MIT/LCS/TR-366

THE CLOCK DISTRIBUTION SYSTEM OF THE MULTIPROCESSOR EMULATION FACILITY

Saed G. Younis

June 1986

The Clock Distribution System of the Multiprocessor Emulation Facility

by

Saed G. Younis

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science in Electrical Science and Engineering

at the

Massachusetts Institute of Technology

June 1986

Signature of Author ...Saed...Younis.............................

Department of Electrical Engineering and Computer Science

May 19, 1986

Certified by ....

Arvind

Thesis Supervisor

Accepted by ....

Chairman, Departmental Committee

The Clock Distribution System of the Multiprocessor Emulation Facility

by

Saed G. Younis

Submitted to the
Department of Electrical Engineering and Computer Science
on 13 June 1986 in partial fulfillment of the requirements
for the Degree of Bachelor of Science in Electrical Science and Engineering

Abstract

Consisting of 32 high-speed processors, the multiple processor emulation facility communicates data between its processors through the use of synchronous, high-bandwidth packet switches residing on the ports of every processor. Because of the synchronous nature of these packet switches, there was a need to design a clock distribution system that can distribute a clock signal to the 32 ports with as little clock skew as possible. The problem was further complicated by the fact that the ports could be anywhere between 4 to 40 feet apart. In addition, this clock distribution system had to meet certain electrical isolation criterion so as to prevent a faulty port from affecting the whole facility.

The clock distribution system achieved its isolation goal by using pulse transformers on the clock lines. It solved the phase error problem by incorporating programmble delay lines on every clock branch. Calibration is done by connecting a calibration probe to a branch. The probe measures the phase error with respect to a reference and informs the system of the required corrective action.

Acknowledgments

Thanks to the DataFlow group under the direction of professor Arvind for providing the resources for this system. Special thanks to Gregory Papadopoulos and Robert Iannuccci for their advice throughout the development of the system. Thanks to Sheeroy Desai, Frederick Herrmann, and Andrew Braunstein for their help in making the implementation of the clock distribution system a reality.

Table of Contents

Table of Figures

Chapter One

Problem Description

1.1 Introduction

The clock distribution system is a component of the Multiprocessor Emulation Facility (MEF). This facility contains 32 processor elements (PE) that communicate data among themselves through the use of high-bandwidth circuit switches. The facility contains 32 circuit switches, one for every PE. To function properly, these circuit switches need to be synchronized with respect to a common clock. The Clock Distribution System (CloDS) is the component of MEF that is responsible for providing the clock signals for the 32 circuit switches.

1.2 Design Goals

The Clock Distribution System was designed to achieve the following goals:

1. CloDS must have a maximum distributed clock frequency of at least 4MHz. However, this goal was increased to 7MHz during the design of the system;

2. CloDS must have a maximum clock skew of +/-10 nanoseconds. In other words, the maximum phase difference allowed between any two clocks, measured at the circuit switch end, must not exceed 10 nanoseconds in either direction, leading or lagging;

3. CloDS must possess total electrical isolation between the main distribution board and the circuit switches of the respective PE's. Electrical isolation was not explicitly specified in terms of volts. The

rational for it, however, was that CloDS ought not to propagate a failure of one PE, such as a ground fault, to any other PE;

4. CloDS must allow the user to select the operating frequency from D.C. to the maximum frequency with small increments. No explicit limit was given for the maximum allowable jump between two consecutive frequency selections, but a maximum of a 2 percent increment was adequate;

5. CloDS must have a long mean time between failures, as well as a short mean time for repair. Because of the nature of the Multiprocessor Emulation Facility, it is very important that CloDS be able to operate for long periods without recalibration. It is also important that the mean calibration time be short;

6. CloDS must be able to send synchronous messages to any PE or a common message to all the PE's in addition to the clock signal. These messages would include such commands as master reset and global halt;

7. CloDS must be portable. In other words, CloDS must not depend in its basic operation on the condition of a specific machine. In the event that the machine that CloDS is plugged into fails, CloDS must be able to perform its basic operation of distributing the clock, after being installed in another machine, without the need for downloading the controlling software;

8. CloDS must support In-line performance logging and transparent recalibration. Because no knowledge is available on the performance of CloDS, it is important to keep a record of performance and drift to give an insight on the mean time between failures without bringing the facility down;

Chapter Two

Design Overview

2.1 Introduction

This chapter provides a general overview of the overall solution. It contains a description of the strategy used to achieve the main goal of distributing the clock. It also contains descriptions of solutions to some of the system's interesting subproblems, as well as some insight into the rational followed in making some of the design decisions.

2.2 General System Overview

Even with the fastest available components, the differences in propagation delays among different units exceed the maximum allowed skew of 10 nanoseconds. For this reason, building CloDS by duplicating clock branches without calibration is unwise. Furthermore, building CloDS without allowing for future recalibration is equally unwise. Consequently, CloDS was built with the understanding that initial calibration and future recalibration are both inevitable.

Calibration, or recalibration, of any system is a process by which the system is put in a feedback loop. The output of the system is compared to a reference and the error is processed and fed back to the input of the system in a corrective manner. This is basically what CloDS is, a large feedback loop (Figure 2-1).The main board of CLoDS has 32 clock branches leaving it. In addition to the 32 clocks that exit the main board of CloDS is a reference clock. This reference clock goes directly to a calibration probe. The calibration probe is a unit which is physically separated from the main board but is logically connected to it through a communication link. To

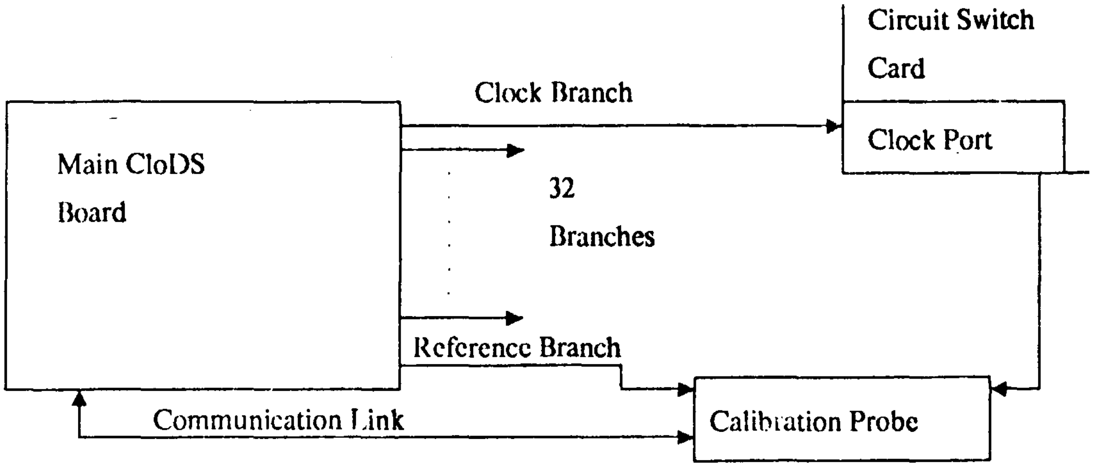


Figure 2-1: The feedback loop of CloDS.

calibrate a clock out of the 32 clocks the calibration probe is connected to that clock at the circuit switch side. The probe then measures the phase difference between this clock and the reference clock that it has. The probe also determines the direction of the difference, lead or lag. Then the probe sends this data to the control circuit of the main board. The main board adjusts the delay on the branch from the clock tree that corresponds to the clock that is being calibrated. The main board performs the adjustment so as to correct for the phase error and instructs the probe to perform the measurement again. This cycle repeats until the error reported is within the allowable bounds.

The concept used here is very similar to a phase-locked-loop. There are two differences, however. In a phase-locked-loop the phase is adjusted by varying the frequency of the clock. In CloDS the phase is adjusted by using programmable delay lines. The second difference is that, in phase-locked-loops, the loop is always closed whereas in CloDS the loop is closed only for the clock branch that the probe is connected to.

2.3 System Subproblems

The following subsections will describe the solutions devised for some of the subproblems of the system.

2.3.1 Clock Electrical Isolation

For CloDS, standard digital line drivers cannot be used because of the poor isolation they would offer between the main CloDS board and the circuit switches that CloDS drives. Instead, two alternatives were investigated. The first option was to use pulse transformers on the driving end of the clock cables. The second option was to use optocouplers at the receiving end of the clock cables.

Pulse transformers have the advantage of short propagation delay. They also have short rise and fall times. They are not very sensitive to temperature variations. In addition, pulse transformers can operate at very high frequencies and are physically robust.

Optocouplers, on the other hand, have the advantage that they are fully digital. They require no additional circuitry to interface them to digital circuits. Most importantly, optocouplers are non-dispersive within their operating ranges; they do not change their propagating characteristics with frequency.

In CloDS pulse transformers were used. The decision to use pulse transformer was because of a design policy that was carried throughout the development of CloDS. This policy stated that to minimize the drift in propagation delay that components exhibit over time, one should use components with low maximum propagation delay. For example, if the maximum propagation delay of a component is about 5 nanoseconds, then it is safe to assume that the propagation delay of this component cannot change over time and due to environment variations, such as temperature fluctuations, by more than 5 nanoseconds if it is kept within its operating conditions. The optocoupler examined had a maximum propagation delay of 50 nanoseconds, while the pulse transformers had a delay of less than one nanoseconds. In addition, the long rise time of the optocoupler might trigger the receiver prematurely because of noise. The decision to use pulse transformers was also influenced by the higher ceiling that they impose on the maximum operating frequency.

2.3.2 Clock Encoding Scheme

From the previous section we saw that CloDS uses pulse transformers on its clock branches. Pulse transformers are relatively dispersive, which is a problem because CloDS is supposed to operate over a wide range of frequencies. To solve this problem, the clock signals are converted from a square wave to a stream of pulses before being sent through the pulse transformers (Figure 2-2). Naturally, the reverse must be done at the circuit switch end to recover the original square wave shape of the clock.

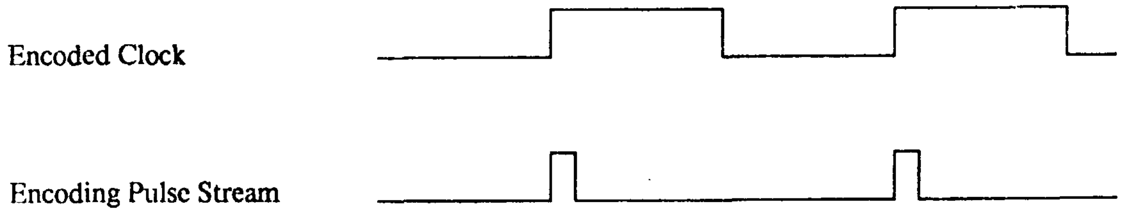


Figure 2-2: Clock encoding scheme.

The dispersive nature of the pulse transformer is due to the fact that transformers are non-linear elements. They pass only that part of the siganl that is varying with time. If a signal has a time section in which it doesn't vary, such as the flat part of a square wave, this part of the signal will be attenuated by the transformer (see photo of Figure 2-3). This is because the B field inside the transformer must be varying with time to induce any voltage in the secondary winding.

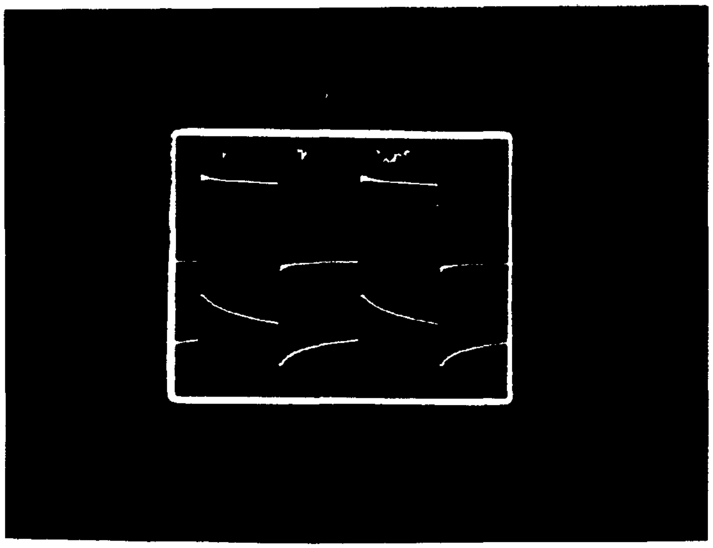


Figure 2-3:Attenuation of a 1MHz square wave through a pulse transformer.

If we place a heavy load on the secondary side of the transformer, like 27Ohms, we note that the output of the transformer will begin to resemble that of the square wave at its input (see photo of Figure 2-4). This can be explained by the fact that with a low resistance at the secondary, the induced voltage will drive a high current through the secondary winding. When the input voltage of the transformer stops varying with time, as in the flat part of a square wave, the transformer's secondary winding starts looking like an inductor. With a high current flowing through it, the square shape is preserved because of the long time it takes for this current to decay. Note from the photo, however, that the flat part of the output square wave is not horizontal but decaying. Note also that at such a high secondary load, the output voltage is much lower than the input voltage.

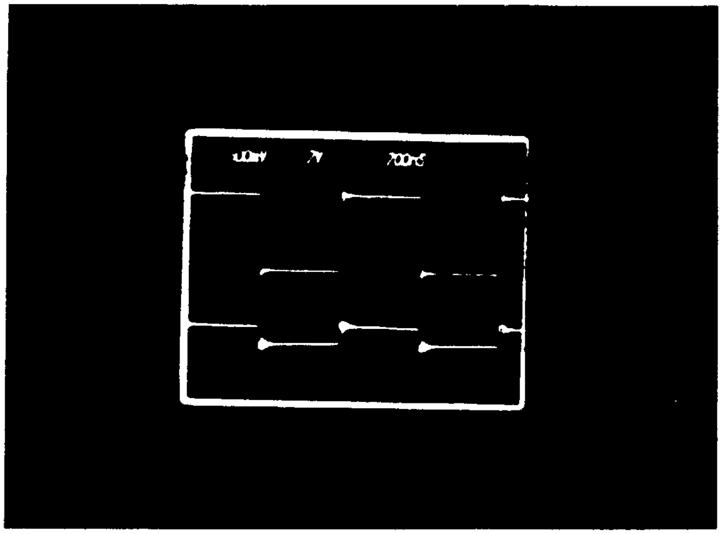


Figure 2-4:Attenuation of a square wave with a high secondary load.

Narrow pulses on the other hand, have no flat parts except when they are at zero volts. This means that the output voltage- will always resemble the input voltage including the flat parts of the signal, since they are at zero volts to begin with (see photo of Figure 2-5). In addition, narrow pulses will continue to pass through the transformer relatively unattenuated to very low frequencies. Attempting to do the same with square waves will fail.

2.3.3 Pulse Drivers

The cables that carry the clock signals are properly terminated on both ends to eliminate reflections. An undesirable effect of the cable impedance matching on the driving end is that it divides the voltage of the cable drivers in half. This means that the voltage swing of the cable driver should be at least double the voltage swing needed by the receiver on the other end. In CloDS, the clock receivers, called ports, need a minimum voltage swing of about 3.4 volts for their proper operation. This requires that the cable drivers have a voltage swing of at least 6.8 volts. In addition,

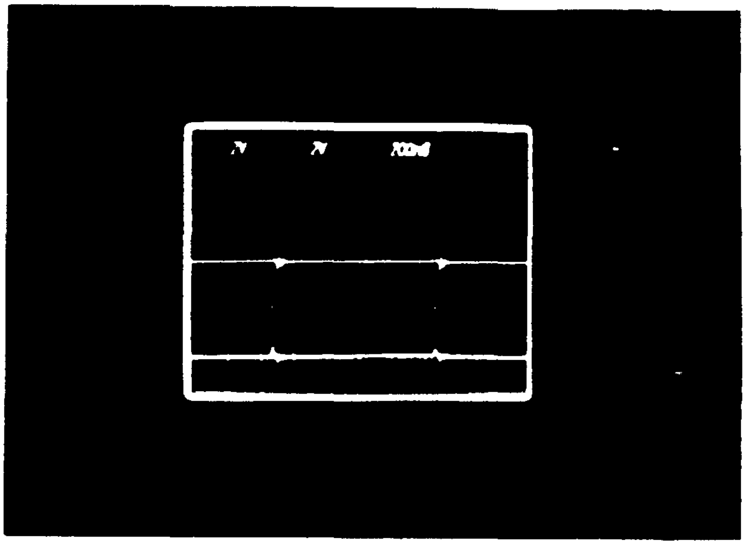


Figure 2-5:Narrow pulses through an isolation transformer.

the cable drivers should provide some extra swing to account for the IR voltage drops across long clock cables. This voltage swing is above that of fast digital drivers, typically 4.0 volts.

A first-cut solution to this problem was to use operational amplifiers to increase the voltage swing of the clock signals. The output of these operational amplifiers would then be fed into voltage followers that would provide the fan-out that is required for every clock cable. A prototype of this solution was constructed and tested. Although this method was eventually abandoned, the constructed prototype offered some valuable insight into the viability of pulse-coding the clock. (See section 6.2 for a detailed description of the implementation of this solution and for the results of the tests performed on it.)

A more elegant solution to the voltage swing constraint was to use pulse transformers that had a secondary to primary ratio of more than unity. Instead of using operational amplifiers and voltage followers on every clock branch, we can use standard digital drivers and step up their voltage swing by using pulse transformers with a 1:2 winding ratio (Figure 2-6). The problem here is that our digital driver must now be able to drive a load which is double the load it had previously. Here, CloDS needed a line driver that can drive a 100 mA load, a driver that has a small propagation delay, and one that has fast rise and fall times. Relief came when the 74F3040 chip was introduced. This chip has a sourcing capability of 50mA and a sinking capability of 160mA. It has a maximum propagation delay of 5.5 nanoseconds and a rise and fall time of 2.5 nanoseconds.

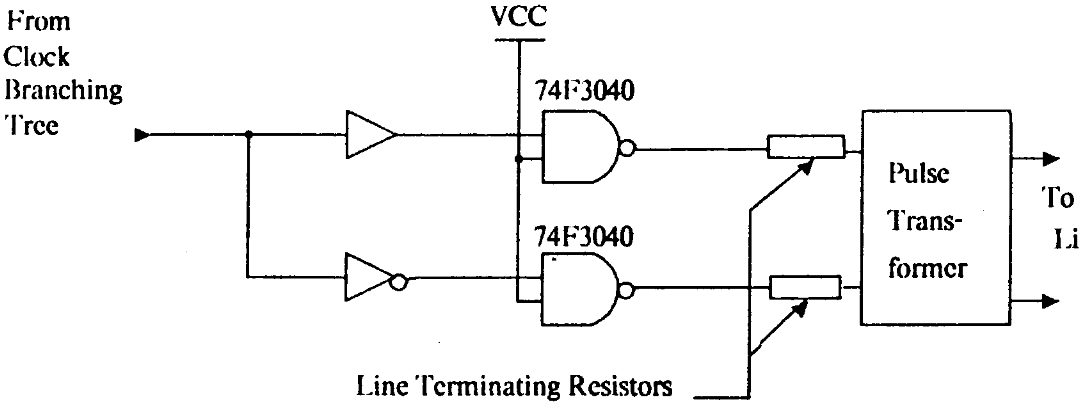


Figure 2-6:The digital line driver.

With the introduction of the 74F3040 chip, the advantages of the second solution were overwhelming when compared to the first one. These advantages were

1. The 74F3040 were completely digital and did not require special supply voltages.

2. The fast voltage follower of the first solution consumed a lot of power and generated a great deal of heat. It radiated enough heat to require

special cooling.

3. The voltage followers where bulky, whereas the 74F3040 came in 16-pin DIP packages.

4. The voltage followers were costly, about $35 per unit.

A prototype of a cable driver using a 74F3040 was constructed. This cable driver demonstrated that it can drive a 93ft cable without signal degradation on the other end of the cable. (See section 6.3 for a detailed description of the digital line driver prototype.)

2.3.4 Clock Recovery Circuit

The waveform of the clock signal as it reaches its respective port is a stream of narrow pulses, about 15ns wide. The port must therefore restore the original square wave shape of the clock signal before it feeds the clock to the circuit switch. Encoding the clock signal as a stream of narrow pulses carries two options. The first option is to let one pulse represent the occurrence of the rising edge of the clock, and to let the next pulse represent the falling edge of the clock (Figure 2-7). The other option is to represent the rising edge of the clock with a pulse and to ignore when the clock falls (Figure 2-8).

For the first option, the added complexity is justified if the phase differences among the falling edges of clocks on different ports are as important as the phase differences among the rising edges. The added complexity of the first option, encoding both the rising and falling edges of the clock, results because it is possible for one port to be exactly 180 degrees out of phase with the rest of the ports. This happens because, while one port interprets a pulse as an indicator of the rising edge, another port may interpret it as an indicator of the falling edge. One can solve this problem if one exploits the fact that CloDS has a maximum operating frequency. If

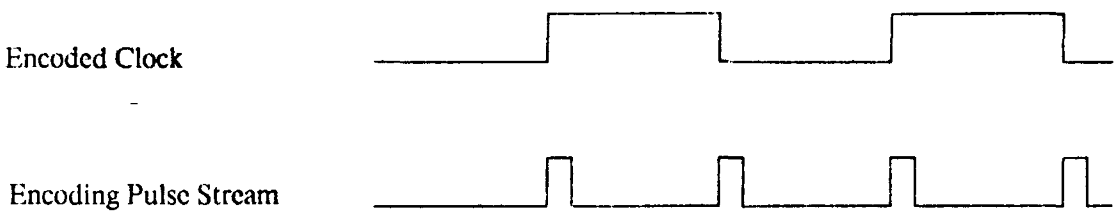


Figure 2-7:Rising and falling edge encoding.

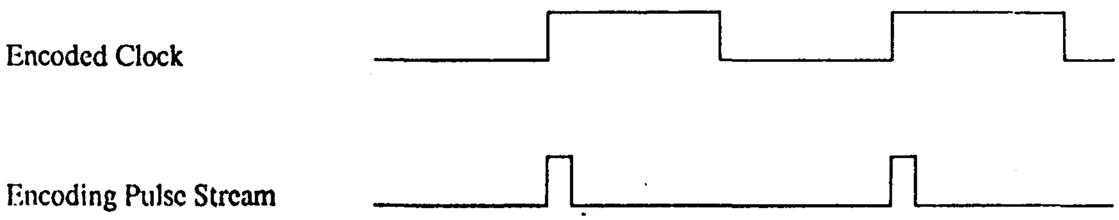


Figure 2-8: Rising edge encoding.

we assume that CloDS has a maximum frequency of 10MHz, then the minimum duration between two consecutive pulses cannot be less than half the period, or 50ns. We can use this to build a port that will force the clock low on the next incoming pulse if the current pulse followed the previous one by less than 50ns. In other words, if the pulse the port sees follows the previous one by more than 50ns then the port uses it to toggle the state of the output clock. If, on the other hand, the pulse follows the previous one by less than 50ns, then the port enters a special state in which it interprets the next pulse as a falling edge indicator regardless of the present state of the clock (Figure 2-9). Pulses that are less than 50ns apart from the previous ones are called synchronizing pulses.

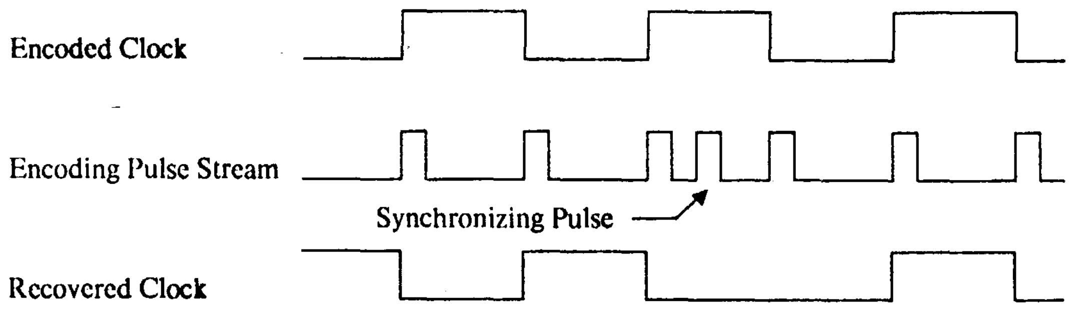


Figure 2-9:Clock synchronization.

Initially, CloDS was expected to deliver clocks with synchronized rising and falling edges. The constraint on the falling edges, that they be synchronized, was eventually dropped during the development of CloDS. A proposed port circuit that recovers clocks encoded with two pulses and recognizes synchronizing pulses according to the above strategy is included in section 6.6.

With the constraint on the falling edges dropped, recovering the clocks at the ports was simplified greatly. A simple single shot monostable multivibrator can do the job nicely. The monostable is triggered by the incoming narrow pulses and resets itself after a predetermined high time. The only constraint is that the high time of the monostable must be less than the minimum clock period of CloDS. Please note however, that this single shot is further complicated by the message decoding circuitry discussed in the next subsection.

2.3.5 Message Encoding and Decoding

The method of sending encoded synchronous messages on the clock cables without affecting the timing of the clocks came as a side benefit from the work that was done to synchronize both the rising and the falling edges of the clocks. With no need to use the synchronizing pulses to correct for the possibility of an inverted clock, because only the rising edge mattered now, the synchronizing pulses can be used to send the synchronous messages. If a pulse follows another by more than 50ns, then the port interprets it as a clock pulse and the monostable is triggered by it. If the pulse follows another by less than 50ns, then the port interprets it as a message pulse and the monostable is forced to ignore it.

The transparent message pulses provide a serial channel between a port and the main board of CloDS. The presence of a message pulse between two clock pulses is a mark, while its absence is a space. In CloDS, a mark is a logical 1 or a high. A space is a logical 0 or a low. On the main board of CloDS a byte is loaded into the message encoder. The encoder first adds a high bit header to the byte and then sends the nine bits on the clock lines as a series of marks and spaces. The message decoders of every port detect the header bit and then assemble the marks and spaces into a byte. The message decoders then inform their circuit switches that a message has been received.

Because the message pulses are inserted between the clock pulses, and because the clock pulses are adjusted to have little phase differences, these messages arrive at the different ports at the same instant. As a result the message decoders on different ports present the message to their circuit switches at the same instant also. Therefore, these messages are fully synchronous and can be used to convey such commands as global resets and synchronous global halts.

2.3.6 Clock Cables

Fiber optics were ruled out as a transmission medium for the clock signals for the same reasons cited against the use of optocouplers in section 2.3.1. Fiber optics are also costly and relatively fragile.

With electrical cables, there are two methods of transmission. The first is the unbalanced method. Here the signal detected at the receivers is measured with respect to the signal ground. The second is the balanced method. Balanced transmissions travel over two conductors. The signal on one conductor is always equal and opposite to the signal on the other conductor. The signal is detected by measuring the difference in voltage between the two conductors with disregard to their common voltage with respect to ground. CloDS uses balanced transmission in all of its clock cables. CloDS uses balanced transmission for all of its probe-to-main-board communication links also.

The main reason for using balanced transmission is isolation. The specifications of CloDS prohibit any direct electrical connection, including signal ground, between the main board and the 32 ports. By using balanced differential drivers and receivers, and by using isolation transformers on both sides of the clock cables, the D.C. component of a specific cable can float without any damage to CloDS or the ports. In addition balanced transmission is immune to both ground noise and to electromagnetic noise present in the environment. Noise coupled to one wire will couple to the other and will cancel it at the receiving end. Moreover, balanced transmission does not contribute to the ground noise because it presents a relatively constant dynamic load to the power supply of the drivers.

In considering electrical wires, one has to worry about signal reflections resulting from improper termination of the line. This limits the selection of cables to ones of known impedances. In addition, the selected cable must have a high bandwidth, and a low dissipative resistance. Typical choices in this category are coaxial and twisted pair cables. Coaxial cables are usually used in unbalanced transmissions while twisted pair cables are typically used in balanced transmissions. Twisted pair cables were used throughout CloDS because it was available in ribbon form which can be mounted to connectors more easily. Initially, the prototypes of CloDS used shielded twisted pair cables; however, unshielded twisted pair cables were used on the final system without any noticeable degradation in performance.

2.3.7 The Main Oscillator

Initially, the main oscillator was considered to be an external unit to CloDS. A waveform generator would be mounted on top of the machine that CloDS is plugged in, and a coaxial cable would feed the signal into the board. The frequency of the oscillator would be varied by CloDS by talking to the waveform generator through a GPIB interface between the the main board and the waveform generator. This idea was dropped in favor of including the main oscillator on the main board. This gave rise to some difficulties, as outlined below.

The specifications of CloDS dictate that the frequency of the main oscillator be programmable. Programmable oscillators come from two main categories. They are, analog programmable oscillators and digital programmable oscillators. The simplest analog programmable oscillator is a voltage-controlled-oscillator (VCO). Digital selection of the frequency of this VCO is done through the help of a digital-to-analog converter (DAC). The DAC will convert a digital word to a voltage and feed that voltage to the VCO. Presenting the DAC with a different word results in a different oscillator frequency (Figure 2-10). On the other hand, a basic digital programmable oscillator consists of a fixed-frequency crystal oscillator and a programmable frequency divider. The frequency of this type is determined by programming the divider. The user loads the divider with a number that corresponds to the result of dividing the period of the desired output frequency by the period of the crystal oscillator. In other words, this is the number of the crystal oscillator periods that can fit within one period of the desired output frequency. Presenting the divider with a different number results in a different output frequency (Figure 2-11).

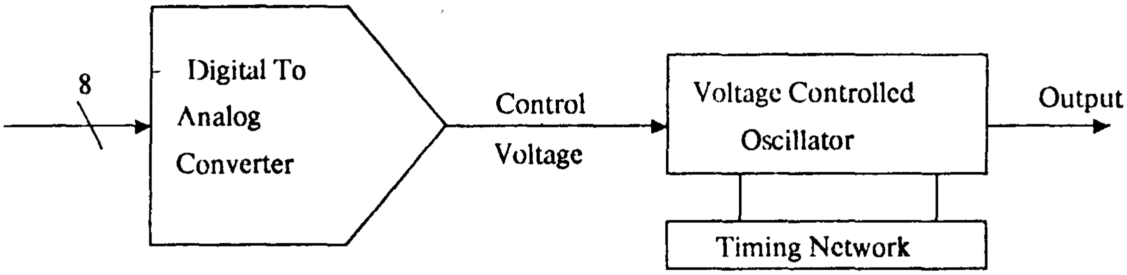


Figure 2-10:A simple programmable analog oscillator.

The advantage of using the analog oscillator is that the distance between two frequency increments is as small as one wishes to make the DAC steps. A 16-bit DAC will result in extremely fine frequency increments. The analog oscillator is also relatively simple. Unfortunately, a programmable analog oscillator is not very stable. Its frequency will vary with temperature, supply voltage and time due to component degradation. This is a high price to pay and is unacceptable for CloDS.

