачи автоматизации ЭЛС, а также описание процесса группировки и выделения классов объектов (агрегатов, исполнительных механизмов и подсистем), представим результат предварительного анализа в виде диаграммы классов модели объекта управления (рис. 2). Здесь использованы следующие обозначения классов: TGraf — графов состояний, которые описывают поведение объектов во времени; TApparatus — родительский для всех устройств и механизмов, задает флаг аварии, временные характеристики включения-выключения агрегата, хранит режим работы (ручной, автоматический или отключен), а также флаг отработки команды управления переключения состояния; TDigIn — сигналов дискретного ввода; TDigOut — сигналов дискретного вывода; TFilter — фильтров, которые используются для фильтрации входных сигналов (медианный фильтр, низких частот и др.); TAnaIn — сигналов аналогового ввода; TDigSensor — датчиков дискретного характера типа «включено-выключено»; TAnaSensor — датчиков аналогового характера (давления и др.); TFurnace — нагревательных элементов, который используется в некоторых видах насосов; TValve — вакуумных клапанов, в том числе пневматических и электромагнитных; TBolt — задвижек, которые в отличие от клапанов имеют два датчика положения («открыто» и «закрыто»); TPump — насосов; TDiffusPump — диффузионных насосов, наследует свойства TPump и TFurnace; TVacCamera — вакуумной сварочной камеры, использует в своем составе три класса объектов (TValve, TBolt, TPump); TBeamGun — электронной пушки, использующей три аналоговых сигнала управления (ток пучка, фокусировки и накала) и один дискретный сигнал управления; TPumpSystem — откачной системы; TWeldSystem — сварочной системы

(так же, как и TVacCamera и TPumpSystem, инкапсулирует TGraf в целях реализации необходимых режимов работы АСУ ТП). Отметим, что классы TVacCamera и TPumpSystem используют в своих реализациях ссылки друг на друга. Это необходимо для осуществления совместной работы объектов этих классов в единой сварочной системе.

Поскольку невозможно детально остановиться на каждом объекте, некоторые положения представим лишь в виде результатов.

Диаграмма классов объекта управления позволяет синтезировать модель объекта управления в объектных понятиях конкретного вида сварки.

На следующем этапе ООД-проектирования создается модель АСУ ТП сварки, которая содержит модель объекта управления и обычно имеет сетевую иерархическую структуру. Типовая схема построения иерархии уровней управления следующая:

система → подсистемы → агрегаты и механизмы → датчики (аналоговые, дискретные и пр.) → сигналы (аналоговые, дискретные и пр.).

При построении объектной модели используется принцип распределенности «интеллекта», заключающийся в том, что каждый объект в модели функционирует по собственному алгоритму, который свойствен объектам его класса. Такой поведенческий алгоритм удобно представить в виде направленного графа состояний или сети Петри с необходимыми временными задержками, которые задают поведение объекта в реальном масштабе времени.

На рис. 3 приведена диаграмма отношений и синхронизации, реализующая модель объекта управления, где специальными обозначениями показан характер отношений между объектами, что и определяет функционирование АСУ ТП в реальном масштабе времени. Наиболее часто применяют следующие виды синхронизации:

