

**МИНИСТЕРСТВО ПО РАЗВИТИЮ ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИЙ И КОММУНИКАЦИЙ РЕСПУБЛИКИ  
УЗБЕКИСТАН**

**ФЕРГАНСКОГО ФИЛИАЛА ТАШКЕНТСКОГО  
УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**Факультет “Компьютерный инжиниринг”**

**Кафедра “Компьютерные системы”**

# **РЕФЕРАТ**

по предмету **“Технология настройки микропроцессорных систем”**

*На тему: Комплексная отладка микропроцессорных систем*

**Подготовил:**

студент группы 642-11 Шералиев Т.

**Принял:**

Ахунджанов У.

*Фергана 2015г.*

## **План**

- 1) Введение
- 2) Отладка
- 3) Функции средств отладки
- 4) Автономная отладка
- 5) Отладка программ
- 6) Заключение
- 7) Список использованной литературы

## Введение

Как правило, микропроцессорная система - это система реального времени, т. е. корректность ее функционирования зависит от времени выполнения отдельных программ и скорости работы аппаратуры. Поэтому система считается отлаженной после того, как рабочие программы правильно функционируют на действительной аппаратуре системы в реальных условиях. Дополнительным свойством, которым должны обладать средства комплексной отладки по сравнению со средствами автономной отладки, является возможность управления поведением МПС и сбора информации о ее поведении в реальном времени.

Тенденция развития средств отладки микропроцессорных систем состоит в объединении свойств нескольких приборов в одном комплексе, в создании универсальных средств, пригодных для автономной отладки аппаратуры, генерации и автономной отладки программ и комплексной отладки системы. Эти средства позволяют вести разработку и отладку, постепенно усложняя аппаратуру и программы. При этом разработка, изготовление и отладка планируются поэтапно с нарастанием сложности; новая, неотлаженная аппаратура и программа вводятся в создаваемую систему, присоединяются к проверенной ее части.

Если отладка программ ведется с использованием эмуляционного ОЗУ, а затем изготавливаются микросхемы ПЗУ, то микропроцессорная система должна быть протестирована.

Средства отладки на последних этапах не должны влиять на правильность функционирования системы, вносить задержки, дополнительные нагрузки.

При комплексной отладке наряду с детерминированным используется статистическое тестирование, при котором МПС проверяется при изменении исходных переменных в соответствии со статистическими законами работы источников информации. Полнота контроля работоспособности проектируемой системы возрастает за счет

расширения диапазона возможных сочетаний переменных и соответствующих им логических маршрутов обработки информации.

### **Отладка**

О правильности функционирования микропроцессорной системы на уровне "черного ящика" с полностью неизвестной внутренней структурой можно говорить лишь тогда, когда произведены ее испытания, в ходе которых реализованы все возможные комбинации входных воздействий, и в каждом случае проверена корректность ответных реакций. Однако исчерпывающее тестирование имеет практический смысл лишь для простейших элементов систем. Следствием этого является тот факт, что ошибки проектирования встречаются при эксплуатации, и для достаточно сложных систем нельзя утверждать об их отсутствии на любой стадии жизни системы. В основе почти всех методов испытаний лежит та или иная гипотетическая модель неисправностей, первоисточником которой служат неисправности, встречающиеся в практике. В соответствии с моделью в рамках каждого метода предпринимаются попытки создания тестовых наборов, которые могли бы обеспечить удовлетворительное выявление моделируемых неисправностей. Любой метод тестирования хорош ровно настолько, насколько правильна лежащая в его основе модель неисправности.

Важным моментом является правильный выбор соотношения между степенью общности модели, стоимостью и степенью сложности формирования и прогона тестов, ориентированных на моделируемые неисправности. Чем конкретнее модель, тем легче создать для нее систему тестов, но тем выше вероятность того, что неисправность останется незамеченной. Если же модель неисправностей излишне общая, то из-за комбинаторного возрастания числа необходимых тестовых наборов и/или времени вычислений, требуемого для работы алгоритмов формирования

тестов, она станет непрактичной и пригодной только для несложных систем.

