

УНИВЕРСАЛЬНЫЙ АЛГОРИТМ ДИАГНОСТИКИ НЕИСПРАВНОСТЕЙ ЭЛЕКТРОННЫХ ЦИФРОВЫХ МАШИН

Г.А. Миронов, В.А. Ермилов, А.И. Шишмарев
(Москва)

Большинство работ по диагностике ограничиваются задачей построения методов локализации неисправностей только некоторого узкого класса. Такие упрощающие допущения значительно облегчают формализацию процесса поиска. Однако вероятность возникновения произвольных неисправностей всегда имеет место в реальных условиях эксплуатации ЭЦМ.

В настоящей работе ставится следующая задача — построить алгоритм определения местоположения и характера любых неисправностей в ЭЦМ в предположении, что имеет место схемный или программный контроль машинных операций. Алгоритм должен быть пригоден для всех способов контроля, определяющих неверно исполняющуюся операцию. Такой контроль имеет место в следующих случаях:

- а) обычное исполнение рабочих программ на машинах с пооперационным схемным контролем;
- б) исполнение испытательных программ с фиксированными результатами всех элементарных тестов, составляющих испытательные программы [1];
- в) исполнение рабочих программ при программном моделировании выполняемых в ней операций и сравнении результатов работы реальных схем и моделей;
- г) параллельное исполнение рабочих программ на двух или более ЭЦМ, входящих в единый комплекс.

Алгоритм диагностики, таким образом, должен быть применим для всех способов контроля за исключением случаев, когда имеет место программный контроль не каждой операции, а значительных групп операций (двойной счет, проверка контрольных соотношений, проверка последовательностей операций, создающих тяжелые режимы и так далее).

Способ отыскания неисправностей должен являться общим для всех видов неисправностей (сбойные, перемежающиеся, постоянные, одиночные и многократные; все возможные логические неисправности составляющих элементов и т.д.). Диагностика возникающих неисправностей в процессе эксплуатации машины должна производиться непрерывно, с использованием информации, накапливаемой в течение требуемого интервала времени наблюдения и учета. Величина этого интервала различна для разных неисправностей и в общем случае определяется характером каждой диагностируемой неисправности. При этом должна обеспечиваться преемственность проведения диагностического поиска от предыдущего к последующему моменту проявления неисправности.

Можно ожидать, что метод, удовлетворяющий перечисленным требованиям, позволит практически исключить возникновение в машине постоянных отказов, вызванных естественным старением элементов. "Стареющие" элементы будут обнаружены заблаговременно в процессе работы машины при нарастании потока сбоев вследствие непрерывного ухудшения рабочих характеристик этих элементов.

Исходными данными для проведения диагностического анализа являются фиксируемая схемным или программным контролем одиночная или групповая ошибка, то-есть проявление неисправностей на одном или нескольких выходах схемы после исполнения машинной операции, а также операнды машинной операции и исходные состояния регистров перед началом выполнения машинной операции. При этом проведение анализа принципиально может осуществляться либо непосредственно после проявления неисправности путем прерывания исполнения рабочей программы, либо во время ближайшей профилактики. Последняя ситуация может иметь место как при проявлении случайного сбоя, так и в многомашинном комплексе при появлении отказа, когда отказавшая машина выводится в резерв для её профилактики, все исходные данные для диагностического анализа, производимого во время профилактики, должны автоматически запоминаться.

И в том и в другом случае проявления каждой неисправности анализируются, тем самым обеспечивается использование всей информации о состоянии машины. Принцип использования всей получаемой информации при каждом проявлении неисправности является основным в предлагаемом методе. При этом первоначальной получаемой информацией для всех неисправностей является искаженный результат.

По характеру проявления неисправностей во времени будем различать сбойную неисправность, перемежающийся отказ и постоянный отказ.

При сбойной неисправности количество получаемой информации зависит только от числа выходов схемы (разрядов), на которых проявилась неисправность, так как никакой другой дополнительной информации получить нельзя, в силу исчезновения этой неисправности.

При этом мы полагаем, что между различными сбойными неисправностями нет никакой связи, то-есть, что причины соответствующих ошибок независимы. В дальнейшем ошибками от независимых сбойных неисправностей будем называть такие проявления этих неисправностей, между которыми не удастся обнаружить связь за интервал времени наблюдения и учета.

В отличие от независимых сбойных неисправностей будем рассматривать перемежающиеся отказы.

Под перемежающимся отказом будем понимать поток одинаковых сбойных неисправностей, проявления этих неисправностей (ошибки) могут быть разными, но причина одна и та же - периодически проявляющаяся неисправность одного и того же элемента. При перемежающемся отказе дополнительной информацией для дальнейшего сокращения сферы предполагаемой неисправности, полученной при обработке первоначальной информации, является информация о последующих проявлениях сбойных неисправностей.

