

# МОДУЛИ ДЛЯ ПЕРСОНАЛЬНОГО ТЕСТЕРА А2ХХ

## Введение

Обязательной частью любого тестера, предназначенного для контроля изделий электронной техники, является компьютер, основной, а может быть единственной, задачей которого в процессе контроля являлась обработка результатов контроля. С появлением настольных, но достаточно мощных компьютеров, ситуация изменилась, и компьютер из составной части превращается в платформу для построения измерительных устройств. Для этого необходимо сделать очевидный шаг – дополнить стандартные ресурсы компьютера измерительными ресурсами и в результате получить персональный тестер.

Это превращение обеспечивается открытой архитектурой персонального компьютера серии IBM PC, позволяющей теперь конфигурировать не только стандартные, но и измерительные ресурсы согласно требованиям конкретного пользователя.

Одним из вариантов реализации этого подхода по созданию контрольно-измерительного оборудования заключается в следующем - контрольно-измерительные (КИ) ресурсы распределяются между платой расширения, установленной в системном блоке и платой (платами), расположенной в выносном блоке.

Таким образом, персональный тестер, показанный на рис. 1, включает в свой состав модули двух типов – КИ адаптер *adp*, расположенный в системном блоке PC, и КИ модуль *mod*, расположенный в выносном блоке, и формирующий следующие сигналы, обеспечивающие разнообразные режимы параметрического и функционального контроля:

- Dr0...Dr23 выходы драйверов,
- Cm0...Cm63 входы компараторов,
- Uc0...Uc3 четырехканальный источник напряжения питания с функцией измерения тока потребления,
- Up, Ug два источника напряжения – измерителя тока.

К этому модулю подключается объект контроля. КИ адаптер выполняет функции аналого-цифрового и цифро-аналогового преобразования, а также обеспечивает управление КИ ресурсами и исполнительным устройством, например зондовой установкой *znd*.

Адаптер и модуль соединяются между собой двумя шинами – ISA и AB. Первая шина обеспечивает связь модуля с системными ресурсами компьютера, а вторая – обмен аналоговыми сигналами.

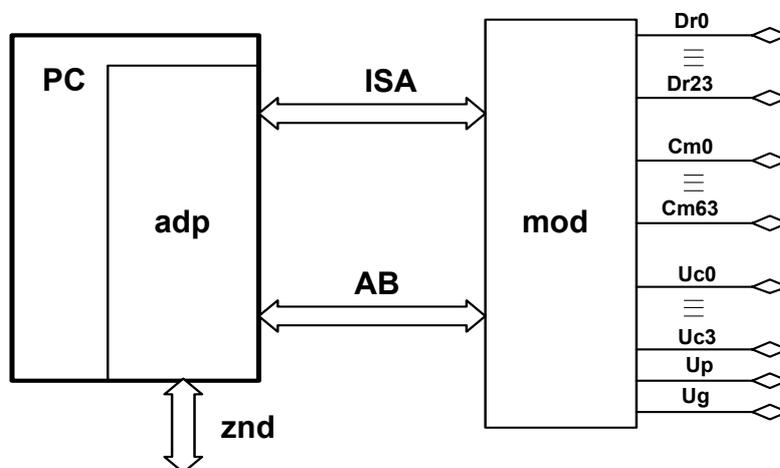


Рис. 1

## КИ адаптер

КИ адаптер, показанный на рис. 2, включает в свой состав следующие устройства:  
двенадцатиразрядный аналогово-цифровой преобразователь ADC –обеспечивает проведение измерений электрических величин. Кроме этого АЦП формирует высокостабильные напряжения  $U_{r+}$  положительной и  $U_{r-}$  отрицательной полярности;

тринадцатиразрядный цифро-аналоговый преобразователь DAC – вырабатывает постоянные напряжения  $U_0 \dots U_7$ , необходимые для формирования электропараметров и для проведения контрольно-измерительных операций;

восьмиканальный аналоговый коммутатор  $MuIn$  – позволяет выбрать и передать на АЦП измеряемую величину;

синхронизатор CHR, состоящий из формирователей длительности цикла обращения к шине ISA и импульса  $CL1$  – обеспечивают динамические характеристики контрольно-измерительных операций. Именно через шину ISA происходит обмен данными между системными ресурсами и КИ ресурсами, поэтому длительность цикла во время функционального контроля определяет его период, а во время параметрического контроля – временные задержки. Поскольку существующие операционные системы не позволяют использовать компьютер для работы в реальном режиме времени с требуемыми параметрами, были созданы эти формирователи. Формирователь длительности цикла обращения обеспечивает два режима работы шины ISA:

- нормальный цикл записи/чтения 16-ти разрядной памяти, длительность которого указана в спецификации на шину ISA,
- удлиненный цикл записи/чтения 16-ти разрядной памяти, длительность которого определяется следующим уравнением

$$T_{isa} = nT_{scl};$$

где:  $T_{isa}$  – длительность цикла обращения к шине ISA,

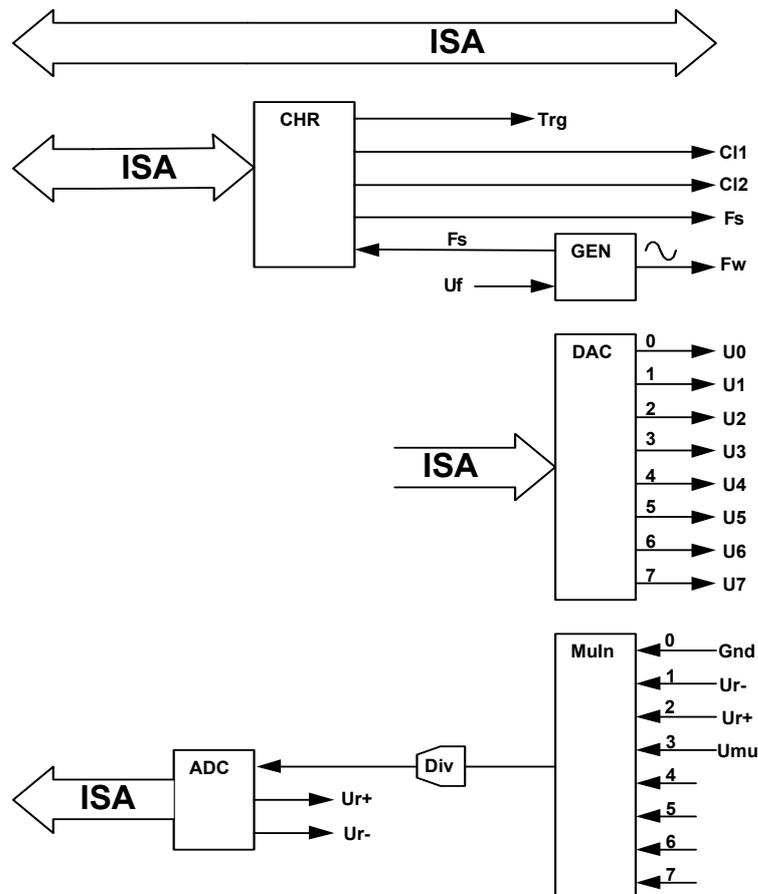


рис2

$T_{scl}$  – длительность периода следования импульсов SYS CLK,  
 $n = 1, 2, \dots$  целое число.

