

1. Что такое система диагностирования?

Это – программные и аппаратные, встроенные и внешние средства диагностирования, обеспечивающие эффективное определение технического состояния (или диагноза).

2. С чего начинать разработку системы диагностирования?

С определения соответствия системы назначению и условиям работы (эксплуатации) объекта диагностирования. С составления Технического задания к Системе диагностирования.

1. Структура системы диагностирования:

Вы выбрали четырехэтапный подход: физический, аппаратный, программный и прикладной. Это логичная структура, потому что она позволяет проверить устройство на всех уровнях:

Физический этап проверяет базовые соединения.

Аппаратный этап тестирует отдельные компоненты.

Программный этап проверяет корректность работы ПО.

Прикладной этап тестирует устройство в реальных условиях.

2. Выбор методов тестирования:

Для каждого компонента вы выбрали специфические методы:

Для датчиков (DHT11, KY-037) вы сравниваете их показания с эталонными значениями.

Для дисплея OLED вы используете различные паттерны отображения.

Для памяти вы применили алгоритм MATS+, который эффективно выявляет ошибки в памяти.

3. Программные решения:

Ваш код включает функции для тестирования дисплея и кнопок. Вы используете различные паттерны отображения и проверяете отклик кнопок, что позволяет всесторонне протестировать эти компоненты.

4. Аппаратные решения:

Выбор компонентов (Arduino NANO, DHT11, KY-037, OLED дисплей) обусловлен их доступностью, надежностью и соответствием требованиям проекта.

5. Анализ результатов:

Вы проводите тесты и анализируете выходные данные, сравнивая их с ожидаемыми результатами. Это позволяет выявить любые отклонения от нормы.

6. Ограничения и возможные улучшения:

Ваша система имеет ограничения, например, по диапазону измерений датчика DHT11 (0-50°C). Возможные улучшения могли бы включать расширение диапазона измерений или добавление новых датчиков.

MATS и MATS+ – это алгоритмы тестирования памяти, которые относятся к классу маршевых тестов (March tests). Эти алгоритмы используются для выявления различных типов неисправностей в памяти, таких как stuck-at faults (залипание битов), transition faults (ошибки перехода), coupling faults (ошибки связи) и др. Давайте рассмотрим их подробнее:

1. MATS (Modified Algorithmic Test Sequence) требует $3n$:

MATS – это базовый алгоритм, который состоит из трех фаз:

Запись 0 во все ячейки памяти.

Для каждой ячейки: чтение 0, запись 1.

Для каждой ячейки: чтение 1.

Обозначение: $\uparrow(w0)$; $\uparrow(r0, w1)$; $\uparrow(r1)$

Где:

\uparrow означает, что направление обхода памяти не важно

\uparrow означает обход памяти от младших адресов к старшим

w0 – запись 0, w1 – запись 1

r0 – чтение 0, r1 – чтение 1

2. MATS+ (Modified Algorithmic Test Sequence Plus):

MATS+ – это улучшенная версия MATS, которая добавляет дополнительную фазу для более тщательного тестирования. MATS+ состоит из четырех фаз:

Запись 0 во все ячейки памяти.

Для каждой ячейки: чтение 0, запись 1.

Для каждой ячейки: чтение 1, запись 0.

Для каждой ячейки: чтение 0.

Обозначение: $\uparrow(w0)$; $\uparrow(r0, w1)$; $\downarrow(r1, w0)$; $\uparrow(r0)$

Где \downarrow означает обход памяти от старших адресов к младшим.

Преимущества MATS+:

1. Более эффективен в обнаружении некоторых типов ошибок по сравнению с MATS.

3. Относительно быстрый, требует всего $5n$ операций, где n – количество ячеек памяти.

В целом, MATS+ представляет собой улучшенную версию MATS, обеспечивающую более надежное тестирование памяти за счет небольшого увеличения сложности и времени выполнения.