LOC	Page
4 * Zvector E7 instruction tests for VRR-e encoded:  5 * E78C VPERM - Vector Permute  7 *	
2 ** Zvector E7 instruction tests for VRR-e encoded:  5 **  6 ** E78C VPERM - Vector Permute  7 **  8 ** James Wekel March 2025  11 **  12 ** basic instruction tests  13 ** basic instruction tests  14 **  15 ** This program tests proper functioning of the z/arch E7 VRR-e  16 ** This program tests proper functioning of the z/arch E7 VRR-e  17 ** Vector Permute instruction.  18 ** Exceptions are not tested.  20 **  21 ** PLEASE NOTE that the tests are very SIMPLE TESTS designed to cat obvious coding errors. None of the tests are thorough. They are not designed to test all aspects of any of the instructions.  24 **  25 **  26 **  27 ** Testcase zvector-e7-19-VPERM  28 **  29 ** Zvector E7 instruction tests for VRR-e encoded:  30 **  31 ** E78C VPERM - Vector Permute  32 **  33 ** # This tests only the basic function of the instructions.  34 **  35 ** Exceptions are NOT tested.  36 **  37 **  38 ** mainsize 2  39 **  39 **  31 ** Exceptions are NOT tested.  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  39 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **  34 **  35 **  36 **  37 **  38 **  39 **  30 **  30 **  31 **  32 **  33 **	***
## E78C VPERM - Vector Permute    7	
## James Wekel March 2025  ## James Merch 2025  ## James Jam	
12 * 13 * basic instruction tests 14 * 15 * This program tests proper functioning of the z/arch E7 VRR-e 17 * Vector Permute instruction. 18 * 19 * Exceptions are not tested. 20 * 21 * PLEASE NOTE that the tests are very SIMPLE TESTS designed to cat 22 * obvious coding errors. None of the tests are thorough. They are 23 * NOT designed to test all aspects of any of the instructions. 24 * 25 **********************************	***
12 *   13 *   basic instruction tests   14 *   15   16 *   This program tests proper functioning of the z/arch E7 VRR-e   17 *   Vector Permute instruction.   18 *   19 *   Exceptions are not tested.   20 *   21 *   PLEASE NOTE that the tests are very SIMPLE TESTS designed to cat 22 * obvious coding errors. None of the tests are thorough. They are 23 *   NOT designed to test all aspects of any of the instructions.   24 *   25 *   *   *   *   *   *   *   *   *   *	
13 *   basic instruction tests   14 *   15   16 *   16 *   16 *   17 *   16 *   17 *   16 *   17 *   17 *   17 *   18 *   18 *   19 *   18 *   19 *	***
This program tests proper functioning of the z/arch E7 VRR-e  Vector Permute instruction.  Exceptions are not tested.  Exceptions are not tests are very SIMPLE TESTS designed to cat  20 *  11 ** PLEASE NOTE that the tests are very SIMPLE TESTS designed to cat  21 ** PLEASE NOTE that the tests are thorough. They are  22 ** obvious coding errors. None of the tests are thorough. They are  23 ** NOT designed to test all aspects of any of the instructions.  24 **  25 ********************************	
17 * Vector Permute instruction.   18 *	****
19	
21 * PLEASE NOTE that the tests are very SIMPLE TESTS designed to cat obvious coding errors. None of the tests are thorough. They are not obvious coding errors. None of the tests are thorough. They are not obvious coding errors. None of the tests are thorough. They are not obvious coding errors. None of the tests are thorough. They are not obvious coding errors. None of the tests are thorough. They are not obvious coding errors. None of the instructions.  24 *  25 **  27 * *Testcase zvector-e7-19-VPERM  28 *  29 * Zvector E7 instruction tests for VRR-e encoded:  30 * * E78C VPERM - Vector Permute  32 * *  33 * # This tests only the basic function of the instructions.  35 * # Exceptions are NOT tested.  36 * # *  37 * *  38 * mainsize 2  39 * numcpu 1  40 * sysclear  41 * archlvl z/Arch  42 *  43 * loadcore "S(testpath)/zvector-e7-19-VPERM core" 0x0	
24 * 25 **********************************	e <b>h</b> e
27 * *Testcase zvector-e7-19-VPERM 28 * * 29 * Zvector E7 instruction tests for VRR-e encoded: 30 * * 31 * * E78C VPERM - Vector Permute 32 * * 33 * #	***
28 * * Zvector E7 instruction tests for VRR-e encoded:  30 * * * * E78C VPERM - Vector Permute  32 * * * * * * * * * * * * * * * * * * *	
30 * * 31 * * E78C VPERM - Vector Permute 32 * * 33 * * #	
31 * * E78C VPERM - Vector Permute 32 * * 33 * * #	
34 * # This tests only the basic function of the instructions. 35 * # Exceptions are NOT tested. 36 * #	
36 * * #	
38 * mainsize 2 39 * numcpu 1 40 * sysclear 41 * archlvl z/Arch 42 * 43 * loadcore "\$(testpath)/zvector-e7-19-VPERM core" 0x0 44 *	
40 * sysclear 41 * archlvl z/Arch 42 * 43 * loadcore "\$(testpath)/zvector-e7-19-VPERM core" 0x0 44 *	
41 * archlvl z/Arch 42 * 43 * loadcore "\$(testpath)/zvector-e7-19-VPERM core" 0x0 44 *	
$43\ *\ loadcore\ "\$(testpath)/zvector-e7-19-VPERM core"\ 0x0$ $44\ *$	
46 * runtest 5	e)
47 * diag8cmd disable # (reset back to default)	
48 * 49 * *Done 50 *	
51 ************************************	***

SMA Ver.	0. 7. 0 zvector-e7-	19- VPERM			03 Apr 2025 15:	: 39: 45 Page 2
LOC	OBJECT CODE	ADDR1	ADDR2	STMI		
				53 *****	****************	*****
				54 * 55 *	CHECK Macro - Is a Facility Bit set?	
				<b>56</b> *	f the facility bit is NOT set, an message is issue	ed and
				57 * 58 *	the test is skipped.	
				<b>59</b> *	Ccheck uses R0, R1 and R2	
				60 * 61 * eg. 62 ******	CHECK 134, 'vector-packed-decimal'	
				62 ******* 63	**************************************	****
				64	CHECK &BITNO, &NOTSETMSG	
				65 . * 66 . *	&BITNO: facility bit number to cl &NOTSETMSG: 'facility name'	heck
				67 68	CLA &FBBYTE Facility bit in Byte	
				69	CLA &FBBIT Facility bit within Byte	
				70 71 &L(1)	.CLA &L(8) SetA 128, 64, 32, 16, 8, 4, 2, 1 bit positions within by	vte
				72	·	,
				73 &FBBYTI 74 &FBBIT	SETA &BITNO/8 SETA &L((&BITNO-(&FBBYTE*8))+1)	
				75 . * 76	NOTE 0, 'checking Bit=&BITNO: FBBYTE=&FBBYTE, FBBI	Γ=&FBBIT'
				77	X&SYSNDX	
				78 * 79 *	Fcheck data area skip messgae	
				80 SKT&SYS 81	CDC C' Skipping tests: ' OC C&NOTSETMSG	
				82	C C' (bit &BITNO) is not installed.'	
				84 *	TEQU *-SKT&SYSNDX facility bits	
				85 86 FB&SYSI	S FD gap	
				87	S FD gap	
				88 * 89 X&SYSNI	QU *	
				90 91	A RO, ((X&SYSNDX-FB&SYSNDX)/8)-1 TTFLE FB&SYSNDX get facility bits	
				92	0 0	
				93 94	GR RO, RO C RO, FB&SYSNDX+&FBBYTE get fbit byte	
				95	RO, =F' &FBBIT' is bit set?	
				96 97 *		
				98 * faci] 99 *	bit not set, issue message and exit	
				100	A RO, SKL&SYSNDX message length	
				101 102	A R1, SKT&SYSNDX message address AL R2, MSG	
				103 104	в ЕОЈ	
				105 XC&SYS	EQU *	
				106	END	

ASMA Ver.	0. 7. 0 zvector- e7- 1	9- VPERM						03 Apr 2025 15: 39: 45 Page	3
LOC	OBJECT CODE	ADDR1	ADDR2	STMT					
				109	*	Low co	ore PSWs	***************	
00000000		00000000 00000000	00001927	111 112	<b>ZVE7TST</b>	<b>START</b>		Low core addressability	
		00000140	00000000	113 114	SVOLDPSW	EQU	ZVE7TST+X' 140'	z/Arch Supervisor call old PSW	
0000000 000001A0 000001A8	00000001 80000000 00000000 00000200	00000000	000001A0	116 117 118		ORG DC DC	ZVE7TST+X' 1A0' X' 000000018000000 AD(BEGIN)	z/Architecure RESTART PSW	
000001A8	0000000 0000200			110		ЪС	AD(DEGIN)		
000001B0 000001D0 000001D8	00020001 80000000 0000000 0000DEAD	000001B0	000001D0	120 121 122		ORG DC DC	ZVE7TST+X' 1D0' X' 000200018000000 AD(X' DEAD' )	z/Archi tecure PROGRAM CHECK PSW 00'	
000001E0		000001E0	00000200	124		ORG	ZVE7TST+X' 200'	Start of actual test program	
				127	********  * Archit  * Regist  * R0  * R1-4  * R5  * R6-R7	cecture ter Usa (1 (1	The actual "ZVE?  *********  e Mode: z/Arch  age:  work)  work)  esting control tal  work)	**************************************	
				138 139	* R9 * R10	Se Tl	irst base register econd base register hird base register	er	
				140 141 142 143	* R12 * R13	E7 (v	7TEST call return 7TESTS register work) ubroutine call		
				144 145 146	* R15		econdary Subroutii	ne call or work  ***********************************	
00000200 00000200 00000200		00000200 00001200 00002200		148 149 150		<b>USING</b>	BEGIN, R8 BEGIN+4096, R9 BEGIN+8192, R10	FIRST Base Register SECOND Base Register THIRD Base Register	
00000200 00000202 00000204	0580 0680 0680			152 153 154	BEGIN	BALR BCTR BCTR	R8, 0	Initalize FIRST base register Initalize FIRST base register Initalize FIRST base register	
00000206 0000020A	4190 8800 4190 9800		00000800 00000800	156 157 158		LA LA	R9, 2048(, R8) R9, 2048(, R9)	Initalize SECOND base register Initalize SECOND base register	

