REQUEST FOR UNIVERSITY HONORS INDEPENDENT STUDY

Name
RAMSES KHALIL

Student Name

Department, Course No.
9041 INET 371
0608 INET 398E-P

Semester of Registration
SUMMER '79

Local Address
9041
0608

Computer Reference Number

Date of Request
AUGUST 21, 1979

Attach additional pages as needed.

1. Describe below, in detail, the goal(s) of the work you propose. What will you study?

I will study the electronics of a high energy physics research project. This will be done at Fermi National Accelerator Lab, working with experiment E-577 Elastic Scattering.

2. Please list major works dealing with this topic (published materials relevant to your project):

Nuclear Instruments and Methods
All Fermi-Lab experiment publications

3. Describe the methodology of your proposed study. For example, if you plan regular conferences with your advisor, how often? What written work will your study produce? If you will be working in a laboratory, what equipment will be needed and what procedures will you follow?

I will work as an Electrical Engineer, using all the necessary and appropriate equipment such as oscilloscopes, V0 meters, etc. I will produce a written paper with schematics of the electronics involved in E-577 at Fermi National Accelerator Lab.
1. **What tangible evidence of the project's completion will you submit?**
   The Paper (Type) with complete set of Schematics and Block Diagrams.

5. **List the courses you have taken (Honors and non-Honors) which provide a background for this study.**
   
<table>
<thead>
<tr>
<th>ELECTRONICS</th>
<th>HONORS</th>
<th>NON-HONORS</th>
</tr>
</thead>
<tbody>
<tr>
<td>270 270A</td>
<td>17 NIU HOURS</td>
<td></td>
</tr>
<tr>
<td>271 271A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>275 275A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>398</td>
<td></td>
<td></td>
</tr>
<tr>
<td>371</td>
<td></td>
<td></td>
</tr>
<tr>
<td>245</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HOURS</th>
</tr>
</thead>
<tbody>
<tr>
<td>75 NIU</td>
</tr>
</tbody>
</table>

   Including:
   - Year Physics
   - Chemistry
   - Calculus
   etc.

---

**Pamela Khalil**
Student Signature

---

Request Approved:

**[Signature]** 9-1-79

(Instructor) (Date)

I hereby certify that the above mentioned independent study does not duplicate in content and/or method similar material offered in a regular course in this, the semester of enrollment or the immediately preceding or immediately subsequent semesters.

---

**[Signature]** 8/13/79

(Department Chairperson) (Date)

---

(Director, University Honors Program) (Date)

---

(For the University Honors Council) (Recorded)
THE ELECTRONICS OF E-577 (Elastic-Scattering)

FERMI-NATIONAL ACCELERATOR LAB

By
Ramses S. Khalil

For, 398E - Honors - Independent Study
August, 1979
I - A ROUGH BLOCK DIAGRAM OF THE ELECTRONICS

A FORWARD ARM
MIXER WIRE

AMP

A FORWARD ARM
C.R. CARD CHNL

FORWARD DATA (HIT OR NOT)

MIXER, ENCODER, ETC.

16 BIT ACCESS
(SENT TO COMP.
AGAIN IN HIT)

COMPUTER
(PDP 11)

M.T.

READ

M.T. (“READ ME”)

BUSY

LE CROY
FAST LOGIC

(Roy)

PROMPT
RESET

TO FOR. ARM A
C.R. ON TRAILING EDGE OF “BUSY” LEVEL
FROM PDP

PROMPT
RESET

TO BACKWARD
ARM C.R. ON TRAILING EDGE OF PDP BUSY
AND IF EVENT
"NO GOOD" (NO M.T.)

SYNC STROBE (U)

A BACKWARD
ARM
C.R. CARD CHNL

A BACKWARD
ARM
MIXER WIRE

AMP

PROMPT
RESET

BUSY

(1'M BUSY READING SCINTILLATOR 3 EVENTS,
PLEASE DON'T SEND ANY NEW STROBES)
What Goes On Roughly (in E577)

Something comes down the beam pipe hitting the beam scintillators and something trips off the backward arm ("s" for slow) scintillators. This coincidence is called a B. S. When this happens Roy's LeCroy generates a B. S. strobe (within nanoseconds). Since the s scintillators are closer to the Portakamp than the B scintillators, the s signal is delayed in a length of cable to arrive coincident with the B signal (see later section on "Fast Logic"). This strobe is sent to the backward arm C. R.'s and causes the wire chamber information to be latched up. (In order to make sure that you catch the wire signal at the C. R. input when the B. S. strobe comes, delay the signal from the wire amps by running the wire amp signals through a carefully calculated length of delay stripline at the plug in end of the C. R. The width of a pulse that a wire amplifier puts out is about 50 ns. The width of Roys fast logic pulse is about 10 ns, so the B. S. strobe has a little leeway in catching the wire amp. signal.

Meanwhile, something goes through the forward arm. It takes about 700 ns. to make this trip one way the LeCroy decision for a good event, "MASTER RIGGER", comes about 1.5-2 ns after the B. S. strobe. The backward C. R. must hold the wire information for that amount of time.
Layout Of The Readout Racks

**RACK #1**
- Threshold Discriminator & Width Generators
- Fan Funnel
- NIIM TO MECL Converters
- CAMAC crate #3
- CAMAC Power Supply

**RACK #2**
- Fan Funnel
- CR crate #3
- CR crate #2
- CR crate #1
- Fan Funnel
- Coincidence Register (CR) crate #0
- Mixer crate
- Fan module
- Power Supply
- LambdaA model

**RACK #3**
- Fan Funnel
- CR crate #9
- CR crate #8
- CR crate #7
- CR crate #6
- CR crate #4
- Power Supply
### The "C.R. Rack"

A rack is a collection of crates. The "C.R. Rack" looks something like this:

<table>
<thead>
<tr>
<th>CRATE 3</th>
<th>CRATE 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRATE 2</td>
<td>CRATE 8</td>
</tr>
<tr>
<td>CRATE 1</td>
<td>CRATE 7</td>
</tr>
<tr>
<td>CRATE 6</td>
<td>CRATE 4</td>
</tr>
<tr>
<td>MIXER</td>
<td></td>
</tr>
<tr>
<td>POWER SUPPLY</td>
<td>CRATE BLANK</td>
</tr>
</tbody>
</table>

**NOTE:** There is **NO CRATE 5.**

### 6) A C.R. crate

<table>
<thead>
<tr>
<th>CRATE</th>
<th>CR. S</th>
<th>CR. S</th>
<th>CR. S</th>
<th>CR. S</th>
<th>CR. S</th>
<th>CR. S</th>
<th>ENC.</th>
<th>ENC.</th>
<th>BUS WIRING</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.R. Companion</td>
<td>φ-3</td>
<td>4-7</td>
<td>8-11</td>
<td>12-15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** That Companion Cards come in two types - A & B. Each Companion card has 4 C.R.'s.
Apparatus

The experiment (E577) has:

- 32 Proportional Wire Chambers
- 1152 Amplifiers (8 Wires/Amp)
- 144 C. R.'s (4 amps per C. R. card, 32 wires per C. R.)
- 9 Encoders (1 Encoder/crate, 16 C. R.'s/Encoder)
- 1 Mixer
- 1 PDP 11 Computer

A Simplification of Address Generation of Hit Wires (for more detail, see later)

There are 32 wires/C. R. and 16 C. R.'s/crate $16 \times 32 = 512$ wires/crate $\times 10$ crates $= 5120$ addresses. We send the address of a wire to the computer only when to hit.

