MULTICS STAFF BULLETIN-58

TO: Distribution

FROM: Steve Webber

DATE: June 23, 1972

SUBJECT: Follow-On Integration Group Task List

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 4

TASK AREA Currently Needed Tasks

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Mew 645 F assembler. Document how to use.	RFM	C	2/7 / 2/21	3/1 / 4/1	Document being developed.	3
Document stack header and stack frame format.	SHW		12/1 / 12/1	2/22 / 3/10	Done.	1
Document call/push/return changeover strategy.	SHW		12/1 / 12/1	3/1 / 3/20	Done.	2
Document follow-on code compatibility. (For users.)	SHW		2/28 / 3/20	3/17 / 4/1	Done.	2
Document code sequences for call/push/return.	SHW		2/21 / 2/21	2/28 / 3/10	Done.	1
Convert hardcore to version II PL/I. Convert include files for alignment.	ALL	M	3/15 /	/	Begin when compiler is ready.	
Estimate bulk store requirements. Will we have enough?	RBS SHW		6/12 / 6/14	6/19 / 6/21	New metering scheme being developed (submitted). Done.	
Design new signalling and crawlout mechanism. Document giving changes.	RJF VLV		3/28 / 3/6	3/27 / 4/3	Done.	4
Scan system to find what changes with new ring hardware.	RJF		2/28 / 3/6	3/27 / 3/27	Done.	
New object segment format for ALM. (New macro names.)	RJF		3/6 / 3/27	3/27 / 5/15	Done.	3

Form revised 6/15/71

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 2 of 4

TASK AREA Currently Needed Tasks

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Design ring alarm software, implement it.	WSS	M	3/13	4/24	Tabled.	6
Convert linker to check for new type ALM programs. (Optional control-call to turn it off.)	CLJ	M	3/27 4/24	4/17 6/30	In progress.	3
interpret_mc_ for 645 processor.	MBW	M	2/23 3/13	5/1	Begun. But shelved for lack of time.	2
Modify programs to use interpret_mc_.	MBW	M	3/13	5/1	Debug, gecos, default_error handler, fiddle. Shelved.	6
Change interpret_mc_ to work on follow-on.	MBW	M	5/1	5/8	Shelved.	1
Remove alarm clock from system.	RJF		12/1 12/1	2/28 3/17	Done.	0
Figure out how we can get MST's to Phoenix.	RBS DMW (DRV)		3/6 4/15	4/3 5/19	Done.	
Do anything necessary to get MST's to Phoenix.	DMW	C	3/6 4/15	4/3 7/3	In progress.	8
Implement new stack header and virtual clock.	WSS		2/14 2/28	3/13 4/10	Done.	4

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 3 of 4

TASK AREA Currently Needed Tasks

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Do we want to retain pre-paging with bulk store?		M				
How to use bulk store on service with 645 drum?	RBS		4/24	5/8	Done. (Use drum as secondary storage probably.)	2
Write Multics bulk store dim; incorporate into page control.	RBS		4/17 5/8	5/8 5/20	Done.	3
Checkout Bulk Store DIM for Multics.		C	4/17 7/1	5/8	Wait for BS to arrive.	
Install Bulk Store on Multics.		C	9/1		Expected hardware problems incorporated into the schedule.	
Go thru system checking for RAR difficulties.					First scan complete. Few apparent problems.	
Define tasks for Performance Analysis.	SHW	C	3/1 3/1			
Reassemble system code.	ALL	C			When assembler is ready. Must be done for follow-on in Phoenix.	
Writeup rules for code generation of follow-on code.	SHW		4/17	5/1	Done. Being revised.	2
Writeup rules and procedures for insuring code compatibility.	SHW	H	4/17	5/1	Done. Being revised.	2

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 4 of 4

TASK AREA Currently Needed Tasks

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Generate program task list for tools/ DEV/SSS.	VLV		3/1 / 3/8	3/15 / 3/15	Done.	
Fix MST generator to use new key words for access attributes.	SHW		5/15 / 5/15	5/26 / 5/26	Done.	
Fix checker to recognize new key words.	SHW		5/15 / 5/15	5/26 / 5/26	Done.	

