

GOOSY
Id.: LMDgen
Version: 1.0
Date: 23-JUN-1994
Revised: June, 23 1994

G_{SI} **O**_{nline} **O**_{ffline} **S** **Y**_{stem}

LMD Data Generator

R.S. Mayer

June, 23 1994

GSI, Gesellschaft für Schwerionenforschung mbH
Postfach 11 05 52, Planckstraße 1, D-64220 Darmstadt
Tel. (0 6159) 71-0

Chapter 1

Preface

1.1 GOOSY Authors and Advisory Service

The authors of GOOSY and their main fields for advisory services are:

M. Richter GOOSY Data Management, VAX/VMS System Manager (Tel. 2394)

R. Barth GOOSY and PAW software (since 1995) (Tel. 2546)

H.G. Essel (GOOSY 1983-1993) Data Acquisition (Tel. 2491)

N. Kurz Data Acquisition (since 1992) (Tel. 2979)

W. Ott Data Acquisition (since 1994) (Tel. 2979)

People who have been involved in the development of GOOSY.

B. Dechant GOOSY software (1993-1995) (Tel. 2546)

R. S. Mayer Data Acquisition (1992-1995) (Tel. 2491)

R. Fritzsche Miscellanea (1989-1995) (Tel. 2419)

H. Grein Miscellanea (1984-1989)

T. Kroll Miscellanea, Printers (1984-1988)

R. Thomitzek Miscellanea, Printers, Terminals (1988-1989)

W. Kynast GIPSY preprocessor (1988)

W.F.J. Müller GOONET networking, Command interface (1984-1985)

H. Sohlbach J11, VME (1986-1989)

W. Spreng Display, Graphics (1984-1989)

K. Winkelmann GOOSY Data Elements, IBM (1984-1986)

1.2 Further GOOSY Manuals

The GOOSY system is described in the following manuals:

- GOOSY Introduction and Command Summary
- GOOSY Data Acquisition and Analysis
- GOOSY Data Management
- GOOSY Data Management Commands
- GOOSY Display
- GOOSY Hardware
- GOOSY DCL Procedures. GOOSY Error Recovery
- GOOSY Manual
- GOOSY Commands

Further manuals are available:

- GOOSY Buffer structures
- GOOSY PAW Server
- GOOSY LMD List Mode Data Generator
- SBS Single Branch System
- TCP-Package
- TRIGGER Bus
- VME Introduction
- OpenVMS Introduction

1.3 Intended Audience

This manual is written for GOOSY and PAW users. It assumes that the reader is familiar with most VAX-VMS concepts and commands. For VAX beginners the 'VMS Introduction' is recommended. For GOOSY beginners the 'GOOSY Introduction' is recommended. Users of the PAW- and related software are kindly asked to refer to the CERN software documentation.

1.4 Overview

- Section 2:
List Mode Data Generator: Introduction.
- Section 3:
LMD Generator Program Template.
- Section 4:
LMD Generator Functions.

The author would be grateful for any critical comment or any suggestion about this manual.

1.5 GOOSY Copy Right

The GOOSY software package has been developed at GSI for scientific applications. Any distribution or usage of GOOSY without permission of GSI is not allowed. To get the permission, please contact Dr. M. Richter at GSI.

Registered Trademarks are not explicitly noted.

Chapter 2

Introduction

2.1 LMD Data Generator: Introduction

The LMD Data Generator (LMDgen) allows the generation of GOOSY List Mode Data of the VME and SBS (Single Branch System) Buffer type 10,1 and event type 10,1 and 10,2. The data are CAMAC, FASTBUS like or user defined random events. One may select:

- processor ID, crate(s) and module types and slots.
- Gaussian and exponential shape
- probability per module
- mean value and sigma for gaussian and
- lowest channel and slope for exponential shape
- offset times channel number per module

The file header may be selected as well as the buffer size. The generated list mode data file consists to the GOOSY convention and may be used to check analysis programs in a realistic way.

Use `GOO$EXAMPLES:MRAN_MAIN.PPL` as a template to write your own program for simulated LMD files.

The second part of this manual describes the used functions.

