decimal to binary, + reverse Instruction Conversion

PROPOSAL

XU-72 COMPUTER SYSTEM

1.0 System Designation: XU-72

2.0 Brief Description:

XU-72 is a large scale general purpose digital computer system whose high speed flexibility and large random access storage capacity are ideally suited to scientific and data processing applications.

3.0 Basis for Proposal of Product:

XU-72 is intended to continue the company's product line in the large scale, general purpose computer field. It will succeed the Univac 1105 in scientific and data processing applications, and will provide a means of program compatibility with the Univac 1105 and Univac 1103A.

The basis on which this proposal is made concerns a System employing gated-pulse circuit logic similar to that used in the 1103A/1105 Systems. Early evaluation indicates that this form of logic permits a relatively easy transition from vacuum tube to transistorized circuitry, and that such circuits can attain operating speeds proposed herein without incurring severe limitations in the circuit layout.

The principle features of the XU-72 which justify initiation of this program are:

a) Use of Solid state elements

A number of advantages, psychological and actual, result from use of solid state elements, such as transistors, in computer design. The first of these is the current notion that vacuum tubex are obsolete. As a result of this and other factors, some government agencies are currently insisting that computers acquired by them employ only solid state elements.

Secondly, the smaller, lower powered, solid state elements lend themselves to ministurization and mechanized assembly techniques with resultant lower operating cost, reduced floor space requirements and lower manufacturing cost.

Thirdly, the anticipated longer life of solid state elements over vacuum tubes will result in longer and more reliable life for computers using these devices. These factors should contribute to lower maintenance cost and longer useful life before obsolescence.

b) Continuation of an accepted Product Line Replacement business.

c) Improvements in Speed and Data Handling Capability over the 1103A Arithmetic speed is increased by factor of 8; logical operations by 2; Core Memory references by ~~6~~ 4: Featured in the system is high capacity random access drum storage, with transfer rate of 34 milliseconds for 4096 36 bit words. Drum storage capacity of up to 2,097,152 words is available.

d) Favorable Competitive Position
The XU-72, with the new features of higher internal speeds, greater storage capacities, improved input/output versatility, direct access to memory, and construction of solid state components, will realize a manufacturing cost of the same magnitude as the Univac 1105.

4.0 Proposed Applications

a) Scientific

Aircraft and Missile Design

Chemical Process Simulation

Military and Naval Tactics

Research

b) Data Processing

Aircraft and Missile Test data reduction

Statistical Data Processing

Inventory Maintenance and Logistics

5.0 System Characteristics

5.1 General

The XU-72 Computer System utilizes the time proven features of the 1103/1105 series of digital computers which, when combined with substantial new features, provides a powerful and competitive large scale computer system. It is particularly suited for complex scientific problems and data reduction problems which require efficient input/output characteristics and very large capacities of internal storage. A means of direct communication between magnetic storage drums, magnetic tape, other peripheral devices and Magnetic Core Storage (MCS), allows uninterrupted internal computation to proceed while data transfers are being made between drum and MCS, and between MCS and input/output devices.

Very efficient mass storage of up to 2,097,152 words provided by multiple "flying head" drums, communicating directly with 32,768 words of MCS, amply provides for the requirement of very large capacities of internal storage. Refer to Figure 1 for the system diagram.

Improved arithmetic operations in both fixed and floating point provide substantial gains in program running time. The internal clock rate, increased by at least a factor of two, also enhances program operating time.

The utilization of all transistorized circuitry not only satisfies the need to provide a computer system constructed with solid state components, but also improves reliability, reduces maintenance and manufacturing costs, reduces power consumption and conserves space.

Self checking of data transfers in the Central Computer and between external storage units and Core Memory will not be supplied. Parity checking will be performed on data transfers between Magnetic Tape and the Tape Control Units.

5.2 Central Computer

The Central Computer governs the sequential acquisition and execution of stored instructions and consists of a Main Control Unit and an Arithmetic Unit. Instructions and operands are transferred between the Main Control Unit and Arithmetic Unit and between the Central Computer, the High Speed Core Memory and the I/O Controller via an exchange (x) register.

5.2.1 Main Control Unit

The instruction repertoire of the 1103A/1105 Computer Systems is retained in the XU-72. Additional instructions may be added to the repertoire; these will be described later in this section.

Following is a listing of the basic instruction repertoire which is presented according to the type classification of the instructions. A typical instruction change should be noted in the External Function (17) instruction assignment. For purposes of program compatibility between the 1103A/1105 and the XU-72, the EF (17) instruction will enter the control word into an interpretive routine which will cause the function to be performed in the XU-72 manner and terminate with the same results as was originally realized on the 1103A/1105. An added EF instruction for the XU-72 repertoire is provided; it is described in the following instruction list.

INSTRUCTION REPERTOIRE

TRANSMIT INSTRUCTIONS

11* Transmit Positive TPuv:#

Replace (v) with (u).

13 Transmit Negative TNuv:

Replace (v) with the complement of (u).

* Octal notation
# Mnemonic notation

12 Transmit Magnitude TMuv:

Replace (v) with the absolute magnitude of (u).

15 Transmit U-address TUuv:

Replace the 15 bits of (v) designated by v_{15} through v_{29}, with the corresponding bits of (u), leaving the remaining 21 bits of (v) undisturbed.

16 Transmit V-address TVuv:

Replace the right-hand 15 bits of (v) designated by v_0 through v_{14}, with the corresponding bits of (u), leaving the remaining 21 bits of (v) undisturbed.

35 Add and Transmit ATuv:

Add D(u) to (A). Then replace (v) with (A_R).

36 Subtract and Transmit STuv:

Subtract D(u) from (A). Then replace (v) with (A_R).

22 Left Transmit LTjkv:

Left circular shift (A) by k places. If j = 0 replace (v) with (A_1); if j = 1 replace (v) with (A_R).

Q-CONTROLLED INSTRUCTIONS

51 Q-controlled Transmit QTuv:

Form in A the number L(Q)(u). Then replace (v) by (A_R).

52 Q-controlled Add QAuv:

Add to (A) the number L(Q)(u). Then replace (v) by (A_R).

53 Q-controlled Substitute QSuv:

Form in A the quantity L(Q)(u) plus L(Q')(v). Then replace (v) with (A_R). The effect is to replace selected bits of (v) with the corresponding bits of (u) in those places corresponding to 1's in Q. The final (v) is the same as the final (A_R).

REPLACE INSTRUCTIONS

21 Replace Add RAuv:

Form in A the sum of D(u) and D(v). Then replace (u) with (A_R).

23 Replace Subtract RSuv:

Form in A the difference D(u) minus D(v). Then replace (u) with (A_R).

27 Controlled Complement CCuv:

Replace (A_R) with (u) leaving (A_L) undisturbed. Then complement those bits of (A_R) that correspond to ones in (v). Then replace (u) with (A_R).

54 Left shift in A LAuk:

Replace (A) with D(u). Then left circular shift (A) by k places. Then replace (u) with (A_R). If u = a, the first step is omitted, so that the initial content of A is shifted.

55 Left Shift in Q LQuk:

Replace (Q) with (u). Then left circular shift (Q) by k places. Then replace (u) with (Q).

SPLIT INSTRUCTIONS

31 Split Positive Entry SPuk:

Form S(u) in A. Then left circular shift (A) by k places.

33 Split Negative Entry SNuk:

Form in A the complement of S(u). Then left circular shift (A) by k places.

32 Split Add SAuk:

Add S(u) to (A). Then left circular shift (A) by k places.

34 Split Subtract SSuk:

Subtract S(u) from (A). Then left circular shift (A) by k places.

TWO-WAY CONDITIONAL JUMP INSTRUCTIONS

46 Sign Jump SJuv:

If $A_{71} = 1$, take (u) as NI. If $A_{71} = 0$, take (v) as NI.

47 Zero Jump ZJuv:

If (A) is not zero, take (u) as NI. If (A) is zero, take (v) as NI.

