

сообщения
объединенного
института
ядерных
исследований
дубна

135/
83

3/1-83

11-82-645

В.А.Вагов, Д.Рубин, В.Х.Хоромская

ЛОГИЧЕСКИЕ АНАЛИЗАТОРЫ.

Селективное накопление данных.

Обзор

1982

Логический анализатор является эффективным прибором для нахождения неисправностей в сложных цифровых системах. При этом особое значение имеет способность логического анализатора производить выборочное накопление информации о работе исследуемых электронных цепей.

I. Система запуска и селективное накопление данных

Запуск*) (triggering) имеет важную, можно сказать, ключевую роль в работе логического анализатора. Система запуска совместно с возможностями селективного накопления данных об исследуемой схеме, позволяет производить отбор необходимой информации. Это облегчает поиск ошибок в цифровых системах и уменьшает объем требуемой памяти логического анализатора.

I. I. Запуск

Самый распространенный метод запуска – запуск параллельной комбинацией сигналов. В этом случае запуск производится тогда, когда в процессе записи сигналов из исследуемой схемы, параллельная комбинация этих сигналов совпадает с заранее установленной "запускающей" комбинацией.

В большинстве логических анализаторов запускающая комбинация устанавливается с помощью ряда тумблеров на передней панели прибора. Каждому входному каналу соответствует свой тумблер, имеющий три положения – 0, 1, x. Положение тумблера 1 означает, что для данного входа запускающая комбинация содержит логическую "1", а 0 – соответственно логический "0". Положение тумблера x показывает, что данный канал в запуске не участвует (его состояние не влияет на запуск).

*) Здесь и далее под "запуском" подразумевается не начало работы анализатора, а момент появления на входе анализатора определенной комбинации входных сигналов (в процессе записи сигналов из проверяемой схемы), под действием которой анализатор переключается из одного режима работы в другой.

Запуск логического анализатора можно осуществить как по появлению заданной комбинации (true-triggering), так и по исчезанию определенной комбинации (false-triggering).

Данные, характеризующие состояние исследуемой системы, могут располагаться относительно момента запуска в 4-х различных вариантах: запускающая комбинация может находиться в начале, в середине и в конце блока данных, а, кроме того, блок данных может записываться с задержкой относительно комбинации запуска (рис.1).

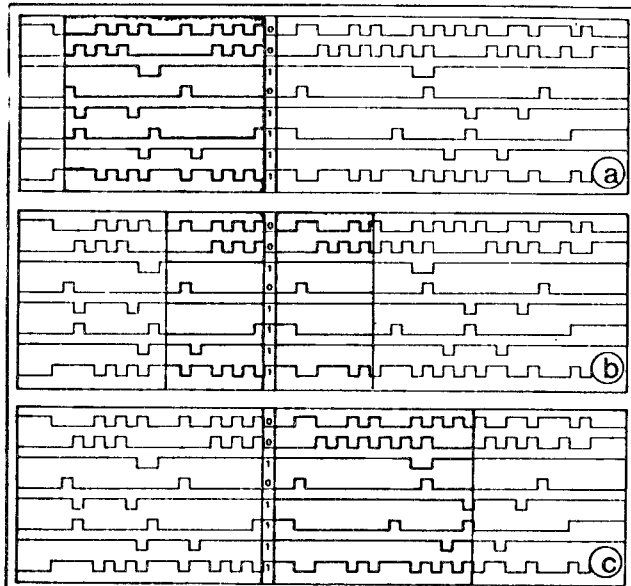


Рис.1. Различные варианты расположения триггерной комбинации (0010 1111) относительно записанного блока данных.

Имеются две возможности задержки начала записи данных относительно запуска: 1) счет числа строб-импульсов от появления заданной комбинации до начала записи (clock-delay) и 2) запись данных после определенного числа появлений заданной комбинации (pass countered triggering). Последний способ очень удобен при исследовании многократно повторяющихся циклов. Предположим, что нас интересует только определенный проход через цикл. В этом случае при исследовании системы заданная комбинация появляется несколько раз (т.е. в каждом цикле), а мы можем выбрать интересующий нас проход через этот цикл.