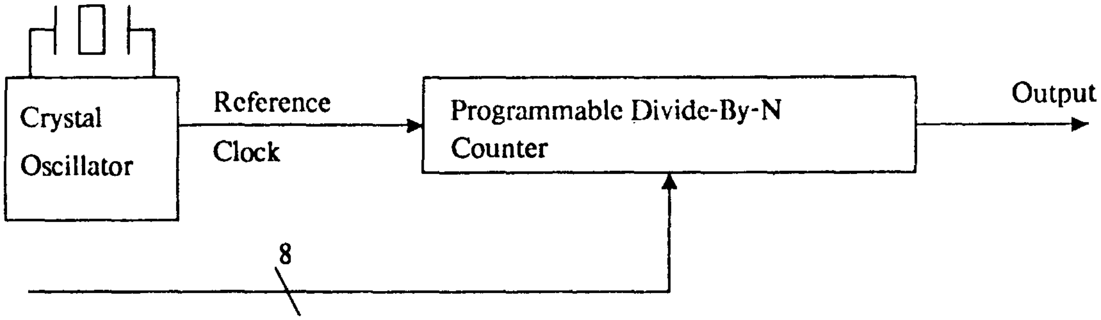


Figure 2-11:A simple programmable digital oscillator.

The digital programmable oscillator is a crystal controlled oscillator. Therefore, its frequency is very stable and does not vary with variations in the operating environment. The digital oscillator does not need any calibration either. The problem with a digital programmable oscillator is that it does not allow for fine selection of frequencies at the high end of its frequency range. For example, consider a-digital programmable oscillator with a reference crystal clock of 10MHz, or a period of 100ns. This means that the user can vary the period of the programmable output frequency by 100ns in either direction. If the current frequency of operation is 100KHz, then decrementing the period by 100ns changes the frequency to 101KHz. This is a 1 percent increment, which is quite good. If the frequency of operation is 3.3MHz, however, then decrementing the period by 100ns changes the output frequency to 5.0MHz. This is about a 30 percent increment and is unacceptable. For CloDS, fine frequency selection is more important at high frequencies than it is at low frequencies. This is because the speed of CloDS determines the speed of the circuit switches. If the circuit switches can operate at 7.0MHz but cannot keep up at 7.1MHz, then CloDS should be able to operate at 7.0MHz. In our example above, the oscillator's frequency jumps from 5.0MHz to 10.0MHz in one increment. Hence, one is forced to operate at 5.0MHz. An oscillator such as the one in the example will be responsible for foregoing potential faster operation of the whole Multiprocessor Emulation Facility (MEF). The period increment of 100ns can be made smaller by using a reference crystal oscillator with higher frequency. Unfortunately,the frequency needed to achieve the desired increment will be higher than what common current digital components can handle without using custom boards and components (the frequency needed is higher than 100MHz!).

We have seen so far that the instability of an analog oscillator makes it unacceptable for CloDS. We have also seen that a digital oscillator is also unacceptable because it does not have fine frequency selection where it counts, at high frequencies. However, all is not lost. The oscillator of CloDS is a hybrid between an analog and a digital oscillator combining the best of both world. The oscillator consists of two main sections. The first is a programmable digital oscillator exactly like the one in the example above. This oscillator uses a 10MHz reference clock. The output of this oscillator is then fed into an analog frequency multiplier (Figure 2-12). This multiplier multiplies the frequency 100 times. The multiplier is built out of a phase-locked-loop with a divide-by-a-hundred scaler in the loop. To get a frequency of 1.0MHz, the user programs the digital oscillator to output a 10KHz signal. This is multiplied by a hundred and the desired 1.0MHz comes out. The benefit of this hybrid oscillator is that, while the period of its digital oscillator varies in increments of 100ns, the period of its output coming out of the multiplier varies in increments of 1ns. This means that the user can select the frequency of the main oscillator with increments of 1 percent in the worst case. In addition, when the phase-locked-loop is in lock, the stability of this hybrid oscillator is as good as that of the crystal-controlled oscillator that it is locked to.

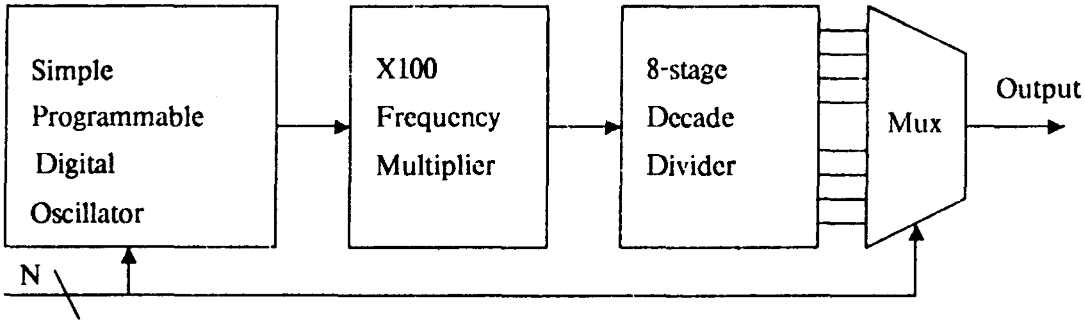


Figure 2-12:The hybrid oscillator.

2.3.8 Phase Error Measurement

The calibration probe of CloDS is the piece that measures the phase difference between the reference clock and the clock at the port it is connected to. Two ways of determining the phases were tried. In both cases a phase detector compared the edge of the reference clock to the the edge of the clock under test. The phase detector then produced a signal that had a pulse width that was equal to the phase difference between the two clocks. The difference between the two tried methods was the way by which the detected phase difference was converted to a digital number that is suitable for transmission to the main board.

In the first method, the error pulse was integrated in a capacitor and the resulting voltage was locked in a sample-and-hold amplifier. The output of the amplifier was then sent to an analog-to-digital converter (ADC) that converted this voltage to a digital word. Larger phase differences resulted in wider error pulses. Wider error pulses resulted in higher voltages when integrated, and corresponded to higher digital numbers from the ADC. This is different from the way a phase-locked-loop uses the error signal. In a phase-locked-loop, the voltage value that results from the phase detector is a result of integrating the stream of pulses over time.

In CloDS, the error voltage is the result of integrating a single error pulse only. CloDS uses only the integrand of a single error pulse because it operates on a wide range of frequencies. If the integration involved the average of the stream of error pulses, then higher frequencies would result in a higher integrand value. The higher error voltage will not be due to more phase error, but will be a result of more error pulses per second for higher frequencies than there are at lower frequencies. Because it will then be difficult to determine which part of the error voltage is due to a phase error and which is due to higher operating frequency, only one pulse is integrated for every error voltage sample.

A prototype of this phase detector was built and tested. This detector failed completely. It was very sensitive to supply-voltage transients and did not give consistent numbers. The problem came from trying to integrate a single pulse, typically 30ns wide. The pulse did not have enough energy to survive the sampling capacitor leakage currents. Section 6.5 includes a detailed description of the constructed circuit and the possible reasons for its failure.

The second method used programmable digital delay lines to measure the width of the error pulse that the phase detector had produced. If the pulse width of the error pulse was wider than the programmed delay, then the probe would increment the programmable delay lines and would try the comparison again. Eventually, the delay of the delay lines approximated the width of the error pulse. The probe then sent the digital word that was presented to the delay lines to the main board. A prototype of this converter showed that such a method was accurate enough to measure the width of the error pulse to within +/- 1.5ns.

It follows, then, that CloDS uses the second method to detect the width of the error pulse (Figure 2-13). An error pulse arrives at point A from the phase detector and triggers the first flip-flop (F1). It also goes directly to one of the AND gate inputs. The output of F1 goes trough the delay lines and exits at point B Xns later. The X here is the programmed delay value. If the delay is greater than the width of the error pulse, then the output of the AND gate never goes high and, consequently, F2 is never triggered. F2 is triggered, however, if the delay was less than the width of the error pulse. By varying the delay and monitoring the output of F2, the probe is able to tell how wide the error pulse is.

To reduce the time of trial and error, the probe changes the delay in a way similar to that used in the successive approximation logic that is common in analog-to-digital converters. On every trial the probe is, hence, able to half the guess range, thus resulting in a worst case that is of the order of *log(n)* instead of *n*, with *n* being the size of the measurement range.

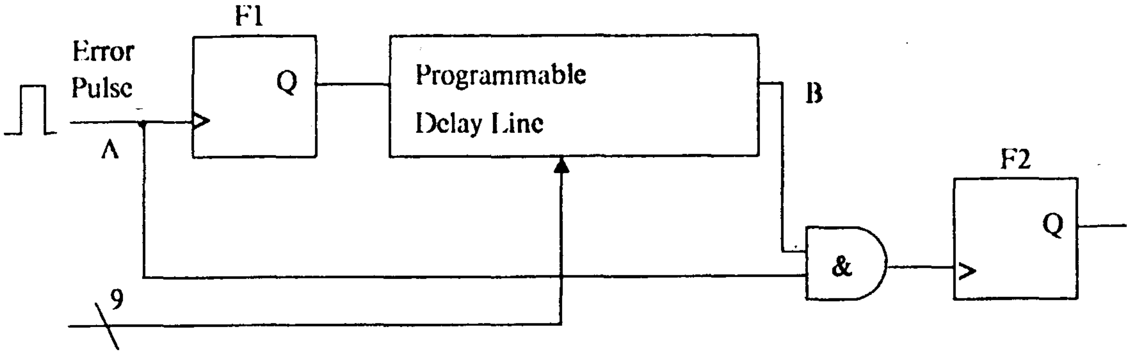


Figure 2-13:A pulse-width detector similar to the one used in CloDS.

In CloDS, the phase detector that generates the error pulse is actually more complicated than mentioned above. In reality, the phase detector of CloDS generates two different error pulses instead of just one. One error pulse corresponds to the time by which the reference clock **leads** the clock under test. The other error pulse corresponds to the time by which the reference clock **lags** the clock under test (Figure 2-14). Please note that one error pulse might be measuring the phase difference between two different clock pulses. This is all right because the control circuit of the probe knows the frequency of the clocks and can therefore ignore the false error pulse. In addition to the above, the phase detector of CloDS always adds 60ns to both the reference and the clock before it generates the error pulses (Figure 2-14). Without this addition, the width of one error pulse, the right error pulse, will approach zero as CloDS aligns the clock with the reference clock. This increases the probability that an element of the probe will go into a metastable state. In addition, a very narrow error pulse will not trigger the F2 flip-flop mentioned above.

2.3.9 Probe Communication Link

While in use, the calibration probe needs to communicate with the main board of CloDS. The probe sends information, such as the phase error, to the main board and

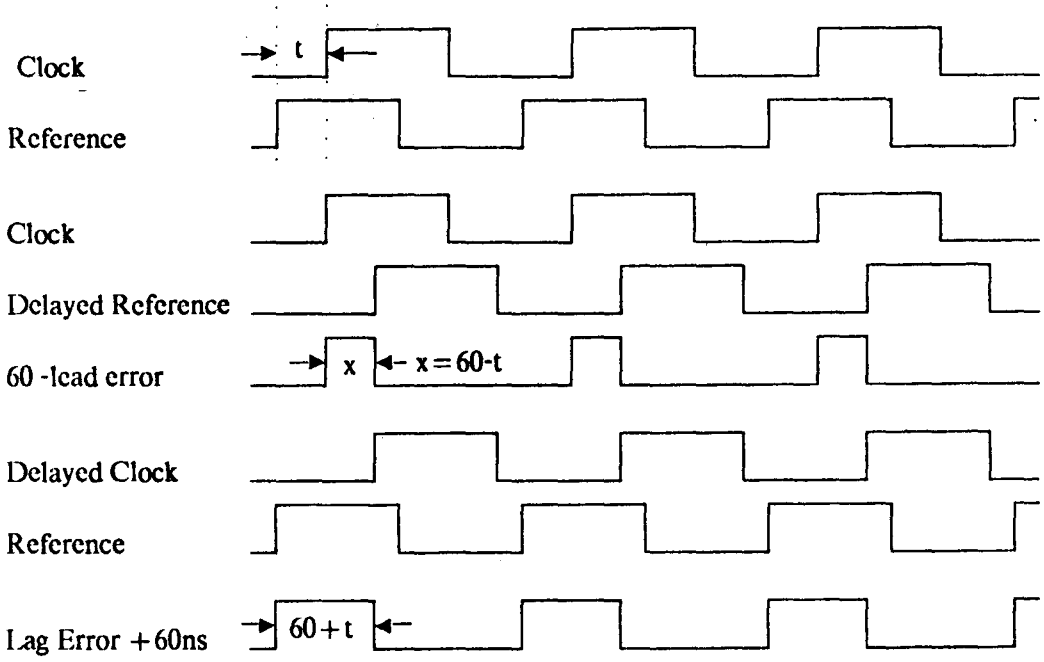


Figure 2-14:Timing diagram of the error pulse.

receives information ,such as the clock frequency, form it. For this reason the communication link should be full-duplex. In selecting a method of communication, one has to determine what protocol to use and how to encode it on the physical medium. In CloDS, the protocol used is the RS-232. This protocol was choosen because it is well understood, and supported by an abundance of integrated circuits. The link must detect the presence of a fault on the communication lines and should have hardware support to automatically shut down the communication link whenever a fault is detected. In CloDS, a fault is defl00ined to be an open circuit or a short. Fault detection on the probe-main-board communication links is important for two reasons. The first is that an undetected short in the communication links can cause other elements of the system to fail. The second reason, and the most important one, is that these communication links carry critical information. For example, the probe-to-main-board link carries such information as the phase error of the clock that is being calibrated. If a fault occurs and CloDS does not know about it, CloDS may incorrectly adjust the delay of a clock branch, thus destroying the timing of that branch. In this case, a fault in the communication links resulted in loss of synchronization of the circuit switches and possibly the whole of (MEF).

In light of the need for fault detection, two methods of transmission were considered. The first was frequency-shift-keying (FSK) with carrier-loss detection. The second was to use differential line drivers and to detect a fault by measuring the D.C. level of the line. The first option was abandoned after a prototype of a FSK modem failed to operate at the required baud rates. The prototype demonstrated that to operate at the required baud rates, a minimum of 1200 Bauds, it had to be augmented with additional filters and overall additional complexity. The FSK method was known to have more complexity than the differential drivers from the start, but it was nevertheless pursued because a working FSK link can be made compatible with available modems and can then be used to connect CloDS to devices other than the probe. However, the prototype demonstrated that the needed circuit complexity would be a high price to pay for the desired compatibility and FSK was dropped. A description of the FSK modem prototype is included in section 6.4.

The differential line driver method, although lacking compatibility, satisfied the fault detection criterion, nevertheless. This method achieves fault detection, as follows. A differential line driver has two output pins that are connected to the line. The driver represents a 1 bit by driving the first pin high and the second low. The driver represents a 0 by driving the first pin low and the second high. Therefore, a potential will always be present at the receiving end of a healthy cable. If a fault occurs, such as a short or an open circuit, the receiver can detect it because no voltage will be present at the receiving end of the cable.[1]

2.3.10 Probe Circuit Breaker

In CloDS the calibration probe gets its power from the main board. The cable that carries the probe-main-board communication links also carries power to the probe. The previous section showed that fault detection cannot be over emphasized. For the same reasons, CloDS must be able to detect faults in the probe power lines. To achieve this fault detection, the main board contains a low-voltage circuit breaker. All power supplied to the probe passes through this circuit breaker. The breaker is initially programmed with the load that a healthy probe presents. The breaker continuously monitors the current passing through it. If the current exceeds that of a normal probe load, the breaker trips, suspecting a short on the probe power lines. In addition, if the current drops below that of a normal probe load, the breaker also trips, suspecting a loose connection on the probe power lines. The breaker also informs CloDS control of the fault that has occurred so that CloDS control can ignore any data it has received from the probe.

2.3.11 Portability

The specifications of CloDS dictate that the main board must be portable. In other words, the main board must be able to perform its main operation, of distributing the clock, with minimum software support from the machine it is moved to, if its host machine fails. To achieve this portability, the main board of CloDS contains a non-volatile memory. Whenever the host processor updates the contents of any register in the main board of CloDS, a copy is sent to the non-volatile memory. Therefore, this memory contains the contents of all the registers at the time of a

[1] The method was described in the National Semiconductor Interface Databook in section 1 pp. 108-109.

failure, including the values of the delays on every clock branch. Hence, when the board is moved to another machine, all that the new machine has to do is to read the contents of the non-volatile memory and write this information back to the respective registers. At this point, CloDS will begin to operate with the same frequency and branch delays as those present when the card was last powered down. However, CloDS cannot perform any calibration of a clock branch without first downloading the main program from the initial host machine.

Chapter Three

Hardware Description

3.1 Physical Organization

The clock distribution system consists of the following physically separated units.

1. One main distribution board.

2. One external probe power supply.

3. One calibration probe.

4. Thirty two clock recovery ports.

The main distribution board is a Nubus card and is plugged in a TI Explorer machine. The clock recovery ports are part of the circuit switch circuitry. Each circuit switch board has one clock recovery port circuit on it. A clock signal, generated at the main CloDS board, is split into 32 clock branches. Each branch is connected through twisted-pair cable to a clock recovery port on a circuit switch.

The calibration probe is only used during calibration or recalibration of a given clock branch. The calibration probe is connected to the main CloDS board whenever it is in use. The probe gets its power from the probe circuit breaker that resides on the main CloDS board. The external power supply was provided because the voltages supplied by the Nubus are not enough to drive this circuit breaker. The external power supply feeds power to the probe through the circuit breaker of the main board. The external power supply was made small enough to fit between the back door and the Nubus back plane of a TI Explorer.

A detailed schematics of CloDS is included in appendix A. Following are the details of each of these separate units.

3.2 The Main Board

3.2.1 General Organization

The largest section of the main board is that of the clock generation and branching logic. This includes the following.

1. The programmable oscillator that generated the master clock at the desired frequency.

2. The synchronous message encoder that encodes transparent messages on the clock lines.

3. The programmable delay lines that allow the alignment of the clock edges.

4. The line drivers and their isolation transformers.

5. A number of programmable control registers.

In addition, the main CloDS board contains Nubus and calibration probe support logic. Components of the support logic are

1. The Nubus interface logic.

2. The probe circuit breaker.

3. The probe serial communication logic.

4. A 6-letter debugging display.

3.2.2 The Programmable Oscillator

The output frequency of the oscillator is programmable in the range from 0.1Hz to 12MHz. In addition, the output frequency can be selected with a minimum of two percent increments over the entire range and a minimum of one percent increments in some ranges. This means that the period can be varied with one nanosecond increments in the range of 6-12MHz, for example. The oscillator can be halted and single stepped also.

The architecture of the programmable oscillator is illustrated in Figure 3-1. The first stage, the programmable divider, uses the 10MHz clock of the Nubus as a reference. The divider can be programmed to divide by any integer in the range from 9 to 1032. This gives a frequency range of 1.1MHz to 968.9KHz. The selection of the output frequency is restricted, however, to the range of 120KHz to 10KHz.

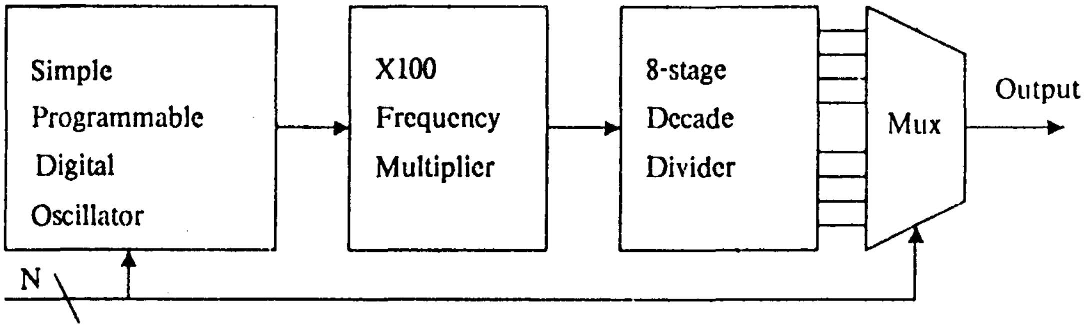

Simple Programmable Digital Oscillator
X100 Frequency Multiplier
8-stage Decade Divider
Mux
Output
N

Figure 3-1: The main oscillator.

The second stage is a X100 frequency multiplier, which produces an output frequency in the range of 12MHz to 1MHz. The multiplier is a phase-locked-loop

(PLL) with a divide-by-100 counter in the loop (Figure 3-2). The multiplier is more complex than a basic PLL because basic PLL's cannot maintain their lock over such a wide range of frequencies that stage one generates. The range of 120KHz to 10KHz has a ratio of 12:1 which is higher than any VCO can track. Although some VCO's allow a 5:1 ratio, they should only be operated at 2:1 ratio if stability is a main concern, as in CloDS. For this reason, the multiplier contains two PLL's. The first is tuned to track frequencies in the range of 2MHz to 4MHz while the second is tuned in the range of 6MHz to 12MHz. The output is selected to be the output of the PLL that is currently in lock.

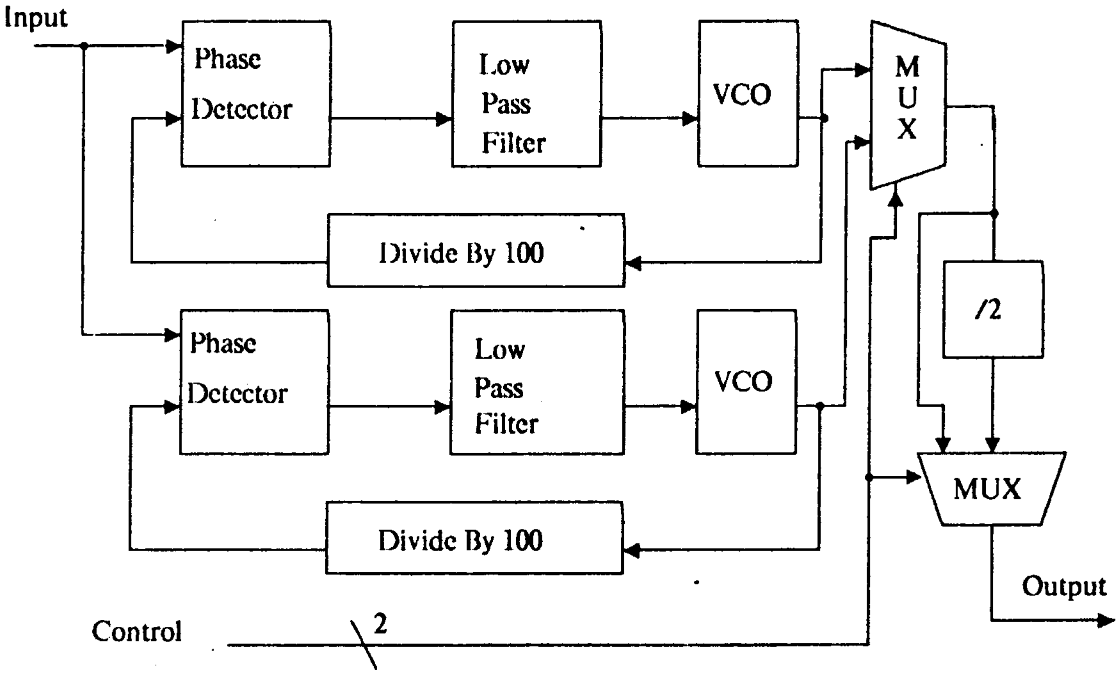


Figure 3-2: The phase-locked-loop of the multiplier.

To generate frequencies in the ranges of 1MHz to 2MHz, the first PLL is used in the 2MHz to 4MHz range and its output divided by two. The same is done with the second PLL for the range of 4MHz to 6MHz. The compromise here is that the output period can be incremented with a minimum of 2ns instead of the finer-grain selection of 1ns increments at the other ranges. This could have been overcome by including two more PLL's tuned to the *missing* ranges, but limited real estate on the main board prohibited this addition.

The third stage of the oscillator is a string of decade dividers and a selector. The output of the multiplier, which is in the range of 12MHz to 1MHz, is passed through a series of decade counters to make frequencies available anywhere in the range of 12MHz to 0.1Hz. The output of the selector can also choose the single-stepping input. This input gives a single clock pulse every time a certain control register is written to.

3.2.3 The Message Encoder

The function of the message encoder is illustrated in the timing diagram of Figure 3-3. The message encoder takes in the square wave produced by the main oscillator and produces a 25ns-wide pulse for every rising clock edge. It also inserts a 25ns-wide pulse about 50ns after a clock pulse if it is sending a message 1 bit.

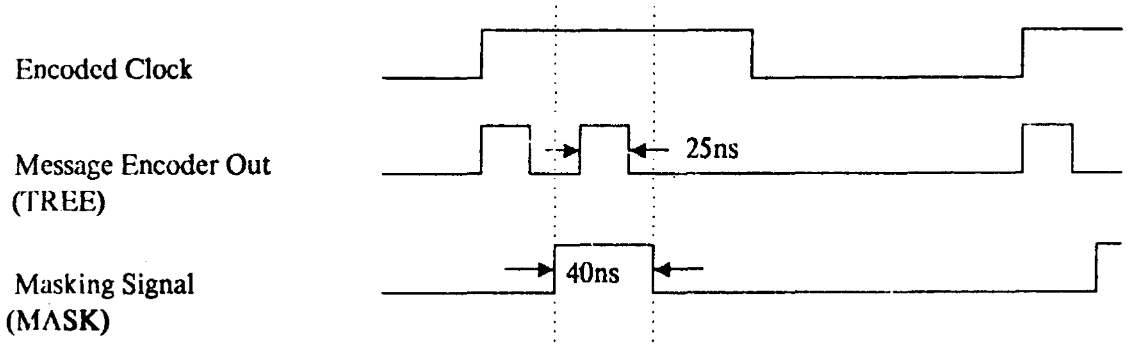


Figure 3-3: Timing diagram of the message encoder.

From chapter 2 we know that the encoded clock pulses are 15ns-wide, not 25ns. The pulses will be contracted at the line drivers of every branch. The message encoder does not produce the 15ns pulses because its output has to still go through the delay lines which put a limit on the minimum pulse width allowed through them before they start degrading the quality of the signal. These pulses are compressed to 15ns at the line drivers after they pass through the delay lines.

In addition to the clock and message pulse stream, the message encoder generates a masking signal (see Figure 3-3). Recall that CloDS must be able to selectively send a synchronous message to the ports. The masking signal is used by the clock drivers at the end of every branch to mask out the message and only pass the clock pulses whenever the message is not intended to their respective ports.

3.2.4 The Clock Branching Clusters

To compensate for the different cable lengths and propagation delays, CloDS uses programmable delay lines on every clock branch. Every delay line can take on one out of eight delays determined by a 3-bit number presented to it.

In CloDS, every clock branch goes through two programmable delay lines, coarse delay line with increments of 5ns and a fine delay line with increments of 3ns. While every clock branch has its own private fine delay line, the clock branches are grouped in clusters of eight with only one coarse delay line per cluster. The grouping is a result of scarce main board real estate (Figure 3-4). From the above the delay lines can compensate for a maximum phase error of 21ns between branches of the same cluster and a maximum of 56ns between branches of different clusters. At approximately 1.5ns/ft cable propagation speed, the clock cables must not differ in length by more than seven (7) feet within the same cluster and by not more than sixteen (16) feet within different clusters. This calculation is based on an optimistic maximum accumulative component propagation delay difference of not more than 10ns between any two branches. Suggestions for improving this limit are included in chapter 7.

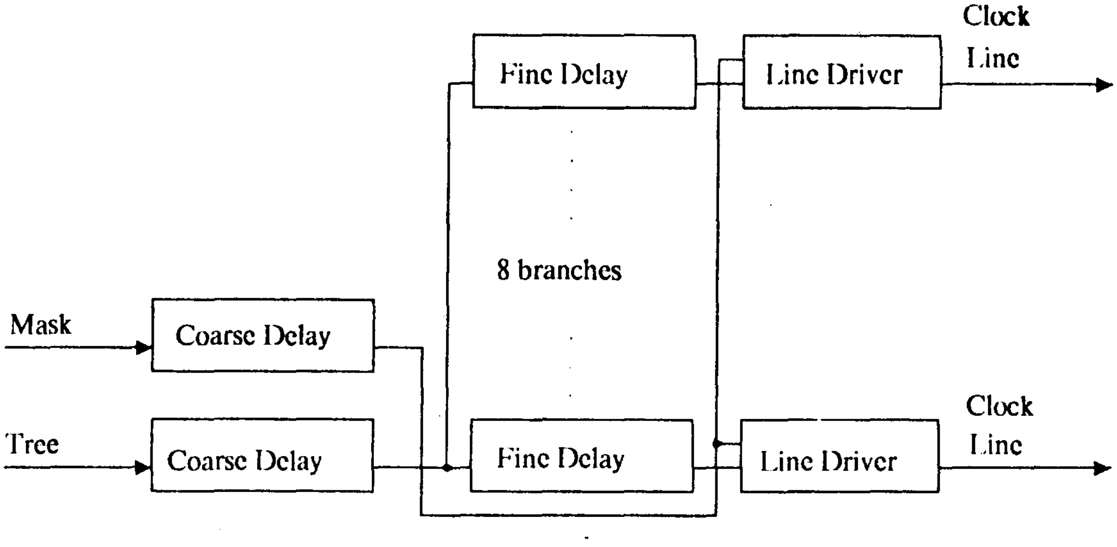


Figure 3-4: Block diagram of one branching cluster.

In addition to the coarse delay line that a cluster has for the clock stream from the message encoder, every cluster has a coarse delay line for the masking signal that comes from the message encoder. The value programmed in this coarse delay line in a cluster is the same value as that for the other coarse delay line in the same cluster. This preserves the timing relation between the masking signal and the encoded message pulses it masks.

At the end of every branch in a cluster lies a line driver. A line driver performs three functions. First it converts the clock pulses that it receives, which are 25ns wide, to pulses that are 15ns wide so that they will not be degraded when passing through the isolation transformers. Secondly, it provides enough driving current and output voltage swing required to differentially drive its clock transmission line. Finally, it distinguishes between clock pulses and messages pulses in the pulse stream presented to it and blocks the message pulses if the message control bit associated with its port is a 1 (note that this bit is a masking bit and not an enable bit).

Figure 3-5 shows a diagram of the differential line driver used in CloDS. The pulses are narrowed to 15ns-wide pulses by the D-flip-flop and the two buffers that reset it 15ns after it is triggered. The required driving current, voltage swing and line impedance matching are provided by the two 74F3040 NAND gates and their terminating resistors. The voltage swing is halved because of the line matching but is doubled again in the the 1:2 pulse transformer. Message pulses are easily detected because they coincide with the masking signal. Message masking is performed by the 2-input NAND gate.

