

[54] **DATA LOOP COMMUNICATION
PRIORITY ESTABLISHING
APPARATUS**

[75] Inventors: **Ray C. Fuller**, Cedar Rapids; **Gary D. Phillips**; **Richard B. Simone**, both of Marion, all of Iowa

[73] Assignee: **Collins Radio Company**, Dallas, Tex.

[22] Filed: **Jan. 6, 1972**

[21] Appl. No.: **215,808**

[52] U.S. Cl. **340/147 R, 340/147 LP, 340/163 R**

[51] Int. Cl. **H04q 5/00**

[58] Field of Search **340/147 LP, 163 R, 340/147 R**

Primary Examiner—**Harold I. Pitts**
Attorney—**Bruce C. Lutz et al.**

[57] **ABSTRACT**

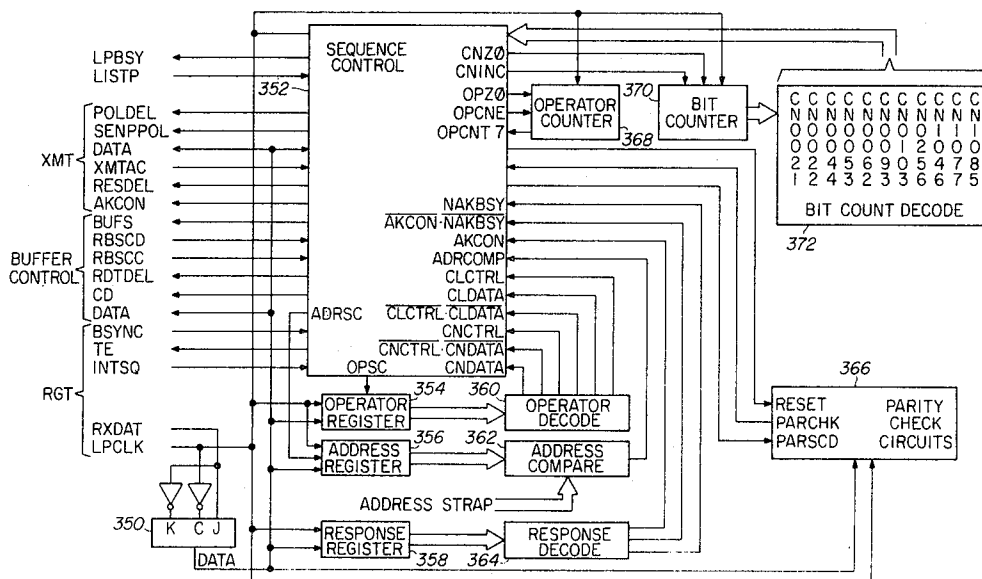
The method of configuring communication units, all of which are connected in series on a loop such that some units may take priority and communicate out of the logical sequential order. This is accomplished using a master unit and first and second sets of slave units. The slave units respond to poll requests. These polls may come either from the master or the previously responding slave unit. The master unit always sends a prepoll before a poll. The first set of slave units will respond to a prepoll or a poll and thus can take priority in transmitting messages. The second set of slaves will respond only to a poll, internally set a flag, and then transmit a poll to further slave units upon receipt of the next prepoll.