**NEXTE7** 

ASMA Ver.	0. 7. 0 zvector-e7-1	9- VPERM					03 Apr 2025 15: 39: 45 Page	6
LOC	OBJECT CODE	ADDR1	ADDR2	STMI				
				227 * result	not a	s expected:	**********	
				228 * 229 * 230 ******		and instructio	est number, instruction under test n m4 **************	
0000030A	45F0 812C	0000030A	00000001 0000032C	231 FAILMSG 232	EQU BAL	* R15, RPTERROR		
					ue aft	er a failed tes	**************************************	
0000030E	5800 8284	0000030E	00000001 00000484	237 FAILCONT 238	EQU L	* RO, =F' 1'	set failed test indicator	
00000312 00000316	5000 8E00 41C0 C004		00001000 0000004	239 240 241	ST LA	RO, FAILED R12, 4(0, R12)	next test address	
00000310 0000031A	47F0 80D4		0000004 000002D4	242	B	NEXTE7	next test address	
				245 * end of 246 ******	*****	**************************************	**************************************	
	5810 8E00 1211	0000031E	00000001 00001000	247 ENDTEST 248 249	EQU L LTR	* R1, FAILED R1, R1	did a test fail?	
00000324 00000328	4780 8258 47F0 8270		00000458 00000470	250 251	BZ B	E0J FAI LTEST	No, exit Yes, exit with BAD PSW	

ASMA Ver.	0. 7. 0 zvector- e7- 1	9- VPERM					03 Apr 2025 15: 39: 45 Page 8
LOC	OBJECT CODE	ADDR1	ADDR2	STMF			
				285 ******** 286 * 287 * 288 ******		HERCULES MESSAGE poin R2 = return address	***********  ted to by R1, length in R0  ***********************************
00000390 00000394	4900 8288 07D2		00000488	290 MSG 291	CH BNHR	RO, =H' O' R2	Do we even HAVE a message? No, ignore
00000396	9002 81CC		000003CC	293	STM	RO, R2, MSGSAVE	Save registers
0000039A 0000039E 000003A2	4900 828A 47D0 81A6 4100 005F		0000048A 000003A6 0000005F	295 296 297	CH BNH LA	RO, =AL2(L' MSGMSG) MSGOK RO, L' MSGMSG	Message length within limits? Yes, continue No, set to maximum
000003A6 000003A8 000003AA	1820 0620 4420 81D8		000003D8	299 MSGOK 300 301	LR BCTR EX	R2, R0 R2, 0 R2, MSGMVC	Copy length to work register Minus-1 for execute Copy message to O/P buffer
000003AE 000003B2	4120 200A 4110 81DE		0000000A 000003DE	303 304	LA LA	R2, 1+L' MSGCMD(, R2) R1, MSGCMD	Calculate true command length Point to true command
000003B6 000003BA	83120008 4780 81C6		000003C6	306 307	DC BZ	X' 83' , X' 12' , X' 0008' MSGRET	Issue Hercules Diagnose X'008' Return if successful
000003BE 000003C0	1222 4780 81C6		000003C6	308 309 310	LTR BZ	R2, R2 MSGRET	Is Diag8 Ry (R2) 0? an error occurred but coninue
000003C4	0000			311 312	DC	Н' О'	CRASH for debugging purposes
000003C6 000003CA	9802 81CC 07F2		000003CC	314 MSGRET 315	LM BR	RO, R2, MSGSAVE R2	Restore registers Return to caller
000003CC 000003D8	00000000 00000000 D200 81E7 1000	000003E7	00000000	317 MSGSAVE 318 MSGMVC	DC MVC	3F' 0' MSGMSG(0), 0(R1)	Registers save area Executed instruction
OUUUUSDO	<i>D</i> ≈00 61E7 1000	JUUUUSE/		310 MDGMAC	1414 C	MDUMDU(U), U(RI)	LACCULCU TIISCI UCCI VII
000003DE 000003E7	D4E2C7D5 D6C8405C 40404040 40404040			320 MSGCMD 321 MSGMSG 322	DC DC	C' MSGNOH * ' CL95' '	*** HERCULES MESSAGE COMMAND *** The message text to be displayed

ASMA Ver.	0. 7. 0 zvector- e7- 1	9- VPERM						03 Apr 2025 15: 39: 45 Page	9
LOC	OBJECT CODE	ADDR1	ADDR2	STMT					
								**************************************	
00000448	00020001 80000000			328	<b>EOJPSW</b>	DC	ОD' O' , X' 0002000	018000000', AD(0)	
00000458	B2B2 8248		00000448	330	E0J	LPSWE	EOJPSW	Normal completion	
00000460	00020001 80000000			332	FAI LPSW	DC	0D' 0' , X' 0002000	018000000', AD(X'BAD')	
	B2B2 8260		00000460		<b>FAILTEST</b>		·	Abnormal termination	
00000170			00000100	001		LIGIL			
				336 337 338	****** * *****	****** Worki 1 *****	**************************************	*************	
00000474 00000478				340 341	CTLRO		F F	CRO	
	0000000								
0000047C 0000047C	0000040			343 344		LTORG	, =F' 64'	Literals pool	
00000480	000018EC			345			=A(E7TESTS)		
00000484 00000488	00000001 0000			346 347			=F' 1' =H' 0'		
0000048A	005F			348			=AL2(L'MSGMSG)		
				349 350	*	some o	constants		
		00000400	0000001	351 352	K	EQU	1024	One KB	
		00001000	0000001	353	PAGE	EQU	(4*K)	Size of one page	
		00010000 00100000	00000001 00000001	354 355		EQU EQU	(64*K) (K*K)	64 KB 1 MB	
		AABBCCDD 000000DD	00000001 00000001	356 357	REG2PATT REG2LOW	EQU	X' AABBCCDD' X' DD'	Polluted Register pattern (last byte above)	
								•	

SMA Ver.	0.7.0 zvector-e7-	19- VPERM					03 Apr 2025 15: 39: 45 Page	10
LOC	OBJECT CODE	ADDR1	ADDR2	STMT				
				360 *======			=======================================	
				361 * 362 * NOTE:	start	data on an addres	s that is easy to display	
				363 * 364 *	wi thi n	Hercul es		
				365 *====== 366				
000048C	0000000	0000048C	00001000	367	ORG	ZVE7TST+X' 1000'		
0001000 0001004	0000000 0000000			368 FAILED 369 TESTING		F' 0' F' 0'	some test failed? current test number	
				371 * 372 *	failar	l magaze and acco	sisted editting	
				373 *		l message and asso		
0001018	40404040 40404040 A7A7A7			374 PRTLINE 375 PRTNUM	DC	C' Test # C' xxx'		
000101B 0001033	40868189 93858440 A7A7A7A7 A7A7A7A7			376 377 PRTNAME	DC	c' failed for ins CL8' xxxxxxxx'	truction '	
0001000		00000033	0000001	378 PRTLNG	EQU	*- PRTLINE		

ASMA Ver.	0. 7. 0 zvector-e7-	19- VPERM					03 Apr 2025 15: 39: 45 Page	12
LOC	OBJECT CODE	ADDR1	ADDR2	STM				
				399 *	E7TEST	DSECT	*************	
0000004 00000006 00000007 00000008 00000010 00000014 00000018 0000001C 00000020 00000028 00000030	0000 00			402 E7TEST 403 TSUB 404 TNUM 405 406 407 408 OPNAME 409 V2ADDR 410 V3ADDR 411 V4ADDR 412 RELEN 413 READDR 414 415 V10UTPUT 416	DC D	, A(0) H' 00' X' 00' HL1' 00' CL8' ' A(0) A(0) A(0) A(0) A(0) FD XL16 FD	pointer to test Test Number  m field - not used  E7 name address of v2 source address of v3 source address of v4 source RESULT LENGTH result (expected) address  gap V1 Output gap	
7000040				417 418 * 419 * 420 * 421 *	test ro	outine will be	here (from VRR-e macro)	
000010A8		0000000	00001927	423 ZVE7TST 424	CSECT ,	, 0F		
				426 ******	******* <b>cros to</b> *****	**************************************	**************************************	
				430 * 431 * macro 432 * 433 434	to gener MACRO VRR_E 8	rate individual &INST	l test	
				435 . * 436 . * 437 438 439 &TNUM 440	GBLA 8	&TNUM &TNUM+1	&INST - VRR-e instruction under test no m fields	
				441 442 443 444 T&TNUM 445	DC A	A(X&TNUM) H' &TNUM	base for test data and test routine address of test routine test number	
				446 447 448	DC 1	X' 00' HL1' 00' CL8' &I NST'	m field instruction name	

