

Eighteen Bit Computers

INTRODUCTION

This article was inspired by communications from three 'outside-of-Minnesota' people who discovered our IT Legacy web site via their internet browsers:

- From George Keremedjiev; Director of the <u>American Computer Museum</u> in Bozeman, MT: "We have the last remaining mainframe computer used by NASA for the Apollo Moon Missions, a UNIVAC 418-II used at Mission Control in Houston. We will be opening a new comprehensive exhibit on computers and the Apollo Moon Missions on June 1, 2015."
- From Duane Craps: "The Mid Atlantic Retro Computer Hobbyists (MARCH) are looking for info on 1219's, 1532's, and 1540's that they rescued from Johns Hopkins APL."
- From Anders Carlsson, Sweden: "Hello. I am an old service engineer from Univac to Unisys. I am trying to find out if there are any more physical units (418-II) in the world. Our simple website has some pictures from our collection, https://datamuseet.wordpress.com."

The VIP Club, here in Minnesota, appreciates the efforts of these people and their organizations in preserving parts of our Information Technology (IT) Legacy. This paper includes communications clips from these people and retirees who have contributed to our web site anthology. *LABenson*



This NASA installation¹ has (left to right) a UNIVAC 1232 I/O Console, two UNIVAC 1218 computers, two UNIVAC dual transport 1240 magnetic tape units, and a UNIVAC 418 computer with operator keyboard/printer. The elapsed time mission clock above the 418 operator/maintenance console shows 14:42.83 hours/minutes.

¹From the Dakota County Historical Society's Lawshe Memorial Museum collection, scanned by Keith Myhre.



Contents

INTRODUCTION	1
Control Unit Tester (CUT) – the 18-bit Beginning	2
American Computer Museum in Montana	4
Mid Atlantic Retro Computer Hobbyist on the East Coast	6
A Swedish UNIVAC Computer Museum	7
NASA installation clips and notes	10
Appendix A - 1218 Bootstrap to load from a paper tape	12
Appendix B – 1968 UNIVAC Computer Genealogy Chart	15
Appendix C – SHIPBOARD UNIFORM AUTOMATED DATA PROCESSING SYSTEM	16

Control Unit Tester (CUT) – the 18-bit Beginning

This extract from a May 1968 computer genealogy chart shows the CUT in red at the left.



Harry Wise, an early UNIVAC employee and IT Legacy Committee history source wrote:

- The Control Unit Tester (CUT) was a spin-off of the Univac 490; the first of a series of 18-bit machines in the 510/580/418/1218 line. The CUT was just what its name implies – it was intended for testing peripheral control units in manufacturing for the Univac 490 and the Univac 1107 peripheral equipments, i.e. tape units, printers, drum memories, etc.
- 2. The 18-bit line really got its start when Univac sold the 18-bit architecture to Westinghouse for them to brand as the Westinghouse Prodac 510 and Prodac 580. They were the same machine with fewer options in the 510. Many of the cards in the Prodac machines were labeled "418".
- 3. The Univac 418 was the commercial 18-bit machine that had a hard time making it through the Blue Bell, PA management. Blue Bell management did not want to be in the "mini-computer" business, in spite of the fact that the 418 was more powerful than the Blue Bell



г

UNIVAC II and UNIVAC III machines. At this time, DEC was just getting started in the mini computer business. The 418 was also a much better machine then the early DEC processors.

4. The Control Unit Tester, the 1218, and the 418 were all inter-twined with the same basic 15pin printed circuit cards. This was the second time that Univac had both a defense and commercial version of the same machine. Only the reverse this time, from commercial to military. {Editor's notes: The first was in 1961 when the 642A processor was repackaged as the UNIVAC 490.}

Genealogy Item	Mostly extracted from our web site, <u>http://vipclubmn.org</u> .
Control Unit Tester	From Ernie Lantto: The CUT dimensions were 44.5" x 46" x 23.5"
1st in 1962	totaling 27.8 cu ft. It weighed ~800 lbs and consumed ~1,000 watts. It
	had eight 18-bit channels (4 at 36 bits) and 8k words of 8-microsecond
	core memory.
S510/80	The Process Control computers for Westinghouse; see Harry Wise's
1st in 1962	comment 2 just below the genealogy chart part on the previous page.
1218 CP-789	Initially used in the NTDS system for the 'Identification Friend or Foe'.
1st in 1963	This demonstrated transponder communications led to the FAA
326 built	terminal control applications. The 1218 dimensions were 72" x 26" x
	28.5" totaling 30.8 cu ft. It weighed 866 lbs and consumed 937 watts. It
	had eight 18-bit channels (4 at 36 bits) and 16k words of 4-microsecond
	core memory.
	Based on the Univac 418 design, the 1218 came with COBOL, very
	necessary for the software development.
418 I, 1st in 1963	The first units came out in 1963.
1224 , first in 1963	This entry is a chart lineage error; the 1224 was a 24-bit ISA machine
	initially using PC cards from the 642A as the 1218 had. This machine had
	applications for the Air Force's intelligence FLEXCOP operations.
CP-818, in 1965	A second lot received the military nomenclature
NTDS Compatible 1218	NTDS compatible was not a new nor a different design, rather a paper
First in 1964	title for the units that were delivered to NASA for tracking in rugged
	environments or on board ship. This unit type was also used at JPL for
	telemetry processing.
	This unit type was also for the FAA's Automated Terminal Radar Systems
	(ARTS I) in Atlanta Georgia. The processors were used for tracking
	aircraft and aircraft transponder communications.
M 1218	The M1218 version was sold to the USMC for their 'Landing Approach
	System-3', the first delivery was Oct. 23, 1965.
M-555	M-555 was in 1967 for the MASU project
1219, CP-848	The 1219 dimensions were 72" x 26" x 28.5" totaling 30.8 cu ft. It
First in 1965	weighed 968 lbs and consumed 1478 watts. It had sixteen 18-bit
367 built	channels (8 at 36 bits) and 32k words of 2-microsecond core memory.
	This was the 2-bay configuration. There was a 3-bay configuration with
	65k words.
418 II	First units in 1965, note that the 418 III came out in 1969

Notes about the items (boxes) on the genealogy chart are as follows:



IT Legacy Article for the Month

Genealogy Item	Mostly extracted from our web site, <u>http://vipclubmn.org</u> .
NTDS Compatible 1219	Used for FAA's ARTS IA system as both the tracking and display driver
First in 1965	system processors.
Moon Beam	The SMIS 1500 became the core of the Navy's on board Management
First in 1965	Information System to keep track of all the logistics such as
	ammunition, foodstuffs, etc. A variation with more memory was
SMIS 1500	designated the AN/UYK-5. This system used for inventory control was
	developed for the AFS-1 through -7, the MARS Class Combat Stores
	Ships. See Appendix C for an expanded definition of the system. Some
	Navy user experiences are at
	http://www.navydp.com/phpBB3/viewtopic.php?f=5&t=210
1818 ILAAS	This was an airborne version of the 1219 using mechanical technologies
First in 1967	from the CP-823 30-bit computer. The characteristics were a 2-
Qty built = 2.	microsecond memory in 4k to 32k words. This unit had only 27
	instructions with 3 index registers mapped onto core addresses. It also
	had just 9 interrupts and 1 Real time clock. It had an Assembler and
	Utility Package. Project start was March 1966, first delivery in May 1967.
1219B	This became the base computer for TARTAR, TALOS, and TERRIER
First in 1968	missile launching from ship installations.
	Digital TALOS Shipboard Missile Fire Control System - Univac utilized a
	smaller word length digital computer. We developed and wrote the
	software to digitize the former analog fire control computer that
	controlled the TALOS shipboard missile system. The system was
	implemented on several Navy cruisers as the MK-152 system.
1819	The 1819 was an airborne version of the 1219 using mechanical
First in 1968, 2	technologies from the CP-823 30 bit computer. They were delivered to
delivered	Sperry on Long Island with the nomenclature CP-914.