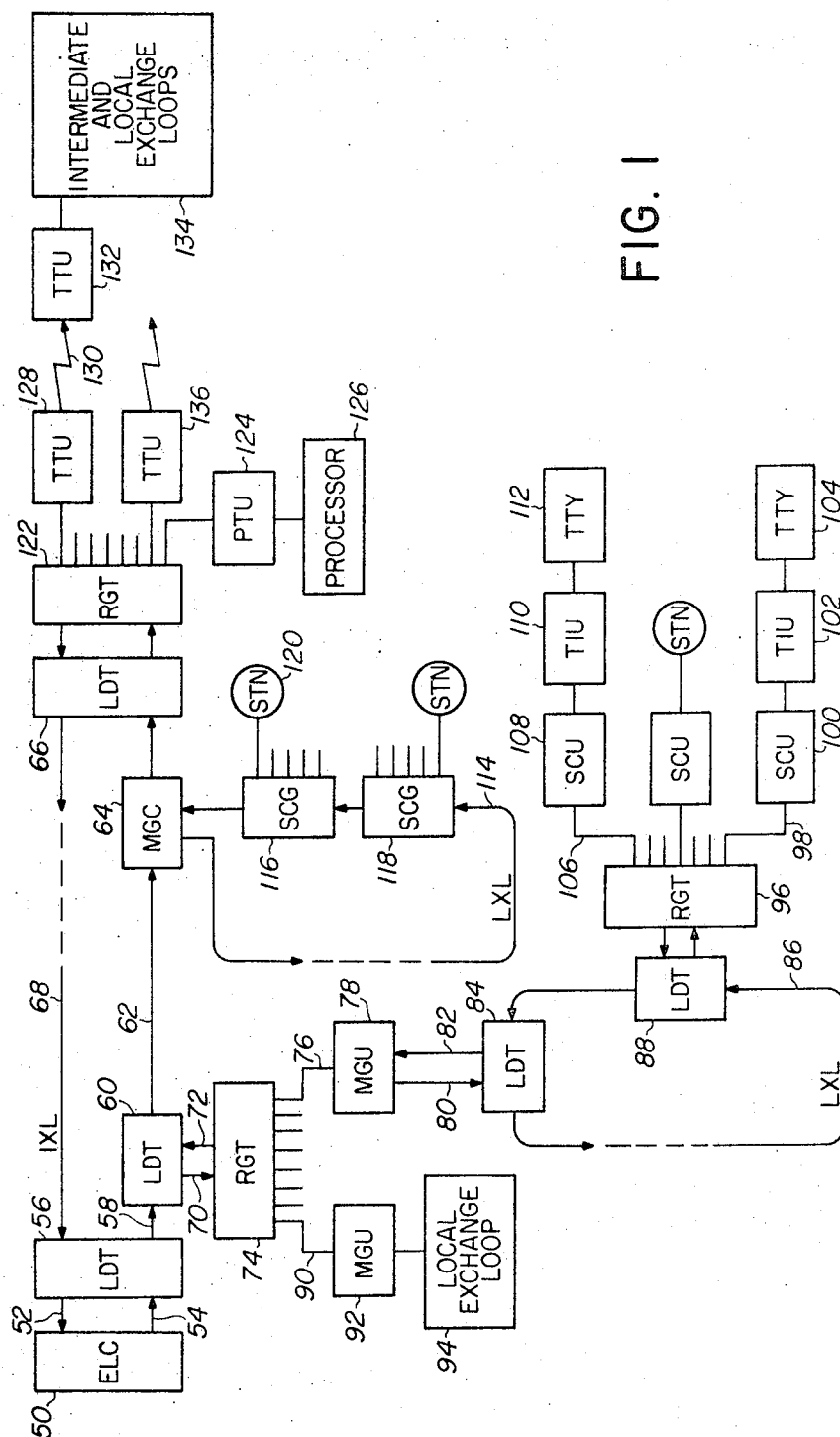
[56] **References Cited**

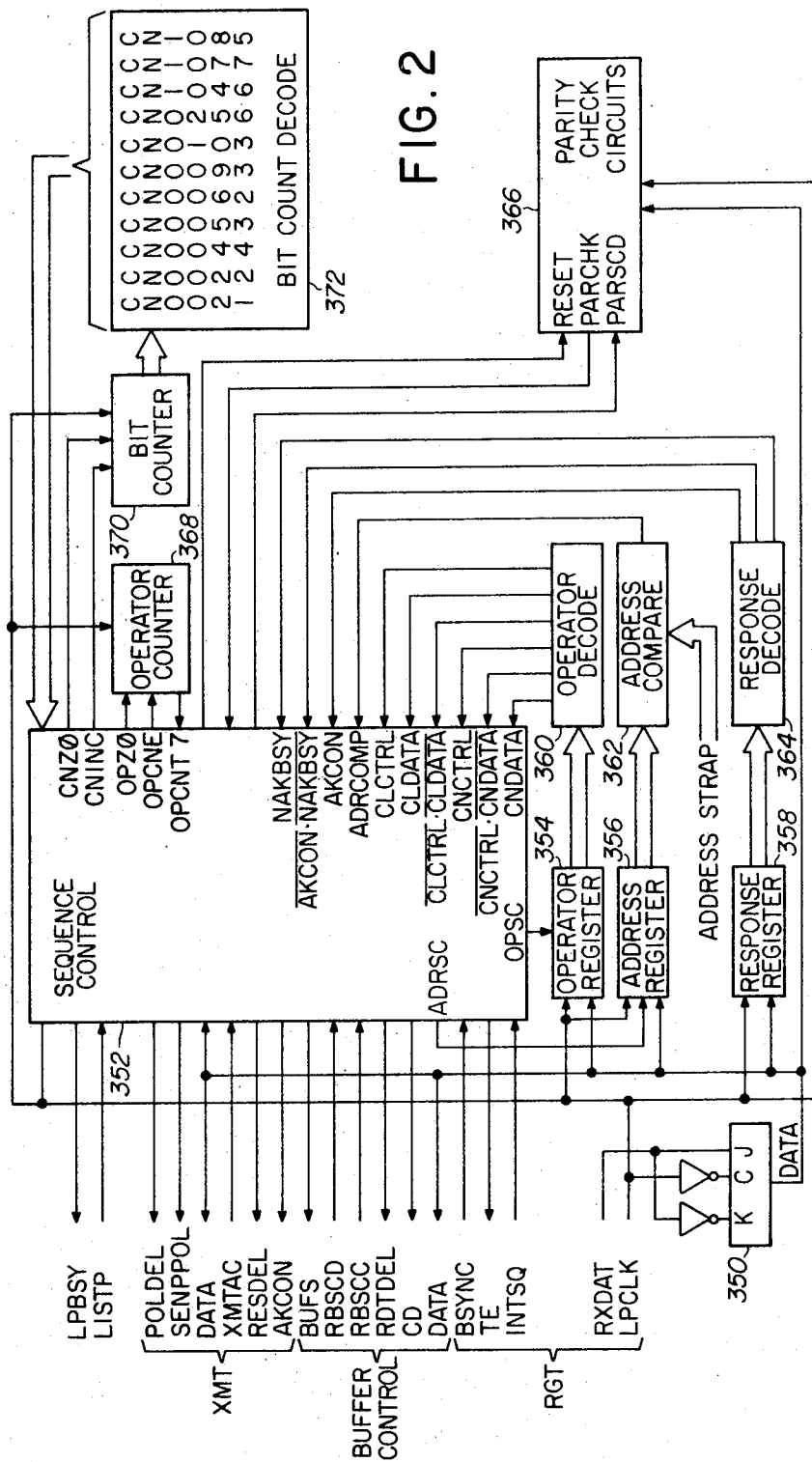
UNITED STATES PATENTS

3,435,416 3/1969 Kretsch **340/163**