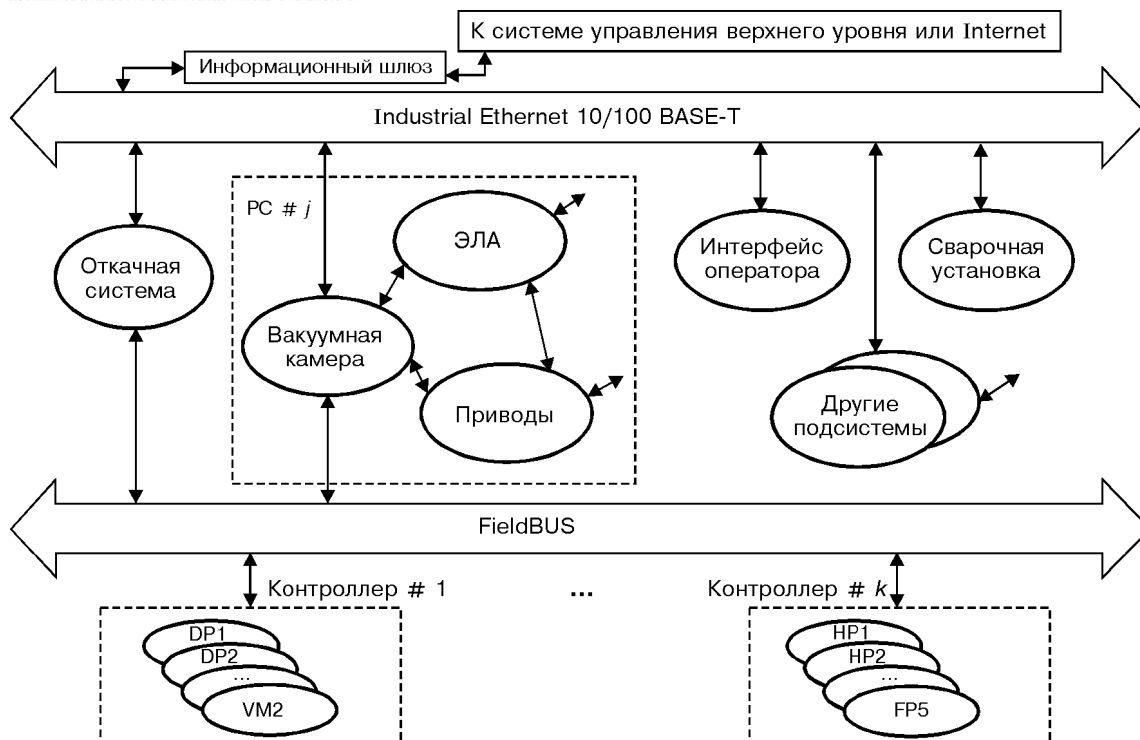


Рис. 4. Концептуальная схема сетевой объектно-ориентированной системы управления АСУ ТП сварки (обозначения см. на рис. 1)

синхронная (жесткая, простая) — действие выполняется независимо от готовности приемного сигнала;

асинхронная — реализуется ожидание взаимной готовности передающего и принимающего объектов;

задержанная — вид асинхронной с оговоренным временем ожидания ответа;

отсроченная — вид асинхронной, но в случае неготовности принимающего передающий не выполняет операцию.

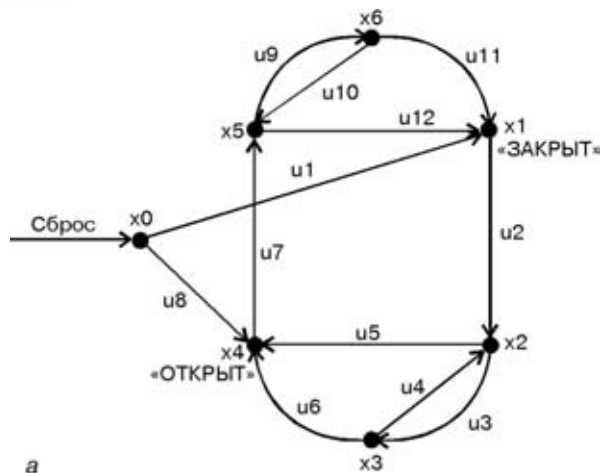
Диаграмма связей и синхронизации объектов используется для декомпозиции АСУ ТП на подсистемы. В зависимости от ее сложности и вида ТП сварки может быть выбрана распределенная, цент-

рализованная, сетевая или другая архитектура. Под сетевой структурой АСУ ТП понимаем такую организацию, при которой информационные каналы связи и протоколы обмена информацией имеют системоорганизующее значение.