Not on the 1968 genealogy chart; in 1969 UNIVAC began development of type U1825, AN/UYK-11. We subsequently built and delivered 130 units to the US Air Force. This computer design mapped the 18-bit ISA onto a 36-bit physical structure using plated wire memory. These units had extremely high reliability, averaging less than a single failure over measured 5-year periods while operating 24/7 as the Minuteman launch & mission control computers.

I do not have an exact count, but there were about 2,000 units with this 18-bit ISA. LABenson

American Computer Museum in Montana

Help sought – {Editor's note, this was distributed to VIP Club members via a newsletter}. The American Computer & Robotics Museum in Bozeman, MT has just acquired the UNIVAC 418-II that was used in Houston during the Apollo Moon Missions at the Mission Control Center. It was implemented along with the three UNIVAC 493 computers for the reception of the telemetry data prior to it being passed on to the IBM 360s on the other side of the room. If any of you has any direct/indirect information about the specific duties of the NASA UNIVAC 418-II, please send a note to labenson@q.com."



<u>Phil Ross responded</u>: In my humble opinion, we ought to put together an unabridged list² (and geographical display/picture?) of all of the Univac computers used in support of the Apollo program ... and provide it with any display of individual items, etc.:

- the DSD 1218-family (from Mercury and Gemini) used at almost all MSFN stations worldwide
- the 418-family used at MCC/Houston
- the 490-family used at MCC/Houston and GSFC/Goddard
- the 1100-family used at NASA/Huntsville and JPL/Pasadena
- the DSD 1230-family (of "telemetry" and "command" computers used at all MSFN and DSN stations worldwide

For example, there were close to fifty 1230 computers used worldwide for each Apollo flight. NASA also started to use the "spare" 1218s (left over from Mercury a/o Gemini at each site) for the MSFN ASTAM project ... for "Automated Site Test and Monitoring" of site equipment to aide in determining pre-flight (launch) site readiness.

When you add 'em all up, Univac (quantitatively) had quite a worldwide presence and provided a lot of mission jigsaw puzzle pieces for the Apollo program.

Phil Ross worked for Univac at NASA/GSFC, DSD/TSD and FSD/Washington DC, UAL/Chicago, RSDC, BBHQ. During the Apollo Project at NASA/GSFC in Greenbelt, he was a systems programmer and then a systems design engineer (for Remington Rand Univac DSD/TSD). Phil wrote programs in AS-1, and CS-1 primarily for the "Telemetry" (downlink streams) computers - which were Univac DSD 642B (modified) computers (aka Univac 1230 computers).



² Editor's note: This paper primarily addresses the use of the 18- bit computer families. The 30-bit line (490 series and 1230 upgrades of the 642B); the 36 bit line (1100 series); and the later 16-bit 3760 communications processor are not delved into by this paper.



Mid Atlantic Retro Computer Hobbyist on the East Coast

Email communications from Duane Craps: I have been looking for tech info for the Mid Atlantic Retro Computer Hobbyist (MARCH) for info on 1219's, 1532's and 1540's that we have rescued from Johns Hopkins APL. We think we will be able to get the 1219 and 1532 publications from the last SPN-42 installation (Whidbey Island, WA) when they shut down. Yes, there are still operational CP-848's in the wild. MARCH hopes to have theirs operational someday.

Not bad for a 50 year old computer!

Our suite of equipment came from Johns Hopkins University Applied Physics Laboratory, where it was used to develop software for the Terrier Surface Missile System. If you know of anyone who might have 1540 info, please pass my request along.

MARCH has a Yahoo group with 474 members: https://groups.yahoo.com/neo/groups/midatlanti cretro/info. In addition, Duane Craps has a yahoo group for old military Univac Technicians: https://groups.yahoo.com/neo/groups/UnivacCo mputers/info. On this web site, there are files for my 1219 emulator, assembler, and programs. The emulator is in visual basic and only runs about 2000 instructions per second, but that is fast enough for utility type operations. On top of the next page is a screen shot of the emulator in operation with a memory dump of the 32-word bootstrap located at memory address 500₈.

I wrote a keyboard UPAC so that MARCH would have something if they get their hardware working. Still hope to run into a copy UPAC 1B

I have received a copy of the FACT test and paper tapes for the 1219B and am using them to squish bugs in my 1219 emulator. I built a paper tape reader with a photodiode array pulling down the internal pull-up resistors on an Arduino microcontroller. The rest is software.



{Editor's notes: 1) The bootstrap base address for the 1219B was 500_8 whereas the 1218 base address was 200_8 . 2) This 1219 picture was provided by Duane Craps.}



📼 🔟 🔀 📔 UNIVAC 1532 UNIVAC 1219 S/N 368 70K CRAPAC 075 031 000000 AU File Name for Punch / Reader Computer Input Computer Output READER PUNCH PRINTER **KEYBOARD** \mathbf{AL} 000031 MSTR CLR ✓ Ignore CR CLR TEXT I/O Chan SR ICR 1 00 // CRAPAC 1.2 // 01 NOV 13 // HELP=(CTRL)H MEMORYS: TYPE LETTER THEN 1-6 OCTAL DIGITS OR LESS AND SPACE I INITIAL S STARTING D DATA M MASK F SCALE-FACTOR (BBVVVV BIT,VAL) FUNCTIONS: 34 R FШ F Scale Function (BBVVV BIT,VAL) FUNCTIONS: / NEXT ADDRESS B BACK 1 ADD C INPUT CHANGE CTRL+L LIST I-T CTRL+D CORE DUMP I-T CTRL+S SEARCH I-T FOR D USING M CTRL+K STORE D AT I-T CTRL+P PUNCH PT I-T; SELF START IF S NOT ZERO CTRLK STORE D FROM I-T MATH:RESULTS GO TO I (+ ADD)(- SUB)(* MUL)(\ DIV BKSLASH) = PRINTS SCALED VALUE SUBJECT TO F (F=0 OCTAL / F=1 DECIMAL / F=210264 BAMS (B17=180)) I 500 \$03400 T 537 (CTRL+D 70330 Р Clear Stop Disc Adv P Func Load Run Start Repeat Run Step (CTRL)D 000500 000510 000520 503400 507310 700002 440004 504606 510002 560535 340507 507201 501300 700000 501100 630525 100003 505601 340540 000534 000002 600513 420003 340516 450536 000534 360000 STOPS 000002 570004 000141 502100 340513 760054 2 1 3 4 0 000530 000151 000712 000123 Ī 100 2 000101 442000 000100 000002 2 B D SKIPS 8 4 3 2 1 0 16. 010000 000101 442000 000100 010000 c в 210264 B 000013 5.625 1560

A Swedish UNIVAC Computer Museum

Mr. Carlsson: thank you for the snapshots of the Univac 418 II bootstrap memory card. I see 18 vertical wires on the card, indicative of the 18-bit word length of the 418 commercial computer series. I do not have the instruction set in front of me, however vaguely recall that many of the input/output instructions for 18-bit computers started with octal code 50. With that in mind, I traced the horizontal wire core locations from top to bottom to get the following bit patterns, i.e. machine code instructions.





The top to bottom physical sequence is not the program execution sequence!

