DATE

6/14

TO: INFORMATION SECTION

☐ MLP ☐ FES ☐ WEC ☐ DHB

☐ Get us on positive

☐ Get one for us

☐ Get Proceedings

☐ Get Book

☐ Get Journal

☐ For Info

☐ Miscellaneous

✓ TO: RECEIVING

☒ Route to: *Joel Conner*
before vault

☐ Hold for Patent release

☐ Dup?

☐ Std. distribution report?

☐ Request - 540?

☐ Reject - send to:

☐ Miscellaneous

TO: RECORD SECTION

☐ No DTI Extension Master

TO: DESCRIPTIVE CATALOGING

☐ Reprint

☐ OK for NSA

☐ Not for NSA

FROM: ☐ Lee M. Thompson - 3-4540
☐ Mary H. Newman - 3-4549
☐ William Matheny - 3-4549
☐ John Mitchell - 3-4549

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

MASTER

NOV 11 6 1967

SYSTEM DESIGN FOR ARTRIX GRAPHICAL PROCESSOR

by

JOHN WILLIAM ESCH

January 4, 1967

DEPARTMENT OF COMPUTER SCIENCE • UNIVERSITY OF ILLINOIS • URBANA, ILLINOIS

RELEASED FOR ANNOUNCEMENT
IN NUCLEAR SCIENCE ABSTRACTS

Report No. 219

SYSTEM DESIGN FOR ARTRIX GRAPHICAL PROCESSOR

by

JOHN WILLIAM ESCH

RELEASED FOR ANNOUNCEMENT
IN NUCLEAR SCIENCE ABSTRACTS

January 4, 1967

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

ACKNOWLEDGEMENT

The author wishes to express his gratitude to his advisor, Professor W. J. Poppelbaum, whose counsel and encouragement have been very helpful.

The author is also indebted to the Department of Computer Science for the assistantship and facilities which made this project possible.

Thanks are also extended to the other members of the Artrix group, Al Irwin, William Kubitz, Peter Oberbeck, and David Rollenhagen, whose help and friendship have been invaluable.

PREFACE

The Artrix system as conceived by Dr. W. J. Poppelbaum is a research venture into display systems.

The basic property of Artrix is its ability to store and display graphic information and to do Euclidian construction all without the aid of a backup digital computer. This is accomplished by on-line processing of TV-frames using digital control of analog signals. ✓

The hardware that the operator sees are a display monitor, a light pen, and a control console. Through the latter the operator indicates to the machine what he wants done and by lighting indicators the machine informs the operator what it has done as well as what state the system is in.

The internal hardware is composed of combinations of digital, analog, and hybrid circuits which makes Artrix a "hybrid computer" in the true sense of the word. It is to be noted that the analog components have to operate at 10 mc frequency, i.e., that we are talking about circuits several orders of magnitude faster than conventional analog circuits.

The Author's Masters Degree project was to design and construct the processor section of the Artrix system.

TABLE OF CONTENTS

ACKNOWLEDGEMENT	iii
PREFACE	iv
LIST OF FIGURES	vi
1. THE ARTRIX PROCESSOR	1
1.1 Processor Functions	1
1.2 Example of Processor Operation	3
2. QUALITATIVE DESCRIPTION	5
2.1 Lissajous Patterns	5
2.2 Hybrid Circuitry to Generate Mathematical Expressions	6
2.3 Counters to Store Points and Magnitudes	7
2.4 Example Describing Internal Operation	8
3. SYSTEM PROBLEMS	12
3.1 Five Hundred Line Resolution	12
3.2 Interlacing	12
3.3 Synchronizing Storage System to the Processor	13
3.4 Synchronizing Counters	14
3.5 Ripple Counter Sneak Pulses	14
3.6 Slope Magnitude and Sign	16
4. TECHNICAL DESIGN	18
4.1 Switching Circuits	18
4.2 Counters	18
4.3 Control Circuits	18
5. SUMMARY AND CONCLUSIONS	23
APPENDIX	26
LIST OF REFERENCES	38

LIST OF FIGURES

FIGURE:	PAGE:
1. Artrix Information Flow Diagram	2
2. Processor Information Flow Diagram	10
3. Switching Circuit Schematic	29
4. Horizontal Master (HM) Counter Schematic	30
5. Vertical Master (VM) Counter Schematic	31
6. Horizontal Point One (HP ₁) Counter Schematic	32
7. Vertical Point One (VP ₁) Counter Schematic	33
8. Horizontal Point Two (HP ₂) Counter Schematic	34
9. Vertical Point Two (VP ₂) Counter Schematic	35
10. Expanding Radius (ER) Counter Schematic	36
11. Control Circuit Schematic	37

1. THE ARTRIX PROCESSOR

1.1 Processor Functions

The processor unit in the "Artrix Information Flow Diagram" Figure 1, on the next page, has several functions. One of these is to communicate with the operator through the control console. The processor continuously lights indicators on the console which inform the operator what state the system is in. In the same manner it informs the operator of its responses to the commands that the operator has sent to the processor through pushbuttons on the console.

Besides communicating with the operator through the control console, the processor takes the commands received from the operator and construction points in the point memory indicated by him and processes this information on-line. The result of this processing is the output signal which effectuates the desired euclidian construction.