Современной тенденцией в построении АСУ ТП является применение сетевых архитектур, в которых могут использоваться несколько видов информационных сетевых каналов связи. На верхних уровнях АСУ ТП чаще всего применяются сети типа Ethernet (Industrial) и модемная связь с протоколами обмена TCP/IP. На уровне контроллеров и датчиков используют FieldBUS-сети, например, PROFIBUS, CANBUS или BITBUS.

Соответствие задач программирования и теории графов

Задача программирования	Задача теории графов
Оптимизация программ, в том числе выделение фрагментов (структуризация) и декомпозиция	Нахождение доминаторов
	Перечисление биокомпонент, контуров, интервалов, гамаков, линейных компонент
	Нахождение вложенных зон графа
	Нахождение вложенных контуров
Проверка правильности программ	Интервальное представление графа
	Нахождение транзитивного и обратного транзитивного замыкания
	Нахождение замыкания относительно множества вершин
Определение порядка обработки операторов при потоковом анализе	Перечисление путей, контуров
	Нумерация вершин
Тестирование программ	Построение длинных последовательностей вершин
	Нахождение покрытия:
	вершин путями
	дуг путями
	требуемых путей путями
	Нахождение кратчайшего пути при наличии дополнительных ограничений
	Нахождение множества фундаментальных циклов
Определение цикломатического числа	



а

```

#define TRUE 1
class TVacuumValve : public TValve, public TGraf {
//... тело определения класса
public:
    enum TStates {x0, x1, x2, x3, x4, x5, x6};
    int State;
    virtual void Graf();
}
void TVacuumValve::Graf() {
    if(Сброс == TRUE) State = x0;
    switch (State) {
        case x0:
            if(u1 == TRUE) State = x1;
            if(u8 == TRUE) State = x4;
            break;
        case x1: // Клапан ЗАКРЫТ
            if(u2 == TRUE) State = x2;
            break;
        case x2:
            if(u3 == TRUE) State = x3;
            if(u5 == TRUE) State = x4;
            break;
        case x3:
            if(u4 == TRUE) State = x2;
            if(u6 == TRUE) State = x4;
            break;
        case x4: // Клапан ОТКРЫТ
            if(u7 == TRUE) State = x5;
            break;
        case x5:
            if(u9 == TRUE) State = x6;
            if(u12 == TRUE) State = x1;
            break;
        case x6:
            if(u11 == TRUE) State = x1;
            break;
        default : State = x0;
            break;
    }
}
    
```

б

Рис. 5. Пример реализации марковской модели объекта «ВАКУУМНЫЙ КЛАПАН»: а — граф состояний; б — текст программы на языке С++

Обычно при выборе типов сетевых интерфейсов и параметров сетевых протоколов следует обеспечить необходимые минимальный такт синхронизации и уровень надежности пересылки данных между объектами АСУ ТП.

В том случае, когда используются сетевые протоколы со случайным характером разрешения кон-

фликтов (например, Ethernet) или же с синхронной передачей данных (например, PROFIBUS-DP), отработка заданий в АСУ ТП может иметь вероятностный характер и сильно зависеть от выбранного такта синхронизации. Например, исследование TCP/IP протокола в сети Ethernet 10Base-T показывает, что для такта синхронизации менее 10 мс отработка заданий подсистем может иметь вероятностный характер, в то время как для протоколов DP сети PROFIBUS — детерминированный.

В нашем случае необходимо обеспечить как простую синхронизацию между объектами (подсистемами), так и жесткую с тактом менее 10 мс. Наличие последней обусловлено необходимостью совместной работы подсистем перемещения электроно-лучевой пушки (вакуумная камера), манипулятора изделия и источника питания (ЭЛА), что в общем случае и определяет качество сварки.