Более сложные анализаторы имеют "запуск последовательностью слов". Здесь запускающей комбинацией является определенная, заранее установленная последовательность входных комбинаций. Эта методика может быть полезной при исследовании различных разветвлений программ. Рассмотрим, например, случай, когда программа начинается со слова А и заканчивается словом D. Между ними возможны три варианта прохода: ABD, ACD, ABCD (рис.2). Эти разветвления можно различить с помощью последовательностей

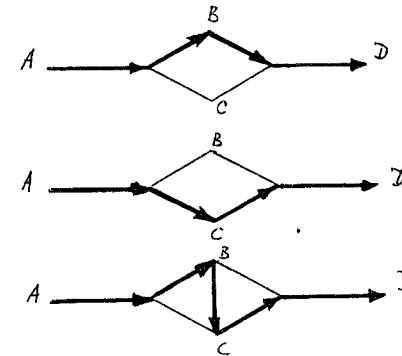


Рис.2. Разделение программных разветвлений с помощью "запуска последовательностью слов".

BD, AC и BC, предшествующих запускающей комбинации D. Таким образом, можно выделить и проследить проход по любому из трех разветвлений программы.

Обычно трудно исследовать причины возникновения временных ошибок, т.к. они наблюдаются редко и в случайные моменты времени. При решении этой проблемы удобно использовать режим "запуск по несравнению". В этом режиме вначале накапливается блок данных, в котором, по предположению, иногда возникает ошибка, например до триггерной комбинации хуз . Затем, убедившись, что этот блок данных содержит правильную информацию, мы перепишем его в эталонную память. После этого включается режим "несравнения" и опять начинается регистрация данных. При этом анализатор ждет вновь появления комбинации хуз и при ее появлении сравнивает вновь записанные данные с эталоном. Если два блока данных полностью совпадают, то вновь начинается накопление и т.д. Если же вновь записанные данные отличаются от эталонных, то накопление данных прекращается и различающиеся сигналы отображаются на экране дисплея для изучения.

Узкие импульсы помех, возникающие в цифровых системах, мешают не только работе самой системы, но и запуску логического анализатора, так как быстродействующий блок распознавания слов может принять их за полезный сигнал и организовать нелегальный запуск. Например, при отладке шин, имеющих драйверы на три состояния, при включении и выключении драйверов генерируются узкие помехи большой интенсивности. В этом случае можно использовать запуск, с помощью которого отбрасываются те запускающие сигналы, ширина которых меньше заранее установленного интервала времени. Этот интервал времени обычно регулируется в пределах от 15 до 300 нс /1, 2/.

И наоборот, иногда импульс помехи служит специально для запуска прибора. В этом случае импульс помехи используется в качестве одного из битов запускающей комбинации. Такой режим позволяет наблюдать причины возникновения помехи.

1.2. Селективное накопление данных

Имеются следующие наиболее распространенные формы селективного накопления данных.

1. Накапливаются только те данные, которые совпадают с запускающей комбинацией. Этот метод используется, например, в случае, когда нужно наблюдать только определенные типы команд и циклов в шине микропроцессора (например, требуется проследить все команды ввода-вывода) во время выполнения программы.

2. Другой тип селективного накопления - использование т.н. квалификаторов. Квалификатор считается самостоятельным каналом, который может быть включен в запускающую комбинацию в качестве дополнительного условия. Важнейшей областью применения этих импульсных квалификаторов является демultipлексация шин микропроцессора. Рассмотрим, например, микропроцессор INTEL 8085, восемь выходов которого при выполнении команд используются вначале как 8 шин адреса (половина 16-разрядного адреса), а потом как шины данных. Если не демultipлексировать сигналы этой шины, то мы не сможем отличить данные от адреса. На выходе микропроцессора во время адресной информации возникает сигнал "Address Latch Enable" (ALE), а во время появления данных - сигналы либо "READ" либо "WRITE". Если использовать эти сигналы в качестве периодических квалификаторов, то можно таким образом разделить сигналы данных и сигналы адреса.

2. Дисплей

В основные функции дисплея входит представление накопленных в памяти данных в наиболее удобной для анализа форме, а также обес-

печение дополнительной информации, облегчающей анализ данных. В анализаторах, построенных на базе микропроцессора, в которых регулировка и контроль прибора происходит с помощью клавиатуры в интерактивном режиме, на дисплее отображается информация, необходимая для управления, а также все характеристики установленных режимов /3/.