Each crate has an address that its encoder puts out. This is a number from 0-9. This is sent to mixer, from PDP11.

The C. R. card generator etc. # within a crate (a # from 0-15), passes it to encoder, which gives it to the mixer, then PDP11.

The encoder generates the group address (a no. from 0-3).

The encoder generates the wire address within a group (4 bits - a no. from 0-7).

So, one of the purposes of the encoder is to generate a crate address. Basically, the encoder, mixer, etc., take 3 addresses: (crate, C. R., wire) and make one number out of it.
Purposes of AMP Card

1. To get a wire chamber pulse up to full MECL limits (since the C. R. is MECL) and give it a good push for transmission over a long wire (need line driver at out end).

2. To reject background noise or any other noise from being picked up and keep all hit signals clear.

3. To try to put out the same width and height on every wire for every hit (width - sons). re: to amplify a small signal uniformly from the chambers to the C. R. (coincidence registers)

In order to understand how the card works, we now have to speak about OPAMPS and MECL.

E-577 PWC Amplifier Scheme

We are now ready to go back and discuss how the signal corresponding to a hit gets from the appropriate PWC (Proportional Wire Chamber) wire to the C. R. A chamber wire sets at ground potential before a "hit". When a charged particle traverses the chamber near the wire, it ionized the chamber gas and a stream of electrons deposit themselves on the wire, pushing its potential negative. (For more detail on the operation of the chambers, see the three appendixes from Nuclear Instrument and Method that I've xerozed. The PWC wire is connected to an amplifier channel (there are 8 channels per amplifier card, so each card handles 8 wires). The amplifier drains the negative charge off the wire, allowing it to float to ground again and
become useful. By setting a resistance on the input end of the amplifier that is in series with the wire, we make the width and amplitude of the signal reasonable. This we can describe now; the front end of the amplifier looks like this:

![Amplifier Diagram]

The point marked "-pulse in" is connected directly to the wire chamber wire. The 100k resistor is to provide line termination for the MWPC wire and for the chamber itself (this value is not well-known and is probably determined empirically—put a pulse on the wire, amplify it, and see where you get the least amount of reflection. It probably depends on the construction of the chamber, etc.

(By the way, the pair of diodes shown are for protection—-if there's an accidental voltage spike of absolute value > 0.7 volts (either sign) one of the two diodes becomes a good conductor and conducts the charge off to ground). The first stage of amplification is done by the 1035 shown, for now, all we need to know is that input impedance to ground is very high (in the ideal case it → ∞), so very little current will flow through it and most current will flow along the path in red through the 2K resistor. So, the wire (a capacitance!) (chamber) discharges through the 2KΩ. If the 2K were removed (an "open") the charge would pile up
on the WPC wire and the pulse would be very long; thus ruining timing information.

With the 2K, the pulse looks roughly like:

(for better picture and details, see references on chambers).

Here the timing information* is pretty good. (By the way, the far end of the 2K goes to "ref", -1.17 - 1.2V. This is the middle of the "MECL swing" (cf. later)). Why 2K? If it were shorted to grid or ref., etc., you'd be in trouble. Now, the amplitude of the voltage into the 1035 is R (for cont. i anyway, - V=IR-, so for R reasonable). We wish a "large enough" amplitude. But we also wish to preserve the shape--too big an R and the wire delays discharging and you get the delayed, broad pulse to the left above and lose timing information. So, someone settled on 2K as a compromise. So, we have some idea of what the input end of the amplifier card has to handle. What do we expect out of it? Well, its output has to travel quite a distance (three delay stripline) and go to a C.R., where its streaked in. Let's recall the discussion we gave on P2 at the beginning of these notes (repeated here for convenience).

*Done on the leading, or first downsloping edge -
What Does On, Roughly

Something comes down the beam pipe hitting the beam scintillators and something trips off the backward arm ("s" for slow) scintillators. This coincidence is called a B.S. When this happens, Roy's LeCroy generates a B.S. strobe (within nanoseconds). Since the s scintillators are closer to the Portakamp than the B scintillators, the s signal is delayed in a length of cable to arrive coincident with the B signal (see later section on "fast logic"). This strobe is sent to the backward arm C.R.'s and causes the wire chamber information to be latched up. (In order to make sure that you catch the wire signal at the C.R. input when the B.S. strobe comes, delay the signal from the wire amps by running the wire amp signals through a carefully calculated length of delay strip-line at the plug in end of the C.R. The width of a pulse that a wire amplifier puts out is about 50 ns. The width of Roy's fast logic pulse is about 10 ns, so the B.S. strobe has a little leeway in catching the wire amp. signal.

Meanwhile, something goes through the forward arm. It takes about 700 A. to make this trip one way - the LeCroy decision for a good event, "MASTER TRIGGER", comes about 1.5 - 2 A. after the B.S. strobe. The backward C.R. must hold the wire information for that amount of time.
The decision for a good event (MASTER TRIGGER) comes when:

i) the counters in the beam go off (scintillation ctrs, not wire chambers)

ii) a proton (determined by combination of forward scintillators and an anticoincidence in the Čerenkov counter) went through the forward arm

iii) a pion went through the backward arm

iv) the angle between the $\pi$ and the $p$, $\theta (P, T)$, was about right for an elastic scattering event.

Thus, a decision for a good event is made when there is rough momentum conservation. The "correct angle is measured something like this (for details, see "A Fast Logic Primer" in a few pages):

\[
\begin{array}{c}
1 \Box \\
2 \Box \\
3 \Box \\
4 \Box \\
5 \checkmark \\
\end{array}
\]

if, e.g., $S_4$ is hit, then, e.g., $F_5$ must be hit, etc.

(This strobe in signal is received on page 6 of C.R. schematic and fanned out to make signals $s, t, u, v, w, x ("u")$ for $\checkmark$ on C.R. schematic pages 1-4). This "u" strobe is a B.S. coincidence if the C.R. is a backward arm one, and is a MASTER TRIGGER if the C.R. is a forward. In any case, the point is that Roy's fast logic pulse is 10 ns wide. We want the output of the amplifier card to be wide enough to have the "leeway" mentioned. But if we make it too wide we get
into trouble also—many B. S. strobes (which could correspond to different

\[
\begin{align*}
\text{Amp out} & \quad \xrightarrow{50 \text{ usec}} \quad \text{C. R. strobe} \quad \xrightarrow{\text{O. K.}} \quad \text{Amp Out} \quad \xrightarrow{200 \text{ usec}} \quad \text{C. R. Strobes} \\
\text{Too Wide An AMP Pulse}
\end{align*}
\]

wires being hit) would catch this one long amp pulse, and you'd be fooled into continually latching that particular wire into the C. R. as a "hit" when it didn't really get hit. How often do B. S.'s come? Recall there's \(\sim 1\) incident pion per 50 buckets and buckets are \(\sim 20\) nsec apart.

\[\Rightarrow 1\ \text{pion per MS, but you could get 2 \(\uparrow\)'s within 20 usec of each other.}\]

So 50 usec for amp width is probably safe. So, we want the AMP OUT pulse width as thin as possible with "leeway" and we settle on 50 usec.