Второй формирователь обеспечивает формирование импульса синхронизации CL1 с программируемой задержкой начала и окончания относительно начала цикла обращения, которые определяются следующими уравнениями:

$$T_{be} = n_1 T_{scl}; \quad T_{en} = n_2 T_{scl};$$

где:  $T_{be}$  и  $T_{en}$  – задержка начала и окончания импульса соответственно,  
 $n_1$  и  $n_2$  – целые числа.

На рис. 3 показаны сигналы, поясняющие работу формирователей. По нарастающему фронту сигнала SMEMW, который вырабатывается шиной ISA, данные Drg записываются в выбранный дешифратор адреса регистр и хранятся в нем пока не будет произведена очередная запись в этот регистр. Импульс CL1 формируется в каждом цикле и может быть использован для синхронизации режимов работы объекта контроля. Параметры импульсов  $T_{isa}$ ,  $T_{be}$ ,  $T_{en}$  должны быть предварительно установлены.

Таким образом, можно получить длительность цикла  $T_{isa}$  в диапазоне от 2 до 15 мкс с дискретностью  $T_{scl}$  равной 120 нс.

Параметры импульса CL1 устанавливаются с той же дискретностью, при этом необходимо помнить, что импульс CL1 должен оставаться внутри цикла ISA.

Кроме этого синхронизатор вырабатывает импульс Trg, предназначенный для внешнего запуска развертки осциллографа.

Делитель напряжения Div.

Генератор GEN синусоидальных-треугольных-прямоугольных импульсов Fw. Частота импульсов определяется величиной напряжения  $U_f$ , в качестве которого может выступать один из следующих, предварительно выбранных с помощью джампера, выходов ЦАП: U0, U2, U6, U7. Кроме этого генератор формирует последовательность прямоугольных импульсов Fs, по частоте совпадающих с Fw.

Все устройства, входящие в состав адаптера, получают доступ к системным ресурсам компьютера через шину ISA.

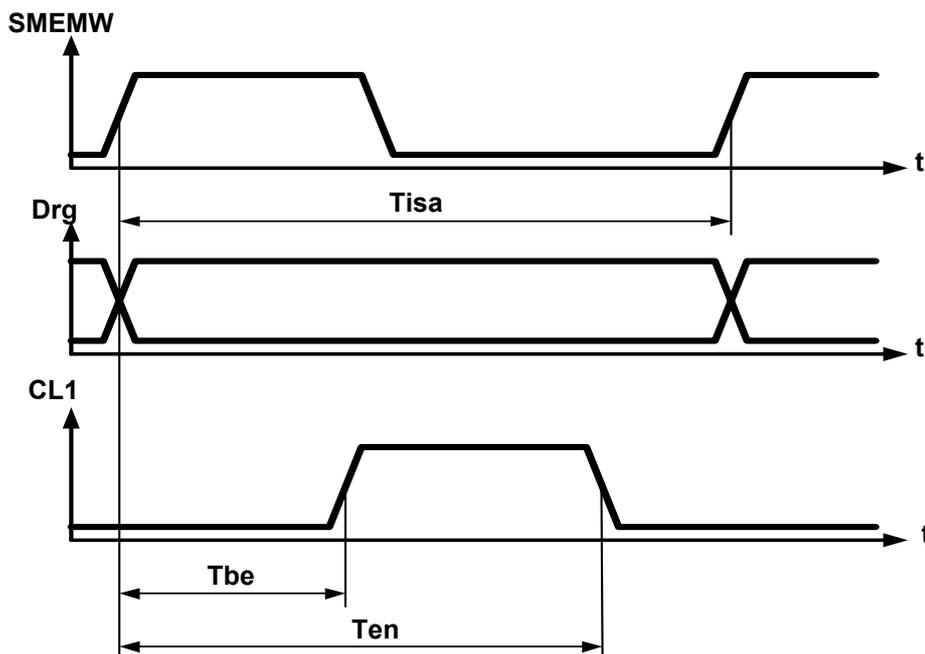
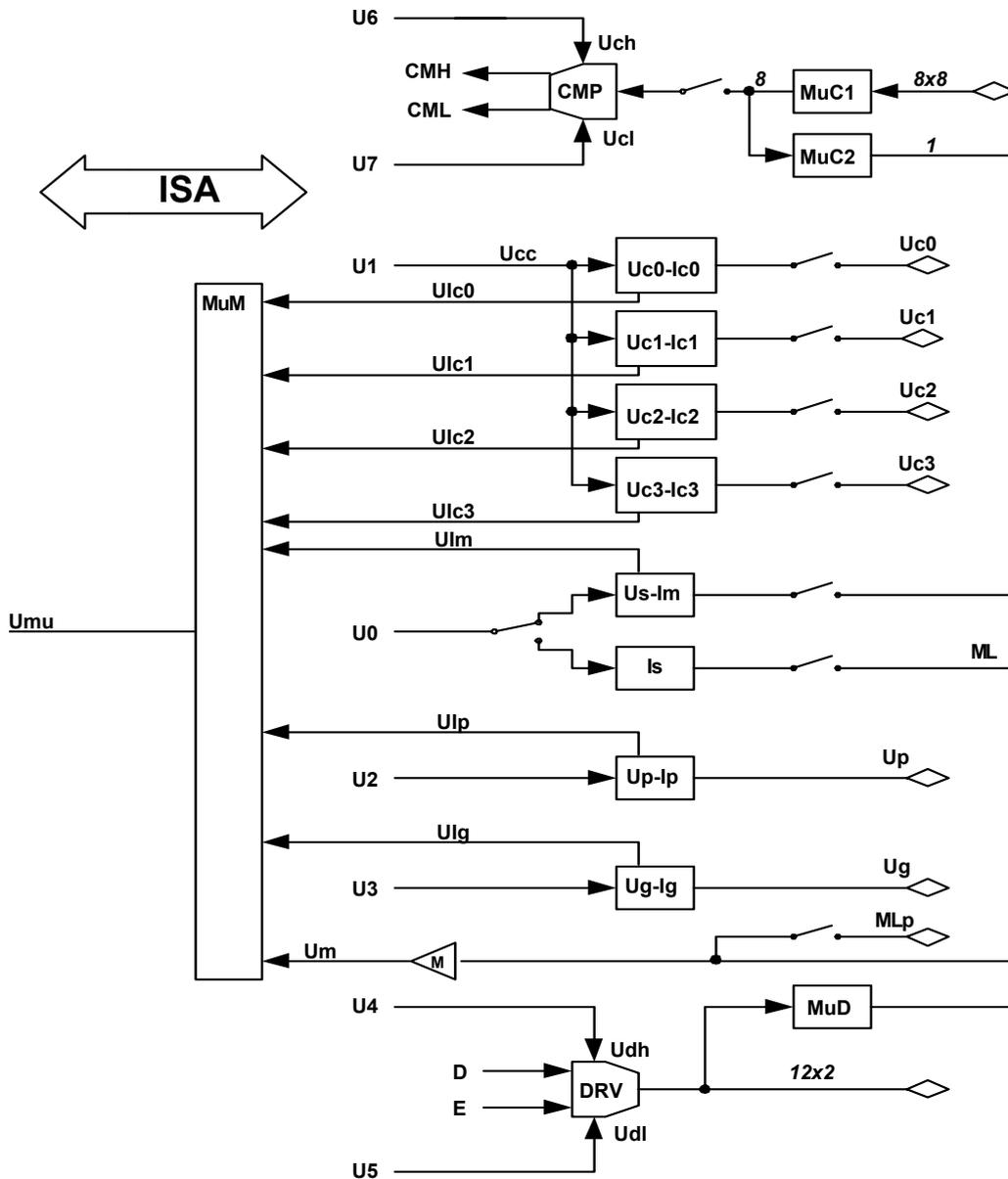