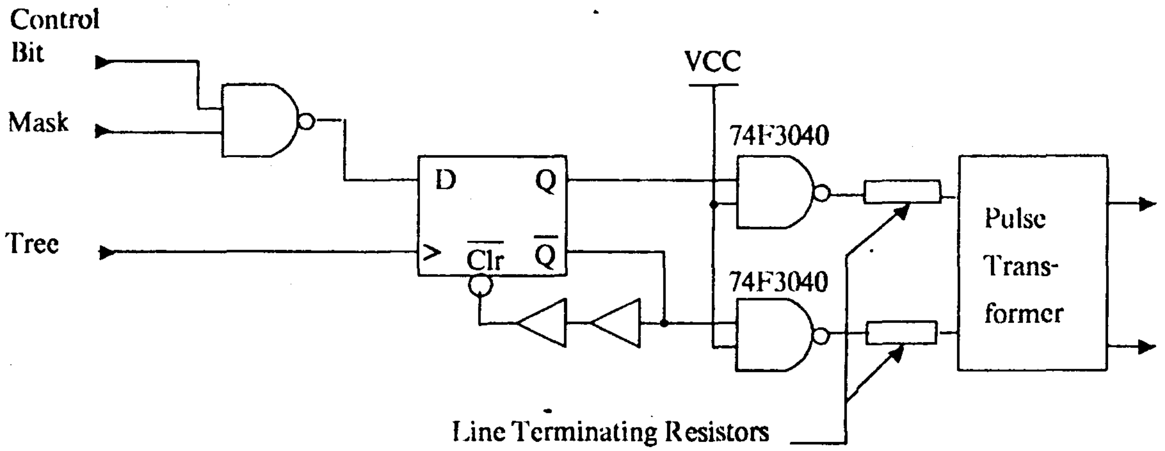


Figure 3-5:A digital line driver.

From the above we see that every clock branch requires four control bits, one message-masking bit, and three bits for its fine programmable delay line. In addition, every cluster requires three control bits for its coarse programmable delay lines. Note that the clock and the masking signal coarse delay lines of a cluster are both programmed with the same delay and, hence, require only three bits of control per cluster. Therefore, every two branches have one 8-bit control register, and every two clusters have one 8-bit control register for a total of 18 registers for the clock branching circuitry.

3.2.5 The Reference Branch

In addition to the 32 clock branches is the reference clock branch. The reference is very similar to the 32 other branches. The reference differs in that both of its delay lines are coarse delay lines with 5ns increments. In addition the reference branch does not have a message masking bit and hence always passes all clock and message pulses.

3.2.6 Nubus Interface

The Nubus interface provide the required signal timing to interface the internal data and address busses of CloDS to those of the Nubus. The interface responds only to single-byte requests. Multiple-byte requests are ignored and will result in a Nubus timeout. The interface resets itself anytime the Nubus _RESET line is asserted regardless of all other conditions. The interface resets CloDS if the Nubus _RESET line is asserted for more than one Nubus cycles. The interface also recognizes idle cycles and ignores them.

On power-up, the interface can only be written to. If the Nubus attempts to read a location at this stage, the interface will treat it as a write. This means that the Nubus will read a random number since the interface never drives the Nubus. The Nubus can read from the card only after writing to a certain location to enable the card. This *kludge* was included because whenever the TI Explorer loaded Lisp after power-up, it read a certain location in the card. Apparently, the Explorer didn't like what it saw there and went into an infinite loop somewhere in its monitor. The TI Explorer worked fine when CloDS ignored these early reads.

The internal bus format of the main board is very similar to that used by Intel 8080 processor chip. It is an 8-bit bidirectional data bus and a separate address bus. This made easy the use of intel processor support chips, used in CloDS communication links, and available static memory chips, used for CloDS storage. All requests to CloDS main board consume 4 Nubus clock cycles.

The Nubus interface decodes only Nubus address lines AD(0:9) and AD(24:31) and, hence, have the following format

FsXXXucc(Hex)

Where s is the number of CloDS Nubus slot, cc are the least significant Byte of the internal CloDS address, and u has the format of 11mm(Binary). The mm here are the two most significant bits of CloDS internal address. Note that the most significant byte in the Nubus address must be an F(Hex) for CloDS to consider the request.

All writes to the CloDS main board are logged in a non-volatile memory (NVM). Data read from the main board comes from one of two sources. If the addressed location is that of a write-only control register, then the data comes from the NVM memory that reflects the contents of the register. If the addressed location is that of a read/write component, such as that of a UART, then the NVM is not enabled and data comes from the addressed component. Therefore, when CloDS is powered down, it retains the contents of all of its control registers. However, information in read/write locations is lost. The only read/write locations that CloDS have are those of the asynchronous communication chip and the card status register. Both are not necessary for basic CloDS operation.

3.2.7 The Probe Circuit Breaker

The probe circuit breaker is physically located on the main board and serves to protect CloDS from any faults that might happen on the main-board-to-probe cables. Power to the probe comes from the external power supply through this circuit breaker. This circuit breaker cuts the power to the probe and disables the communication lines to it in the event of a fault. The breaker is fully implemented in hardware and will continue to provide protection even if the host machine fails. Fault here is the occurrence of one of the following.

1. A short or an open circuit on the main-board-to-probe power lines.
2. A short or an open circuit on the main-board-to-probe communication lines.

In addition to cutting the power on the occurrence of a fault, the breaker circuitry can do limited diagnostics to try and determine the cause of the fault. These diagnostics include detection of the following conditions.

1. Bad breaker relay.
2. Blown fuse of the probe power line.
3. Blown fuse of the probe ground line.

The breaker detects a power fault by constantly monitoring the supplied current (Figure 3-6). This is done by measuring the voltage drop across a power resistor through which the probe gets its power. If this voltage deviates too much from that of a healthy probe, the breaker trips. The upper and lower voltage limits are adjusted when installing the probe for the first time. Note that the inputs to the comparators are conditioned by low-pass filters. These filters prevent the breaker from being overly sensitive; after all, the breaker must not trip because the user was wearing a woolen sweater when he touched the probe. Detection of a fault on the communication lines is done by the communication transmitters and receivers. The breaker trips whenever the communication circuitry informs CloDS control of a fault in its lines. The breaker can be tripped at any time under software control also.

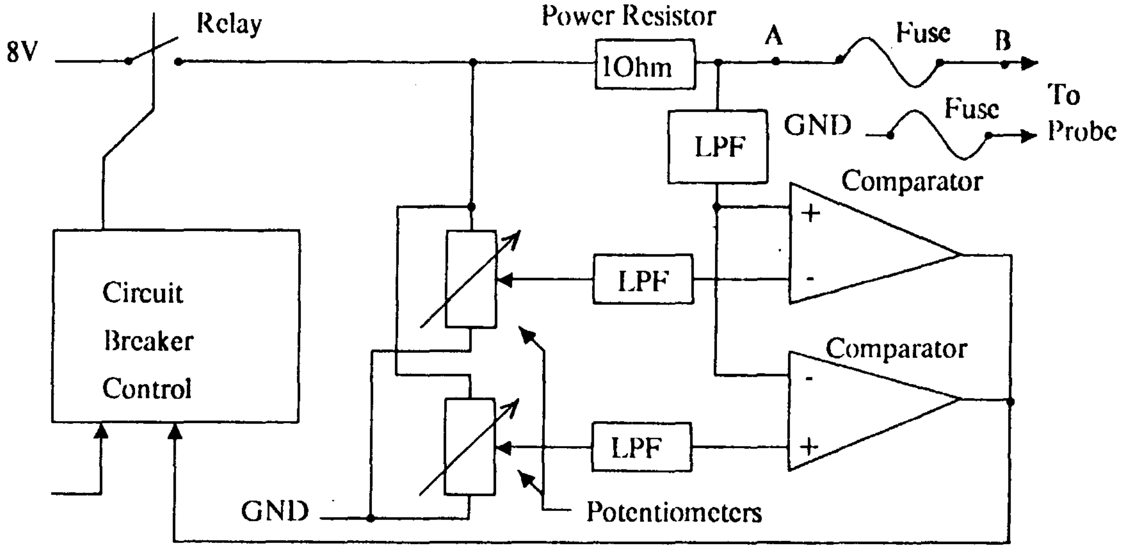


Figure 3-6: Block diagram of the probe circuit breaker.

To detect for blown fuses, the breaker monitors the voltage across each fuse. If there is a voltage, chances are the fuse is blown. Pull-up or pull-down resistors are placed on opposite ends of a fuse in a manner to force the voltages on the fuse ends to voltages opposite to those present during normal operation so that a fuse failure can result in a different voltage on the faulty test point. The same technique is used to detect a bad relay. For example, in Figure 3-6, if point A was high but point B was low when the breaker is switched on, then fuse A is probably blown. If both are low when the breaker is on, then the relay or the external power supply might be bad.

Whenever the probe is turned on by the software, the breaker masks any fault conditions for about half a second. During this time a fault will not trip the breaker.

This delay serves two functions. First, it allows power transients to die out and, hence, prevent tripping the breaker under normal conditions. Second, it allows the software enough time to read in the fuses' status. Note that, once the breaker trips, all the fuses will appear to have been blown. If the software needs more time, it can force the breaker to ignore one or more fault conditions while it performs its own special diagnostics. However, one should remember that while the breaker is being prevented from tripping, something might very well be frying! Therefore, this fault-masking should be avoided whenever possible, and should never be sustained for any extended period of time.

The probe is forced in an off state when the main board is powered up.

3.2.8 Serial Communications Logic

Serial communication is handled by an Intel 8251 UART. The input clock to this chip is derived by dividing the Nubus clock. The clock frequency is 19.2KHz. Operating in a /16 mode, the baud rate is 1200 Baud. This is fast enough for all present communication needs; however Baud rates as high as 19.2K Baud are possible if the need arises.

The differential communication receivers have the added capability of detecting faults. The line receivers are the National Semiconductor DS88LS120 differential receivers. These receivers output a 1 if the voltage at their positive inputs is higher than at their negative inputs. They output a zero under the reverse condition. The receivers have a hysteresis of 200mV under normal conditions and will not change the state of their outputs unless the polarity on their inputs swings in the opposite direction by more than 200mV. The thresholds are at +/-200mV and are symmetrical around the zero voltage crossing.

These receivers can be programmed to have asymmetric thresholds. For example a positive threshold at +700mV and a negative threshold at +200mV. For fault detection, two receivers are connected as in Figure 3-7. If the input voltage is higher than 700mv or more negative than -700mV, the output of one of them would be 1 and the output of the circuit is zero (no fault). If the voltage difference is between +200mV and -200mV, both outputs will be zero and the fault flag will go high. For this reason, each receiver uses three 88LS120 line receivers, two for fault detection with asymmetric thresholds and one for data with symmetric thresholds.

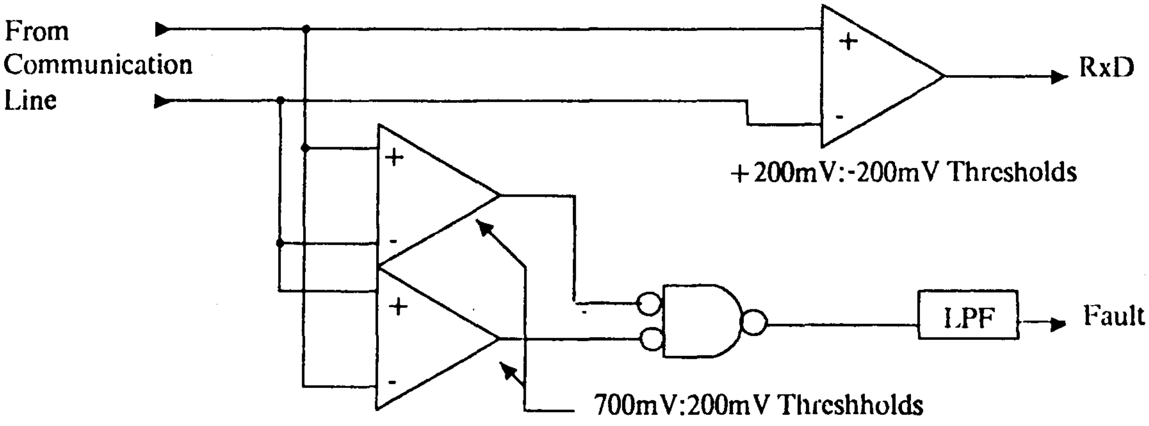


Figure 3-7:Diagram of a communication line receiver.

3.2.9 Debugging Display

The debugging display is a 6-digit seven segments display. The seven segment digits can display any on/off combination of segments and not only numbers. Associated with this display is an 8-bit register. Writing a number to this location displays the message that is programmed in the display EPROM at the address corresponding to this number. A total of 256 6-digit messages are possible with this display. (See chapter 4 for a listing of these messages.) The display is only useful when debugging the board with an extender card as this is the only time that the display can be seen.

3.3 The External Power Supply

The probe requires a 5V, at 0.5A, supply. The problem is that the cable supplying power to the probe is 25ft long. IR drops across the long cable will reduce the voltage at the probe. In addition, the way the breaker detects power faults, by measuring the voltage drop across a power resistor, further reduces the voltage delivered to the probe. For these reasons, the probe has its own voltage regulator and the voltage supplied to it must be higher than 5V, at least 8V for the probe linear regulator to function properly. The Nubus does have a 12V power supply. Unfortunately, the 12V rail cannot supply the required current with the other cards plugged in the machine. Attempts to do so will overload the supply and affect the operation of other cards in the machine. For this reason, CloDS has an external power supply for the probe.

The power supply is basically an 8V, 1.5A, linear power supply. It is designed to fit between the TI Explorer back door and the Nubus back plane. The supply takes its A.C. power from one of the two power outlets at the back of the TI Explorer. The power supply feeds an 8V output to the circuit breaker on the main board through the second Nubus I/O connector (P3).

The external power supply differs from a standard linear supply by the following. The 8V is available at the output of the supply only if there is a 19.2KHz tone on its control input. This tone is always supplied by the main CloDS board through the same connector. The presence of anything but this tone will cause the external power supply to shut down. The tone must be present and stable for a period of time before the supply will switch on and it must remain stable for one second. The supply also generates a warning tone whenever the Explorer is powered up and the tone is not present. The reason for this added complexity is to prevent the supply from driving 8 Volts into a non-CloDS board if the user forgets to disconnect the external power supply before plugging a different card in CloDS's slot.

The external power supply does this using a tone decoder with a low pass filter on the output, and a cutoff relay.

3.4 The Calibration Probe

The function of the probe is to measure the phase difference between a clock at a port that is being calibrated, and the reference clock present at the probe. The probe consists of the following sections.

1. The phase measurement circuitry.

2. The probe control and communication circuitry.

3. The probe power supply.

4. The status display.

3.4.1 The Phase Measurement Circuitry

The calibration probe in CloDS functionally has three inputs. The first input is the reference clock. The reference clock comes to the probe in the same format that a clock branch reaches a port, a stream of narrow pulses. The probe, therefore, contains a clock recovery circuit to recover the square shape of the reference clock. The second input of the probe is the clock from one of the 32 ports. The format of this signal is a square wave and is only passed through an optocoupler for isolation before being used by the probe. The third signal is the signal that comes from the port and tells whether the port is currently receiving any synchronous messages or not. This is a true/false signal and is only passed through an optocoupler for isolation before it is used. Note that the output of the reference recovery circuit is also passed through an optocoupler. The reason here is not isolation, but to match the added propagation delay that the other clock acquires while passing through the optocoupler.

To measure the phase difference between the two clocks, the clock and the reference are fed into two measuring branches with each branch having three stages. The first branch measures the time from the rising edge of the clock to the first following rising edge of the reference and is called the leading indicator. The other measures the time from the rising edge of the reference clock to the first following rising edge of the clock and is called the lagging indicator. Note that one of the two indicators may be measuring the phase difference between two different clock and reference pulses. For example it may be measuring the phase between the rising edge of the *nth* clock on the reference line and the *nth+1* clock on the clock line. This is not critical since the control of the probe is intelligent enough to know which indication is the correct one.

From the above we can see that the two branches are identical except that their inputs have been switched. Let us now consider the leading indicator (Figure 3-8). The job of CloDS is to force the clocks to have a phase difference that is close to zero with respect to the reference clock. This means that if a standard phase detector is used, the width of the error pulses it produces will tend to zero as the clocks are more and more aligned with the reference. Pulses that are very narrow are very hard to measure accurately, especially if one recalls that the probe has to complete its measurement of the phase error based on one clock pulse. For this reason, in the leading indicator, the reference clock is delayed by 60ns before it is fed to the indicator. This means that a zero phase error will result in a 60ns error pulse from the phase detector, a pulse that can be easily and accurately measured.

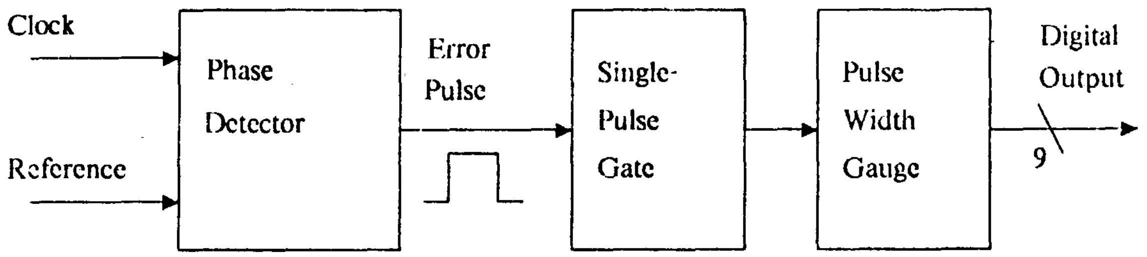


Figure 3-8:Leading indicator stages.

Coming out of the phase detector, the stream of error pulses is fed into the second stage which is the single-pulse gate. The function of this gate is to allow only one error pulse through whenever it is reset by the probe control. This is required because the digital pulse width gauge produces erroneous measurement if more than one error pulse is fed to it while it is performs the measurement.

The final component of the leading indicator is the digital pulse width gauge. Because both the leading and the lagging indicator have a single-pulse gate stage, they both share the same pulse width gauge. Sharing is accomplished by the probe control resetting only one single-pulse gate at a time.

The final stage uses programmable digital delay lines to measure the width of the error pulse that the phase detector has produced. If the pulse width of the error pulse was wider than the programmed delay, then the probe would increment the programmable delay lines and would try the comparison again. Eventually, the delay of the delay lines approximated the width of the error pulse. The probe then sent the digital word that was presented to the delay lines to the main board. A prototype of this converter showed that such a method was accurate enough to measure the width of the error pulse to within +/- 1.5ns.

An error pulse arrives at port A from a single-pulse gate and triggers the first flip-flop (F1) (Figure 3-9). It also goes directly to one of the AND gate inputs. The output of F1 goes trough the delay lines and exits at point B Xns later. The X here is the programmed delay value (by T0-T8). If the delay is greater than the width of the error pulse, then the output of the AND gate never goes high and consequently, F2 is never triggered. F2 is triggered, however, if the delay was less than the width of the error pulse. By varying the delay and monitoring the output of F2 the probe is able to tell how wide the error pulse is. The control clears F2 before it gates an error pulse in from one of the two single-pulse gates.

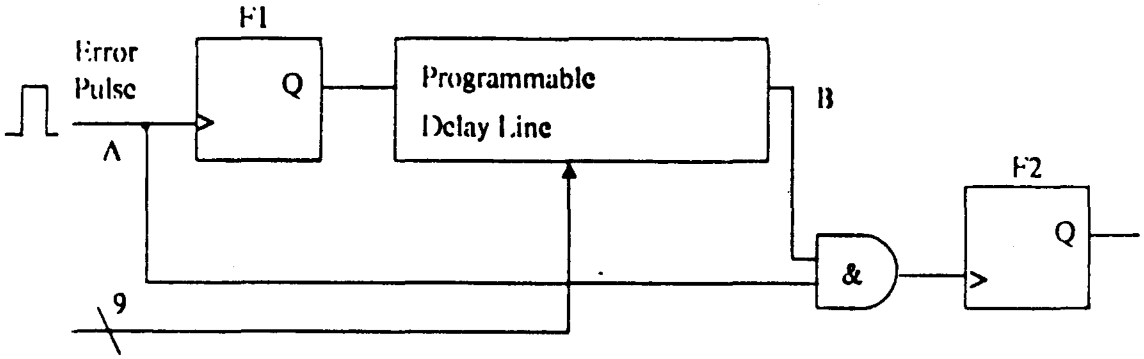


Figure 3-9:A pulse-width gauge of the probe.

CloDS uses two 10ns increment and one 3ns increment delay lines. This corresponds to nine bits delay control (T0-T9). The probe control increments the delays as follows. Initially it puts ones on all the three delays, T0-T8 equal 001001001(Binary). Then it increases the delay by 10ns at a time using the 10ns increment delay lines. If F2 goes low after an increment then the last 10ns increment is removed which will make F2 go high on the next sample. Now the control proceeds to increase the delay by 3ns increments until F2 goes low again. By this time T0-T8 will reflect the pulse width presented to it within 3ns.

To reduce the time of trial and error, the probe changes the delay in a way similar to that used in the successive approximation logic that is common in analog-to-digital converters. On every trial the probe is hence able to half the guess range, thus resulting in a worst case that is of the order of *log(n)* instead of *n* with *n* being the number of allowable T0-T8 combinations.

We now return to the third input. This input pulses every time the port being calibrated receives a message pulse. Coming out of the optocoupler, this input is fed to a D-flip-flop. The flip-flop is reset by the control of the probe whenever it wants to determine if this port is currently receiving any messages. The control then polls the output of the flip-flop. If the flip-flop goes high, then the port has received a message. Remember that this is the method by which CloDS automatically determines which port the probe is connected to. The main board sends synchronous messages to one port and asks the probe if the port it is connected to is currently receiving any messages. If the probe affirmatively responds, the port number is determined.

3.4.2 The Probe Control and Communication Circuitry

The probe control and communication circuitry is simplified by using the Intel 8751H microcontroller. This is a single chip microprocessor that contains the following.

- 4 KByte EPROM.
- 128 Byte RAM.
- 32 I/O pins (bidirectional).
- 2 16-bit counters.
- An asynchronous transmitter and receiver.

In the probe one of the timers is used as a baud rate generator. The 32 I/O pins in the 8751H can each be programmed to be an input or an output. In CloDS the pins are programmed as shown in Figure 3-10.

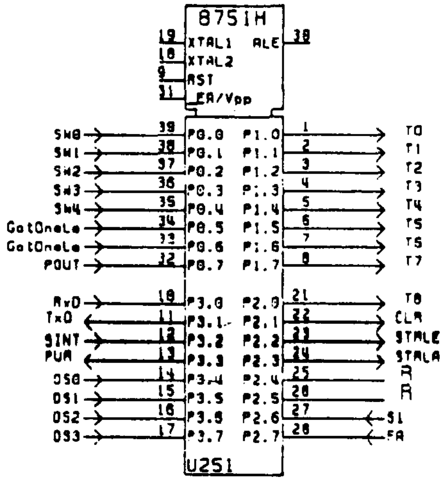


Figure 3-10:I/O of the M8751.

The table in Figure 3-11 list the programmed functions of these pins.

The five dip switches (SW0-SW4) indicate which program, out of a number of programs present in the EPROM, the processor must branch to on power up or after a reset. Additional control is also provided to allow for single-stepped operation of the processor. These are two powerful debugging tool included to ease microcontroller software development. The probe communications drivers and receivers are identical to those used in the main board, including their fault detection circuitry.

T_0:T_8	Output	These outputs program the delay lines.
CLR	Output	This output clears the gauge flip-flop before a sample.
STRLE	Output	This output gates one pulse through the single-pulse lead gate.
STRLA	Output	This output gates one pulse through the single-pulse lag gate.
S1	Input	This input tells if the port the probe is connected to received any messages.
DS_0:DS_3	Output	This is the BCD nibble that feed the 7-segment display of the probe.
PUR	Output	Reset the port message flag (S1).
Txd	Output	Serial data transmit line.
RxD	Input	Serial data receive line.
SINT	Input	Single-step mode interrupt pin.
FA	Input	This input is the communication fault flag.
SW_0:SW_4	Input	These reflect the position of the probe configuration switches.
GotOneLE	Input	This input tells that the an erorr pulse has been gated from the lead indicator into the pulse width gauge.
GotOneLA	Input	This input tells that the an erorr pulse has been gated from the lag indicator into the pulse width gauge.
POUT	Input	Output the pulse width gauge flip-flop F2.
R		Reserved.

Figure 3-11:Pin definitions.

3.4.3 Probe Power Supply

The probe power supply is a simple 5V, 3A, linear power supply. With an active probe load of about 0.6mA, the probe power supply is substantially overrated. This is because the calibration probe, like CloDS, is an experimental device and, hence, must support any possible future modifications or additions. The supply overrating proved valuable when the probe itself was calibrated, since additional circuitry was connected to it to simulate working conditions.

The probe power supply has a shutdown switch. The switch kills the power to the probe not by opening the circuit but rather by placing a heavy load in parallel to the probe load on the probe power lines. This trips the probe circuit breaker at the main board and hence disconnects both the power and communication lines. This indirect shutdown procedure indicates that the breaker still works. In the event that this does not work, then disconnecting the main-board-to-probe cable will also trip the breaker, but now it will trip because of a fault on the communication lines.

3.5 The Clock Recovery Ports

The clock recovery port recovers the square shape of the clock and assembles the incoming stream of message pulses into a byte and then signals the circuit switch that a message has arrived. The port also provides output of both the recovered clock and message pulses to a connector that the calibration probe can connect to.

The clock arrives at the port in the format of narrow pulses. These pulses pass through a 1:2 pulse transformer for both isolation and voltage doubling (Figure 3-12). Connected in parallel with the inputs of the transformer is a line termination resistor to eliminate signal reflection. The output of the transformer is connected to the clock input of a D-flip-flop. The -Q output of the flip-flop is connected to its -CLR input through a 60ns delay line. This means that the output of the flip-flop goes high for about 70ns, accounting for propagation delay of flip-flop for every incoming clock. Note that if a pulse follows another by less than 140ns it will pass undetected by this flip-flop. Such a pulse, however, will trigger the other flip-flop, the message flip-flop. From the above, the maximum clock frequency possible is 7MHz, otherwise, the pulses will be interpreted as messages. A way of increasing this limit to about 13MHz without sacrificing the message operation is included in section 5.2.

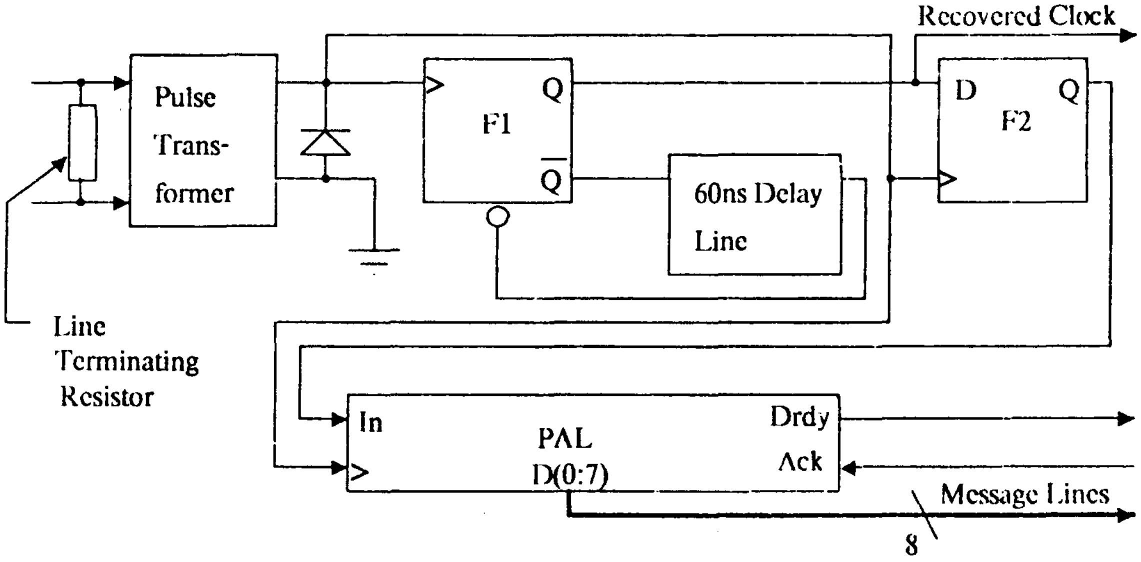


Figure 3-12:Clock and message recovery port.

The message pulses are assembled in a custom PAL. This PAL has two inputs, a clock input which is the output of the clock flip-flop (F1), and a message input which is the output of the message flip-flop (F2). Initially, the PAL is in a HUNT state. If the clock goes high, while the message flip-flop is high then the PAL goes into a shift state. In this state the PAL shifts in the output of the message flip-flop on the rising edge of the clock. This PAL shifts in eight bits and then HOLDs. In this state the PAL outputs the assembled byte and pulls the DRDY high to indicate the reception of an 8-bit message. The PAL returns to the HUNT state only after the circuit switch acknowledges by pulling the DACK high. Note that the first message bit is a start bit and will not be part of the assembled byte.

Each circuit switch contains a port circuit. The calibration probe contains a simplified port circuit that recovers the clocks but ignores any messages.

Chapter Four

Software Support

4.1 Introduction

To operate correctly CloDS needs software support. The software is in two parts. The first is the software that runs on the host machine and controls the main board of CloDS. This software is written in Lisp on the TI Explorer machine. The second is the software that runs on the calibration probe microcontroller and controls the operation of the probe. This software is written in Intel MCS-51 assembly language and runs on the 8751H chip.

This chapter will list all the addresses of the control registers in the main board of CloDS and their effect on CloDS's operation. Description of the actual software implementation will follow in a later document.

4.2 Main Board Addresses

The main board decodes only Nubus address bits AD(0:9) and AD(24:31). The main board treats the remaining Nubus address lines as "don't cares". Therefore, writing or reading addresses differing only in address bits AD(10:23) will result in the same action. For psimplicity, I shall list the addresses of the control registers assuming all the don't cares are zeros. The operator should remember that the given addresses are not the only addresses that the board will respond to. Note that CloDS only responds to byte #0 transfers of the Nubus. This means that legal addresses are those that are divisible by 4. For example addresses 0,4,8,C,... are legal while 1,2,3,5,6,7,9,A,B,... are not.

All addresses and data below are in hexadecimal format and X below is the number of the slot that CloDS is installed in. CloDS is relocatable in a Nubus cage.