Also, we know that chambers are noisy enough as it is and this electronics picks up noise nicely, so we'd want the output stage a little isolated from the "more sensitive" parts or guts of the amplifier so noise doesn't work its way in backwards. Also, we've obviously got to drive this stuff through a sizeable length of delay cable, and both these points argue for a line driver at the C. R. end of the amplifier.

The Coincidence Registers (CRS)

We have already noted the need to latch up information from the backward arm wire chambers. To restate this a bit, we need a station to hold the information from the backward arm chambers (corresponding
to the B. S. strobe which latches that information in) about 2, use until either:

a) MASTER TRIGGER comes, in which case the C. R. crate must release the wire information and address to the computer on demand and then reset itself (be ready for the next event) when the computer is no longer busy analyzing this information (about 10-12 ms later. This type of reset is one variety of what is called a "PROMPT RESET", or

b) MASTER TRIGGER does not come, in which case the backward arm C. R.'s must be reset (made ready for new information. This is also a "PROMPT RESET"). In either case, resetting involves throwing away the bad information. (Since there's about 1 M. T. every .1 sec (for E290), without requiring M. T., over 99% of what goes on tape would be crap).

There is another reason to latch up information, and this applies to all the wires. This is that the computer is relatively slow in digesting information and the information has to sit and wait for the computer to take it. So, the C. R. (or "Coincidence Register" because it latches up information corresponding to various coincidences) must be at least a latch.

Now, an amplifier card puts out a MECL rectangular pulse, which has to travel a long distance and arrives at the C. R. card slurred.

(By the way, a single C. R. card has 32 channels, each channel corres-
ponding to a wire). A given channel on the C. R. card receives a succession of degraded, slurred pulses, along with noise fluctuations. The amplitude of a received pulse depends on how far it has to travel, and that depends on which chamber it comes from. So at the receiving end of the C. R. we have the following additional requirements:

I) Get rid of the noise. (The cable picks up an awful lot of it over the long distance from the chamber to the portacamp--its like an antenna).

II) Catch every input pulse without losing any.

III) Get the amplitudes back up to MECL values and square the pulse shape out.

P. S. (MECL valves are from -.8v to -1.6v)

We will now discuss the essentials of the C. R. briefly and then follow that by filling in some of the details. These essentials may be understood by referring to the following figure.

![Diagram](image-url)
A Simplified Channel of a C. R. Card

The positive and inverse output pulses of a wire's amplifier card are received here by the receiver on the left. (This receiver gets the amplitudes back up to MECL levels, squares the pulse shapes out, and cuts down on noise by receiving differentially). A "Strobe In" signal from the fast logic sends the positive pulse through the next and gate. (This Strobe In signal corresponds to a "B. S." coincidence* if this is a slow arm chamber channel, and to a "Master Trigger"* if this is a fast arm channel). The next receiver is made into an S-R latch by the use of positive feedback. When the positive pulse arrives there it is latched up until a reset pulse comes. When the computer wishes to read this particular C. R., it sends a strobe which allows the pulse to propagate through the next and gate, thus putting the pulse on-line to the encoder (which determines which wire had the hit and sends that information to the computer).

We mention now that the "Strobe In" or "Coincidence Gate" signal is sent in from the Fast Logic on a white wire which comes in the top of the C. R. (as it faces you when it's getting plugged into the crate) on a "twinax connector" (see page 6 of C. R. schematic). This twinax input may not be left open.

Construction and Some Nomenclature Repeated

Refer to the pictures shown on the next few pages. (Recall that) there are two sets of pins on the back of a C. R. (The "back" is where

*See earlier.
the C. R. S. plug into the crate. The twinax input for "Strobe In" is on the front. These two sets of pins are called J1 and J2. A C. R. card has two sides, left and right, when looked at from the front. The suffix R (e.g. "J1R" or "J2R") for a set of pins means "right". No suffix means "left". So, e.g., "pin J1-9R" means the ninth pin from the top on the right side of J1. This system of nomenclature for the back sets of pins is used in the Encoder and Mixer also.

The front of the C. R. also has a connector. Signals from the wire chambers come in here on plug-in paddles which are connected to the ends of the striplines. The twinax input at the front top has already been mentioned.

We now fill some of the details into our previous discussion. (If the reader is not interested in the details, he may safely skip to the sections on "Prompt and Group Reset" and "Carry"). Below we show a section of the C. R. schematic corresponding to part of one channel. This circuit is repeated once for each channel, the 32 channels taking up PP1-4 of the C. R. schematic.
The front connector of the C.R. and the type of paddle at the end of a stripline which plugs into it.
The use of differential receiving eliminates noise common to both lines, as a differential amplifier samples the difference between the two inputs. The input called "TRUE" ("Pr" on this channel) takes the positive signal from the amplifier, and the input called "BAP" ("35" on this channel) takes the inverse signal. Pulses less than \(\sim 100\) millivolts amplitude are eliminated by causing the negative input of the first receiver/amplifier to sit \(\sim 100\) millivolts above the positive input in the quiescent state. \(Th^+\) and \(Th^-\) are voltages input from elsewhere that are set \(\sim 100\) m.v. apart to accomplish this. (These thresholds are set by potentiometers in C.R. companion A and are adjustable.) The threshold \(\uparrow\) is usually set at \(\sim -2.1\) volts. Threshold \(\downarrow\) (the more positive of the thresholds) is set at \(\sim -2.0\) volts. The resistor is for termination of the stripline, whose characteristic impedance is about that.

The inputs are fed to the receiver through a capacitor which provides D.C. isolation (otherwise an amplifier output and a threshold would be forcing the two ends of a line to two different voltages, causing a lot of current to flow). Assuming that the amplifier has an infinite input impedance, the input signals see simple series R.C. circuits.
To see how things work, consider first the quiescent state.

In the quiescent state, $P_0$ is at MECL low ($\sim -1.6$ volts) and there's 0.5 volt sitting across the capacitor. The positive input of the receiver sits at threshold up (say -21 volts) and the negative input sits at threshold down (-2.0 volts). Now suppose a hit signal comes along. This will be a degraded, slurred MECL pulse. For the sake of discussion, let's say it swings from -1.4 volts to -10 volts. The rise time of a pulse arriving from the amplifier is probably 20 nano-seconds. We want our R.C. circuit to pass this, so choose $RC > 20$ ns. At the high end for choice of a C, we are limited by what fits on the board easily. So $C = 0.1 \mu f$ seemed a good choice and $RC \sim 1 \mu sec$. So our pulse passes through and low frequency noise (e.g. 60 c.p.s. pickup) is cut out. (The positive input pin to the receiver then swings, in our example, from -2.1 to -1.7 to 21V, and the negative input from -2.0 to -2.4 to -2.0).