An additional restraint on the processor is that the operator is not to be able to observe any time lag between his last command and the completion of the construction by the processor. Consequently the processor's circuits (digital, analog, and hybrid) must be very fast.

It is also a function of the processor to synchronize the entire Artrix system so that all units have the same time base and phase.

Figure 1.

Artrix Information Flow Diagram.

1.2 Example of Processor Operation

Referring to Figure 1 again, Artrix Information Flow Diagram, consider the case where the operator wishes to draw a line segment between some Point 1 and some other Point 2. The procedure would be as follows: The operator pushes the button labeled "Line". This sends the line mode of operation command to the processor which responds by lighting up that button. The operator then pushes the "Reset Processor" button which sends that command to the processor and is also lit in acknowledgement. The operator now takes the light pen and points it at the point where he would like one end of the line to be. He pushes the enable button which transmits that point to the processor which in turn responds by lighting up the "Point 1 Stored" button. Since this happens instantaneously as far as the operator can tell, he can immediately indicate a point on the other end of the line segment, Point 2. When he does, by pushing the enable button, "Point 2 Stored" lights up. Again, because of the processor's speed, the "Execute Construct" button lights, telling the operator that the line he wants can be written into memory just by pushing that button.

Suppose the operator has done this and now wants to construct a circle such that the line segment just described determines the radius and a third point the center of the circle. The procedure is quite similar. The operator pushes the "Circle" mode button which the processor immediately lights in acknowledgement. Because the processor has already stored Point 1, (the center), the operator does not have to indicate that point again. Therefore,

instead of resetting the entire processor, he can just reset Point 2. ("Point 2 Stored" light goes out). This is necessary because in the "Line" mode Point 2 stores the lines magnitude and not the actual point. The operator now indicates Point 2 again and pushes the enable button. When he does, the processor lights "Point 2 Stored" and "Execute Construct". Since this circle (which the processor is ready to write into memory) is not centered at the desired point, the operator now resets Point 1 (Point "1" Stored and "Execute Construct" lights go out) and indicates the new center of the circle and pushes the enable button. "Point 1 Stored" again lights as does "Execute Construct" which tells the operator that he has translated the circle to the desired new center and the processor is ready to write that circle into memory. This process of translation can be done in any of the construction modes just by resetting Point 1 and indicating a new Point 1.

2. QUALITATIVE DESCRIPTION

2.1 Lissajous Patterns

These patterns are most commonly seen on an oscilloscope where the horizontal input is one sinusoid and the vertical input is another sinusoid of a different magnitude and frequency. The resulting trace on the oscilloscope is called a Lissajous Pattern. Under special conditions this pattern takes the form of a circle or a line.

The restrictions for a circle to result are that the a.c. component of both inputs have the same frequency and magnitude. In addition, they must be 90° out-of-phase with each other as, for example, $\sin\omega t$ and $\cos\omega t$ are. There are no restrictions on the d.c. components because they only determine the translation of the Lissajous Pattern. In general, then, a circle can be expressed mathematically as: $X = R \cos\omega t + X_1$, $Y = R \sin\omega t + Y_1$. Note that the horizontal deflection, X , and the vertical deflection, Y , have the same frequency, ω , and the same magnitude, R , (the radius of the circle).

The restrictions for a line to result are that the a.c. components of both inputs have the same frequency and phase. The magnitude then determines the shape. In general the line can be expressed mathematically as: $X = \Delta X \sin\omega t + X_1$, and $Y = \Delta Y \sin\omega t + Y_1$. Note that X and Y have the same frequency, ω , and phase, $\sin\omega t$, but different magnitudes, ΔX and ΔY . Also, as for the circle, the translation, (X_1, Y_1) is arbitrary.

Going back to the construction in Section 1.2, we note that

both lines and circles were constructed by specifying (with the light pen) two points in succession: For the circle, its center (X_1, Y_1) , and any point (X_2, Y_2) on its circumference; for the line segment, its end points (X_1, Y_1) and (X_2, Y_2) . To obtain the desired circle, the processor generates the sweep:

$$X = r t \cos \omega t + X_1, Y = r t \sin \omega t + Y_1 \text{ where } r \gg \omega,$$

i.e., we generate a circle with slowly expanding radius compared to the rate at which the circle is being swept out. At some time later, $t = \tau$, when $X = X_2$ and $Y = Y_2$ simultaneously the radial expansion is stopped, leaving the desired circle $X = r \tau \cos \omega t + X_1$, and $Y = r \tau \sin \omega t + Y_1$ for $t > \tau$. Note that $r \tau = R$, the radius of the circle, where $R^2 = (X_2 - X_1)^2 + (Y_2 - Y_1)^2$.

To obtain a line segment the processor generates the sweep:

$$X = (X_2 - X_1) | \sin \omega t | + X_1$$

$$Y = (Y_2 - Y_1) | \sin \omega t | + Y_1$$

Note that this line segment has (X_1, Y_1) at one end of the line segment and not at the midpoint of the line segment because the processor generates the absolute magnitude of $\sin \omega t$.