На основе изложенного выше можно предложить концептуальную схему сетевой системы управления объектно-ориентированной АСУ ТП сварки (рис. 4). Структура такой системы управления имеет несколько уровней взаимодействия подсистем. На верхнем расположены объекты подсистем: «Интерфейс оператора», «Сварочная установка», «Откачная система», «Вакуумная камера», «ЭЛА», «Приводы» и др. У всех подсистем верхнего уровня есть информационные каналы связи с сетью Industrial Ethernet 10/100 Base-T, которая имеет шлюз в сеть более высокого уровня (например, заводскую) или в Internet. На схеме показано, что объекты подсистем «Вакуумная камера», «ЭЛА» и «Приводы» желательно реализовать в одном контроллере (компьютере) для обеспечения жесткой временной синхронизации. На нижнем уровне расположены контроллеры с моделями объектов агрегатов, механизмов, датчиков и сигналов. Объекты «Вакуумная камера» и «Откачная система» имеют прямые информационные связи с объектами нижнего уровня через протоколы обмена по интерфейсу FieldBUS. Предложенная система взаимосвязи объектов в общем случае инвариантна количеству и типам контроллеров, поэтому разработку программного и аппаратного обеспечения можно вести параллельно.

Такая архитектура в целом не противоречит типовой двухуровневой архитектуре АСУ ТП ЭЛС, которая в настоящее время используется в промышленных образцах технологических систем [4].

Отметим, что в настоящее время тенденция создания объектно-ориентированных «открытых» систем управления АСУ ТП возрастает [5–8]. Для разработки и моделирования таких систем созданы специальные языковые средства проектирования, например система проектирования UML — Universal Modeling Language [9].

Рассмотрим проблему, возникающую при реализации абстрактных моделей, с помощью методов и средств ООР. Как методология ООД дает программистам математический аппарат, который может быть положен в основу построения управляющих программ и реализаций абстракций объекта управления, динамического по своей природе. Этот математический аппарат базируется на использовании марковской схемы моделирования и аналитического

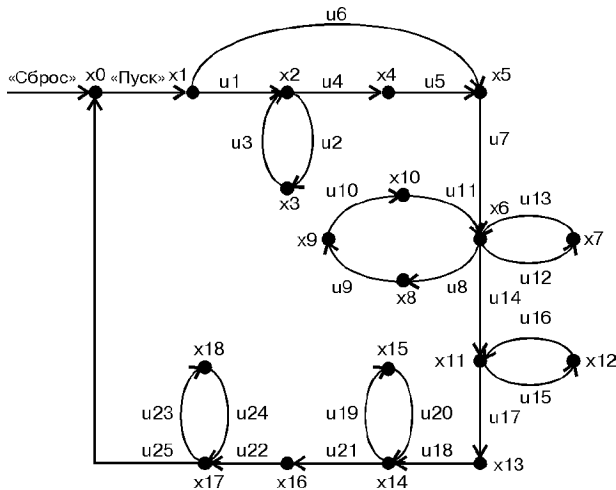


Рис. 6. Пример сложного управляющего графа

исследования сложных систем типа систем массового обслуживания. При этом ключевым понятием является марковский процесс, представляющий собой модель с конечным множеством состояний, описывающих функционирование системы. В связи с этим модель марковского процесса может быть использована как обобщенная для динамической системы [10]. Для реализации марковских процессов могут быть использованы направленные графы состояний. Разработка моделей объекта управления и АСУ ТП в целом ведется от абстрактных моделей марковских процессов более высокого порядка к таковым более низкого порядка. Поэтому ООД в результате дает иерархическую систему марковских схем, реализованных в виде направленных графов состояний.

В иерархической схеме базовых классов (см. рис. 2) каждый объект инкапсулирует объект типа TGraf, который задает поведение объекта в реальном масштабе времени. Реализацию объекта типа TGraf на языке машинного программирования можно формализовать, например, с помощью стандартного блока «switch ... case of». Для организации работы такой модели в реальном масштабе времени необходимо жестко тактировать граф с периодом интервала реального масштаба времени. Это следует обеспечить соответствующим выбором как операционной системы контроллера, так и компилятора языка программирования. Например, для работы в реальном масштабе времени графов состояний с тактом 0,1 с можно использовать операционную систему MS Windows NT и систему программирования Borland C++Builder, а для работы с тактом менее 1 мс необходима операционная система OS-9 и компилятор языка Microsoft Ultra C++.