The receiver output swings high once the positive input exceeds the negative input.

The input strobe sends the high level through to the latch, whose output is wire or'd to its input. (In MECL wire or'ing is allowed, and the high voltage always wins). If the input (pin 10) goes high it causes the output (pin 7) to go high. If the input then goes low, the feedback keeps the output high.
The reset pin (pin 9) is kept at \(-1.2\) volts (the midpoint of the MECL range) in the quiescent state. To reset the latch, supply a full positive MECL pulse to pin 9 through input BD. Then pin 9 goes higher than pin 10 and the latch is reset even if the input to pin 10 is still high.

Prompt and Group Reset

We have already mentioned that PROMPT reset comes if the trailing edge of Computer Busy or the lack of Master Trigger comes. In either case, PROMPT reset is to reset many C. R. cards at once (all C. R.'s or all backward arm C. R.'s, respectively). The prompt reset signals are first generated by the Fast Logic in "N. I. M."

(N. I. M. is a digital family that has a low level of -0.7 and a high level of 0.0 volts. N. I. M. stands for Nuclear Instruments Methods, and those levels have been its standards for years). Since the C. R.'s use MECL these levels are passed through a NIM to MECL converter*. They then go to a piece of the C. R. companion cards (which is just a 1 \(\rightarrow\) 4 fan-out device for this particular purpose--recall that there is one companion for every 4 C. R.'s) and then go to the C. R.'s.

We've also mentioned that an encoder "heads" a C. R. (generates the full 13 bit addresses of all hit wires in the C. R.). When the encoder has the data from a single C. R. latched in, it resets that one C. R. This

*basically just a diode since NIM = MECL + 0.7 volts D. C. = MECL + diode drop.
resetting of an individual C. R. by the encoder is called a "GROUP RESET". The word "group" is used because it's resetting 32 wires at once.

**Why a Group Reset, Isn't Prompt Reset Enough?**

Yes it would be, if the system were designed in a different way. However, we'll soon see that in this system, a group reset is needed otherwise a given C. R. card will be read and re-read out forever. See "WHY A GROUP RESET, ISN'T PROMPT RESET ENOUGH, REVISITED, a few pages hence.

An OR of all 32 bits on a C. R. is generated by 8 four input OR gates. The output of this OR (a in the figure) is called the hit bus and is high if there has been a hit on the C. R.
So far, except for the Group Reset signal, we have a picture of a one way conversation that goes from the C.R. to the encoder. (Group Reset is sort of the answer). The conversations of the C.R.'s with the encoder are facilitated by a level called "Carry"— the and of Carry and Hit Bus is required for the C.R. to be allowed to put its data on-line for the encoder. This is because this system's encoder can only read one C.R. at a time, so the C.R.'s are read sequentially and then Group Reset sequentially. This is accomplished by the Carry level's being passed sequentially from one C.R. to the next, like school kids passing an attendance sheet around the room.

So, when a given C.R. has had a hit, his Hit Bus goes up. If he's got carry he releases his data to the encoder and the encoder then Group Resets his latches causing his Hit Bus to shut up. If the C.R. doesn't have Carry yet, he keeps his Hit Bus high. If Hit Bus is low (either from being Group Reset or from not having any data in the first place) and the C.R. has Carry, well, its of no use to him, so he just passes Carry on the next C.R. A slight simplified schematic of this part of the C.R. is shown on the top of the next page.
BD = name of reset input on schematic of CR - PP 1-4.

"i" = name of strobe signal on schematic of CR - PP 1-4.

Maybe this is easiest to remember with Ho-Shang's charming analogy (teacher ← encoder, C.R. → kid, Carry → Walkie-Talkie):

Imagine a classroom with 16 kids all sitting in a row. The teacher (encoder) has a Walkie-Talkie (Carry) to the first kid. When a kid has the Walkie-Talkie he has the right to ask questions. So, the first kid asks
his question. (Some other kids have their hands up, but they have to wait until they get the walkie-talkie). As soon as the teacher has heard the question, he tells only that kid to lower his hand (Group Reset) and pass the walkie-talkie to the next kid (passing Carry).

**WHY A GROUP RESET, ISN'T PROMPT RESET ENOUGH?**

One might ask -why not wait until the computer isn't busy and reset everybody then instead of Group Resetting C. R.'s now while it's dead time anyway? Answer--In this system, the end of Carry In and Hit Bus Off is needed to pass Carry (Out) to the next C. R. (See figure on last page). If a C. R. has had hits and doesn't get group reset, he will keep carry and will be read over and over again. This has been a problem with this system!

**Some Detail on Propagation of Carry**

Again, we now supply some detail which the reader may safely skip if he likes. On the next page is a copy of page 6 of the C. R. schematic. Toward the left we see how Carry is passed to the next C. R.:

If there have been no hits, Hit Bus comes in low, is inverted at gate S, and Hit Bus is anded with Carry In to be sent to next C. R. as Carry.

If there's been a hit, the high on Hit Bus gets latched (at gate (T)). The Reset pin of the latch (pin 4 of gate (T)) is held at -1.2 volts in the quiescent state by the voltage divider shown. Hit bus is anded with Carry at gates K to make "Ready" which is the signal that gates the data
to the encoder. Ready also combines with the Group Reset signal from the encoder (bottom of the schematic page) to reset. The data latches in the C.R. When this happens, Hit Bus comes back here low. The combination of Hit Bus low and Group Reset come and gone (i.e., now low) resets the Hit Bus latch (drives pin 4 of gate T. 8 volts above its quiescent value of -1.2 volts). This sends Carry Out to the next C.R. Note: Group Reset is generated by the encoder in each crate and is bussed in parallel to all the C.R.'s in the crate. (We've just seen that the circuitry permits the Group Reset from the encoder to reset only the C.R. that has Ready (Carry • Hit Bus)).

Address Generation

We recall that the purpose of the system is to pass addresses of hit wires to the computer. We also recall that there are 16 C.R.'s on a crate. One piece of the wire address is the address of its C.R. within a crate.

If you hold a C.R. in your hand and take a look at it (see drawing, next page, due to Cavanaugh) you'll notice that there are eight little switches in a DIP package. This is indicated on page 5 of the C.R. schematic. Switches 5, 6, 7 and 8 are for the C.R. address within a crate. (Switch 5 is the most significant bit, switch 8 is the least significant bit). Switches 1, 2, 3, 4 were intended to specify the wire plane boundary on the Cluster mode, and hence are not used.
FIGURE: A CR WITH THE ADDRESS SWITCH SHOWN

SWITCH

5 = HIGHEST ORDER BIT OF ADDRESS

6 = NEXT "" "" "" ""

7 = NEXT

8 = LOWEST ORDER BIT OF ADDRESS
intended to specify the microplane boundary in the cluster mode, and hence are not needed.

An example of setting the C.R. address, drawn by Cavanaugh, is shown.