Дополнительной информацией для сокращения сферы, в которой находится предполагаемый постоянный отказ, являются результаты исполнения машинных операций, специально подбираемых для целенаправленного поиска отказавших элементов. Отправной точкой для целенаправленного поиска является первоначальная информация.

Как первоначальная, так и дополнительная информация об-
разуется в момент T при выполнении элементарной машинной
операции R , входящей в машинную операцию Θ (по термино-
логии [2]). Буквой R будем обозначать также выходную ячей-
ку (регистр) элементарной машинной операции (ЭМО).

Через \bar{R} будем обозначать правильный результат элементар-
ной машинной операции, через \tilde{R} - неправильный результат на
регистре R . Через $T-I$ обозначаем время предыдущего проявле-
ния какой-нибудь неисправности, через $T+I$ - время последующе-
го проявления.

Всякое проявление неисправности, первоначальное или до-
полнительное подвергается анализу с помощью специального ал-
горитма - алгоритма общего диагностического анализа.

Алгоритм общего диагностического анализа состоит из двух
частей. Первая часть его служит для построения всех возможных
"маршрутов", по которым может поступать искаженный сигнал на
каждый из тех разрядов регистра, на котором результат отличен
от правильного. Для выполнения этой задачи используется опи-
сание блока - схемы, реализующей ЭМО [2], на некотором языке
и результаты предварительного анализа работы схемы при исход-
ных данных, соответствующих \bar{R} . Некоторые из маршрутов мо-
гут действовать в течение всех K тактов работы машины, за ко-
торые выполняется ЭМО.

Вторая часть алгоритма общего диагностического анализа
осуществляет построение таблицы предположений, содержащей ве-
роятные причины ошибки \tilde{R} . Время работы этого алгоритма оп-
ределяется с одной стороны сложностью схемы, а с другой сто-
роны - ограничениями на класс возможных неисправностей.. Как
было указано, излагаемый метод позволяет определить место лю-
бых неисправностей. При этом, однако, в ряде случаев время
его работы становится весьма большим.

Рассмотрим более подробно те виды неисправностей, которые
могут иметь место в многотактных схемах.

Неисправность некоторого элемента может проявиться все-
го лишь на одном из K тактов, в течение которых формируется
результат ЭМО. Это может произойти вследствие самоустранения
неисправности (при сбое) или потому, что на всех тактах, кро-
ме одного, вырабатываемый элементом сигнал не является суще-
ственным аргументом для результата.

Такие проявления неисправности можно назвать простыми. Более сложные проявления неисправности представляют собой неверную работу элементов на любом числе тактов из K . Если допустить все возможности, то, вообще говоря, для каждого из элементов может иметь место 2^K проявлений неисправности. К этому следует добавить, что одновременно может присутствовать произвольное число неисправных элементов со сложными проявлениями неисправностей.

Если не установлены ограничения на классы возможных неисправностей, то таблица предположений должна составляться для всех возможных неисправностей и их проявлений. Для значительных K время работы алгоритма становится неприемлемо большим. Уменьшения времени можно достигнуть, устанавливая ограничения на предполагаемые неисправности. Первым таким ограничением может явиться допущение, что в таблицу предположений занесены вначале наиболее вероятные неисправности. Если наиболее вероятных неисправностей, могущих быть причиной \tilde{N} , обнаружить не удастся, то производится поиск менее вероятных неисправностей.

Наиболее вероятными можно считать неисправности, проявляющиеся в течение всех K тактов. При явных отказах, проявляющихся при всех режимах работы схемы и фиксированном режиме неисправного элемента только такие неисправности имеют место.

Менее вероятными являются те неисправности из всех 2^K , которые проявляются в течение i тактов подряд ($1 \leq i \leq K$). Число таких неисправностей равно $\frac{K(K+1)}{2} - 1$. Самыми редкими встречающимися являются остальные неисправности из 2^K .

С другой стороны наиболее вероятными по расположению являются неисправности одного элемента, менее вероятными — неисправности в нескольких элементах в порядке возрастания числа подозреваемых элементов. Определение таких предполагаемых неисправностей ведется путем независимого составления таблиц предположений для каждого из неверных разрядов и последующего пересечения полученных таблиц (множеств подозреваемых элементов). Наиболее вероятными считаются неисправности, входящие в непустое пересечение всех образованных множеств. Если такое пересечение пусто, то наиболее вероятными считаются кратные неисправности, входящие в непустые пересечения наибольшего числа множеств.

Процедура пересечения множеств может производиться для неисправностей из 2^K последовательно в порядке убывания их вероятностей.

Несмотря на указанные ограничения время выполнения алгоритма для операций умножения, деления и других с $K \approx 30 + 60$, может оказаться весьма большим. Это имеет место, когда неисправности проявляются на маловероятных сочетаниях из 2^K возможных неисправностей.