Курсор дисплея облегчает редактирование данных, с его помощью выполняются различные манипуляции и сравнение данных и блоков данных. Например, при работе с логической временной диаграммой курсор представляет собой вертикальную яркую линию, которая помогает определять состояние всех каналов в какой-то момент времени. При этом на дисплее появляется код данной логической комбинации, на которую указывает вертикальный курсор. В некоторых логических анализаторах можно отметить с помощью горизонтального курсора один ряд таблицы состояний, а затем нажатием кнопки вызвать изображение соответствующей временной диаграммы, начиная от состояния, отмеченного горизонтальным курсором. Таким образом, можно сравнивать шаги программы и оказываемое ими действие на электронику.

2.1. Формат данных

Данные представляются на экране дисплея в следующих форматах:

1) в виде логической временной диаграммы, 2) в виде таблицы состояний и 3) в графической форме.

1. Логическая временная диаграмма изображает сигналы каналов в виде реконструированной диаграммы в реальном масштабе времени (рис.3).

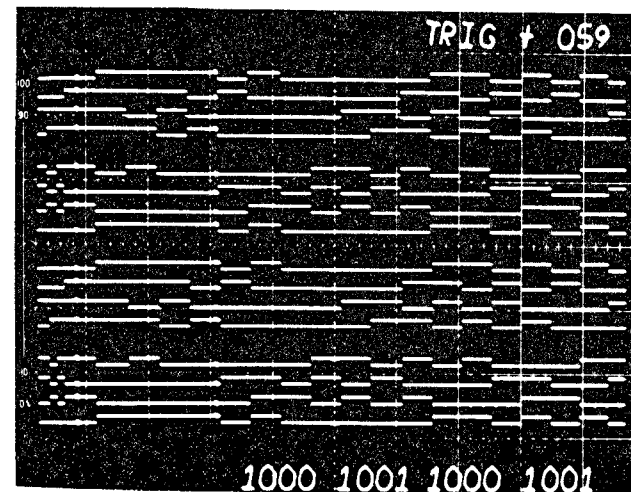


Рис.3. Формат изображения в виде временной диаграммы.

Такой формат изображения удобен при исследовании проблем электроники (временные ошибки, смещение фронтов и т.д.). Основное преимущество его использования в том, что можно одновременно представить довольно большую последовательность данных и в случае необходимости можно часть диаграммы "растянуть" во времени для более детального изучения. Обычно коэффициент умножения равен $\times 10$ или $\times 25$, и его значение также дается на экране дисплея. Некоторые модели имеют также возможности усиления изображения по вертикали.

2. Таблица состояний представляет данные, стробированные одновременно на нескольких параллельных каналах, в виде таблицы, использующей различные цифровые коды. Этот формат удобен при изучении проблем программного обеспечения.

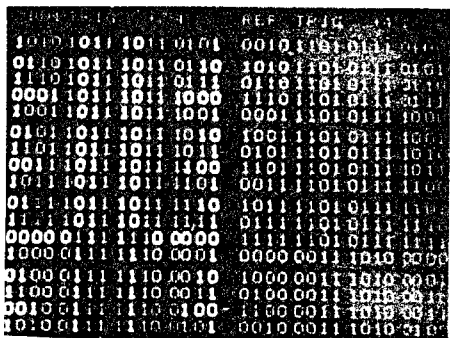


Рис. 4. Формат изображения в виде таблицы состояний.

Типы используемых кодов включают: двоичный, восьмеричный, шестнадцатеричный, ASCII и мнемокод. Самым распространенным является двоичный код ввиду его универсальности. Поэтому такой формат изображения имеется почти в каждом логическом анализаторе. Восьмеричный и шестнадцатеричный коды применяются в основном при работе с мини-ЭВМ и микропроцессорами⁴, а мнемокоды — при отладке программного обеспечения. В некоторых анализаторах³ можно разделить накопленные данные на группы и для каждой группы использовать наиболее удобный вид представления данных.

3. С помощью графических форматов типа "map" (карта) и "граф" можно без детального изучения быстро просмотреть работу системы. В обоих форматах стробированные данные изображаются в виде отдельных точек на дисплее.

В формате "map" вертикальная координата точки определяется старшими разрядами записанного состояния, а горизонтальная — младшими разрядами соответствующего данной точке состояния. Формат "map" может быть как статического, так и динамического типа. При динамическом формате "map" точки-состояния связаны между собой векторами, которые изображают последовательность движения с точки на точку. Яркость точки показывает относительную частоту появления данного состояния. В некоторых моделях⁵ имеется также возможность "сканирования" в динамическом режиме. Сканирование может производиться как вручную, так и автоматически, в быстром или медленном режиме. Во время сканирования метка "+" движется по изображенным точкам (в прямом и обратном направлениях) в такой же последовательности, в какой были записаны данные.