В качестве примера рассмотрим марковскую модель объекта «ВАКУУМНЫЙ КЛАПАН», граф состояний которого приведен на рис. 5, а. После сброса модель переходит в состояние x0. Имеются также два устойчивых состояния: x1 — «ЗАКРЫТ» и x4 — «ОТКРЫТ». В процессе работы граф может временно находиться в состояниях x2, x3, x5 и x6, которые можно характеризовать как промежуточные. Для реализации данной модели в виде программы используем язык C++. На рис. 5, б показан

фрагмент кода программы, соответствующий графу на рис. 5, а.

Можно отметить преимущество программирования объектов по такой формальной методике, которое заключается в наглядности и взаимоднозначности модели объекта и программного кода. Это позволяет существенно сократить сроки разработки и повысить надежность работы ПО АСУ ТП сварки.

Известный среди программистов афоризм, утверждающий, что «каждая последняя выявленная ошибка в программе является предпоследней», отражает важную проблему поиска и устранения ошибок в ПО АСУ ТП сварки. Существенным преимуществом ООД подхода является то, что при конструировании и отладке программ возникающие задачи можно свести к задачам теории графов либо использовать их в качестве основы для решения. К ним относятся задачи тестирования и проверки правильности программы, оценки сложности и времени исполнения и др. В таблице показана связь между такими задачами и задачами теории графов из [11].

В качестве примера рассмотрим управляющий граф технологического режима «ОТКАЧКА ВАКУУМНОЙ КАМЕРЫ» (рис. 6). Вершина x1 является входом, а x0 — выходом графа. Пусть необходимо построить программу тестирования данного графа. Для этого найдем минимальное покрытие вершин путями

$$\mu_1 = [\text{«Сброс»}, \text{«Пуск»}, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{16}, u_{17}, u_{18}, u_{19}, u_{20}, u_{21}, u_{22}, u_{23}, u_{24}, u_{25}] \quad (1)$$

и дуг путями

$$\phi_1 = \mu_1, \quad (2)$$

$$\phi_2 = [\text{«Сброс»}, \text{«Пуск»}, u_6, u_7, u_{14}, u_{17}, u_{18}, u_{21}, u_{22}, u_{25}]. \quad (3)$$

Таким образом, имеем два минимальных покрытия μ_1 и ϕ_2 , которые можно положить в основу тестовой программы для рассматриваемого управляющего графа.

Дополнительным преимуществом данного подхода является также то, что программисты вынуждены документировать свои управляющие графы и точно планировать процесс отладки и тестирования программ. Это позволяет обеспечить полноценное сопровождение и обслуживание по ПО АСУ ТП сварки.

Выводы

1. Современные АСУ ТП сварки представляют собой сложные информационно-управляющие системы с развитым математическим обеспечением. При разработке ПО таких систем основной проблемой является выбор подходящей формальной методики проектирования, которая бы позволяла трансформировать «внешнюю сложность» самой задачи автоматизации во «внутреннюю простоту» архитектуры и программного кода ПО АСУ ТП сварки. ООД подход, использующий методы декомпозиции, абстракции и иерархии, позволяет реализовать принцип «внутренней простоты» как для самого ПО, так и для АСУ ТП в целом. В результате уменьшаются сроки разработки и внедрения, а также достигается

повышенная надежность функционирования ПО разработанной системы управления АСУ ТП сварки.

2. Использование результатов теории графов позволяет формализовать процедуры отладки и тестирования ПО АСУ ТП сварки, что дает возможность разрабатывать программы со сложной логикой работы, которые практически не имеют ошибок.