Chapter 3

LMDgen Program Template

3.1 LM Data Generator Template

The LMD Generator (LMDgen) allows the generation of GOOSY List Mode Data generation of the VME and SBS (Single Branch System) type 10,1 and event type 10,1 and 10,2 (see chapter 2).

copy the P1/1 code GOO\$EXAMPLES:MRAN_MAIN.PPL and modify it to your needs. This example creates a LMD file similar to the event structure used by the KAOS collaboration in 1992. The available functions are described in chapter 4.

Compile with `$ comp MRAN_MAIN.`

link with `$ link MRAN_MAIN.`

MRAN_MAIN

CALLING run goo\$examples:MRAN_MAIN

PURPOSE Program(sample) to generate events of typ 10,1 and subevents of type 10,1 and type 10,2.

Description

CALLING run goo\$examples:MRAN_MAIN

FUNCTION An event structure is defined by calling a set of routines. Several event 'types' may be specified with a trigger number. For each event 'type' the subevents may be individually defined. With the generated ListMode file, analysis programs, unpack routines etc. can be tested.

The event definitions is given by the event type, subtype (10,1), trigger number and a pointer to the subevent definition (type=10).

After the event definition the crates and sub-crates with processor ids have to be specified for CAMAC and FASTBUS crates.

In the CAMAC crate definition, the parameters mentioned below ar given directly.

For FASTBUS, first the crate is defined, then the the modules in the crate have to be defined with their module id and the slot number together with the following parameters.

In CAMAC crates and FASTBUS modules, the following parameters may be specified:

- 2., 3. subevent type (=10) and subtype
4. Processor id
5. Subcrate number
6. Control number
7. Number of channels

Shape of random generated distribution:

- a. gaussian shape
 - b. exponential function
8. Mean value (a) or Lower Edge of exp. slope (b)
 9. Sigma:
 - (a) sigma of gauss function
 - (b) slope of exponential function

- (a) $\text{Sigma} \geq 0$. (b) $\text{Sigma} < 0$.
10. Offset of R_MEAN with the channel number
11. Probability a channel is fired ($0. \leq P \leq 1.$)

REMARKS

Template to be modified by the user individually.
Copy GOO\$EXAMPLES:MRAN_MAIN.PPL and adapt it to your problem to solve, compile and link.

Implementation

Version 1.00
Author R.S.Mayer
Last Update 12-Dec-1992
Object libr. -

Updates

Updates	Date	Purpose
	14-Dec-1993	Changed argument typ in I\$RAN_CAMDEF() (RSM)
	24-Jun-1994	Documentation (RSM)

Internals

Utility EXAMPLES
Compile lib. GOOINC
Home direct. GOO\$EXAMPLES
Created 12-Dec-1992

Chapter 4

LMDgen Functions

4.1 LM Data Generator Functions

for the LMD generation, the following functions are available:

- **I\$RAN_EVENTINI**: Initialize trigger number.
- **I\$RAN_CAMDEF**: Define CAMAC crate.
- **I\$RAN_FBDEF**: Define FASTBUS crate.
- **I\$RAN_MODDEF**: Define module(s) in FASTBUS crate.
- **I\$RAN_SHODEF**: Show definition of event for trigger number.
- **I\$RAN_OPNFIL**: Open file, define buffer size, write file header.
- **I\$RAN_EVFIL**: Generate event for given trigger number
- **I\$RAN_CLSFIL**: Close file.

FUNCTION Initializes an event. Creates the pointer array to a structure, where the event-, subevent- and Fastbus module information will be held. The Pointer array must be NULL() when calling the procedure and will be set.

 The event specification is the top of the event definition tree. Its identification is the pointer array P_RANEVT:

 The second level is the CAMAC and FASTBUS definition (I\$RAN_CAMDEF resp. I\$RAN_FBDEF). Their identification is the processor id and the subcrate number. In case of CAMAC, the definition is complete.

 The third level in the event definition is only relevant for FASTBUS. Each module in a crate has to be specified (I\$RAN_MODDEF). The crate is defined by its processor id and the subcrate number.

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date Purpose 05-Nov-1993 Bugfix return status (RSM)
----------------	---

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Source file	I\$RAN_EVENTINI.PPL
Created	12-Dec-1992

I\$RAN_OPNFIL

```
STS=I$RAN_OPNFIL(C_FILNAM,C_RUN,C_EXP,  
                 P_EVTCTRL,outbuf_siz,  
                 L_spanint,L_prompt)
```

PURPOSE Open output LMD file for random event generation Prompts for GOOSY file header.