7 Claims, 13 Drawing Figures







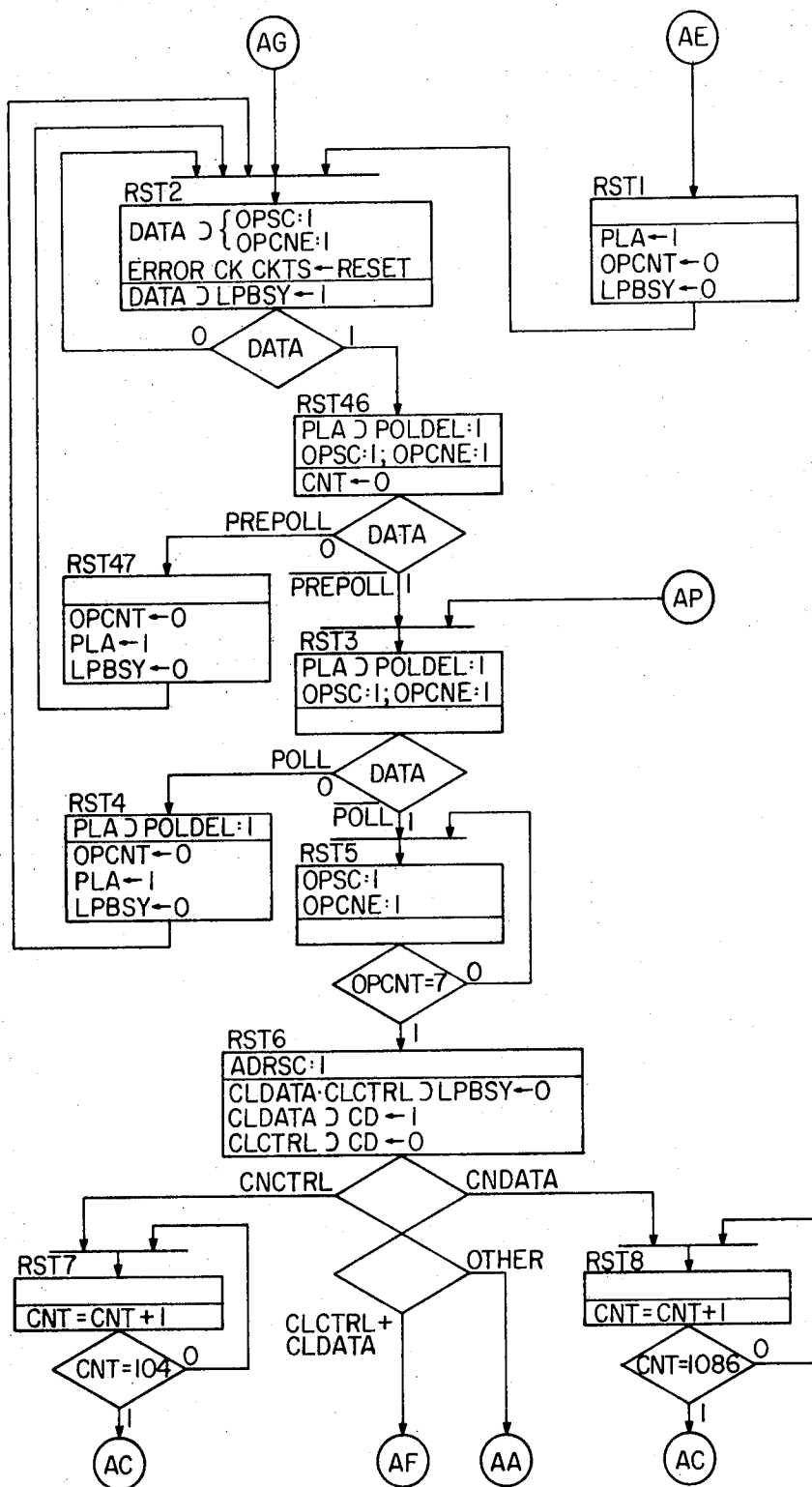


FIG.3A

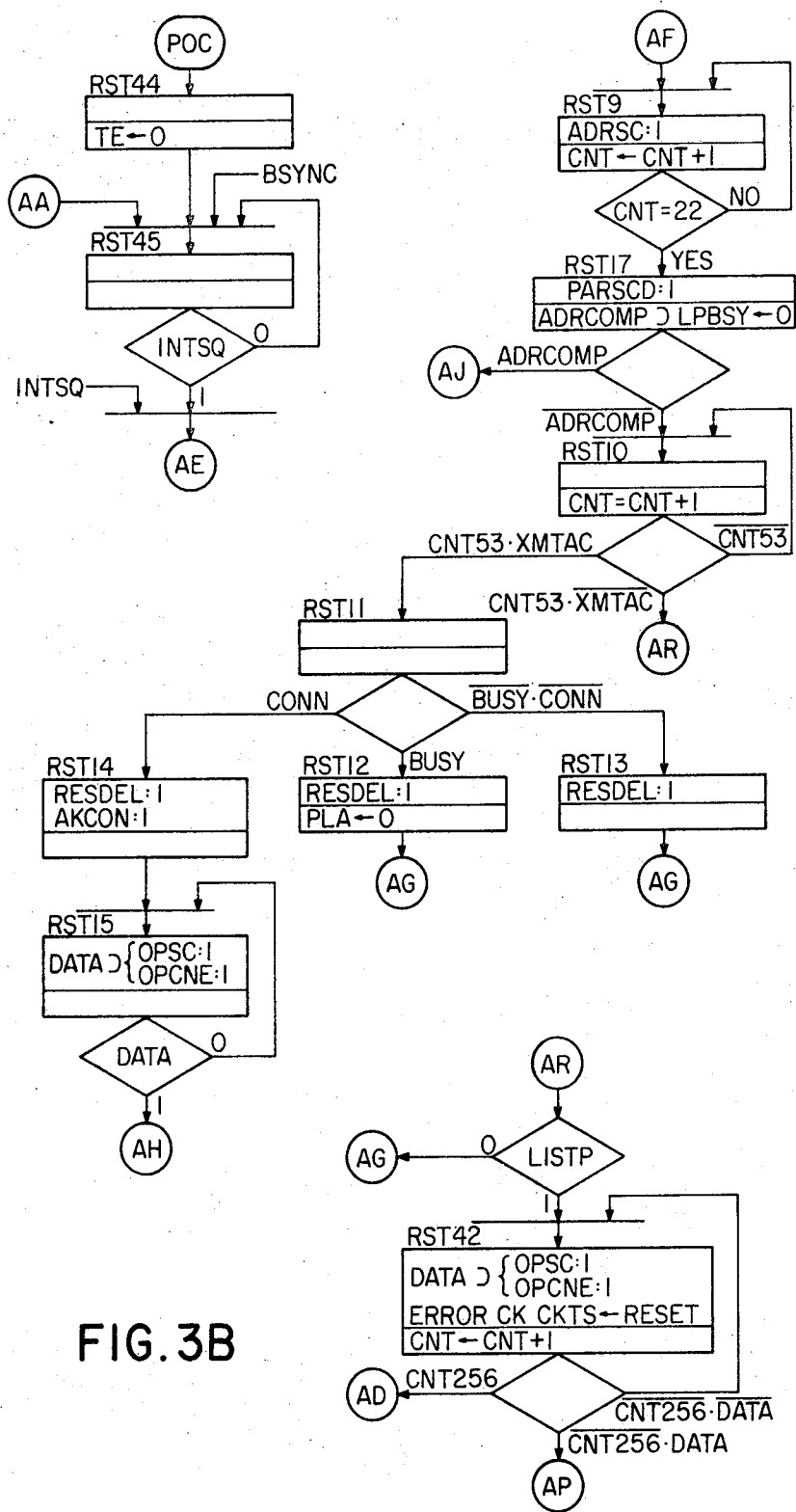


FIG. 3B