### **Функции средств отладки**

Сроки и качество отладки системы зависят от средств отладки. Чем совершеннее приборы, имеющиеся в распоряжении инженера-разработчика, тем скорее можно начать отладку аппаратуры и программ и тем быстрее обнаружить ошибки, локализовать источники, устранение которых обойдется дороже на более позднем этапе проектирования.

Средства отладки должны:

- 1) управлять поведением системы или/и ее модели на различных уровнях абстрактного представления;
- 2) собирать информацию о поведении системы или/и ее модели, обрабатывать и представлять на различных уровнях абстракции;
- 3) преобразовывать системы, придавать им свойства контролепригодности;
- 4) моделировать поведение внешней среды проектируемой системы.

Под управлением поведением системы или ее модели понимаются определение и подача входных воздействий для запуска или останова системы или ее модели, для перевода в конкретное состояние последних. Чтобы определить место субъективной неисправности, которая может быть внесена на любой стадии проектирования, необходимо уметь собирать информацию о поведении системы и представлять ее в тех формах, которые приняты для данного проекта. Например, это могут быть временные диаграммы, принципиальные электрические схемы, язык регистровых передач, ассемблер и др.

В общем случае нельзя локализовать источник ошибки проектируемой системы, имея информацию о поведении системы только на ее внешних выводах, поэтому проектируемую систему преобразовывают. Например, прежде чем изготавливать однокристалльную микроЭВМ с теми или иными "зашивками" ПЗУ, программы отлаживают на эмуляционном кристалле, у

которого магистраль выведена на внешние контакты и вместо ПЗУ установлено ОЗУ.

### **Автономная отладка**

Процесс отладки прототипа проектируемой системы должен начинаться с отладки аппаратуры и отладки программ.

Отладка аппаратуры предполагает тестирование отдельных устройств микропроцессорной системы - процессора, ОЗУ, контроллеров, блока питания, генератора тактовых импульсов путем подачи тестовых входных воздействий и приема ответных реакций. Тестовые входные воздействия и ответные реакции определяются, исходя из спецификаций на устройства, а также структурных схем устройств. При этом проверяются реальная аппаратура прототипа, спецификации, структурные схемы и отлаживаются тесты. После отладки отдельных устройств проверяется их взаимодействие. Процессор системы работает с шинами адресов, данных и управления. Анализируя их сигналы, можно проконтролировать выполнение программы в процессоре.

Поскольку ША и ШД синхронные, их работу лучше всего проверить с помощью методов логических состояний. Перед анализом последовательностей данных на этих шинах необходимо удостовериться в том, что сигналы, управляющие взаимодействием процессора с другими устройствами, выдаются в соответствующем порядке. Поскольку ШУ состоит из линий, работающих асинхронно, необходимо просматривать сигналы многих линий в течение одного и того же промежутка времени. Для анализа асинхронной работы линий управления необходимо также наблюдать за сигналами на них при возникновении определенного события, чтобы можно было четко разделить и идентифицировать различные состояния линий. Например, среди сигналов ШУ могут быть сигналы длительностью всего несколько наносекунд, но могут также возникать кратковременные ложные узкие импульсы, вызванные перекрестными помехами или шумами.

После того как доказана работоспособность ШУ, проводится дальнейшая проверка работы аппаратуры при различных режимах адресации процессора и кодах выбираемых данных. Для проверки выполнения процессором инструкций разрабатывается тестовая программа, которая помещается в ОЗУ или ППЗУ. При этом проверяется временная диаграмма сигналов и прохождения данных в системе (как осуществляется передача информации по отношению к строб-сигналам). Если тестовая программа - системный проверяющий тест пройдет успешно, можно утверждать, что автономно аппаратура отлажена.