Pg 5

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 1

TASK AREA Peripheral I/O Development

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Design the PIM.	NIM RFM	M	1/1 1/1	2/28	Initial draft ready for distribution. Not quite complete.	2
Code the PIM.	RBS MAM	M	3/13	4/3	Delayed for other jobs.	5
Modify printer DIM to use the PIM.	RBS MAM	M	3/27	4/10	Delayed for lack of personnel.	1
Modify punch DIM to use the PIM.	RBS MAM	M	3/27	4/10	Delayed for lack of personnel.	1
Modify card reader DIM to use the PIM.	RBS MAM	M	3/27	4/10	Delayed for lack of personnel.	1
Checkout of PIM and new DIM's.	RBS MAM	M	4/10	5/8	Delayed for lack of personnal.	4
Modify printer DCM for PRT-300 operation.	MAM	H	4/10	5/1		3
Modify IOM software to use address extension for extended IOM.		M				

Form revised 6/15/71

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 2

TASK AREA BOS Follow-On Tasks

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Write and checkout BOS bulk store DIM.	RBS RFM		5/8 5/8	5/22 5/19	Done.	2
Write BOS T&D for bulk store.	RBS RFM	C	5/22	6/12	Wait until we know what we need.	?
Audit current BOS programs for base register usage.	DRV		3/27 3/27	4/3 5/1	No problems. Done.	1
Modify BOS to shift interrupt vector to 0 absolute.	DRV RFM WSS (NIM)		3/27 5/1	5/10 5/15	Done.	2
Lay out memory map for Multics boot-loads.	SHW NIM		2/16	2/21	Redone. BOS map has been generated.	1
Modify BOS to understand new control unit format (intfault incl alm).	DRV WSS		3/27 5/1	4/17 5/15	Done.	2
Modify BOS for use of new +privileged instructions. Clock reading.	DRV WSS		3/27 5/1	4/17 5/15	Done. 1	2
Implement new LDAC/IMW strategy for IOM.	RFM		3/27	4/3	No change necessary.	1
Modify "APND" to use new hardware SDW/ PTW format.	WSS DRV	C	4/10 5/1	4/24 6/16	Done.	2
Write BOS T&D to checkout new follow-on CPU.		H	4/24	6/12	(355 Patcher may help.) With load (bulk store, disks) page faults in instructions.	
Write BOS 1 card loader for follow-on.	RFM	C	6/26	6/30		1

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 2 of 2

TASK AREA BOS Follow-On Task

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Modify DESEG + SDW handling internal to BOS.	DRV WSS		3/27 5/1	4/10 5/15	Done.	3
Modify BOS loader (1 card program) for MTS-500.		M				1
Modify BOS loading programs to read tape with MTS-500.		M				1
Checkout BOS bootloader with MTS-500 and DSU-181/190.		M				1
Modify BOS MST writing programs (ntape).		M				2
Checkout BOS ntape package.		M				1
Modify BOS disk DIMS for DSU-181/190.		M				2
Checkout BOS DSU-181/190 DIM.		M				1
Modify BOS to reside on DSU-181/190's.		M				2
Checkout BOS running on DSU-181/190's.		M				2
Write 355 Dumper and Patcher for Multics core image.	RBS		5/22 5/22	5/31 5/31	Done.	1

Form revised 6/15/71

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 2

TASK AREA DataNet 355 Development

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Add fast list service subroutine to 355.	RBS		2/8	2/28	Done.	3
Train Doug Wells on HSLA, TTY DIM, etc.	RBS DMW		2/21 2/21	5/1	Done.	?
Analyze Phoenix written HSLA init code.	DMW RBS		4/24 4/24	5/1 5/5	Done.	1
Design init code.	RBS DMW		5/1 4/24	5/8 5/5	Done.	1
Design main module code.	RBS DMW	H	5/8 5/8	5/15	Begun. Delayed for PHOENIX TAPES proposal.	1
Add HSLA simulation capability to 355 simulator.	RBS	H	5/15 6/5	5/22 6/12	Begun. Delayed for PHOENIX TAPES proposal.	1
Debugging under simulator.	DMW	H	5/22 6/12	6/19 7/10	Begun. Delayed for PHOENIX TAPE proposal.	4
Debugging on Development machine.	DMW	H	6/19 7/10	7/17 8/7	Begun. Delayed for PHOENIX TAPE proposal.	4
Dynamically reconfigure DN-355.	DRV (RBS)	H				6
Convert software to use IDA instead of ICA.	RBS	H				2

P:) 9))

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 2 of 2

TASK AREA DataNet 355 Development

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Make test to see if GIOC simulation will be fast enough.	RBS	C			Paper analysis is complete.	