Row	Binary Bits	Octal Code(s)	Instruction
1	010 000 001 000 000 000	20 1000	Add contents of Adr 1000 ₈ to A
2	011 100 000 010 010 111	34 0227	Move contents of Adrs 227 ₈ to the
			Interrupt Address Register.
3	010 000 000 000 000 000	20 0000	Add A
4	101 000 100 111 000 110	5047 06	Shift A left 6 bits
5	101 000 010 100 000 000	5024 00	Wait for interrupt
6	101 001 000 000 000 000	51 0000	Inclusive OR
7	000 000 000 000 000 000	00 0 0000	
8	110 000 000 010 010 100	60 0224	Jump if Au Zero to 224 ₈
9	000 000 000 000 000 000	000000	
10	101 000 100 001 000 001	5021 01	Test Input on channel 1
11	000 000 000 000 000 000	000000	
12	100 101 000 000 000 000	45 0000	Store Al
13	100 000 000 000 000 000	40 0000	Store Zero
14	110 000 000 010 010 010	60 0222	Jump Au Zero to 222
15	100 000 000 000 000 000	40 0000	Store Zero
16	011 101 000 000 000 000	35 0000	Jump to
17	101 000 001 001 000 000	5011 00	Load input channel with 0 (zero)
18	101 000 010 001 000 000	5021 00	Test Input on channel 0
19	101 000 010 110 000 000	5026 00	Output a word on channel 0
20	000 000 000 000 000 000	000000	
21	000 000 000 010 001 110	000216	216 ₈
22	000 000 000 000 000 000	000000	
23	000 000 000 010 001 110	000216	216 ₈
24	101 000 001 001 000 000	5011 00	Load input channel from nxt adrs
25	101 000 001 011 000 000	5013 00	Load input channel, fct code
26	011 111 000 000 000 001	37 0001	Enter B with constant 1
27	101 000 111 011 000 000	5073 00	Load special register w/0
28	010 100 000 001 010 101	24 0125	Multiply x contents of adrs 125 ₈
29	101 000 001 111 000 000	5017 00	Terminate external function
30	011 110 000 000 000 000	36 0000	Enter B
31	101 000 001 101 000 000	5015 00	Terminate input Ch
32	101 000 111 010 000 001	5072 01	Load Index Control Register with 1

Of course, these 32 words of non-destructive readout include some constants for external function codes to the device from which larger programs are being read – usually, the first program loaded is a more sophisticated loader. A few of the instructions are also used to setup an initial interrupt process for the selected channel.

Appendix A of this paper lists a 1218 paper tape load bootstrap program that is executed beginning at address 200₈. The row 4 instruction above shifts the accumulator by six bits, the number of bits of a paper tape character.



We do have a couple of bootstrap listings in pdf copies of 1218 technical manuals. A paper tape listing is Appendix A of this paper. The instruction codes for the 1218 are almost identical to those used by the 418 II as illustrated in this card image from Anders.