При этом на экране высвечивается код состояния, на которое указывает метка, в выбранном виде представления (двоичный, восьмеричный и т.д.).

В формате "граф" данные изображаются в виде точек, соответствующих последовательности накопления данных, где горизонтальная позиция определяется временем записи, а вертикальная координата — комбинацией логических состояний.

С помощью режимов "map" и "граф" можно исследовать, например, такие проблемы, как ошибочное попадание программы в запретную зону и т.д.

2.2. Сравнение данных и блоков данных

Как мы уже отмечали, с помощью данных, записанных в эталонную память, можно быстро и эффективно проверять вновь накопленные данные. С помощью экрана оператор может сравнить полученные данные с эталонными. Существуют следующие типы сравнения данных.

1. Режим "исключающее ИЛИ". В этом режиме содержимое эталонной памяти и содержимое памяти накопления одновременно появляются на экране дисплея в виде двух расположенных рядом таблиц состояний, в которых одновременные состояния находятся на одной горизонтали. Для более удобного наблюдения различий отличающиеся данные имеют большую яркость.

2. В режиме "held" на экране дисплея появляется содержимое выделенной области эталонной памяти. При этом вновь накопленные данные высвечиваются рядом тоже в виде таблицы, которую можно двигать до нужного положения. После этого можно сравнивать два блока данных в режиме "слежение со смещением" (т.е. обе таблицы двигаются вместе).

3. Режим "поиск". В этом режиме вновь накопленные данные автоматически перемещаются на экране до тех пор, пока для слова, отмеченного курсором в эталонной памяти, не найдется такое же слово в памяти накопления. После этого движение данных на экране прекращается и высветится число периодов стробирования между данным словом и запускающей комбинацией.

В логических анализаторах для изображения данных используются семисегментные светодиодные дисплеи и дисплеи на телевизионных трубках. Дисплеи на семисегментных светодиодах применяются в наиболее простых и дешевых анализаторах логических состояний, а также в анализаторах для микропроцессоров. Большинство ЛА, построенных на базе микропроцессора, используют телевизионную трубку.

3. Дополнительные устройства и характеристики

Мы уже рассмотрели основные функциональные блоки логических анализаторов. При этом, конечно, диапазон возможностей анализатора меняется в зависимости от его типа и стоимости. Но, кроме основных блоков, для этих приборов существует дополнительная аппаратура, которая во многих случаях значительно облегчает выполнение задачи, а также дает возможность решать проблемы очень высокого уровня.

Одна из таких дополнительных возможностей - аналоговый канал, все чаще используемый в логических анализаторах. Возможны два способа организации аналогового канала в ЛА. Первый способ - объединение осциллографа и анализатора в одном приборе^{6/}. Другой способ - создание цифрового осциллографа с АЦП^{7/}. Недостаток первого варианта в том, что с помощью этого прибора нельзя наблюдать одиночные сигналы.

Модель PI-540^{7/} содержит аналого-цифровой преобразователь с частотой 50 МГц. Эта модель дает возможность наблюдать одиночные сигналы. Ее аналоговый канал позволяет регистрировать сигналы с максимальной частотой 15 МГц и разрешающей способностью по амплитуде 6 бит. Допустимая память для АЦП составляет 1024 разряда. Аналоговый канал можно запускать с помощью цифровых каналов самого логического анализатора. Такой прибор идеален для наблюдения не очень быстро меняющихся аналоговых сигналов, например при проверке электромеханических устройств. Недостатком аналоговых каналов, использующих АЦП, является относительно медленная частота стробирования, не позволяющая наблюдать высокочастотные сигналы.

Следующим типом дополнительной аппаратуры являются стандартные интерфейсы. С помощью стандартных интерфейсов можно записывать данные из анализатора в устройства внешней памяти, например, на миниатюрные гибкие диски или на магнитофонную кассету для последующего

изучения. Этот же интерфейс обеспечивает возможность записи данных в эталонную память из других устройств, например из памяти ЭВМ. В этом случае, при подключении анализатора к ЭВМ, интерфейс кроме двусторонней передачи данных обеспечивает полный контроль прибора со стороны ЭВМ.