2.2 Hybrid Circuitry to Generate Mathematical Expressions

There are five kinds of hybrid circuits used in the processor. The high speed high precision Digital to Analog Converters are used to transform the digital points (X_1, Y_1) and (X_2, Y_2) , into analog equivalents. The low offset, low noise Digital Controlled Analog Switches have outputs which float for a logical "0" input

and are equivalent to the analog inputs for a logical "1" input.

The DCVGLA are very accurate linear amplifiers which convert an input sinusoid to another sinusoid of different amplitude. The new amplitude is determined by digital inputs and consequently this device is called a digital controlled variable gain linear amplifier. The Controlled Rectifiers rectify a sinusoidal signal full wave positive, full wave negative, or not at all---depending on the digital control signals. This device has low distortion and offset so that no D.C. error is introduced into the output. The Comparator has the ability to compare two analog signals, positive or negative, over a wide range of voltages and output a logical "1" when they compare within a very small tolerance.

2.3 Counters to Store Points and Magnitudes

There are seven nine-bit counters, HM, VM, HP₁, VP₁, HP₂, VP₂, and ER. The HM, Horizontal Master, and VM, Vertical Master, counters run continuously. Their count specifies the horizontal and vertical coordinates of the Storage System's deflections. They also serve to synchronize the storage system to the processor. The HP₁, (Horizontal Point One), and VP₁ (Vertical Point One), counters contain the coordinates (X₁, Y₁) when Point One is stored. The HP₂, (Horizontal Point Two), and VP₂ (Vertical Point Two) counters contain the coordinates (X₂, Y₂) when Point Two is stored in the circle mode and the magnitudes $|X_2 - X_1|$, $|Y_2 - Y_1|$ in the line mode. The ER, (Expanding Radius) counter contains the magnitude of the radius, R, in the circle mode.

The logic schematics for each of the counters can be found in the Appendix, Figures 4 through 10, pp. 30 to 36.

2.4 Example Describing Internal Operation

The logic circuits for the hardware described below is shown on Figure 11, p. 37, Control Circuits Schematic.

In the example of Section 1.2, the operator wants to first construct a line. When he pushes the line mode button, a flip-flop is set in the control circuits section of the processor, which lights the line button light (processor response) and switches gates for line mode operation within the processor. When the reset processor button is pushed, the HP_1 and VP_1 counters are made to run synchronously with the HM and VM counters and the HP_2 and VP_2 counters are set to zero. When the processor receives the first construction point, Point One, it stops the HP_1 and VP_1 counters, lights the "Point 1 Stored" button, and starts the HP_2 and VP_2 counters counting. When the second construction point is indicated the processor also stops the HP_2 and VP_2 counters and lights the "Point 2 Stored" and "Execute Construct" buttons. Because the reaction time of the operator is in the millisecond range and that of the processor's hybrid circuits in the microsecond range, all of the processor's circuits have settled long before the operator can push the "Execute Construct" button.

More precisely, when HP_1 and VP_1 stop counting, the digital to analog converters wired to them convert their digital translation to an analog one which can be added with the proper sinusoid to form

the mathematical expression for a line. (See Figure 2, Page 10).

When HP_2 and VP_2 stop counting, their magnitude is switched to the DCVGLA which amplify $\sin\omega t$ terms to that amplitude. Some of the gating that occurs when the line mode flip-flops is set, includes the type of sinusoid going into the DCVGLA (in this case, both $\sin\omega t$ terms) and the type of rectification.

Continuing with the example, when the circle mode button is pushed, a flip-flop is set in the processor and again the circle mode light is on, the line mode flip-flop is reset, and similar gating is done inside the processor. When Point Two is reset this time, it starts counting in synchronism with HM and VM counters. When Point Two is indicated, the ER counter starts counting and its output, because of the gating, now controls the DCVGLA. When the outputs compare with both X_2 and Y_2 simultaneously, the ER counter is stopped, the "Radius Stored" button, and "Execute Construct" buttons are lighted.

To translate the circle to a new point, "Point 1 Stored" is reset, causing HP_1 and VP_1 to start counting in synchronism with HM and VM counters and the console to be updated. Indicating a new Point One causes the counters to stop and a new analog translation to be added to the circle of desired radius.

Because of the design of the processor, several interesting features are available to the operator. Since the line operation only uses HP_1 , VP_1 , HP_2 , and VP_2 counters, an operator can save a radius in the ER counter as long as he likes while he draws lines. In addition, since the magnitude of a line is stored in the HP_2 and VP_2

Figure 2.

Processor Information Flow Diagram

counters, both a line and a circle can be saved and translated at will without destroying the contents of any of the HP_2 , VP_2 , or ER counters.

3. SYSTEM PROBLEMS

3.1 Five Hundred Line Resolution*

Our T.V. system has 500 vertical lines and about another 25 for vertical blanking. To insure that the processor be able to equal this resolution, nine-bit counters (512 lines) were used. The procedure adopted is to start the VM counter from zero at the end of vertical blanking and have it count each line. As it overflows, it is held until the end of vertical blanking, when it again starts counting.

To get 500 line horizontal resolution, the frequency of the T.V. deflection system's horizontal lines was multiplied by 512 to get 8,062 Mhz. A crystal controlled clock at this frequency drives the HM counter which overflows then at the line frequency of the T.V. system. The clocking signals which are generated to accomplish this are called TV and TH respectively.