Рис3

## КИ модуль

КИ модуль (рис. 4) включает в свой состав ряд устройств.

Четырехканальный источник питания – измеритель тока потребления  $Uc1-Ic1 \dots Uc4-Ic4$  имеет четыре выхода, на которых формируется напряжение питания четырех объектов контроля (ОК), при этом величина напряжения на всех четырех каналах  $Uc1 \dots Uc4$  одинаковая и равна входному напряжению  $Ucc$ . Каждый из четырех каналов содержит преобразователь  $Uic1 \dots Uic4$



**Рис4**

тока потребления в напряжение, которое через коммутатор **MuM** поступает на адаптер, и далее на АЦП. Так происходит измерение тока потребления по выбранному с помощью коммутатора **MuM** каналу. Необходимо добавить, что существуют три диапазона

тока потребления, которые устанавливаются одновременно для всех каналов. Каждый канал может быть отключен от ОК.

Измерительная линия **ML** – это электрическая цепь, которая выделена для облегчения описания процесса контроля.

В общем виде этот алгоритм состоит в следующем: с одной стороны к измерительной линии подключается ОК, с другой стороны – измерительный ресурс, обеспечивающий один из двух режимов контроля, – это заданный ток или заданное напряжение. С третьей стороны к

измерительной линии подключается измерительный ресурс, обеспечивающий собственно измерение: в первом случае это измеритель напряжения, а во втором – измеритель тока.

Формирователь напряжения смещения – измеритель тока утечки

Us-Im имеет два диапазона измерения тока и обеспечивает режим параметрического контроля, при котором на выбранный вывод ОК подается заданное напряжение и измеряется ток.

Два источника напряжения – измерителя тока  $U_p-I_p$ ,  $U_g-I_g$  имеют повышенную мощность

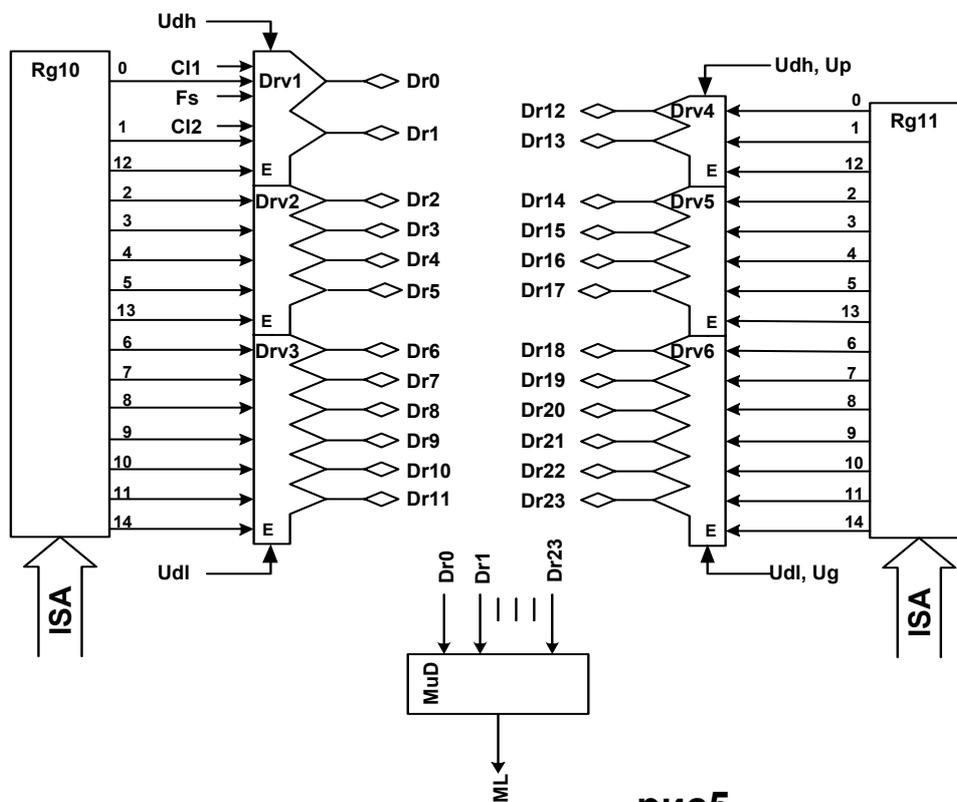


рис5

и значительно расширяют область применения модуля.

