

ISSN 1811-1807

# ҒЫЛЫМИ ЖУРНАЛ

Б. ТОРАЙҒЫРОВ АТЫНДАҒЫ ПАВЛОДАР МЕМЛЕКЕТТІК УНИВЕРСИТЕТІ



3'2007



ПМУ хабаршысы  
Вестник ПГУ

**ФИЗИКА-МАТЕМАТИКАЛЫҚ СЕРИЯ**

43

С. Торайғыров атындағы Павлодар мемлекеттік университетінің ғылыми журналы  
Научный журнал Павлодарского государственного университета им. С. Торайгырова

1997 жылы құрылған  
Осыман в 1997 г.

С. Торайғыров  
атындағы ПМУ-ның  
докладчик С.Б.  
атындағы ғылыми  
КІТАПХАНАСЫ

# ПМУ ХАБАРШЫСЫ

## ВЕСТНИК ПГУ

СЕРИЯ ФИЗИКО-МАТЕМАТИЧЕСКАЯ

**3** 2007

Научный журнал Павлодарского государственного университета  
им. С. Торайгырова

### **СВИДЕТЕЛЬСТВО**

о постановке на учет средства массовой информации  
№ 4532-Ж

выдано Министерством культуры, информации и общественного согласия  
Республики Казахстан  
31 декабря 2003 года

---

---

#### **Главный редактор:**

Тлукунов С.К., д-р физ.- мат. наук, проф.

#### **Редакционная коллегия:**

Абдильдин М.М., д-р физ.-мат. наук, академик НАН РК  
Бахтыбаев К.Б., д-р физ.-мат. наук, проф.  
Данаев Н.Т., д-р физ.-мат. наук, академик НИА РК  
Оспанов К.Н., д-р физ.-мат. наук, проф.  
Отелбаев М.О., д-р физ.-мат. наук, академик НАН РК  
Тюреходжаев А.Н., д-р физ.-мат. наук, проф.  
Уалиев Г.У., д-р физ.-мат. наук, проф., академик НАН РК  
Сейтахметова Г.Н. (тех. редактор)  
Жукенов М.К., (отв. секретарь)

---

---

За достоверность материалов и рекламы ответственность несут авторы и рекламодатели.  
Мнение авторов публикаций не всегда совпадает с мнением редакции.  
Редакция оставляет за собой право на отклонение материалов.  
Рукописи и дискеты не возвращаются.  
При использовании материалов журнала ссылка на «Вестник ПГУ» обязательна

## **ВЗАИМОДЕЙСТВИЕ ДРАЙВЕРА И МОДЕЛИ ТЕХНОЛОГИЧЕСКОГО АГРЕГАТА ПРИ ИМИТАЦИОННОМ МОДЕЛИРОВАНИИ СИСТЕМ УПРАВЛЕНИЯ**

**А.Ю. Ковылин**

**Павлодарский государственный университет  
им. С. Торайгырова г. Павлодар,**

*Мақалада басқару жүйесінің имитацианды моделдеу кезінде технологиялық агрегаттың моделі мен драйверінің өзара әрекет етуі зерттеледі.*

*В статье исследуется алгоритм взаимодействия драйвера и модели технологического агрегата при имитационном моделировании систем управления.*

*In the article is researched the algorithm of interconnection of driver and model of technological aggregation under simulation technique of control system.*

В современных условиях взаимоотношений между предприятиями и потребителями становится очевидной необходимость постоянно идти в ногу с быстроразвивающимися технологическими новинками, которые сильно облегчают и упрощают производственный процесс. Техническое перевооружение легкой промышленности, ускоренное внедрение новых интенсифицированных технологических процессов невозможно без использования современного высокопроизводительного оборудования и средств вычислительной техники, комплексной автоматизации.

Разработка и внедрение автоматизированных систем управления технологическими процессами (АСУТП) на

предприятиях позволяет решать задачи оперативного управления производством на более высоком уровне по сравнению с традиционным применением локальных средств автоматизики.