	U	VI	VAC	2	8		1	1-		Actions	100		-			E	-176					
DESCRIPTION	UMP JUMP INDIRECT EXABLE RIFERIOPTS AND JUMP INDIRECT STORE LOCATION AND JUMP INDIRECT STORE LOCATION AND JUMP INDIRECT	TEST KEYS TEST NO BORROW	TEST OVERFLOW TEST NO OVERFLOW TEST EVEN PARITY TEST EVEN PARITY	LOAD INPUT CHANNEL		LOAD OUTPUT CHANNEL LOAD EXTERNAL FUNCTION CHANNEL	STOP INPUT ON CHANNEL	STOP OUTPUT ON CHANNEL TEST INPUT ON CHANNEL	TEST EXTERNAL FUNCTION ON CHANNEL	OUTPUT OVERRIDE	ALLOW ALL INTERRUPTS	ALLOW FUNCTION INTERRUPTS	PREVENT ALL INTERRUPTS	PREVENT FUNCTION INTERRUPTS	WAIT FOR INTERRUPT	STOP ON KEY SETTINGS		position.	idexing renguens execution true of fes-	ested as a constant in the range -3777 to the register named in the parentheses, contents of Y.	bei. ; V = inclusive OR; # = exclusive OR	
OPERATION	$ \begin{array}{l} Y + JAR \\ (Y_{1+2}) - JAR \\ (Y_{1+2}) - JAR \\ DARLES INTERENTES, (Y_{1+3}) + JAR \\ (AR) + 1 - Y, Y_{1+2} - AR \\ (AR) + 1 + LOCATION SPECIFIED BY (Y); \\ (AR) + 1 + LOCATION SPECIFIED BY (Y); \end{array} $	IF KEYS DESIGNATED BY K ARE SET, IF BORROW INDICATOR OFF,	IF OVERFLOW INDICATOR ON. IF OVERFLOW INDICATOR OF, IF SUM OF 1'S IN (AU)/VAL) IS ODD, IF SUM OF 1'S IN (AU)/VAL) IS EVEN,	LOAD 1/0 CHANNEL K FROM (IAR) +1 AND	(IAR) +2, INITIATE INPUT, THEN (IAR) +3 - IAR	SAME AS LIC EXCEPT OUTPUT IS INITIATED SAME AS LIC EXCEPT EXTERNAL FUNCTION	IS INITIATED STOP INPUT ON CHANNEL K	STOP OUTPUT ON CHANNEL K IF INPUT CHANNEL K IS IDLE, (IAR)+2 + IAR	IF OUTFUT CHANNEL & IS IULE, INVEST INT IF EXTERNAL FUNCTION CHANNEL K IS IDLE,	OUTPUT ONE WORD ON CHANNEL K	ALLOW ALL INTERRUPTS (EXCEPT EXTERNAL IF PFI USED)	ALLOW EXTERNAL INTERRUPTS	PREVENT ALL INTERRUPTS	PREVENT EXTERNAL INTERRUPTS	STOP CENTRAL PROCESSOR (BUT NOT 1/0) TILL INTERRUPT	STOP IF KEYS DESIGNATED BY K ARE SET		egister or quantity subscripted. N being each bit	operand with *. ART will add i to octal op coos. It Z = The low-ord to 16 bits b	acting as one register () = Contents of $\alpha_{\rm eff}(x) = \alpha_{\rm eff$	the storage address in \sim - compare any of this symmetry of this symmetry Λ = Logical and the storage register	figuration in the tion (bits 5-0).
IMING IN IS	N 4 4 4 D	2.3-3	2.3-3 2.3-3 2.3-3 2.3-3 2.3-3	10		10	2	2.3-3	2.3-3	10-12	3	3	2	2	2	2.3		in the r	T, prefix	seulators ister agister	symbol, th of the ins symbol, t ess.	e instruct
SEE T EG.	-111-	1.1	1111	1		1.1	1	1 1 1	1 1	1	1	1	1	1	1	1		positio	n in AR	lator lator ver Acci dress R	the + the + f the + hat add	nteger of th
5 J W	15 25 25 26 30 26 26	0505	5052 5053 5055 5055	5011	*****	5012 5013	5015	5016 5021	5023	5026	5030 or or	5032 60 60	5034 5035 5035	5036 or 5037	5024 or 5025	8056		ecify bit	ns tructio ble sitive	Accumu Accumu and Lov ctive Ind tion Ad	re Desig heft of w-order right o	isigned i fer 6 bi
OP				-									1	1	1	×	:0	ts sp	dexal dexal	ower pper pper he ac	Compa On the the lov on the specifi	The un low-ord
PER- OCTAL ND OP YPE CODE	****	× 1	1111	×	4	××	×	* * :	XX	×							2	-	s In S	122421		
ART OPER. OCTAL OP AND OP CODE TYPE CODE	JANOTTONOONU AMUT I	TK TNB TNB	SKIP TTEP	N OIT	2	709 200 200 200	OUT STIC K	NPUT STOC K	100 K TFC K	A NO	i vvi	ONTRO MF1	PT CC	1993	INT WFI	9012 X	LEGEN	Subscrip	To index 1 = 1n S = 51	AL = L AU = U R = T IR = T	= = 0 ≻	X
ART OPER OCTAL OP AND OP CODE TYPE CODE		Tin Tin		N NI	2	TUG LFC CC	00110 K	NPUT STOC	TFC K	× ^0		NUTRO	2 14	E C		aors ×	regen	Subscrip		30	₩ 	" ¥
DESCRIPTION ART OPER OCTAL OP AND OP CODE TYPE CODE		TIN T	Auto Manual Manual Manua Georgenetica Manual	COUPLEMENT A COUPLEMENT A	LOAD AU STORE AU	LOND AL STORE AL STORE AL	MASKED SELECTIVE LOAD	STORE ADDRESS OF AL	LOAD B REGISTER WITH "KONSTANT" III TTOC K LOAD INDEX CONTROL REGISTER	STORE INDEX CONTROL REGISTER LOAD SPECIAL REGISTER	STORE SPECIAL REGISTER AND INACTIVATE	SHIFT RIGHT AU SHIFT RIGHT AL	SHIFT RIGHT A			JUMP IF B REGISTER NON-ZERO	TEST B REGISTER FOR EQUALITY	TEST ANY LOCATION FOR ZERO	To Indee COMPARE LONER COMPARE LONER NUMBER OF UPPER		13/#P ON 12 72/#O C = 0 = 1 .13/#P ON 12 72/#O C = 7 .13/#P ON 42 M POSTIVE	JUMP ON AL ZERO K = JUMP ON AL NON-ZERO K = JUMP ON AL POSITIVE
OPERATION DESCRIPTION ART OPER-IOTAL OPEN-OPEN-OPEN-OPEN-OPEN-OPEN-OPEN-OPEN-	Поль на Аво то цовек 11-11-11-24 Аво то цовек 11-21-24-34 Аво то цовек и «константи 10-21-24-34 11-21-24-34 Аво то цовек и «константи 200 то цовек и «константи 10-21-24-34 11-21-24 Аво то цовек и «константи 200 то ценк и «константи 200 то ценк и «константи 20	- 41 JF (A) ARE REGUTIVE AND (AL.,) = (401)-1 - 41, IF OTHERNEE, (AU) - AL.	しいやーキル AND	(A) + A COMPLEMENT A LIVE A LI) + AU LOAD AU () - Y STORE AU () - Y	7 = AL LORD AL LORD AL 1000 AL 1000 K =		Title X Title X Title X<	A HR LOAD BREGISTER WITH "KONSTANT"	TOR + Y ₁₀ / (1010 - 000.001 + Y ₁₀) STORE INDEX CONTROL REGISTER to - SR / (1010 × 000.000 + Y ₁₀) LOAD SPECIAL REGISTER	R) + Y ₅₋₀ : 0 -5R4 STORE SPECIAL REGISTER AND INACTIVATE	IFT AU RIGHT REMOVED & BIEFT RIGHT AU UFT AL RIGHT PL BIEFT RIGHT AU UFT AL RIGHT POSITIONS SHIFT RIGHT AL	IFT A RIGHT) SHIFT RIGHT A IFT AU LEFT) (END-ARGUNDIX BIT SHIFT LEFT AU		PETA LET TO ERD-ANDURON KUT PSTATUTE ERD-ANDURON KUT LESS ACTUAL SWITT + ABOLLER LESS ACTUAL SWITT + ABOLLER	(18) 7 0, (18) -1 + 18 AND Y + 14R JUNP IF B REGISTER NON-ZERO	(18) = 0, (1A8)+1 + IAR (18) = (Y), (1A8)+2 + IAR TEST B REGISTER FOR EQUALITY	(18) 2 (V), (18)+1 + 18 (10) 0. (18)+2 - JAR TEST ANY LOCATION FOR ZERD	10 Index 1.1.1.1/1; FFT CD ACCORDINGLY COMPARE LONER ACCORDIN (ALL) I (AU/A / Y1); FET CD COMPARE LONER MAXED BY UPPER	со брим, сомоттам, мот со срум, сомоттам, мот со срум, сомоттом, и на со съзя мы сомотом, на со съзя мы сомотом, на со съзя мы сомотом, ман съз кади, сомотом, на со съзя мы сомотом, ман съз кади, сомотом, ман съз кади, ман съз кади, сомотом, ман съз кади, ман съз кади, м	(40) = 0, (40) = 0, (40) # 9 (40) # 9 (40	(AL) = 0, (AL) = 0, (AL) 2000 (A.A. 2210 (A. 2210)))))))))))))))))))))))))))))))))))