It is worth noting here that, since the magnitude of the field of view of the T.V. screen corresponds to the maximum magnitude of the counters, the magnitude of the complement of the counter's contents plus the magnitude of the counter's counters equals the magnitude of the field of view.

3.2 Interlacing

Because T.V. systems are interlaced (all even lines, then all odd lines), the vertical counters had to be designed to take this

*The logic circuit schematics and an explanation of the logic signal names can be found in the Appendix.

fact into account. Since successive even and odd lines in a normal sequence differ only in the least significant bit, the line count was fed into the second least significant bit and the least significant bit changed according to whether or not it was an even or odd field.

The effect of doing this is to count by twos through the even field, set the least significant bit and count by twos through the odd field, reset the least significant bit, and start over.

3.3 Synchronizing Storage System to Processor

Normally, the T.V. system operates off the crystal in the Camera Control Unit which operates at 31.5 kHz, twice the frequency of line rate, 15.75 kHz. To synchronize the T.V. system to the processor, the output of the second most significant bit of the HM counter is fed into the T.V. Camera Control Unit as an external sync. (Remember the most significant bit of the HM counter changes at the line frequency 15.75 kHz). However, this is insufficient for two reasons: When the system is turned on, the Camera Control Unit can lock onto either phase of the 31.5 kHz sync. signal. Consequently, the system could begin operating with the T.V. system's horizontal line beginning in the middle of the processor horizontal line (or vice-versa, depending on which is taken as a reference). This phase problem was solved by sending the T.V. Camera Control composite sync. signal to the processor control unit. Here the phase is monitored continuously and adjusted automatically if the system for some reason should slip out of phase.

The second problem centers around the Camera Control Unit. Even when the phase is adjusted, the horizontal sync. signals produced by the Camera Control Unit lag in time the sync. signals from the processor due to time delays in the Camera Control Unit. This was corrected by shifting the sync. signal from the processor to the Camera Control Unit ahead in time with respect to the old sync. signal by the same amount as the delays in Camera Control Unit.

3.4 Synchronizing Counters

In the examples of Section 2.4 it was stated that some of the counters had to be set running synchronously with the HM and VM Counters. The original scheme was to set the counters to zero when the reset button was pushed and start them counting when the HM and VM counters both overflowed simultaneously. However, because of a strange property of the integrated circuit flip-flops used, the ripple type counters could not reliably be set to zero. The solution was to monitor their count also, and, when they both overflowed, to stop them and wait for the next HM and VM overflow to start them again.

3.5 Ripple Counter Sneak Pulses

In Section 3.4 it was assumed that it would be easy to tell when a counter overflowed. This is usually done by examining all the stages of the counter and judging it an overflow when they change from all ones to all zeros. There are no problems when all the stages change at the same time as in a synchronous counter, but

in a ripple counter there are time delays which allow false counts for very short times. Because of the very fast speed of the integrated circuits used, they are able to pick up the false counts easily. For characteristics used see reference in Bibliography on T.I. circuits.

To explain in more detail let us examine the operation of a ripple counter. Say that the changing of a stage from a logical "0" to a logical "1" on its reset side causes the next stage following it to trigger and that there is a delay of τ seconds from the "0" to the "1" transition until the next stage flip-flop has changed states. Assume you are monitoring the reset side of the counter with an "And" gate for all ones (which means the set side would be all "0"). When this condition occurs, you know overflow has occurred. Consider the case where the contents of the counter is a string of zeros followed by a string of ones on the reset side. When the next trigger pulse comes along, the first stage will change, causing the second stage to change until the first stage is reached which had a one on the reset side. When the stage before it changes from zero to one, there is a time during which all stages have a one on the reset side before that stage changes from a one to a zero. Because most gates are faster than their compatible flip-flops, sneak pulses will almost always occur under this condition. To give the reader an idea of the problem faced in debugging the processor the sneak pulses were only 20 ns wide.

3.6 Slope Magnitude and Sign

Recall that in the construction of lines the HP_2 and VP_2 counters are started from zero, when Point One is indicated and stopped when Point Two is indicated. Since the operator can specify either end of the line first, there are four possible classes of lines. The different classes are determined by the horizontal deflection of Point Two being larger or smaller than that of Point One and similarly for the vertical deflection. If the horizontal (vertical) deflection of Point Two is larger than that of Point One, then there is no problem, for the HP_2 (VP_2) counter contains the correct count. If, however, this is not the case, then the HP_2 (VP_2) counter contains the one's complement of the desired count because of the properties discussed in Section 3.1. That is to say, if the horizontal (vertical) deflection of Point Two is smaller than that of Point One the count in the HP_2 (VP_2) counter will be the one's complement of the desired count.

The problem of knowing whether or not to use the one's complement or the exact value was solved by the following technique. For ease of understanding consider first the horizontal case (the vertical case being handled the same way). Take a flip-flop and set it each time the HM counter overflows, reset it each time Point One is reached ($\therefore$ HP_2 overflows), and hold its contents when Point Two is designated. Thus, if Point Two is designated at a point where the horizontal deflection of Point Two is larger than that of Point One, then the flip-flop output will be an "0", since Point One has been passed and the HM counter has not overflowed yet. If the hori-

zontal deflection of Point Two is less than that of Point One, the flip-flop output will be a "1". In the first case, logical "0", the HP_2 contains the correct value, and in the second, logical "1", HP_2 contains the one's complement of the correct value.