**EXAMPLE:** ADDRESS ZERO, (0).

![Diagram of address zero](image)

**EXAMPLE:** ADDRESS FIVE, (5).

![Diagram of address five](image)

So there are 36 bus lines from the C.R.'s to the encoder - 32 data lines and 4 address lines.
The C. R. Companions - Some General Stuff

(An understanding of the C. R. companions is not necessary for the discussions of the Encoder and Mixer to follow, so the reader may safely skip to the sections on the Encoder if he wishes. However, the companions are of great importance in day to day working with the system, and a knowledge of them is important for troubleshooting).

Recall that there are two types of C. R. Companion cards - type "A" and type "B". There is one type B companion per crate and it goes in the left-most slot on the crate. There are three type A companions per C. R. crate and they go in slots 6, 11, 16. (Slot 1 is the left-most slot). The slots for type A companion have the bottom connector missing in the crate and this is an easy way to find the slot where a type A companion should go.

Type B has everthing type A has, plus some additional stuff to be discussed later. For now, we simply remark that (contrary to rumor!) types A and B may not be used interchangeably. (In reference
to the schematics, type B card is the UNION of type B schematic and type A schematic).

A drawing of the fronts of the two types of companions is given on the next page. Note the differences in the colors.

C. R. Companions have one twinax input connector (for Prompt Reset), two "test points", and two holes for potentiometer adjustment. The potentiometers are used for setting the threshold for C. R. inputs. By varying these potentiometer settings the C. R. can be allowed to accept different magnitude input pulses.

The bottom test point corresponds to the bottom potentiometer (in zerox) and the top test point to the top potentiometer. The bottom potentiometer sets the reference voltage for the threshold (midpoint of the range between threshold up and threshold down settings). More about this later.

Type B companion is recognizable by the fact that the test points are red and violet, while type A has yellow and green test points.

The "reset" twinax input connector gets a cable from N. I. M. - MECL converters for Prompt Reset which tells the C. R. 's to ignore the event and reset the latches. But the companions have another, more interesting function, and this we shall discuss next.
Subject: CR Companions

- Red: ○ → ○ ← PROMPT Reset (from Fast Logic)
  - TWINAX INPUT
  - CR Threshold Test Point
- Violet: ○ ← CR Threshold Pot for Adjustment
  - CR Threshold Center Voltage Test Point
- Yellow: ○ ← Green
  - CR Threshold Center Voltage Pot for Adjustment

Single Width CAMAC-like Module

CRC "B"

CRC "A"
A Parallel Path for Carry

We've been talking alot about Carry lately, and about how its passed from one C.R. to the next, sequentially. Consider the following problem: Suppose a C.R. is pulled from the crate. Then the rest of the C.R.'s will be messed up. (Or, with reference to P6 of the C.R. schematics, if a C.R. is pulled out, pin J1-9 (carry in entering) is not connected via any circuit to pin J19R (carry out leaving for next C.R.).

Sippach, et al.* have invented a cute way of getting out of this problem. Basically, he lets Carry propagate through the Companions of the C.R.'s at the same time as it propagates through the C.R.'s. If a given C.R. is missing or broken in such a way as to 'hang up Carry, the associated Companion card simply hands Carry to the next C.R. (Each companion handles 4 C.R.'s).

To understand how this happens, see the diagram on the next page and the left side of the companion schematic on the page following. Consider the sequence of and gates shown on the right of the next page. These are supposed to correspond to and gates "J" in diagram on bottom of page CR-4. Only the CARRY input is shown. One C.R.'s carry in becomes the next C.R.'s carry out, etc. There is a one correspondence between the and gates on the C.R.'s and and gates on the companion. The corresponding and gates on the companion are shown to the left. The OPEN signal originates at the ENCODER and is fed to the beginning of both chains.

The output of each C. R. gate is simply connected to the output of each corresponding companion gate. Inputs a to the companion card gate test for the presence of the associated C. R. card gate in a manner that we shall describe shortly; they are logical 0 (-1.6V) if the C. R. is present, and logical 1 (-0.8V) if C. R. is absent. The presence of a HIGH on the previous output of a C. R. gate forces one input of the associated ("previous") companion gate HIGH. If the output of the C. R. gate is not high, then the other input to this comp. gate is high, the companion gate's output is high, forcing the input to the next C. R. gate high. E. G., if, as in diagram of last page, C. R. card #3 is pulled from the rack and output of C. R. #2 = HIGH — both inputs of companion gate 3 are HIGH — comp. gate 3 output HIGH — dangling end forced HIGH input to C. R. gate 4 forced HIGH.

Now: how does companion gate sample for presence of C. R. gate?

We wish to have a MECL low -1.7V at point X (see diagram, last page) if the C. R. isn't there and 0.8V if it is. If the C. R. is present, putting 1.6V at 9 won't do much except draw a lot of current. Since 

\[ 2 = \frac{V}{R} \]

put in some resistors to provide voltage drop, i.e., fig. to left. With two equal resistors, if C. R. is present, X winds up at midpoint of 0.8 and 24V - 1.6.
The way this is done is the following: First of all we need -2.4V and -0.8 V. From the schematic sheet labeled FNAL MWPC READOUT TYPE A COINCIDENCE REGISTER COMPANION we generate this

Since one companion handles 4 C.R.'s, we bus the -2.4V to each of the 4 C.R.'s: The output of pin J1-16 on companion is sent on a bus line to pin J1-16 of each C.R. On the C.R., pin J1-16 is physically jumped to pin J1-15R and then is sent back to the companion (received at PIN 2). If the C.R. card is missing from the crate, there is no connecting between J116 and J115R in the C.R. card slot and then (J115R, J1-11, J1-5R, J1-4R) on comp. sees an open

(The J1-16 - J1-16 connection is made only if CR is in A or B crate.)
Threshold $T^H$ & $T^V$ Generation for CRs.

Cavanagh has written an explanation of this.

Student's Name

Subject: Threshold Generation

Instructor's Name: R.C.

Date: 9 Feb, 77

Each CR companion has the shown circuit used to generate $T^H$ and $T^V$ for the 4 CRs associated.

A center voltage is determined by the GREEN Point. Pin 2 of Op amp cannot be far from pin 3 and pin 2 draws essentially no current. Thus the same current flows through each 1K resistor tied to pin 2. Hence $T^H$ is equal below pin 2 as $T^V$ is above pin 2.
The yellow test point adjusts $ThV$ as a consequence of the above argument $ThV$ fol.

This circuit was originally designed by Stan Bristol.

In the DISCRIMINATOR THRESHOLD GENERATOR CARD for the amplifiers 308A op amps are used in stead of 741's. With 741's $ThV$ won't go above $-1.0$ volt, whereas with 308A, $ThV$ will go up to about $-.4$ volt.
Differences Between Companions A and B

In addition to the functions provided by companion A, companion B provides emitter termination and current balancing for the data output gates of the C.R. These concepts will be discussed in appendixes.

Additional Remarks on C.R.'s and Companions

Stripline conventions and pin-channel identification on the C.R.'s will be covered in an appendix.

Ho Shang Vaid has written a brief procedure for testing C.R. Companions; this will also be in an appendix.

We next need to discuss how the system generates the wire address within a C.R., how the cycle of reading C.R.'s is started, ended, etc. These are some of the jobs of the encoder.