4.2.1 Write Only Addresses

All of the following addresses are control register addresses. Writing to them will write to the control register and will send a copy to the non-volatile memory (NVM). Reading from them will access the data in the NVM at that address.

Addresses FX000000 through FX00003C

These addresses control the value of the fine delay-line of the branches and contain the message-masking bits for these branches also. Each address controls two branches. For example, the four least significant bits of address FX000000 control branch #0 while the most significant four bits control branch #1. The format of their data is as follows.

$$[M_2D_2D_2D_2M_1D_1D_1D_1]$$

- M_n

  These are message-masking bits. Writing a 1 to one of them will prevent synchronous messages from being sent on that clock branch to its respective port.

- $D_nD_nD_n$

  These bits control the value of the fine delay-line of their branch.

n is either 1 to 2.

FX000040

This address has the following format.

$$[F\ D_2D_2D_2\ T\ D_1D_1D_1]$$

- F

  This is the fault enable bit. If this bit is set to 0, then a short or an open circuit on the probe communication lines **will not** trip the circuit breaker.

- $D_1D_1D_1$

  This is the value of the coarse delay-line of cluster #0.

- T

  This is the overload masking bit. If this bit is set to 0, then an overload or an open circuit on the probe power lines **will not** trip the circuit breaker.

- $D_2D_2D_2$

  This is the value of the coarse delay-line of cluster #1.

Address FX000044

This address has the following format.

$$[R\ D_2D_2D_2\ R\ D_1D_1D_1]$$

- R

  These bits are reserved for future use.

- $D_1D_1D_1$

  This is the value of the coarse delay-line of cluster #2.

- $D_2D_2D_2$

  This is the value of the coarse delay-line of cluster #3.

Address FX000048

This address has the following format.

$$[C\ D_2D_2D_2\ P\ D_1D_1D_1]$$

- C

  Setting this to 0 disables all the clock branches. This bit should be set to 1 for normal operation.

- $D_1D_1D_1$

  This is the value of the first coarse delay-line of the reference branch.

- P

  Setting this bit to 1 switches on the probe breaker. The breaker may shut down, however, if a fault occurs even if this bit is 1. Setting this bit to 0 always shuts the breaker down. This is an edge triggered input. For the breaker to switch on, the bit has to be cleared before it is set to 1.

- $D_2D_2D_2$

  This is the value of the second coarse delay-line of the reference branch.

Addresses FX000050 and FX000054

These addresses are the locations of the main oscillator control registers. Location FX000050 contains the eight least significant bits of the digital divider part of the oscillator (O_0:O_7). Location FX000054 has the following format.

$$[B\ Y\ V\ S_1S_1S_1\ O_8O_9]$$

- B

  This is the divide-by-two bit. If this bit is 1, the output frequency of the selected VCO in the main oscillator will be divided by two.

- Y

  Setting this bit to 1 puts CloDS in single-step mode. This bit should be set to 0 for normal operation.

- V

  This is the VCO selection bit. If this bit is 0, the output of VCO #1 will be selected in the main oscillator. If this bit is 1, VCO #2 is selected.

- $S_1S_1S_1$

  These bits select the range of the main oscillator. Their value corresponds to the power of ten that the output of the selected VCO will be divided by. For example, if the output of the VCO is 6MHz and $S_1S_1S_1$ is 4, the main oscillator frequency will be 600Hz.

- O_8O_9

  These are the most significant bits of the digital divider in the main oscillator.

To set the frequency, one has to first choose the correct VCO. VCO #1 can track from 2MHz to 4.2MHZ, while VCO #2 can track from 5MHz to 12MHz. The VCO frequency is determined by the following equation.

$$f=\frac{10^9}{8+O}$$

The oscillator's output frequency, and, hence, CloDS's frequency, is determined by

$$f=\frac{10^9 10^{-SSS_2-B}}{8+O}$$

Where O is equal to the value of O_0:O_9.

Address FX000058

Whenever CloDS is in single-step mode. Writing or reading this location outputs one clock pulse from CloDS to the circuit switches.

Address FX00005C

This is the address of the synchronous messages output register. Writing a byte to this location will send this byte to the enabled ports on the clock lines.

Address FX000060

This is the debugging display location. Writing a number to this location will display a message on the debugging display from the address that corresponds to this number in the EPROM.

Address FX000078

Writing or reading to this address enables the card. The card can always be written to but must be enabled to be successfully read. The output frequency of a disabled card is 2.5MHz regardless of the contents of locations FX000050 and FX000054.

address FX00007C

Writing or reading this address disables the card.

4.2.2 Read Write Addresses

The contents of these locations are not backed up in the NVM.

Address FX000140

This is the probe status register and is used to determine the condition of the probe power supply and communication lines. It has the following format:

$$[BR\ R\ R\ GF\ F\ CM\ P_1P_0]$$

- BR

  If this bit is 1, the breaker is on.

- R

  These bits are reserved.

- GF

  If this bit is 1, the breaker last tripped off because of a blown ground line fuse.

- F

  If this bit is 1, then the breaker last failed because of a fault on the communication lines.

- CM

If this bit is 0, then the breaker last tripped because of a fault in the probe power lines or an overload on the breaker.

- P_1P_0

These monitor the 8V probe power rail. They indicate the following conditions:

P_1	P_2	Condition
0	0	Bad breaker relay or external power supply.
0	1	Blown 8V fuse.
1	1	Normal operation.
1	0	Physically impossible (something is seriously wrong!).

Addresses FX000100 and FX000104

Address FX000100 is the command and status register of the 8251 UART of the main board, while address FX000104 is the data register of the UART. (Reference Intel Microprocessors Databook (1984) for information on configuring this chip.)

Addresses FX000060-FX000074

Addresses FX000080-FX0000CC

Addresses FX000200-FX0002CC

All the above addresses are available for use as a NVM scratch pad.

Chapter Five

Future Improvements

5.1 Introduction

The implemented CloDS has a number of performance limits. While some of these limits are very rigid, others are extendable. This chapter will illustrate some of these limits and what can be done in the future if they become the bottleneck to the performance of the MEF.

5.2 Top Operating Frequency

Currently, CloDS has a maximum operating frequency of 7.0MHz. The clock recovery circuit on every port is where this limit is. Figure 5-1 shows the currently used circuit of a port. The key here is the delay line, Because of the 60ns delay line the clock flip-flop will ignore pulses that are less than 120ns away from the previous pulse. If we make the delay shorter, we can detect pulses that are closer than 120ns. The problem here is that installing a shorter delay will affect the operation of the message decoder. It will also affect the calibration procedure because the message pulses will not be detected. Furthermore, shorter delay means shorter pulse width of the recovered clock.

To increase the operating frequency limit and preserve the integrity of the message encoder and maintain the same clock width, the delay is replaced by the circuit shown in Figure 5-2. This has the effect of increasing the limit to 14MHz while preserving the previous functionality. The PAL of the message decoder, and encoder on the main board, must be replaced by faster versions since they can only function up to 10MHz.

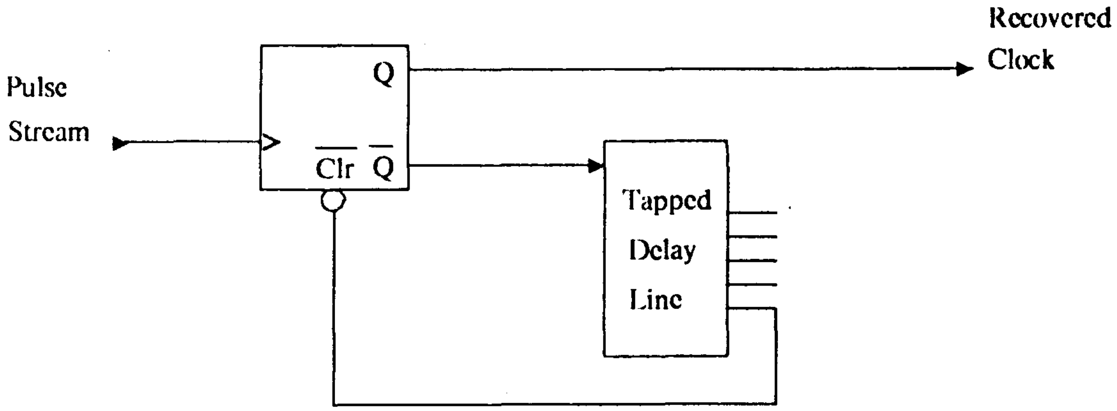


Figure 5-1: Circuit of the current port.

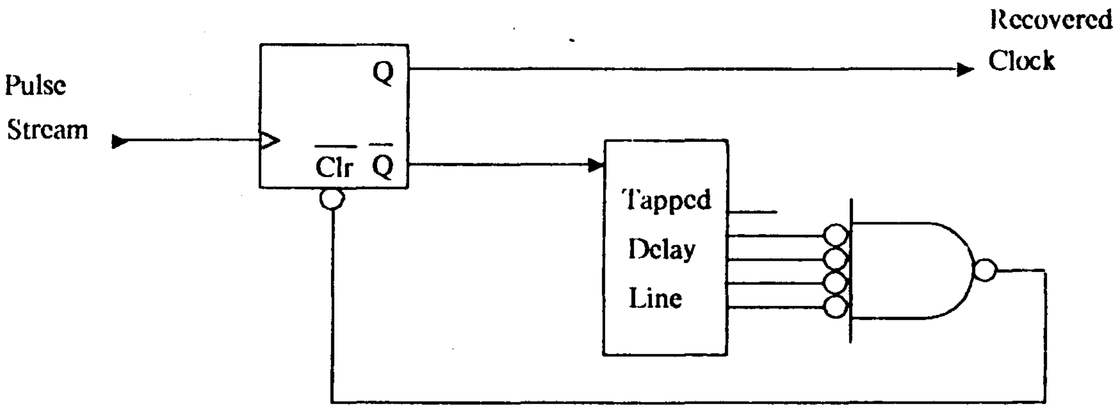


Figure 5-2: Diagram of the suggested alteration.

5.3 Maximum Phase Error

Currently, CloDS can compensate for a maximum phase difference of 21ns between branches on the same cluster and 56ns between branches on different clusters. The 56ns limit between different clusters can be easily extended to 91ns by replacing the 5ns-per-increment coarse delay-lines with a 10ns-per-increment delay lines on the main board. Increasing the compensating limit between branches of the same cluster is not so simple, however. Changing the fine delay-lines can destroy the timing of the message-masking signal and impede the operation of the message encoder.

5.4 Number of ports

CloDS currently supports 32 clock ports. However, it can be expanded to 64, 128 or 256 ports fairly easily. This is done by wiring another CloDS board. This board need only contain the clock branching clusters. All other functional units will be shared. This board will be connected to the original board through two wires. These wires carry the two signals that come out of the message encoder on the main board. The second board must contain its own NVM to back up the contents of the cluster control registers. The two connections are labeled MASK and TREE in the schematics in appendix A.

Chapter Six

Prototypes Construction and Testing

6.1 Introduction

A number of ideas pertaining to how specific sections of CloDS should be built were tried during the development of CloDS. Consequently, a number of prototypes were designed and constructed to examine the viability of some of these ideas. Some of these prototypes proved the concreteness of the ideas behind them, while others proved the opposite. This chapter will describe these prototypes in detail and the results of the tests that were performed on them. The last section of this chapter will describe ideas that were not used in the final implementation of CloDS because they became more complicated than necessary after the specifications of CloDS were changed. Please note that such ideas did not fail; they just became unnecessary. However, description of these ideas where included in this document for completeness.

6.2 Analog Line Drivers

The following is a description of the analog line driver prototype. The idea of using this line driver was eventually abandoned in favor of the digital line drivers. However, the analog prototype gave some valuable insight into the feasibility of encoding the clocks as a series of narrow pulses. This section describes the theory behind this prototype, its construction, and the results of the experiments that were performed on it to verify the viability of the clock encoding scheme.

6.2.1 The Differential Line Drivers.

The differential drivers must provide enough currents to drive the long clock lines. Because the differential line is properly terminated on both ends, the differential drivers should have a voltage swing across the twisted pair terminals equal to double the voltage needed to trigger the receiving gate, plus double the voltage dropped across the transmission line. Because the minimum voltage needed to trigger a FAST gate is 2.0V, and because the lines could be as long as 40ft, a conservative minimum limit of 8V p-p was placed on the differential driver swing. To convert the small voltage swing of a FAST buffer to the desired differential swing, the output of the each buffer was connected to two operational amplifiers.

As shown in Figure 6-1, one amplifier was connected as an inverting amplifier while the other was connected as a non-inverting amplifier. Because the typical output of a 74F244 buffer, which drives these operational amplifiers, swings from 0 to 3.4V, the gain of both amplifiers was chosen to be 4V to guarantee the saturation of the amplifiers when the output of the 74F244 is high.

After choosing the gain, the values of the resistors R1-R4 were selected to achieve the desired gain. Because the used amplifier (LH0032) is a high bandwidth op amp, it requires external frequency compensation. This is provided by the external capacitors Cc and Ca. Values of these capacitors were chosen so as to yield the highest slew rate possible while maintaining low overshoot. From the plot shown in Figure 6-2, Cc was chosen to be about 1pF and Ca was chosen to be about 0.8nF.[2] Because of the fact that the circuit board can have a capacitance much higher than 1pF, it was decided that no compensation capacitor be added, and to make sure that the circuit board have a capacitance of less that 2pF between the pins intended for Cc. The slew rate of these amplifiers, can now be determined from the table shown

[2] Table from the National Semiconductor Hybrid DataBook.

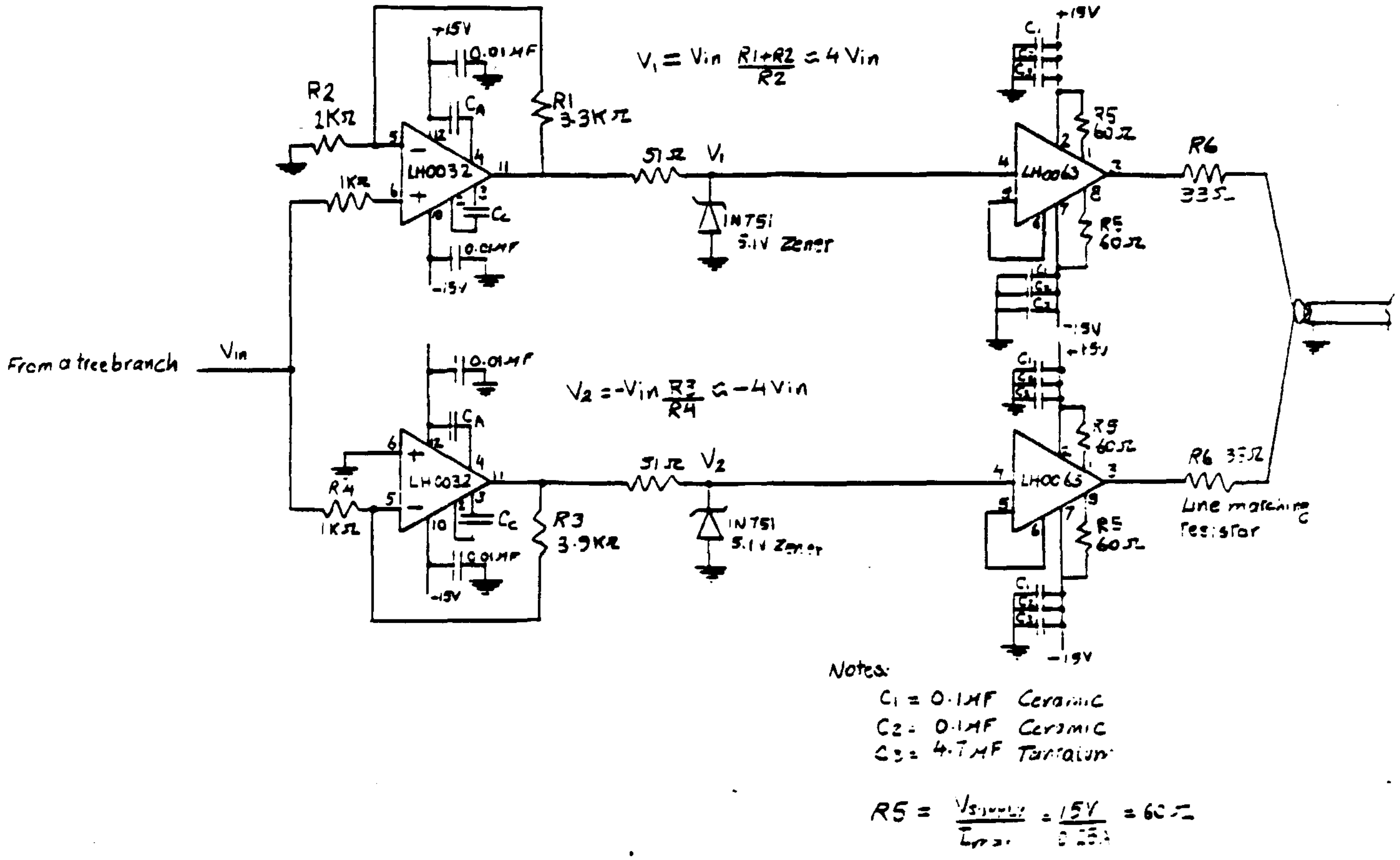


Figure 6-1: Diagram of a differential driver.

in Figure 6-3 and for a Cc of about 2pF the slew rate is approximately 900V/us.[3]

Going through the amplifiers the TTL swing of a FAST buffer is converted into a voltage swing of 0 to 10V on the non-inverting end and a 0 to -10V on the inverting end. This voltage swing is reduced by the zener diode to yield a swing of 0 to 5.1V on one line driver, and a 0 to -5.1V on the other. The two line drivers are the National Semiconductor Damn Fast Voltage Followers (LH0063) with a slew rate of

[3] table from the National Semiconductor Hybrid DataBook.

about 6000V/us and a propagation delay of about 2ns. These particular voltage followers were chosen because of their ability to drive large capacitive loads, i.e., transmission lines, without any degradation in their slew rate or their propagation delay. Because the slew rate of these voltage followers is considerably faster than that of the amplifiers driving them, the signal rise time on the cable can be considered as being a function of the rise time of the op amps and not the considerably faster voltage follower.

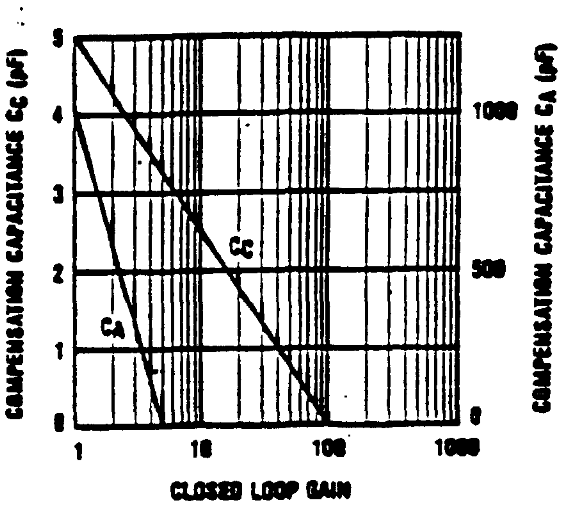


Figure 6-2: Recommended value of compensation capacitor

capacitor vs. closed-loop gain for optimum slew rate.

Although the voltage followers have a driving capability of about 250mA and therefore, cannot be overloaded, current limiting resistors were added to prevent a short-circuited clock line from damaging the drivers. The value of this resistor was calculated to be 60ohms. In addition to current limiting, the power supplies for the LH0063 should be bypassed by capacitors to prevent oscillation. For this particular device the necessary bypassing consisted of two 0.1uF ceramic capacitors in parallel with a 4.7uF solid tantalum capacitor for each supply lead, as shown, to guarantee

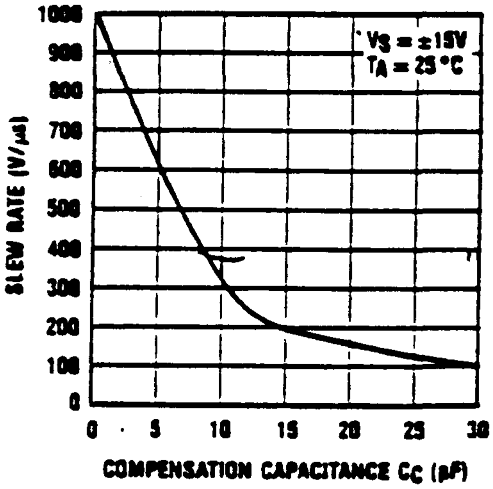
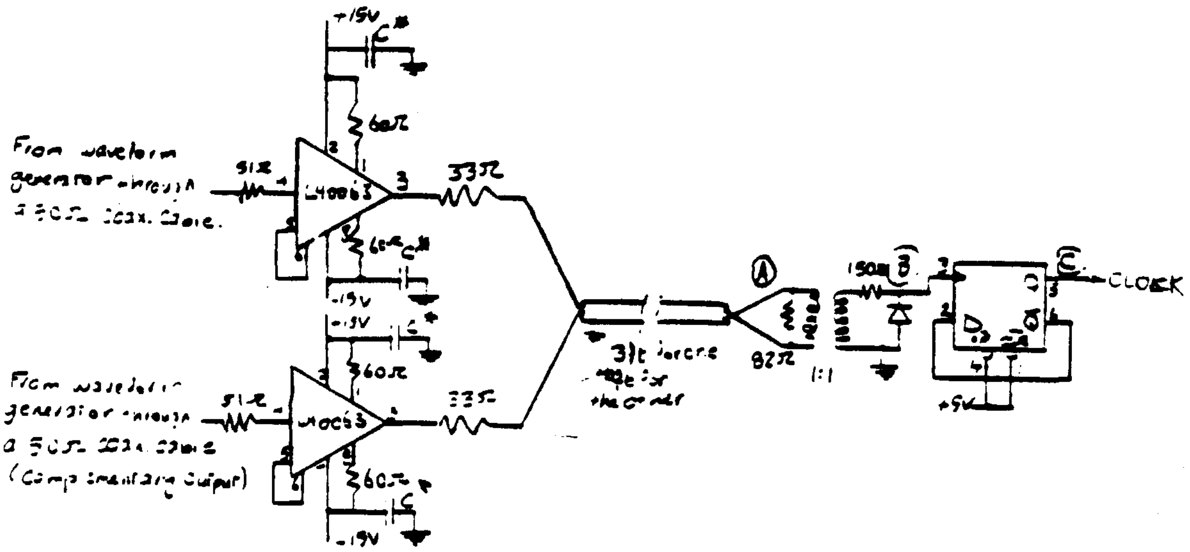


Figure 6-3:LH0032 slew rate vs. frequency compensation capacitance.

oscillation free operation. Coming out of the pair of drivers, is a 0 TO 5V swing on one driver and a 0 to -5V on the other. This combined voltage differential of 10V is immediately divided by two because the driven line is properly terminated on both ends yielding a voltage swing of about 5V across the twisted pair leads at the driver end of the cable. This resulted in a voltage swing of 3.5 volts at a port after going through 40ft of cable and through the isolation transformer. This was enough to trigger a FAST gate with about 1.0 volt of safety margin at a clock frequency of 4MHz.

6.2.2 Circuit Construction and Testing

To test the proposed driver, the circuit shown in Figure 6-4 was constructed. Because of the frequency of operation, and because of the uncommon pin-out of the linear voltage followers, all available prototyping methods were inadequate and two custom shielded-printed-circuits had to be designed and manufactured (Figure 6-5). Testing was performed to verify the validity and stability of the clock encoding



* C is made of two 0.01μF ceramic capacitors in parallel, in parallel with a 4.7μF solid tantalum capacitor.

Figure 6-4: Diagram of the tested circuit.

scheme, and to study the behavior of the isolation transformers under different driving conditions.

6.2.3 Testing for Maximum Frequency.

To test for maximum frequency, the circuit shown in Figure 6-4 was driven differentially by a waveform generator. The frequency was increased until the clock could not be recovered because of the DC-component blocking of the isolation transformer. This is shown in the photos in Figure 6-6 taken of the waveform at points (A) and (B) at different frequencies. The maximum operational frequency is about 10MHz, and, therefore, the maximum allowable clock frequency is 5MHz, to

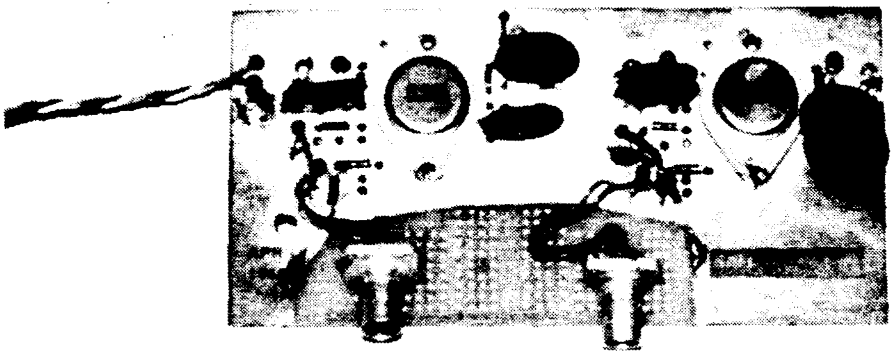


Figure 6-5:Photo of the constructed printed circuit.

allow enough space between two clock pulses for the synchronizing pulses to occupy. Remember that both the rising and falling edges of the clock were encoded as pulses here and, hence, every clock period took two pulses.

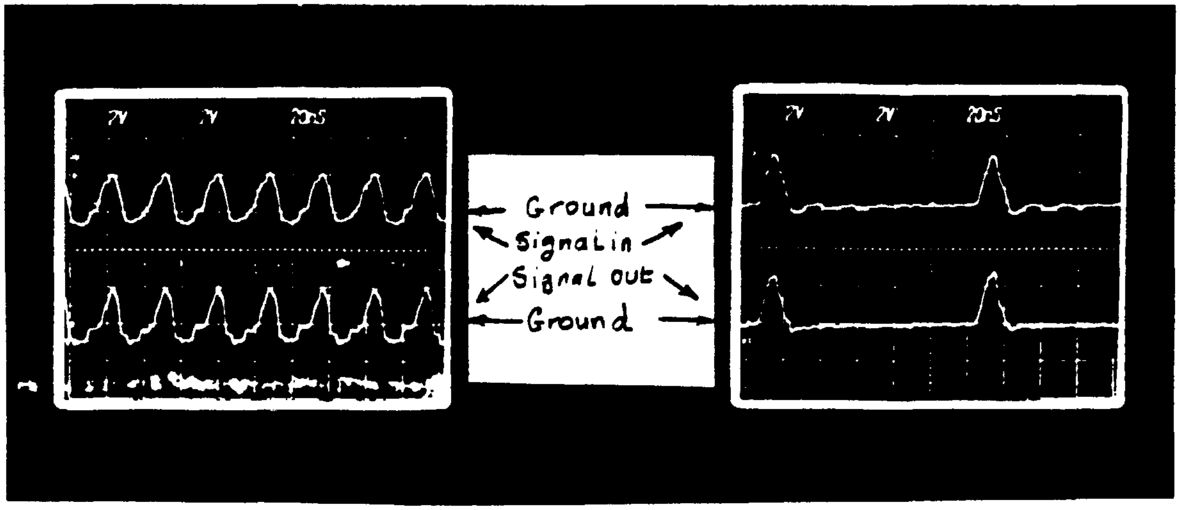


Figure 6-6:Photos of the pulses through the isolation transformer.

6.2.3.1 Testing for Clock Skew

To test for clock skew, two circuits of the one shown in Figure 6-4 were constructed. The two circuits where identical except that the connecting cable was 40ft. long in one and 3ft. in the other. The recovered clocks at the two ports where aligned using the programmable delay lines. To fully simulate the conditions at which these circuits work, the two receiving ports were supplied by two isolated and independent power supplies. Outputs of the two ports were connected to an HP5316a Universal Counter. The average time between the the edges of the clocks was measured to be about 0.7ns and then the circuit was left running for four days. Occasional reading of the average time between the two edges showed that it remained at 0.7 for this period. This does not guarantee that the time average did not vary between the readings, but in the absence of computerized data acquisition devices, this test showed that if the average time varied, it did not vary by much. Deviation of more than 10ns in one direction and back to 0.7ns in seven hours was felt unlikely. The photo in Figure 6-7 illustrates the fine tolerance to which the tested circuit was calibrated at 4MHz.

6.2.3.2 Line Termination Testing

The photo in Figure 6-8 shows the effect of not properly terminating the line; only the driving ends were terminated.

Terminating the line with an 82ohm resistor gave the best results in preventing reflections, as shown in Figure 6-9. In Figure 6-8, for this specific line, driving the clock with pulses about 130ns apart superposes the reflected and the transmitted waves resulting in high amplitude pulses at the isolation transformer. This could damage the FAST gate after the transformer. Properly terminating the line on both ends eliminates this problem, as shown in the photo in Figure 6-9.

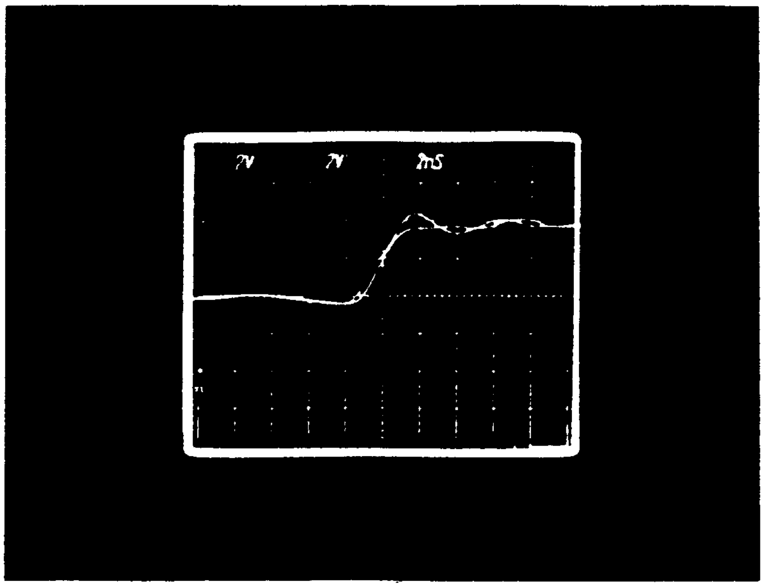


Figure 6-7: The edges of two recovered clocks.