International DESCRIPTION DESCRIPTION OF AND	4 (U): (M): +U, A00 T0 LORER A00 T0 LORER M 2.3 (U): (Y): + X, MAD MOI T0 LORER A "GORTATI" M 2.3 (U): Y: - X, MOI T0 LORER A "GORTATI" MOI T0 LORER A "GORTATI" M 2.4 (U): Y: - X, MOI T0 LORER A "GORTATI" MOI T0 LORER A "GORTATI" M 2.3 (U): - Y: - Y, MOI T0 LORER A "GORTATI" MOI T0 LORER A "GORTATI" M 2.3 (U): - PL, (U): - AL MOI T0 LORER A "GORTATI" M 2.3 (U): - AL MAD MOI T0 LORER A "GORTATI" M 2.3 (U): - AL MAD MOI T0 LORER A "GORTATI" M 2.3 (U): - AL MAD MOI T0 LORER A "GORTATI" M	+ 4.1. IF (A) ARE REGATIVE AND (41., 1) = (2, 401) - 1 + A1, IF OTHERRISE, (401) + AL.	4 441/X/17-441 Au 411/X/17-441 Au 411/X/17-441 Recutation (1) 212 - 1-041-441 Contruction (1) 212 - 1-041-441 Contruction (1) 212 - 1-041-441 Contruction (1) 213 - 1-041-441 Contruction (1) 214 - 141 -	27 - (A)+A COUPLEMENT A L	4 (Y)+AU LOAD AU 8 1 (V)+AU 10AU 8 1 (V)+AU 10AU 10AU 10AU 10AU 10AU 10AU 10AU 10	4 (Y) = AL 4 (Y) = AL 4 (Y) = Y (200 AL 5 (AL) = Y (200 AL) = Y (200		M ¹¹ :0 ¹ + 11:0 STORE ADMESS OF AL 5 (M ¹) → IR 0:01 → Y STORE REGISTER	4 2-1/R 2 -1/R 2 -1/R 2 -1/R 2 -1/R 2 -1/R 10.0 NIDEX CONTROL REDISTER 10 1 -10 1	4 (10 ²⁶⁰ + Y ₂) { r1(100 + 000001 + V ₁₋₃) stone invex control. Relater 2 (K ₄₀ + S ₁₀) { r1(100 / 000000 + V ₁₋₃) LOAD SPECIAL RELATER	4 (58) + Y ₁₋₀ : 0 + 58, 0 + Y 1-0; 0 + 58, 0.LEARY CLEARY AND INACTIVATE	12+K/2 Built AU ROHT (ENDOFF)K BIT RIGHT AU 12+K/2 Built AU ROHT (ENDOFF)K BIT AU RIGHT AL 12+K/2 BHILT AL RIGHT AL	27+K/3 SHIFT A RIGHT A SHIFT RIGHT A SHIFT RIGHT A SHIFT A RIGHT A SHIFT A RIGHT A SHIFT AU LEFT A CARDANDIX BIT SHIFT RIGHT A		UT+K/0 SHITA FILE TRO-MODION (ST TO A COLLE A	6 IF (MP \$ 0, (IR)-1.4 IR AND Y + IAR JUMP IF B REGISTER NON-ZEND TO 250 SK	IF (IR) = 0, (IAR)+1 + IAR TEST B REGISTER FOR EQUALITY LEGEN	6 IF (IR) # (V), (IR)+1 + IR 1 F (IR) = 0, (IR)+2 + IAR 1 EST ANY LOCATION FOR ZERO	4 (AL): (Y): SET CD ACCORDINGLY COMPARE LONER To Index To Index 4 (AU): (AU): (Y): SET CD COMPARE LONER COMPARE LONER To I = I 4 (AU): (AU): (Y): SET CD COMPARE LONER COMPARE LONER To I = I	2 IF CD EQUAL CONSTITUT, 1 JUMP ON EQUAL. JUMP ON EQUAL. JUMP ON EQUAL. 2 IF CD LEST NON-CONSTITUT, 1 JUMP ON ECUAL. JUMP ON ECUAL. JUMP ON ECUAL. 2 IF CD LEST NON-CONSTITUT, 1 JUMP ON ECUAL. JUMP ON ECUAL. JUMP ON ECUAL. 2 IF CD LEST NON-CONSTITUT, 1 JUMP ON ECUAL. JUMP ON ECUAL. JUMP ON ECUAL. 2 IF CD LEST NON-CONSTITUT, 1 JUMP ON ECUAL. JUMP ON ECUAL. JUMP ON ECUAL.	2 IF (AU) = 0, 2 IF (AU) = 0, 2 IF (AU) = 0, 2 IF (AU) = PONTWE, 2 IF (AU) = PONTWE, 2 IF (AU) = PONTWE, 4 - AUA 1 AUAF ON AN INSTITUTE 2 IF (AU) = PONTWE, 4 - AUA 2 IF (AU) = PONTWE, 2 IF (AU) = PONTWE, 4 - AUA 2 IF (AU) = PONTWE,	2 IF (AL) = 0, JURPO ON AL 2010 K = 1 (AL) = 0, JURPO ON AL 2010 K = 1 (AL)
XE MANG SEE MANG TELE MANG TEL	13 4 A(1)+(1) = A1 A00 10.00EK 14 5 (1)+(1)+A1 A00 10.00EK 15 5 (1)+(1)+A1 A00 10.00EK 13 6 (1)+(1)+A1 A00 10.00EK 13 6 (1)+(1)+A1 A00 10.00EK 13 4 (1)+(1)+A1 A00 10.00EK 14 4 (1)+(1)+A1 A00 10.00EK 13 4 (1)+(1)+A1 A00 10.00EK 13 4 (1)+(1)+A1 (1)+(1)+(1) 13 (1)+(1)+(1)+(1) (1)+(1)+(1)+(1) (1)+(1)+(1)+(1)+(1)+(1)+(1)+(1)+(1)+(1)+	+ 4.1.1 F (A) ARE RECAVINE AND (A.1.) = 0, AU3 - 1 + AL1 F OPREMUE, (AU) + AL.		- 2.7 - (A) + A COMPLEMENT A LIN K	1,5 4 (Y1) + AU LOAD AU 00 11,5 4 (AU) + Y STORE AU 00	1,5 4 (7)+AL LOBD AL 1,5 4 (AL)+Y STORE AL 2,3 7,4L LODD AL	J.S. 4 (Y ₁) + AL _N (r/ AU _N) = 1 MASKED SELECTIVE LOAD 0 3710 K	+ 0 ^{11:0} + ¹ 1:0 5 100 K 1,5 6 (M) + IR 1.040 B REGISTER 2 100 2 100 1,5 6 (M) + IR 5 100 X 2 100 X	2 K _{2,m} + 100 x - 100 NG SEGISTER WTH "KUNSTANT"	4 1070 + Y ₁₀ (1070) = 000.01 + Y ₁₀ 2 K ₁₀ + SR K ₁₀ + SR 10700 + Y ₁₀ 10700 SPECIAL REGISTER	4 (58) + Y ₈₋₀ ; 0 + 58, STORE PECIAL REDISTER AND INACTIVATE Λ 1,5 4 0-+Y V ₈₋₀ ; 0 + 58, Δ	- 2.27+K/3 BNIFT AU RIGHT AU RUNT RUDHT AU - 2.27+K/3 BNIFT AL RUDHT AU BNIFT RUDHT AU BNIFT RUDHT AU	- 227+K/3 SHIFT A RIGHT SHIFT RIGHT A SHIFT RIGHT A - 227+K/3 SHIFT A A - 227+K/3 SH	- 22+K/3 SHIFT ALLEFT POSITIONS SHIFT LEFT A SHIFT LEFT A	- 4.7+K/3 SHIFT ALER Generation K SOALE A SOALE A SALE A S	- 6 IF (IN) 50, (IN)-1 + IR AND Y + IAR JUHP IF B REGISTER NON-ZEAD	- 8 IF (IR) = 0, (IAR)+1 + IAR - 8 IF (IR) = (Y), (IR)+2 + IAR TEST B REGISTER FOR EQUALITY LEGEN	- 6 IP (18) 2 (18) + 1 + 18 TEST ANY LOCATION FOR ZERO	List 4 Aut.: (Y): aft to Accelenual.Y Contrast. Longin To interval. List 4 [(Aut.: Aut.)]: [(J): aft to Accelenual.Y Contrast. Longin To interval. List 4 [(Aut.: Aut.)]: [(J): aft to Accelenual.Y Contrast. Longin Maximum	2 1 0 <td>- 2 IF (AU) = 0, JUMP CM 4J 2EHO CO = CO = - 2 IF (AU) = 6, JUMP CM AJ 3EHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7</td> <td>- 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 200P 00 AL IONE2E00 NK = 2 IF</td>	- 2 IF (AU) = 0, JUMP CM 4J 2EHO CO = CO = - 2 IF (AU) = 6, JUMP CM AJ 3EHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7 - 2 IF (AU) = 76, JUMP CM AJ MON-ZEHO Y = 7	- 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 100P 00 AL IONE2E00 NK = 2 IF (ALL = 0, 200P 00 AL IONE2E00 NK = 2 IF