44 Q-Jump QJuv:

If $Q_{35} = 1$, take (u) as NI. If $Q_{35} = 0$, take (v) as NI. Then, in either case, left circular shift (Q) by one place.

ONE-WAY CONDITIONAL JUMP INSTRUCTIONS

41 Index Jump IJuv:

Form in A the difference D(u) minus 1. Then if $A_{71} = 1$, continue the present sequence of instructions; if $A_{71} = 0$, replace (u) with (A_R) and take (v) as NI.

42 Threshold Jump TJuv:

If D(u) is greater than (A), take (v) as NI; if not, continue the present sequence. In either case, leave (A) in its initial state.

43 Equality Jump EJuv:

If D(u) equals (A), take (v) as NI; if not, continue the present sequence. In either case leave (A) in its initial state.

ONE-WAY UNCONDITIONAL JUMP INSTRUCTIONS

45 Manually selective Jump MJjv:

If the number j is zero, take (v) as NI. If j is 1, 2, 3, 4, 5, 6, or 7, and the correspondingly numbered MJ selecting switch is set to "jump", take (v) as NI; if this switch is not set to "jump", continue the present sequence.

37 Return Jump RJuv:

Let y represent the address from which CI was obtained. Replace the right-hand 15 bits of (u) with the quantity y plus 1. Then take (v) as NI.

14 Interpret IP:

Let y represent the address from which CI was obtained. Replace the right hand 15 bits of (F_1) with the quantity y + 1. Then take (F_2) as NI.

STOP INSTRUCTIONS

56 Manually selective Stop MSjv:

If j = 0, stop computer operation and provide suitable indication. If j = 1, 2, 3, 4, 5, 6, or 7, and the correspondingly numbered MS selecting switch is set to "stop", stop computer operation and provide suitable indication. Whether or not a stop occurs, (v) is NI.

57 Program Stop PS --

Stop computer operation and provide suitable indication.

EXTERNAL EQUIPMENT INSTRUCTIONS

17 Interpretive External Function EF-v:

Select a unit of external equipment and perform the function designated by (v).

76 External Read ERjv:

If j = 0, replace the right-hand 8 bits of (v) with (1OA).

77 External Write EWjv:

If j= 0, replace (1OA) with the right-hand 8 bits of (v). Cause the previously selected unit to respond to the information in 1OA.

61 PRint PR-v:

Replace (TWR) with the right-hand 6 bits of (v). Cause the typewriter to print the character corresponding to the 6-bit code.

63 PUnch PUjv:

Replace (HPR) with the right-hand 6 bits of (v). Cause the punch to respond to (HPR). If $j = 0$, omit seventh level hole; if $j = 1$, include seventh level hole.

External Function EFuv:

Select a unit of external equipment and perform the function designated by u and (v).

SEQUENCED INSTRUCTIONS

71 MultiPly MPuv:

Form in A the 72-bit product of (u) and (v), leaving in Q the multiplier (u).

72 Multiply Add MAuv:

Add to (A) the 72-bit product of (u) and (v), leaving in Q the multiplier (u).

73 Divide DVuv:

Divide the 72-bit number (A) by (u), putting the quotient in Q, and leaving in A a non-negative remainder R. Then replace (v) by (Q). The quotient and remainder are defined by: $(A)_i = (u) \cdot (Q) + R$, where $0 \leq R < |(u)|$. Here $(A)_i$ denotes the initial contents of A.

74 Scale Factor SFuv:

Replace (A) with D(u). Then left circular shift (A) by 36 places. Then continue to shift (A) until $A_{34} \neq A_{35}$. Then replace the right-hand 15 bits of (v) with the number of left circular shifts, k, which would be necessary to return (A) to its original position. If (A) is all ones or zeros, k = 37. If u is A, (A) is left unchanged in the first step, instead of being replaced by $D(A_R)$.

75 RePeat RPjnw:

This instruction calls for the next instruction, which will be called NIuv, to be executed n times, its "u" and "v" addresses being modified or not according to the value of j. Afterwards the program is continued by the execution of the instruction stored at a fixed address F_1. The exact steps carried out are:

(a) Replace the right-hand 15 bits of (F_1) with the address w.

(b) Execute NIuv, the next instruction in the program, n times.

(c) If j = 0, do not change u and v.

If j = 1, add one to v after each execution.

If j = 2, add one to u after each execution.

If j = 3, add one to u and v after each execution.

The modification of the u-address and v-address is done in UAK and VAK. The original form of the instruction in storage is unaltered.

(d) On completing n executions, take (F_1), as the next instruction. F_1 normally contains a manually selective jump whereby the computer is sent to w for the next instruction after the repeat.

(e) If the repeated instruction is a jump instruction, the occurrence of a jump terminates the repetition. If the instruction is a Threshold Jump or an Equality Jump, and the jump to address v occurs, (Q) is replaced by the quantity j, (n-r), where r is the number of executions that have taken place.

FLOATING POINT INSTRUCTIONS

64 Add FAuv:

Form in Q the normalized rounded packed floating point sum $(u) + (v)$.

65 Subtract FSuv:

Form in Q the normalized rounded packed floating point difference $(u) - (v)$.

66 Multiply FMuv:

Form in Q the normalized rounded packed floating point product $(u) \circ (v)$.

67 Divide FDuv:

Form in Q the normalized rounded packed floating point quotient $(u) \overset{\circ}{\div} (v)$.

01 Polynomial Multiply FPuv:

Floating add (v) to the floating product $(Q)_i \circ (u)$, leaving the packed normalized rounded result in Q.

02 Inner Product FIuv:

Floating add to $(Q)_i$ the floating product $(u) \circ (v)$ and store the rounded normalized packed result in Q. This instruction uses MC location $F_4 = 00003$ for temporary storage, where $(F_4)_f = (Q)_i$.

03 Unpack UPuv:

Unpack (u), replacing (u) with $(u)_M$ and replacing $(v)_c$ with $(u)_c$ or its complement if (u) is negative. The characteristic portion of $(u)_f$ contains sign bits. The sign portion and mantissa portion of $(v)_f$ are set to zero.

NOTE: The subscripts M and C denote the mantissa and characteristic portions.

04 Normalize Pack NPuv:

Replace (u) with the normalized rounded packed floating point number obtained from the possibly unnormalized mantissa in $(u)_i$ and the biased characteristic in $(v)_c$.

NOTE: It is assumed that $(u)_i$ has the binary point between u_{27} and u_{26}; that is, that $(u)_i$ is scaled by 2^{-27}.

05 Normalize Exit NEj-:

If j = 0, clear the normalize exit flip-flop (designated NFF); if j = 1, set NFF to 1.

(a) When NFF is set to 1, the normalize, round, and pack sequence of each of the subsequent floating point instructions is omitted. For FA, FS, or FD, the result is a quasinormalized double-length mantissa in A, with the units position at a_{61} and sign bits to the left; while for FM the units position is at a_{62} with sign bits to the left.

(b) If NFF is set to 1, it will remain set until it is cleared by an NEj-instruction.

(c) NFF must be cleared for the FP, FI, and the NP instructions.

TERMINOLOGY

WORD LENGTH

36 binary digits (bits)

INSTRUCTION WORD

oc	u	v
6 bits	15 bits	15 bits

oc Operation code

u First execution address

v Second execution address

For some of the instructions, the form jn or jk replaces the u address; for others the form k replaces the v address.

j one-digit octal number modifying the instruction

n four-digit octal number designating number of times instruction is to be performed.

k seven-digit binary number designating the number of places the word is to be shifted to the left.

ADDRESS ALLOCATIONS (OCTAL)

MC	00000-37775	16,382* 36-bit words
Q	37776	1 36-bit word
A	37777	1 72-bit word
MC	40000-77777	16,384 36-bit words

* The full 16,384 words of this MCS bank is available for data transfers between MCS and external devices.