ASMA Ver.	0. 7. 0 zvector-e7-	19- VPERM					03 Apr 2025 15: 39: 45 Page 13
LOC	OBJECT CODE	ADDR1	ADDR2	STM			
				449 450	DC DC	A(RE&TNUM+16) A(RE&TNUM+32)	address of v2 source address of v3 source
				451 452 453 REA&TN		A(RE&TNUM+48) A(16) A(RE&TNUM)	address of v4 source result length result address
				454 455 V10&TN 456	DS NUM DS DS	FD XL16 FD	gap V1 output gap
				457 . * 458 * 459 X&TNUN		<b>0F</b>	
				460 461 462	LGF VL	R1, V2ADDR v22, O(R1)	load v2 source use v22 to test decoder
				463 464 465	LGF VL	R1, V3ADDR v23, 0(R1)	load v3 source use v23 to test decoder
				466 467 468	LGF VL	R1, V4ADDR v24, 0(R1)	load v3 source use v23 to test decoder
				469 470	&I NST	V22, V22, V23, v24	test instruction (dest is a source)
				471 472	VST BR	V22, V10&TNUM R11	save v1 output return
				473 474 RE&TNU		0F	xl16 expected result
				475 476 477	DROP MEND	R5	
				479 * 480 * macı 481 *	ro to gen	erate table of poi	inters to individual tests
				482 483 484	MACRO PTTAB GBLA		
				485 486 &CUR 487 . *	LCLA SETA	&CUR	
				488 TTABLI 489 . LOOP 490 . *		0F	
				491 492 .*	DC	A(T&CUR)	
				493 &CUR 494 495 *	SETA AI F	&CUR+1 (&CUR LE &TNUM).	L00P
				496 497	DC DC	A(0) A(0)	END OF TABLE
				498 .* 499	MEND	. ,	
				500			

							03 Apr 2025 15: 39: 45 Page	
LOC	OBJECT CODE	ADDR1	ADDR2	STMT				
					*****	******	**********	
				503 *	E7 VRI	R-e tests	**********	
				504 ****** 505	PRINT		**************************************	
				506 *	1 101 10 1	DATA		
				507 * E78	C VPERM	- Vector Permute		
				508 *	VDD -	·		
				509 * 510 *	vkk- e	instruction followed by		
				511 *			ed result (V1)	
				512 *		16 byte V2 sou	rce	
				513 * 514 *		16 byte V3 sou	rce	
					2M - Vec	ctor Permute		
				516 *				
				517	Unn r	VDEDM		
00010A8				518 519+	VRR_E DS	VPERM OFD		
0010A8		000010A8		519+ 520+	USING		base for test data and test routine	
0010A8				521+T1	DC	A(X1)	address of test routine	
0010AC	0001			522+	DC	H' 1'	test number	
0010AE 00010AF	00 00			523+ 524+	DC DC	X' 00' HL1' 00'	m field	
0010H	E5D7C5D9 D4404040			525+	DC	CL8' VPERM	instruction name	
00010B8	00001134			<b>526</b> +	DC	A(RE1+16)	address of v2 source	
00010BC 00010C0	00001144 00001154			527+ 528+	DC DC	A(RE1+32) A(RE1+48)	address of v3 source address of v4 source	
0010C0 00010C4	00001134			529+	DC DC	A(16)	result length	
00010C8	00001124			530+REA1	DC	A(RE1)	result address	
00010D0				531+	DS	FD	gap V1 output	
00010D8 00010E0	00000000 00000000 0000000 00000000			532+V101	DS	XL16	VI output	
0010E0 00010E8	0000000 0000000			533+	DS	FD	gap	
				<b>534</b> +*				
00010F0	E210 5010 0014		00000010	535+X1	DS	OF	110	
00010F0 00010F6	E310 5010 0014 E761 0000 0806		00000010 00000000	536+ 537+	LGF VL	R1, V2ADDR v22, O(R1)	load v2 source use v22 to test decoder	
0010FC	E310 5014 0014		00000014	538+	ĹĠF	R1, V3ADDR	load v3 source	
0001102	E771 0000 0806		00000000	539+	VL	v23, 0(R1)	use v23 to test decoder	
0001108 000110E	E310 5018 0014 E781 0000 0806		00000018 00000000	540+ 541+	LGF VL	R1, V4ADDR v24, O(R1)	load v3 source use v23 to test decoder	
001101	E766 7000 8F8C		0000000	542+		V24, U(N1) V22, V22, V23, v24	test instruction (dest is a source)	
000111A	E760 5030 080E		000010D8	<b>543</b> +	VST	V22, V101	save v1 output	
0001120	07FB			544+	BR	R11	return	
0001124 0001124				545+RE1 546+	DC DROP	OF R5	xl16 expected result	
001124	00000000 00000000			547	DC		0000 00000000000000000000' result	
00112C	00000000 00000000			F 40	D.C.	WI 401 0000000000000000000000000000000000		
0001134 000113C	00000000 00000000 0000000 00000000			548	DC	XL16' 00000000000000	0000 0000000000000000 v2	
)00113C )001144	0000000 0000000			549	DC	XL16' 0000000000000	0000 0000000000000000 v3	
00114C	0000000 00000000							
001154	00000000 00000000			550	DC	XL16' 0000000000000	0000 00000000000000000000 v4	
001170								
00115C	00000000 00000000			551				

ASMA Ver.	0. 7. 0 zv	ector- e7- 19	9- VPERM					03 Apr 2025 15: 39: 45 Page 15
LOC	OBJECT	CODE	ADDR1	ADDR2	STMT			
00001168					553+	DS	OFD	
00001168			00001168		55 <b>4</b> +	USING		base for test data and test routine
00001168	000011B0				555+T2	DC	A(X2)	address of test routine
0000116C	0002				<b>556</b> +	DC	H'2'	test number
0000116E	00				557+	DC	X' 00'	0. 11
0000116F	00 E5D7C5D0	D4404040			558+ 559+	DC	HL1' 00' CL8' VPERM	m field
00001170 00001178	E5D7C5D9 000011F4	D44U4U4U			560+	DC DC	A(RE2+16)	instruction name address of v2 source
00001178 0000117C	00001114				561+	DC	A(RE2+10) A(RE2+32)	address of v2 source
00001180	00001214				562+	DC	A(RE2+48)	address of v4 source
00001184	0000010				<b>563</b> +	DC	A(16)	result length
00001188	000011E4				564+REA2	DC	A(RE2)	result address
00001190	0000000				565+	DS	FD	gap V1 output
00001198 000011A0	00000000				566+V102	DS	XL16	vi output
000011A0	00000000				<b>567</b> +	DS	FD	gap
00001140	0000000	0000000			568+*	DO	10	8 <sub>a</sub> h
000011B0					569+X2	DS	<b>OF</b>	
000011B0	E310 5010			0000010	<b>570</b> +	LGF	R1, V2ADDR	load v2 source
000011B6	E761 0000			00000000	571+	VL	v22, 0(R1)	use v22 to test decoder
000011BC	E310 5014			00000014	572+ 572	LGF	R1, V3ADDR	load v3 source
000011C2 000011C8	E771 0000 E310 5018			0000000 0000018	573+ 574+	VL LGF	v23, 0(R1) R1, V4ADDR	use v23 to test decoder load v3 source
000011C8	E781 0000			00000018	575+	VL	v24, 0(R1)	use v23 to test decoder
00001102 000011D4	E766 7000			0000000	<b>576</b> +		V21, U(1) V22, V22, V23, v24	test instruction (dest is a source)
000011DA	E760 5030			00001198	<b>577</b> +	<b>VST</b>	V22, V102	save v1 output
000011E0	07FB				578+	BR	R11	return
000011E4					579+RE2	DC	OF	xl16 expected result
000011E4 000011E4	FFFFFFF	FFFFFFFF			580+ 581	DROP DC	R5	FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
000011E4 000011EC	FFFFFFF				J01	DC	ALIO FFFFFFFFFF	THE PETERFEFFFFF TESUIC
000011E6	FFFFFFF				582	DC	XL16' FFFFFFFFFFFF	FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
000011FC	FFFFFFF	FFFFFFF						
	FFFFFFF				<b>583</b>	DC	XL16' FFFFFFFFFFF	FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
0000120C					704	DC	VI 401 EFFEPPPPFFFFF	
00001214 0000121C					584	DC	XL16 FFFFFFFFFFF	FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
00001210	rrrrrr	rrrrrr			585			
					586	VRR E	VPERM	
00001228					<b>587</b> +	DS _	OFD	
00001228	00004070		00001228		588+	USING	*, R5	base for test data and test routine
00001228	00001270				589+T3	DC	A(X3)	address of test routine
0000122C 0000122E	0003 00				590+ 591+	DC DC	H' 3' X' 00'	test number
0000122E	00				591+ 592+	DC	HL1' 00'	m field
00001221	E5D7C5D9	D4404040			593+	DC	CL8' VPERM	instruction name
00001238	000012B4				<b>594</b> +	DC	A(RE3+16)	address of v2 source
0000123C	000012C4				595+	DC	A(RE3+32)	address of v3 source
00001240	000012D4				596+	DC	A(RE3+48)	address of v4 source
00001244 00001248	00000010 000012A4				597+ 598+REA3	DC DC	A(16) A(RE3)	result length result address
00001248	000012A4	00000000			599+ 599+	DS DS		
	00000000				600+V103	DS DS	XL16	gap V1 output
00001260	0000000	00000000						
00001268	0000000	0000000			601+	DS	FD	gap
					602+*			