000/105 1860 0° 156 1890 00 1590 45 0005 1900 45 0005 1900 45 0005 1910 45 0005 1910 45 005	11 13 4 0.11+70-AL 71 1 5 0.117-34 47 71 7 7 71 1 2 0.17-34 47 71 7 7 71 1 2 0.17-34 47 74 74 74 71 1 2 0.17-34 400 70 14 7 71 1 4 0.17-77-8L 400 70 14 7 71 1 1 1 14 14 14 7 7 1 1 1 1 14 14 7 7 1 1 1 1 14 14 14 14 14 8 1 1 1 1 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14		2 - + AUJ/OT/-AL AND 3 - + AUJ/OT/-AL MU 3 - + AUJ/OT/-AL MU 3 - + AUJ/OT/-AL MU 33 - + AUJ/OT/-AL MU 33 - 1 AUJ/OT/-AL COLUMPC 0R 33 - 1 2 AUJ/OT/-AL 333 - 1 2 2 333 - 1 2 3 333 - 1 2 3	863 = 2.7 = (A) + A COMPLEMENT A	10 1,3 4 (Y)+AU LOUDAU &	12 (1,5 4 (1)**A.L LOBO A.L 44 (1,5 4 (ALL)+Y STORE A.L LOAD A.L. 20 2 3 7 ± ALL LOAD A.L. 1.0.0.0 H.K.Y.**ONTAAT********************************		Nat - 4 (N-L)(-a)^{-1} + 1(-a) STORE ADDRESS OF AL 28 15 6 (N) + 18 (N) + 18 (N) + 18 20 15 5 (N) + 18 (N) + 18 (N) + 18	36 1 4 2 + 1R - LOAD REGISTER WITH "KONSTANT" - TOC K 3072 2 K_a, + 1CR ,	72 4 (107h - Y ₁₋₀) (1107h - 000001 - Y ₁₋₀) 5 TONE INDEX CONTROL INEGISTER 5013 2 K ₄₋₀ - 5R (1107h z 000000 - Y ₁₋₀) LOAD SPECIAL REGISTER	73 4 581, + Y ₁₋₃ : 0 + 584, STORE SPECIAL REGISTER AND INACTIVATE AII 40 1,5 4 0 + Y V ₁₋₃ : 0 + 584, OLEMAY	341 - 2,2+K/3 SHIFT AU RIGHT (ENDOFT)K BITT RIGHT AU 362 - 2,7+K/3 SHIFT AU RIGHT (ENDOFT)K BITT RIGHT AU	566 - 2.2+K/3 SHIFT A RIGHT SHIFT RIGHT A 568 - 2.2+K/3 SHIFT A RIGHT SHIFT RIGHT A 588 - 2.2+K/3 SHIFT AN LEFT SHIFT RIGHT A	904 - 2.7+K/3 SHIFT ALLEFT POSITIONS SHIFT LEFT A 27+K/3 SHIFT ALLEFT POSITIONS	SMH - 4/3 +K/3 SMFTA LERO-ADDONO K BIT POSITIOS OF TIL ERO-ADDONO K BIT LESS ACTUAL SMFT → ABBOLITE LESS ACTUAL SMFT → ABBOLITE	73 = 6 IF (R) # 0, (IR)-1 + IR AND Y + AR JUND IF B REGISTER NON-ZEND III	56 - 8 IF (IR)= (IAR)+1 + IAR TEST B REGISTER FOR EQUALITY LEGEN	5 - 6 IF (1) - 6 (1) - 10 (1) - 18 TEST ANY LOGATION FOR ZERO	ComParison C	80 F61 - 2 F Co. Equiv.LonderTon, Juse On Equiv.LonderTon,	60 - 2 F (AU) = 0, UupP OK AI ZENO CO	5 2 IF (AL) = 0, (AL) JUNE OR AL ZELO K <t< td=""></t<>
04% 0.0% 1.0% 1.0% 0.0% 0.0% 0.0% 0.0% 0.0	γ 1 1 3 4 0.01 10.00EA 2 1 1 3 0.01 10.00EA 0.01 10.00EA 2 11 - 13 0.01 17.01 + M.Y. match and 0.01 0.01 0.00EA 0.01 10.00EA 7 11 - 13 0.01 10.01 + M.Y. match and 0.01 0.01 0.00EA 0.01 10.01 + M.Y. M.Y. M.Y. M.Y. M.Y. M.Y. M.Y. M.	4 AL IF (A) ARE NEGATIVE AND OLL;)= α (AU) - 1 - ALL IF OTHERNES, (AU) - AL. TRA - AL	γ 2 1 4 λλ.//// + λt. 100 100 100	= 503 = 2.7 - (A) + A CONPLEMENT A CONPLACIÓN CONPLEMENT A CONPLEMENT A CONPLEMENT A CONPLEMENT	Y 18 1,5 4 (Y)+A,U LOAD AU 90 10	Y 12 [13 4 [1] + [100 ÅL Y 14 [15 4 [1]] + [100 ÅL 7 10 2 3 7 4 [1] + [100 ÅL		Y Xi - 4 Mitting ¹ + 1(i.e) Strong Address of AL Y Xi Xi	Z 36 1 4 Z + IR 2 100 K 200 K 775	Y 72 4 10 ¹²⁰ + Y ₁₋₀ 71 ¹²⁰ + V ₁₋₀ 570 ¹⁸ + MDEX CONTROL REDISTER 0 K K 5073 2 K ₁₊₀ - 5R ³ 810000 - Y ₁₋₀ L0000 5PECIAL REDISTER 0 K	Y 73 4 (58) + Y ₀ -0; 0 + 38, STORE SPECIAL REGISTER AND INACTIVATE A Y 40 LS 4 0-Y Y ₀ -0; 0 + 38, CLEARY CLEARY 0 A <t< td=""><td>K Soul - 2.3+K/3 Suirt Au Biolr1 Cabora K art Suirt Right Au Suirt Right Au Suirt Right Au K Soul - 2.3+K/3 Suirt Au Right Embody Suirt Right Au Suirt Right Au K Soul - 2.3+K/3 Suirt Au Right Positions Suirt Right Au</td><td>K 504 - 27+K/3 5HIT A RIGHT A RUELT A RUELT A SHITT RUELT A RUELT RUELT A RUEL</td><td>K S09 - 2.7+K3 SHIFT ALLEFT POSITIONS SHIFT LEFT A SHIFT ALLEFT POSITIONS</td><td>K 894 - 4.7+K/3 BHFFA LEFT ERG-MOUNDING BET 4.7+K/3 BHFFA LEFT ERG-MOUNDING BET LESS AGTUAL SHFF + ABGULTEE LESS AGTUAL SHFF + ABGULTEE</td><td>Y 73 - 6 IF (IN) F 0, 10 + 14 R AND Y = MR JUMP IF B REGISTER MON-ZERO M</td><td>Y S6 - 8 IF (IR) = 0, (IAN+1 + IAR LEGEN LEGEN LEGEN</td><td>Y S - 6 IF (48) - 7 (Y), (18)+1 + 48 - 6 IF (13) - 0, (18)+2 + 48 TEST ANY LOCATION FOR ZERO</td><td>Y COMPARE LONER ComPARE LONER To Index Y 00 1,5 4 I/U/J.SET CD COMPARE LONER MARCH DE UNPER To Index Y 00 1,5 4 I/U/J.ALIJ.EU/J.Y.Y.J.SET CD COMPARE LONER MARCH DE UNPER To Index 1 = In</td><td>Y IBerefit - 2 IF CD EGNLAL CONTOTION, IF POT CD EGNLAL CONTOTION, Y JUSE OR COLL JUSE OR C</td><td>Y S0 </td><td>Y Ki - 2 IF (AL) = 0, I. (AL) = 0, Y JUNPO (BA A. ZERO) K = Y Ki - - II (AL) = 0, I. (AL) & PO(IP)(R, I. (AL) & PO(IP)(R, IP)(R, IP)(R, IP)(R, I. (AL) & PO(IP)(R, IP)(R, IP)</td></t<>	K Soul - 2.3+K/3 Suirt Au Biolr1 Cabora K art Suirt Right Au Suirt Right Au Suirt Right Au K Soul - 2.3+K/3 Suirt Au Right Embody Suirt Right Au Suirt Right Au K Soul - 2.3+K/3 Suirt Au Right Positions Suirt Right Au	K 504 - 27+K/3 5HIT A RIGHT A RUELT A RUELT A SHITT RUELT A RUELT RUELT A RUEL	K S09 - 2.7+K3 SHIFT ALLEFT POSITIONS SHIFT LEFT A SHIFT ALLEFT POSITIONS	K 894 - 4.7+K/3 BHFFA LEFT ERG-MOUNDING BET 4.7+K/3 BHFFA LEFT ERG-MOUNDING BET LESS AGTUAL SHFF + ABGULTEE LESS AGTUAL SHFF + ABGULTEE	Y 73 - 6 IF (IN) F 0, 10 + 14 R AND Y = MR JUMP IF B REGISTER MON-ZERO M	Y S6 - 8 IF (IR) = 0, (IAN+1 + IAR LEGEN LEGEN LEGEN	Y S - 6 IF (48) - 7 (Y), (18)+1 + 48 - 6 IF (13) - 0, (18)+2 + 48 TEST ANY LOCATION FOR ZERO	Y COMPARE LONER ComPARE LONER To Index Y 00 1,5 4 I/U/J.SET CD COMPARE LONER MARCH DE UNPER To Index Y 00 1,5 4 I/U/J.ALIJ.EU/J.Y.Y.J.SET CD COMPARE LONER MARCH DE UNPER To Index 1 = In	Y IBerefit - 2 IF CD EGNLAL CONTOTION, IF POT CD EGNLAL CONTOTION, Y JUSE OR COLL JUSE OR C	Y S0	Y Ki - 2 IF (AL) = 0, I. (AL) = 0, Y JUNPO (BA A. ZERO) K = Y Ki - - II (AL) = 0, I. (AL) & PO(IP)(R, I. (AL) & PO(IP)(R, IP)(R, IP)(R, IP)(R, I. (AL) & PO(IP)(R, IP)(R, IP)