Form revised 6/15/71

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 2

TASK AREA New Software Development

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Write programs to save the history reg's.	SHW NIM	C	6/4	6/12 6/18	To be incorporated into FIM and II, etc. Done for FIM.	
Write programs to use the mode register.	SHW NIM	C	6/4	6/12 6/18	To be incorporated into FIM and II, etc. Done for FIM.	
Write GECOS system simulator using BAR mode.		L				
Study MPC and PSIA Specs.	NIM RFM	C	3/13 3/13	3/20		1
Study DSS-181 and DSS-190 Specs.	NIM RFM	C	3/13 3/13	3/20		1
Study MTS-500 Specs.	NIM	C	3/13 3/13	3/20		1
Examine changes to iom_manager for new disk channel.	NIM RFM	C				1
Code changes to iom_manager for new disk channel.		C				2
Examine modification to PIM for DSS-181/190 and MTS-500.		C				1
Design and code DSS-181/190 DIM's (Version 1).	LJS	C				4

F) 11

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 2 of 2

TASK AREA New Software Development

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Checkout DSS-181/190 DIMS.	LJS	C				2
Design MTS-500 DCM to use PIM.		H				4
Extend PIM to handle channels with multiple devices.		H				2
Code MTS-500 DCM.		H				2
Checkout MTS-500 DCM.		H				2
Design DSS-190 DIM Version II.		M				
Code DSS-190 DIM Version II.		M				
Code DSS-190 DIM Version II.		M				
Checkout DSS-190 DIM Version II.		M				
Change term_ to unsnap links in new way (new object segs).						

Form revised 6/15/71

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 1

TASK AREA 645F Bootload Checkout Summary

TASK DESCRIPTION	PERSONNEL	START	FINISH	CHANGES/STATUS
Collection 1				See attached forms.
63 segments				
Collection 2				
52 segments				
Collection 3				
13 segments				
SSS				
48 segments				
Tools				
24 segments				

GROUP FUNCTION Follow-On Integration Group Task List DATE June 23, 1972 PAGE 1 of 1

TASK AREA Miscellaneous

TASK DESCRIPTION	PERSONNEL		START	FINISH	CHANGES/STATUS	
Let linker make argument type checks (optional control).	CLJ	L	3/27	5/8		6
Change binder to make argument type checks (optional control).	MJS MAW	L				
Come up with new ALM manual - for users.		M				
Arrange for more TTY lines for follow-on configuration.		H				
Implement software for ARPA Net on follow-on hardware.		H				
Design dynamic reconfiguration of bulk store (paging device).	RBS	M				

Form revised 6/15/71

INCLUDE FILE CHANGES

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
controller_data.incl	pl1	drv	new	6/10	6/10	
dn355.incl	pl1	rbs	16-12	6/13	6/13	
fault_vector.incl	alm	drv	new	6/10	6/10	
fault_vector.incl	pl1	shw	16-10	6/10	6/10	
history_regs.incl	pl1	drv	new	6/10	6/10	
its.incl	pl1	shw	16-10	6/10	6/10	
mc.incl	alm	shw	16-10	6/10	6/10	
mc.incl	pl1	shw	16-10	6/10	6/10	
mode_reg.incl	pl1	drv	new	6/10	6/10	
page_info.incl	alm	rbs	16-12	6/10	6/13	
ptw.incl	alm	shw	16-10	6/10	6/10	RBS
ptw.incl	pl1	shw	16-10	6/10	6/10	RBS
sdw.incl	alm	shw	16-10	6/10	6/10	RBS
sdw.incl	pl1	shw	16-10	6/10	6/10	RBS
slt.incl	alm	shw	16-10	6/10	6/10	
slt.incl	pl1	shw	16-10	6/10	6/10	
slt_init.incl	alm	shw	16-10	6/10	6/10	NIM
slt_template.incl	alm	shw	16-10	6/10	6/10	NIM
slte.incl	alm	shw	16-10	6/10	6/10	NIM
slte.incl	pl1	shw	16-10	6/10	6/10	NIM
stack_frame.incl	alm	rbs	16-12	6/13	6/13	SHW
tty.incl	pl1	rbs	16-12	6/13	6/13	
util.incl	pl1	drv	16-10	6/10	6/10	