Основными инструментами создания АСУТП являются средства разработки, а именно прикладное программное обеспечение для программируемых логических контроллеров (ПЛК) и для пользовательских приложений операторов, которые осуществляют удаленное оперативное диспетчерское управление технологическими процессами. К таким инструментам относятся продукты GE Fanuc Proficy Machine Edition и Simlicity Plant Edition. Proficy Machine Edition отвечает в данном случае за контроллерную часть, а соответственно Simlicity Plant Edition отвечает за часть SCADA (система сбора данных и оперативного диспетчерского управления). Эффективность при построении системы АСУТП достигается при использовании следующего алгоритма:

- Максимально учитывая возможности программных средств для программирования контроллеров, логика, связанная непосредственно с оборудованием и его работой, должна разрабатываться в контроллерной части приложений.

- Все, что не возможно осуществить на контроллере, а также назначение для SCADA – систем (отображение полной информации по технологическим процессам, ведение и архивирование информации в технологической базе данных), должно разрабатываться в части SCADA приложений.

Следовательно, для имитационного моделирования процесса работы технологических агрегатов необходимо программное обеспечение Proficy Machine Edition.

Обычно при разработке систем АСУТП возникает проблема с отладкой определенного драйвера, разработанного для некоторой единицы оборудования. В таких случаях разработчики применяют имитационное моделирование,

основанное на описанном ниже алгоритме, который включает в себя входные, выходные параметры и сам обработчик этих параметров (см. рисунок 1).

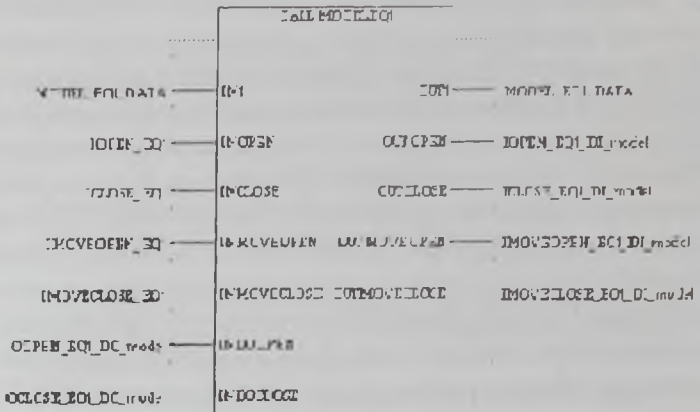


Рисунок 1 – Блок модели работы задвижки с входными и выходными параметрами

Драйверы оборудования содержат алгоритм, включающий логику работы единицы оборудования. Входными и выходными параметрами могут быть сигналы DI (дискретные сигналы, определяющие состояние оборудование), AI (аналоговые сигналы, определяющие количественный параметр, взятый с оборудования), DO (дискретные сигналы, определяющие конкретное действие для оборудования) и AO (аналоговые сигналы, определяющие количественный параметр, устанавливаемый на оборудование). Определяющее действие драйвер оказывает на AO и DO сигналы, которые непосредственно идут на оборудование с целью задать тот или иной режим работы или установить какой-либо параметр. Также существуют во входных и выходных параметрах внутренние структуры данных (ВСД), которые содержат важные данные для алгоритма.

Для моделей оборудования ситуация немного меняется, на входных и выходных параметрах можно увидеть также DI, AI, DO, AO сигналы и ВСД, но определяющими являются сигналы AI, DI. В самом обработчике также обрабатываются эти сигналы, и результаты выдаются на выходные параметры.

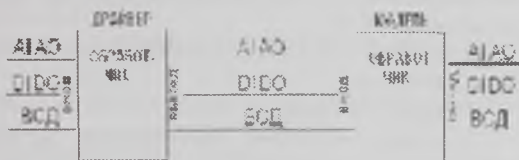


Рисунок 2 - Взаимодействие драйвера оборудования и модели задвижки

Рассмотрим модель для единицы оборудования – задвижка (см. рисунок 2). Для симуляции открывания и закрывания задвижки требуется определить максимальное и минимальное время открывания и закрывания задвижки (мс).

```
MOVE_TIME_MAX := 12000.00;
```

```
MOVE_TIME_MIN := 1000.00 ;
```