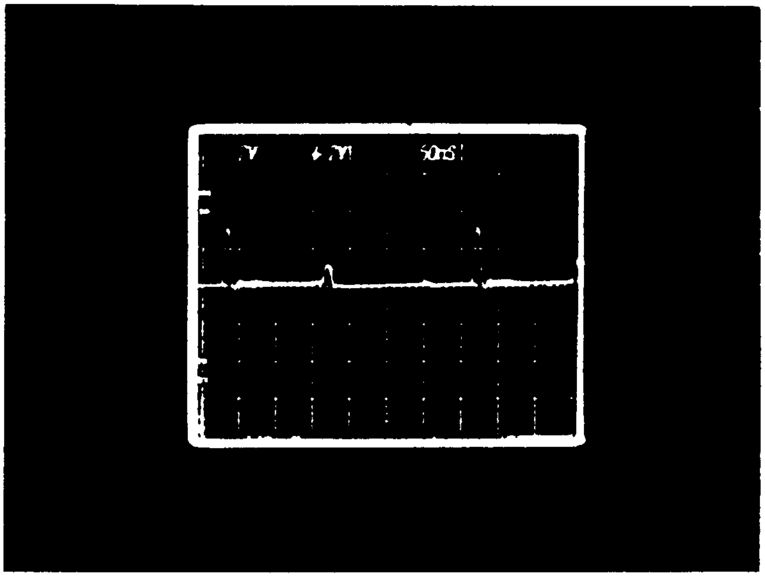


Figure 6-8: Driving end voltage of an improperly terminated line.

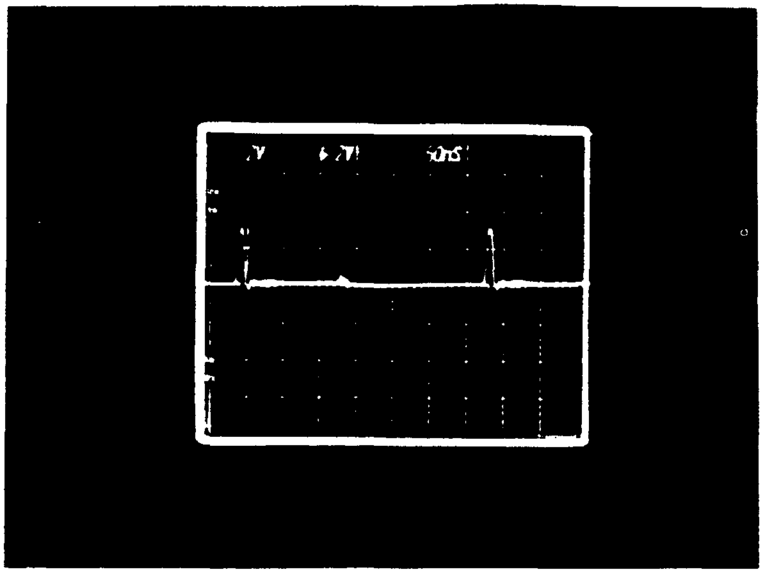


Figure 6-9:Driving end voltage of a properly terminated line.

6.2.3.3 Pulse Count Testing

To make sure that the recovery circuits were not occasionally missing clock pulses, the circuit shown in Figure 6-10 was constructed.

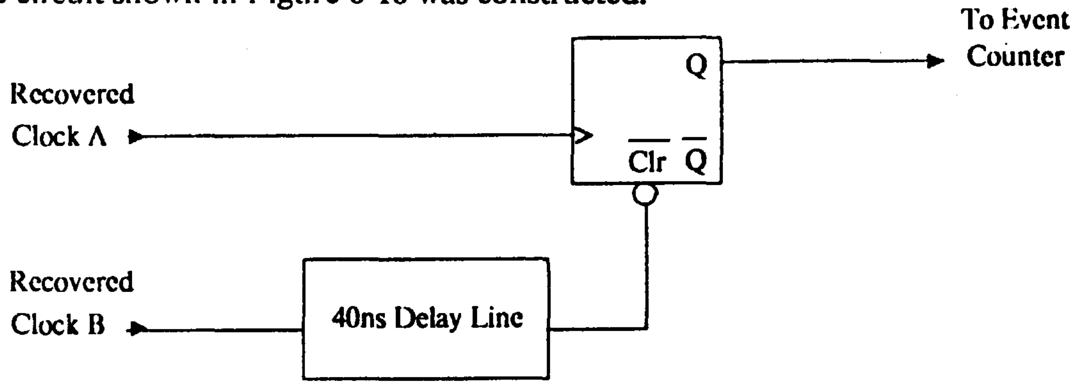


Figure 6-10:Diagram of the circuit used to detect missed clocks.

The output of this circuit was connected to an event counter. The delay used on one of the inputs to the circuit shown in Figure 6-10 was included to prevent metastability problems that might arise from the two clock edges being very close to one another. In addition, the flip-flop was reset before the initiation of the count. In the event that any of the two clock recovery circuits missed a single clock pulse, the output of the flip-flop goes high and the counter registers the event. If the counter counted N events, then either 2N or 2N+1 errors would have occurred, except for N=0 meaning no errors because the flip-flop was initially reset and the counter is triggered on the rising edge. After running for three days, the number of errors registered was zero, or, no clocks were missed.

6.3 The Digital Line Driver

This section describes the prototype built to test the proposed differential digital line driver. Results of the tests done on this prototype showed it to operate according to the specifications of CloDS. Figure 6-11 shows the printed circuit board that was built for this prototype. The double-sided shielded printed circuit board was used instead of other faster prototyping methods to keep the noise transients at a low level. This is because the prototype was used to test the performance of the digital line driver, and board noise might shadow the noise of the driver under test.

Figure 6-12 shows the schematic of the constructed prototype. A square wave signal is turned into a pulse by the first D-flip-flop. This flip-flop resets itself after about 10ns and outputs a 15ns-wide pulse every time it is triggered by an incoming clock. The true output of the flip-flop feeds one NAND gate while the false output feeds the other NAND gate. The NAND gates are 74F3040 gates. They have a sink capability of 160mA. Because the gates receive opposite polarity signals, their outputs are always out of phase. The gates connect to the driver transformer through two resistors. These resistors serve to match the impedance of the line, as well as to limit the current that flows from one gate to the other through the transformer because of the opposite polarity of the gates. The transformers on both ends of the cable are one-to-two step-up transformers and serve to cancel the effect of the

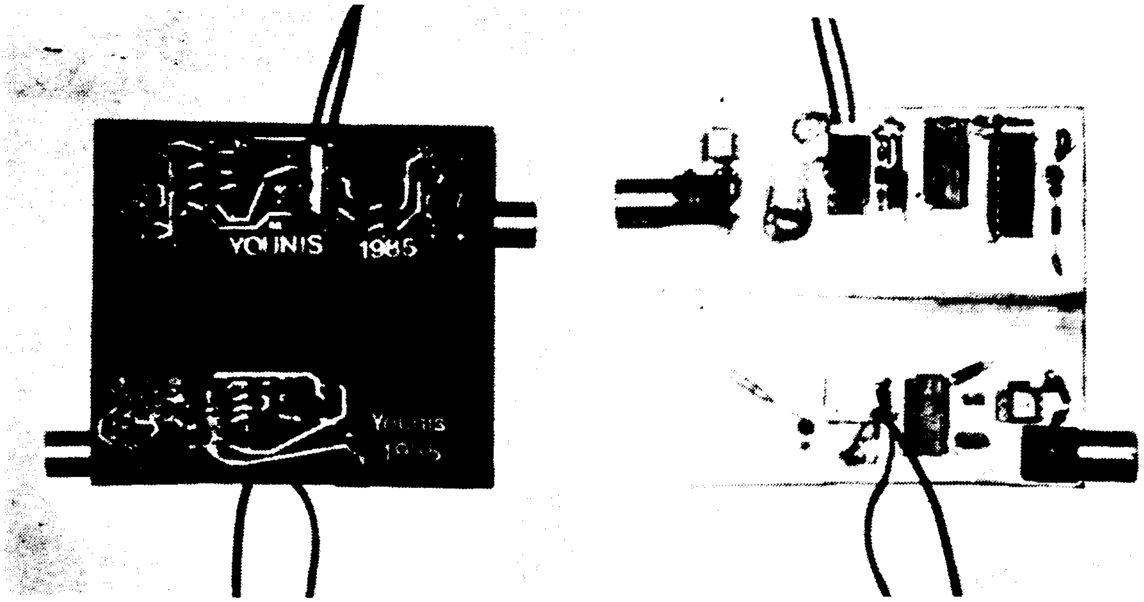


Figure 6-11:Photo of the digital line driver prototype.

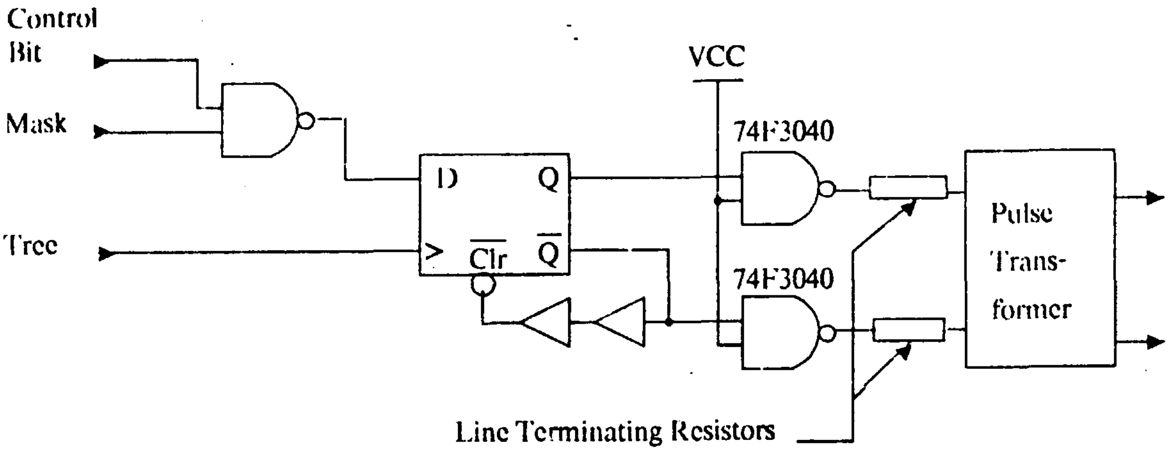


Figure 6-12:Schematic of the digital line driver.

matching resistors on the driving voltage, they half the voltage.

A line matching resistor is connected to the line in parallel with the transformer to eliminate reflections on the receiving end of the cable. The other end of the receiving transformer is connected to the input of the receiver flip-flop. A diode protects the receiver flip-flop from negative voltage swings of the line. This flip-flop resets itself after 50ns through the delay line connected to its inverted output. Effectively, this flip-flop converts the 15ns-wide pulse back to a square wave but with a fixed 50ns pulse width.

Using this prototype, clocks were sent and successfully recovered after 93ft of twisted-pair cable. The prototype continued to operate up to a frequency of 9.1MHz. At higher frequencies the frequency of the recovered clock was half that the input frequency. This is because the delay line continues to clear the flip-flop 50ns after it resets it. Hence, any clock pulse received within 100ns of the previous one is ignored. This limit is not a major one, and the prototype continued to operate at frequencies as high as 20MHz, and at a cable length of 93ft, with a lower value delay line. This proved that this driver is an adequate line driver for CloDS.

6.4 The FSK Communication Modem

In CloDS, the calibration probe communicates with the main board through a bidirectional, high reliability link. To achieve the high reliability, CloDS had to use links that detect faults, such as shorts or loose connections, on their communication lines. A frequency-shift-keying (FSK) modem achieves this fault detection through loss-of-carrier detection. While other less complex methods are available, FSK was pursued because of its potential for compatibility with standard communication equipment and its ability toconnect CloDS to devices other than the probe. For this reason an FSK modem, shown in Figure 6-13, was constructed.

The schematic of this modem is shown in Figure 6-14. The modem consists mainly of the EXAR XR-2066 and the XR-2211 modulator/demodulator chip pair. The

Figure 6-13:Photo of the constructed modem.

values of the timing components were chosen according to the data sheets to yield a baud rate of 1200 Baud. While this modem worked well at 300 Baud, it failed to operate at the intended 1200 Baud. At this higher rate it exhibited a great deal of jitter and needed additional signal filtering. The would-be complexity of a working modem was projected to be more than what CloDS could afford, especially on the probe side where board real estate is scarce. This method was eventually abandoned and the potential compatibility was sacrificed.

6.5 Analog Pulse Width Gauge

The calibration probe of CloDS generates error pulses that reflect the phase difference between the reference clock and the clock under calibration. The width of this pulse is equal to the phase error it conveys plus a constant that the probe adds to protect against metastable conditions (see section 2.x.x). The probe has to convert the width of the error pulse to a digital number before it can transmit it to the main CloDS board. This is the job of the pulse width gauge in the probe.

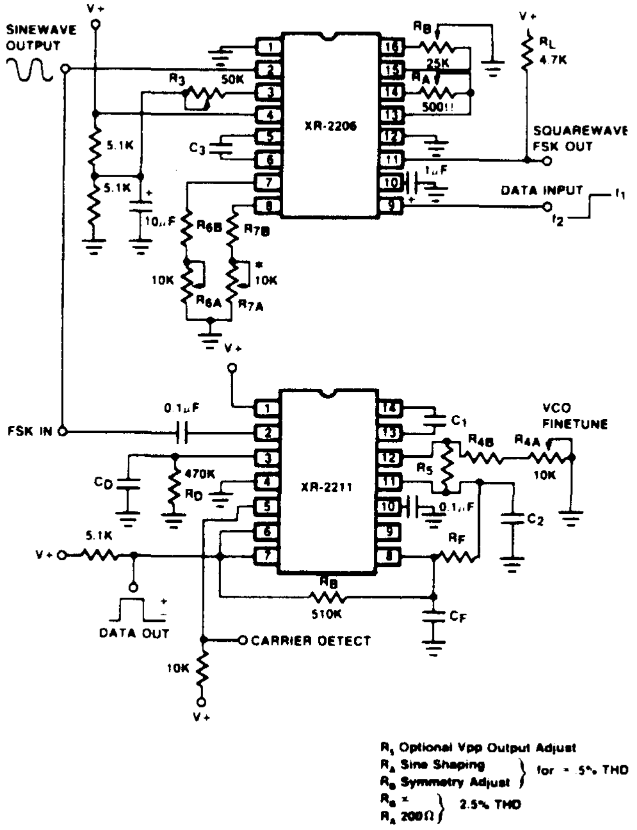


Figure 6-14:Schematic diagram of the FSK modem.

The analog pulse width gauge shown in Figure 6-15 was the first solution tried to the needed pulse-width-gauge. In this circuit the gauge converts the width of the error pulse to a voltage by integrating the error pulse in a charge capacitor. Immediately after the integration, the resulting voltage is held in a sample-and-hold amplifier to preserve its value until it is sampled by the analog-to-digital converter (ADC). The main advantage of using the analog gauge is that its resolution can be made extremely fine without much trouble by using a more accurate ADC.

The proposed circuit used a LM398 sample-and-hold amplifier. Initially, the control logic of the probe would discharge the C1 capacitor by pulling point B low. The control logic brings point B high and then gates an error pulse to point A (Figure 6-15). The voltage on C1 will continue to rise as long as the pulse on point A is high.

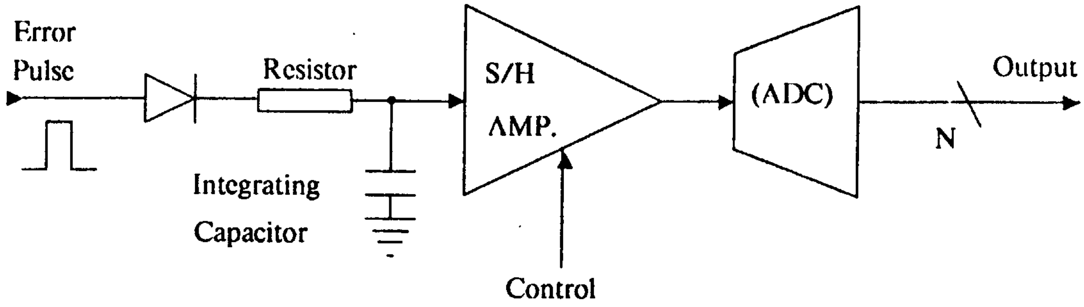


Figure 6-15:Schematics of the analog pulse width gauge.

After the error pulse goes low, the control logic locks the value of the voltage on Cl in the sample-and-hold amplifier by toggling the STROBE input. The control logic then starts a conversion cycle of the ADC. The proper selection of Cl and R1 is essential for the correct operation of the gauge. The time constant (R x C) of these components should be about ten times the maximum width that an error pulse can have. This is important because the integrand voltage should be approximately a linear function of the width of the error pulse (Figure 6-16).

A prototype of the above was constructed and tested. The prototype failed. The digital word produced had no relation to the width of the error pulse. Furthermore, the value changed with every sample. A bad side effect of choosing the time constant according to the above criterion is that the value of the integrated voltage is much less than the high voltage of the error pulse (Figure 6-16). For example, if the error pulse had TTL levels, then the value of the integrated voltage will be in the range of tenths of volts which makes it sensitive to ground noise. However, dropping the constraint on the time constant will require the probe to perform complicated

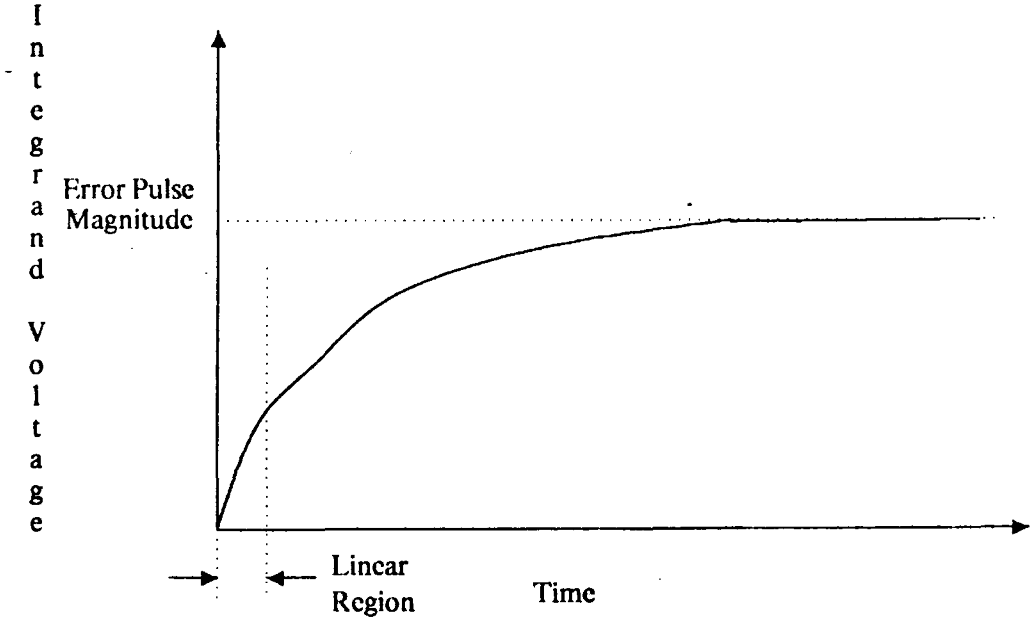


Figure 6-16:Time constant selection of C1 and R1.

calculations on the digital word as the integrand voltage becomes an exponential function of the pulse width.

In the constructed circuit, the magnitude of ground noise was higher than the calculated integrand voltage. In addition, because the integrated pulse was very narrow, about 70ns, and because the gauge can only integrate one pulse for every sample, the total charge integrated in C1 was very low. With low charge, the resulting voltage is severely degraded by capacitor leakage.

Note that with some signal conditioning and sensitive buffering stages, the gauge might be made to work. However, these additions to the simple analog gauge would have required board space that is not available in the portable probe.

From the above results, or rather the lack of any result, CloDS uses a more concrete digital gauge. The resolution of the digital detector is less than that of this analog gauge. However, the digital gauge has a feature that was not present in the simple analog gauge: it works.

6.6 Two-Pulse Coding

This section describes circuits proposed for encoding and decoding the clocks with two narrow pulses per encoded clock pulse. The following circuits were proposed when the specifications of CloDS demanded that both the rising and the falling edges of the clocks be synchronized at the receiving ports. During the development of CloDS, the requirement that the falling edges of different clocks be synchronized was dropped. Consequently, the following circuits became more complicated than needed and were never built. The circuits were included here in case the specifications for a future CloDS demanded synchronization of both edges.

6.6.1 Theory of Operation

In this method CloDS sends a narrow pulse that coincides with the rising edge of the encoded clock and another narrow pulse that coincides with the falling edge of the encoded clock. On the receiving end, the narrow pulse stream toggles a flip-flop on, on the arrival of one narrow pulse, and off, on the arrival of the following pulse, thus recovering the clock (Figure 6-17).

Using this method introduces the problem that although the 32 recovered clocks will have exactly the same frequency and have edges within 10ns of each other, it is possible for some of them to be exactly 180 degrees out of phase with the others. In other words, a pulse that is triggering a rising edge in one port might be triggering a falling edge in another.

This problem can be solved if the receiving ports know of an upper bound on the main clock frequency. If we assume that 5MHz is an upper bound, then the ports know that the shortest time between any two pulses is greater than 100ns. Therefore, if the receiving ports should receive a pulse before 100ns has elapsed since the arrival of the previous pulse, they can use it as a phase indicator and they can set

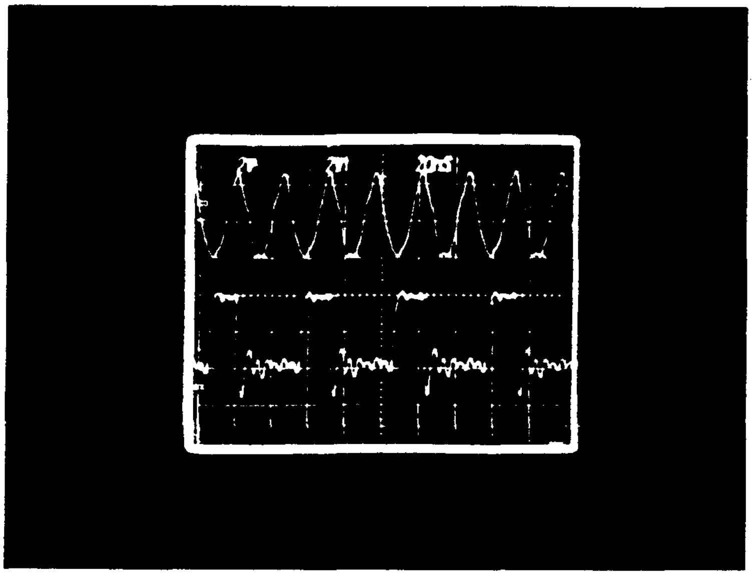


Figure 6-17:Photo of the pulse stream and the recovered clock.

their state so that the next pulse would result in a high-to-low transition of the clock.

After all the ports have been synchronized, this synchronizing pulse will have no effect on the phase of the recovered clock. However, these neutral pulses can be counted and used by the receiving ports to recover a message hidden on the clock line. For example, if we assign the number 3 to a synchronous common message, then this message will be detected at the 32 ports at the same time immediately after counting three neutral synchronizing pulses.The counting is stopped and the message is identified immediately after the clock receives two clock pulses which are more than 100ns apart with no pulses between them, an event that signals the end of a transmission of a message, as shown in the timing diagram in Figure 6-18.

6.6.2 The Clock and Message Encoding Circuit.

The clock and message Encoding circuit has two inputs and one output. The two inputs consist of the main clock input and of a 4-bit data bus with a strobe representing the message input. The output of this circuit is one line carrying

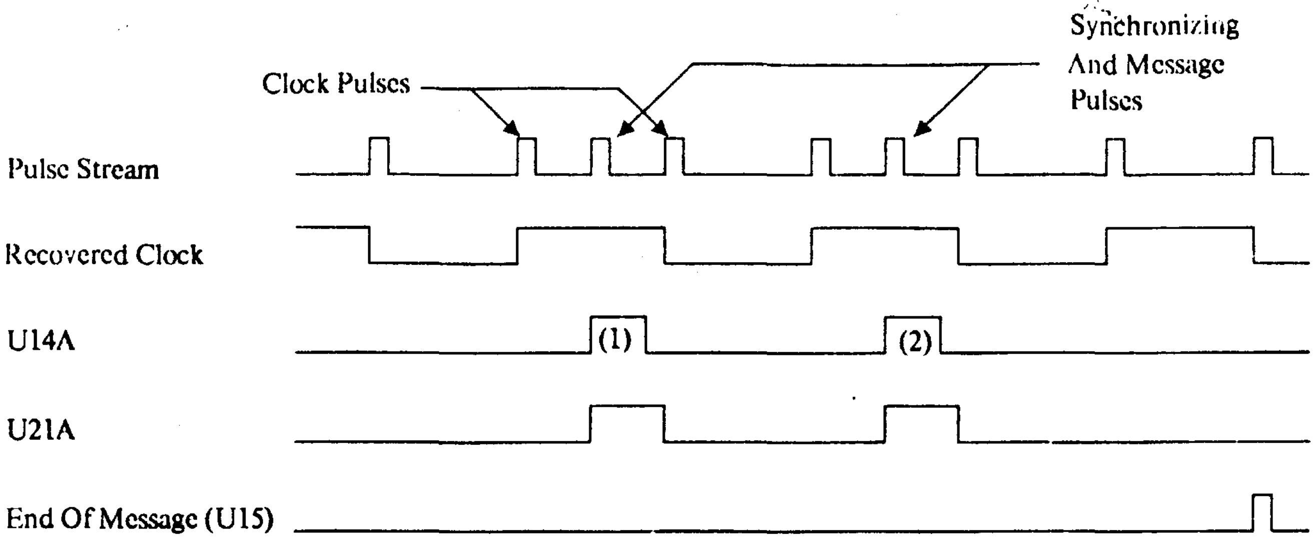


Figure 6-18:Timing diagram of the decoding circuit.

information representing both the clock and the encoded message, as shown in Figure 6-18. The main oscillator may be of any type. U1a is added to normalize the shape of the waveform to a stream of pulses about 20ns in width, as shown in Figure 6-19.

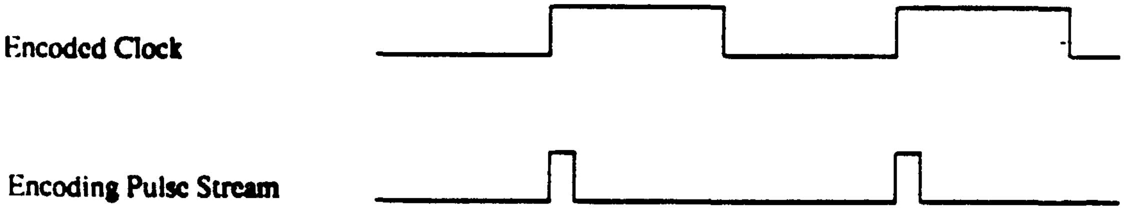


Figure 6-19:Timing diagram of pulse normalization.

In addition, U2b is a divide-by-two counter indicating the phase of the main clock at the receiving ports. This is important because if the ports are synchronized to have the same phase as this counter, then the message encoder will use this information to send the neutral synchronizing pulses at the right time, i.e., before a pulse that causes a high-to-low transition, and, therefore, does not affect the phase of the clocks at the ports.

To describe the operation of the message encoder, let us take the case right after power-up, at which time some port clocks will be in phase with U2b but others will not. To synchronize the clocks, the binary number 0001 is loaded into U3. This is done by presenting the 4-bit inputs with 0001 and holding the LOAD line low until U2b has gone through a low-to-high transition. Assuming that the timing of the LOAD signal has been done correctly, U3 will contain the number 0001. This causes TC (pin 15 on U3) to go high, enabling CEP (pin 7 on U3). At this time, and as soon as U2b goes high, all the inputs of U6a will be high and U2a is enabled by its CLR going high. At the same time, the pulse that caused U2b to go high causes U1b to be set. After the delay of 50ns U1b is reset and the negative output (pin 6 on U1b) goes high. This forces U2a to generate a pulse about 20ns wide and about 50ns after the last clock pulse. When U2b goes high for another time it decrements the counter (U3).

In this example a binary 0001 was loaded, decrementing once causes TC (pin 15 on U3) to go low and prohibits the sending of any other synchronizing pulses. Both the clock pulses (from pin 5 on U1a) and the synchronizing pulses (from pin 5 on U2a) are ORed by U4a and then normalized by U9a to have a width of about 10ns.

It can be seen here that the delayed pulses, referred to as synchronizing pulses elsewhere, are never sent except when U2b, which is the flip-flop holding the phase of the clock at the ports, is high. For this reason, sending any synchronizing pulses after the initial phase matching will not affect the phases of the clock at the ports, as these pulses will always precede a pulse causing a high-to-low transition, which is the state the ports will set themselves to at the reception of a synchronizing pulse. It can be seen also that loading U3 with the binary number 0100 will result in sending exactly four synchronizing pulses.

6.6.3 The Clock Recovery Circuit.

As shown in Figure 6-20, the isolation transformer is immediately followed by a clipping diode to short out any negative voltages that come out of the transformer. The signal is then fed to the clock input of a flip-flop (pin 3 U10a). This is the divide-by-two counter that recovers the square wave from the stream of pulses sent to the ports. The signal is sent through adjustable delay lines to account for the variations in cable lengths and component delays of the various ports.

Because the lines between the CloDS and the ports can be between 4 to 40ft, the variation in propagation delay between the circuits of different ports could be as high as 60ns. This is because the propagation speed through a twisted-pair line is approximately 1.5ns per foot.

The included delay lines in the clock recovery circuit consists of a delay line with a delay of 10ns per tap up to 70ns, used for course adjustment, followed by a delay of 2ns per tap up to 14ns, used for fine clock-edge adjustment. The delays of these lines are programmed by dip switches. Coming out of the second delay line (U12), the recovered clock is fed to a FAST driver to increase the fan-out. Because the clock comes out of a 74F244 gate, the FAST manual should be consulted whenever the number of gates that this clock signal can drive is unknown.