23 segments, 23 assigned, 23 begun, 23 coded, 23 edited, 9 checked.

(END)

BOS DEVELOPMENT

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
apnd	incl.alm	wss	yes	6/13	6/15	DRV,RBS
argnul	incl.alm	drv	yes	yes	6/26	
boot	alm	wss	yes	6/15	6/15	DRV
bos_common	incl.alm	drv	yes	yes	5/26	WSS
bos_data	incl.alm	drv	yes	yes	5/26	WSS
bos_iom_manager	incl.alm	drv	yes	yes	5/26	WSS
bosequ	incl.alm	drv	yes	yes	5/26	WSS
conv	incl.alm	drv	yes	yes	5/26	WSS
cvadd	incl.alm	drv	yes	yes	5/26	
dump	alm	wss	yes	yes	5/30	DRV
equ	incl.alm	drv	yes	yes	5/26	WSS
fill	incl.alm	drv	yes	yes	5/26	WSS
getcon	incl.alm	drv	yes	yes	5/26	WSS
getportinfo	incl.alm	wss	yes	yes	5/26	SHW
intflt	incl.alm	drv	yes	yes	5/26	WSS
iom_error	incl.alm	drv	yes	yes	5/26	WSS
iom_ntape	incl.alm	drv	yes	yes	5/26	
iom_readt	incl.alm	drv	yes	yes	5/26	
iom_rwd170	incl.alm	drv	yes	yes	5/26	
iom_rwd270	incl.alm	drv	yes	yes	5/26	WSS
loaddm	alm	drv	yes	yes	5/26	WSS
nextline	incl.alm	drv	yes	yes	5/26	
patch	alm	drv	yes	yes	5/26	WSS,RBS
proc	incl.alm	wss	yes	5/30	6/15	
restore_machine_state	incl.alm	drv	yes	yes	5/26	
rwsec	incl.alm	drv	yes	yes	5/26	WSS
save_machine_state	incl.alm	drv	yes	yes	5/26	
scan	incl.alm	drv	yes	yes	5/26	WSS
setup	alm	drv	yes	yes	5/26	
util	alm	drv	yes	yes	5/26	

30 segments, 30 assigned, 30 begun, 30 coded, 30 edited, 19 checked.

(END)

TOOLS DEVELOPMENT

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
check_cd	alm					
copy_dump_seg_	pl1					
crossref_	pl1	shw	5/22	5/22	5/22	SHW
deh_test2	alm					
format_dump_line_	alm					
generate_mst	pl1	shw	5/22	5/22	5/22	SHW
gm_error_	pl1	shw	5/22	5/22	5/22	SHW
gm_util1_	pl1	shw	5/22	5/22	5/22	SHW
gm_write_first_seg_	pl1	shw	5/22	5/22	5/22	SHW
get_collection_	pl1	shw	5/23	5/23	5/23	SHW
get_def_ltn_	alm					
get_defptr	alm					
hash_index	alm					
loading_summary_	pl1	shw	5/23	5/23	5/23	SHW
ms_	eplbsa					
online_dump	pl1					
patch_ring_zero	pl1					
print_aste_ptp	pl1					
print_dump_seg_name_	pl1					
ring_zero_dump	pl1					
sslt_manager_	alm	shw	5/23	5/23	5/23	SHW
stack_frame_	eplbsa					
test_faults_	alm					
traffic_control_queue	pl1					

24 segments, 8 assigned, 8 begun, 8 coded, 8 edited, 8 checked.

