

ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ КАРТЫ НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ С ЗАДАНЫМИ ПАРАМЕТРАМИ

Наркевич И.Г., Деменковец Д.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Бахтизин В.В. – кандидат технических наук

В работе рассматривается программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Рассматриваются существующие модели неисправностей запоминающих устройств. Предложен пример использования разработанного программного средства.

Вычислительная система может состоять из большого множества различных элементов, большинство которых не являются обязательными. Но одним из важнейших элементов, без которого вычислительная система не может функционировать, является запоминающее устройство. Это устройство служит для записи и хранения данных.

Причиной неисправного состояния запоминающего устройства является присутствие какого-то физического или механического дефекта либо их множества, при этом их количество и многообразие практически неограниченно. В зависимости от технологических особенностей при производстве памяти и внешних факторов при её эксплуатации могут появляться новые типы и разновидности дефектов. Установление факта появления дефекта и его классификация является весьма трудоёмкой и часто неразрешимой задачей. Это объясняется в первую очередь тем, что появление дефекта чаще всего можно определить лишь по косвенным признакам, как правило, по факту неправильной работы запоминающего устройства.

Современный этап развития методов и средств тестирования ОЗУ сопряжён со следующими основными проблемами. Во-первых, это неуклонный рост ёмкости ОЗУ, которая превышает 10^9 бит. Во-вторых, существенное отличие физической структуры ОЗУ от их логической организации, когда физически соседние запоминающие элементы имеют логически удалённые адреса. И, в-третьих, многие разновидности физических дефектов и отказов современных ОЗУ могут быть описаны только с помощью сложных моделей неисправностей ОЗУ, которые трудно обнаруживаются классическими тестами и их модификациями.

Для аналитического описания неисправных состояний памяти используются математические модели неисправностей, которые тем или иным образом отражают реальные физические дефекты запоминающих устройств. Основные физические дефекты этих устройств найдены и построены математические модели неисправностей, которые эти дефекты отражают. На основе имеющихся математических моделей можно построить компьютерные модели, которые будут иметь поведение, похожее на поведение реальных физических запоминающих устройств.

Первые модели неисправностей запоминающих устройств были предложены в начале 1980-х годов, которые включали константные неисправности, неисправности дешифратора адреса, неисправности взаимного влияния и кодочувствительные неисправности. Функциональные неисправности запоминающих устройств подразделяются на два подмножества: неисправности матрицы запоминающих элементов и неисправности электронного обрамления. Второе подмножество включает неисправности дешифраторов адреса и неисправности логики чтения-записи. Доминирующее значение имеют неисправности матрицы запоминающих элементов памяти, которые часто называют ячейками памяти.

К неисправностям матрицы ячеек запоминающих устройств в первую очередь относят неисправности, в которых участвуют: одна ячейка; две ячейки; несколько ячеек, в общем случае более чем две, без ограничений на их количество.

К неисправностям, затрагивающим одну ячейку запоминающего устройства, относят:

1. Константные неисправности (stuck-at faults – SAF). Неисправный запоминающий элемент памяти постоянно находится в состоянии логического нуля (s-a-0) или логической единицы (s-a-1), независимо от операций, выполняемых с неисправным элементом и другими элементами запоминающего устройства.

2. Переходные неисправности (transition faults - TF). Подобные неисправности характеризуются невозможностью перехода состояния неисправного запоминающего элемента из 0 в 1 (TF↑) или из 1 в 0 (TF↓) при выполнении соответствующих операций записи.

Среди неисправностей, в которых участвуют две ячейки запоминающего устройства, выделяют следующие неисправности:

1. Неисправности взаимного влияния (coupling fault - CF). При описании данной неисправности выделяют влияющую ячейку, определяемую её адресом i , изменение логического состояния которой воздействует на состояние зависимой ячейки с адресом j . Различают три типа неисправностей взаимного влияния:

а) инверсные неисправности взаимного влияния (inverse coupling faults - CF_{in}). При наличии данной неисправности изменение значения b_i влияющей ячейки вызывает инвертирование значения b_j зависимой ячейки. Возможны следующие виды CF_{in}: $\wedge(\uparrow, \bar{b}_j)$, $\wedge(\downarrow, \bar{b}_j)$, $\vee(\uparrow, \bar{b}_j)$, $\vee(\downarrow, \bar{b}_j)$. Символ \wedge и \vee задают взаимное расположение влияющего и зависимого запоминающих элементов памяти. Первый символ \wedge означает, что запоминающий элемент с меньшим адресом влияет на запоминающий элемент с большим адресом ($i < j$), а символ \vee используется в случае, когда адрес влияющего запоминающего элемента больше адреса зависимого запоминающего элемента ($i > j$);