6.6.4 The Message Decoding Circuit.

The message decoding circuit aligns the clock phase with the master reference. In addition, the message decoder should decode any messages sent on the clock lines

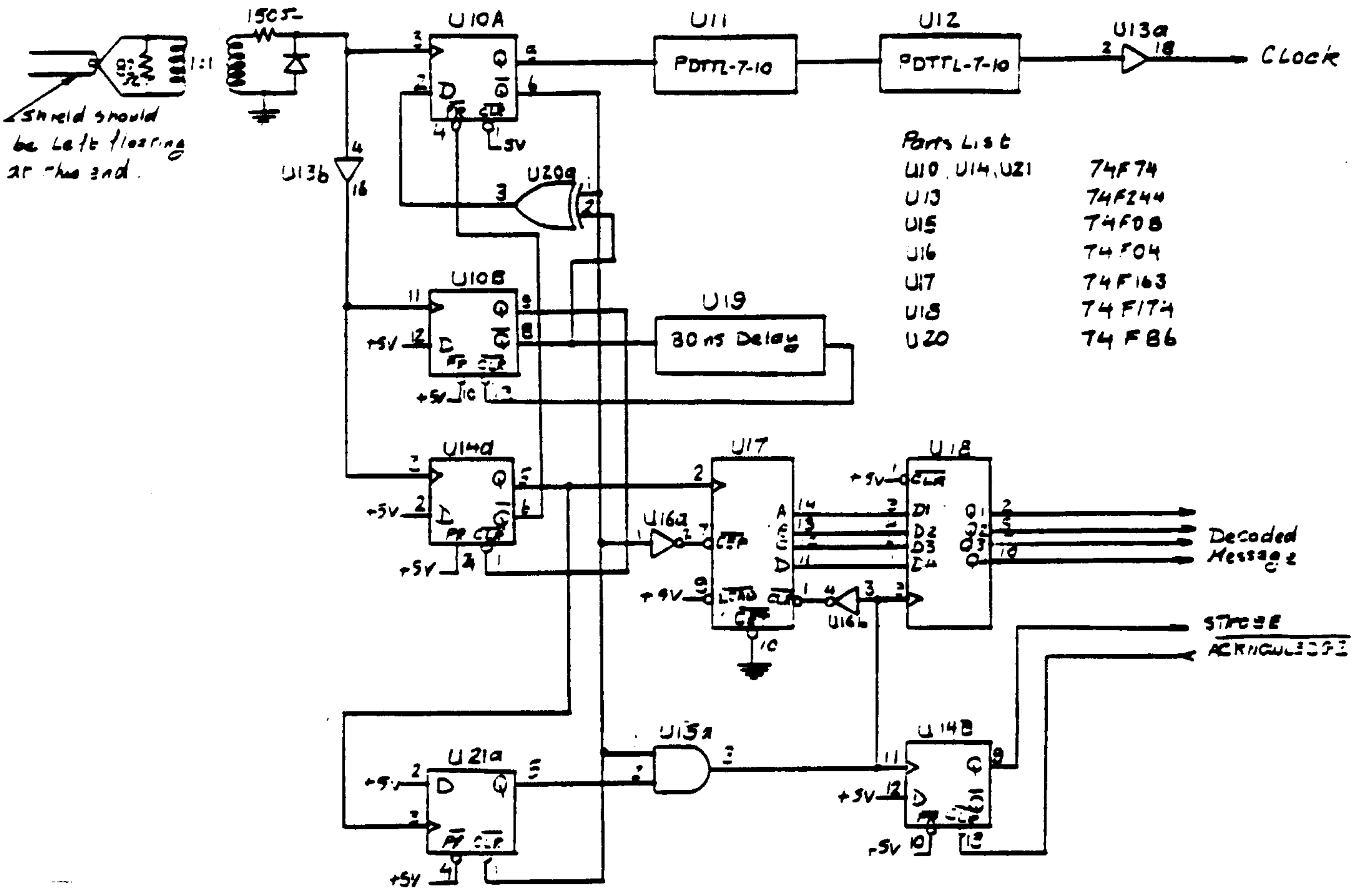
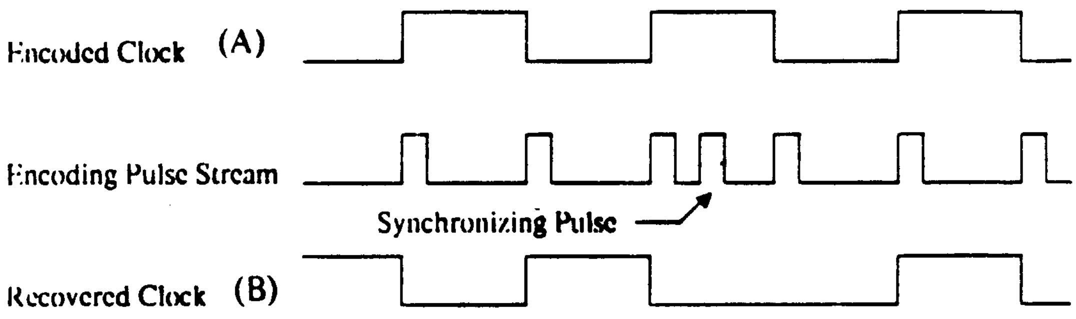


Figure 6-20:Diagram of the clock recovery circuit.

and present them to the processor.

In the event that no synchronizing pulses are sent, all pulses reaching the port will be at least 100ns apart. U10b will always be reset before the arrival of any pulse because U10b resets itself after 80ns of the arrival of the last pulse (see Figure 6-20). With U10b reset, the XOR gate (U20a) acts as an inverter and any pulse reaching the port toggles the state of the clock (pin 5 on U10a). In contrast, U14a never changes state, because at the arrival of a pulse, its CLR is held low by U10b, which is reset at the time. Therefore, PR of U10a is always held high.

If a pulse arrives less than 80ns after a previous pulse, U10b will not have had enough time to reset and the pulse will not alter the clock phase because the XOR will be acting as a non-inverting buffer. This pulse will set U14a which will remain high until U10b resets itself. U14a going high causes U10a to be preset. As a result, a pulse following a synchronizing pulse always results in a high-to-low transition of the clock regardless of its previous state, thus the name *synchronizing pulse*, as shown in Figure 6-21. Note that a synchronizing pulse will affect the phase of the clock only if, on its arrival, the clock was low.



(A) is an in-phase clock at a different port.
(B) is an out-of-phase clock at a port that is synchronized after the arrival of a synchronizing pulse.

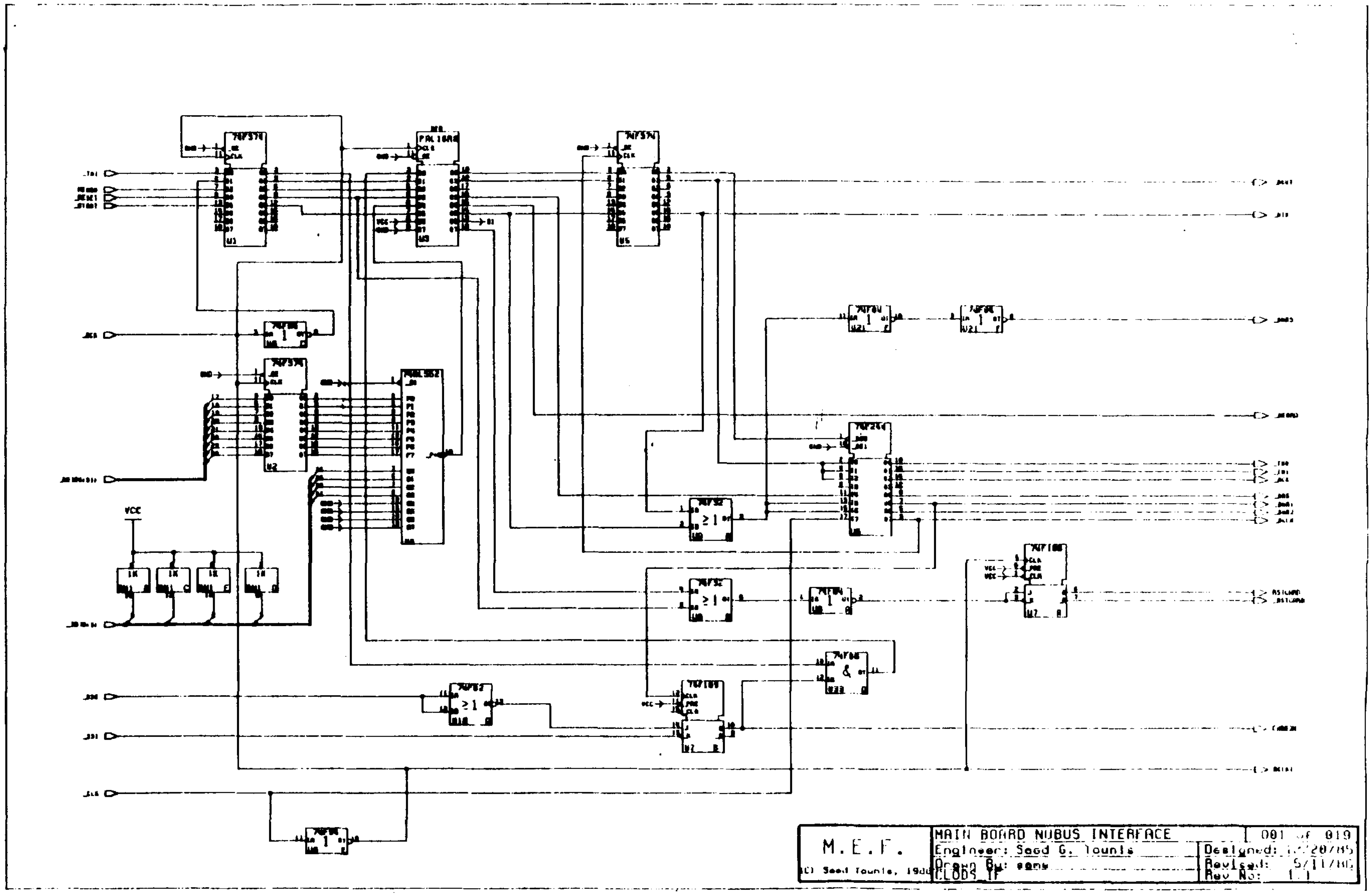
Figure 6-21: Phase synchronizing of recovered clocks.

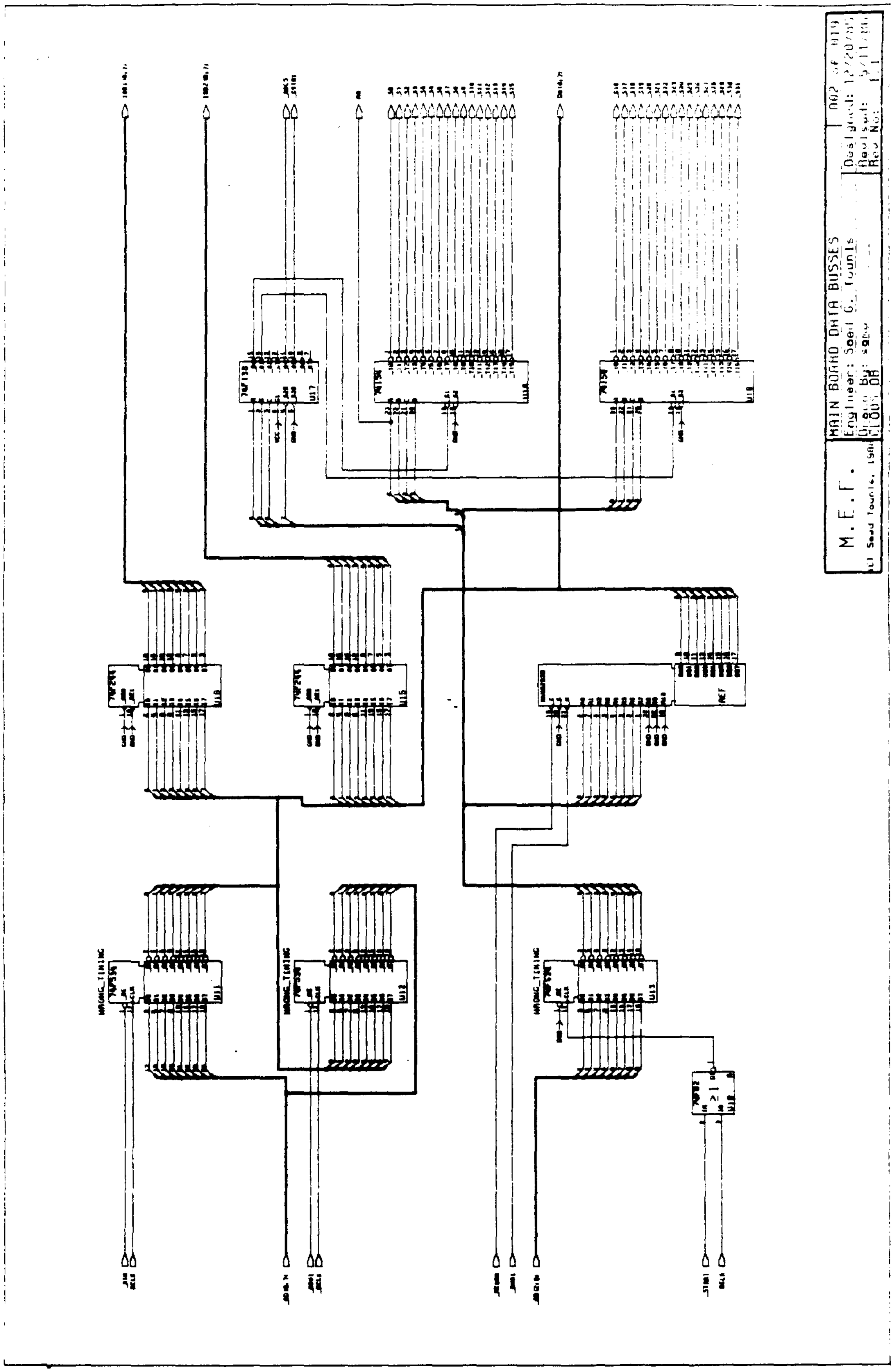
Assuming that the clocks at the different ports have already been synchronized, and are in-phase, let us consider the operation of the message decoder. The arrival of a synchronizing pulse sets U14a high until U10b resets itself. This short lasting pulse at the output of U14a causes the counter (U17) to increment. This continues to happen until the transmission of the synchronizing pulses sizes. This is detected by receiving a pulse no sooner than 100ns after the last pulse. At this time, the output of U21a will be low and the output of U15 will go high on the falling edge of the recovered clock. This causes the counter output to be latched in U18 and U14b to be set indicating the reception of a message. This will also reset the counter so that a different message can be received. It is interesting to note that the output of U14b will go high on exactly the same falling edge of the clock at all the 32 ports. This means that the messages received are synchronous and can be used to initiate actions of a synchronous nature.

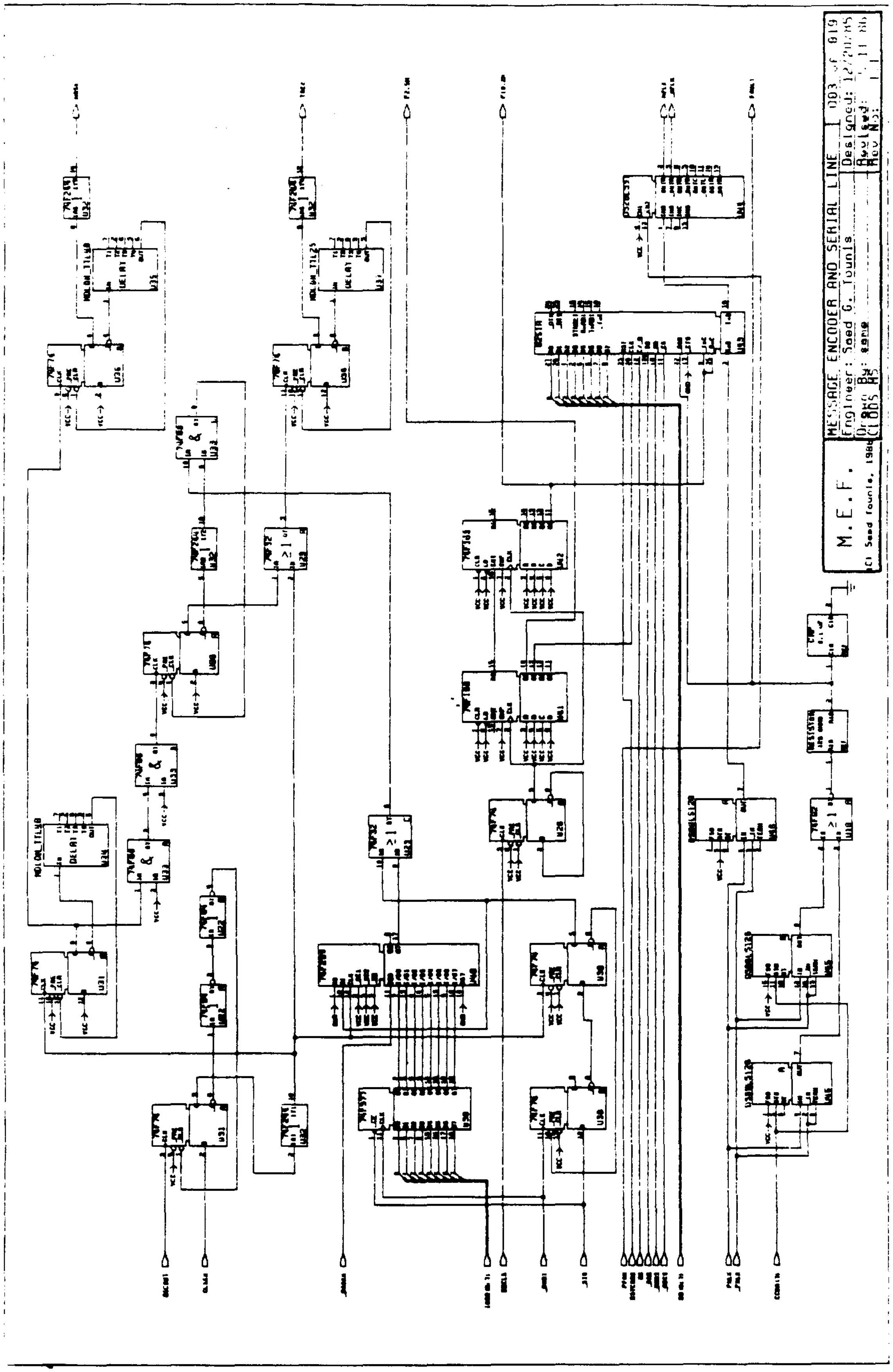
Appendix A

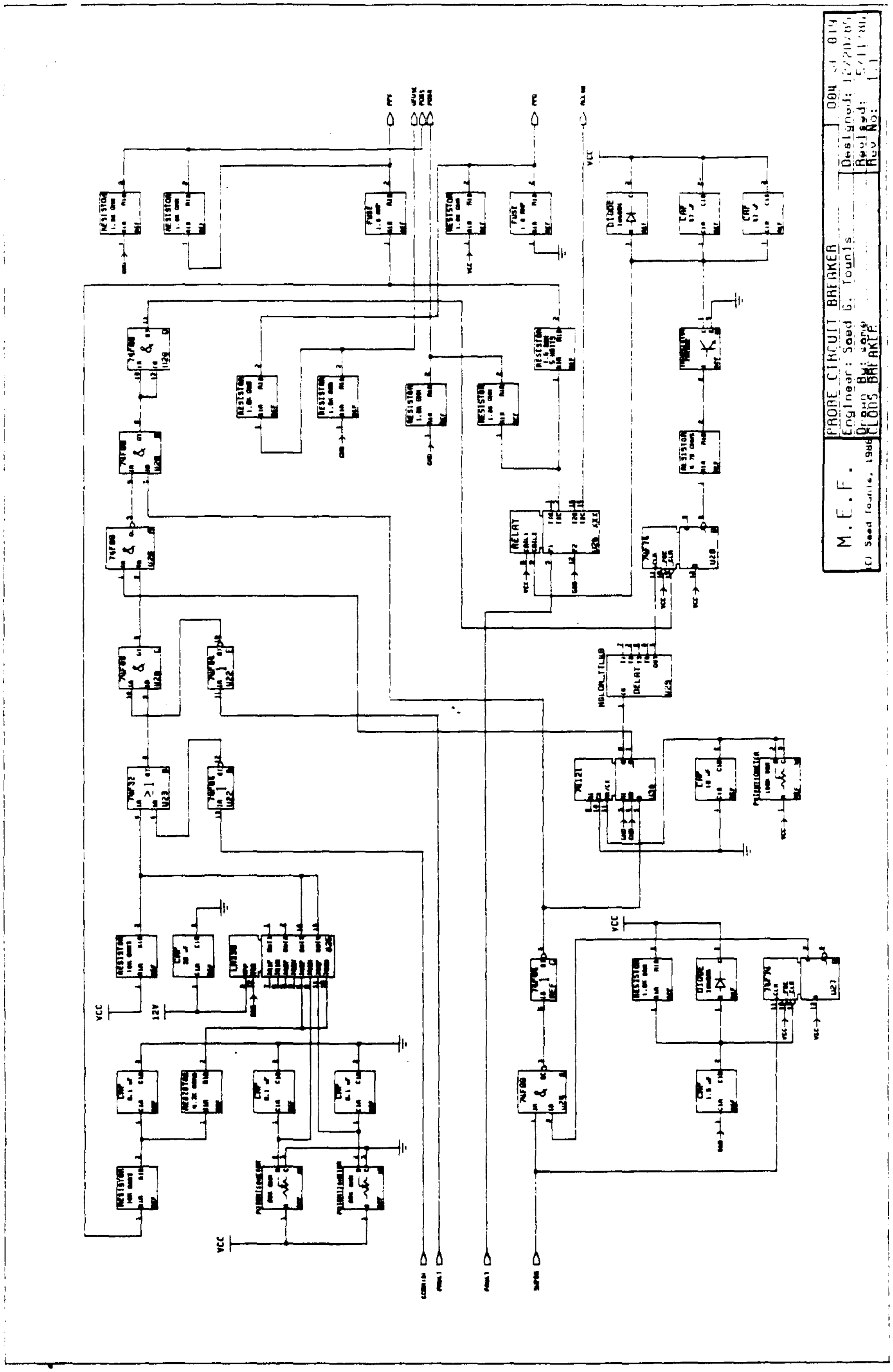
CloDS Schematic Diagrams

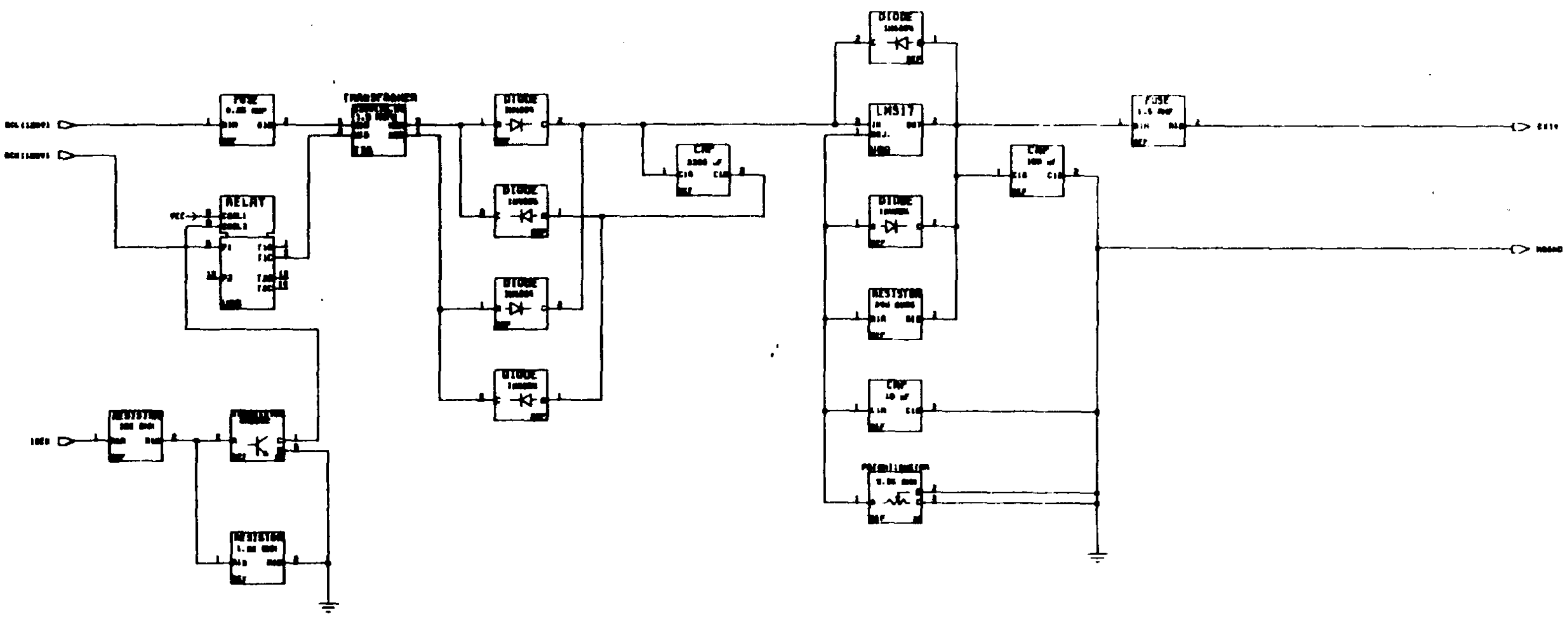
This appendix contains the schematic diagrams of the main board of CloDS, the calibration probe, and of a clock recovery port.


74F374
PAL16R8
74F374
74F04
74F374
74ALS52
74F04
74F04
74F244
74F32
74F32
74F04
74F109
74F08
74F109
74F02
74F04
VCC
GND
M.E.F.
(C) Saed Younis, 1986
MAIN BOARD NUBUS INTERFACE
001 of 019
Engineer: Saed G. Younis
Designed: 12/20/85
Drawn By: same
Revised: 5/11/86
CLOCS IF
Rev No: 1.1


M.E.F.
(C) Saad Tounis, 1986
MAIN BOARD DATA BUSSES
Engineer: Saad G. Tounis
Drawn By: same
002 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1
74F138
74F244
74F374
WRONG_TIMING
U11
U12
U13
U14
U15
U17
U18
74F32
REF


M.E.F.
(C) Saed Younis, 1986
MESSAGE ENCODER AND SERIAL LINE
Engineer: Saed G. Younis
Drawn By: same
CLOOS MS
003 of 019
Designed: 12/20/85
Revised: 1 11 86
Rev No: 1 1
74F74
74F08
74F32
74F04
74F194
74F299
74F373
MDLDM_TTL40
MDLDM_TTL25
DELAY
DS26LS120
DS26LS31
RESISTOR
CAP
U31
U32
U33
U34
U35
U36
U37
U38
U39
U40
U41
U42
U43
U44
U45
U46
VCC

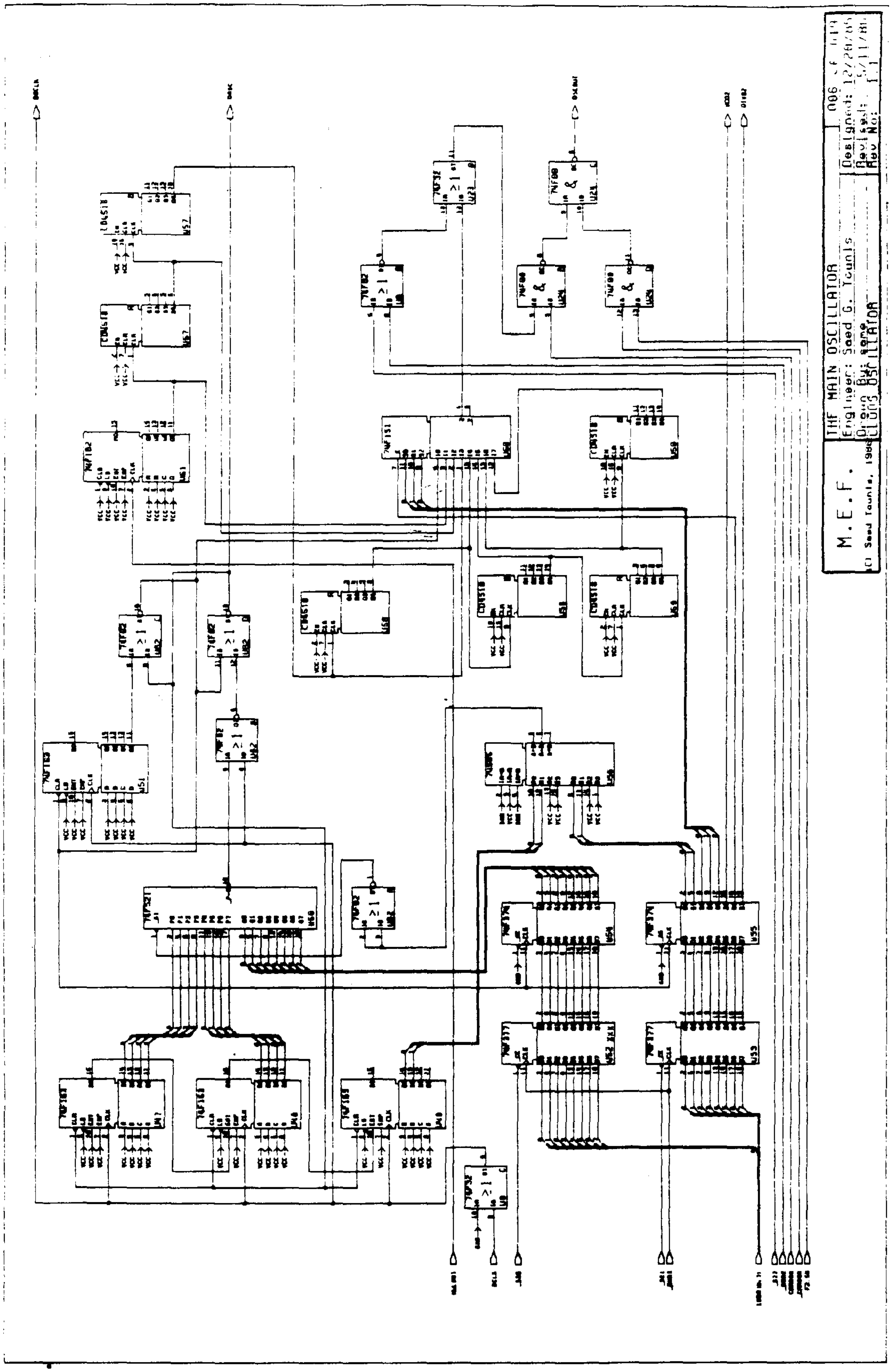

M. E. F.
(C) Seed Younis, 1988
PRORE CIRCUIT BREAKER
Engineer: Seed G. Younis
Drawn By: same
CLODS BREAKER
004 of 019
Designed: 12/20/85
Revised: 5/11/88
Rev No: 1.1
VCC
12V
GND
RESISTOR
CAP
FUSE
DIODE
RELAY
POTENTIOMETER
74F00
74F04
74F32
74F74
74121
U20
U22
U23
U27
U28