(END)

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
absadr	alm	drv	5/27	6/2	6/2	
bootstrap1	alm	nim	5/17	5/24	5/24	SHW,RBS
bootstrap2	alm	drv	5/27	6/2	6/2	
clock_	pl1	drv	5/27	6/2	6/2	
clock_init	pl1	drv	5/27	6/2	6/2	
compare_trap	pl1	clj	16-12	6/9	6/15	
delete_segs	pl1	wss	17-0			
device_control	alm	rbs	17-1	yes	6/12	SHW
dn355	pl1	rbs	17-2	yes	6/12	
dn355_util	alm	rbs	17-2	yes	6/12	SHW
emergency_shutdown	alm	drv	5/27	6/2	6/2	RBS
fault_error	alm	shw	16-12	6/13	6/15	
fim	alm	shw	16-12	6/13	6/15	
formline_	alm	rbs	6/15	6/15		
free_store	alm	rbs	16-11	yes	6/12	
freecore	pl1	rbs	16-11	yes	6/12	SHW
get_ptrs_	alm	rbs	16-11	yes	6/12	SHW
ii	alm	nim				
init_collections	pl1	wss	17-0			
init_sst	pl1	drv	16-10	6/13	6/13	RBS
initialize_dims	pl1	drv	16-10	6/13	6/13	RBS
initialize_faults	pl1	drv	16-10	6/13	6/13	RBS
initializer	pl1	wss	17-0			
iom_data_init	pl1	rfm				
iom_manager	alm	rfm				
lot_maintainer	alm	clj	16-12	6/9	6/15	
make_fv_code	alm	drv	16-10	6/13	6/13	
make_sdw	pl1	wss	17-0			
master_mode_init	alm	clj	17-1	6/16		
master_mode_ut	alm	clj	17-1	6/16		
master_pxss_page	alm	shw	17-0	6/13	6/21	
master_vclock	alm	drv	17-0	6/21	6/21	
page_error	alm	rbs	17-1	yes	6/12	SHW
page_fault	alm	rbs	16-11	yes	6/12	SHW
pc	pl1	rbs	16-11	yes	6/12	DRV
pc_trace	alm	rbs	16-11	yes	6/12	SHW
pc_wired	pl1	rbs	16-11	yes	6/12	
pd_util	alm	rbs	16-11	yes	6/12	
pds	alm	shw	16-12	6/13	6/21	
pl1_operators	alm	rbs	16-12			
pl1_operators_	alm					
prds	alm	drv	16-10	6/13	6/13	
pre_link_1	alm	clj	16-12	6/9	6/15	
pre_link_2	alm	clj	16-12	6/9	6/15	
pre_page	alm	rbs	16-11	6/15	6/15	SHW
pxss	alm	shw	16-12	6/13		
scas_init	pl1	drv	5/27	6/2	6/2	RBS
scs	alm	drv	5/27	6/2	6/2	RBS
scs_init	pl1	drv	5/27	6/2	6/2	RBS
segment_loader	pl1	wss	17-0			
shutdown_switch	alm	nim				
signaller	alm	wss	17-0			
slt_manager	alm	nim				
syserr	alm	drv	17-0	6/20	6/21	
sys_info	alm	drv	5/27	6/2	6/2	
terminate_proc	pl1	drv	5/27	6/2	6/2	

```
trace_rsw                 alm       drv       5/27      6/2       6/2
tty_inter                 pl1       rbs       16-12     6/13      6/13      DRV
update_sst_pl1            pl1       clj       17-1      6/20
wire_stack                alm       nim       17-0
wired_fim                 alm       shw       16-12     6/13
wired_hardcore_data_      alm       clj
wired_pim                 pl1       wss       17-0

63 segments, 62 assigned, 57 begun, 48 coded, 43 edited, 13 checked.

(END)
```