ARGUMENTS

C_FILNAM LMD file name

C_RUN Default run ID.

C_EXP Default experiment name

P_EVTCTRL Pointer to control structure

outbuf_siz Output file record size

L_spanint no buffer spanning after L_spanint buffers

L_prompt Flag: prompt for file header
0 Do not prompt for information.
1 Prompt for information.

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_OPNFIL(C_FILNAM,C_RUN,C_EXP,
P_EVTCTRL,outbuf_siz,
L_spanint,L_prompt)

ARGUMENTS

C_FILNAM	LMD file name Input CHAR(*) VAR
C_RUN	Default run ID. Input CHAR(*) VAR
C_EXP	Default experiment name Input CHAR(*) VAR
P_EVTCTRL	Pointer to control structure Input POINTER (= NULL()) Output POINTER
outbuf_siz	Output file record size Input BIN FIXED(31)
L_spanint	no buffer spanning after L_spanint buffers 0 or 1: no buffer spanning Input BIN FIXED(31)
L_prompt	Flag: prompt for file header 0 Do not prompt for information. 1 Prompt for information. Input BIN FIXED(15)

FUNCTION Open output LMD file for random event generation. Select record size and buffer spanning. The pointer is set to the control structure SI\$RANEVCL. Prompts for GOOSY file header.

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
	30-May-1994	Update new buffer convention /RSM

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_CAMDEF

**STS=I\$RAN_CAMDEF(P_RANEVT,I_TYP,LSUBTYP,IPROCID,
H_SUBCRATE,H_CONTROL,I_CHAN,
R_X0,R_SIGMA,I_OFFSET,R_PROB)**

PURPOSE	Adds the definition of a CAMAC crate and subevent.
ARGUMENTS	
P_RANEVT(2)	Pointer to event definition structure
I_TYP	Event type (implemented for type 10)
LSUBTYP	Event subtype (implemented for subtype 1)
IPROCID	Processor id
H_SUBCRATE	Subcrate number
H_CONTROL	Control number = proc. type (see GOOINC(\$FICTREP))
I_CHAN	number of channels
R_X0	(a) Mean of gaussian distribution or (b) Lower edge of exponential distribution
R_SIGMA	(a) ($\geq 0.$): Sigma of gaussian distribution or (b) ($< 0.$) : Slope of exponential distribution Automatic offset with channel number
R_PROB	probability a channel will be fired.
Return type	BIN FIXED(31)
Status codes	-
Initialize	-
Include name	-

Description

CALLING STS=I\$RAN_CAMDEF(P_RANEVT,I_TYP,I_SUBTYP,I_PROCID,
H_SUBCRATE,H_CONTROL,I_CHAN,
R_X0,R_SIGMA,I_OFFSET,R_PROB)

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure Input POINTER

I_TYP Event type (implemented for type 10) Input BIN FIXED(15)

I_SUBTYP Event subtype (implemented for subtype 1) Input BIN FIXED(15)

I_PROCID Processor id Input BIN FIXED(15)

H_SUBCRATE Subcrate number Input BIN FIXED(7)

H_CONTROL Control number = proc. type (see GOOINC(\$FICTREP)) Input BIN FIXED(7)

I_CHAN number of channels Input BIN FIXED(15)

R_X0 (a) Mean of gaussian distribution or (b) Lower edge of exponential distribution Input BIN FLOAT(24)

R_SIGMA (a) ($\geq 0.$): Sigma of gaussian distribution or (b) ($< 0.$) : Slope of exponential distribution Input BIN FLOAT(24) Automatic offset: Mean of gaussian or edge of exponential distribution is shifted with channel_number * I_OFFSET. Input BIN FIXED(15)

R_PROB probability a channel will be fired. (0.: never, 1.:always, 0. > R_PROB < 1.: randomly chosen. Input BIN FIXED(7)

FUNCTION

A CAMAC crate and subcrate definition will be appended to an existing event definition which is referenced by the pointer array P_RANEVT. Only events of typ 10,1 and subevents of type 10 are supported. For several crate definitions (sub- events) at least one of the two following parameters must be different: I_PROCID or H_SUBCRATE. For the CAMAC subevents the following parameters may be specified: 1. Pointer to the event definition structure 2., 3. subevent type and subtype 4. Processor id 5. Subcrate number 6. Control number 7. Number of channels Shape of random generated distribution:

- a. gaussian shape
- b. exponential function

8. Mean value (a) or Lower Edge of exp. slope (b) 9. Sigma: (a) sigma of gauss function
(b) slope of exponential function
(a) Sigma ≥ 0 . (b) Sigma < 0 .
10. Offset of R_MEAN with the channel number 11. Probability a channel is fired ($0. \leq P \leq 1.$)

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
	14-Dec-1993	Changed H_CHAN to L_CHAN (RSM)

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_FBDEF

<p style="text-align: center;">STS=I\$RAN_FBDEF(P_RANEVT,I_TYP,I_SUBTYP,I_PROCID, H_SUBCRATE,H_CONTROL)</p>
--

PURPOSE	Adds the definition of a FASTBUS crate and subevent
ARGUMENTS	
P_RANEVT(2)	Pointer to event definition structure
I_TYP	Event type (implemented for type 10)
I_SUBTYP	Event subtype (implemented for subtype 1)
I_PROCID	Processor id
H_SUBCRATE	Subcrate number
H_CONTROL	Control number = proc. type (see GOOINC(\$FICTREP))
Return type	BIN FIXED(31)
Status codes	-
Initialize	-
Include name	-

Description

CALLING	STS=I\$RAN_FBDEF(P_RANEVT,I_TYP,I_SUBTYP,I_PROCID, H_SUBCRATE,H_CONTROL)
ARGUMENTS	
P_RANEVT(2)	Pointer to event definition structure Input POINTER
I_TYP	Event type (implemented for type 10) Input BIN FIXED(15)
I_SUBTYP	Event subtype (implemented for subtype 1) Input BIN FIXED(15)

L_PROCID	Processor id Input BIN FIXED(15)
H_SUBCRATE	Subcrate number Input BIN FIXED(7)
H_CONTROL	Control number = proc. type (see GOOINC(\$FICTREP)) Input BIN FIXED(7)
FUNCTION	A FASTBUS crate and subcrate definition will be appended to an existing event definition which is referenced by the pointer array P_RANEVT. Only events of typ 10,1 and subevents of type 10 are supported. For several crate definitions (sub- events) at least one of the two following parameters must be different: L_PROCID or H_SUBCRATE. For the FASTBUS subevents the following parameters may be specified: 1. Pointer to the event definition structure 2., 3. subevent type and sub-type 4. Processor id 5. Subcrate number 6. Control number
NOTE	Need to specify FB modules by calls of I\$RAN_MODDEF

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date Purpose
----------------	--------------

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_MODDEF

```

STS=I$RAN_MODDEF(P_RANEVT,LPROCID,H_SUBCRATE,
                 LMODID,H_SLOT,H_RANGE,H_EVTFLAG,
                 LCHAN,R_X0,R_SIGMA,
                 LOFFSET,R_PROB)

```

PURPOSE Adds the definition of a CAMAC crate and subevent.

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure

LPROCID Processor id

H_SUBCRATE Subcrate number

LMODID FASTBUS manufactors module id

H_SLOT FASTBUS crate slot number

H_RANGE Range 'bit' (0 or 1)

H_EVTFLAG Event flag

LCHAN Number of channels in the module

R_X0 (a) Mean of gaussian distribution or (b) Lower edge of exponential distribution

R_SIGMA (a) ($\geq 0.$): Sigma of gaussian distribution or (b) ($< 0.$) : Slope of exponential distribution Automatic offset with the channel number

R_PROB Probability a channel will be fired.

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_CAMDEF(P_RANEVT,I_TYP,I_SUBTYP,I_PROCID,
H_SUBCRATE,H_CONTROL,I_CHAN,
R_X0,R_SIGMA,I_OFFSET,R_PROB)

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure Input POINTER

I_PROCID Processor id Input BIN FIXED(15)

H_SUBCRATE Subcrate number Input BIN FIXED(7)

I_MODID FASTBUS manufactors module id Input BIN FIXED(15)

H_SLOT FASTBUS crate slot number Input BIN FIXED(7)

H_RANGE Range 'bit' (0 or 1) Input BIN FIXED(7)

H_EVTFLAG Event flag Input BIN FIXED(7)