Измерительный коммутатор MuM обеспечивает подключение к адаптеру и далее к АЦП измеряемого напряжения.

Источник тока смещения  $I_s$  формирует на выходе ток, величина которого пропорциональна входному напряжению и предназначен для обеспечения режима параметрического контроля, при котором на выбранный вывод ОК подается заданный ток смещения и измеряется напряжение.

Блок драйверов Drv (рис. 5), формирует импульсы с калиброванными уровнями и фронтами и состоит из двух групп.

Первая группа это драйверы Drv1, Drv2, Drv3, а вторая – Drv4, Drv5, Drv6.

У драйверов первой группы верхний уровень выходного напряжения  $U_{dh}$  определяется величиной напряжения  $U_4$ , а нижний уровень  $U_{dl}$  – соответственно  $U_5$ . У драйверов второй группы имеется возможность выбора выходных уровней. Этот выбор заключается в следующем: либо выходные уровни те же, что и для первой группы, то есть  $U_{dh}$ ,  $U_{dl}$ , либо эти уровни задаются величинами  $U_p$ ,  $U_g$ . Такая возможность позволяет подавать на объект контроля импульсы с различными выходными уровнями.

Каждая группа содержит по три многоканальных драйвера, а именно: двухканальный, четырехканальный и шестиканальный.

Кроме основных входов, по числу каналов, каждый драйвер имеет дополнительный вход E, с помощью которого можно перевести все выходы драйвера, по числу каналов, в третье состояние, то есть запрещает его работу. В разрешенном режиме состояние выхода драйвера определяется

состоянием соответствующего разряда регистра, который им управляет. Для первой группы это регистр Rg10, а для второй – Rg11.

Цифрами на рисунке обозначены номера разрядов соответствующих регистров. Другими словами, выход драйвера транслирует его входное состояние с учетом сигнала на управляющем входе E. Необходимо отметить дополнительные возможности драйвера Drv1, а именно: его канал Dr0 кроме состояния разряда 0 может формировать на выходе последовательность импульсов C11 или Fs (см. описание адаптера), а канал Dr1 – последовательность C12

На рис.6 показана последовательность выходных состояний одного из драйвера, если на его вход подать логическую последовательность 10110001 в моменты времени T0...T7, предварительно необходимо установить величину верхнего Vdh и нижнего Vdl уровней.

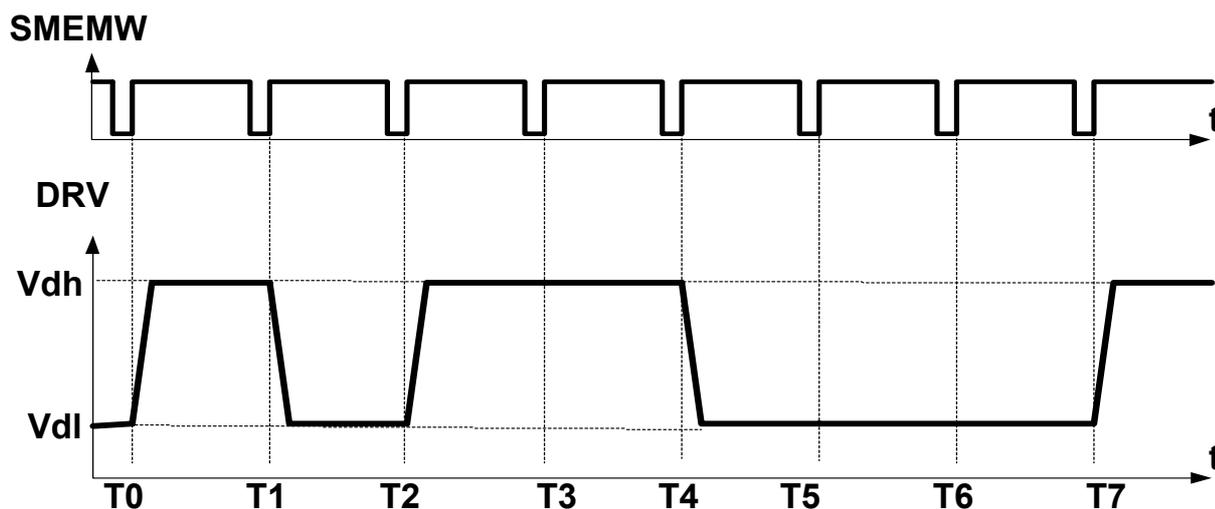
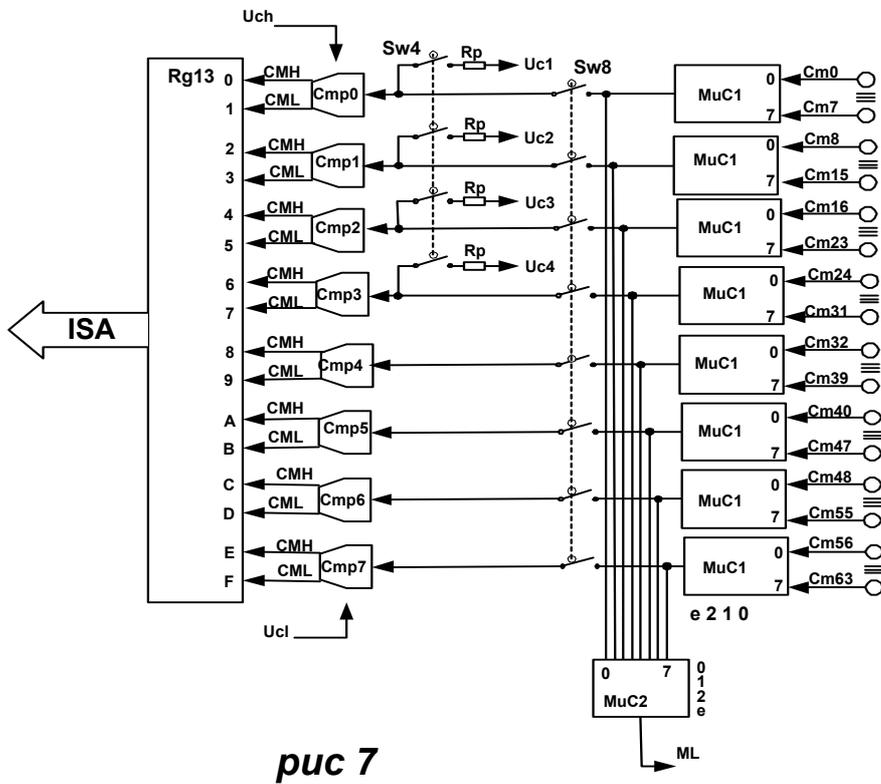


Рис.6

Коммутатор драйверов MuD обеспечивает подключение одного из 24 драйверов ( или, что то же самое, выводов объекта контроля ) к измерительной линии ML.