The problem of slope magnitude is thus solved by looking at the outputs of two flip-flops (one for the horizontal and one for vertical) and using the one's complement or not of the HP_2 and VP_2 counters if the respective flip-flops is a logical "1" or not.

The problem of slope sign can be solved by full wave rectifying the output of the DCVCLA in a plus or minus direction according to the same flip-flop signals. Full wave rectifying in a plus direction for logical "0" and in a minus direction for a logical "1".

4. TECHNICAL DESIGN

4.1 Switching Circuits

From the discussion in 3.6 on Slope magnitudes and the knowledge that the radius is stored in the ER counter, it is obvious that the inputs to the DCVGIA come from several different places depending on certain conditions.

Consider the DCVGIA for the horizontal positions. In the circle mode its digital control signals come from the ER counter. The bits of that counter are labeled Q_0ER , Q_1ER , ... Q_8ER where Q_0 is the least significant bit and Q_8 the most significant bit. The control signal for the circle mode of operation is CR. In the line mode the control signals are CH and $\overline{CH}$. CH is a logical "1" when the one complement of HP_2 counter is to be taken and $\overline{CH}$ is a "1" when the actual count of HP_2 is to be taken. Both are logical "0" in the circle mode of operation when CR is a "1". That is to say

$$CR \cdot CH = CR \cdot \overline{CH} = CH \cdot \overline{CH} = "0".$$

The DCVGIA for the vertical position has control signals CR, CV, and $\overline{CV}$ with the same meanings only applying to the ER and VP_2 counters. Figure 3, labeled Switching Circuit Schematics, gives the details.

4.2 Counters

The ripple counter, although easy to design, can have many deficiencies as noted in several earlier sections. The circuits used were the Texas Instruments Series 74 Integrated circuits which are

described in the reference on T.I. integrated circuits. The flip-flops trigger on a "0" to "1" transition; consequently, to get an increasing count the output of the reset or $\bar{Q}$ side is fed to the trigger of the next most significant stage for an increasing binary count. Because these flip-flops have enabling control signals, all of them have to be wired to the enabling state as appropriate. Examples of the ripple counters used are in the Appendix under VM, VP₁, VP₂, and ER. (Figures 5, 7, 9, and 10, respectively).

The synchronous counter, although harder to design, is more elegant in its operation and easier to use. Here the enabling controls of each flip-flop are very useful, because to have synchronous operation, each stage must look at all less significant stages. If all of them are at a logical "1", it changes state when the next clock pulse occurs. As opposed to the ripple counter where the clock pulse goes only to the first stage, the synchronous counter sends the clock pulse to every stage. Because there is an eight input NAND in the series 74 integrated circuits, it was possible to design the basic synchronous counter with at most one gate between stages. Examples of the synchronous counters used are in the Appendix under HM, HP₁, and HP₂. (Figures 4, 6, and 8, respectively).

4.3 Control Circuits.

It is suggested that at this point the reader skim over the "Logical signals" section of the Appendix since it will be assumed the reader knows what the control signals talked about in this section stand for. It will also help the reader if he follows the discussion

on the control circuit schematic in the Appendix. (Figure 11). First, let us give a description of the circuits involved in the command, response, and state of the system signals. Since the circle and line mode of operation are mutually exclusive (and, therefore, cannot be done at the same time), the circle and line mode push buttons control opposite sides of an RS flip-flop, the outputs of which are CR for circle and $\overline{CR}$ for line. These then light indicators on the processor card rack and the Control Console as well as switch gates inside the processor. P_1 and $\overline{P_2}$ are the outputs of a similar RS flip-flop whose inputs are the gated light pen signal on one side and OP-RP and OP-R1 on the other side. This flip-flop lights indicators on the processor and control console and also switches gates. All light pen pulses are gated to the P_1 flip-flop; and, if that has already been set, they are also gated to the P_2 flip-flop, which is reset by OP-RP or OP-R₂. The P_2 flip-flop lights indicators and switches gates also. The RS (Radius Stored) flip-flop is set by H_{coinc} and V_{coinc} being a logical one at the same time when in the circle mode operation and P_2 has been set. It is reset by OP-RP or OP-RR and its output also lights indicators and does gating. The PWG signal is the output of an RS flip-flop which is set by XC and reset by the $\overline{XC}$ signal. The two remaining responses are formed from signals already discussed.

$$PR = P_1 \cdot P_2 \cdot RS \text{ and } XR = P_1 \cdot P_2 \cdot \overline{CR} + P_1 \cdot RS \cdot CR.$$

Section 3.1 discussed 500 vertical line resolution. This is accomplished by generating the clocking signal TV. This signal is equivalent to the last stage of the HM counter, $\overline{Q}_8 HM$, until the VM counter is all zeros when TV is gated to a zero until the

end of vertical blanking when it again is allowed to be equivalent to $\overline{Q_8^{HM}}$.