Заметим, что оценка времени работы алгоритма дает требования на число "контрольных выходов" из схемы и частоту использования этих выходов. Такие выходы являются дополнительными и служат для определения места случайных сбоев и тех отказов, которые проявляются только на определенных исходных данных ЭМО (любое их изменение приводит к исчезновению ошибки). В этих случаях в отличие от явных отказов невозможно получение дополнительной информации о неисправности. Контрольные выходы увеличивают количество информации при первоначальном её проявлении, так как по существу разделяют сложную ЭМО на более простые, что обеспечивает снижение времени составления предположений. Это позволяет получить приемлемое время работы алгоритма и локализацию места тех неисправностей, для которых невозможно получение дополнительной информации без контрольных выходов, а также ускоряет процесс поиска неисправностей при отказах.

Поиск места неисправности при явных отказах производится с помощью алгоритма формирования диагностической процедуры.

Этот алгоритм представляет собой целенаправленный последовательный подбор тестов для локализации постоянных отказов. Отказы должны определяться с точностью до логического элемента, за исключением тех случаев, когда отказы элементов принципиально не различимы. Такой поиск может быть выполнен двумя способами.

Первый способ основан на анализе структуры схемы и списка (таблицы) предположений, сформированного алгоритмом общего диагностического анализа. Алгоритм производит изменение исходных данных, на которых зафиксирован \bar{R} , так, чтобы сократить список предположений. Если после выполнения ЭМО с этими измененными исходными данными не удастся локализовать неисправность, то опять производится изменение исходных данных для дальнейшего сокращения списка предположений. Другими словами, в алгоритме производится выбор тестов, позволяющих локализо -

вать место неисправности в зависимости от конкретной ситуации.

Второй способ основан на поиске неисправного элемента по заранее составленному диагностическому словарию для элементарной машинной операции R . Очевидно, что такие словари должны быть заранее составлены какими-либо методами [3],[4],[5].

Достоинством первого метода формирования диагностической процедуры является возможность поиска неисправностей в режимах работы схем, близких к тем, в которых проявилась неисправность. При выборе каждого из последующих тестов метод основывается на изменении исходного режима (исходных данных), тогда, как во втором методе режимы никак не связаны с исходным режимом, имевшем место при получении первоначальной информации. Здесь "близость" режима оценивается в вероятностном смысле.

Охарактеризованные содержательно алгоритмы, являются составными частями общего алгоритма квалификации и исправления неисправностей.

Исходными данными для работы алгоритма квалификации и исправления неисправностей являются:

1) рабочая программа

2) неправильно исполнившаяся машинная операция Θ и в ней неправильно исполнившаяся элементарная машинная операция R . Факт неправильного исполнения операции в момент T устанавливается пооперационным контролем, управление при этом передается на программу квалификации и исправления, а исполнение рабочей программы прерывается.

3) состояние памяти в момент T (тем самым известны операнды рабочих операций, в том числе и операций Θ и R),

4) таблицы предположений S_{T-i} , образующие к моменту T путем регулярного выполнения алгоритма классификации и исправления неисправностей до момента T . В момент T имеется всего 1 таблиц предположений. Первоначально при $T=0$ S_{T-i} - пусты, $1 \leq i \leq l$.

Ниже приведен алгоритм квалификации и исправления неисправностей:

1. Опросить все выходные регистра всех ЭМО операции Θ и записать в память их состояния (в том числе и выходного регистра ЭМО R , на котором зафиксирован неправильный результат \tilde{R}).

2. Повторить машинную операцию с теми же исходными данными, которые имели место при \tilde{R} .

В общем случае для такого повторения необходимо выполнить часть программы, предшествующую Θ . В дальнейшем будем обозначать такую часть программы через Θ^* .

3. Если результат на выходном регистре R ЭМО получается правильным (\bar{R}), перейти к п.4; если результат на выходном регистре R ЭМО вновь получается неправильным (\bar{R}), перейти к п.18.

4. Квалифицировать зафиксированную неисправность как сбойную.

5. Записать в память правильный результат \bar{R} .

6. Произвести поразрядное сравнение \bar{R} и \tilde{R} и записать в память номера неисправившихся разрядов регистра $R - p_e \{p_1, p_2, p_3, p_4, \dots, p_2\}$. Полученное множество p_e служит исходным материалом для проведения общего диагностического анализа.

7. Выполнить общий диагностический анализ. После составления таблицы предположений S_T^Θ управление передается к п.8.

8. Образовать все возможные пересечения S_T^Θ с хранящимися ранее таблицами предположений S_{T-1}, \dots, S_{T-1} :

$$S_{T-1}^1 = S_T^\Theta \cap S_{T-1}; \quad 1 \leq i \leq l.$$

9. Если все S_{T-1}^1 пусты, то перейти к п.12, в противном случае - к п.10.