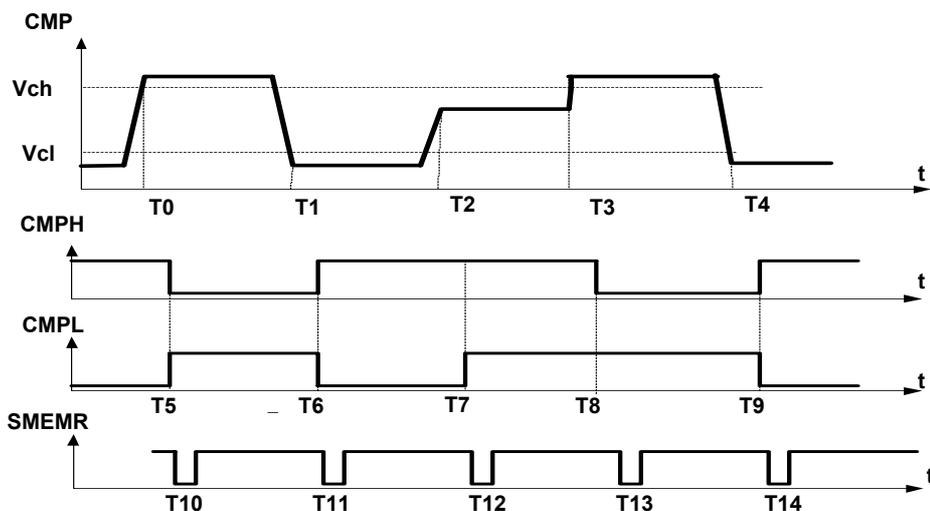
Блок компараторов Cmp ( см. рис 5 ), структура которого представлена на рис. 7, содержит 8 двухуровневых компараторов. Выходные сигналы с объекта контроля поступают на входы Cm0...Cm63 коммутаторов MuC1. Эти коммутаторы управляются таким образом, что у всех восьми коммутаторов одновременно переключаются одноименные каналы. Далее через восьмиканальный ключ Sw8 сигналы поступают непосредственно на входы компараторов. Этот ключ замкнут в режиме функционального контроля и разомкнут в режиме параметрического контроля, исключая влияние входных токов компараторов на точность параметрического контроля. Четырехканальный ключ Sw4 подключает к источнику питания объекта контроля нагрузочные резисторы Rp, обеспечивая контроль выходов с открытым коллектором и им подобным.

Коммутатор MuC2 в режиме параметрического контроля подключает к измерительной линии ML выбранный вывод объекта контроля.



Работа одного из компараторов показана на рис.8. На вход компаратора CMP поступает входной сигнал, состояние которого меняется в моменты времени  $T_0 \dots T_4$  (предварительно необходимо установить верхний  $V_{ch}$  и нижний  $V_{cl}$  уровни компарирования). Эти изменения будут обработаны компаратором и появятся на его выходах  $CMPH$  и  $CMP_L$  в моменты времени  $T_5 \dots T_9$  и по переднему фронту сигнала  $SMEMR$ , формируемого шиной ISA, в моменты времени  $T_{10} \dots T_{14}$  будут записаны в регистр оперативной памяти компьютера.

Рис. 8



## Программное обеспечение

Программа контроля для персонального тестера разрабатывается на языке программирования Object Pascal в так называемой среде быстрого создания Windows-приложений Delphi.

При этом используется многомодульная структура проекта Delphi. Главный программный модуль проекта реализует алгоритм контроля, используя элементарные процедуры и функции управления КИ ресурсами, содержащиеся в дополнительном программном модуле с именем FPV.pas, и в библиотеке Lfp.dll.

Кроме процедур и функций этот программный модуль содержит глобальные переменные, определяющие работу процедур.

### **Переменные** (после двоеточия указан тип):

Rg0s...Rg17s:word – в этих переменных хранится промежуточное содержимое регистров управления КИ ресурсами. Над этими переменными можно производить все операции, предусмотренные их типом.

Rg0b...RgAb, Rg10b...Rg17b, Rg20b...Rg27b:word – эти переменные с абсолютными адресами представляют собой содержимое регистров управления, адресуемых из программы, и физически расположенных на плате КИ адаптера и КИ модуля.

Эти переменные допускают только те операции обмена данными с переменными вида RgXs, которые обеспечивает их схемотехническая реализация.

Rg131s...Rg136s:word – используются для диагностики.

**Tscl, Tisa, Td1, Td2**:Double – хранят значения следующих величин в мкс:

**Tscl** – длительность периода следования импульсов SYS CLK,

**Tisa** – длительность удлиненного цикла шины ISA,

**Td1** – величина задержки перед выполнением серии измерений,

**Td2** – величина задержки между отдельными измерениями в серии;

**A, X, Y**:Double – эти переменные представляют формируемые и измеряемые величины.

**Kda, Sda**:Double – поправки для DAC (ЦАП);

**Kad, Sad**:Double – поправки для ADC (АЦП).

**Ar\_Y, Ar\_P, Ar\_Q** : Double - эти массивы содержат соответственно следующие данные о серии измерений ( процедура X\_n ):

- результаты измерений,
- остаточные погрешности,
- квадраты остаточных погрешностей.

**Sgm**:Double – среднеквадратичное отклонение серии измерений.

**N** : integer – количество измерений в серии;

**Chn**: Char – номер канала ЦАП;

**Chan**: Word – номер канала коммутатора MuIn;

**Fw**:word – представление нормы частоты;

**Nmd**: Char – номер активного Кимодуля, может принимать значение 1 или 2 ( активным является тот модуль, с которым происходит обмен данными );

**ChIcc** : Char – номер активного канала измерителя тока потребления, может принимать значения 0, 1, 2, или 3 ( активным является тот канал, к которому в данный момент времени подключен АЦП для измерения тока потребления);

Процедуры установки обеспечивают установку требуемого значения задаваемой величины (параметра контроля). Однако это не означает, что установленное значение станет активным. С этой точки зрения процедуры установки разделяются на два типа. Первые управляют постоянно активной величиной (постоянно активным ресурсом), изменение значения которой сразу изменяет соответственно параметры контроля.