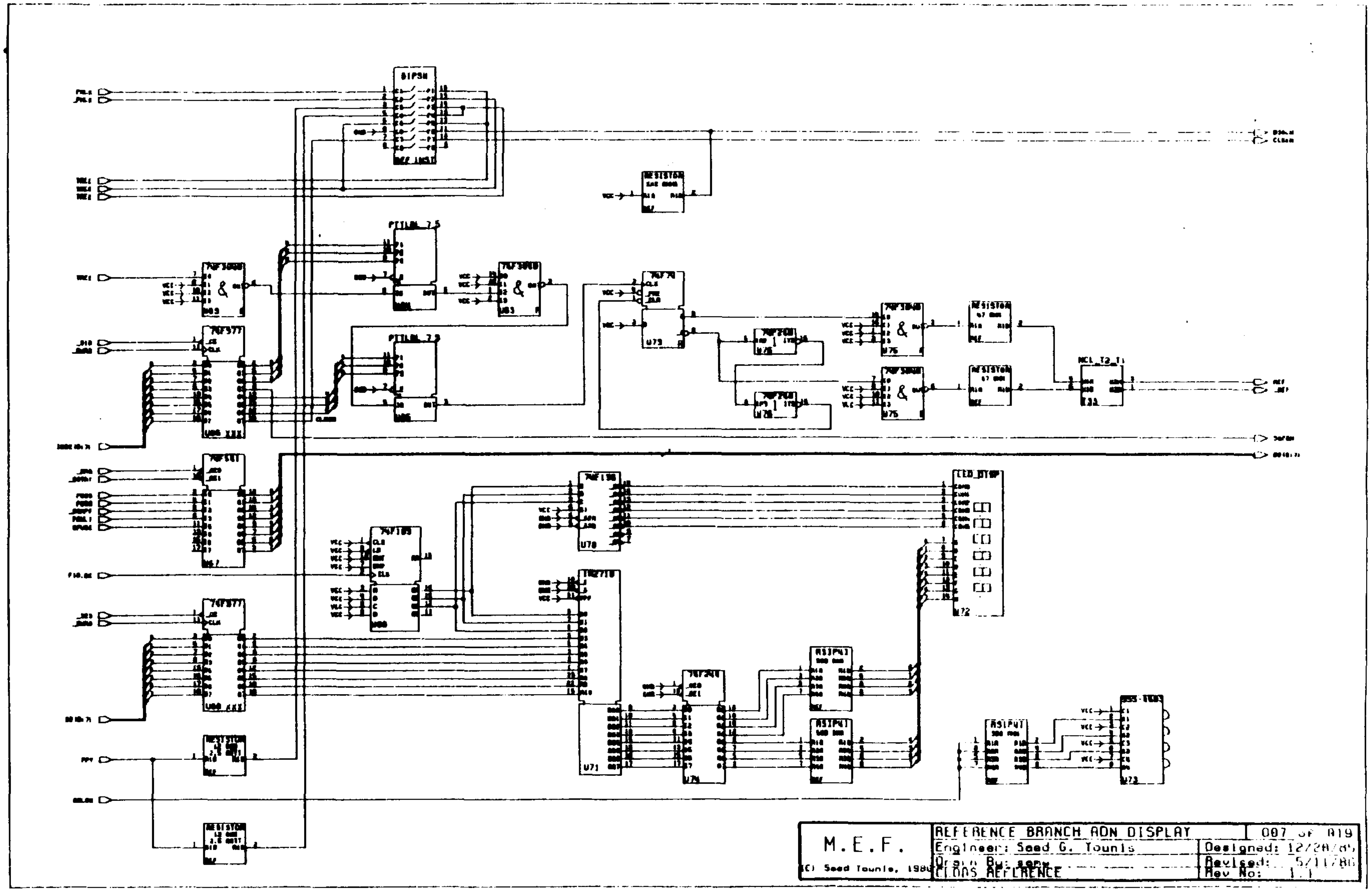


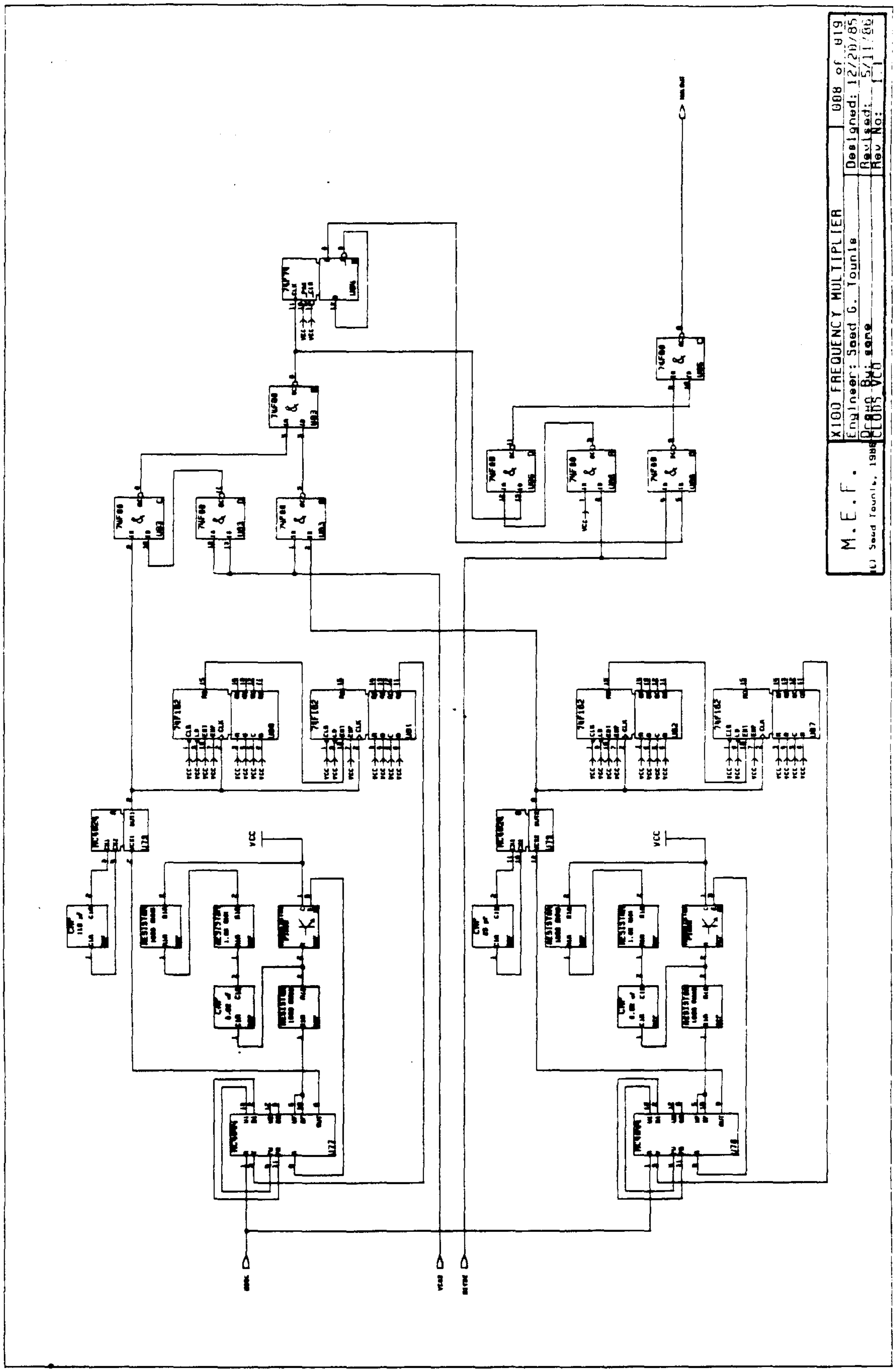
M.E.F.
(C) Saed Younis, 1988

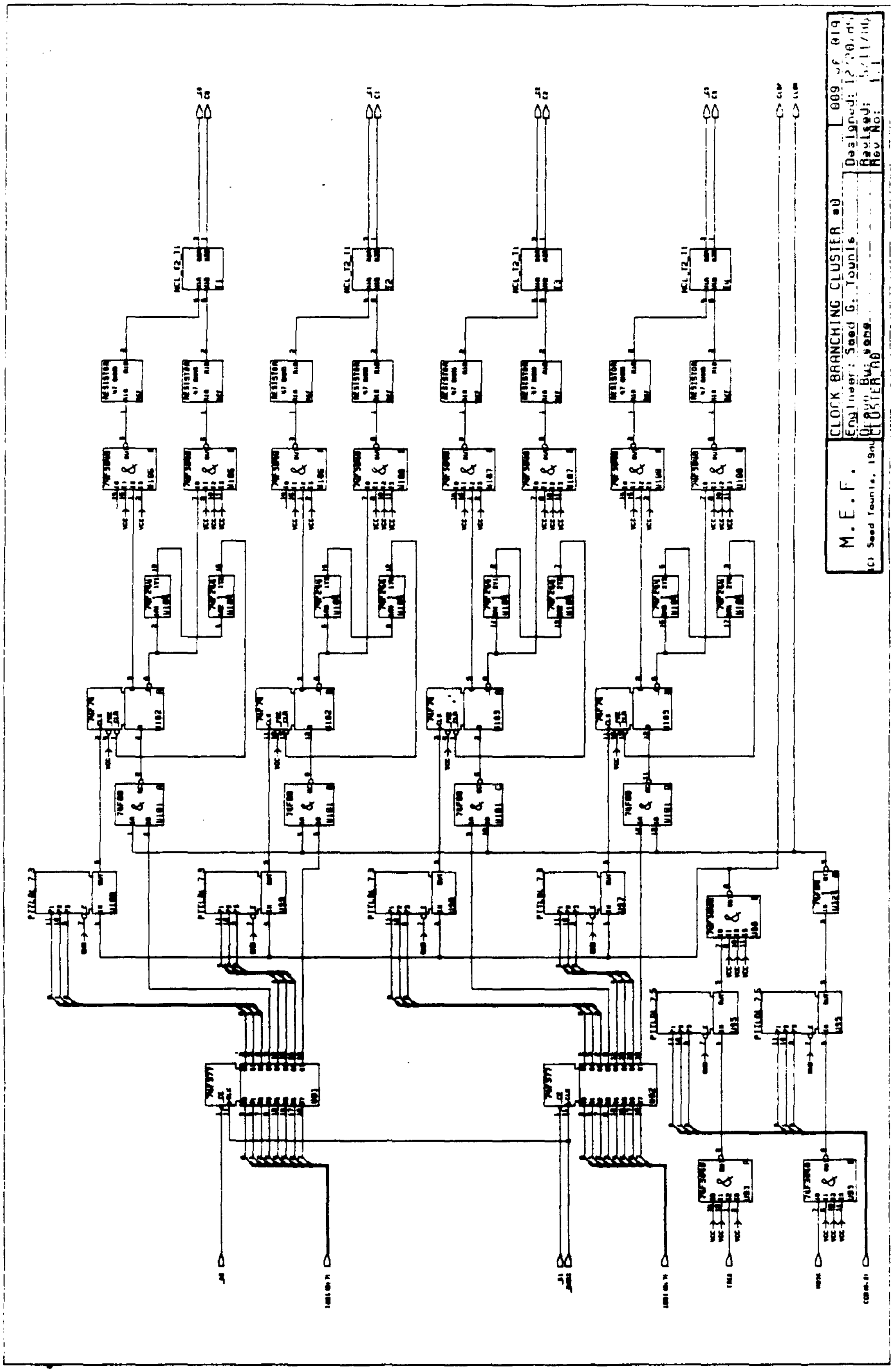
PROBE EXTERNAL POWER SUPPLY
Engineer: Saed G. Younis
Drawn By: same
CLOUS EXTERNAL PS

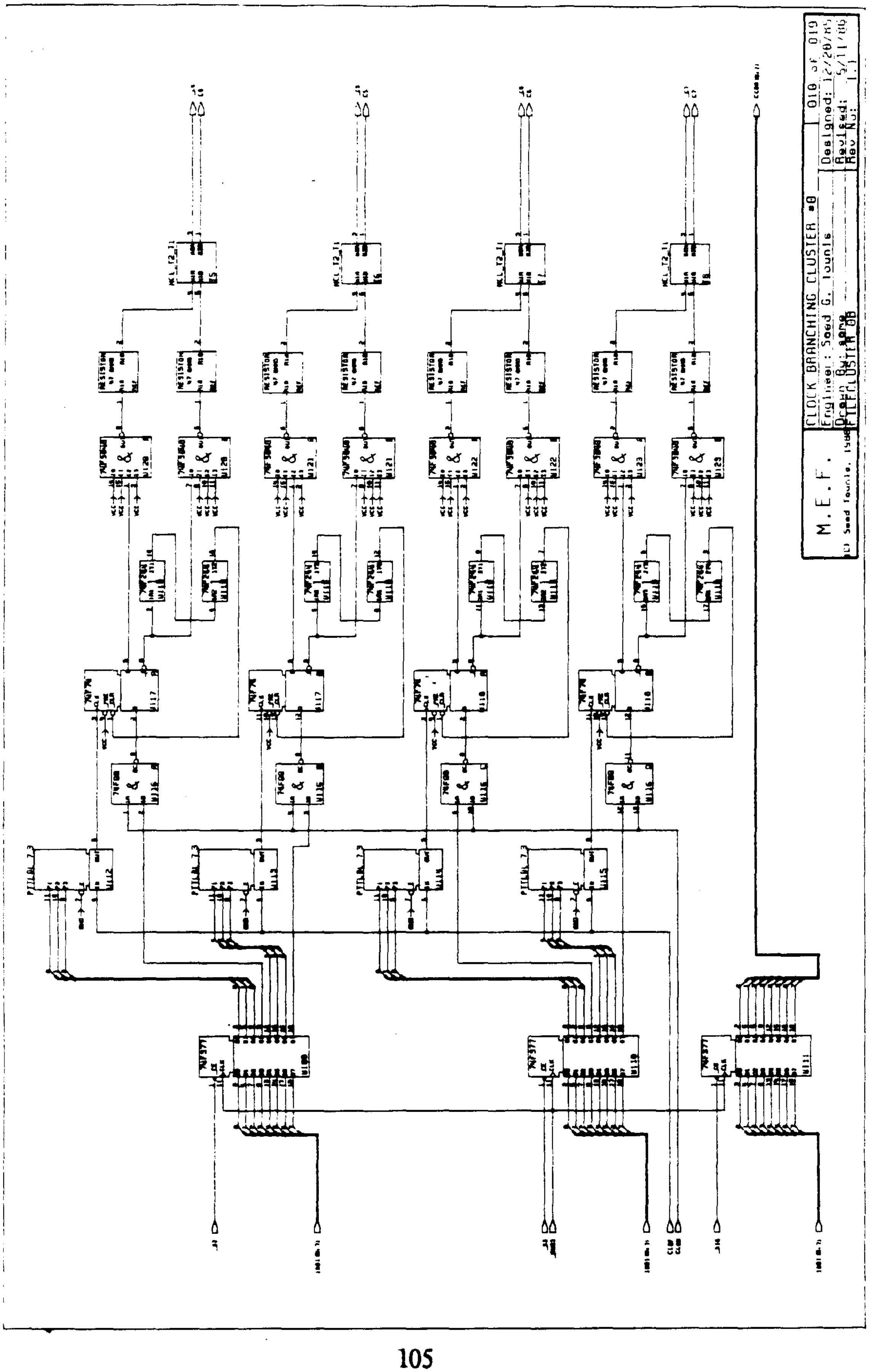
005 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: [.]


M.E.F.
(C) Saed Younis, 1986
THE MAIN OSCILLATOR
Engineer: Saed G. Younis
Drawn By: same
CLOCK OSCILLATOR
006 of 009
Designed: 12/28/85
Revised: 5/11/86
Rev No: 1.1
74F163
74F182
CD4518
74F32
74F02
74F00
74F151
74F521
74F374
74F377
74F86
VCC
GND
OSC
OSCOUT

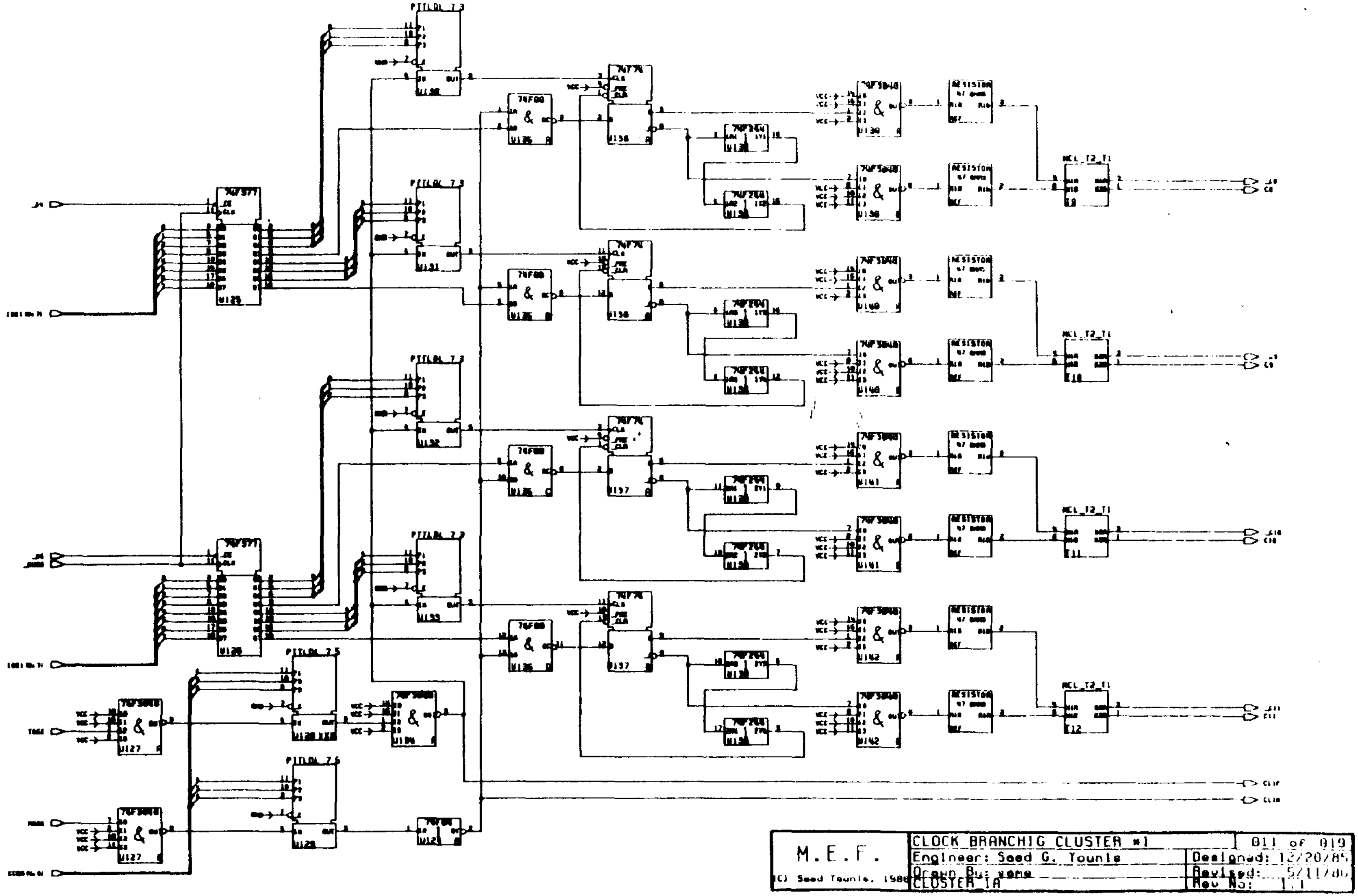

DIPSW
RESISTOR
74F577
74F541
PTILBL_7_5
74F74
74F190
LED_DISP
RSIPW1
M.E.F.
(C) Seed Tounis, 1986
REFERENCE BRANCH ADN DISPLAY
007 of 019
Engineer: Seed G. Tounis
Designed: 12/28/85
Drawn By: seny
Revised: 5/11/86
CLOOS REFLRENCE
Rev No: 1.1


X100 FREQUENCY MULTIPLIER
Engineer: Saed G. Younis
Drawn By: gene
M. E. F.
(c) Saed Younis, 1986
008 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1
74F00
74F182
74F74
VCC
VCC

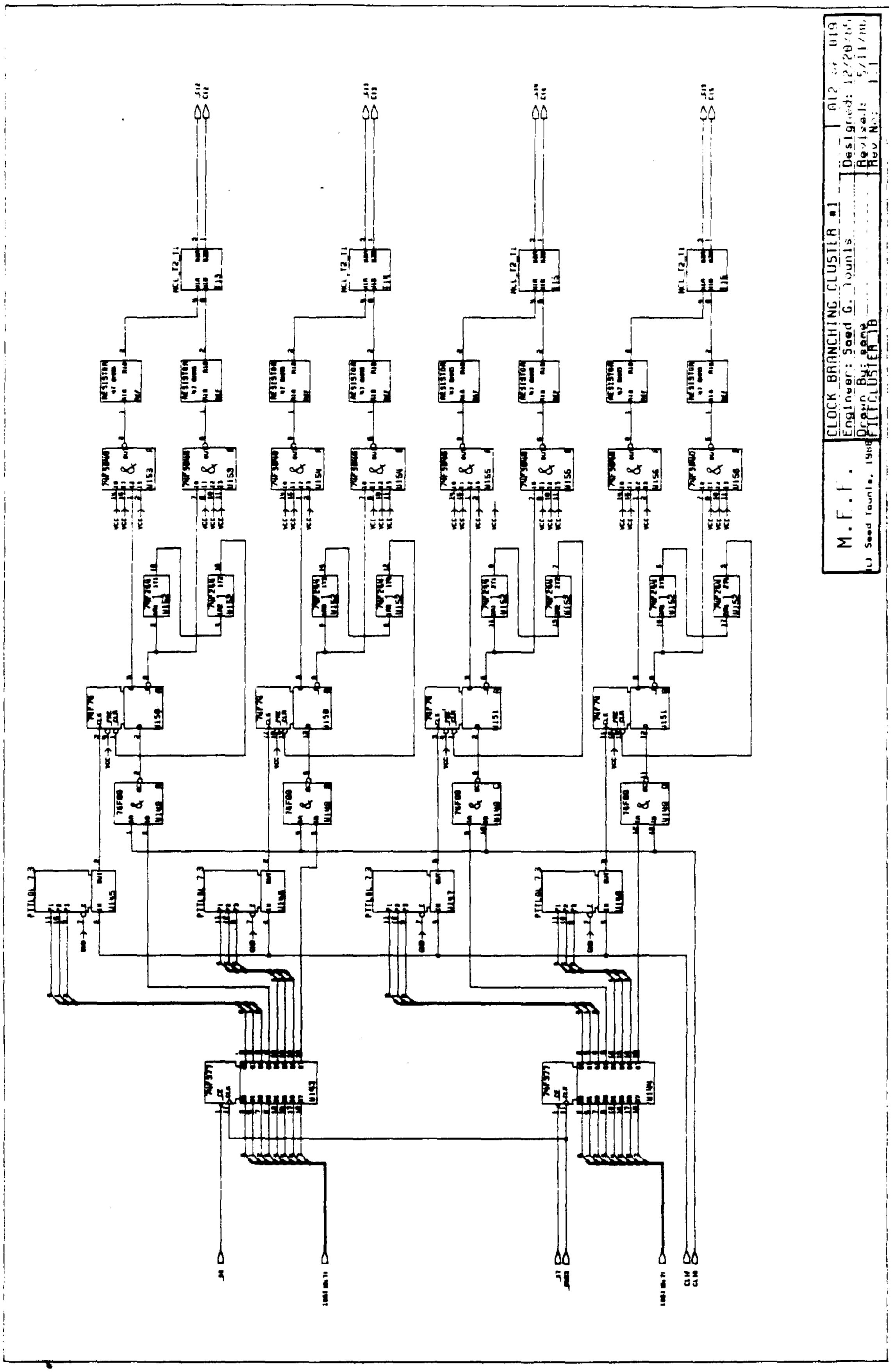

M.E.F.
(C) Saed Younis, 1990
CLOCK BRANCHING CLUSTER #0
Engineer: Saed G. Younis
Drawn By: same
CLUSTER n0
009 of 019
Designed: 12/20/89
Revised: 5/11/90
Rev No: 1.1
74F3040
RESISTOR
MCL_F2_11
74F74
74F00
74F377

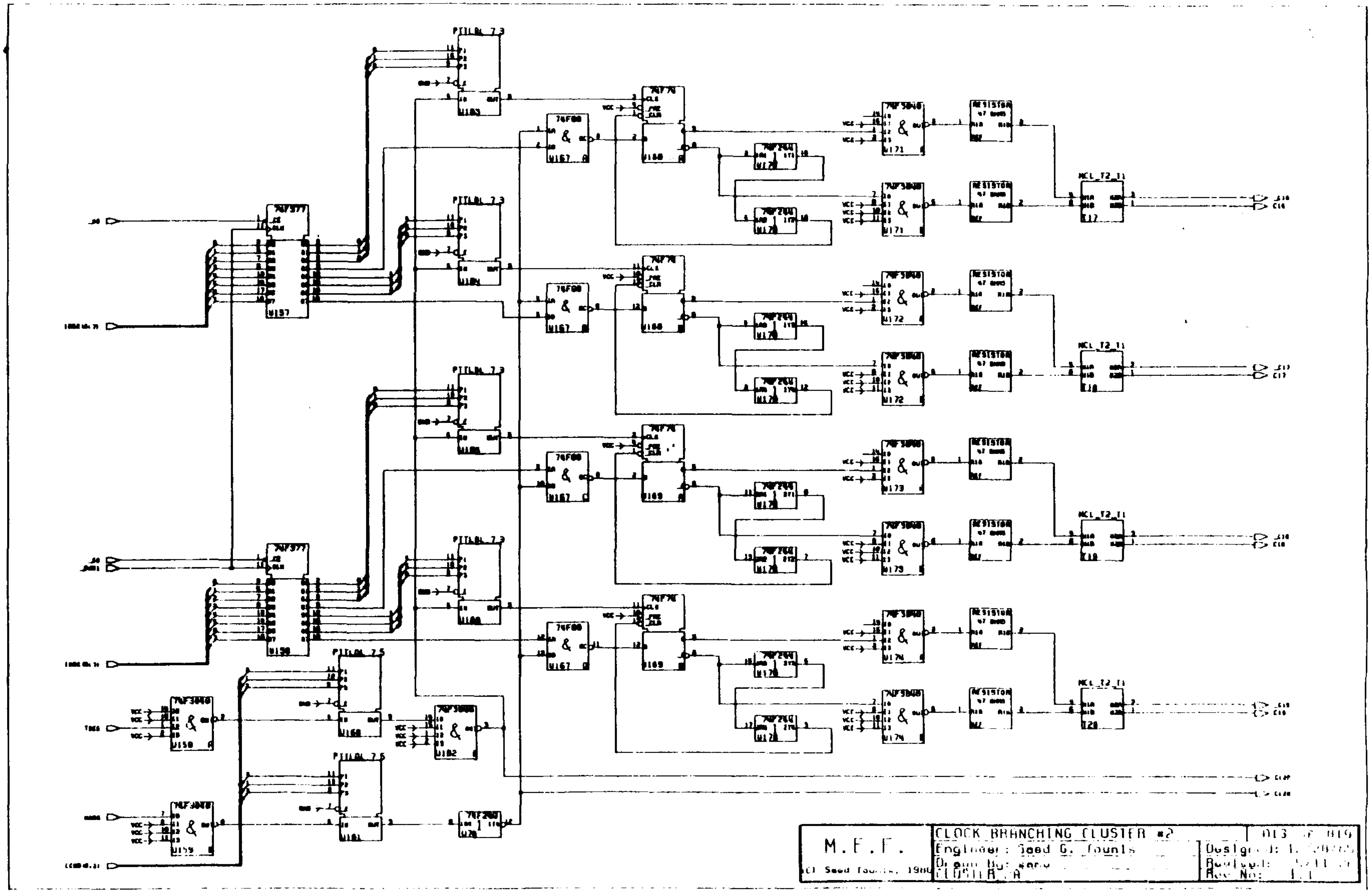


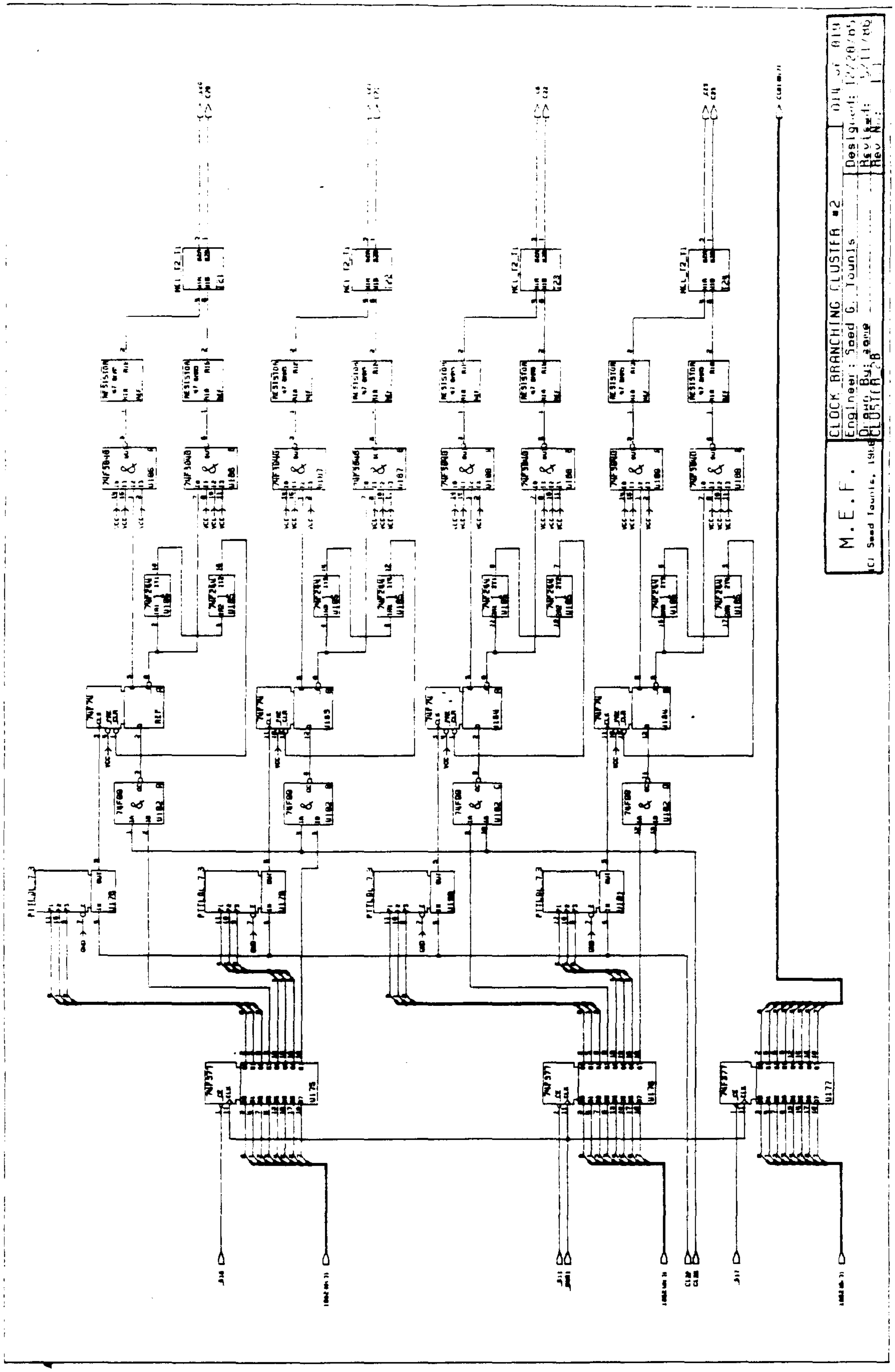
M.E.F.
(C) Seed Tounis, 1986
CLOCK BRANCHING CLUSTER #0
Engineer: Seed G. Tounis
Drawn By: gene
FILE: CLUSTER 0B
010 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1
74F377
74F00
74F74
RESISTOR
MCL_T2_T1