Section 3.2 discussed Interlacing where the least significant bit had to be set for odd fields and reset for even fields. For the VM and VP_1 counters this was easily accomplished. If E is a logical one when VB occurs, then set the least significant bit; and if E is a logical zero when VB occurs, reset it. The VP_2 counter is more complicated because of its operation in the line mode. The procedure here is to set it when VB occurs if ϕ is a logical one where $\phi = \overline{E} \cdot CR + P_1 \cdot \overline{CR}$ and to reset it when VB occurs if θ is a logical one where $\theta = E \cdot CR + P_1 \cdot \overline{CR}$.

In Section 3.3 two problems were discussed: Line phase and Camera Control Unit delay. The former was solved by having TH not be equivalent to TM, but by having $TH = \overline{TM} \cdot (\overline{A_{1-8}} HM + A_{1-8} HM \cdot CS)$. The latter was solved by not having the sync. signal equivalent to $\overline{Q_7^{HM}}$ but equivalent to $Q_7^{HM} \cdot Q_6^{HM_1}$ which amounted to the appropriate time shift.

Section 3.4 discusses the synchronizing of counters. The actual procedure is to set the counter to zero, then wait until both the HM and VM counters overflow and start the counters counting. The latter is done by resetting the OF_1 and OF_2 flip-flops when the $OP-R_1$ and $OP-R_2$ pushbuttons are pushed respectively, and setting them when HM and VM overflow occurs. Then $\xi H = OF_1 \cdot \overline{P_1}$, $\eta H = OF_2 \cdot \overline{P_2} \cdot (CR + P_1)$, $\xi V = \xi H \cdot (CL_2 \rightarrow HP_1) + \overline{A_{1-8}} VP_1$, and $\eta V = \eta H \cdot (CL_2 \rightarrow HP_2) + \overline{A_{1-8}} VP_2$.

The problem of sneak pulses in ripple counters, which was discussed in Section 3.5, was solved by clocking all critical signals at a time when no sneak pulse could occur. In this case the edge of the clocking signal opposite to that one which does the triggering.

The problem of slope magnitude and sign discussed in Section 3.6 were solved by obtaining the control signals CV , $\overline{CV}$, CH , and $\overline{CH}$. Essentially CV and $\overline{CV}$ are opposite sides of an R.S. flip-flop except that both are gated to zero when CR is a logical 1. The flip-flop is set whenever the VM counter overflows, is reset whenever the VP_2 counter overflows, and is held when P_2 is a logical 1. Similarly, CH and $\overline{CH}$ are opposite sides of an R.S. flip-flop, except that both are gated to zero when CR is a logical 1. The flip-flop is set whenever the HM counter overflows, is reset whenever the HP_2 counter overflows, and is held when P_2 is a logical 1.

It is again brought to the reader's attention that the Appendix contains both an explanation of all logical signals and the circuit schematics of the entire processor.

5. SUMMARY AND CONCLUSIONS

In this thesis it was the author's objective to give the reader an understanding of what the Artrix Graphical Processor can do. First, the processor communicates with the operator by receiving push button commands and lighting indicators to inform the operator of its responses to those commands and of the state of the system. Secondly, the processor does automatic on-line graphical processing, of which the end results are Euclidian construction. And, thirdly, it synchronizes the rest of the Artrix system.

The successful operation of the processor clearly demonstrates the feasibility of doing on-line graphical processing using hybrid circuitry. It is important to note that the hybrid circuits used respond orders of magnitude faster than conventional analog circuits. It is this speed that makes the concept of on-line graphical processing using hybrid circuits so practical and elegant: Practical and elegant because the systems involved can be designed more simply and straight forwardly. This is because the graphical display is essentially analog in nature and control and switching essentially digital. Consequently the hybrid system can take the best from both worlds.

After working with the Artrix system the author has come to several conclusions. Since the operator is controlling the graphical processor with a light pen, the resolution of his work is limited by how fine a line and small a point he can see. Consequently, any processor need have an accuracy only as great as

the operator can control the processing. This suggests that one very practical way to do processing is to give the operator control of the parameters instead of having his constructions done automatically. For example, in the circle mode of operation a standard circle could appear with a small spot at its center. The operator then could translate the circle to the desired position, by locking the circle's center to the light pen. The exactness of the move is determined by the operator's ability to position the center. Similarly, he could next expand the radius to pass through some point. Again the accuracy is determined by the operator's ability to position the edge of the circle. Notice that in the above discussion the operator selects the type of geometric figure he wants to work with. It appears on the monitor, and the operator controls all the parameters of the general equation of the figure which he changes to suit his desire. The accuracy of the final figure is limited by the resolution of the display system and the sensitivity of the operator's controls.

Such a system is not confined to a digital, analog, or hybrid system because of the fact that the way the parameter are formed and changed is up to the designer. However, it is the author's feeling that systems using a digital computer are designed around the computer and not designed around the problem at hand. Also, there is a tendency in such systems to push design problems that arise into the computer to be solved by programming. This is wrong. The system should be designed to solve these problems inherently. It is because hybrid circuitry is so versatile that

these problems can be solved more easily using them. The system that may best meet these needs is one which has function generators whose basic building block is the operational amplifier and whose interconnections are determined by logic circuitry and digital controlled analog switches. Such a hybrid system could easily be designed to incorporate most of the problems of a graphical processor.