4. Применение логических анализаторов

Сфера использования цифровой измерительной техники делится на 3 основные области: разработка, серийное производство и эксплуатация. И для каждой сферы применения, измерительная техника должна иметь различные характеристики.

Создание логических систем включает разработку электронной схемы и программного обеспечения. При разработке цифровых систем инженеры и программисты работают вначале параллельно, а потом совместно.

Рассмотрим основные фазы разработки цифровых систем (рис.6) и типы приборов, требующихся на различных ее этапах^{8/}. Например, цифровой вольтметр и осциллограф удобно использовать на начальном этапе работы. С помощью этих приборов можно убедиться, что электронные и временные параметры цепей находятся в пределах допустимых границ.

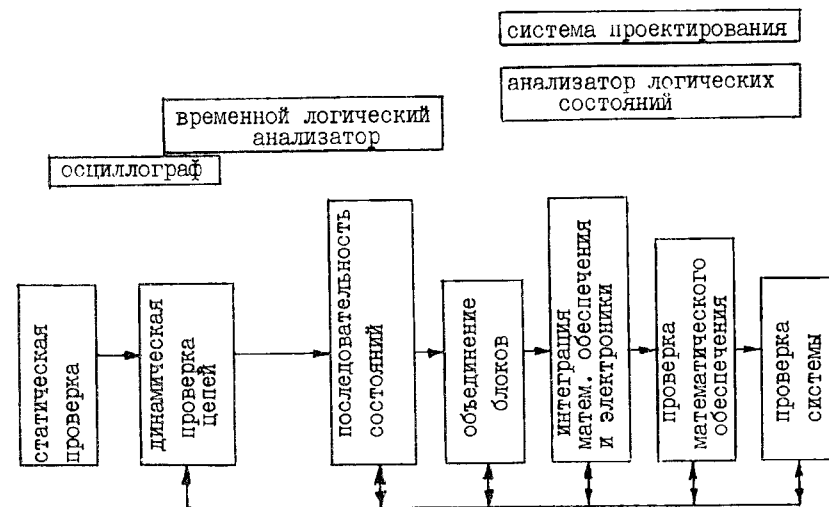


Рис.5. Этапы разработки цифровых систем.

Затем для проверки отдельных функций системы необходим прибор, позволяющий просмотреть в широком диапазоне работу системы в целом. Для этой цели оптимальным прибором является логический анализатор временных диаграмм.

После этого требуется объединить разработанные программы и проверить совместную работу электроники и программного обеспечения. До этой фазы разработка программного обеспечения происходила самостоятельно, с помощью т.н. "систем проектирования" ("logic development system"). Теперь, когда обе части работы совместились, необходимо проверить изменения состояний на линиях адресов, данных и управляющих линиях, происходящие под действием программных шагов. Эту задачу программисты и инженеры решают вместе, с помощью анализатора логических состояний. После этого начинается окончательная настройка системы, т.е. устранение ошибок, появляющихся в очень сложных взаимосвязях системы, и редких статистических ошибок. Здесь целесообразно использование ЛА, имеющих сложную систему запуска и селективного накопления данных.

В сфере производства для ускорения проверки работы серийных систем требуются анализаторы, работающие под управлением ЭВМ.

В сфере технического обслуживания основным является вопрос уменьшения времени нахождения ошибки, важны также малый вес и компактность прибора. Для уменьшения времени ремонта можно использовать такие возможности: объединение в одном приборе различных методов измерения; сравнение данных, получаемых от тестируемой системы, с образцами правильных данных. Для хранения эталонной информации можно применять кассеты и миниатюрные гибкие диски.

Можно ожидать, что развитие цифровых измерительных приборов в будущем будет продолжаться, но, возможно, не так быстро, как в предыдущие годы, а постепенно, вместе с развитием цифровой техники. Основные характеристики логических анализаторов уже установились, но можно выделить несколько направлений, по которым началось и продолжается развитие ЛА. Анализаторы будут строиться на основе микропроцессоров, будут иметь полную клавиатуру и интерактивный язык для программирования работы анализатора; все шире используются интерфейсы для диагностики с помощью ЭВМ и устройства памяти для хранения эталонных данных.