б) неисправности взаимного влияния прямого действия (idempotent coupling faults - CF_{id}). При изменении значения b_i влияющего запоминающего элемента происходит принудительная установка определённого логического значения 0 или 1 в зависимом запоминающем элементе. Различают восемь неисправностей прямого действия: $\wedge(\uparrow, 0)$, $\wedge(\uparrow, 1)$, $\wedge(\downarrow, 0)$, $\wedge(\downarrow, 1)$, $\vee(\uparrow, 0)$, $\vee(\uparrow, 1)$, $\vee(\downarrow, 0)$, $\vee(\downarrow, 1)$;

в) статические неисправности взаимного влияния (state coupling faults CF_{st}). Переход зависимой ячейки в какое-либо состояние b_j возможен при определённом значении b_i влияющей ячейки. Возможно восемь неисправностей CF_{st}: $\wedge(0, 0)$, $\wedge(0, 1)$, $\wedge(1, 0)$, $\wedge(1, 1)$, $\vee(0, 0)$, $\vee(0, 1)$, $\vee(1, 0)$, $\vee(1, 1)$.

2. Кодочувствительные неисправности (pattern sensitive faults - PSF) рассматриваются как обобщение моделей неисправностей взаимного влияния. Для подобных неисправностей логическое состояние или изменение логического состояния одного запоминающего элемента запоминающего устройства может зависеть от содержимого (0 или 1) или от логических переходов из 1 в 0 или из 0 в 1 влияющих запоминающих элементов запоминающих устройств. В случае кодочувствительной неисправности PSF_k, в которой участвует k запоминающих элементов запоминающего устройства, подразумевается, что влияющими запоминающими элементами в предельном случае могут быть любые из $k - 1$ из N запоминающих элементов запоминающего устройства, а зависимым один из оставшихся $N - k + 1$ запоминающих элементов. На практике при рассмотрении кодочувствительных неисправностей вводятся ограничения на количество запоминающих элементов k , так и на их местоположение.

В зависимости от эффекта влияния на базовый запоминающий элемент различают несколько классических типов кодочувствительных неисправностей NPSF_k:

1. Пассивными кодочувствительными неисправностями (passive NPSF – PNPSF) являются неисправности, при которых состояние базового запоминающего элемента не может быть изменено для определённого кода в $k - 1$ соседних запоминающих элементах запоминающего устройства.

2. Под активными кодочувствительными неисправностями (active NPSF – ANPSF) понимают неисправности, в которых базовый запоминающий элемент изменяет своё состояние из-за изменения кода в соседних запоминающих элементах. Изменение кода для подобных неисправностей происходит в результате изменения состояния на противоположное только в одном соседнем элементе, в то время как остальные ячейки сохраняют предыдущее состояние.

3. Статические кодочувствительные неисправности (static NPSF – SNPSF) характеризуются тем, что для определённой комбинации значений в соседних запоминающих элементах базового запоминающего принудительно устанавливается в состояние 0 или состояние 1. Главным отличием статических неисправностей от активных кодочувствительных неисправностей является длительность процесса установления неверного значения в базовой ячейке. Для статических неисправностей это время существенно больше.

В структуре классических моделей неисправностей запоминающих устройств одно из самых видных мест в силу сложности их обнаружения тестами памяти занимают связные неисправности. Под связными неисправностями понимают неисправности различных типов, которые включают в себя общие ячейки.

В случае двух неисправностей взаимного влияния, например $(\uparrow, 1)$ и $(\uparrow, 0)$, с адресами ячеек i, j, l и g , участвующих в этих неисправностях и расположенных по возрастанию $i < j < l < g$, при $j = g$ две неисправности взаимного влияния, имеющую общую ячейку-жертву, трансформируются в связную неисправность. Возможны любые сочетания типов и подтипов неисправностей, участвующих в конкретной связной неисправности. Их многообразие и количество всегда являлись препятствием для анализа их обнаружения конкретным тестом.

В результате было создано программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Данное приложение позволяет создавать карту неисправностей, изменяя состояние конкретной выбранной ячейки памяти или распределяя случайным образом определённое количество неисправностей с заданными параметрами, сохранять созданную карту в файл и загружать сохранённую карту из файла.

Список использованных источников:

1. Ярмолик, В. Н. Контроль и диагностика вычислительных систем : учеб. пособие / В. Н. Ярмолик. – Минск : Изд-во : Бестпринт, 2019. – 387 с.
2. Ярмолик, С. В. Маршевые тесты для самотестирования ОЗУ : монография / С. В. Ярмолик, А. П. Занкович, А. А. Иванюк. – Минск : Изд. центр БГУ, 2009. – 271 с.