При автономной отладке аппаратуры могут потребоваться приборы, умеющие: а) выполнять функции аналогового прибора, т. е. измерять напряжение и ток; воспроизводить форму сигнала, подавать импульсы определенной формы и т. д.; б) подавать последовательность сигналов одновременно на несколько входов в соответствии с заданной временной диаграммой или заданным алгоритмом функционирования аппаратуры, представленным в спецификации на языке высокого уровня, или другим способом; собирать значения сигналов многих линий в течение одного и того же промежутка времени, который определяется задаваемыми, программируемыми событиями - комбинацией или последовательностью сигналов на линиях, например, ложным сигналом на линии; обрабатывать и представлять собранную информацию либо в виде временной диаграммы, либо в виде диаграммы или таблицы логических состояний, либо на языке высокого уровня, например, языке регистровых передач.

Для автономной отладки аппаратуры широко используются осциллографы, вольтметры, амперметры, частотомеры, генераторы импульсов, позволяющие отлаживать аппаратуру на схемном уровне. Чтобы автономно отладить аппаратуру МПС на более высоком уровне, применяют логические анализаторы, генераторы слов, пульты, комплексы диагностирования.

## **Отладка программ**

Отладка программ микропроцессорной системы проводится, как правило, на тех же ЭВМ, на которых велась разработка программ, и на том же языке программирования, на котором написаны отлаживаемые программы, и может быть начата на ЭВМ даже при отсутствии аппаратуры МПС. При этом в системном программном обеспечении ЭВМ должны находиться программы (интерпретаторы или эмуляторы), моделирующие функции отсутствующих аппаратных средств. Так, разработка и автономная отладка программных средств может вестись на больших ЭВМ, миниЭВМ, микроЭВМ, система команд которых не совпадает с системой команд используемого микропроцессора. Кроме того, при отладке программ может отсутствовать внешняя среда микропроцессорной системы, ее также необходимо моделировать.

Проверка корректности программ, т.е. проверка соответствия их внешним спецификациям, осуществляется тестированием. Программы проверяются на функционирование с различными исходными данными. Результаты функционирования программ сравниваются с эталонными значениями.

Отладка программ подразделяется на следующие этапы: планирование отладки; составление тестов и задания на отладку; исполнение программ; информирование о результатах исполнения программ по заданным исходным данным; анализ результатов, обнаружение ошибок и локализация неисправностей.

Существует два способа начального тестирования программ: пошаговый режим и трассировка программ.

В пошаговом режиме программа выполняется по одной команде за один раз, а пользователь анализирует содержимое памяти, регистров и т.д., чтобы проверить, соответствуют ли результаты ожидаемым.

Пошаговый режим может быть трудоемким, если средства отладки будут требовать отдельных команд после каждого шага для того, чтобы показать необходимую информацию в понятном для пользователя виде. Имеются средства отладки, автоматически показывающие после каждого шага содержимое регистров процессора и ячеек памяти, используемых в последней команде, и несколько следующих команд. Пошаговый режим является весьма мощным средством предварительного тестирования, так как позволяет обнаруживать неисправности, прежде чем они существенно исказят программу и данные. Кроме того, неоднократно проходя отдельными шагами через один и тот же участок объектной программы, программист может легко изменять содержимое регистров и ячеек памяти, особенно если средства отладки имеют динамически обновляемый дисплей, и тем самым проверить работу программы в разных условиях. Этот интерактивный режим отладки программы позволяет разработчику постоянно упреждать, что будет делать его программа, и оперативно обнаруживать ошибку. Однако пошаговый режим с автоматическим показом результатов возможен только тогда, когда средства отладки содержат в своем составе дисплей с прямым доступом в память, так как после каждого шага на экране дисплея нужно показывать большой объем информации.

Исполнение программ осуществляется по шагам последовательно во времени и в соответствии с заданиями, содержащимися в операторах. При этом производится переработка значений переменных и определение оператора приемника. Если в ходе исполнения программы регистрируется последовательность операторов, реализуемых на каждом шаге процесса, то получается трасса или маршрут исполнения программы, который для конкретной программы зависит только от значений исходных данных.