I_CHAN number of channels in the module Input BIN FIXED(15)

R_X0 (a) Mean of gaussian distribution or (b) Lower edge of exponential distribution Input BIN FLOAT(24)

R_SIGMA (a) ($\geq 0.$): Sigma of gaussian distribution or (b) ($< 0.$) : Slope of exponential distribution Input BIN FLOAT(24) Automatic offset: Mean of gaussian or edge of exponential distribution is shifted with channel_number * I_OFFSET. Input BIN FIXED(15)

R_PROB probability a channel will be fired. (0.: never, 1.:allways, 0. > R_PROB < 1.: randomly chosen. Input BIN FIXED(7)

FUNCTION Module definitions will be appended to an existing FASTBUS crate, i.e. subevent definition, in an existing event. The reference to the event definition is the pointer array P_RANEVT.

Different modules in a FASTBUS subevent are identified by their slot number H_SLOT.

For the FASTBUS module(s) the following parameters must be specified: 1. Pointer to the event definition structure 2. Processor id 3. Subcrate number 4. Manufactors module id 5. Slot number 6. Range 'bit' 7. Event flag 8. Number of channels
Shape of random generated distribution:

- a. gaussian shape
- b. exponential function

9. mean value (a) or lower edge of exp. slope (b) 10. Sigma: (a) sigma of gauss function

(b) slope of exponential function

(a) Sigma ≥ 0 . (b) Sigma < 0 .

11. Offset of R_MEAN with the channel number 12. Probability a channel is fired (0. $\leq P \leq 1$.)

EMARKS

Define the modules in the same order as you will read out your FAST-BUS crate during the experiment

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
	14-Dec-1993	Changed H_CHAN to L_CHAN (RSM)

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_SHODEF

STS=I\$RAN_SHODEF(P_RANEVT)

PURPOSE Shows the complete definition tree for random event generation.

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_SHODEF(P_RANEVT)

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure Input POINTER

FUNCTION Shows the complete definition tree for random event generation.

Implementation

Version 1.00

Author R.S.Mayer

Last Update 12-Dec-1992

Object libr. GOOSHLIB

Shar. Image GOOIOSHR

Shar.Im.lib. GOOSHR

Updates

Updates	Date	Purpose
	14-Dec-1993	SI\$RANMOD modified (HA\$RANMOD_noofch) (RSM)

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_EVFIL

STS=I\$RAN_EVFIL(P_RANEVT,P_EVTCTRL)

PURPOSE Generate an random event and write it to LMD file

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure

P_EVTCTRL Pointer to control structure

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_EVFIL(P_RANEVT,P_EVTCTRL)

ARGUMENTS

P_RANEVT(2) Pointer to event definition structure Input POINTER (=NULL()) Output POINTER

P_EVTCTRL Pointer to control structure Input POINTER (= NULL()) Output POINTER

FUNCTION Generate an random event and write it to LMD file.

 The event is characterised by the pointer
 P_RANEVT to the event definition.

 The output is characterised by the definitions
 where P_EVTCTRL points to.

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
	14-Dec-1993	SI\$RANMOD modified (IA\$RANMOD_noofch) (RSM)

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_CLSFIL

STS=I\$RAN_CLSFIL(P_EVTCTRL)

PURPOSE Close output LMD file for random event generation

ARGUMENTS

P_EVTCTRL Pointer to control structure

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_CLSFIL(P_EVTCTRL)

ARGUMENTS

P_EVTCTRL Pointer to control structure Input POINTER

FUNCTION Closes output LMD file for random event generation. Select record size and buffer spanning. P_EVTCTRL points to the control structure SI\$RANEVCL.