FIXED ADDRESSES

F1	00000 or 40001
F2	00001
F3	00002
F4	00003

ARITHMETIC SECTION REGISTERS

A	72-bit accumulator with shifting properties
A_R	right hand 36 bits of A
A_L	left hand 36 bits of A
Q	36-bit register with shifting properties
X	36-bit exchange register

NOTE: Brackets denote contents of, for example,

(A) means contents of A (72-bit word in A);

(Q) means contents of Q (36-bit word in Q)

WORD EXTENSION

D(u) 72-bit word whose right hand 36 bits are the word at address u, and whose left hand 36 bits are the same as the left most bit of the word at u.

S(u) 72-bit word whose right hand 36 bits are the word at address u, and whose left hand 36 bits are zero.

D(Q) 72-bit word - right hand 36 bits are in register Q, left hand 36 bits are same as left most bit in register Q.

S(Q) same as D(Q) except left 36 bits are zero.

$D(A_R)$, $S(A_R)$ are similarly defined.

L(Q) (u) 72-bit word - left hand 36 bits are zero, right hand 36 bits are the bit by bit product of corresponding bits of (Q) and word at address u.

L(Q') (v) 72-bit word - left hand 36 bits are zero, right hand 36 bits are the bit by bit product of corresponding bits of the complement of (Q) and word at address v.

An increase in the versatility and efficiency in the XU-72 will be realized by the inclusion of certain additional instructions in the repertoire. Such instructions are given in the following list, ~~and may be added according to their~~ economic feasibility. The complete instruction repertoire for the XU-72 must be firm three months after the initiation of the development program.

(1) Generalized Repeat - This will add an increment from a 30 bit static register to UAK or VAK or both or neither depending on the value of J.

(2) Internal Function Command - This command will be used to set up certain conditions in various registers and also to read these registers. The UAK part will contain either information bits or an adress. As As an example, the internal function command will be used to load the increment register for the generalized repeat. As another example, the internal function will be used to read the error register.

(3) Increment Next Instruction - This command which is a command code and/or address modifier, will add the (u) to PCR of the next instruction. Such modifiers can be obtained from any memory location.

(4) Add Add Command - This command will add (u) to A and then add (v) to A.

(5) Magnitudinal Threshold Jump - This command will compare the absolute value of (u) with the absolute value of A, and if (u) is greater than A jump to v for the next instruction.

(6) Polynomial Multiply - Multiply the contents of A by (u), then shift right the number of positions indicated by the polynomial shift register then add (v) to A.

(7) Split Threshold Jump - This command will be executed in the same manner as the threshold jump, except that the split subtract and split add sequence will be used in place of the double extension subtract and add. ?

(8) Right Shift - In A shift (u) in A ℓ places to the right and store A back in (u).

(9) v to u command - transmit V of (u) to U of (v).

(10) u to v command - transmit U of (v) to V of (u).

(11) Split Equality Jump - This is the same as the equality jump, except that split subtract and split add sequences are used in place of the double length subtract and double length add. ?

4μs 2μs

The execution time of instruction will be reduced by a factor of at ? least two through the use of a 1-megacycle clock rate (the arithmetic sequence speeds are increased by a factor of 8 or more). The clock rate will be derived from a master oscillator whose frequency will be adjustable for marginal checking purposes.

Every effort will be made to preserve compatibility with existing 1103A/1105 programs. In the area of storage address, however, some modifications of programs will be required. The Q and A addresses will be changed from 31000 through 31777, to 37776 for the Q register and the accumulator will be changed from 32000 through 37777 to 37777. This change is necessary to eliminate use of a full 4096 word group for the two arithmetic register addresses. The addresses 37776 and 37777 in magnetic core storage will be usable in block transfers but will not be accessible by program instructions. The addresses 40000 through 77777 which at present are the drum addresses, will now address a second bank of 16,384 words of magnetic core storage. Therefore, we will have program compatibility with the exception that the A and Q addresses must be changed. This would be similar to the change-over from 1103 to 1103A.

5.2.2 Arithmetic Unit

Fixed and Floating Point Arithmetic facilities will be standard. Basic 2. arithmetic times will be reduced by factor of 8 or more. This time improvement will be obtained primarily through the use of a 4 megacycle arithmetic section. Additional time improvements will result from changes in arithmetic processes such as grouping and translation of accumulator stages to reduce borrow propagation time.

5.3 High Speed Core Memory

The high speed storage of the XU-72 will be a 36-bit, parallel, coincident current, magnetic core memory arranged in two banks of 16,384 words each. Access time for any address will be 1.5 microsecond for single storage references and cycle time of the memory will be 3 microseconds.

The addresses will range from 00000 through 37777 and 40000 through 77777 for the two core banks, respectively. The lower order range of addresses will be those presently used in the 1103A/1105 computers, for 3-4096 word core banks and the A and Q registers. Addresses 37776 and 37777 will be reserved for A and Q registers in place of the address group 31000 through 37777 now reserved for these registers. The higher range of addresses, 40000 through 77777 will be those presently used for magnetic drum storage in the 1103A/1105 computers.

Either bank of memory may be accessed from the Central Computer or from the I/O Controller. These two banks of memory are independent of each other and may be accessed simultaneously via a combination of the above mentioned controls. ?

5.4 Input/Output Controller

The I/O controller receives commands from the Central Computer via an External Function Instruction (EF) and without further program intervention, controls the transfer of data between High Speed Core Memory and any of the Input/Output units.

On transfers between Magnetic Drum Storage and High Speed Core Memory, a single EF instruction is sufficient to transfer 4096 words between designated drum and core storage address groups.

On transfers between Magnetic Tape Storage or Punched Card Equipment and High Speed Core Memory, a single EF instruction is sufficient to transfer up to 4095 blocks of up to 120 words each. The designated number of blocks are transferred without further program intervention, started in the core address given in the EF instruction. <

One Magnetic Drum ~~Storage~~ Control Unit, up to 4 magnetic tape control units, one punched card unit or one line printer may be controlled by the I/O controller. Improve

5.5 Magnetic Drum Storage

The Magnetic Drum System provides the XU-72 with a 16 physical large volume, rapid access memory in optional capacities of up to 2,097,152 - 36 bit words. The proposed drum storage system consists of a drum control unit and from 2 to 8 high density flying head drums. Each of the 2 to 8 physical drums will be sub-divided into 16 logical drums; each logical drum has a capacity of 16,384 4 words. The maximum optional system would provide storage capacity equal to. 128 of the 1103A Computer drums. <

Data transfers between magnetic drum and magnetic core storage will be in blocks of 4096 words only. A single External Function (EF) instruction will initiate the block transfer and will be made up of the following:

a) 1 bit selecting Magnetic Drum Control

b) 7 bits, selecting logical drum

c) 2 bits, selecting one of 4-4096 word drum groups

d) 3 bits, selecting core bank and 4096 word group

e) one bit designating direction of data transfer, drum to core or core to drum.

The flying head drum used will be approximately 22" in diameter and rotate at 1750 RPM. The recording density, 235 ppi, will then give a word rate of 500 KC or a period of 2 microseconds between words. In order to match the core speed for efficient block transfers, an interlace of 4 will be used in all drum references. Words may then be transferred in or out of consecutive addresses since the drum word interval will be 8 microseconds and the magnetic core storage cycle time will be 6 microseconds. Transfers between drum and core will commence at any angular position of the drum and thus will always require one drum revolution (34 milliseconds) for a 4096 word block transfer.

5.6 Magnetic Tape Storage

The Magnetic Tape Storage Units will employ solid state, high performance, compatible tape handlers and will have the operational characteristics similar to the 1103A/1105 Magnetic Tape System.

Programmable options include recording or reading variable block length and fixed block length (UNIVAC) modes and reading in the continuous data input mode. Recording densities of 128 ppi for Univac compatibility, 250 ppi, or higher, for high performance operation, and tape speeds of 100 ips and 150 ips are supplied. <

1090

A maximum of 12 tape handlers may be controlled by one tape control unit. Up to four tape control units may be employed with a total complement of 48 tape handlers. In the maximum system, four tape handlers may be operated simultaneously, moving more than one million bits per second.