Трассировка программ больше пригодна для отладочных средств, имеющих медленный, последовательный терминал. Программа-отладчик выполняет непрерывно команду за командой и выводит содержимое

регистров процессора на терминал после каждого шага. Некоторые отладчики выводят также на терминал команды в дизассемблерной форме. Но при этом содержимое памяти не выводится на терминал и разработчик должен сам делать выводы об изменениях в ней. Отслеживание программы продолжается автоматически до тех пор, пока не будет остановлено извне. Результатом трассировки программы будут данные на экране дисплея или же в случае использования в качестве терминала печатающего устройства - длинная распечатка с ходом выполнения программы. Программист, анализируя эти данные, может обнаружить ошибки. Трассировка программ не дает, однако, возможности изменять содержимое памяти и регистров и может послужить причиной того, что программа разрушит себя или свои данные прежде, чем отслеживание будет остановлено.

Отдельные участки программы после проверки, используя пошаговый режим или трассировку, можно объединить и проверить с помощью установки контрольных точек, вводимых в программу и прерывающих ее исполнение, для передачи управления программе-отладчику. По контрольным точкам можно по желанию выполнить избранные участки программы и проанализировать результаты. Контрольные точки устанавливаются обычно для конкретной команды, но в некоторых системах предусматриваются прерывания программы при чтении или записи данных в определенные ячейки памяти. Возможны и более сложные условия прерывания программы.

Расстановка контрольных точек предполагает, что программист связывает с ней точный адрес памяти. Для некоторых отладчиков программист задает абсолютный шестнадцатеричный адрес. Последние отладчики допускают символьные значения адресов, которые программист определяет в исходной программе; это позволяет значительно экономить время, распечатывая после каждого редактирования и транслирования программы новую копию листинга.

При тестировании можно планировать проверку всех возможных маршрутов исполнения программы для разных исходных переменных. Однако это реализуемо только для очень простых программ небольшого объема при малых диапазонах изменения исходных данных. Поэтому при планировании отладки программ применяют критерии полноты тестирования, которые, однако, не гарантируют полной проверки программ. Выбор критерия зависит от наличия ресурсов для тестирования и структурной сложности отлаживаемой программы. Критерии характеризуются глубиной контроля программ и объемом проверок.

В процессе отладки основная часть неисправностей в программах обнаруживается и затем устраняется. Однако всегда возможен пропуск нескольких неисправностей.

Средства отладки программ должны:

- а) управлять исполнением программ (останавливать, изменять порядок, запускать и т. д.);
- б) собирать информацию о ходе выполнения программы;
- в) обеспечивать обмен информацией (диалог) между программистом и ЭВМ на уровне языка программирования;
- г) моделировать работу отсутствующих аппаратных средств микропроцессорной системы.

### **Заключение**

Существуют пять основных приемов комплексной отладки микропроцессорной системы:

- 1) останов функционирования системы при возникновении определенного события;
- 2) чтение (изменение) содержимого памяти или регистров системы;
- 3) пошаговое отслеживание поведения системы;
- 4) отслеживание поведения системы в реальном времени;
- 5) временное согласование программ.

Комплексная отладка завершается приемосдаточными испытаниями, показывающими соответствие спроектированной системы техническому заданию. Для проведения комплексной отладки МПС используют логические анализаторы и комплексы: оценочные, отладочные, развития микропроцессоров, диагностирования, средств отладки.

### **Список использованной литературы**

**Шафрин, Ю.** Информационные технологии. В 2-х ч.: учеб.пособие / Ю. Шафрин. - М. : БИНОМ. Лаборатория знаний, 2002 -**Ч. 2:** Офисная технология и информационные системы. - 2002. - 336 с. - 1 экз.

**Информационные технологии управления:** учеб. пособие / Под ред. Ю. М. Черкасова. - М. : ИНФРА-М, 2001. - 216 с. - (Высш. образование). - 1 экз.

<http://www.wikipedia.org>