APPENDIX

A_{i-j}	Logical And of bits i through j.
$\overline{A_{i-j}}$	Logical And Not of the not side of bits i through j.
CH	CH means Complement Horizontal if a logical one.
<u>CH</u>	Don't complement Horizontal if a logical one. $CH \cdot \underline{CH} = 0$
CL_1 CL_2	Because the gates have a limited fan out, the nine-bit counters need more than one signal to clear them. CL standing for Clear.
CR	CR stands for Circle and is a logical one when the processor is in the circle mode of operation.
CS	CS stands for the Composite Sync. generated by the T.V. Camera Control Unit.
CV	CV means Complement Vertical if a logical one.
<u>CV</u>	<u>CV</u> means don't complement Vertical if a logical one. $CV \cdot \underline{CV} = 0.$
E	E stands for Equivalence and is a logical one when the two high order bits of the Horizontal Master Counter are the same.
EF	EF stands for Expanding Frequency and is the frequency which drives the Expanding Radius Counter.
ER	ER stands for Expanding Radius Counter whose contents is the radius of circles drawn.
H_i	H_i is the i^{th} bit of the digital signal which controls the Horizontal DCVGLA.
HM	HM stands for the Horizontal Master Counter whose contents correspond to the horizontal deflection in the T.V. system.
HP_1	HP_1 stands for Horizontal Point One Counter and contains the horizontal coordinate of Point One.
HP_2	HP_2 stands for Horizontal Point Two Counter and contains the horizontal coordinate of Point Two for circles and the horizontal projection of the line for lines.
H_{coinc}	H_{coinc} stands for coincidence of the horizontal sweep with the horizontal coordinate for Point Two.

LP	LP stands for Light Pen Pulse which occurs once each time the operator indicates a point on the display in the point memory and pushes the Enable Button.
OF	OF stands for Overflow and implies the condition that a Counter has gone from all ones to all zeros.
OP	OP stands for an Operator-controlled signal.
PR	PR stands for Processor Reset and is a logical one when the processor is reset.
PWG	PWG stands for Processor Write Gate and is a logical one when the processor output is being written into the display memory.
P_1	P_1 stands for Point One and is a logical one when P_1 has been indicated by the operator.
P_2	P_2 stands for Point Two and is a logical one when P_2 has been indicated by the operator.
Q_i	Q_i is the i^{th} bit of a counter.
$\overline{Q}_i$	$\overline{Q}_i$ is the not side of the i^{th} bit of a counter.
RP	RP stands for Reset Processor and is a logical zero when the processor reset button is pushed.
RR	RR stands for Reset Radius and is a logical zero when the Radius Reset Button is pushed.
RS	RS stands for Radius Stored and is a logical one when the proper radius for the circle has been found.
R_1	R_1 stands for Reset Point One and is a logical one when Horizontal and Vertical Point One Counters are being reset.
R_2	R_2 stands for Reset Point Two and is a logical one when Horizontal and Vertical Point Two Counters are being reset.
TH	TH stands for Trigger Horizontal and is the clocking signal for all Horizontal Counters.
TM	TM stands for Trigger Master and is the 8.062 mega hertz square wave.
TV	TV stands for Trigger Vertical and is the clocking signal for all Vertical Counters.
VB	VB stands for the Vertical Blanking signal generated by the T.V. Camera Control Unit.

V_{coinc}	V_{coinc} stands for coincidence of the Vertical Sweep with the Vertical Coordinate for Point Two.
V_i	V_i is the i^{th} bit of the digital signals which control the Vertical DCVGLA.
VM	VM stands for Vertical Master Counter whose contents correspond to the Vertical Deflection in the T.V. system.
VP_1	VP_1 stands for Vertical Point One Counter and contains the Vertical Coordinate of Point One.
VP_2	VP_2 stands for Vertical Point Two Counter and contains the Vertical Coordinate of Point Two for circles and the Vertical Projection of the line for lines.
XC	XC stands for Execute Construction and is a logical one when the Execute Construct Button is pushed.
XR	XR stands for Execute Ready and is a logical one when a construction is ready to be written into memory.
ηH	ηH is a logical one when the Horizontal Point Two Counter is to count.
ηV	ηV is a logical one when the Vertical Point Two Counter is to count.
θ	θ is a control signal for the Vertical Point Two Counter low order (least significant) bit. It is a logical one when that bit is to be triggered into the <u>reset</u> state.
ξH	ξH is a logical one when the Horizontal Point One Counter is to count.
ξV	ξV is a logical one when the Vertical Point One Counter is to count.
ϕ	ϕ is Control Signal for the Vertical Point Two Counter low order (least significant) bit. It is a logical one when that stage is to be triggered into the <u>set</u> state.

Figure 3:
Switching Circuit Schematic

Figure 4.
 Horizontal Master (HM) Counter Schematic

Figure 5.
 Vertical Master (VM) Counter Schematic

Figure 7.
Vertical Point One (VP₁) Counter Schematic.

Figure 8.
 Horizontal Point Two (HP₂) Counter Schematic

Figure 9.
Vertical Point Two (VP₂) Counter Schematic.

Figure 10.
Expanding Radius (ER) Counter Schematic.