Appendix - MECL Termination and Current Balancing

In MECL, the output line of a gate is the emitter of the output transistor. Bases are gate inputs. (In TTL, collectors are gate outputs).

While in some MECL series external resistors must be connected to the bases, for the MECL 10000 series (the series used in this system) gates do not require external resistors for their inputs, but do require external resistors from their emitter outputs to a power line called $V_{TT}$ (~2.5 volts) to complete a current path to the collector. The MECL handbook recommends about 200-$\Omega$. 

\[ V_{TT} \text{ power supply} \]

\[ \text{Current from gate to emitter} \]
EMITTER TERMINATION OF C.R. GATES AND COMPANION B

Refer, e.g., to page 1 of C.R. schematic. The first receiver (gate A2) obviously has both its outputs terminated. \( V_{TT} \) comes from companion. The next gate is the 10197 (AQ). Its emitter resistor is after the latch (AG), i.e., \( R \), to left takes care of both pin 7 for the latch and pin 3 for the 10197.

the last gates (gates H, etc.) have no obvious termination. The emitter termination for these is done on C.R. companion B as shown to the left. Why not just put this termination on the C.R. itself? Recall that the outputs of all C.R. channel 1's are bussed together, and similarly for channels 2, 3, ..., 32. (These bus lines are visible in the rack of the C.R. crate if you pull the C.R. out). (Recall that in MECL, wire or'ing is allowed and "the high guy wins".) The bus lines lead to the crate encoder. Hence only one C.R. has carry, the output and gates of all the other C.R.'s are sitting low, so a given channel bus line has at most one high gate connected to him.

Since it's easier to put one resistor on a companion than sixteen on C.R.'s, we do that.
CURRENT BALANCING AND C. R. COMPANION B

The receivers shown on companion B schematic are for current balancing. To define current balancing, consider the following figure of a MECL gate in a logical "1" state:

Imagine 1000 gates all hooked up to $V_{TT}$ and $V_{EE}$. Imagine they all change states at once (!) You're then demanding a large amplitude current spike from the power supplies. Although it may not be healthy for them, the supplies will supply it. However,

i) the wires have nonzero inductance. With big current spikes, you generate enough "back E. M. F." to slur pulses.

ii) A large amplitude current spike will radiate onto neighboring lines. This can be bad. If, e.g., a clock line is nearby it can introduce spurious clock pulses into the system, which could really do a system in. (Note that the crate to crate buses are very long and pass near a lot of stuff).

To get out of large amplitude current changes, use gates with true and inverted outputs and connect both to $V_{TT}$. When one output changes state,
so does its inverse, and current requirements from the power supplies remain about constant. It would be nice to current balance every gate we use. Unfortunately, this greatly increases the amount of wiring. So we just current balance outputs whose signals go on long lines, because these lines tend to go near other cards, etc. Any line that travels more than 8 or 10 inches should be current balanced. So, we current balance buses, etc.

What should we current balance on the C.R.'s? The outputs of the "last and gates" have to travel to the crate encoder, which means distances up to the length of a crate. A crate is about a foot and a half long. So we should current balance those outputs.

Instead of generating the inverse signal on the same gate in this case we make the inverse signal on another gate. This is where COMPANION B comes in again. Here the output of the receiver (10115) on companion B is to be the inverse of the C.R. outputs. That way, most of the length of the line carries constant current.
Purpose of the Encoder

Now that we've seen how the information from the wire chambers is stored, we need to discuss how it's passed on to the computer when the computer wants it. The job of picking out and "encoding" (converting to binary number form) the right information to send, and sending it at the right time, belongs to the encoder.

Since the computer has only 16 data lines, we obviously can't give it 5000 bits for 5000 wires at once. To cut down on how much we have to send, we pass on only the addresses of hit wires. Some addresses up to 5000 involve about 13 bits, we've got to send the addresses serially. For reasons of obtaining maximum speed in the hardware, the system works roughly in the way we've mentioned: the encoder considers a group of eight wires at a time, scans it for hits, sends the addresses on upon finding any, then moves to the next group of eight wires, etc. When an entire C.R. has had its four groups read the encoder Group Resets that C.R. and moves on to the next one.

To facilitate this reading, the crate encoder takes the carry signal that he's received from the computer* (or from the previous crate's encoder+) and gives it to the first C.R. in his crate. Carry is

*if we wish to read only this one crate ("Single Encoder Mode"—see later) or if we are reading all the C.R. crates sequentially ("Full System Readout Mode") and this is crate zero.

+if we are reading all the crates sequentially (Full System Readout Mode") and this is not crate zero.
passed from C. R. to C. R. as each is read. When the crate is finished being read, the crate's encoder takes Carry back from the last C. R. in the crate and sends it back to the computer** or to the next crate's encoder++. In this system, the encoder gives the addresses to the mixer (which sends it on to the computer) in several pieces:

1. There is the crate, or encoder address, which is set by DIP switches on the encoder and sent off to the mixer as 4 bits,

2. The C. R. address (which C. R.), which (as we've seen) is set by switches on the C. R., sent on bus lines to the encoder while that C. R. is being read, and then passed to the mixer when the encoder finds a hit, 4 bits.

3. The group address—a binary number from 0-3 indicating in which of the C. R.'s four groups of wires the hit was found,

4. The wire address within a group. This is a number from 0-7. The system adds 1 and sends the mixer a 4-bit binary number from 1-8.

All four pieces of the address leave the encoder on page 4 of the encoder schematic.

**If we wish to read only this one crate ("S. E. M.") or if we are reading all the C. R. crates sequentially and this is the last crate (crate 9).

++If we are reading all the crates sequentially and this is not the last crate (crate 9).
Physical Construction

An encoder consists of two boards, "A" and "B", which are connected together by some ribbon cable to form a single module. *(Trouble-shooting results showed that several samples of this cable lying around our portakamp were defective, and indeed, recently I discovered that the reason that one of the spare encoders was not working was simply that several individual wires on the ribbon cable were shorted to each other).*

Modes of the System

Originally the system was designed to include several different options: the various "Cluster Modes" and the "All Hits Mode". In the All Hits Mode, the encoder passes on the address of every hit wire. In a Cluster Mode, if a cluster of neighboring wires were to be hit, to save time the encoder would send on the addresses of the first and last wires in the cluster only.

The system has never been made to work in any of the cluster modes. We shall not include any discussion of the cluster modes in this writeup except to note now that since the cluster electronics has been built onto the boards and provides some input signals in the quiescent state that the All Hits mode needs to work, a failure of some cluster mode electronics can cause the system to fail on the All Hits mode. Cluster mode electronics is included on the system schematics.

*Whether a given page of encoder schematic refers to board A or board B is indicated on the lower right of that schematic page.*
We shall always assume that we are working in the All Hits mode. As the reader has probably gathered from recent footnotes, the All Hits mode has two sub-modes—the "Full System Readout Mode" and the "Single Encoder" or "Partial System Readout Mode."

The Single Encoder mode was the mode used during the successful data taking runs of E290. In this mode the computer specifies that it wants to read a particular encoder's crate by specifying the crate address and sending Carry and some other signals.