ASMA Ver.	0. 7. 0 zvector-e7-1	9- VPERM					03 Apr 2025	15: 39: 45	Page	16
LOC	OBJECT CODE	ADDR1	ADDR2	STMI						
00001270				603+X3	DS	<b>0F</b>				
	E310 5010 0014		0000010	604+	LGF	R1, V2ADDR	load v2 source			
00001276	E761 0000 0806		00000000	605+	VL	v22, 0(R1)	use v22 to test decoder			
0000127C	E310 5014 0014		00000014	606+	LGF	R1, V3ADDR	load v3 source			
00001270	E771 0000 0806		00000014	607+	VL	v23, 0(R1)	use v23 to test decoder			
00001288	E310 5018 0014		00000000	608+	LGF	R1, V4ADDR	load v3 source			
0000128E	E781 0000 0806		00000000	609+	VL	v24, 0(R1)	use v23 to test decoder			
00001232	E766 7000 8F8C		0000000	610+		V24, V(N1) V22, V22, V23, v24	test instruction (dest	e a courc	<u>a)</u>	
0000129A	E760 5030 080E		00001258	611+	VST	V22, V103	save v1 output	is a sourc	<i>e)</i>	
0000123A	07FB		00001230	612+	BR	R11	return			
000012A0	OTED			613+RE3	DC	OF	xl16 expected result			
000012A4				614+	DROP	R5	Allo expected result			
000012A4	OOFFFFFF FFFFFFF			615	DC		FFFF FFFFFFFFFFFOO'	resul t		
000012A4 000012AC	FFFFFFF FFFFF00			013	DC	ALIO OUTTITITITI	TITE TEFFETEEFFEE	1 esui t		
000012AC	FFFFFFF FFFFFFF			616	DC	VI 16' EEEEEEEEEEE	FFFF FFFFFFFFFFFF	v2		
000012B4 000012BC	FFFFFFF FFFFFFF			010	DC	ALIO FFFFFFFFFF	rrr rrrrrrrrrrr	٧L		
000012BC 000012C4	00000000 00000000			617	DC	VI 16! 000000000000	0000 00000000000000000	v3		
000012C4 000012CC	0000000 0000000			017	DС	XL10 0000000000000		VS		
000012CC 000012D4	10010203 04050607			618	DC	VI 16! 100102020405	0607 08090A0B0C0D0E1F'	<b>v4</b>		
000012D4 000012DC	08090A0B 0C0D0E1F			010	DC	AL10 1001020304030	JOO7 UOUSUAUDUCUDUEIF	V4		
000012DC	USUSUAUD UCUDUEIF			619						
				620	VDD E	VPERM				
00001950										
000012E8		00001950		621+	DS	OFD * DE	has fan tast data and t	taat manti		
000012E8	00001220	000012E8		622+	USING		base for test data and to	test routi	ne	
000012E8	00001330			623+T4	DC	A(X4)	address of test routine			
000012EC	0004			624+	DC	H' 4'	test number			
000012EE	00			625+	DC	X' 00'	C! -1 I			
000012EF	00			626+	DC	HL1' 00'	m field			
000012F0	E5D7C5D9 D4404040			627+	DC	CL8' VPERM	instruction name			
000012F8	00001374			628+	DC	A(RE4+16)	address of v2 source			
000012FC	00001384			629+	DC	A(RE4+32)	address of v3 source			
00001300	00001394			630+	DC	A(RE4+48)	address of v4 source			
00001304	00000010			631+	DC	A(16)	result length			
00001308	00001364			632+REA4	DC	A(RE4)	result address			
00001310	0000000 00000000			633+	DS	FD	gap V1 output			
00001318	0000000 00000000			634+V104	DS	XL16	vi output			
00001320	0000000 00000000			695	DC	ED	dan			
00001328	0000000 00000000			635+ 636+*	DS	FD	gap			
00001330				637+X4	DS	<b>0</b> F				
00001330	E310 5010 0014		0000010	638+	LGF	R1, V2ADDR	load v2 source			
00001336	E761 0000 0806		00000010	639+	VL	v22, 0(R1)	use v22 to test decoder			
00001336 0000133C	E310 5014 0014		00000000	640+	LGF	R1, V3ADDR	load v3 source			
00001330	E771 0000 0806		00000014	641+	VL	v23, 0(R1)	use v23 to test decoder			
00001342	E310 5018 0014		00000000	642+	LGF	R1, V4ADDR	load v3 source			
00001348 0000134E	E781 0000 0806		00000018	643+	VL	v24, 0(R1)	use v23 to test decoder			
0000134E	E766 7000 8F8C		5000000	644+		V24, U(N1) V22, V22, V23, v24	test instruction (dest	is a source	e)	
0000135A	E760 7000 8F8C E760 5030 080E		00001318	645+	VIERNI	V22, V104	save v1 output	is a sourc	~ <i>,</i>	
0000135A 00001360	07FB		0001010	646+	BR	R11	return			
00001364	V.1 D			647+RE4	DC	0F	xl 16 expected result			
00001364				648+	DROP	R5	ario capecteu resurt			
00001364	OOFFFFFF FFFFFOO			649	DC		FFOO OOFFFFFFFFFFFOO'	resul t		
00001304 0000136C	00FFFFFF FFFFFF00			<b>U 1</b> U	<b>D</b> 0	ALIO OUIIIIIIIII		I Cour C		
00001300	FFFFFFF FFFFFFF			650	DC	XI.16' FFFFFFFFFFFF	FFFF FFFFFFFFFFFFF	v2		
	FFFFFFF FFFFFFF				20			.~		
00001376	00000000 00000000			651	DC	XI.16' 0000000000000	0000 00000000000000000	<b>v</b> 3		
00001001				001	20			, <b>U</b>		

ASMA Ver.	0. 7. 0 zved	ctor- e7- 19	)- VPERM					03 Apr 2025	15: 39: 45	Page	17
LOC	OBJECT (	CODE	ADDR1	ADDR2	STMI						
0000138C 00001394 0000139C	00000000 00 10010203 04 18090A0B 00	4050617			652	DC	XL16' 1001020304050	0617 18090A0B0C0D0E1F'	v4		
	18090AUB U	CODUETE			653 654	VRR_E					
000013A8 000013A8 000013A8	000013F0		000013A8		655+ 656+ 657+T5	DS USING DC		base for test data and taddress of test routine	est routi	ne	
000013AC 000013AE 000013AF	0005 00 00				658+ 659+ 660+	DC DC	H' 5' X' 00'	test number m field			
000013B0 000013B8	E5D7C5D9 D4 00001434	4404040			661+ 662+	DC DC	CL8' VPERM A(RE5+16)	instruction name address of v2 source			
000013BC 000013C0 000013C4	00001444 00001454 00000010				663+ 664+ 665+	DC	A(RE5+48)	address of v3 source address of v4 source result length			
000013C8 000013D0	00001424 00000000 00				666+REA5 667+	DC DS	A(RE5)	result address gap V1 output			
000013D8 000013E0 000013E8	00000000 00 00000000 00 00000000 00	0000000			668+V105 669+			gap			
000013F0 000013F0	E310 5010 (	0014		00000010	670+* 671+X5 672+		OF R1, V2ADDR	load v2 source			
000013F6 000013FC	E761 0000 (E310 5014 (	0806 0014		0000000 0000014	673+ 674+	VL LGF	v22, 0(R1) R1, V3ADDR	use v22 to test decoder load v3 source			
00001402 00001408 0000140E	E771 0000 ( E310 5018 ( E781 0000 (	0014		00000000 00000018 00000000	675+ 676+ 677+	LGF	R1, V4ADDR	use v23 to test decoder load v3 source use v23 to test decoder			
00001414 0000141A 00001420	E766 7000 8 E760 5030 0 07FB			000013D8	678+ 679+ 680+	VPERM VST	V22, V22, V23, v24 V22, V105 R11	test instruction (dest i save v1 output return	s a sourc	e)	
00001424 00001424					681+RE5 682+	DC DROP	OF R5	xl16 expected result	_		
00001424 0000142C 00001434	00010203 04 08090A0B 00 00010203 04	CODOEOF			683 684			0607 08090A0B0C0D0E0F' 0607 08090A0B0C0D0E0F'	result v2		
0000143C 00001444	08090A0B 00 FFFFFFFF FI	CODOEOF FFFFFFF			685			FFF FFFFFFFFFFF	v3		
00001454	FFFFFFF FI 00010203 04 08090A0B 00	4050607			686	DC	XL16' 0001020304050	0607 08090A0B0C0D0E0F'	v4		
00001468					687 688 689+		VPERM OFD				
00001468 00001468	000014B0		00001468		690+ 691+T6	USI NG DC	*, R5 A(X6)	base for test data and taddress of test routine	est routi	ne	
0000146C 0000146E 0000146F	0006 00 00				692+ 693+ 694+	DC	X' 00'	m field			
00001470 00001478 0000147C	E5D7C5D9 D4 000014F4 00001504	4404040			695+ 696+ 697+	DC DC	CL8' VPERM	instruction name address of v2 source address of v3 source			
00001480 00001484	00001304 00001514 00000010 000014E4				698+ 699+ 700+REA6	DC DC	A(RE6+48) A(16)	address of v4 source result length result address			
00001400	OUUUITET				/ OO TREAU	DC	A(MLU)	1 CSult auul CSS			