Далее определяется минимальная и максимальная позиция открывания и закрывания задвижки.

```
POS_MIN := 0.0; // 0%
```

```
POS_MAX := 1000.0; // 100%
```

Перенос из ВСД значения текущей позиции задвижки, время перемещения задвижки в локальные переменные.

```
WORD2_AS_REAL(IN1:= IN1[0],IN2 := IN1[1], Q => LPosition);
```

```
WORD2_AS_REAL(IN1 := IN1[2],IN2 := IN1[3], Q => IMoveTime);
```

Определение времени цикла.

```
ICycleTime := DeltaT;
```

Далее должна производиться проверка на превышение текущего времени передвижения задвижки максимальной

и минимальной границы.

Вычисление значения перемещения задвижки.

```
LDelta := (POS_MAX - POS_MIN) * ((ICycleTime)) /
((IMoveTime));
```

Алгоритм обработки открытия и закрытия задвижки включает в себя установку флага ITmpBool путем логических операций при использовании входящих в обработчик сигналов DI и DO.

```
ITmpBool := (NOT INDIOPEN[0].x[0]) AND
(INDOOPEN = 1);
```

```
KMOpen := ITmpBool;
```

```
ITmpBool := (NOT INDICLOSE[0].x[0]) AND
(INDOCLOSE = 1);
```

```
KMClose := ITmpBool;
```

Производится симуляция равномерного открытия и закрытия задвижки путем приращения к текущей позиции задвижки переменной LDelta.

```
IF (KMOpen = 1) THEN
```

```
  LPosition := LPosition + LDelta;
```

```
ELSIF (KMClose = 1) THEN
```

```
  LPosition := LPosition - LDelta;
```

```
END_IF;
```

Выдача на DO сигнал значения 0, если задвижка еще не достигла минимальной или максимальной границы, что соответствует состоянию открытой или закрытой задвижки.

```
OUTOPEN := 0;
```

```
OUTCLOSE := 0;
```

Выдача на DO сигнал соответствующего значения при достижении текущего положения задвижки минимальной или максимальной границы.

```
IF (LPosition >= POS_MAX) THEN
```

```
  LPosition := POS_MAX;
```

```
OUTOPEN := 1;
```



```
ELSIF (LPosition <= POS_MIN) THEN  
LPosition := POS_MIN;  
OUTCLOSE := 1;  
END_IF;
```

Перенос из локальных переменных в ВСД уже прошедших один цикл переменных.

```
REAL_AS_WORD2(IN: = LPosition, Q1 => OUT1[0],  
Q2 => OUT1[1]);
```

```
REAL_AS_WORD2(IN: = IMoveTime, Q1 => OUT1[2],  
Q2 => OUT1[3]);
```

Выдача на DI сигналы значения, равного открывающейся или закрывающейся задвижке.

```
OUTMOVEOPEN := KMOpen ;
```

```
OUTMOVECLOSE := KMClose;
```

Таким образом, рассмотренный выше алгоритм позволяет не используя технологическое оборудование, смоделировать процесс его работы, исключив ряд проблем, связанных с взаимодействием драйвера агрегата и модели его функционирования.

## ЛИТЕРАТУРА

1. Емельянов А.И. Капник О.В. Проектирование систем автоматизации технологических процессов / Справочное пособие по содержанию и оформлению проектов. – 3-е изд., перераб. и доп. – М.: Энергоатомиздат, 1983. – 400 с.: ил.

2. Петров И.В. Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования / Под ред. проф. В. Дьяконова.- М.: СОЛОН-Пресс, 2004. -256 с.: ил.

3. Шлеер С. Меллор С. Объектно-ориентированный анализ: моделирование мира в состояниях/ Пер. с англ. Киев: Диалектика, 1993.

4. Трудоношин В.А. Пивоварова Н.В. Математические модели технических объектов. В 9 кн. Кн. 4. – М.: Высшая

школа, 1986.

5. Controllers&I/O Series 90-30: официальный сайт производителя. <URL:<http://www.geindustrial.com/cwc/products?id=S90-30&famid=12>>

6. Справочная система: <<http://www.Technolink.ru/>>