The Full System Readout Mode was first made to work at the end of the summer of 1978 by Burt Pifer and Ho-Shang Vaid. In this mode, all the C. R. crates are read sequentially with the proper set of commands from the computer.
A Bit About the Computer Readout Cycle so That We Can Understand
the Readout Request Cycle.

At the beginning of a readout cycle the encoder is off. Data is
being latched into some of the C. R. 's. The computer, meanwhile, is
merrily working on some program its been given--perhaps making
histograms out of old data, maybe beginning or ending a run-those
sorts of things. Then, perhaps Master Trigger comes, indicating that
we have an event. At this instant (call it \( t_A \)) a computer "Interrupt" is
generated which tells the computer to stop what it was doing and start
working on a readout cycle for this new event. The computer then tells
the branch driver (the "BD011"--one of its external memory locations)
to write some words into the mixer. These words are the signals that
the mixer needs to make an encoder work, the address of the crate to
be read, etc.

The mixer then turns the encoder on, and the encoder quickly
sends two words to the mixer, filling it. Say this occurs at \( t_B \). Mean-
while, upon finishing the write instruction, the computer tells the BD011
that it wants to read some words from the mixer (i.e., the data from the
PWC crates. This part of the computer cycle is a "DHA"). It takes the
computer a while \(( \sim 100 \text{ us})\) to put together this read command (it must
involve about 20 separate instructions, each \( \sim 5 \text{ us} \) long). Once this is
done, say at \( t_C \), the computer starts reading words out of the mixer
at the rate of about one per microsecond.
So, if you were to take an oscilloscope and look at Readout Request, triggering, say, on the Start pulse (see later) that starts the encoder working, you'd see something like this:

\[ \text{READOUT REQUEST} \]

\[ \text{MIXER BUSY} \]

\[ \text{~100ns} \]

\[ \text{~10ns} \]

\[ \text{~10ns} \]

\[ \text{~10ns} \]

\[ \text{~10ns} \]

\[ \text{~10ns} \]

\[ \text{(to 60mA clock pulses plus some propagation delay)} \]

\[ \text{READ REQUEST} \]

\[ \text{MIXER FULL} \]

\[ \text{REQUEST REQUEST WILL ACKNOWLEDGE} \]

\[ \text{OUT OF SCALE DURING TO CORRECT REQUEST REQUEST WILL ACKNOWLEDGE} \]
We also see from the schematic that pin 13 of gate N (the pin that latches flip-flop M2) is simply (R. R. A.) \cdot (R. R.). So, if the flip-flop M2 was latched because the mixer was full, the eventual shutting off of Readout Request by a returned low pulse on Accept will clear the input of M2, thus allowing the next word to get sent on.

We also see that the generation of a Readout Request generates a \( jj^* \), which allows the control bit for a new data word to enter the main control loop as can be seen on the next page. So one loop is finished.

Suppose, however, that an "invalid hit" was found. Returning to the left-hand side, note that \( \bar{W} = \bar{X} \) is high in this case. \( \bar{W} \) combines with Q2 of (AA) on page 3 to push pin 3 of gate AD high. Now, there are two possibilities: either the present group number is 3 (in which case \( \bar{\mu} \) from the group counter is high) or it is not (in which case \( \bar{T} \bar{T} \) from the group counter is high). Referring to the next page, we see that in the second case, flip flop (AF) goes high, which, through \( S S \) adds one to the group counter and resets the accumulator. Simultaneously with this, pin 4 of gate AM goes up, and the main control loop is entered again. So everything begins as before, except with the next group.

If the group address was already 3, we have to change C. R.'s. In that case we can see from P3 of the schematic that pin 9 of (AM) (extreme left side of page) goes up, mimicking a "Start" pulse. Since the

\*Actually, \( jj \) is usually high when Readout Request is. Pins 9, 10, 11 of gate (S) are high in the All Hits Mode. Pin 12 we'll talk about soon.
old C. R. has already been group reset, Ready is at the new C. R., and the new C. R.'s data is loaded in on a "Parallel Load Latches" pulse. Then everything goes on as before.

**Scanning Vs. Reading**

The flip-flops on schematic page 3 are called the **scanning flip-flops**. The flip-flops on page 5 are called the **readout flip-flops**. The readout flip-flops become full if the mixer is full and then they have to cause the encoder to hold data until the mixer has space.

In that case, it would be nice (and save time) if we could have the system continue to plow through data zeros and eliminate them and locate that will have to be read once the readout flip-flops are ready. This function is called **scanning**. From our previous discussion, we see that the smaller control loops contained totally on P3 (these are called the scanning loops) continue to work even if the readout latches are full. So, e.g., we can continue scanning C. R. \#n+1 while the reading is tied up at C. R. \#n. We will see later that in the Full System Readout Mode, we can continue a scan onto the next C. R. crate if the readout is held up in this crate.

**Losing A Word In The Traffic Jam, Revisited**

We now return to the question we hinted at before regarding the "traffic backup" figure—what's to stop us from clocking the accumulator at the wrong time and "losing word 4"?
To answer this, notice where pin 12 of gate (S) (page 5, encoder schematic) comes from—it's the OR of \( Q, \overline{i}, \overline{d}, \) and \( Q_2 \) of flip-flop M2. (Recall that \( Q \) and \( \overline{i} \) both come from page 3 of the schematic and are separated by a clock pulse, then, a clock pulse after \( \overline{i}, \overline{d} \) comes on, then, a clock pulse after that, M2 comes on). So pin 12 has the power to keep \( \overline{j} \) off for at least 4 consecutive clock pulses (whether Readout Request is on or not). If \( \overline{j} \) is off, "word 5" can never enter the control loop.

To be more specific, suppose we've put words 1 and 2 in the mixer.

Consider the 3rd word. It gets clocked into the accumulator by \( \overline{d} \).

Now, \((R, R, A \cdot RR)\) is true, \(hh\) is made and word 3 goes into data flip-flop (AH). Readout Request for word 3 comes high and stays high for a long time. This pushes \( \overline{j} \) up and the 4th word goes into the accumulator. \( \overline{j} \) then stays low until flip-flop M2 is freed = until R.R. goes down = until the mixer takes word 3 and everybody moves along one notch.
Making a Start Pulse in the Single Encoder Mode

We now discuss how the encoder generates the "Start" pulse for the main control loops shown earlier. To make this pulse in this mode, the encoder must receive three things from the mixer: a signal called Single Encoder Flag, a signal called Readout Flag, and the correct encoder address. Looking at page 2 of the encoder schematic (next page), we see that these signals come in on the bottom of the page. It's easy to trace how they are anded to send Carry out to the first C.R. in the crate. Now, pin J1-11 (middle of the page, comes in from encoder companion) sits high in this mode, and this pulls \( b \) and \( LLL \) low. \( b \) is connected to the reset pins of the scanning flip-flops, and \( LLL \) is connected to the reset pins of the readout flip-flops. The low level releases the (quiescently asserted) reset pins and allows the flip-flops to work. Looking at pins 10 and 11 of gate O, we see how pp is part of the combination needed to generate a Start. PP comes from page 5 of the schematic and is just the OR of all the O's of the scanning and readout flip-flops. We notice that we also need a low on J1-10R. J1-10R is Carry Out coming back from the last C.R. in this crate. If it's low there's still unread data in this crate. So we make a Start in this mode if the computer asks us to and the encoder's control loop isn't already tied up and there's unread data in the crate.