ASMA Ver.	0. 7. 0 zvector- e7- 1	9- VPERM					03 Apr 2025 15: 39: 45 Page	18
LOC	OBJECT CODE	ADDR1	ADDR2	STM				
00001490 00001498 000014A0	00000000 00000000 00000000 00000000 000000			701+ 702+V106	DS DS	FD XL16	gap V1 output	
000014A8	0000000 0000000			703+ 704+*	DS	FD	gap	
000014B0 000014B0 000014B6 000014BC 000014C2	E310 5010 0014 E761 0000 0806 E310 5014 0014 E771 0000 0806		00000010 00000000 00000014 00000000	705+X6 706+ 707+ 708+ 709+	DS LGF VL LGF VL	OF R1, V2ADDR v22, O(R1) R1, V3ADDR v23, O(R1)	load v2 source use v22 to test decoder load v3 source use v23 to test decoder	
000014C8 000014CE 000014D4 000014DA 000014E0	E310 5018 0014 E781 0000 0806 E766 7000 8F8C E760 5030 080E 07FB		0000018 0000000 00001498	710+ 711+ 712+ 713+ 714+	LGF VL VPERM VST BR	R1, V4ADDR v24, O(R1) V22, V22, V23, v24 V22, V106 R11	load v3 source use v23 to test decoder test instruction (dest is a source) save v1 output return	
000014E4 000014E4 000014E4	FFFFFFF FFFFFFF			715+RE6 716+ 717	DC DROP DC	0F R5	xl16 expected result  FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	
000014EC 000014F4	FFFFFFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFF			717	DC DC		FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	
000014FC 00001504 0000150C	FFFFFFF FFFFFFF F0E0D0C0 B0A09080 70605040 30201000			719	DC	XL16' FOEODOCOBOAO	9080 7060504030201000' v3	
00001514 0000151C	00010203 04050607 08090A0B 0C0D0E0F			720 721	DC	XL16' 000102030405	0607 08090A0B0C0D0E0F' v4	
00001528 00001528		00001528		722 723+ 724+	VRR_E DS USING	VPERM OFD * R5	base for test data and test routine	
00001528 0000152C 0000152E	00001570 0007 00	00001020		725+T7 726+ 727+	DC DC DC	A(X7) H' 7' X' 00'	address of test routine test number	
0000152F 00001530 00001538	00 E5D7C5D9 D4404040 000015B4			728+ 729+ 730+	DC DC DC	HL1' 00' CL8' VPERM A(RE7+16)	m field instruction name address of v2 source	
0000153C 00001540 00001544	000015C4 000015D4 00000010			731+ 732+ 733+	DC DC DC	A(RE7+32) A(RE7+48) A(16)	address of v3 source address of v4 source result length	
00001548 00001550 00001558	000015A4 00000000 00000000 00000000 00000000			734+REA7 735+ 736+V107	DC DS DS	A(RE7) FD XL16	result address gap V1 output	
00001560 00001568	00000000 00000000 00000000 00000000			737+ 738+*	DS	FD	gap	
00001570 00001570 00001576 0000157C 00001582 00001588 0000158E 00001594 0000159A	E310 5010 0014 E761 0000 0806 E310 5014 0014 E771 0000 0806 E310 5018 0014 E781 0000 0806 E766 7000 8F8C E760 5030 080E		00000010 00000000 00000014 00000000 0000018 00000000	739+X7 740+ 741+ 742+ 743+ 744+ 745+ 746+ 747+	DS LGF VL LGF VL LGF VL VPERM VST	OF R1, V2ADDR v22, O(R1) R1, V3ADDR v23, O(R1) R1, V4ADDR v24, O(R1) V22, V22, V23, v24 V22, V107	load v2 source use v22 to test decoder load v3 source use v23 to test decoder load v3 source use v23 to test decoder test instruction (dest is a source) save v1 output	
000015A0 000015A4 000015A4	07FB			748+ 749+RE7 750+	BR DC	R11 OF R5	return xl 16 expected result	

T 0.0	OR IFICE CORE	4 D D D 4	ADDDO	CITI III			-		O
LOC	OBJECT CODE	ADDR1	ADDR2	STMT					
0015A4	FFFFFF00 FFFFFF00			751	DC	XL16' FFFFFF00FFFF	FFOO FFFFFFOOFFFFFFFF	resul t	
0015AC	FFFFFF00 FFFFFFFF			770	D.C.	VI 101 EFFEEGAFEE	PERO PEREPONERE PERO	0	
0015B4 0015BC	FFFFFF00 FFFFFF00			752	DC	XL16 FFFFFFUUFFFF	'FF00 FFFFFF00FFFFFFFF'	v2	
0015EC	FFFFFFOO FFFFFFFF FOEODOCO BOA09080			753	DC	VI 16' FOFODOCOROAO	9080 7060504030201000'	v3	
015C4 0015CC	70605040 30201000			733	ьс	ALIO FOLODOCOBOAO	7000304030201000	VJ	
015CC	00010203 04050607			<b>754</b>	DC	XI.16' 000102030405	0607 08090A0B0C0D0E0F'	$\mathbf{v4}$	
0015DC	08090A0B OCODOEOF			701	ЪС	ALIO OUOIO2000100	ooo oo	V -	
				755					
				<b>756</b>		VPERM			
0015E8				757+	DS	OFD		_	
0015E8	00001000	000015E8		758+	USING		base for test data and		9
0015E8	00001630			759+T8	DC	A(X8)	address of test routine		
0015EC	0008			760+	DC	H' 8'	test number		
)015EE )015EF	00 00			761+ 762+	DC DC	X' 00' HL1' 00'	m field		
015EF 0015F0	E5D7C5D9 D4404040			762+ 763+	DC	CL8' VPERM	instruction name		
015F8	00001674			764+	DC	A(RE8+16)	address of v2 source		
015FC	00001684			<b>765</b> +	DC	A(RE8+32)	address of v3 source		
001600	00001694			<b>766</b> +	DC	A(RE8+48)	address of v4 source		
01604	0000010			<b>767</b> +	DC	A(16)	result length		
01608	00001664			768+REA8	DC	<b>A(RE8)</b>	result address		
01610	00000000 00000000			769+	DS	FD	gap V1 output		
01618	00000000 00000000			770+V108	DS	XL16	V1 output		
01620	00000000 00000000			771+	nc	FD	dan		
001628	0000000 00000000			771+ 772+*	DS	ΓV	gap		
001630				772+X8	DS	<b>0F</b>			
001630	E310 5010 0014		00000010	774+	LGF	R1, V2ADDR	load v2 source		
001636	E761 0000 0806		00000000	775+	VL	v22, 0(R1)	use v22 to test decoder		
00163C	E310 5014 0014		00000014	776+	LGF	R1, V3ÀDDR	load v3 source		
001642	E771 0000 0806		0000000	777+	VL	v23, 0(R1)	use v23 to test decoder		
01648	E310 5018 0014		00000018	778+	LGF	R1, V4ADDR	load v3 source		
	E781 0000 0806		0000000	779+	VL	v24, 0(R1)	use v23 to test decoder		
01654	E766 7000 8F8C		00001010	780+	VPERM		test instruction (dest	is a source	
0165A 01660	E760 5030 080E 07FB		00001618	781+ 782+	VST BR	V22, V108 R11	save v1 output return		
01664	U/FB			782+ 783+RE8	DC	OF	xl16 expected result		
01664				784+	DROP	R5	Allo expected result		
01664	11223344 55667788			785	DC		7788 718191A1B0B1B2B3'	resul t	
00166C	718191A1 B0B1B2B3								
001674	11223344 55667788			786	DC	XL16' 112233445566	7788 99AABBCCDDEEFF00'	<b>v2</b>	
00167C				707	D.C.	WI 401 000E4E0004	F404 M4040414P0P4P0P0	0	
001684				787	DC	XL16' 020F1F203141	5161 718191A1B0B1B2B3'	v3	
	718191A1 B0B1B2B3			788	DC	VI 16! 000109090405	0607 19101A1D1C1D1E1E	**/	
01694 0169C	00010203 04050607 18191A1B 1C1D1E1F			100	ъС	ALIU UUUIU&USU4US	0607 18191A1B1C1D1E1F'	<b>v4</b>	
01030	IGIGIALD ICIDIEIF			789					
				790	VRR E	VPERM			
0016A8				791+	DS DS	OFD			
0016A8		000016A8		792+	USING		base for test data and		9
0016A8	000016F0			793+T9	DC	A(X9)	address of test routine		
016AC	0009			794+	DC	H' 9'	test number		
0016AE	00			795+	DC	X' 00'	0. 1.1		
0016AF	00 EFRZCERO RAA0A0A0			796+	DC	HL1' 00'	m field		
0016B0	E5D7C5D9 D4404040			<b>797</b> +	DC	CL8' VPERM	instruction name		