6 tape

5.7 Other Peripheral Devices

Displays

Engineering design for these features is not included in the attached manpower and cost estimate.

5.7.1 Punched Card Equipment

Punched card reader may be attached on line to an Input/Output channel. Card rate will be 300, or more, cards per minute.

Too slow

A card punch may also be attached to an Input/Output channel in an on line manner and will operate at a rate of 150 cards per minute.

not l
consist

5.7.2 Line Printer

A line printer, operating at 600, or more, lines per minute may be connected to an Input/Output channel.

5.8 Supervisory Desk Console

Indicator lights and controls for operation and maintenance of the computer will be located on a desk type console. Location of the desk will be flexible to accommodate variations in arrangement of the computer cabinets.

Human engineering will be applied in the Console design so as to minimize the physical size but not limit the operational control.

5.9 Physical Characteristics

5.9.1 Cabinet and Floor Layout

The XU-72 will be housed in modular cabinets, each approximately 60" wide, 30" deep and 72" high. Exclusive of tape handlers, drums, operators console and other peripheral devices and assuming use of two tape channels, four cabinets will be required. Tentative assignment of sections of the machine to the cabinets is as follows:

1. Central Computer
2. Two 16,384 word banks of High Speed Core Memory
3. Magnetic Drum Control and I/O Controller
4. Two Magnetic Tape Channels.

The 60 inch cabinets will consist of two identical sections, each capable of mounting 750 chassis cards on 3 panels. The panels will be arranged in a 3-deep configuration with the outer 2 panels hinged for access to the center. Air intakes, filters and blowers will be housed in the base of the cabinet and room air will be forced up through the chassis card array and exhausted from the top of the cabinet. Each cabinet will contain all power supplies required for use within the cabinet so that additions or deletions to the system will always be self powered. Power supplies will be located at the top of the cabinets to minimize generation of heat near the electronic circuit components.

Interconnection between sections of the system will be made through the base of the cabinets. The cabinets will be mounted on a shallow false floor to conceal the cable connections. The initial floor layout will have all cabinets abutting each other in a single row to minimize cable lengths. Magnetic Drums and tape handlers will be arranged in rows parallel to the electronic system cabinets. Other configurations will be possible, depending on the limitations imposed by cable length restrictions. With a full complement of tape handlers (assume Uniservo II) and magnetic drums, the XU-72 system will require a floor area approximately 20 x 40 feet, not including maintenance facilities.

5.9.2 Power and Cooling Requirements

Individual power supplies, located in each cabinet, will supply the approximately 2 KW of DC power required by the electronic circuits in the cabinet. Additional DC supplies will be located in Magnetic Drum Control to supply approximately 4 KW for a maximum of 8 drums, and in each Magnetic Tape Control Unit, to provide approximately 36 ~~3~~ KW for a maximum of 12 tape handlers.

Primary power for the DC supplies will be obtained from a motor-alternator set generating 400 cycle, single phase power. The M-A set will serve to isolate line voltage transients and provide a 400 cycle source to enable use of compact, efficient DC power supplies. An alternator capacity of approximately 20 KVA would be required.

Additional AC power (230V, 3 phase, 60 cycle) will be required at a rate of 1 KVA per drum and 1.5 KVA per tape handler. The 60 cycle power will be controlled by a stabiline voltage regulator.

All electronic circuit components will be cooled by forcing air at room temperature through the cabinets. Temperature control must be provided by the customer to keep room air temperature in the range of 65° to 80°F. Use of Uniservo II will require cooling water at a rate of 3 1/2 gallons per minute for each tape handler. Water temperature must be from 45° to 50°F and in the pressure range of 13 to 45 PSI.

6.0 Auxiliary Equipment

An auxiliary equipment desk provided with the XU-72 will contain facilities for reading and punching of paper tape and a typewriter for monitoring computer replies. Capabilities of the equipment will be equivalent to that now provided for 1103A and 1105 Computers and will include ? paper tape reader - 200 cps minimum; paper tape punch - 60 cps minimum; and typewriter - 10 cps. Higher performance units will be included if available early in the development program.

7.0 Maintenance Facilities

Provision of facilities for efficient maintenance will be a major factor in the design of XU-72. Some of the features of XU-72 which will speed and simplify system maintenance are the following:

Chassis types will be held to a minimum and adequate spares of each type recommended for each installation.

The chassis design will bring strategic test points to the outer edge of the dard, allowing easy access for electrical checking.

To a large extent, each cabinet will contain its own maintenance indicators and controls, and will be capable of independent operational test.

Marginal checking facilities will be provided. Varying of clock frequency appears the most feasible means of marginal test at this time.

Adequate test routines will be generated as part of the XU-72 development.

10.0 Cost

10.1 Development Cost

The engineering activity in this program includes all design, development, product engineering, drafting and publications efforts required to accomplish the XU-72 System, as described in this proposal. Necessary engineering supervision costs required to construct the XU-72 prototype and all engineering test costs involved are included. The Manpower/Cost Extension Chart, attached, provides the engineering cost information. Coats for drafting publications, manufacturing required for engineering test models and mathematician/programmers, are listed in the breakdowns.

Costs which will be incurred as a result of programming services to System users, generation of auto-coding programs and the building of library routines are in the category of Sales Support and appear elsewhere.

10.2 System Manufacturing Cost

The manufacturing cost of the XU-72 System, described herein, and listed in the Manufacturing Cost Summary, is derived from a cost estimate provided by the Manufacturing Cost Estimating Department on December 3, 1958. The engineering prototype and the early production models are estimated according to the attachment, with these exceptions, viz, the engineering prototype test costs are included in the development cost and the installation costs may not be applicable, therefore, the total manufacturing should be appropriately reduced.

Manufacturing Cost Summary

Three XU-72 Cabinets @ $3960 (includes cabinet space for the Central Computer, Drum Control and I/O Controller, and Tape Control	$11,880.00

3100 Printed Circuit Cards @ $70 (for the above)	$217,000.00
Back Panel Wiring (for the above)	31,315.00
Two 16,384 Word Core Memories @ $152,821.	305,642.00
Control Console	34,740.00
Inter-Cabinet Cabling	11,968.00
False Flooring	7,367.00
Twenty Tape Handlers @ $6809 (Uniservo II cost used as reference)	136,180.00
Motor Alternator and Switchgear	4,560.00
Stabiline Regulator	1,432.00
Systems and Unit Test	56,160.00
Installation	23,921.00
Eight Drum Storage Units @ $15,974.	127,792.00
TOTAL	$969,957.00

PROPOSAL

COMPANY CONFIDENTIAL

ERA-891 – V (REV. 3-58)

LABOR CLASSIFICATION	GRADE	1st QTR 59			2nd QTR 59			3rd QTR 59			4th QTR 59			1st QTR 60			2nd QTR 60		
		APR ~~OCT~~	MAY ~~NOV~~	JUNE ~~DEC~~	JULY ~~JAN~~	AUG ~~FEB~~	SEPT ~~MAR~~	OCT ~~APR~~	NOV ~~MAY~~	DEC ~~JUNE~~	JAN ~~JULY~~	FEB. ~~AUG~~	MAR ~~SEPT~~	APR ~~OCT~~	MAY ~~NOV~~	JUNE ~~DEC~~	JULY ~~JAN~~	AUG ~~FEB~~	SEPT ~~MAR~~
Electrical Eng.		27	28	28	28	28	28	28	28	28	27	26	26	23	21	20	18	16	15
Mechanical Eng.		5	6	6	6	6	6	6	6	6	4	4	4	4	3	3	3	3	3
Technicians		12	16	16	16	16	16	16	16	16	15	15	15	13	13	13	11	10	10
Drafting		4	6	8	11	13	15	18	18	18	18	19	19	19	19	19	19	17	17
Specifications		1	2	2	2	2	2	2	2	1	1	1	1	1					
Ind. Design		1	1	1	1	.5	.5	.5	.5	.5	.3	.3	.3						
Publications		1	2	3	4	4	4	4	5	5	6	6	8	8	8	8	8	8	8
ENG. LABOR		$ 98,700.00			$ 113,500.00			$ 119,675.00			$ 114,176.00			$ 101,425.00			$ 85,425.00		
MFG. LABOR																			
MISC. DIR. CHGS.		5,000.00			5,000.00			5,000.00			2,000.00			2,000.00			4,000.00		
BURDEN		103,635.00			119,175.00			125,659.00			119,886.00			106,497.00			89,697.00		
MATERIALS		50,000.00			45,000.00			15,000.00											
QUARTERLY TOTAL		$ 257,335.00			$ 282,675.00			$ 265,334.00			$ 236,062.00			$ 209,922.00			$ 179,122.00		
CUMULATIVE TOTAL		$ 257,335.00			$ 540,010.00			$ 805,344.00			$ 1,041,406.00			$ 1,251,328.00			$1,430,450.00		