NASA installation clips and notes

This item was clipped from a 1965 UNIVAC newsletter, Twins Cities section.



JUNE-JULY, 1965

Astronauts Could Depend On Reliable Univac Computers

Highly reliable Univac 1218 Computers operated a total of 910 hours during the historic Gemini 4 space mission earlier this month — without a single failure.

Twelve 1218's, strung out at bases surrounding the earth, kept track of information within the capsule and the astronauts' physical condition during the four-orbit flight.

The computers summarized and stored the data that was beamed to the ground from the capsule. The data was transmitted for final processing on two Univac 490 computers at Goddard space flight center, Greenbelt, Md., and the Houston mission control center.

The systems also enabled the National Aeronautics and Space Administration controllers at Goddard to select and examine specific in-flight information such as the astronauts' pulse beat, blood pressure and temperature, on a real-time (no delay) basis.

The 1218 computers, made in Univac Plant 3, are installed at Cape Kennedy; Bermuda; the Canary Islands; Corpus Christi, Texas; Guaymas, Mexico; Kauai, Hawaii; Carnarvon, Australia; Wallops Island, Va.; Greenbelt, Maryland, and aboard two tracking ships on the Atlantic Missile Range.

From Ron Q. Smith Subject: UNIVAC 1218 in Australia - A UNIVAC 1218 was installed at the Honeysuckle Creek tracking station. <u>http://www.honeysucklecreek.net/station/technical.html</u>

From Phil Ross: Toward the end of the Honeysuckle Creek web document, there is a picture of the computer room, showing the two Univac M642B computers with an extended memory unit (EMU) in between. "Telemetry" computer is the left one; "Command" computer is the right one (at every site, worldwide). Trivia: The "Honeysuckle Creek" (Canberra) site is now a camping area, but the 26 meter DSN dish has since been moved about 7 miles north to the current NASA/JPL CDSCC site where it'll be used in the near future for educational purposes.





From Arlyn Solberg: Univac had both 494's at Mission Control Center (Bldg. 30) and 1230's which were being installed at Apollo tracking sites around the world. The 1230's were replacing the Gemini program 1218's which were connected to U494's at Goddard space Center in Greenbelt, MD which in turn was connected to Houston's 494's via three 40.8 kbs land lines. There was also a 1218 computer in Building 30 that drove all the big screen displays in Mission Control. I believe it interfaced with the 494's receiving satellite tracking data in real time and driving the big wall displays showing the satellite and a couple of track revolutions. I had two programmers and two engineers on a one-year contract that was extended several times. The Manned Spacecraft Center was south of Houston about 20 miles; chosen as the best location in the USA by Lyndon Johnson who was President at the time. The 'Campus' consisted of about 20 buildings of various sizes on about 100 acres. I recall that Vic Benda (subsequently helped start AIC) probably wrote most of the programs for the 418 II and Bob Malnatti helped design interface hardware between the two systems.

From Jerry Anderson: "In the mid 1970's, I sold heavily modified Univac 3760³ communications computers to NASA NASCOM. These units replaced three UNIVAC 494s (two live and one standby) and a room full of multiplexers (probably 18 or so). Whereas the communication speed for the older equipment was 9.6 kbs, the 3760's could process at a breathtaking (56 kbs) and the footprint was a fraction of what had been in place.

This was a big deal for NASCOM in that the underlying software, operations, and maintenance protocols also changed significantly. UNIVAC managed all of the hardware and software integration as well as the implementation and it went really well. UNIVAC personnel also worked very closely with NASA engineers and operations persons throughout the entire process.

The UNIVAC Salt Lake City engineering team was extraordinary as was Nate McCawley of the software division (already under Contract to NASCOM). These changes modernized the operations and ultimately saved NASA lots of money. I hope that this helps. Since my neighbor was with NASCOM at the time, I checked with before drafting this message."

Editor:

Lowell A. Benson is a 1966 BEE graduate from the University of Minnesota. His UNIVAC positions started in 1960; from 1963 to 1966, he was a computer operator in plant 1 working with the 1004, 1206, 1218, 1219, 1232, 1240, CP-667, and SS-80. In 1994 UNISYS gave him opportunity to find other employment. He joined the VIP Club in 2002, elected to the Board in 2005 when he volunteered to co-chair the IT Legacy Committee. He started the vipclubmn.org web site in 2006 then began the 'Article for the Month' series in April 2007 [http://vipclubmn.org/documents.html#Articles.]

Thanks to all who have contributed to this article.

³ The U3760 was a variation of the U1616 16-bit computer, both developed in St. Paul. It was the first UNIVAC/Sperry machine to incorporate an IBM byte-serial interface. The U3760 was superseded by the DCP-40 which used microprogramming techniques from the AN/UYK-20 16 bit computer.





VIP CLUB

Appendix A - 1218 Bootstrap to load from a paper tape.

1218 COMPUTER OPERATOR'S SECTION

Paragraph 3-3d

<u>d.</u> MANUAL READING OF A PROGRAM. - At times it is necessary to inspect each instruction of a program or routine, displaying each instruction consecutively at a slow speed (STEP MODE). This may be performed as follows:

STEP 1. Place the SEQ STOP/STOP switch momentarily in the down position.

STEP 2. Push the OP STEP MODE button.

STEP 3. Place the MASTER CLEAR switch momentarily in the down position.

STEP 4. Set the starting address in the P register,

STEP 5. Place the RESTART/START STEP switch in the down position.

With STEP 5 completed the contents of the address which was placed in the P register will now be displayed in the Z register. The S register contains the address of the contents of Z and the P register contains S+1. The instruction in the Z register has not been performed at this time. When the start switch is again depressed the instruction in Z will be performed and the next instruction will be read. The computer stops with Z containing the next instruction. S containing the address of Z and P containing S+1. STEP 5 is repeated as many times as necessary to complete the program or routine.