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
act_proc	pl1					
activate	pl1	drv	16-12	6/15	6/15	
activate_segs	pl1					
active_all_rings_data	alm					
active_hardcore_data	alm					
add_memory	pl1					
all_rings_util_	alm	drv	6/4	6/4	6/4	RBS
boundfault	pl1	drv	16-12	6/15	6/15	RBS
build_template_dsegs	pl1	drv	16-12	6/15	6/15	
build_template_pds	alm	drv	16-12	6/15	6/15	
condition_	alm					
copy_fdump	pl1					
deact_proc	pl1					
deactivate	pl1	drv	16-12	6/15	6/15	RBS
deactivate_segs	pl1					
dn355_init	pl1	rbs	16-12	6/13	6/13	
find_	pl1					
gate	alm					
get_aste	pl1	drv	16-12	6/13	6/15	RBS
get_defname	alm	clj				
get_proc_id	alm					
imp_gioc	pl1					
imp_status	pl1					
init_branches	pl1					
init_proc	pl1					
init_processor	alm	drv	17-1	6/15	6/15	
init_root_sys_info	pl1	drv	16-12	6/13	6/15	RBS
init_str_seg	pl1					
initalize_kst	pl1	clj				
ioam_util_	pl1					
level	alm					
link_snap	pl1	clj				
lock	pl1					
makestack	pl1					
makeunknown	pl1	clj				
move_device	pl1					
pc_abs	pl1					
plm	pl1					
reconfig	pl1					
reversion_	pl1					
ring_0_cleanup	pl1					
seg_fault	pl1	drv	16-12	6/15	6/15	RBS
setfaults	pl1					
shutdown	pl1					
signal_	pl1					
start_cpu	pl1					
stop_cpu	pl1					
tape_reader	pl1					
tc_init	pl1					
tty_init	pl1	rbs	16-12	6/13	6/13	DRV
unwinder_	pl1					
validate_arg	pl1	clj				

52 segments, 17 assigned, 12 begun, 12 coded, 12 edited, 7 checked.

(END)

COLLECTION 3 DEVELOPMENT

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
cu_	alm					
default_error_handler_	pl1					
find_operator_name_	pl1					
get_por_	pl1					
get_simple_names_	pl1					
get_tpr_	pl1					
init_admin_	alm					
ios_	alm					
linkage_error_	pl1					
listen_	pl1					
look_at_link_	pl1					
special_messages_	pl1					
tape_checksum_	alm					

13 segments, 0 assigned, 0 begun, 0 coded, 0 edited, 0 checked.

(END)

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED
absentee_dim_util_	pl1					
comp_acc_	alm					
cart_io_mmes_	pl1					
cart_non_trapping_mmes_	pl1					
cart_trapping_mmes_	pl1					
dartmouth_mme_handler_	alm					
catmk_util_	eplbsa					
db_alm	alm					
db_assign	pl1					
db_break	pl1					
db_data	alm					
db_reg	pl1					
debug	pl1					
error	fortran					
fiddle_	pl1					
lc_avriables_	pl1					
lc_compute_result_	pl1					
lc_constants_	pl1					
lc_entries_	pl1					
lc_labels_	pl1					
lc_load_	pl1					
lc_store_arg_	pl1					
lc_test_condition_	pl1					
legal_f_	pl1					
legal_frame	pl1					
lisp	pl1					
lisp_	pl1					
lisp_buffer_man_	pl1					
lisp_char_fns_	pl1					
lisp_compiler_	pl1					
lisp_define_	pl1					
lisp_err_	pl1					
lisp_garbage_collector_	pl1					
lisp_get_atom_	pl1					
lisp_io_	pl1					
lisp_lex_	pl1					
lisp_list_utils_	pl1					
lisp_oblist_fns_	pl1					
lisp_operators_	alm					
lisp_prologue_interpreter_	pl1					
lisp_property_fns_	pl1					
lisp_save_	pl1					
lisp_special_fns_	pl1					
lisp_utils_	pl1					
pl1_frame_	pl1					
random_	alm					
round_	eplbsa					
unpack_relbits_	eplbsa					

48 segments, 0 assigned, 0 begun, 0 coded, 0 edited, 0 checked.

(END)

NEW RING STUFF DEVELOPMENT

PROGRAM	LANG	WHO	BEGIN	CODED	EDITED	CHECKED

0 segments, 0 assigned, 0 begun, 0 coded, 0 edited, 0 checked.

(END)