DATE PREPARED 4-7-59 REVISION NO:	MANPOWER/COST EXTENSION CHART	PROJECT: XU72 COMPUTER - RECAP

COMPANY CONFIDENTIAL

PROPOSAL

COMPANY CONFIDENTIAL

ERA-891 – V (REV. 3-58)

LABOR CLASSIFICATION	GRADE	3rd QTR 60			4th QTR 60			QTR			QTR			QTR			QTR		
		~~APR~~	~~MAY~~	~~JUNE~~	~~JULY~~	~~AUG~~	~~SEPT~~	~~OCT~~	~~NOV~~	~~DEC~~	~~JAN~~	~~FEB~~	~~MAR~~	~~APR~~	~~MAY~~	~~JUNE~~	~~JULY~~	~~AUG~~	~~SEPT~~
		OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUNE	JULY	AUG	SEPT	OCT	NOV	DEC	JAN	FEB	MAR
Electrical Eng		14	14	14	8	7	7												
Mechanical Eng		2	2	2	1	1	1												
Technicians		10	10	10	10	10	10												
Drafting		15	14	14	12	9	7												
Specifications																			
Ind. Design																			
Publications		7	7	7	7	7	7												

						GRAND TOTALS
ENG. LABOR	$ 72,925.00	$ 52,125.00				757,951.00
MFG. LABOR						
MISC. DIR. CHGS.	2,000.00	5,000.00				30,000.00
BURDEN	76,571.00	54,731.00				795,851.00
MATERIALS						
QUARTERLY TOTAL	$151,496.00	$ 111,856.00				110,000.00
CUMULATIVE TOTAL	$1,581,946.00	$1,693,802.00				1,693,802.00

DATE PREPARED 4-7-59
REVISION NO:

MANPOWER/COST EXTENSION CHART

PROJECT: XU72 COMPUTER - RECAP

COMPANY CONFIDENTIAL

COMPANY CONFIDENTIAL

ERA-891-V (REV. 3-56)

		3 rd Year			4th Year			5 th Year			6th Year			7th Year			8th Year		
LABOR CLASSIFICATION	GRADE	QTR			QTR			QTR			QTR			QTR			QTR		
		~~APR~~ ~~OCT~~	~~MAY~~ ~~NOV~~	~~JUNE~~ ~~DEC~~	~~JULY~~ ~~JAN~~	~~AUG~~ ~~FEB~~	~~SEPT~~ ~~MAR~~	~~OCT~~ ~~APR~~	~~NOV~~ ~~MAY~~	~~DEC~~ ~~JUNE~~	~~JAN~~ ~~JULY~~	~~FEB~~ ~~AUG~~	~~MAR~~ ~~SEPT~~	~~APR~~ ~~OCT~~	~~MAY~~ ~~NOV~~	~~JUNE~~ ~~DEC~~	~~JULY~~ ~~JAN~~	~~AUG~~ ~~FEB~~	~~SEPT~~ ~~MAR~~
Elect. Eng.			5.0			2.5			1.0			1.0			1.0			1.0	
Mech. Eng.			2.0			1.0			.5			.5			.2			.2	
Engr. Ass't.			1.0			1.0			1.0			1.0			0			0	
Tech.			5.0			2.0			1.0			1.0			1.0			1.0	
Draft.			5.0			4.0			2.0			2.0			1.0			1.0	
Pub. Writer			2.83			.7			.5			.5			.1			.1	
ENG. LABOR		148,450.00			65,190.00			35,560.00			35,560.00			19,840.00			19,840.00		
MFG. LABOR																			
MISC. DIR. CHGS.																			
BURDEN		153,010.00			67,210.00			36,660.00			36,660.00			20,200.00			20,200.00		
MATERIALS																			
QUARTERLY TOTAL		301,460.00			132,400.00			72,220.00			72,220.00			40,040.00			40,040.00		
CUMULATIVE TOTAL		301,460;00			433,860.00			506,080.00			578,300.00			618,340.00			658,380.00		