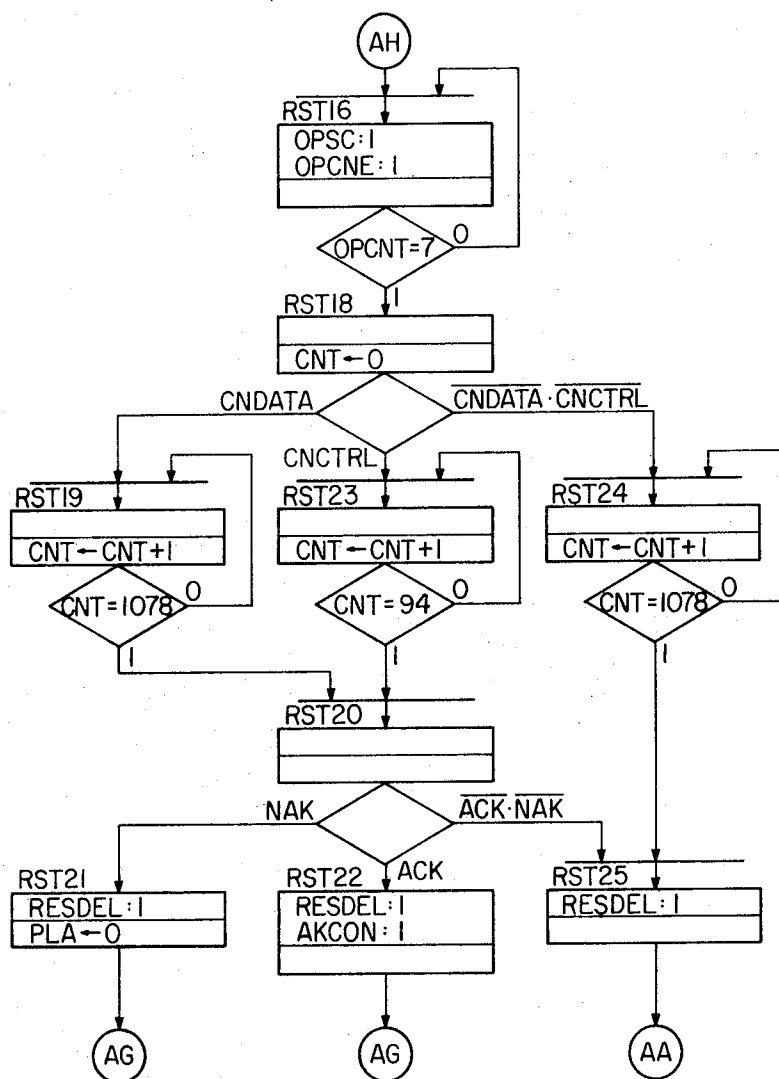


FIG. 3C

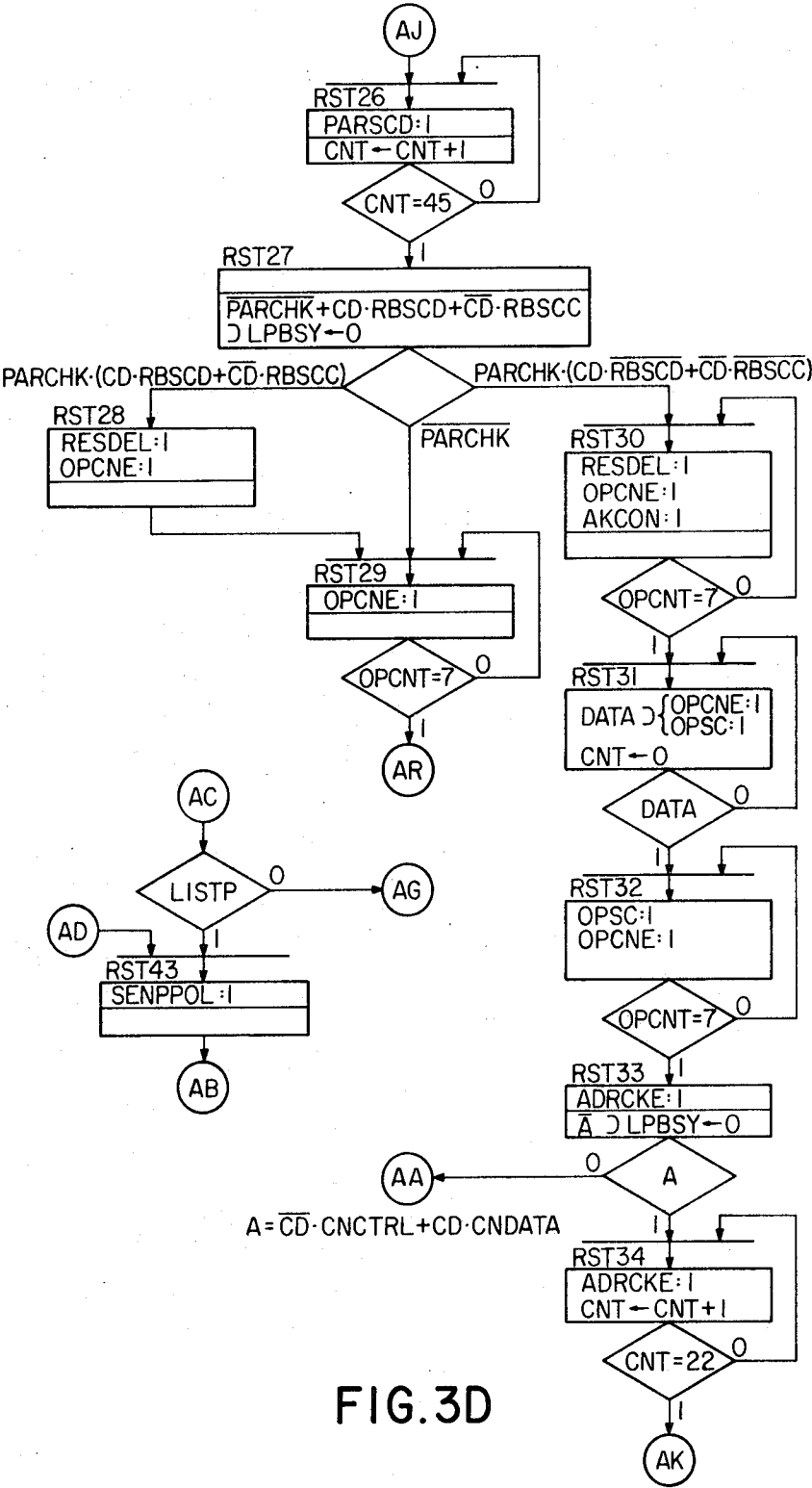


FIG.3D

SHEET 07 OF 13

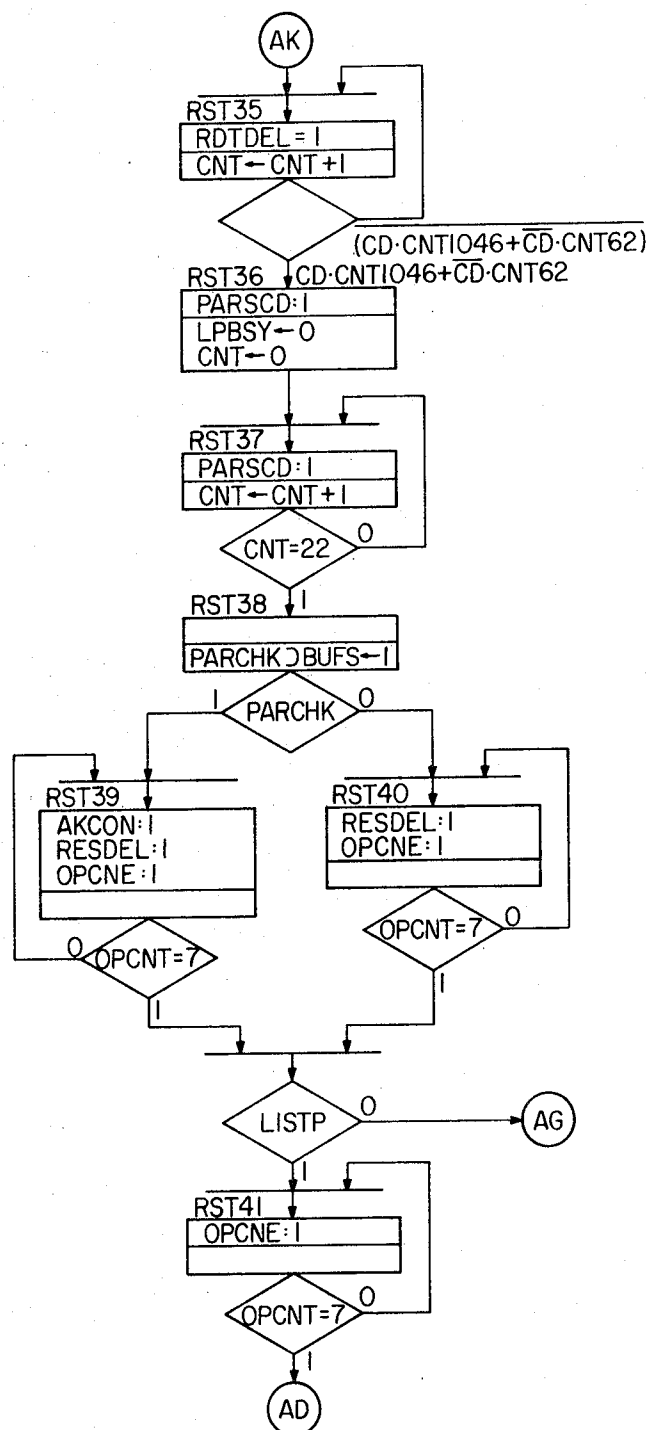


FIG. 3E