Литература

1. Martin Marshall, What to look for in logic timing analysers ,
Electronics, March 29, 1979, p.109.

2. Jeffrey H. Smith, A logic state analyser for microprocessor systems , Hewlett-Packard Journal, January 1977, p.2.
3. Logic state analyser for mini/macro and random logic analysis , Hewlett-Packard, Models 1610 A/B, June 1979.
4. Andy Santoni, Testers are getting better at microprocessor flaws , Electronics, December 23, 1976, p.57.
5. A display formatter - the indispensable tool for the data domain , Tekscope, vol. 8, №4, 1976.
6. PM 3540 logic and real-time analysis in a compact instrument , Phillips, Test & Measuring Instruments.
7. Jonah McLeod, Logic analysers - sharp fault finders getting sharper , Electronic Design, March 29, 1980, p.48.
8. Bruce Farly, When logic analysers meet development systems , Electronics, September 13, 1979, p.141.

Рукопись поступила в издательский отдел
1 сентября 1982 года.

НЕТ ЛИ ПРОБЕЛОВ В ВАШЕЙ БИБЛИОТЕКЕ?

Вы можете получить по почте перечисленные ниже книги, если они не были заказаны ранее.

Д13-11182	Труды IX Международного симпозиума по ядерной электронике. Варна, 1977.	5 р. 00 к.
Д17-11490	Труды Международного симпозиума по избранным проблемам статистической механики. Дубна, 1977.	6 р. 00 к.
Д6-11574	Сборник аннотаций XV совещания по ядерной спектроскопии и теории ядра. Дубна, 1978.	2 р. 50 к.
Д3-11787	Труды III Международной школы по нейтронной физике. Алушта, 1978.	3 р. 00 к.
Д13-11807	Труды III Международного совещания по пропорциональным и дрейфовым камерам. Дубна, 1978.	6 р. 00 к.
	Труды VI Всесоюзного совещания по ускорителям заряженных частиц. Дубна, 1978 /2 тома/	7 р. 40 к.
Д1,2-12036	Труды V Международного семинара по проблемам физики высоких энергий. Дубна, 1978	5 р. 00 к.
Д1,2-12450	Труды XII Международной школы молодых ученых по физике высоких энергий. Приморско, НРБ, 1978.	3 р. 00 к.
	Труды VII Всесоюзного совещания по ускорителям заряженных частиц, Дубна, 1980 /2 тома/	8 р. 00 к.
Д11-80-13	Труды рабочего совещания по системам и методам аналитических вычислений на ЭВМ и их применению в теоретической физике; Дубна, 1979	3 р. 50 к.
Д4-80-271	Труды Международной конференции по проблемам нескольких тел в ядерной физике. Дубна, 1979.	3 р. 00 к.
Д4-80-385	Труды Международной школы по структуре ядра. Алушта, 1980.	5 р. 00 к.
Д2-81-543	Труды VI Международного совещания по проблемам квантовой теории поля. Алушта, 1981	2 р. 50 к.
Д10,11-81-622	Труды Международного совещания по проблемам математического моделирования в ядерно-физических исследованиях. Дубна, 1980	2 р. 50 к.
Д1,2-81-728	Труды VI Международного семинара по проблемам физики высоких энергий. Дубна, 1981.	3 р. 60 к.
Д17-81-758	Труды II Международного симпозиума по избранным проблемам статистической механики. Дубна, 1981.	5 р. 40 к.
Д1,2-82-27	Труды Международного симпозиума по поляризационным явлениям в физике высоких энергий. Дубна, 1981.	3 р. 20 к.
Р18-82-117	Труды IV совещания по использованию новых ядерно-физических методов для решения научно-технических и народнохозяйственных задач. Дубна, 1981.	3 р. 80 к.

Заказы на упомянутые книги могут быть направлены по адресу:
101000 Москва, Главпочтамт, п/я 79
Издательский отдел Объединенного института ядерных исследований

Вагов В.А., Рубин Д., Хоромская В.Х.

11-82-645

Логические анализаторы. Селективное накопление данных /обзор/.

Цифровые измерительные приборы - логические анализаторы /ЛА/ используются при разработке и эксплуатации сложных цифровых систем. Подробно рассмотрены система запуска ЛА, возможности селективного накопления данных об исследуемой схеме и сравнения данных, а также форматы их представления.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1982

Vagov V.A., Rubin D., Khoromskaya V.Ch.

11-82-645

Logic Analyzers. Selective Data Gathering /Survey/.

A logic analyzer /LA/ is an effective instrument for development of digital devices and for computer maintenance problems. Data triggering and selective data gathering capabilities of LA are considered. Various logic analyzer displays are shown.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1982

Перевод авторов.