DATE PREPARED: 2-24-59
REVISION NO 1. 4-2-59

MANPOWER/COST EXTENSION CHART

PROJECT: XU72 Continuation Engineering

COMPANY CONFIDENTIAL

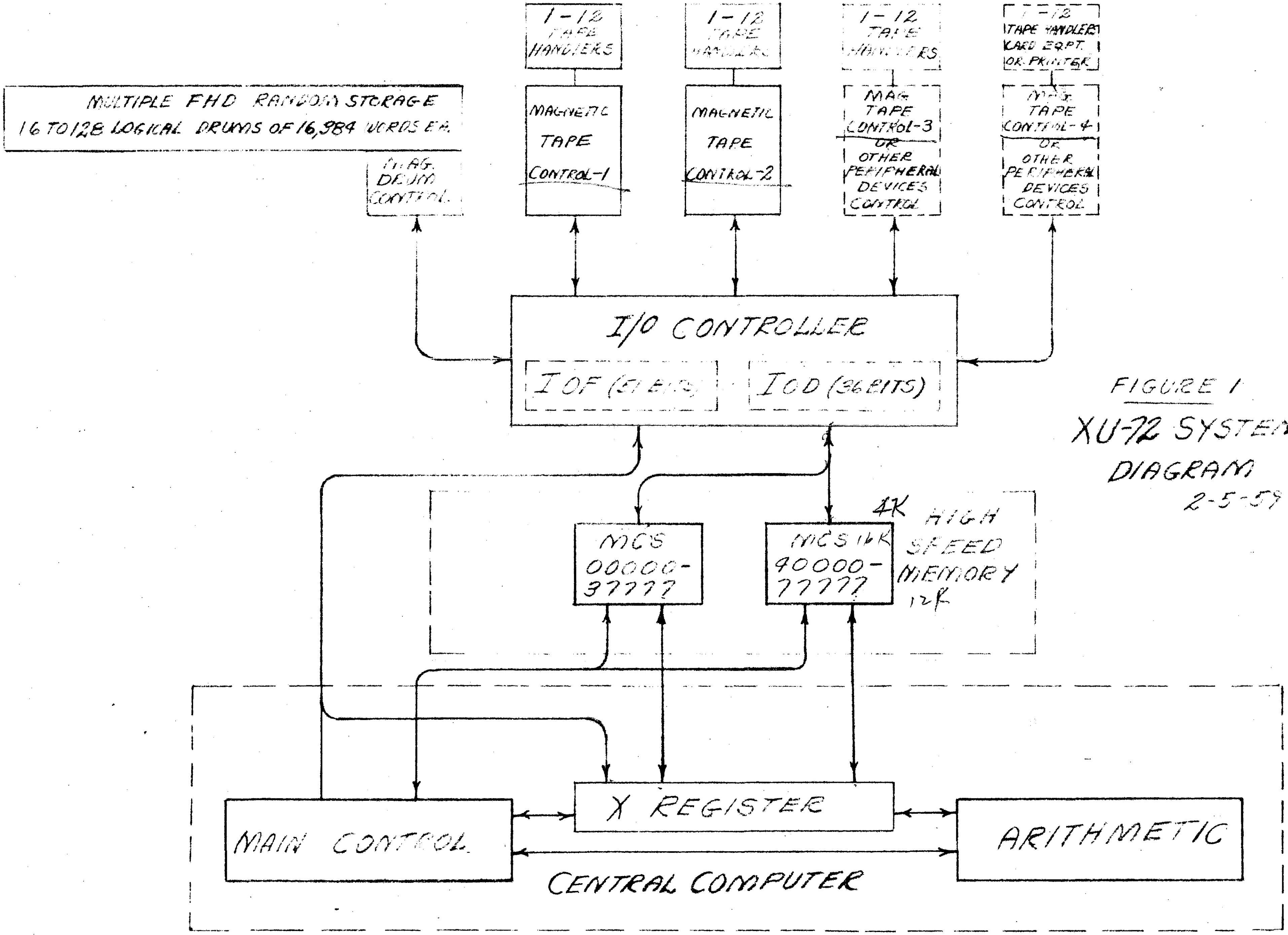
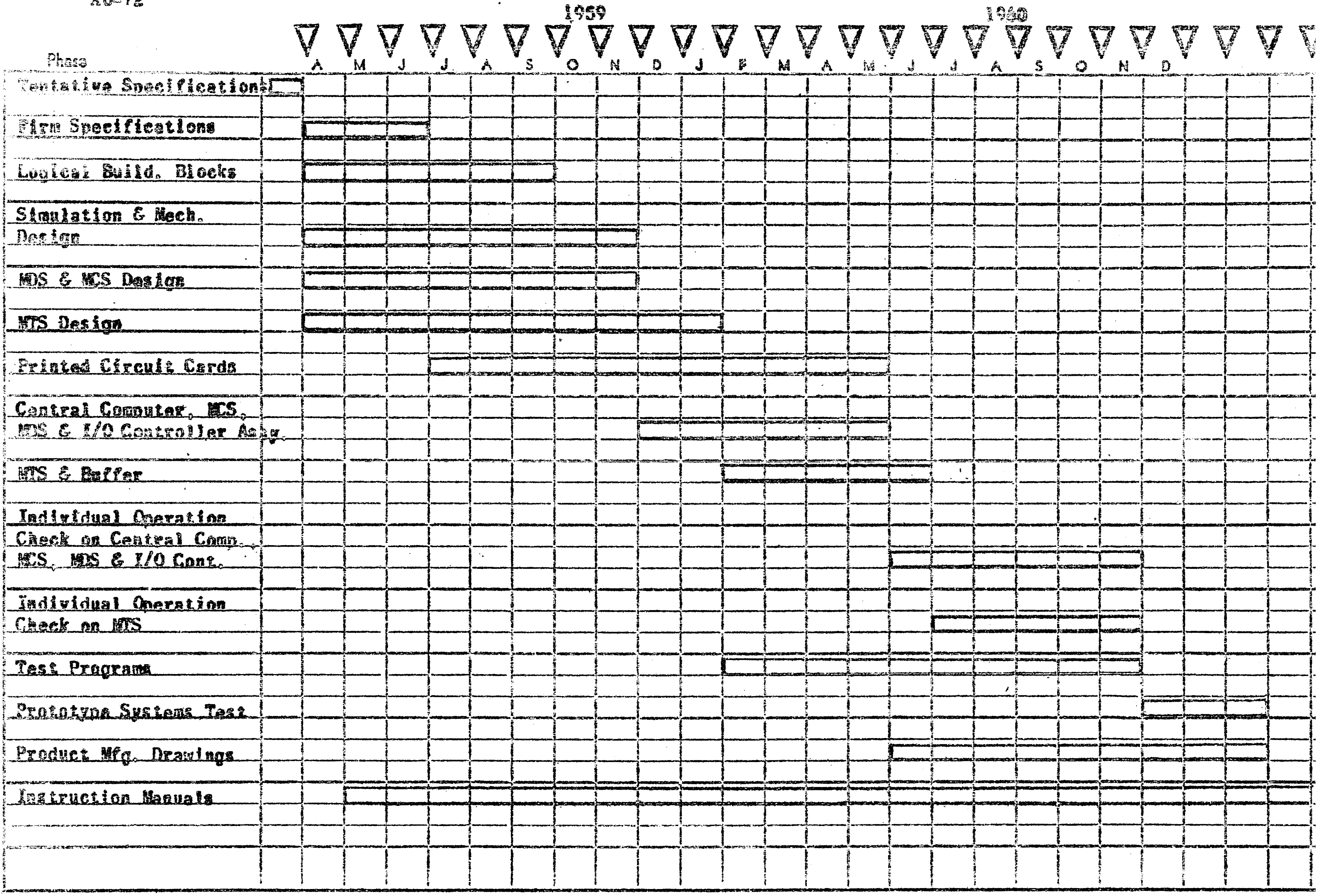


FIGURE 1
XU-72 SYSTEM DIAGRAM
2-5-59

Project #XD-72

PROJECT STATUS CHART

Revision____



A-1120

Chart assumes a project initiation date of April 1, 1959

III. Features

The XU-72, as presently planned, includes the following features:

- High speed, large capacity, and system flexibility provide a powerful tool for solving contemporary scientific and data-processing problems. Desirable features of the UNIVAC Scientific computers are retained, such as parallel binary arithmetic on 36 bit words and two-address logic, while important performance factors are greatly improved at little increase in cost of the system.

- Program compatibility between successive models of UNIVAC Scientific computers has been preserved, as in the past. An 1103A program can be run on the XU-72 with no more difficulty than an 1103 program can be run on the 1103A.

- Basic arithmetic speeds are eight times faster than in the UNIVAC 1100 series. This improvement is obtained by using the new transistors now available with a 4 megacycle clock in place of the 1100 series vacuum tubes and a 500 kilocycle clock.

- Internal magnetic core memory is four times as fast as in the UNIVAC 1100 series, with over twice the capacity. This improvement provides a 3 microsecond cycle time in each of two 16,384 word memory segments. When faster, less expensive memories are available, they can be incorporated in the XU-72 without major redesign of the rest of the system.

- The greatest performance gain is in the XU-72's magnetic drum subsystem. Storage capacity is 128 times larger than in the 1103A. Average and maximum latency times (for transfers between drum and core memory) have been reduced a thousand-fold - from milliseconds to microseconds. A total number of 2,097,152 words are arranged in 512 groups of 4096 words each. Any of these 4096 word groups can be transferred to or from the core memory in 34 milliseconds.

- Systems flexibility is considerably improved by providing eight direct-access channels to two independent memory segments, in addition to the drum channel. One core memory segment may be transferring at maximum speed to or from the drum subsystem (a word every 8 microseconds) while simultaneously up to 8 magnetic tape units time-share the same 16,384 word core memory segment with a running program. In addition, the other core segment can be utilized for input-output and computations.

* Transistorized design permits a substantial reduction in installation requirements. The XU-72, with eight drums and 24 Uniservo II Tape Units, will occupy 800 square feet of floor space, versus 2,760 square feet for the 1105 with one drum and 24 Uniservo II's. The XU-72 system will use less than one-third of the power necessary to operate the 1105.

Minimum and Maximum System

Minimum Operational Systems	Maximum Operational System
Arithmetic & Control	Arithmetic & Control
16,384 Word Core Memory	32,768 Word Core Memory
No Drum	8 Drums
1 Input-Output Channel	8 Input-Output Channels
No Buffer	8 Buffers
1 Tape Control Unit	8 Tape Control Units
8 Magnetic Tape Units	12 Magnetic Tape Units per Channel
Typewriter, Paper Tape System	Typewriter, Paper Tape System
No Card Unit	1 Card Read/Punch Unit
	Off-line peripheral equipment

IV. Competitive Situation

Sperry Rand has an established position in the commercial market for large scientific computers. That this market exists, is expanding, and is a desirable one in which to maintain a leading position, is shown both by the volume of business and the announcement by IBM of a new computer, the IBM 7090, tailored for this market.