Процедуры второго типа работают с ресурсами с управляемой активностью. В активном состоянии ресурса поведение процедур обоих типов одинаково, в неактивном – процедуры второго типа не оказывают влияния на параметры контроля. Активизацией ресурсов управляют процедуры Подключения-Отключения.

Процедуры установки включают в свой состав:

**SeTisa**(Tisa:Double) - процедура обеспечивает установку длительности удлиненного цикла шины Tisa. Длительность цикла задается в микросекундах и выбирается из диапазона 2...15 мкс включительно и имеет дискретность равную длительности периода следования импульсов Tsc1.

**SeCl1**(V1,V2:Double) – процедура обеспечивает установку времени задержки начала Tbe=V1 и окончания Ten=V2 импульса CL1. Временные величины имеют дискретность равную длительности периода следования импульсов Tsc1, задаются в микросекундах и выбираются из диапазона 0...64Tsc1.

**SeCl2**(V1,V2:Double) – процедура обеспечивает установку времени задержки начала Tbe=V1 и окончания Ten=V2 импульса CL2. Временные величины имеют дискретность равную длительности периода следования импульсов Tsc1, измеряется в микросекундах и выбирается из диапазона 0...64Tsc1.

**SeUcc**(V:Double) – процедура обеспечивает установку величины напряжения питания Ucc=V объекта контроля, которая задается в милливольтх, имеет дискретность 2 мВ и выбирается из диапазона –8...+8 В.

**SeUs**(V:Double) – процедура обеспечивает установку величины напряжения смещения Us=V с аналогичными характеристиками.

**SeUdh**(V:Double) – процедура обеспечивает установку величины напряжения верхнего уровня драйверов Udh =V с аналогичными характеристиками.

**SeUdl**(V:Double) – процедура обеспечивает установку величины напряжения нижнего уровня драйверов Udl =V с аналогичными характеристиками.

**SeUch**(V:Double) – процедура обеспечивает установку величины напряжения верхнего уровня компараторов Uch =V с аналогичными характеристиками.

**SeUcl**(V:Double) – процедура обеспечивает установку величины напряжения нижнего уровня компараторов Ucl =V с аналогичными характеристиками.

**SeUp**(X:Double) – процедура обеспечивает установку величины напряжения Up=X с аналогичными характеристиками.

**SeUg**(V:Double) – процедура обеспечивает установку величины напряжения Ug =V с аналогичными характеристиками.

**SeIs**(V:Double) – процедура обеспечивает установку величины тока смещения Is=V, которая задается в микроамперах, имеет дискретность 0.25 мкА и выбирается из диапазона –1000...+1000 мкА.

Измерительные процедуры обеспечивают проведение серии задержанных измерений величины тока или напряжения с определением результата в виде среднего арифметического  $\bar{Y}$  и среднеквадратического отклонения  $S_{gm}$ . Поэтому для нормальной работы процедур необходимо определить переменные  $N$ ,  $Tsc1$ ,  $Tisa$ ,  $Td1$ ,  $Td2$ .

Измерительные процедуры включают в свой состав:

**MeIc**(ChIcc:Char) – процедура обеспечивает измерение тока потребления в выбранном посредством параметра ChIcc канале.

**MeIm** – процедура обеспечивает измерение тока, протекающего через измерительную линию.

**MeUm** – процедура обеспечивает измерение потенциала измерительной линии.

**MeIp** – процедура обеспечивает измерение тока на выходе источника напряжения Up.

**MeIg** – то же для источника напряжения Ug.

#### **Процедуры управления коммутаторами:**

**MuIn**(Chan:Word) – процедура обеспечивает соединение выхода коммутатора MuIn с входом, номер которого определяется параметром Chan. Если Chan:= f, то коммутатор отключает все входы от выхода.

**MuC**(Cha : word) – процедура обеспечивает подключение к измерительной линии ML одного из выводов Cm0...Cm63 контролируемого изделия в зависимости от значения параметра Cha:

Cha	Вывод	Cha	Вывод	Cha	Вывод	Cha	Вывод
00	Cm0	20	Cm16	40	Cm32	60	Cm48
01	Cm1	21	Cm17	41	Cm33	61	Cm49
02	Cm2	22	Cm18	42	Cm34	62	Cm50

03	Cm3	23	Cm19	43	Cm35	63	Cm51
04	Cm4	24	Cm20	44	Cm36	64	Cm52
05	Cm5	25	Cm21	45	Cm37	65	Cm53
06	Cm6	26	Cm22	46	Cm38	66	Cm54
07	Cm7	27	Cm23	47	Cm39	67	Cm55
10	Cm8	30	Cm24	50	Cm40	70	Cm56
11	Cm9	31	Cm25	51	Cm41	71	Cm57
12	Cm10	32	Cm26	52	Cm42	72	Cm58
13	Cm11	33	Cm27	53	Cm43	73	Cm59
14	Cm12	34	Cm28	54	Cm44	74	Cm60
15	Cm13	35	Cm29	55	Cm45	75	Cm61
16	Cm14	36	Cm30	56	Cm46	76	Cm62
17	Cm15	37	Cm31	57	Cm47	77	Cm63

**MuD** (*Cha* : word) – процедура обеспечивает подключение к измерительной линии ML одного из выводов Dr0...Dr23 контролируемого изделия в зависимости от значения переменной *Cha*:

<i>Cha</i>	Драйвер	<i>Cha</i>	Драйвер	<i>Cha</i>	Драйвер
FF0	Dr0	F0F	Dr8	0FF	Dr16
FF1	Dr1	F1F	Dr9	1FF	Dr17
FF2	Dr2	F2F	Dr10	2FF	Dr18
FF3	Dr3	F3F	Dr11	3FF	Dr19
FF4	Dr4	F4F	Dr12	4FF	Dr20
FF5	Dr5	F5F	Dr13	5FF	Dr21
FF6	Dr6	F6F	Dr14	6FF	Dr22
FF7	Dr7	F7F	Dr15	7FF	Dr23

FFF – все драйверы отключены

Процедуры выбора диапазонов должны предшествовать процедуре подключения соответствующего ресурса. К ним относятся:

**RngIc**(*Rng*:Char) – процедура обеспечивает выбор диапазона тока потребления одновременно для четырех каналов в зависимости от значения параметра *Rng* следующим образом:

<i>Rng</i>	Диапазон	Дискретность
H	-10...+10 мА	2 мкА
M	-1...+1 мА	0,5 мкА
L	-100...+100 мкА	50 нА

**RngIm**(*Rng*:Char) – процедура обеспечивает выбор диапазона тока измерительной линии в зависимости от значения параметра *Rng* следующим образом:

<i>Rng</i>	Диапазон	Дискретность
H	-2...+2 мА	1 мкА
L	-4...+4 мкА	2 нА

**RngUm**(*Rng*:Char) – процедура обеспечивает выбор диапазона тока измерительной линии в зависимости от значения параметра *Rng* следующим образом:

<i>Rng</i>	Диапазон	Дискретность
H	стандартный	стандартная
L	уменьшенная	уменьшенная

Парные процедуры подключения **On** и отключения **Of** ресурса:

**OnMI**; **OfMI** – коммутируют измерительную линию ML к контакту разъема MLp.

**OnUs**; **OfUs** – коммутируют источник напряжения Um к соответствующему контакту разъема.

**OnIs**; **OfIs** – коммутирует источник тока Is к измерительной линии ML.

**OnUc0**; **OfUc0** – коммутируют канал Uc0 источника питания к соответствующему контакту разъема.

**OnUc1**; **OfUc1** – то же канал Uc1.

**OnUc2**; **OfUc2** – то же канал Uc2.

**OnUc3**; **OfUc3** – то же канал Uc3.

**OnCmp; OfCmp** – коммутируют входы компараторов с выходами соответствующих коммутаторов MuC1, изображенных на рис. 7. Другими словами эти процедуры выполняют функцию ключа Sw8, изображенного на том же рисунке.

**OnTisa** – переводит шину ISA в режим удлиненного цикла с длительностью установленной в процедуре **SeTisa**(Tisa:Double).

**OfTisa** – возвращает шину ISA в режим нормального цикла.

Дополнительные процедуры облегчают реализацию некоторых режимов работы и включают в свой состав:

**ToDrv0**(Str:Char) – процедура обеспечивает три режима работы драйвера Dr0 в зависимости от значения параметра Str:

Str:='C' – драйвер транслирует импульс CL1, при этом разряд 0 регистра Rg10 является разрешающим (высокий уровень),

Str:='F' – драйвер транслирует импульсы Fs, формируемые на цифровом выходе генератора управляемого напряжением,

Str:='D' – драйвер транслирует состояние разряда 0 регистра Rg10.

**ToDrv1**(Str:Char) – процедура обеспечивает два режима работы драйвера Dr1 в зависимости от значения параметра Str:

Str:='C' – драйвер транслирует импульс CL2, при этом разряд 1 регистра Rg10 является разрешающим (высокий уровень),

Str:='D' – драйвер транслирует состояние разряда 0 регистра Rg10.

**U0To**(St:char) – процедура обеспечивает использование выходного напряжения U0 цифроаналогового преобразователя DAC в зависимости от значения параметра St:

St:='I' – U0 определяет значение выходной величины генератора тока Is для реализации режима 'задание тока Is – измерение напряжения Um',

St:='U' – U0 определяет значение выходной величины генератора напряжения Us для реализации режима 'задание напряжения Us – измерение тока Im',

**I\_Drv0**(E:Boolean) – процедура обеспечивает инвертирование выходного сигнала драйвера Dr0.

**Delay**(V:Double) – процедура обеспечивает паузу в работе шины ISA. Величина паузы численно равна значению параметра V, выраженному в мкс, при этом шина ISA должна находиться в режиме удлиненного цикла.

**Trg** – процедура формирует импульс, предназначенный для внешней синхронизации осциллографа.

Вспомогательные процедуры и функции используются для построения элементарных процедур, описанных выше, и включают в свой состав:

**MuM**(Ch1 : word); **RngUm**(Rng:Char);

**RgW**(var RgS, Rg1, Rg2 : word);

**Me**(Chan:Word);

**Dac4**(chn:char;V:Double); **IncFw**; **DecFw**;

**max191**:word; **ADC**:Double;

## Погрешности и поправки

Как известно, основное уравнение теории измерений связывает измеряемую величину  $X[e]$ , результат измерения  $Y[e]$ , систематическую  $Dc[e]$  случайную  $D~[e]$  погрешности следующим соотношением:

$$Y[e]=X[e]+Dc[e]+D~[e] \quad [1]$$

Знак [e] после обозначения величины подчеркивает то, что численно она выражена в единицах измерения этой величины. Для повышения точности измерений необходимо скомпенсировать систематическую составляющую погрешности введением поправки, которая численно равна этой составляющей, но имеет обратное значение.

Рассмотрим подробнее алгоритм технической реализации этой задачи, который в общем виде можно разбить на следующие два этапа:

1 – измеряемая величина  $X[e]$  поступает на вход преобразователя измеряемой величины в цифровой код  $A$ . Этот преобразователь является источником погрешности измерения  $Dc[e]+D\sim[e]$ . Учитывая схемотехнику преобразователя, очевидно, что основной вклад в систематическую составляющую погрешности описывается линейным уравнением. Таким образом этот этап можно описать следующим уравнением

$$A=K[e] * X[e]+S+D\sim, \quad [2a]$$

где:  $K[e]$  отражает погрешность коэффициента передачи преобразователя;

$S$  отражает погрешность сдвига начального уровня.

2 – на этом этапе код  $A$  фиксируется, а приобретенная систематическая погрешность компенсируется введением поправки:

$$Y[e]= (A-S)/K[e]= (K[e] * X[e]+S+D\sim-S)/ K[e]= X[e]+D\sim[e]. \quad [3a]$$

Процедуры типа *MeX* реализуют это уравнение и для их нормального выполнения должны быть предварительно определены переменные *Kad* и *Sad*, которые имеют свои значения для разных процедур и диапазонов.

В отношении формирователей уравнение [1] полностью справедливо за исключением того, что  $X[e]$  формируемая величина, а  $Y[e]$  это – результат формирования.

Что касается этапов, то здесь произошли существенные изменения. На первом этапе вводится поправка и формируется соответствующий цифровой код  $A$ , а на втором этапе этот код подается непосредственно на вход формирователя, который вносит погрешность, компенсирующую введенную поправку, и формирует на выходе результат без систематической погрешности. Эти этапы описываются уравнениями [2b] и [3b]

$$A=(X[e]-S[e]+ D\sim[e])/K[e], \quad [2b]$$

$$Y[e]= K[e]* A-S [e]= X[e]+D\sim[e]. \quad [3b]$$

Работу формирователей обеспечивают процедуры типа *SeX*, для нормальной работы которых необходимо предварительно определить переменные *Kda* и *Sda*.