When Carry Out comes back high from the last C.R. a "Partial Readout Finished" signal is sent off to the mixer.
NOTES: All pins labelled "4..." or "5..." don't exist. Only J1, J2, J1R, J2R pins exist.

Carry out of last CR
This Crate

EXAMPLE: 6 (in ground loop)
1 3 5 6 7 8
...and then side
(0) 5 3 0 1 0
...next on the side.
Note that in this mode it's possible to get a Start pulse at the end of the readout -- if pp comes back high before Carry Out (J1-10R) does. This potentially can mess you up (although apparently no disasters have occurred yet on account of this) and one of these days we should build some propagation delay into pp or something.

In the Full System Readout Mode making a Start is a bit different and we'll talk about that later.

We should point out that Start is one encoder clock pulse (~60 ns) long. (This is because Start starts the control loops and one clock pulse later the first flip-flop in the loop is filled, so pp goes down). (So, for example, if you want to look at a Readout Request cycle on an oscilloscope simultaneously with Start, note the difference in periods!).

The "Start" Address DIP Switches--Partial Readout Mode

The mixer sends the address of the particular encoder that it wants to "start" on a bus line to all the encoders. Somehow, a given encoder has to know that he's the fellow being summoned and the other encoders have to know enough to sit still. If you look at an encoder, you'll notice that there are two sets of DIP switches. One set we've already spoken about--its on page 4 of the schematic and is used to send the encoder address to the mixer. The other set we use to receive an address to make a start.
At the mixer the intended encoder address is generated differentially. The four pairs of bits are put on bus lines and are received here on page 2. For each pair, the wire-or of two switches is used—one left open and the other closed. You could do this by making the wire-or of each pair high if the address is correct and then test for that condition with an and gate.

For example, the address shown is 0011 = 8. Then if the correct address were to come from the mixer, all ones would be put at the input of the and and it would pass. If the wrong address were to come some zeros would be put there.

In this system we actually test for all zeros instead of all ones.
Starts and Finishes in the Full System Readout Mode--The Encoder and its Companion

We begin by warning the reader that some of the designer's comments on the schematics here are misleading.

Consider that we are in the Full Mode and about to start reading crate zero. The levels Readout Flag, Full System Readout Flag, and Carry In (all from the mixer for crate zero, Carry In from previous encoder companion otherwise) are anded on the bottom of page 2 of the encoder schematic to send Carry In to the first C. R. and simultaneously to the encoder companion (off J1-8 here) for this crate. "Carry In Scan" comes in from the mixer, and with R. F. and F. S. R. F., puts a low out on \( Q \) releasing the reset pins on crate 0's scanning flip-flops. For crate 0, "Carry In Readout" is the same signal as Carry In Scan, and this with R. F. and F. S. R. F. puts a low on \( LLL \) releasing the readout flip-flops for the duration of the readout. Simultaneously, "Carry In Scan Bar" (lower left on page 2), which is an output (J2-14R 2 here) of this encoder's companion (ignore Cavanaugh's remark "to next encoder" here) is sitting high and allows the creation of a Start pulse which lasts for one 60ns clock period: \( \text{Start} = \overline{PP} \cdot \text{Carry In Scan Bar} \cdot \text{Carry In Scan} \) (all 3 are levels). At this point, the readout loops in crate 0's encoder start working.
Assume that the scanning (but not the reading) of crate zero is finished now. Carry Out coming back from the last C.R. goes to two places--J1-10R on page 2 of the crate 0 encoder, and J1-9R of the crate 0 encoder companion. (The companion schematic is included here as the next page). Now, J1-8 on the companion has been sitting high because that's where Carry In from the encoder came to at the same time that it was on its way to the first C.R. So, J2-4 on companion 0 goes high and this becomes Carry In for the crate 1 encoder.

J1-8R on companion 0 has been sitting high because its connected to J1-8R on encoder 0. So J2-5 on companion 0 goes high, and this is Carry In Scan for crate 1's encoder. (Similarly, J2-5R goes low, and that's C.I.S. Bar for the crate 1 encoder, which is received there but not used!). Also, J2-14R on companion 0 now goes low, and this goes to J2-14R on this encoder (I suggest you ignore Cavanaugh's phrase there "... to next encoder"). This signal is Carry In Scan Bar, and as you can see from the (crate 0) encoder schematic (p. 2), its purpose is to inhibit the type of multiple Start pulse problem that is potentially present in the single encoder mode.

Now things start happening in crate 1. In particular, its set up to start scanning, but not to start reading (since we haven't finished reading crate 0 yet). To see this, note the following: we remarked already that Carry In has gone to the crate 1 encoder. Readout Flag and Full System Readout Flag are bussed to all the encoders at once on the
crate-to-crate bus, so the crate 1 encoder can send Carry to his companion and to his first C.R. Encoder 1 has also gotten Carry In Scan, so he's already put a low on $\overline{p}$, releasing his scanning flip-flops. Since $pp'$ is high for crate 1, Carry In Scan and Carry In Scan Bar* (which has been sitting high all this time for crate 1, since it's the inverse output of a gate that hasn't been activated yet) both being high create a Start for crate 1. So, we start scanning, but we don't start reading crate 1 (because LLL hasn't been pulled low yet to free the readout flip-flops).

Now back to crate 0, which we were still reading. Assume we've just finished reading it. We use $pp'$ as the only indicator for this. So $pp'$ for crate 0 goes high+. Anded with Carry In Readout, Readout Flag, and Full System Readout Flag, this raises encoder pin J1-10 (middle of P2) which is connected to pin J1-10 on the crate 0 companion. So J2-6 on the crate 0 companion goes high. This signal is Carry In Readout and it goes to the crate 1 encoder and allows crate 1 to begin its readout. Meanwhile, J2-6R on companion 0 goes low. This signal goes two places

*Note that there are, unfortunately, two signals that Cavanaugh calls by this name. Here we mean the one that comes from companion pin J2-14R to J2-14R on page 2 of the encoder schematic (lower right).

+This is because the scanning and reading loops have run out of data and the encoder clock is still ticking, so zeros get clocked into the flip-flops (no "reset" was generated).
(N. B. because Cavanaughs "... to next encoder" comment on the
schematic is misleading) -- to the next encoder (enters there on J2-12,
where its received and not used anyway!) and to J1-11 on this (crate 0)
encoder as "Carry In Readout Bar" (i.e., pushes J1-11 low). So \( \text{and } \) and \( \text{go high and crate 0's flip-flops are reset. And so on until all}
the crates are finished.

Now notice the two DIP switches on page 2 of the encoder. Close
these on the last encoder and leave them open on all the other encoders.
When the last encoder is finished being read, pin 15 of gate Q goes high,
and this signal is FULL READOUT FINISHED, which goes to the mixer
and turns off the readout.