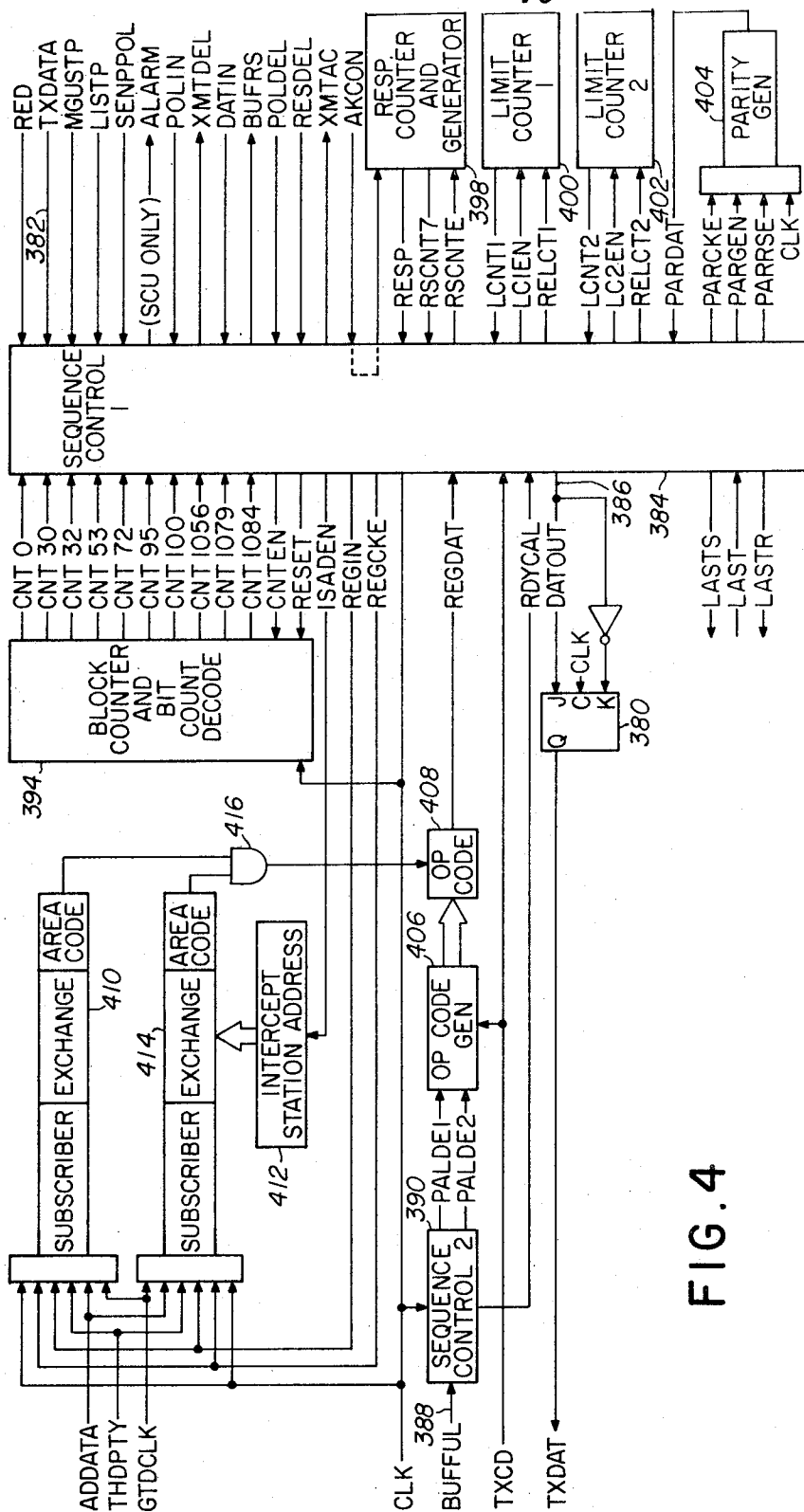
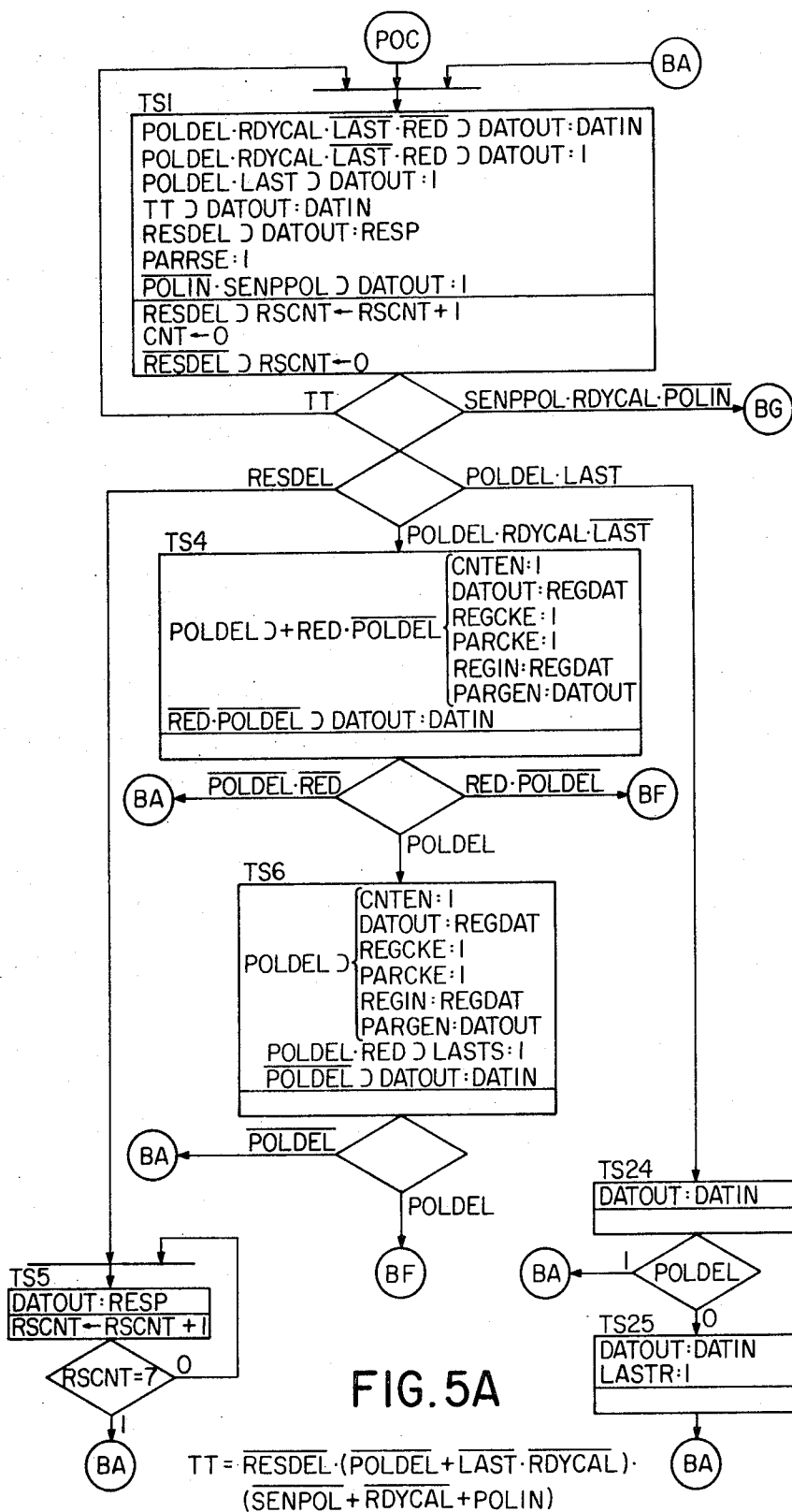
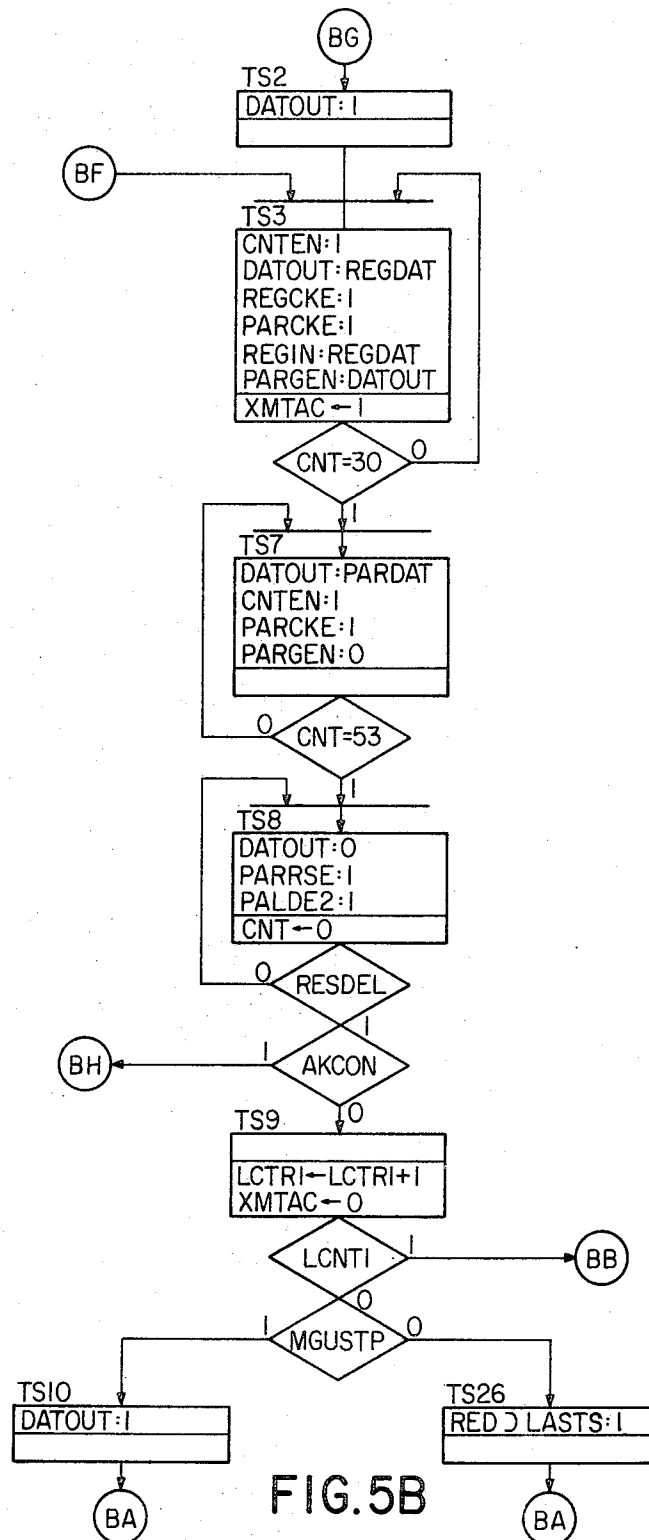
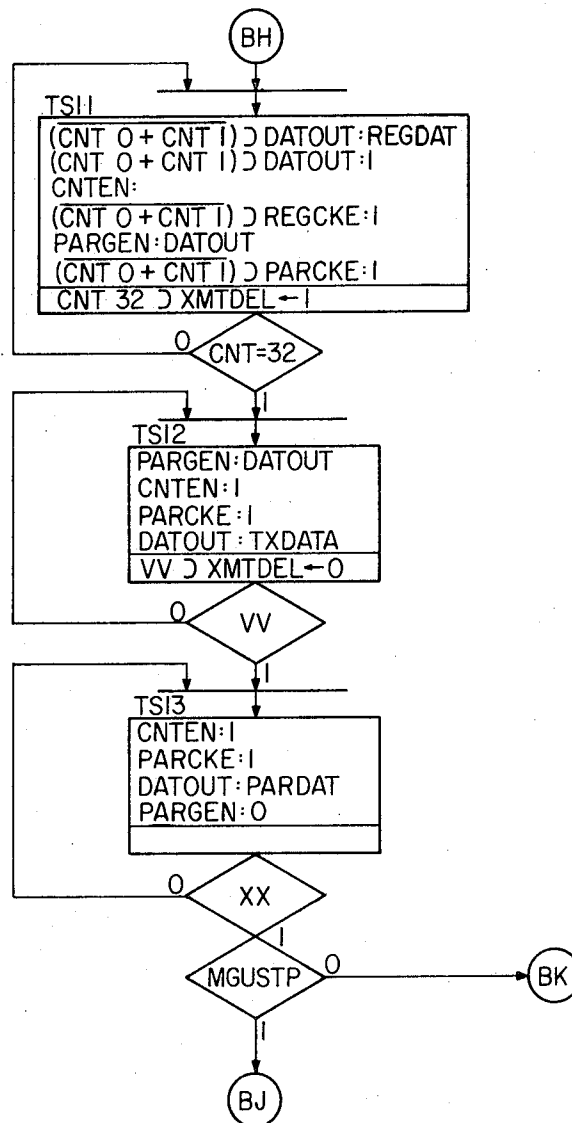