1. Буч Г. Объектно-ориентированный анализ и проектирование (с примерами приложений на C++): Пер. с англ. — М.: Бином, 2000. — 560 с.
2. Четверко А. И., Назаренко О. К., Кайдалов А. А. Тенденции развития // Машиностроение. Энциклопедия. — Т. IV-6: Оборудование для сварки. — Гл. 1: Оборудование для электронно-лучевой сварки. — М.: Машиностроение, 1999. — С. 368–378.
3. Назаренко О. К. Вторично-электронная следящая система с электромеханическим приводом // Автомат. сварка. — 1998. — № 12. — С. 47–50.

The use of object-oriented design in development of CAM control systems for welding is considered. A fragment of an electron beam welding unit was selected as an example of a complex object of control. Object-oriented analysis is performed, a diagram of relations and synchronization is plotted and a conceptual solution for a network control system for welding CAM system is suggested.

4. Назаренко О. К., Кайдалов А. А. Микропроцессорные системы локального управления параметрами процесса электронно-лучевой сварки и электромеханическим комплексом // Машиностроение. Энциклопедия. — Т. IV-6: Оборудование для сварки. — М.: Машиностроение, 1999. — С. 360–363.
5. Рыженко А. И., Свирид У. Н. Объектно-структурированная АСУ ТП мукомольного завода // Совр. технологии автоматизации. — 2000. — № 3. — С. 46.
6. A welding cell with its own Web site / T. P. Quinn, J. D. Gilsinn, W. Ripply // Welding J. — 2000. — № 1. — P. 46–48.
7. Networking of welding applications / W. Ripply, J. Gilsinn, L. Flitter // Ibid. — 2000. — № 1. — P. 49–53.
8. No other PC-based shape-cutting control touches it. BURN 10 shape-cutting motion control // Ibid. — 1999. — № 6. — P. 41.
9. Буч Г., Рамбо Д., Джекобсон А. Язык UML. Руководство пользователя: Пер. с англ. — М.: ДМК, 2000. — 432 с.
10. Энциклопедия кибернетики: В 2 т. — Киев: Укр. сов. энцикл., 1974. — Т. 1. — 607 с.
11. Евстигнеева В. А. Применение теории графов в программировании / Под ред. А. П. Ершова. — М.: Наука, 1985. — 352 с.

Поступила в редакцию 05.10.2000,
в окончательном варианте 22.01.2001



Вторая
Международная
специализированная
выставка

Экспо Сварка 2001

23 – 26 октября 2001 г.
Москва, Экспоцентр, пав. №4, "Форум"

ОСНОВНЫЕ РАЗДЕЛЫ ВЫСТАВКИ

- технологии и оборудование для сварки и термической резки
- сварочные материалы и принадлежности, их производство
- автоматизация сварочных работ
- контрольно-измерительная техника
- техническая диагностика, приборы и методы неразрушающего контроля испытаний
- спецодежда и средства защиты сварщиков, охрана окружающей среды
- сертификация сварочного оборудования и технологий
- информационное обеспечение, подготовка кадров, менеджмент

НОВЫЕ РАЗДЕЛЫ

- технологии обработки поверхности
- гальваническое оборудование и технологии
- технологии термического напыления

Уважаемые господа!
Приглашаем Вас принять участие во Второй Международной специализированной выставке сварочного оборудования и родственных технологий "ЭКСПОСВАРКА • 2001"

ОРГАНИЗАТОРЫ:
Министерство экономического развития и торговли Российской Федерации, РНТСО, М.С.И. при содействии ЗАО "Экспоцентр", ООО "Мессе Сервис", АО "Евроэкспо"





290 4013, 202 9349

WWW.MSI-VYSTAVKI.RU

В РАМКАХ ВЫСТАВКИ ПРОЙДУТ КОНФЕРЕНЦИИ И СЕМИНАРЫ