3-4, WIRE MEMORY (BOOTSTRAP),

<u>a.</u> GENERAL. - The 1218 Computer has 32_{10} memory address locations (00200₈ through 002378) which contain the wired memory bootstrap program for an initial load or bootstrap program. The computer can have a bootstrap written for paper tape, magnetic tape, or Teletype. The following paragraphs describe the paper tape bootstrap as a typical example.

The wired load program provides the ability to enter an initial package of utility routines which may subsequently be used to enter and debug more sophisticated programs. These memory address locations have unique characteristics in that they operate in a special type of non-destructive read-out mode. They are not accessible to the programmer for store type instructions.

The wired memory bootstrap program first locks out all Interrupts then loads in the 1218 Utility Package. The utility package is formulated so that only the basic load routine is read into memory addresses 00240_8 through 00404_8 via wired memory load program. Control is then given to a temporary check sum verification routine of the utility package which verifies the basic load routine. A utility package routine then reads the balance of the utility package. If the check sum verification is incorrect, the computer will come to an unconditional STOP with AU equal to the tape check sum and AL equal to the load check sum.

<u>b.</u> OPERATING INSTRUCTIONS. – Instructions for the LOAD MODE and AUTOMATIC RECOVERY MODE are described below:

(1) LOAD MODE. - The load mode is used when the manual selection of the wired load program is desired and the computer is in the Master Clear state.

STEP 1. Place paper tape Utility Package in reader (assuming paper tape input).

ORIGINAL

3-17



Paragraph 3-4<u>b</u>(1) 1218 COMPUTER OPERATOR'S SECTION

STEP 2. Master Clear the computer,

STEP 3. Set the LOAD MODE selector button.

STEP 4. Depress START switch. If tape and load check sums agree the computer will come to an unconditional STOP with AU and AL cleared.

(2) AUTOMATIC RECOVERY MODE. - If a fault condition occurs during the running of a program and the AUTO RECOVERY switch is in the UP position, an interrupt will address 002008 (starting address of the wired load program). This locks out all Interrupts and initiates the wired load program which loads in the Utility Package. For paper tape input, the Paper Tape Utility Package must have been placed in the reader (may be positioned on any leader frame) for the recovery to be completed.

NOTE

A fault condition	occurring during t	he running of a	program with the	AUTO
RECOVERY switch in	i the NEUTRAL posit	ion, will cause	a jump to address	00000.
Action will contin	ue as programmed.			

C. WIRED MEMORY PAPER TAPE LOAD PROGRAM.

00200	50	3400	SIL O	Lockout all Interrupts
00201	50	7310	ENTSR 10	Special Register Active-Stack 0
00202	50	7 2 01	ENTICR 1	Set ICR
00203	50	13k	EXF k	Enable EF BUFFER (k is the channel number)
00204	00	0235	0 0235	Terminal Address Designator
00205	00	0234	0 0234	Initial Address Designator
00206	36	0000	ENTBK 0	Initialized word count
00207	42	0003	STRB 0003	Store current word count
00210	70	0002	ENTALK 2	Initialize character count
00211	44	0004	STRAL 0004	Store character count
00212	70	0000	ENTALK O	Clear AL
00213	50	11k	IN k	Read a frame
00214	00	0002	0 0002	Data buffer
00215	00	0002	0 0002	Data buffer
00216	50	21k	SKPIIN k	Active check
00217	34	0216	JP 0216	Active

3-18

ORIGINAL



1218 COMPUTER Paragraph OPERATOR'S SECTION 3-4c 00220 50 4606 LSHAL 6 Assemble word SLSET 0002 Enter character in A 00221 51 0002 00222 63 0225 JPALNZZ 0225 76 code found 00223 0003 ENTAU 0003 Anything in yet 10 00224 NO - still leader 60 0213 JPAUZ 0213 00225 57 0004 ISK 0004 YES - is word formed 00226 0213 JP 0213 NO - read rest 34 00227 45 0236 STRALB 0236 YES - store 00230 56 0233 BSK 0233 Load routine all in 00231 34 0207 JP 0207 NO - continue 00232 34 0240 JP 0240 YES - to Utility routine 00233 00 0147 0 0147 Number of words in load 00234 00 0151 0 0151 Start Reader, Enable Reader Disable Punch codes 00235 ------00236 Not Used _ _ ____ 00237 -------00001 00 0000 00 0000 B-register index reference *00002 00 0000 00 0000 Data buffer *00003 00 0000 00 0000 Word count *00004 00 0000 00 -0000 Character index

* These addresses (normally used as Index Registers) are used as temporary storage areas during the execution of store type instructions contained in the wired memory load.

ORIGINAL

3-19

{Editor's Note: that this bootstrap listing came from a bit savers web site document.}



Appendix B – 1968 UNIVAC Computer Genealogy Chart





Appendix C – SHIPBOARD UNIFORM AUTOMATED DATA PROCESSING SYSTEM⁴

The Shipboard Uniform Automated Data Processing System (SUADPS) serves as a good example of ADP systems used aboard ship. This ADP system is currently in use aboard carriers, tenders, AFSs, and repair ships. The system consists of a computer, an operating system, applications programs, and clerical procedures. The SUADPS was designed to satisfy afloat supply and accounting requirements through maximum automation of routine functions and at the same time provide a wide range of options that can be exercised at the shipboard level. All major files are maintained on magnetic tape, thus greatly reducing manual filing. Input is introduced to the system via the media of punched cards or magnetic tapes and now input from scanners may become possible. The computer system updates the appropriate magnetic tape files through a series of computer runs to reflect quantitative and monetary changes occasioned by the transactions processed. The computer produces outputs reflecting inventory balances, replenishment position, financial and accounting data historical data and exception data requiring supply personnel attention.

EQUIPMENT Shipboard ADP equipment that comprises the SUADPS is identified by both a military designation and a civilian designation. The complete computer system, consists of the components listed below, and carries the military designation AN/UYK-5(V). It is also commonly known by its civilian designation U-1500. When referring to the entire system or individual components m military communications, it is correct to use the military designation. The following equipments comprise the AN/UYK-5(V) computer system: CPU—The central processing point of a computer system. Electronically gathers the input data and produces output. MAGNETIC TAPE UNIT—A high-speed input and output device for the CPU. INPUT/OUTPUT TELETYPEWRITER—A low-speed input/output device for operator communication with the computer. CARD READER-PUNCH-INTERPRETER (CPRI)—Provides the required capability read, punch, and interpret EAM cards. HIGH-SPEED PRINTER-Primary means to of providing printed output from the system.

SUPPORT EQUIPMENT In addition to the primary AN/UYK-5(V) computer components for processing of data, the source documents (input) must be provided by keypunches and verifiers. In some instances interpreters and card sorters are required for off-line or auxiliary jobs (small things not controlled by the computer). This equipment is designated as support equipment. The information contained them is intended to familiarize you with the characteristics and purposes of the various types of support equipment and not to qualify you as an operator.

SYSTEM FILES The SUADPS is a uniform system among the various ship types (SUADPS-207 and SUADPS-AV (207)) in that common routine functions are performed the same manner, and the AN/UYK-5(V) (U-1500) computer system is used by all. There are three basic types of files maintained in the SUADPS: 1) tape files, 2) manual files, and (3) output files. However, system file names vary among the different ships. Certain magnetic tape files which are maintained in both SUADPS-207 (AD/AR, AS, AFS) and SUADPS-AV (207) (CVs, CVNs, LPHs, MAGs) segments are described in the online article.

⁴<u>http://navyadministration.tpub.com/14242/css/14242_196.htm</u> from John Westergren.