FIG. 4

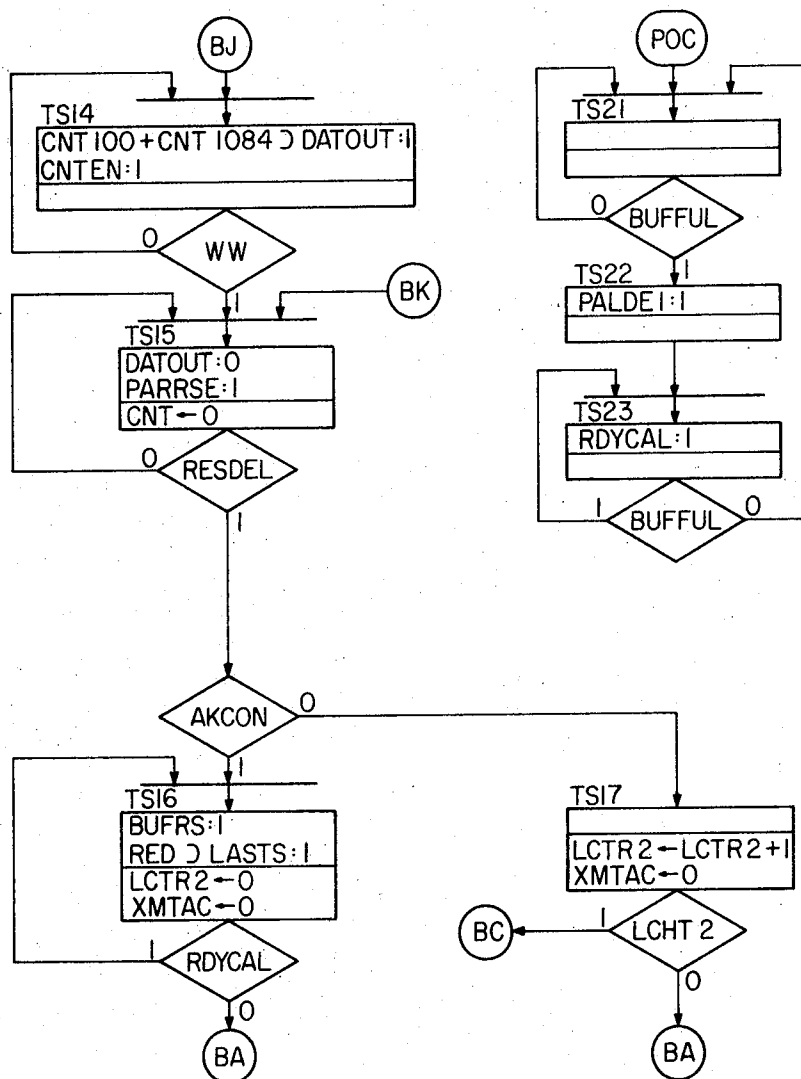






$VV = TXCD \cdot CNT = 1056 + \overline{TXCD} \cdot CNT = 72$
 $XX = TXCD \cdot CNT = 1079 + \overline{TXCD} \cdot CNT = 95$