ASMA Ver	0. 7. 0 zvector- e7- 1	9- VPFRM					03 Apr 2025	15: 39: 45	Page	20
nomi ver.	o. r. o zvector er r	O VI LIMI					00 Apr 2020	10.00.10	ruge	~0
LOC	OBJECT CODE	ADDR1	ADDR2	STMT						
					<b>-</b>	. (770				
000016B8	00001734			798+	DC	A(RE9+16)	address of v2 source			
000016BC	00001744			799+	DC	A(RE9+32)	address of v3 source			
000016C0	00001754			800+	DC	A(RE9+48)	address of v4 source			
000016C4	0000010			801+	DC	A(16)	result length			
000016C8	00001724			802+REA9	DC	A(RE9)	result address			
000016D0	00000000 00000000			803+	DS	FD	gap V1 output			
000016D8	00000000 00000000			804+V109	DS	XL16	V1 output			
000016E0	00000000 00000000									
000016E8	00000000 00000000			805+	DS	FD	gap			
				806+*	<b>-</b>					
000016F0				807+X9	DS	0F				
000016F0	E310 5010 0014		00000010	808+		R1, V2ADDR	load v2 source			
000016F6	E761 0000 0806		00000000	809+	VL	v22, 0(R1)	use v22 to test decoder			
000016FC	E310 5014 0014		00000014	810+	LGF	R1, V3ADDR	load v3 source			
00001702	E771 0000 0806		0000000	811+	VL	v23, 0(R1)	use v23 to test decoder			
00001708	E310 5018 0014		00000018	812+		R1, V4ADDR	load v3 source			
0000170E	E781 0000 0806		0000000	813+	VL	v24, 0(R1)	use v23 to test decoder		`	
00001714	E766 7000 8F8C		00001000	814+	VPEKM	V22, V22, V23, v24	test instruction (dest i	s a source	e)	
0000171A	E760 5030 080E		000016D8	815+	VST	V22, V109	save v1 output			
00001720	07FB			816+	BR	R11	return			
00001724				817+RE9	DC	OF	xl16 expected result			
00001724	70605040 20201000			818+ 819	DROP	R5	1000 FFFFFFFFFFFFFF			
00001724 0000172C	70605040 30201000 FFFFFFF FFFFFFFFFFFFFFFFFFFFFFFF			919	DC	AL10 700030403020	1000 FFFFFFFFFFFFF	resul t		
00001720	FFFFFFFF FFFFFFF			820	DC	VI 16' EEEEEEEEEEE	FFFF FFFFFFFFFFFF	v2		
00001734 0000173C	FFFFFFFF FFFFFFF			020	DC	ALIO FFFFFFFFF	TTT TTTTTTTTTTTT	٧L		
00001730	70605040 30201000			821	DC	YI 16! 706050403020	1000 F0E0D0C0B0A09080'	v3		
00001744 0000174C	F0E0D0C0 B0A09080			021	DC	AL10 700030403020	1000 FOEODOCODOA09080	VJ		
00001740	10111213 14151617			822	DC	XI.16' 101112131415	1617 08090A0B0C0D0E0F'	<b>v4</b>		
0000175C	08090A0B OCODOEOF			022	DC	ALIO TOTTILIOTTIO	1017 OGGOGNOBOCOBOLOI	<b>V</b> -		
00001700	000001102 00020201			823						
				824	VRR E	VPERM				
00001768				825+	DS _	OFD				
00001768		00001768		<b>826</b> +	<b>USING</b>	*, <b>R5</b>	base for test data and t	est routi	<b>ne</b>	
00001768	000017B0			827+T10	DC	A(X10)	address of test routine			
0000176C	000A			828+	DC	H'10'	test number			
0000176E	00			829+	DC	X' 00'				
0000176F	00			830+	DC	HL1' 00'	m field			
00001770	E5D7C5D9 D4404040			831+	DC	CL8' VPERM	instruction name			
00001778	000017F4			832+	DC	A(RE10+16)	address of v2 source			
0000177C	00001804			833+	DC	A(RE10+32)	address of v3 source			
00001780	00001814			834+	DC	A(RE10+48)	address of v4 source			
00001784	00000010			835+	DC	A(16)	result length			
00001788	000017E4			836+REA10	DC	A(RE10)	result address			
00001790	00000000 00000000			837+	DS	FD VI 10	gap V1 output			
00001798	00000000 00000000			838+V1010	DS	XL16	vi output			
000017A0	00000000 00000000			920	nc	En	don			
000017A8	00000000 00000000			839+ 840+*	DS	FD	gap			
000017B0				840+** 841+X10	DS	0F				
000017B0	E310 5010 0014		0000010	842+		R1, V2ADDR	load v2 source			
000017B0 000017B6	E761 0000 0806		00000010	843+	VL	v22, 0(R1)	use v22 to test decoder			
000017BC	E310 5014 0014		0000000	844+		R1, V3ADDR	load v3 source			
000017BC	E771 0000 0806		00000014	845+	VL	v23, 0(R1)	use v23 to test decoder			
000017C2	E310 5018 0014		00000000	846+	LGF	R1, V4ADDR	load v3 source			
000017CE	E781 0000 0806		00000010	847+	VL	v24, 0(R1)	use v23 to test decoder			
COUSTION	31 0000 0000		5000000	~ <b>.</b> ,		, 0 (202)				

LOC	OBJECT COD	E ADDR1	ADDR2	STMI				
00017D4	E766 7000 8F8			848+		V22, V22, V23, v24	test instruction (dest is a source)	
0017DA	E760 5030 080	E	00001798	<b>849</b> +	VST	V22, V1010	save v1 output	
0017E0	07FB			850+	BR	R11	return	
0017E4				851+RE10	DC	0F	xl16 expected result	
0017E4				852+	DROP	R5		
0017E4 0017EC	FFFFFFFF FFFF FF0506FF FF07			853	DC	XL16' FFFFFFFFFFF	FFFF FF0506FFFF0706FF' result	
0017EC 0017F4	FFFFFFF FFFF			854	DC	XL16' FFFFFFFFFFF	FFFF FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	
0017FC	FFFFFFF FFFF			077	D.C.	VI 101 0101EE0000EE	10004 EFOCOCHEEOGOCHEI	
001804 00180C	0101FF02 02FF FF0506FF FF07			855	DC	XL16' 0101FF0202FF	0304 FF0506FFFF0706FF' v3	
001814	20212223 2425	2627		856	DC	XL16' 202122232425	2627 38393A3B3C3D3E3F' v4	
00181C	38393A3B 3C3D	3E3F		0.5%				
				857 858	VRR E	VPERM		
001828				859+	DS DS	OFD		
001828		00001828		860+	USING		base for test data and test routine	
001828	00001870			861+T11	DC	A(X11)	address of test routine	
00182C	000B			862+	DC	H' 11'	test number	
00182E	00			863+	DC	X' 00'		
00182F	00			864+	DC	HL1' 00'	m field	
001830	E5D7C5D9 D440	4040		<b>865</b> +	DC	CL8' VPERM	instruction name	
001838	000018B4			<b>866</b> +	DC	A(RE11+16)	address of v2 source	
00183C	000018C4			867+	DC	A(RE11+32)	address of v3 source	
001840	000018D4			<b>868</b> +	DC	A(RE11+48)	address of v4 source	
001844	00000010			869+	DC	A(16)	result length	
001848	000018A4			870+REA11	DC	A(RE11)	result address	
001850	00000000 0000			871+	DS	FD	gap	
001858	00000000 0000			872+V1011	DS	XL16	V1 output	
001860	00000000 0000			070	D.C.	TIP.		
001868	00000000 0000	0000		873+ 874+*	DS	FD	gap	
001870				875+X11	DS	0F		
001870	E310 5010 001	4	00000010	<b>876</b> +	LGF	R1, V2ADDR	load v2 source	
001876	E761 0000 080		00000000	<b>877</b> +	VL	v22, 0(R1)	use v22 to test decoder	
00187C	E310 5014 001	4	00000014	878+	LGF	R1, V3ADDR	load v3 source	
001882	E771 0000 080	6	00000000	879+	VL	v23, 0(R1)	use v23 to test decoder	
001888	E310 5018 001		0000018	880+	LGF	R1, V4ADDR	load v3 source	
00188E	E781 0000 080		0000000	881+	VL	v24, 0(R1)	use v23 to test decoder	
001894	E766 7000 8F8		000015==	882+		V22, V22, V23, v24	test instruction (dest is a source)	
00189A	E760 5030 080	E	00001858	883+	VST	V22, V1011	save v1 output	
0018A0	07FB			884+	BR	R11	return	
0018A4 0018A4				885+RE11 886+	DC DROP	OF R5	xl16 expected result	
0018A4	01FFF9FF FFF8	03FF		887	DC		03FF FF0506FFFF070600' result	
0018AC	FF0506FF FF07	0600						
0018B4	FFFFFFF FFFF			888	DC	XL16' FFFFFFFFFF	'FFFF FFFFFFFFFFFFFFF v2	
0018BC	FFFFFFF FFFF			000	D.C.	W 1010101010000	AAAA EEAKAAEEEEAKAAAA	
0018C4	0101F902 02F8			889	DC	XL16 0101F90202F8	0304 FF0506FFFF070600' v3	
0018CC	FF0506FF FF07			000	D.C	WI 101 10001000011	1007 00000400000000000000000000000000000	
0018D4 0018DC	10021203 0415 38393A3B 3C3D			890	DC	XL16' 100212030415	1607 38393A3B3C3D3E3F' v4	
MIODE	JOJJJAJU JUJU	JEJF		891				
				892				
				893				
	0000000			894	DC	F'O' END OF T	ART T	