LIST OF REFERENCES

1. "Quarterly Technical Progress Report", (Hardware Systems Research), Department of Computer Science, University of Illinois, Urbana, Illinois. July-September, 1965.
2. "Quarterly Technical Progress Report", (Hardware Systems Research), Department of Computer Science, University of Illinois, Urbana, Illinois. October-December, 1965.
3. "Quarterly Technical Progress Report", (Hardware Systems Research), Department of Computer Science, University of Illinois, Urbana, Illinois. January-March, 1966.
4. "Quarterly Technical Progress Report", (Hardware Systems Research), Department of Computer Science, University of Illinois, Urbana, Illinois. April-June, 1966.
5. T.I. Series 54/74 Integrated Circuits, Storer, William A., Editor. Texas Instruments, Incorporated, 1966.

**CONTRACTOR'S RECOMMENDATION FOR
DISPOSITION OF RESEARCH DOCUMENT**

(See Instructions on Reverse Side)

1. TITLE OF DOCUMENT:

System Design for Artrix Graphical Processor

2. PERSONAL AUTHOR(S)

John William Esch

3. CONTRACTOR'S NAME AND ADDRESS

University of Illinois, Urbana, Illinois 61801

4. CONTRACT NUMBER

US AEC AT(11-1)-1469

5. DATE OF DOCUMENT

January 6, 1967

6. REPORT NUMBER

(219)

COO-1469-0048

Complete only one item below, either 7, or 8, or 9.

☐ 7. DOCUMENT HAS BEEN
OR WILL BE SUBMITTED
FOR JOURNAL PUBLICA-
TION.

a. NAME OF JOURNAL

b. ESTIMATED DATE OF PUBLICATION

a. DATE OF MEETING

b. LOCATION

c. SPONSORING ORGANIZATION

NSA

(Check one: d or e)

d. ☐ AEC's standard announcement and distribution procedures may prevail after date of meeting.e. ☐ Recommend that there be no public announcement or distribution because:☐ 8. DOCUMENT HAS BEEN
OR WILL BE PRESENTED
ORALLY.☒ 9. DOCUMENT IS A PROG-
RESS REPORT NOT IN-
TENDED FOR JOURNAL
PUBLICATION OR ORAL
PRESENTATION.

(Check one: a or b)

a. ☒ AEC's standard announcement and distribution procedures may prevail.b. ☐ Recommend that there be no public announcement or distribution.SUBMITTED
BY(Please print
or type)

NAME

John William Esch

ORGANIZATION

University of Illinois

TITLE

Research Assistant

DATE

January 6, 1967

SIGNATURE

INSTRUCTIONS

A. General

AEC Procurement Regulation 9-4.5109-8, as revised, requires that a copy of the following types of documents generated by university type offsite contractors be submitted to AEC's Division of Technical Information Extension (DTIE), P.O. Box 62, Oak Ridge, Tennessee 37830:

Reprints

Manuscripts of articles submitted for journal publication

Manuscripts of papers intended for oral presentation (when a written version exists)

Annual and other progress reports.

These should be sent to the contract administrator for transmittal to DTIE. One completed copy of this form should accompany the first submission of each such document. *Exceptions:* This form need not be sent with reprints. The contract administrator may specify other changes in these instructions.

B. Instructions for Items on Form

Items 1 through 5. Self-explanatory.

Item 6. Each document must be identified by a report number. The first element in this number will be a code. Contractors who have applied for, and received, a unique report code should use this code. Other contractors should use the code of the responsible AEC Operations Office. The Operations Office codes are as follows: Albuquerque—ALO; Chicago—COO; Idaho—IDO; Nevada—NVO; New York—NYO; Oak Ridge—ORO; Richland—RLO; San Francisco—SAN; Savannah River—SRO. The second element should be the identifying portion of the contract number. The third element should be a sequential number assigned to the particular document only. *Example:* Technology College, Contract AT(11-1)-5000, administered by Chicago Operations Office, should number manuscripts and reports COO-5000-1; COO-5000-2, etc. The number should appear in the upper right corner of the first page of each document submitted to the AEC and in the acknowledgment section of manuscripts sent to journals or conferences.

Item 7. If this item is checked, DTIE will not at first announce or distribute the document. When journal publication has taken place, DTIE will feel free to announce the published version in *Nuclear Science Abstracts* (NSA). If DTIE has not found evidence of journal publication within eight months, the author(s) will be contacted to determine exact publication status. If it is determined at that time that journal publication will not be forthcoming, DTIE will consider whether the document should be publicly announced or distributed. In each such case a definite agreement will be reached with the author(s) before any action along these lines is taken by DTIE.

Item 8. If this item is checked DTIE will withhold announcement or distribution of the document at least until oral presentation has taken place. If item 8.d. is checked DTIE will feel free after the date of the meeting to (1) announce the document in NSA, (2) distribute it to AEC contractors and depository libraries, and (3) make it available for public sale through the Clearinghouse for Federal Scientific and Technical Information. If item 8.e. is checked, DTIE will take no action along these lines unless a definite agreement is reached with the author(s).

Item 9. If item 9.a. is checked, DTIE will feel free to (1) announce the document in NSA, (2) distribute it to AEC contractors and depository libraries, and (3) make it available for public sale through the Clearinghouse for Federal Scientific and Technical Information. If item 9.b. is checked, DTIE will take no action along these lines unless definite agreement is reached with the author(s).