A comparative analysis of the advertised features of the IBM-7090 with the same features of the XU-72 shows the following:

IBM 7090	XU-72
Transistorization, providing simplified installation and low maintenance cost.	The XU-72 is transistorized, and offers simple installation and low maintenance cost.
High speed computation (parallel binary, built in floating point) at lower cost then with decimal arithmetic.	The XU-72 offers a new concept of large capacity, very fast access internal memory, based on the Floating Head Drum, and its computing speeds are comparable to the 7090.
Program compatibility. IBM 709 programs will run unchanged; a 2000 instruction compatibility subroutine permits running IBM 704 programs.	The XU-72 offers program compatibility with the various models of UNIVAC Scientific computers. An 1103A program can be run on the XU-72 as readily as an 1103 program can be run on an 1103A. The XU-72 is much more compatible program-wise with the UNIVAC 1100 series than is the 7090 with the 704.

IBM's 704 versus Sperry Rand's 1103-1103A IBM 704 installations began in December, 1955, a year before the first UNIVAC 1103A. About a hundred 704's have been installed, plus 18 older 701's. Production and installation figures for the 1103, 1103A and 1105 computers are as follows:

	Sold	Rented	Stored or Under conversion	Total
Winter, 1959	22	9	4	35
Fall, 1959	24	13	4	41

IBM 7090 versus Sperry Rand XU-72 The 7090 is scheduled for delivery starting in December, 1959. For profitability reasons IBM would prefer to postpone replacement of 704's until at least 1962 or later, so that the first two years of 7090 production can be expected to aim at (1) retirement of 701's (2) replacement of UNIVAC Scientific computers, where the expanded work load beyond system capability offsets the disadvantage of a change in vendors; (3) new customers.

Sperry Rand should be able to deliver production quantities of XU-72's before late 1961. We may reasonably assume that the majority of our preent customers will stay with us because the prospect of eventually handling their expanding workloads without major systems revisions should be a sufficiently persuasive factor. In addition, the expanded drum storage offers qualitative advantages to new customers, advantages which IBM can incorporate in the 7090 only with great difficulty if at all.

Our estimated rental price for the minimal operational unit of the XU-72 is approximately two-thirds of the listed rental price for the minimal operational unit of the 7090. This comparison indicates that we might enter IBM's market with the persuasive factor of lower cost for a minimal operational system.

The main competition presented to Sperry Rand by the 7090 is in three components.

1. High performance magnetic tape handlers.
2. High-speed, low cost magnetic core memory.
3. Inexpensive but less powerful on-line printer.

Considerations for Customer Approach

Because the XU-72 is a continuation in basic technology of the Scientific line of products, sales efforts will be aided by the customer's previous acquaintance with the capability, flexibility, and speed of the Univac computers. Our sales staff, therefore, can stress effectively the benefits accruing from additional and favorably competitive features of the XU-72, as listed below.

The reliable, rapid- access drum storage subsystem is well integrated with the core storage section for efficient use in selecting alternative programs or data. The combination of storage capacity and transfer speed is unique.

Program compatibility minimizes changeover cost for old customers and facilitates new customer installations while minimizing programming development costs for Sperry Rand.

Transistorization permits high operating speed, simplified site preparation, and low maintenance cost.

Memory segmentation and the XU-72 system logic minimize synchronization and queueing difficulties in complex problems, such as those involving drum references made concurrently with computation or multiple tape transfer.

The Uniservo II, coupled with the superior drum system, offers cost/performance advantages over the IBM high-performance tape subsystem.

The Univac line of peripheral off-line equipment can be used effectively with the system, and the "New Univac" may well prove to be an excellent satellite computer to the XU-72.

The XU-72 has 8 channels to which tape control units may be attached or which will permit the addition of other input-output units, such as an on-line printer. To any number of the 8 channels, additional units may be connected, thus providing an extensive growth capacity for peripheral equipment designed to meet the expanding needs of individual users of the system.

INSTRUCTION REPERTOIRE

TRANSMIT INSTRUCTIONS

11* Transmit Positive TPuv:#

Replace (v) with (u).

13 Transmit Negative TNuv:

Replace (v) with the complement of (u).

12 Transmit Magnitude TMuv:

Replace (v) with the absolute magnitude of (u).

15 Transmit U-address TUuv:

Replace the 15 bits of (v) designated by v_{15} through v_{29}, with the corresponding bits of (u), leaving the remaining 21 bits of (v) undisturbed.

16 Transmit V-address TVuv:

Replace the right-hand 15 bits of (v) designated by v_0 through v_{14}, with the corresponding bits of (u), leaving the remaining 21 bits of (v) undisturbed.

35 Add and Transmit ATuv:

Add D(u) to (A). Then replace (v) with (A_R).

36 Subtract and Transmit STuv:

Subtract D(u) from (A). Then replace (v) with (A_R).

22 Left Transmit LTjkv:

Left circular shift (A) by k places. If j = 0 replace (v) with (A_1); if j = 1 replace (v) with (A_R).

Q-CONTROLLED INSTRUCTIONS

51 Q-controlled Transmit QTuv:

Form in A the number L(Q) (u). Then replace (v) by (A_R).

* Octal notation

# Mnemonic notation

52 Q-controlled Add QAuv:

Add to (A) the number L(Q)(u). Then replace (v) by (A_R).

53 Q-controlled Substitute QSuv:

Form in A the quantity L(Q)(u) plus L(Q')(v). The replace (v) with (A_R). The effect is to replace selected bits of (v) with the corresponding bits of (u) in those places corresponding to 1's in Q. The final (v) is the same as the final (A_R).

REPLACE INSTRUCTIONS

21 Replace Add RAuv:

Form in A the sum of D(u) and D(v). Then replace (u) with (A_R).

23 Replace Subtract RSuv:

Form in A the difference D(u) minus D(v). Then replace (u) with (A_R).

27 Controlled Complement CCuv:

Replace (A_R) with (u) leaving (A_L) undisturbed. Then complement those bits of (A_R) that correspond to ones in (v). Then replace (u) with (A_R).

54 Left shift in A LAuk:

Replace (A) with D(u). Then left circular shift (A) by k places. Then replace (u) with (A_R). If u = a, the first step is omitted, so that the initial content of A is shifted.

55 Left Shift in Q LQuk:

Replace (Q) with (u). Then left circular shift (Q) by k places. Then replace (u) with (Q).

SPLIT INSTRUCTIONS

31 Split Positive Entry SPuk:

Form S(u) in A. Then left circular shift (A) by k places.

33 Split Negative Entry SNuk:

Form in A the complement of S(u). Then left circular shift (A) by k places.

32 Split Add SAuk:

Add S(u) to (A). Then left circular shift (A) by k places.

34 Split Subtract SSuk:

Subtract S(u) from (A). Then left circular shift (A) by k places.

TWO-WAY CONDITIONAL JUMP INSTRUCTIONS

46 Sign Jump SJuv:

If $A_{71} = 1$, take (u) as NI. If $A_{71} = 0$, take (v) as NI.

47 Zero Jump ZJuv:

If (A) is not zero, take (u) as NI. If (A) is zero, take (v) as NI.

44 Q-Jump QJuv:

If $Q_{35} = 1$, take (u) as NI. If $Q_{35} = 0$, take (v) as NI. Then, in either case, left circular shift (Q) by one place.

ONE-WAY CONDITIONAL JUMP INSTRUCTIONS

41 Index Jump IJuv:

Form in A the difference D(u) minus 1. Then if $A_{71} = 1$, continue the present sequence of instructions; if $A_{71} = 0$, replace (u) with (A_R) and take (v) as NI.

42 Threshold Jump TJuv:

If D(u) is greater than (A), take (v) as NI; if not, continue the present sequence. In either case, leave (A) in its initial state.