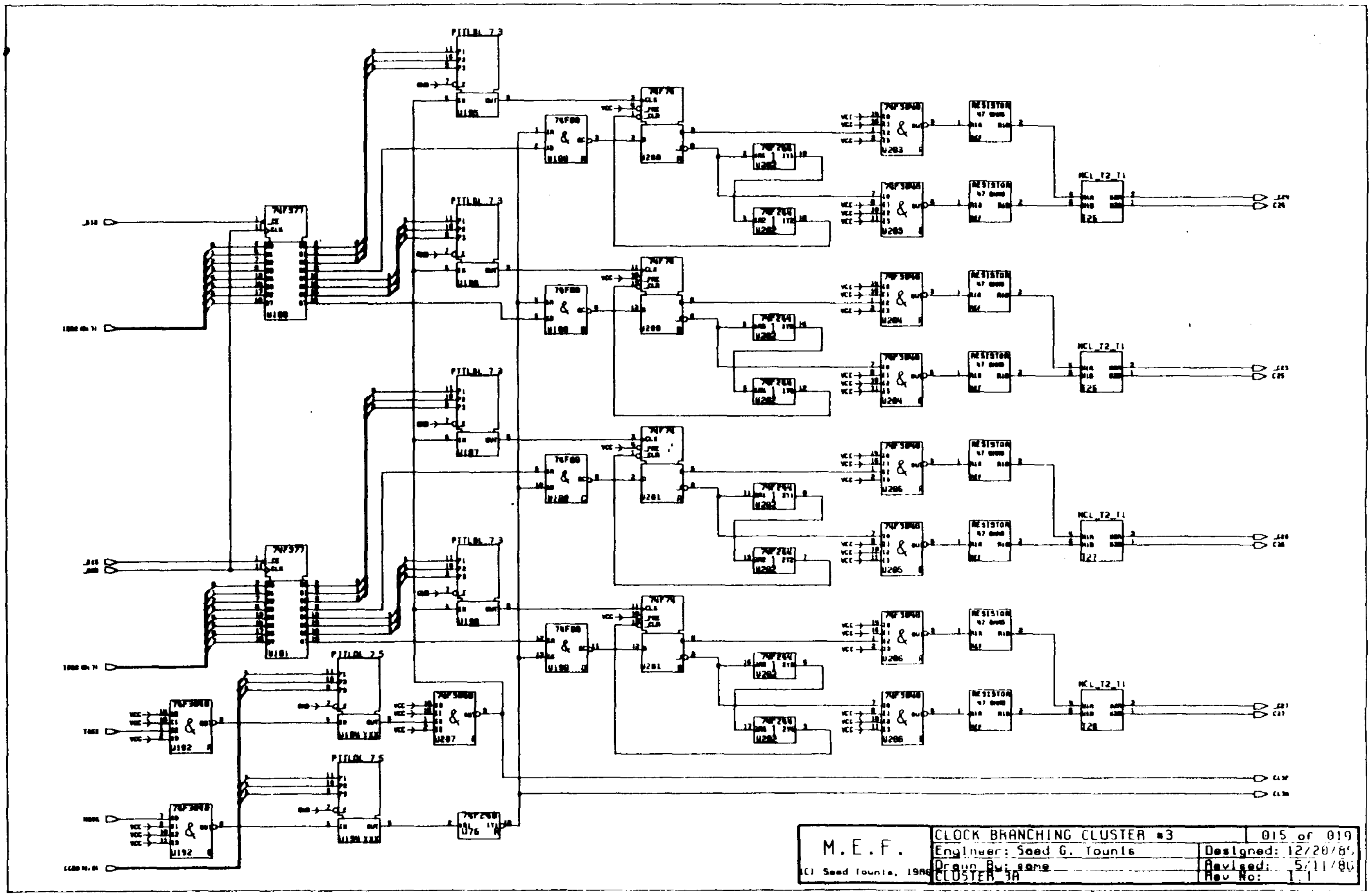


M.E.F.	CLOCK BRANCHIG CLUSTER #1	011 of 019
	Engineer: Seed G. Tounie	Designed: 12/20/85
(C) Seed Tounie, 1986	Drawn By: same	Revised: 5/11/86
	CLUSTER IA	Rev No: 1.1


M.F.F.
(c) Saed Younis, 1988
CLOCK BRANCHING CLUSTER #1
Engineer: Saed G. Younis
Drawn By: same
FILTCLUSTER_1B
012 of 019
Designed: 12/20/88
Revised: 5/11/88
Rev No: 1.1
RESISTOR
74F3040
74F244
74F74
74F00
74F377


74F377
U157
U158
PITLBL 7.3
U183
U184
U185
U186
74F00
U167 A
U167 B
U167 C
U167 D
74F74
U168
U169
74F244
U170
74F3040
U171
U172
U173
U174
RESISTOR
47 OHMS
MCL_T2_T1
E17
E18
E19
E20
PITLBL 7.5
U160
U161
U159
U162
74F260
U70
VCC
GND
M.F.F.
(C) Seed Founis, 1988
CLOCK BRANCHING CLUSTER #2
Engineer: Saed G. Younis
Drawn By: same
CLUSTER_A
013 OF 019
Designed: 1.20/88
Revised:
Rev No: 1.1


M.E.F.
(C) Saed Younis, 1986
CLOCK BRANCHING CLUSTER #2
Engineer: Saed G. Younis
Drawn By: same
CLUSTER 2B
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1
014 of 019
74F374
74F08
74F74
74F244
RESISTOR
VCC
GND



M.E.F.

(C) Seed Tounis, 1986

CLOCK BRANCHING CLUSTER #3

015 of 019

Engineer: Seed G. Tounis

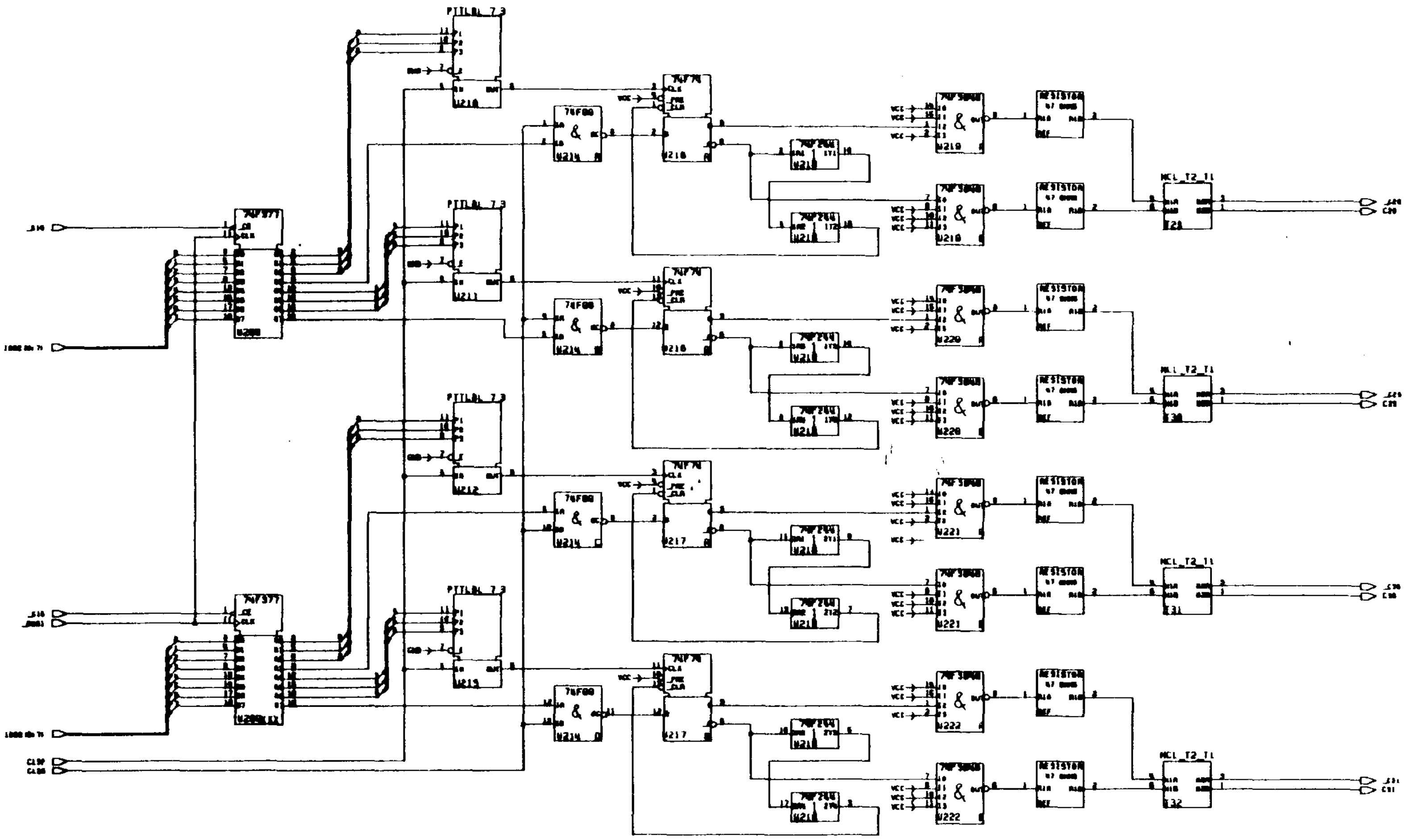
Designed: 12/20/85

Drawn By: same

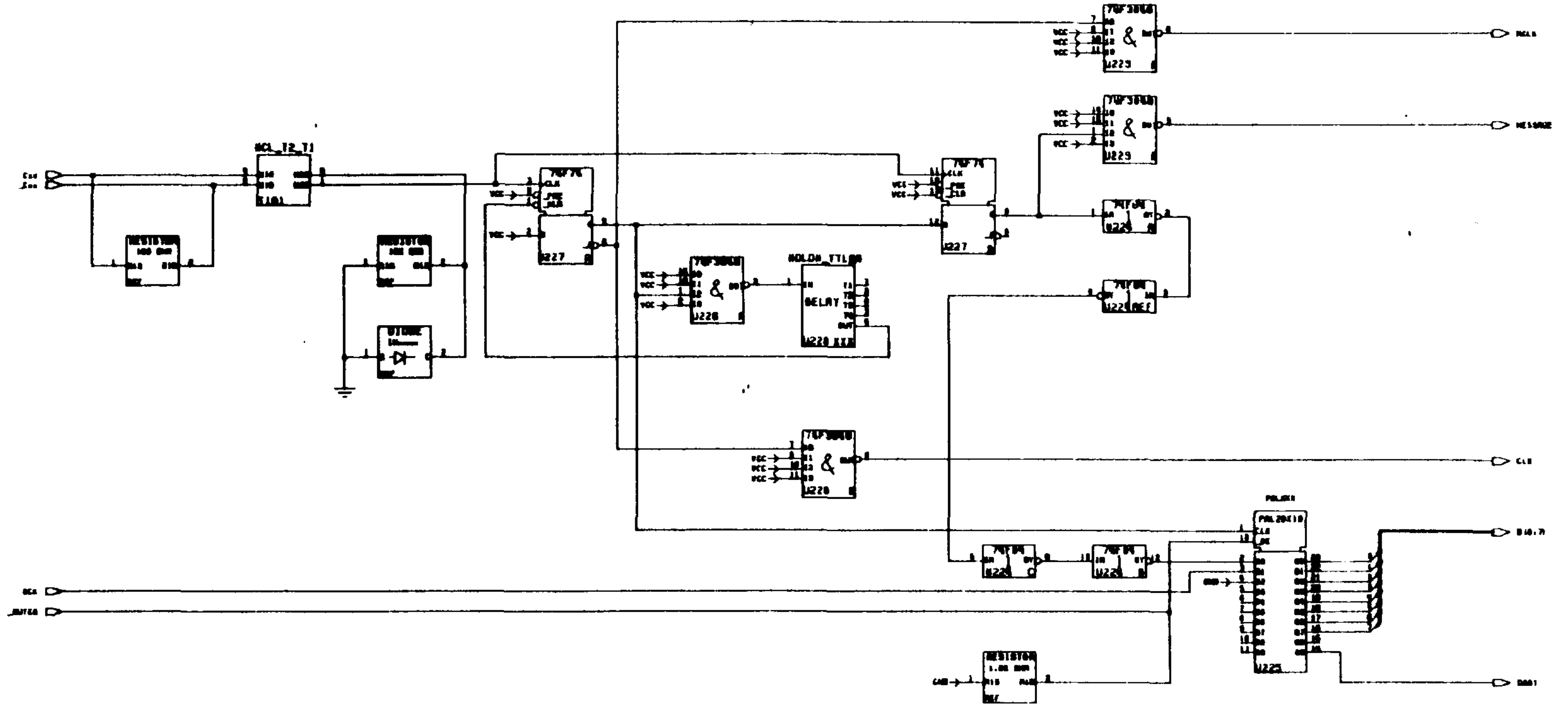
Revised: 5/11/86

CLUSTER 3A

Rev No: 1.1



M.E.F.	CLOCK BRANCHING CLUSTER #3	016 of 019
	Engineer: Seed G. Tounis	Designed: 12/20/85
(C) Seed Tounis, 1986	Drawn By: same	Revised: 5/11/86
	CLUSTER 3B	Rev No: 1.1



M.E.F.

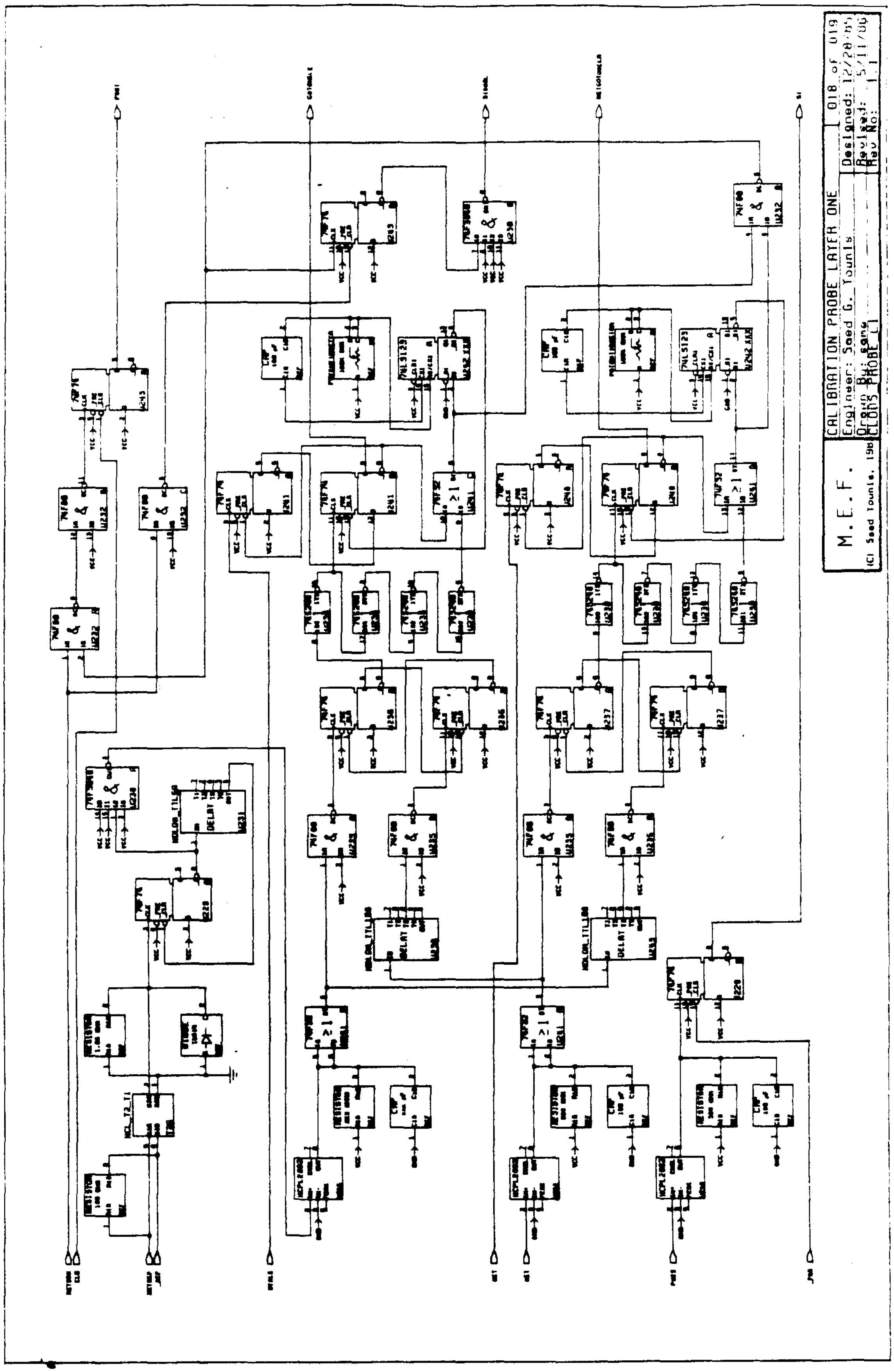
(C) Seed Iounis, 1986

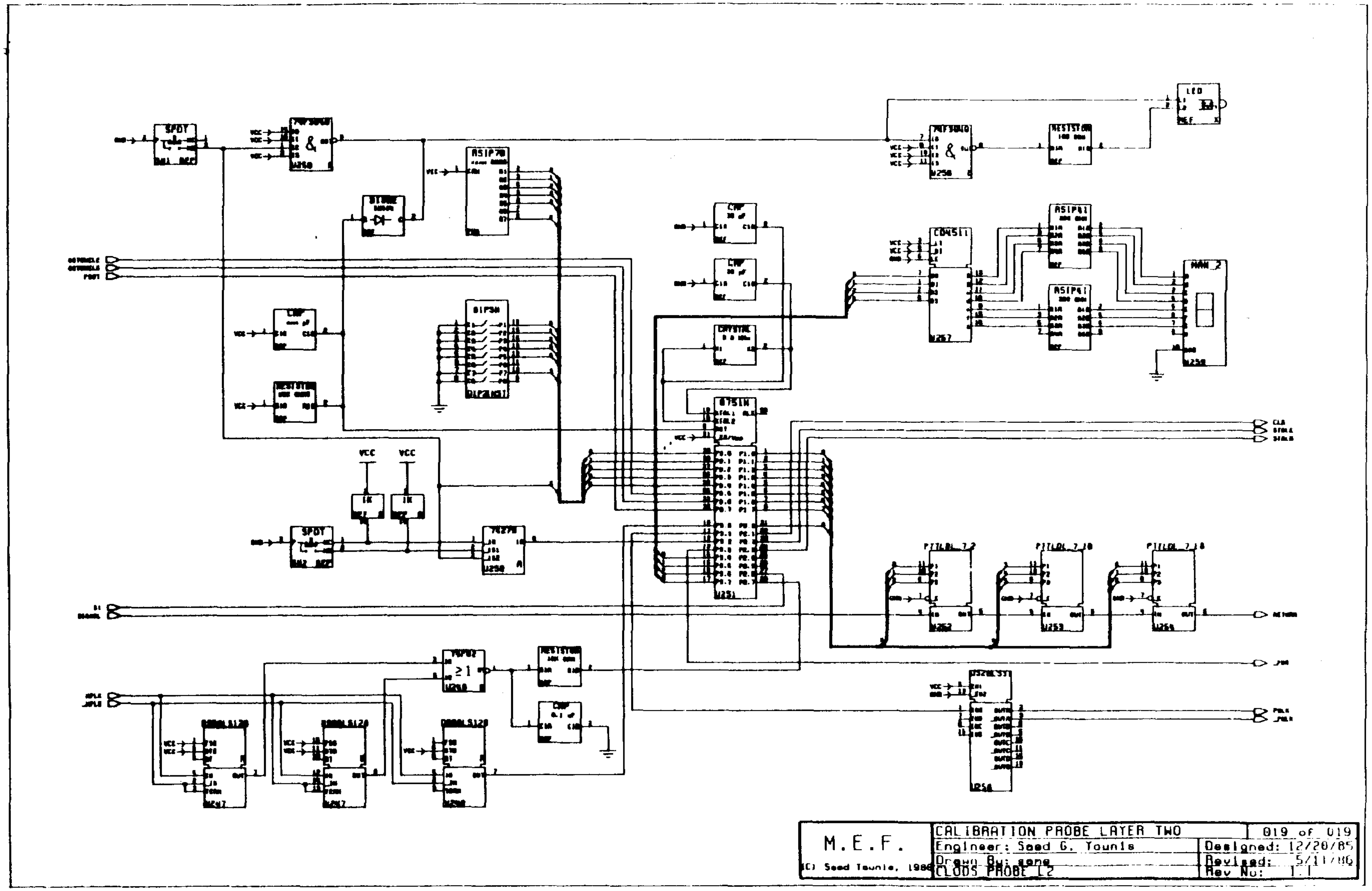
CLOCK AND MESSAGE RECOVERY PORT | 017 of 019

Engineer: Saed G. Iounis | Designed: 12/20/85

Drawn By: same | Revised: 5/11/86

CLODS PORT | Rev No: 1.1


M.E.F.
(C) Saed Younis, 198
CALIBRATION PROBE LAYER ONE
Engineer: Saed G. Younis
Drawn By: same
CLOCS PROBE L1
018 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1


M.E.F.
(C) Seed Tounie, 1986
CALIBRATION PROBE LAYER TWO
Engineer: Seed G. Younis
Drawn By: gene
CLODS PROBE L2
019 of 019
Designed: 12/20/85
Revised: 5/11/86
Rev No: 1.1
SPDT
LED
RESISTOR
CD4511
A51P41
MAN_2
A51P70
DIODE
CAP
CRYSTAL
8751H
DIP5H
74279
74F02
74F30
PITLDL_7_2
PITLDL_7_10
PITLDL_7_18
U2520L531
U250
U251
U252
U253
U254
U256
U257
U258
VCC

Appendix B

Photographs of CloDS Units

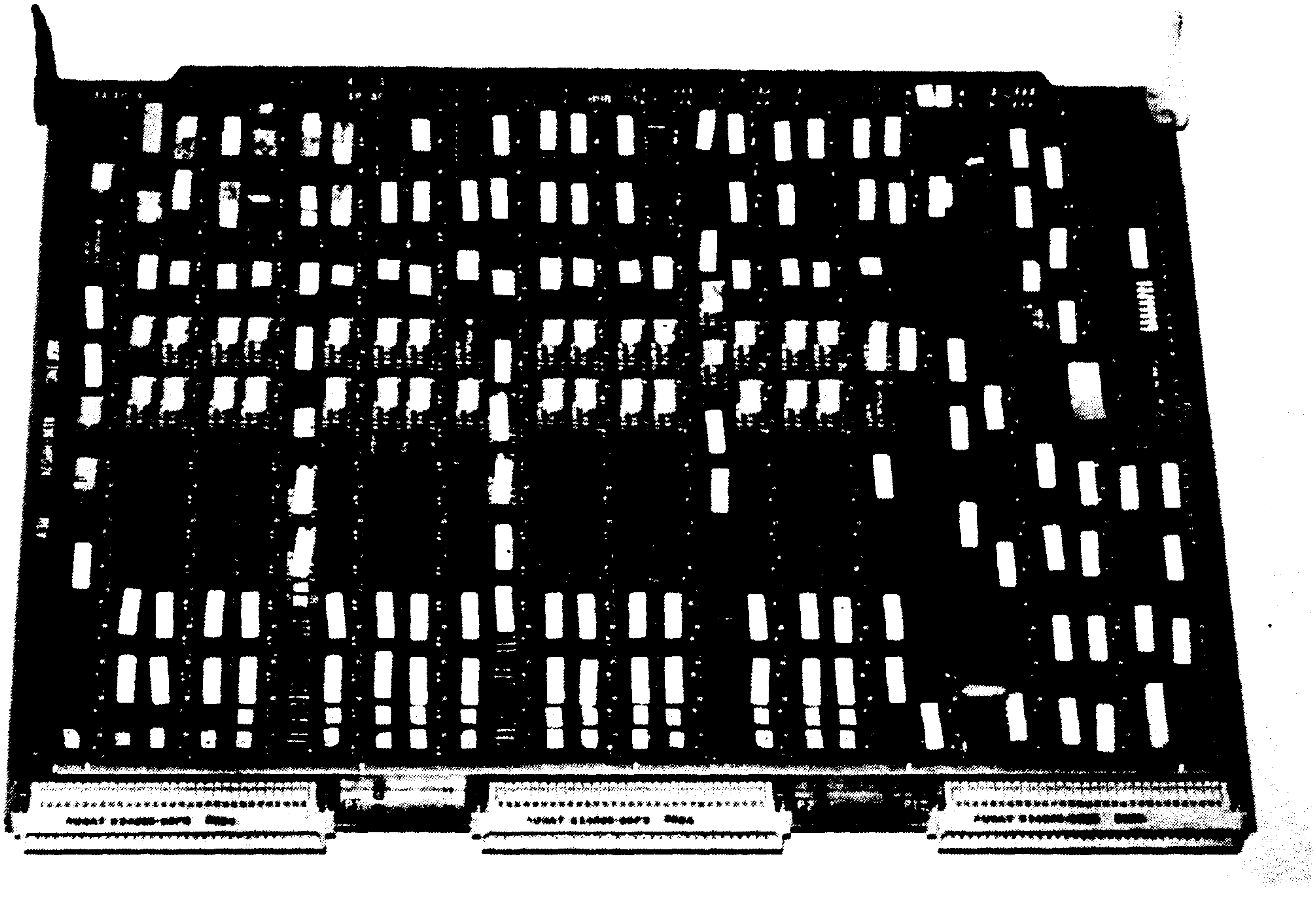


Figure 6-22:Photo of the main CloDS board.

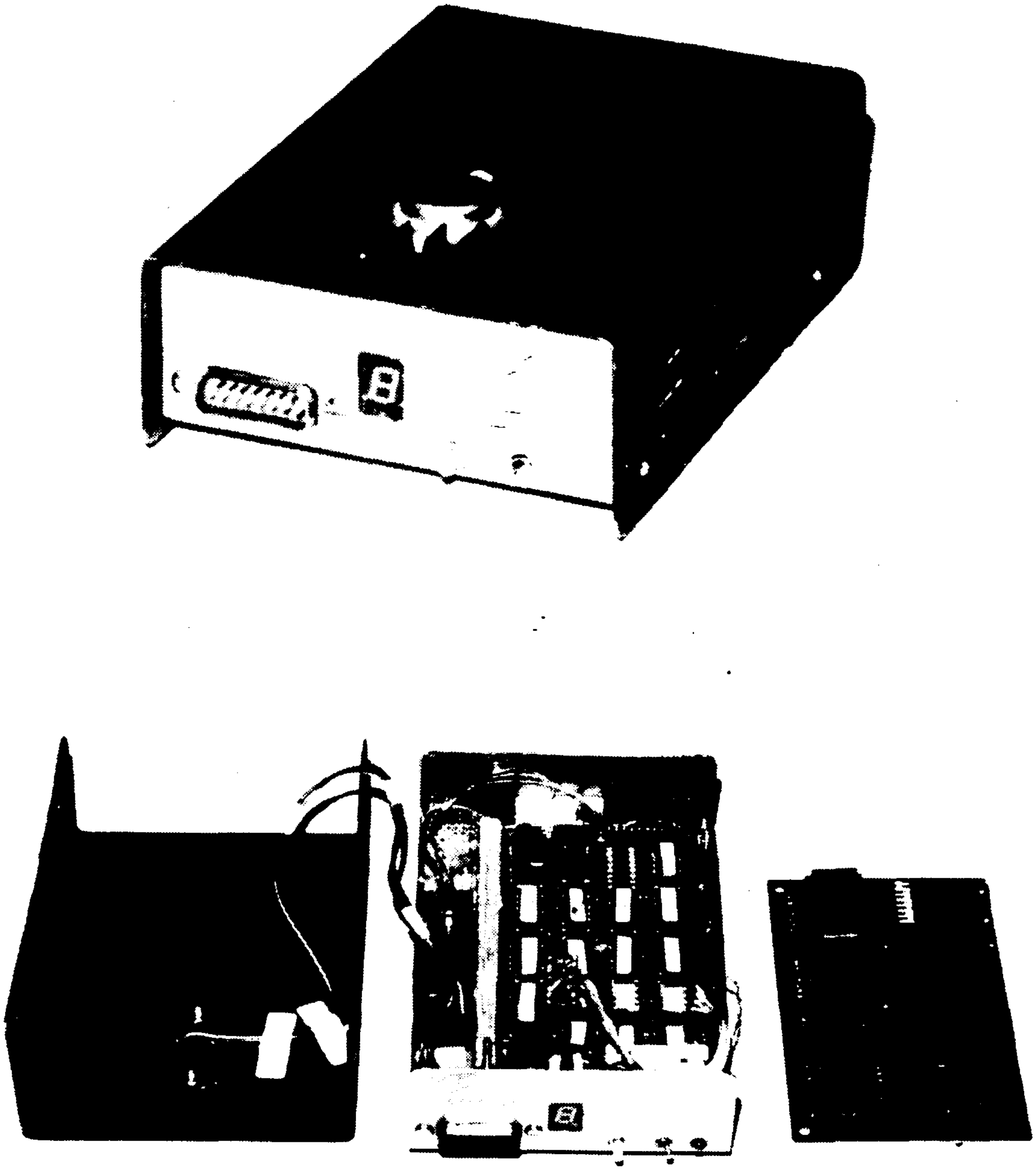


Figure 6-23:Photos of the calibration probe.