### ***Технические данные***

В приложении приведены основные технические данные и нумерация разъемов для подключения объекта контроля. Эти сведения относятся к персональному тестеру изображенному на рис. 1, то есть один КИ адаптер – один КИ модуль.

Однако к одному адаптеру можно подключить два модуля. В этом случае необходимо учитывать следующее:

- первая и вторая группа драйверов второго модуля управляются регистрами Rg20 и RG21 соответственно;
- компараторы второго модуля соединены с регистром Rg23;
- перед выполнением процедур управления какого-либо модуля, переменной Nmd присвоить значение номера выбранного модуля, в прочем это условие необходимо соблюдать и для одномодульного варианта.

### ***Исполняемые программы***

Рассмотрим процесс разработки исполняемой программы, которая реализует режим измерения тока утечки. Алгоритм решения этой задачи выглядит следующим образом:

1. Установим ресурсы в исходное состояние.
2. Включим требуемый канал коммутатора для подсоединения к измерителю выбранного вывода объекта контроля.
3. Зададим напряжение, при котором будет измеряться ток утечки.
4. Произведем измерения тока утечки.
5. Продемонстрируем результаты измерений.

Посмотрим, как выглядит реализация этого алгоритма. Необходимо заметить, что это измерение состоит из серии отсчетов с последующим вычислением среднего арифметического и среднеквадратического отклонения. Первая характеристика определяет измеряемую величину, а вторая оценивает погрешность измерения. Однако, для повышения производительности можно произвести однократное измерение. В качестве измеряемой выберем цепь Dr19, которая соединена с контактом 38 разъема ХТ3. Для того чтобы задать ток, подключим резистор величиной 200 КОм между контактами 38 и 70.

Сначала создадим новый проект с именем UsIm, а форме присвоим имя F\_UsIm. Подключим к проекту модуль FPV. Поместим на форму кнопку, с помощью которой будет инициироваться процесс измерения тока утечки, и присвоим ей имя Eхе. Поместим на форму метки Label1 и Label2, которые будут показывать результат измерений и среднеквадратическое отклонение соответственно. Слева от первой метки поместим метку с именем Is=, а слева от второй – с именем Sgm=. Справа от каждой из двух первых меток поместим по метке с именем nA. Таким образом, за несколько минут мы построили интерфейс пользователя в графической форме, который обеспечит доступ как к измерительным, так и к системным ресурсам разрабатываемого персонального тестера.

Теперь можно переходить непосредственно к разработке исполняемой программе контроля, реализующей решение поставленной задачи. Дважды щелкнем по кнопке и пройдем обозначенные выше пять шагов:

**1. Nmd:='1';**

// Выбран первый КИмодуль

**N:=10;**

// Измерение будет содержать 10 отсчетов

**OfUs; OfIs;**

// От измерительной линии отключаются

// генераторы напряжения и тока смещения

**Rg10b:=\$0000; Rg11b:=\$0000;**

// Драйверы переводятся в закрытое состояние

**Tscl:= 0.11972;**

// Задается длительность периода импульсов SYSCLK

**Tisa:=2; SeTisa(Tisa); OnTisa;**

// Задается величина удлиненного цикла

// и включается этот режим

**RngUm('H');**

// Включается верхний диапазон измерения

**2. MuD(\$3ff);**

// Измерительная линия подключается к цепи Dr19

**3. Kda:=1.989; Sda:=-7.0;**

// Устанавливаются поправочные коэффициенты

// для напряжения смещения Us

**SeUs(400);**

// Задается величина напряжения смещения Us=400mV

**OnUs;**

// К измерительной линии подключается напряжение смещения

**4. RngIm('L');**

// Выбирается диапазон тока утечки

**Kad:=0.3075; Sad:=-0.50;**

// Устанавливаются поправочные коэффициенты

// для измерителя тока утечки

**Td1:=100000; Td2:=100;**

// Устанавливается задержка в мкс начала измерений

// и пауза между отсчетами в серии

**MeIm;**

// Иницируются измерения

**5. Label1.Caption:=Format("%.1f',[Y]);**

// Показывается измеренная величина тока утечки

// как среднее арифметическое серии отсчетов

**Label2.Caption:=Format("%.1f',[Sgm]);**

// Показывается среднеквадратическое отклонение

// серии отсчетов

**For i:=Low(Ar\_Y) To High(Ar\_Y) Do**

**Label7.Caption:=Label7.Caption+Format("%.1f',[Ar\_Y[i]])+#10;**

// Показывается вся серия отсчетов.

Так выглядит программа измерения тока утечки с подробными комментариями. На разработку этой программы было затрачено около 20 минут.

### ***Особенности применения***

Представленная система модулей может быть использована как для производственного контроля так и для лабораторных исследований различных изделий электронной техники ( от операционных усилителей до микропроцессоров ).

Наибольший эффект достигается от группового контроля однотипных изделий.

Некоторые из указанных в приложении технических характеристик могут быть изменены. Это относится к диапазонам, а соответственно к разрешающим способностям LSB, задаваемых и измеряемых величин. Диапазоны задаваемых величин можно расширить, если применить внешние источники с повышенным напряжением питания. Для изменения диапазона измеряемой величины необходимо на этапе изготовления изменить номинал определяющего этот диапазон резистора. Кроме этого процедура ***RngUm*** позволяет уменьшить диапазоны измеряемых величин примерно в 4 раза. При этом необходимо помнить, что любое изменение диапазона влечет за собой соответствующее изменение поправок.

### ***В заключение***

можно сказать, что в распоряжении разработчика измерительного оборудования имеется все необходимые составные части или, как они здесь называются, модули. Это и аппаратные средства, к которым относятся КИ модуль и КИ адаптер, и программный модуль FPV.pas.

Если к этому добавить компьютер, с установленной средой программирования Delphi5, то можно приступить к разработке двух недостающих частей персонального тестера, а именно: графического интерфейса пользователя и исполняемой программы контроля.

Такая архитектура тестера позволяет эффективно использовать измерительные ( аппаратные ) ресурсы посредством изменения интерфейса пользователя и программы контроля. А поскольку для их разработки применяется современная система разработки программ, получившая широкое распространение, это позволяет до минимума сократить трудоемкость работ по подготовке измерительного оборудования для решения конкретной задачи.

*Телефон для контактов 532-82-70*