43 Equality Jump EJuv:

If D(u) equals (A), take (v) as NI; if not, continue the present sequence. In either case leave (A) in its initial state.

ONE-WAY UNCONDITIONAL JUMP INSTRUCTIONS

45 Manually selective Jump MJjv:

If the number j is zero, take (v) as NI. If j is 1, 2, 3, 4, 5, 6, or 7, and the correspondingly numbered MJ selecting switch is set to "jump", take (v) as NI; if this switch is not set to "jump", continue the present sequence.

37 Return Jump RJuv:

Let y represent the address from which CI was obtained. Replace the right-hand 15 bits of (u) with the quantity y plus 1. Then take (v) as NI.

14 Interpret IP:

Let y represent the address from which CI was obtained. Replace the right hand 15 bits of (F_1) with the quantity y + 1. Then take (F_2) as NI.

STOP INSTRUCTIONS

56 Manually selective Stop MJjv:

If j = 0, stop computer operation and provide suitable indication If J = 1, 2, 3, 4, 5, 6, or 7, and the correspondingly numbered MS selecting switch is set to "stop", stop computer operation and provide suitable indication. Whether or not a stop occurs, (v) is NI.

57 Program Stop PS --

Stop computer operation and provide suitable indication.

EXTERNAL EQUIPMENT INSTRUCTIONS

17 Interpretive External Function EF-v:

Select a unit of external equipment and perform the function designated by (v).

76 External Read ERjv:

If j = 0, replace the right-hand 8 bits of (v) with (10A).

77 External Write EWjv:

If j = 0, replace (10A) with the right-hand 8 bits of (v). Cause the previously selected unit to respond to the information in 10A.

61 PRint PR-v:

Replace (TWR) with the right-hand 6 bits of (v). Cause the typewriter to print the character corresponding to the 6 bit code.

63 PUnch PUjv:

Replace (HPR) with the right-hand 6 bits of (v). Cause the punch to respond to (HPR). If j = 0, omit seventh level hold; if j = 1, include seventh level hole.

External Function EFuv:

Select a unit of external equipment and perform the function designated by u and (v).

SEQUENCED INSTRUCTIONS

71 MultiPly MPuv:

Form in A the 72-bit product of (u) and (v), leaving in Q the multiplier (u).

72 Multiply Add MAuv:

Add to (A) the 72-bit product of (u) and (v), leaving in Q the multiplier (u).

73 Divide DVuv:

Divide the 72-bit number (A) by (u), putting the quotient in Q and leaving in A a non-negative remainder R. Then replace (v) by (Q). The quotient and remainder are defined by: $(A)_i = (u) \cdot (Q) + R$, where $0 \leq R < \left|(u)\right|$. Here $(A)_i$ denotes the initial contents of A.

74 Scale Factor SFuv:

Replace (A) with D(u). Then left circular shift (A) by 36 places. Then continue to shift (A) until $A_{34} \neq A_{35}$. Then replace the right-hand 15 bits of (v) with the number of left circular shifts, k, which would be necessary to return (A) to its original position. If (A) is all ones or zeros, k = 37. If u is A, (A) is left unchanged in the first step, instead of being replaced by $D(A_R)$.

75 RePeat RPjnw:

This instruction calls for the next instruction, which will be called NIuv, to be executed n times, its "u" and "v" addresses being modified or not according to the value of j. Afterwards the program is continued by the execution of the instruction stored at a fixed address F_1. The exact steps carried out are:

(a) Replace the right-hand 15 bits of (F_1) with the address w.

(b) Execute NIub, the next instruction in the program, n times.

(c) If j = 0, do not change u and v.

If j = 1, add one to v after each execution.

If j = 2, add one to u after each execution.

If j = 3, add one to u and v after each execution.

The modification of the u-address and v-address is done in UAK and VAK. The original form of the instruction in storage is unaltered.

(d) On completing n executions, take (F_1), as the next instruction. F_1 normally contains a manually selective jump whereby the computer is sent to w for the next instruction after the repeat.

(e) If the repeated instruction is a jump instruction, the occurrence of a jump terminates the repetition. If the instruction is a Threshold Jump or an Equality Jump, and the jump to address v occurs, (Q) is replaced by the quantity j, (n-r), where r is the number of executions that have taken place.

FLOATING POINT INSTRUCTIONS

64 Add FAuv:

Form in Q the normalized rounded packed floating point sum (u) + (v).

65 Subtract FSuv:

Form in Q the normalized rounded packed floating point difference (u) - (v).

66 Multiply FMuv:

Form in Q the normalized rounded packed floating point product $(u) \cdot (v)$.

67 Divide FDuv:

Form in Q the normalized rounded packed floating point quotient $(u) \div (v)$.

01 Polynomial Multiply FPuv:

Floating add (v) to the floating product $(Q)_i \cdot (u)$, leaving the packed normalized rounded result in Q.

02 Inner Product FIuv:

Floating add to $(Q)_i$ the floating product $(u) \cdot (v)$ and store the rounded normalized packed result in Q. This instruction uses MC location F_4 = 00003 for temporary storage, where $(F_4)_f = (Q)_i$.

03 Unpack UPuv:

Unpack (u), replacing (u) with $(u)_M$ and replacing $(v)_c$ with $(u)_c$ or its complement if (u) is negative. The characteristic portion of $(u)_f$ contains sign bits. The sign portion and mantissa portion of $(v)_f$ are set to zero.

NOTE: The subscripts M and C denote the mantissa and characteristic portions.

04 Normalize Pack NPuv:

Replace (u) with normalized rounded packed floating point number obtained from the possibly unnormalized mantissa in $(u)_i$ and the biased characteristic in $(v)_c$.

NOTE: It is assumed that $(u)_i$ has the binary point between u_{27} and u_{26}; that is, that $(u)_i$ is scaled by 2^{-27}.

05 Normalize Exit NEj-:

If j = 0, Clear the normalize exit flip-flop (designated NFF); if j = 1, set NFF to 1.

(a) When NFF is set to 1, the normalize, round, and pack sequence of each of the subsequent floating point instructions is omitted. For FA, FS, or FD, the result is a quasinormalized double-length mantissa in A with the units position at a_{61} and sign bits to the left; while for FM the units position is at a_{62} with sign bits to the left.

(b) If NFF is set to 1, it will remain set until it is cleared by an NEj-instruction.

(c) NFF must be cleared for the FP, FI, and the NP instructions.

An increase in the versatility and efficiency in the XU-72 will be realized by the inclusion of certain additional instructions in the repertoire. Such instructions are given in the following list.

(1) Generalized Repeat - This will add an increment from a 30 bit static register to UAK or VAK or both or neither depending on the value of J.

(2) Internal Function Command - This command will be used to set up certain conditions in various registers and also to read these registers. The UAK part will contain either information bits or an address. As an example, the internal function command will be used to load the increment register for the generalized repeat. As another example, the internal function will be used to read the error register.

(3) Increment Next Instruction - This command which is a command code and/or address modifier, will add the (u) to PCR of the next instruction. Such modifiers can be obtained from any memory location.

(4) Add Add command - This command will add (u) to A and then add (v) to A.

(5) Magnitudinal Threshold Jump - This command will compare the absolute value of (u) with the absolute value of A, and if (u) is greater than A jump to V for the next instruction.

(6) Polynomial Multiply - Multiply the contents of A by (u), then shift right the number of positions indicated by the polynomial shift register then add (v) to A.

(7) Split Threshold Jump - This command will be executed in the same manner as the threshold jump, except that the split subtract and split add sequence will be used in place of the double extension subtract and add.

(8) Right Shift - In A shift (u) in A K places to the right and store A back in (u).

(9) v to u command - transmit V of (u) to U of (v).

(10) u to v command - transmit U of (v) to V of (u).

(11) Split Equality Jump - This is the same as the equality jump, except that split subtract and split add sequences are used in place of the double length subtract and double length add.