WII VCI.	0. 7. 0 zvector- e7	- IJ- VI LIMI						US A	or 2025 1	J. JJ. 1J	1 age	24
LOC	OBJECT CODE	ADDR1	ADDR2	STMI								
		00000016	00000001	967 V22	EQU	22						
		00000017 00000018	00000001 00000001	968 V23 969 V24	EQU EQU	23 24						
		00000019 0000001A	00000001 00000001	970 V25 971 V26	EQU EQU	25 26						
		0000001B	00000001 00000001	972 V27	EQU	27						
		0000001C 0000001D	00000001	973 V28 974 V29	EQU	29						
		0000001E 0000001F	00000001 00000001	975 V30 976 V31 977	EQU EQU EQU EQU EQU EQU EQU EQU	22 23 24 25 26 27 28 29 30 31						
				978	END							

SYMBOL   TYPE   VALUE   LENCTH   DEFN   REFERENCE   STATE	ASMA Ver. 0.7.0	zvector	- <b>e7- 19- VPER</b> I	M											03	Apr	2025	15: 39	: 45	Page	25
TILRO F 00000474	SYMB0L	TYPE	VALUE	LENGTH	DEFN	REFE	RENCE	S													
TILRO F 00000474	RECT N	т	00000200	9	159	110	1/18	1/0	150												
DECNIM   C																					
TTESTS		C						104	103												
TTESTS F 000018EC 4 889 203 DIT X 0000103B 18 833 282 DITST U 0000031E 1 247 208 DITST U 00000031E 1 247 208 DITST U 00000000 1 258 208 DITST U 000000000 1 258 208 DITST U 00000000		4					203														
## STATES   1		E T		12																	
SECOND		Y		18																	
301		II		10																	
Supersyman		ĭ		4			250														
AILCONT   U 00000306		Ď		8			200														
FILED F 00001000 4 368 239 248   FAILES   FAILES		Ĭ		1		000															
FAILINGS   U   0000030A		F		4		239	248														
ALLESW D 00000400 8 332 334		Î		1			~ 10														
ALLIEST I 00000470		Ď		8																	
BROOL		Ť		4																	
MAGE   1   00000000   6440   0   1   355   1   354   1   354   355   1   354   355   1   354   355   355   354   355   3		F		8			185	187													
Mathematical Color   Mathema		1		6440		-01	-00	-0,													
Mathematical Color   Mathema		Ū		1		353	354	355													
B		Ŭ		i i		200	JU 1	550													
NSG I 00000390		Ŭ		î																	
SGCMD   C   000003BE		Ĭ		4		195	273														
SGMSG   C   000003E7   95   321   297   318   295   318   295   318   295   318		Ċ		9																	
SGMC		Č		95				295													
SGOK		Ĭ																			
SGRET	<b>I</b> SGOK	Ī		2																	
F   000003CC		Ι		4			310														
NPAME   C   00000008		F		4																	
PACE U 00001000 1 353 PRITS C 00001051 18 386 262 263 264 PRITLINE C 00001008 16 374 378 272 PRITLING U 00000033 1 378 271 PRITNAME C 00001018 3 375 264 PRINUM C 00001018 3 375 264 PRINUM C 00001018 3 375 264 PRINUM C 00000000 1 924 112 162 165 184 186 187 188 193 212 213 238 239 270 271 274 PRINUM C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 FINAME C 00000000 1 937 574 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 FINAME C 00000000 1 937 1937 1938 1938 1938 1938 1938 1938 1938 1938	IEXTE7	U	000002D4	1	205	224	242														
PRTS	PNAME	C	8000000	8	408	266															
PRILINE C 00001008 16 374 378 272  PRILING U 00000033 1 378 271  PRILING U 00000033 1 378 271  PRILING C 00001033 8 377 266  PRINUM C 00001018 3 375 264  RO U 00000000 1 924 112 162 165 184 186 187 188 193 212 213 238 239 270 271 274  R1 U 00000001 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 571 572 573 574 575 604 605 606 607 608 609 638 639 640 641 642 643 672 673 674 675 676 677 706 707 708 709 710 711 740 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 U 00000000 1 937 812 U 00000000 1 937 812 U 00000000 1 937 813 U 00000000 1 938 815 U 00000000 1 938		U	00001000	1	353																
PRING U 00000033 1 378 271  PRINAME C 00001033 8 377 266  RO U 00000000 1 924 112 162 165 184 186 187 188 193 212 213 238 239 270 271 274  R1 U 00000001 1 925 194 219 220 287 299 314  R1 U 00000001 1 925 194 219 220 288 249 272 304 318 536 537 538 539 540 541 570	PRT3	C	00001051	18	386	262	263	264													
PRINAME C 00001033 8 377 266 PRINUM C 00001018 3 375 264  RO U 00000000 1 924 112 162 165 184 186 187 188 193 212 213 238 239 270 271 274  R1 U 00000001 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570		C	00001008	16	374	378	272														
RTNUM C 00001018 3 375 264   0 00000000 1 924 112 162 165 184 186 187 188 193 212 213 238 239 270 271 274   290 293 295 297 299 314   0 00000001 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570   571 572 573 574 575 604 605 606 607 608 609 638 639 640 641   642 643 672 673 674 675 676 677 706 707 708 709 710 711 740   741 742 743 744 745 774 775 776 777 778 779 808 809 810 811   810 U 0000000A 1 934 150 159 160   811 U 0000000B 1 935 216 217 544 578 612 646 680 714 748 782 816 850 884   812 U 0000000C 1 936 203 206 223 241   813 U 0000000E 1 938   814 U 0000000E 1 938   815 U 00000002 1 936 232 257 277 278   817 U 00000002 1 936 232 257 277 278   818 U 00000002 1 926 195 260 261 270 273 274 291 293 299 300 301 303 309 314 315   818 U 00000004 1 928		U	00000033	1																	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	PRTNAME	C		8	377	<b>266</b>															
R1 U 00000001 1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 571 572 573 574 575 604 605 606 607 608 609 638 639 640 641 642 643 672 673 674 675 676 677 706 707 708 709 710 711 740 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 U 000000000 1 935 216 217 544 578 612 646 680 714 748 782 816 850 884 813 814 U 00000000 1 937 814 U 00000000 1 938 814 815 U 00000000 1 938 814 815 U 00000000 1 938 815 U				3																	
1 925 194 219 220 248 249 272 304 318 536 537 538 539 540 541 570 571 572 573 574 575 604 605 606 607 608 609 638 639 640 641 642 643 672 673 674 675 676 677 706 707 708 709 710 711 740 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 U 0000000000000000000000000	<b>(0</b>	U	0000000	1	924							188	193	212	213	238	239	270	271	274	
571 572 573 574 575 604 605 606 607 608 609 638 639 640 641 642 643 672 673 674 675 676 677 706 707 708 709 710 711 740 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 812 813 812 813 812 813 812 813 812 813 842 843 844 845 846 847 876 877 878 879 880 881 811 812 813 81																					
642 643 672 673 674 675 676 677 706 707 708 709 710 711 740 741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 810 U 0000000A 1 934 150 159 160 811 U 0000000B 1 935 216 217 544 578 612 646 680 714 748 782 816 850 884 812 U 0000000C 1 936 203 206 223 241 813 U 0000000D 1 937 814 U 0000000E 1 938 815 U 0000000F 1 939 232 257 277 278 816 817 818 818 818 818 818 818 818 818 818 818	<b>l</b> 1	U	0000001	1	925							<b>304</b>									
741 742 743 744 745 774 775 776 777 778 779 808 809 810 811 812 813 842 843 844 845 846 847 876 877 878 879 880 881 810 U 0000000000000000000000000																					
812 813 842 843 844 845 846 847 876 877 878 879 880 881 810 U 0000000A																					
R10       U       00000000A       1       934       150       159       160         R11       U       0000000B       1       935       216       217       544       578       612       646       680       714       748       782       816       850       884         R12       U       0000000C       1       936       203       206       223       241         R13       U       0000000D       1       937       938																				811	
R11       U       00000000B       1       935       216       217       544       578       612       646       680       714       748       782       816       850       884         R12       U       0000000C       1       936       203       206       223       241         R13       U       0000000D       1       937       938       938       938       938       938       939       232       257       277       278       939       232       257       277       278       938       939       300       301       303       309       314       315         R2       U       00000002       1       926       195       260       261       270       273       274       291       293       299       300       301       303       309       314       315         R3       U       000000003       1       927         R4       U       000000004       1       928	110		0000000	_					843	844	845	846	847	876	877	878	879	880	881		
12     U     0000000C     1     936     203     206     223     241       213     U     0000000D     1     937       214     U     0000000E     1     938       215     U     0000000F     1     939     232     257     277     278       22     U     00000002     1     926     195     260     261     270     273     274     291     293     299     300     301     303     309     314     315       23     U     00000003     1     927       24     U     00000004     1     928				1						0.1-	0.1-	0.00						00:			
U 000000D 1 937 214 U 000000E 1 938 215 U 000000F 1 939 232 257 277 278 22 U 00000002 1 926 195 260 261 270 273 274 291 293 299 300 301 303 309 314 315 23 U 0000003 1 927 24 U 0000004 1 928				1						612	646	680	714	748	<b>782</b>	816	850	884			
U     0000000E     1     938       U     000000F     1     939     232     257     277     278       U     00000002     1     926     195     260     261     270     273     274     291     293     299     300     301     303     309     314     315       U     00000003     1     927       U     00000004     1     928		Ü		1		203	206	223	241												
U 0000000F 1 939 232 257 277 278 U 00000002 1 926 195 260 261 270 273 274 291 293 299 300 301 303 309 314 315 U 00000003 1 927 U 00000004 1 928		Ü		1																	
U 00000002 1 926 195 260 261 270 273 274 291 293 299 300 301 303 309 314 315 U 00000003 1 927 U 00000004 1 928		U		1		000	0-~	0~~	070												
U 00000003 1 927 U 00000004 1 928				1						070	071	004	000	000	000	004	000	000	014	015	
U 00000004 1 928	2			1		195	260	261	270	273	274	291	293	299	300	301	303	309	314	315	
				1																	
55				1		000	00~	040	050	050	F.0.0	- 40	4		F.0.0	011	000	0.40	050	000	
	15	U	00000005	1	929													648	656	682	
690 716 724 750 758 784 792 818 826 852 860 886	10	**	0000000		000	690	716	724	750	758	784	792	818	826	852	860	886				
86 U 00000006 1 930				1																	
7 U 00000007 1 931				1		1.40	150	150	1 = 4	150											
8 U 00000008 1 932 148 152 153 154 156				1						156											
9 U 00000009 1 933 149 156 157 159	Y Y	U	00000009	1	933	149	156	157	159												