FIG. 5C



WW = TXCD · CNT = 1084 + TXCD · CNT = 100

FIG. 5D

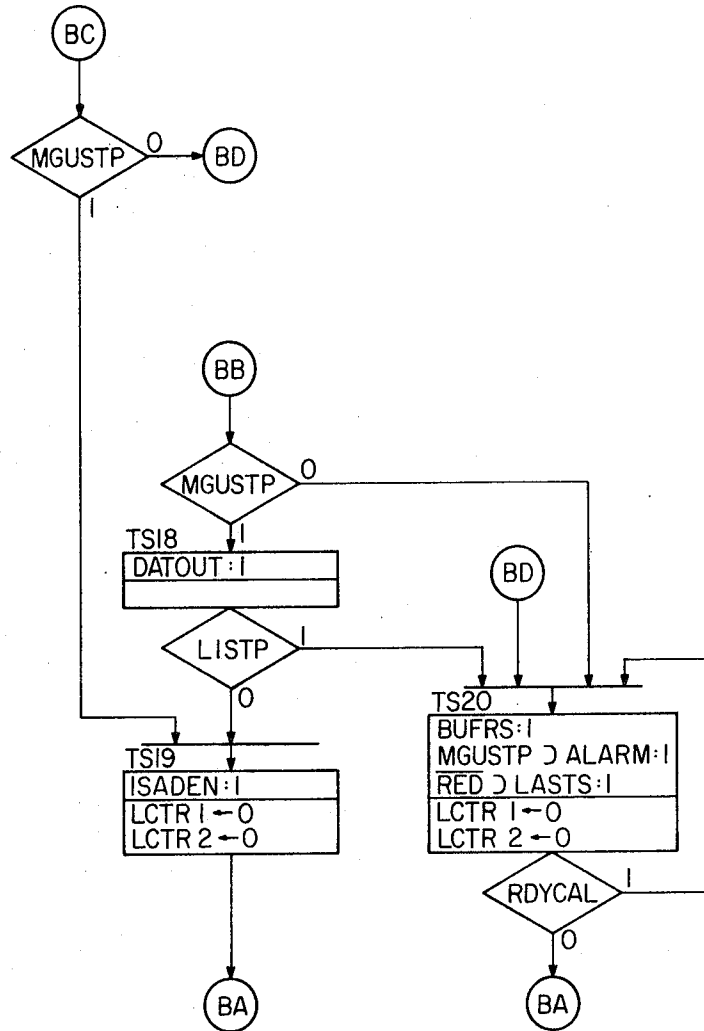


FIG. 5E

DATA LOOP COMMUNICATION PRIORITY ESTABLISHING APPARATUS

The present invention is generally related to electronics and more specifically related to communication switching system.

A co-pending application in the name of L. D. Hungerford et al. Ser. No. 208,548, entitled "Direct Switch System," filed Dec. 16, 1971, having the designation 4,064 and assigned to the same assignee as the present invention describes a communication system wherein a plurality of units connected serially on a loop respond to a poll signal originated at a control unit prior to sending messages. As will be realized, the devices closest to the control unit are most likely to receive the poll signal and thus may, in heavy traffic, prevent devices much further around the loop from ever receiving a poll so that they may transmit communications.

The communication problem of later units may be solved by requiring each unit to forward the poll to the next unit after completion of a single transmission. This is basically the way the problem is solved on the intermediate exchange loop of the above-referenced Collins application. However, the local exchange loop is designed such that a master or control unit is allowed to regain control of the loop between each message or communication for the purpose of interjecting messages into the loop for reception by units on the loop. This is necessary to prevent tie-up or clogging of the main or high speed loop.

This is solved, according to the present invention, by having the slave units on the local exchange loop respond only to a poll signal. The control unit will initially send a prepoll which, if unanswered, will result in the sending of a poll signal. The first unit to respond with a communication will set an internal flag so that the reception of the next prepoll signal by the control unit will result in an output of a poll signal from that slave unit. Thus, the poll will be forwarded to the next unit on the loop desiring to communicate. The control unit may at any time delay sending further prepoll signals and instead supply messages to units on the loop. When no further messages are forthcoming from the control unit, it can then send further prepolis which will be changed to polls by the last unit to transmit a message and this poll will continue around the loop to the originating control or master unit.

One or more of the slave units may be given priority whereby they will respond to either a prepoll or a poll in the transmission of messages. It will be realized, however, that too many of these priority units will cause the same clogging conditions in the local exchange loop as occasionally happened previously and which the present invention was designed to eliminate or substantially reduce.

It is therefore an object of the present invention to provide a priority and non-priority situation in polled loop connected units.

Other objects and advantages of the invention will be apparent from a reading of the specification and claims in conjunction with the drawings wherein:

FIG. 1 is a block diagram of an overall direct switching communication system as more completely described in the above referenced co-pending application and which may advantageously embody the present invention;

FIG. 2 is a block schematic diagram of a receive module portion of a SCU or other unit connected to the local exchange loop of FIG. 1;

FIGS. 3A-3E illustrate the flow diagram applicable to the apparatus of FIG. 2;

FIG. 4 is a block diagram of the transmit module portion of a SCU or other device connected to the local exchange loop of FIG. 1; and

FIGS. 5A-5E are flow diagrams applicable to the transmit module of FIG. 4.

DIRECT SWITCHING APPARATUS OF FIG. 1

A more detailed description and understanding of FIG. 1 may be obtained from the above-referenced patent application. However, in general the apparatus of FIG. 1 provides communication between any units on a specific loop such as the local exchange loop 86 and communication between SCU's 100 and 108. This is accomplished by having the master control unit 78 transmit a poll which is received by one of the SCU's which wishes to transmit a message. This unit then transmits the message in response to the received poll. The transmitted message contains an address which is decoded by the appropriate receiving SCU who changes an acknowledge portion as appropriate before forwarding the message back to the original sender. If communications are desired between SCU's on different loops such as between loops 86 and 114, the SCU such as 100 will transmit a message in response to a receive poll and this message is stored in a buffer in MGU 78. MGU 78 changes the acknowledge section of the message to indicate to SCU 100 that a successful reception has occurred. MGU 78 then awaits a turn on intermediate exchange loop 68 so that it may transmit the recently received message to MGC 64. After MGC 64 receives and acknowledges its message, it calls an appropriate station such as 120 on local exchange loop 114 and provides station 120 with the message. After station 120 acknowledges receipt of the message, it returns a reply message to SCU 100, buffer by buffer, back through intermediate exchange loop IXL to local exchange loop 86. Then, SCU 100 transmits the next portion of the message.

As explained above and as will be apparent, the polls sent out by MGU will be received first by the closest SCU's such as 100 as illustrated. If SCU 100 has a large quantity of messages to transmit, it is conceivable that a poll will never reach SCU 108 and it could not provide communications. Thus, as explained in the succeeding flow diagrams, MGU 78 supplies a prepoll on line 86 which comprises a logic "1" followed by a predetermined number of 0's. This prepoll is not recognized by SCU 100 unless an internal flag is set due to it being the last SCU unit to send a message. In this instance the prepoll is changed to a poll which comprises two logic 1 bits. The following SCU's such as 108 may now receive this poll signal sent by 100 and respond thereto. This poll signal will continue around the loop until received by a SCU having a message to transmit or by MGU 78. Any following unit having a message to transmit will likewise later change the next prepoll from the control unit to a poll and forward this poll around the loop for use by any later units. When the poll is finally forwarded to MGU 78, it realizes that all SCU's on the loop have had the opportunity to transmit a message

and it thus sends a new poll so that the sequential transmission by all SCU's containing messages can continue.

RECEIVE CONTROL MODULE

FIG. 2

FIG. 2 illustrates the receive control module found in the LRT sections of the various units of FIG. 1. As will be noted in the referenced patent application, there are two receive control modules in the master group control block and one in each of several other blocks.

As indicated on the left-hand side of FIG. 2, there are leads coming from and going to the RGT interface as well as leads coming from and going to the buffer sections of the appropriate devices and leads going to and coming from the transmit control section. Within the FIG. a J-K flip-flop 350 is illustrated as well as a sequence control unit 352, and operator register 354, an address register 356 and a response register 358. Cables are utilized to connect the last three referenced registers to operator decode 360, address compare 362, and response decode 364, respectively. In addition a parity check circuit 366 is illustrated along with an operator counter 368, a bit counter 370, and a bit count decode 372. It should be realized that other polynomial check codes may be used other than parity and that further references to parity are by way of example only. A receive data (RXDAT) supplies input data from the RGT to a J input of the J-K flip-flop 350 and through an inverting circuit to the K input of J-K flip-flop 350. A loop clock input supplies data through an inverter to the C or clock input of flip-flop 350 as well as to blocks 352-358 and 366-370. The bit sync and initialization sequence inputs are supplied to the sequence control block 352 and a transmit enable is supplied from control block 352 to the RGT. The leads to and from the buffer portion of the particular unit in which the receive control module is utilized are, respectively, BUFS (bufferfull set), RDTDEL (receive data delimiter), and CD (control/data) each of which is obtained from sequence control 352. Another line to the buffer unit is a data lead which is obtained from the output of J-K flip-flop 350. The J-K flip-flop 350 provides the one-half bit delay, which was previously mentioned, by having the loop clock 180° out of phase with the received data. A further J-K flip-flop in FIG. 4 provides a further half bit delay in the same manner in the transmit control module to produce the one full bit delay for each module connected to a loop. The data lead from f/f 350 is also supplied to blocks 352-358 and 366 and is output to the transmit control block. Two leads labeled RBSCD and RBSCC (receive busy select call data and receive busy select call control, respectively) supply signals from the buffer portion to the sequence control unit 352.