10. Образовать все возможные сочетания непустых пересечений S_{T-1}^1 .

Образовать пересечения S_{T-1}^1 для каждого из сочетаний.

Все непустые пересечения, найденные в результате этой операции, объединить в множество S_T^* .

11. Анализируемая неисправность квалифицируется как проявление перемежающегося отказа. Если среди элементов S_T^* есть такой, что возможна замена элемента (достигнута требуемая точность), то производится переход к п.16, в противном случае переход к п.12.

12. Квалифицировать неисправность с таблицей предположений S_T^Θ как независимую сбойную.

13. Поместить S_T^Θ в хранилище ранее обнаруженных таблиц предположений присвоив ей номер S_{T-1} . У всех остальных

1 таблиц предположений уменьшить нижний индекс на 1.

14. Проверить $1 + 1 > 1_{\text{доп}}$ (здесь $1_{\text{доп}}$ - допустимое число учитываемых таблиц предположений).

Если условие не выполнено, то переход к п.17, в противном случае переход к п.15.

15. Исключить из хранилища таблиц предположений с наименьшим нижним индексом. Исключенную таблицу можно выдать на внешние носители для последующего ручного или автоматического статистического анализа. Перейти к п.17.

16. Выдать информацию для замены элемента s_T^{θ} и все s_{T-1}^{θ} , участвовавшие в образовании элемента множества s_T^{θ} , достаточного для замены исключаются из дальнейшего рассмотрения (стираются). Производится нумерация оставшихся таблиц. При этом возможна выдача информации на внешние носители для какого-либо дальнейшего статистического анализа.

17. Перейти к продолжению рабочей программы. Выполняется операция расположенная вслед за θ^* .

18. Квалифицировать обнаруженную неисправность как отказ.

19. Выполнить алгоритм формирования диагностической процедуры.

20. Если в результате выполнения алгоритма формирования диагностической процедуры зафиксированы неисправные элементы, перейти к п.21, в противном случае перейти к п.23.

21. Останов. Обслуживающий персонал должен заменить неисправные элементы. После пуска переход к п.22.

22. Передать управление в то место рабочей программы, где начинается θ^* .

23. Выдать информацию о неразличимых элементах. Останов. После поиска и устранения неисправного элемента переход к пункту 22.

Приведенный алгоритм классификации и устранения неисправностей является лишь грубым приближением к реально используемым и служит для пояснения существа метода.

Как видно из изложенного, предлагаемый метод основан на использовании всей информации о неисправностях, получаемой в процессе контроля.

Для реализации этого метода используется значительная емкость запоминающих устройств. Однако можно изменить алгоритм

квалификации и исправления неисправностей так, что без существенного снижения эффективности метода можно использовать только внешние носители (перфокарты, перфоленты). При этом информация о случайных сбоях анализируется не сразу после проявления сбоя, а периодически. После проявления сбоя необходимая информация выдается на перфоносители и продолжается решение. Если частота сбоев превысила допустимые пределы или накопились много информации за большой промежуток времени, то производится решение задачи анализа сбоев. При этом с перфоносителей вводятся данные о сбоях, описания схем ЭМО, на которых сбоем проявились, таблицы предположений о прежних проявлениях сбоев, программы диагностического анализа. Результатом являются данные о проявлении перемежающихся отказов (в идеальном случае информация об элементах - источниках этих отказов) и таблицы предположений о независимых сбоях и перемежающихся отказах, для которых точность определения неисправных элементов мала. При наличии постоянного отказа выполнение рабочей программы прерывается, очевидно сразу. Вводятся с перфоносителей описание ЭМО В, программы диагностического анализа и формирования диагностической процедуры.

Следует ожидать, что изложенный метод позволит устранять стареющие элементы на этапе проявления этого старения в виде перемежающихся отказов. Тогда постоянные отказы, которые придется устранять могут иметь только внезапный характер. Отсюда следует, что предлагаемая методика существенно повышает эффективность работы машины.

Л И Т Е Р А Т У Р А

1. Г.А. Миронов. Основные проблемы программного контроля дискретных автоматических систем.
Настоящий сборник.
2. Н.А. Криницкий, Г.А. Миронов, Г.Д. Фролов.
Использование понятия элементарная машинная операция для анализа вычислительных систем.
Настоящий сборник.
3. Г.А. Миронов, Д.Э. Федотова. Методика построения полных контролирующих и диагностических тестов. Настоящий сборник.
4. З.А. Вадова. Полный тест АУ для машины БЭСМ-6. Настоящий сборник.
5. А.В. Горбунов. Некоторые вопросы автоматического построения систем диагностических тестов для ЭВМ. Настоящий сборник.