SYMBOL	TYPE	VALUE	LENGTH	DEFN	DEFF	RENCE	C					
SIMBUL	IIFE	VALUE	LENGIN	DEFN	KEFE	KENCE	. <b>.</b>					
E1	F	00001124	4	545	<b>526</b>	<b>527</b>	<b>528</b>	<b>530</b>				
E10	F	000017E4	4	851	832	833	834	836				
E11	F	000018A4	4	885	866	867	868	870				
<b>E2</b>	F	000011E4	4	579	<b>560</b>	<b>561</b>	<b>562</b>	<b>564</b>				
E3	F	000012A4	$ar{f 4}$	613	<b>594</b>	<b>595</b>	<b>596</b>	598				
E4	F	00001364	$\overline{4}$	647	628	629	630	632				
E5	F	00001424	$ar{4}$	681	662	663	664	666				
E6	F	00001121 000014E4	$\dot{4}$	715	696	697	698	700				
E7	F	000011L1	4	749	730	731	732	734				
EE8	F	000015A4 00001664	4	<b>783</b>	<b>764</b>	<b>765</b>	766	76 <b>8</b>				
ES	F	00001004	4	817	704 798	703 799	800	802				
	_			530	790	799	OUU	0U2				
EA1	A	00001008	4									
EA10	A	00001788	4	836								
EA11	A	000011848	4	870								
EA2	A	00001188	4	564								
EA3	A	00001248	4	598								
EA4	A	00001308	4	632								
EEA5	A	000013C8	4	666								
EEA6	A	00001488	4	700								
EEA7	A	00001548	4	734								
EA8	A	00001608	4	768								
EEA9	A	000016C8	4	802								
READDR	A	00000020	4	413	219							
EG2LOW	U	00000DD	1	358								
EG2PATT	U	AABBCCDD	1	357								
ELEN	A	000001C	4	412								
PTDWSAV	D	00000380	8	283	270	274						
PTERROR	Ī	0000032C	4	257	232							
PTSAVE	$\overline{\mathbf{F}}$	00000374	$ar{f 4}$	280	257	277						
PTSVR5	F	00000378	$\overline{4}$	281	258	276						
5KL0001	Ū	00000016 0000004E	ī	177	193	~						
SKT0001	č	0000001E	20		177	194						
SVOLDPSW	Ü	00000224	0		1//	134						
11	A	0000140 000010A8	4	521	902							
10	A	000010A8 00001768	4	827	911							
110			4	861	912							
	A	00001168	4									
2	A	00001168	4	555	903							
3	A	00001228	4	589	904							
<b>[4</b>	A	000012E8	4	623	905							
<b>15</b>	A	000013A8	4	657	906							
<u> </u>	A	00001468	4	691	907							
7	A	00001528	4	725	908							
8	A	000015E8	4	759	909							
<b>'9</b>	A	000016A8	4	793	910							
'ESTI NG	F	00001004	4	369	213							
'NUM	H	0000004	2	404	212	<b>260</b>						
SUB	A	00000000	4	403	216							
TABLE	F	000018EC	4	901								
<b>'0</b>	U	00000000	1	945								
<b>1</b>	Ū	00000001	- 1	946	215							
10	ĬĬ	0000000A	1	955								
11	ĬĬ	0000000H	1	95 <b>6</b>								
112	II	0000000B	1	957								
13	II	0000000C	1	958								
714	Ü	000000D	1	959								
15	U	000000E	1	959 960								

SYMB0L	TYPE	VALUE	LENGTH	DEFN	REFER	ENCES														
			EENGTH		KLI LK	LITOLD														
16	U	00000010	1	961																
17 18	U U	00000011 00000012	1 1	962 963																
19	Ŭ	00000012	1	964																
1FUDGE	X	00001088	16	395	215															
101	X	000010D8	16	532	543															
1010	X	00001798	16	838	849															
1011	X	00001858	16	872	883															
(102 (103	X X	00001198 00001258	16 16	566 600	577 611															
103 104	X	00001238	16	634	645															
105	X	00001318 000013D8	16	668	679															
106	X	00001498	16	702	713															
107	X	00001558	16	736	747															
108	X	00001618	16	770	<b>781</b>															
109	X	000016D8	16	804	815															
10UTPUT	X	00000030	16	415	220															
[2 [20	U U	00000002 00000014	1	947 965																
21	Ŭ	00000014	1	966																
22	Ŭ	00000016	1	967	537	542	543	571	576	577	605	610	611	639	644	645	673	678	679	
					707	712		741	<b>746</b>	747	775	<b>780</b>	781	809	814	815	843	848	849	
			_				883													
23	U	0000017	1	968				576	607	610	641	644	675	678	709	712	<b>743</b>	<b>746</b>	777	
0.4	<b>T</b> T	00000010	4	000				845	848	879	882	CAA	077	070	711	710	745	740	770	
24	U	0000018	1	969				576 847	609 848	610 881	643 882	044	677	0/8	/11	712	745	740	779	
25	U	00000019	1	970	700	013	014	047	040	001	002									
26	Ŭ	00000018	î	971																
27	Ü	0000001B	<u>-</u>	972																
28	U	000001C	1	973																
29	Ų	0000001D	1	974					0=0	~~~	~ 40		000	0.40	0=0					
2ADDR	A	00000010	4	409	536	<b>570</b>	604	638	672	706	740	774	808	842	876					
73 730	U	00000003 0000001E	1 1	948 975																
31	II	0000001E	1 1	976																
3ADDR	Ä	00000011	4	410	<b>538</b>	572	606	640	674	708	742	776	810	844	878					
4	Ü	00000004	ĺ	949	000	0.2		0.10	0.1				010	011	0.0					
4ADDR	A	0000018	4	411	<b>540</b>	<b>574</b>	608	642	676	710	744	778	812	846	880					
5	U	00000005	1	950																
6	U	00000006	1	951																
7	U	00000007	<u> </u>	952																
(8 (9	II	00000008 00000009	1	953 954																
0001	Ü	0000003 000002A8	1	183	171	184														
1	F	000010F0	$\overline{4}$	535	521	-0-														
10	F	000017B0	4	841	827															
11	<u>F</u>	00001870	4	875	861															
2	F	000011B0	4	569	555															
3	F	00001270	4	603	589															
<b>4</b> 5	r E	00001330 000013F0	4	637 671	623 657															
.5 [6	F	000013F0 000014B0	4	705	691															
7	F	00001400	4	739	725															
8	F	00001670	4	773	759															
9	F	000016F0	$\overline{4}$	807	793															

wa ver. U. 7. U	zvector	`- e7- 19- VPER	M								US	Apr 2023	15: 39: 45	rage	28
SYMBOL	TYPE	VALUE	LENGTH	DEFN	REFE	RENCE	ES								
0001 E7TST (E7TESTS) L2(L'MSGMSG) '1'	U J A R	000002D0 00000000 00000480 0000048A	1 6440 4 2	345 348	114 203 295	116	120	124	367	112					
' 1' ' 64' ' 0'	F F H	00000484 0000047C 00000488	4 4 2	344	238 188 290										