Implementation

Version 1.00

Author R.S.Mayer

Last Update 12-Dec-1992

Object libr. GOOSHLIB

Shar. Image GOOIOSHR

Shar.Im.lib. GOOSHR

Updates

Updates	Date	Purpose
---------	------	---------

Internals

Utility		IO
Compile lib.		GOOINC
Home direct.		GOO\$IO
Created	12-Dec-1992	

I\$RAN_BUFFIL

STS=I\$RAN_BUFFIL(P_EVENT,P_EVTCTRL)

PURPOSE Writes a random event to the LMD file

ARGUMENTS

P_EVENT Pointer to the event data

P_EVTCTRL Pointer to control structure

Return type BIN FIXED(31)

Status codes -

Initialize -

Include name -

Description

CALLING STS=I\$RAN_BUFFIL(P_EVENT,P_EVTCTRL)

ARGUMENTS

P_EVENT Pointer to event data Input POINTER Output POINTER (may be modified)

P_EVTCTRL Pointer to control structure Input POINTER (= NULL()) Output POINTER

FUNCTION Packs a random event of type 10,1 into the buffer of a LMD file. P_EVENT is the pointer to the event data. Record size, buffer spanning etc. is characterised by the definitions where P_EVTCTRL points to.

Implementation

Version	1.00
Author	R.S.Mayer
Last Update	12-Dec-1992
Object libr.	GOOSHLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
	30-May-1994	Update for new buffer convention /RSM

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_GAUSS

$R = I\$RAN_GAUSS(R_MEAN,R_SIGMA)$

PURPOSE Real function calculating (a) gaussian distribution of random numbers
(b) random numbers with an exponential distribution.

ARGUMENTS

R_MEAN (a) mean value (b) lower edge of exponential
distribution

R_SIGMA (a) $R_SIGMA \geq 0$. : Sigma of the gaussian
distribution.
(b) $R_SIGMA < 0$. : slope of the exp. distribution.

Return type BIN FLOAT(24)

Status codes -

Initialize -

Include name -

Description

CALLING $R = I\$RAN_GAUSS(R_MEAN,R_SIGMA)$

ARGUMENTS

R_MEAN (a) mean value (b) lower edge of exponential
distribution.
Input BIN FLOAT(24)

R_SIGMA (a) $R_SIGMA \geq 0$. : Sigma of the gaussian
distribution.
(b) $R_SIGMA < 0$. : slope of the exp. distribution. Input BIN
FLOAT(24)

FUNCTION Returns a random number with the probability of (a) gaussian distribution with mean and sigma (b) exponential function with its lower edge at R_MEAN and a slope R_SIGMA (R_SIGMA < 0.).

Implementation

Version	1.00
Author	R.S.Mayer, Y.Leifels
Last Update	12-Dec-1992
Object libr.	GOOSHRLIB
Shar. Image	GOOIOSHR
Shar.Im.lib.	GOOSHR

Updates

Updates	Date	Purpose
---------	------	---------

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

I\$RAN_PROB

B = I\$RAN_PROB(R_PROB)

PURPOSE Returns bit '1' with the probability R.

ARGUMENTS

R_PROB Probability (0.>= R_PROB >= 1.)

Return type BIT(1) ALIGNED

Status codes -

Initialize -

Include name -

Description

CALLING B = I\$RAN_PROB(R_PROB)

ARGUMENTS

R_PROB Probability (0.>= R_PROB >= 1.) Input BIN FLOAT(24)

FUNCTION Returns bit '1' with the probability R and bit '0' with the probability (R-1). R_PROB: 0.: allways '0'
1.: allways '1'

Implementation

Version 1.00

Author R.S.Mayer

Last Update 12-Dec-1992

Object libr. GOOSHR LIB

Shar. Image GOOIOSHR

Shar.Im.lib. GOOSHR

Updates

Updates	Date Purpose
---------	--------------

Internals

Utility	IO
Compile lib.	GOOINC
Home direct.	GOO\$IO
Created	12-Dec-1992

Contents

1	Preface	1
1.1	GOOSY Authors and Advisory Service	1
1.2	Further GOOSY Manuals	2
1.3	Intended Audience	3
1.4	Overview	3
1.5	GOOSY Copy Right	3
2	Introduction	5
2.1	LMD Data Generator: Introduction	6
3	LMDgen Program Template	7
3.1	LM Data Generator Template	8
	MRAN_MAIN	9
4	LMDgen Functions	11
4.1	LM Data Generator Functions	12
	ISRAN_EVENTINI	13
	ISRAN_OPNFIL	15
	ISRAN_CAMDEF	18
	ISRAN_FBDEF	21
	ISRAN_MODDEF	23
	ISRAN_SHODEF	26
	ISRAN_EVFIL	28
	ISRAN_CLSFIL	30
	ISRAN_BUFFIL	32
	ISRAN_GAUSS	34
	ISRAN_PROB	36