From the transmit control block a single input is received labeled XMTAC (transmit active). This transmit active lead is supplied to the sequence control unit which returns five signals to the transmit control block of FIG. 4 via leads POLDEL, SENPPOL, DATA, RESDEL, and AKCON. These last referenced leads refer to poll delimiter, send prepol, data, response delimiter, and acknowledge/connect, respectively. A further lead from the sequence control is the LPBSY (loop busy) lead when the receive control module is being used in certain units.

The operator register 354 receives an additional input CPSC (opcode register shift control). The address register 356 also has an additional input ADRSC (address shift control). The operator decode has six outputs supplied to the sequence control 352. The first two are call control and call data (CLCTRL and CLDATA, respectively). The next is a signal which indicates that it is neither a call control or a call data (CLCTRL-CLDATA). Two further inputs from the operator decode supply a signal for connect control or connect data (CNCTRL and CNDATA, respectively). Finally, a signal is supplied indicating that the input decoded is neither a connect control or a connect (CNCTRL-CNDATA).

The address compare 362 supplies a single input labeled ADRCOMP to the sequence control 352. The response decode 364 supplies three inputs to sequence control 352 of acknowledge/connect (AKCON), negative acknowledgement/busy (NAKBSY), and neither of the above (AKCON-NAKBSY). The parity check circuit 366 was previously indicated as receiving the loop clock and the data signal, and it also receives a reset input and a parity shift delimiter (PARSCD) from the sequence control block. In addition a parity check output (PARCHK) is supplied from 366 to the sequence control 352. The operator counter 368 receives two control signals OPZO and OPCNE (operator zero and operator count enable, respectively). The sequence control unit 352 also supplies the operator the bit counter 370 with an input CNINC (bit counter clock line). The operator counter 368 returns an input to sequence control labeled OPCNT7 (operator count 7).

The bit counter 370 also receives an input CNZO which is the bit counter reset line. This input accomplishes the same result as the OPZO input to counter 368. The bit counter 370 supplies a multiple lead output to the bit counter decode 372 which provides outputs, upon the decoding of counts of 21, 22, 44, 53, 62, 93, 103, 256, 1,046, 1,077, and 1,085, back to the sequence control 352.

In view of the extensive use in mnemonics in FIGS. 2 and 3, a mnemonic list is provided below as a simple reference to the mnemonics used in these two figures.

RECEIVE MODULE MNEMONIC LIST

- ACK (Acknowledge) Positive response field in a connect message.
- ADRCKE (Address Clock Enable) Enables called address transfer from receive module to transmit module.
- ADRCOMP (Address Compare) Checks for comparison between received address and receive module address.
- ADRSC (Address Register Shift Control) Enables the loading of the 24-bit address register.
- AKCON (Acknowledge/Connect) Indicates positive response to reception of a message from receive to transmit module.
- BSYNC (Bit Sync) Logic "1" indicates demodulator is in sync; logic "0" indicates demodulator is out of sync.
- BUFS (Buffer Full Set) Decode signal which establishes that the buffer requested is full to the buffer control from the receive module.

BUSY (Busy) Response field in a **SELECT** call message which indicates to the receive module that the called party is busy.

CD (Control/Data) Line from receive module which indicates the type of **CONNECT** block received. Logic "1" implies data; logic "0" implies control.

CLCTRL (Call Control) **SELECT** call control field which indicates a control message will follow. (Operator code plus start bit — 11101100)

CLDATA (Call Data) **SELECT** call data field which indicates a data message will follow. (Operator code plus start bit — 11100011)

CNCTRL (Connect Control) **CONNECT** message supervisory opcode. (Operator code plus start bit — 11110000)

CNDATA (Connect Data) **CONNECT** message data opcode. (Operator code plus start bit — 11111111)

CN0021 A bit counter decode of count 21.

CN0022 A bit counter decode of count 22.

CN0044 A bit counter decode of count 44.

CN0052 A bit counter decode of count 52.

CN0062 A bit counter decode of count 62.

CN0093 A bit counter decode of count 93.

CN0103 A bit counter decode of count 103.

CN0128 A bit counter decode of count 128.

CN1046 A bit counter decode of count 1046.

CN1077 A bit counter decode of count 1077.

CN1085 A bit counter decode of count 1085.

CNINC Bit counter clock line.

CNZO Bit counter reset line.

CONN (Connect) Positive response field in a **SELECT** call message.

CNT (Counter) Modulo 2048 counter to count message bits.

INTSQ (Initialization Sequence) Normally a logic "0." Generates a one shot logic "1" after 1537 zeros have been counted on the line.

LPCLK (Loop Clock) Clock line from Regenerative Tap (RGT).

LISTP (Local-Intermediate Loop Strap) Logic "1" enables MGC poll generator for local loop polling.

LPBSY (Loop Busy) Indicates receive module may be in the process of receiving a call.

NAK (Negative Acknowledgement) Negative response field in a **CONNECT** message.

NAKBSY Indicates negative response to reception of any message.

OPCNE (Opcode Counter Enable) Enables the modulo 8 counter 368 to increment.

OPCNT (Opcode Counter) Modulo 8 counter 368.

OPCNT7 Decoded count 7 of operator counter.

OPSC (Opcode Register Shift Control) Enables the loading of an 8-bit opcode register.

PARCHK (Parity Check) Signals good message parity.

PARSCD (Parity Shift Delimit) Delimits parity check field.

PLA (Poll Active) Enables poll delimit to be sent to the transmit module when a poll is detected.

POC (Power On Clear) Establishes all initial conditions for receive module during system connection.

POLL (Poll) Signal for activating non-priority loop devices (Operator code — 1000000)

POLDEL (Poll Delimit) Signal to delimit loop poll condition to the transmit module.

PREPOLL (Prepoll) Start bit signal for activating priority (Red) units and for alerting non-priority (White) units that they may be activated on the next message if they are in the next position on the loop after the last white unit to transmit. (Operator code plus start bit — 10000000)

RBSCC (Receive Busy Select Call Control) Busy to a **SELECT** call control message.

RBSCD (Receive Busy Select Call Data) Busy to a **SELECT** call data message.

RDTDEL (Receive Data Delimit) Signal which delimits the control parameters or data portion of a **CONNECT** message.

RESDEL (Response Delimit) Signal which delimits response field to the transmit module.

RXDAT (Receive Data) Data line from RGT.

SENPPOL (Send Prepoll) Requests the transmit module to send a prepoll signal or start bit.

TE (Tap Enable) Synchronously routes data from regenerative tap unit (RGT) to receive module after POC.

XMTAC (Transmit Active) Signal which delimits the condition which implies that the transmit module has sent a **SELECT** call message.

The purpose of the receive module is to take the incoming message and check its operator, address and response portions to determine the type of message being received along with checking its parity to make sure there are no mistakes in the message before transferring the message into the data buffer and/or indicating to the transmit control section of the LRT to send a return response of ACK, NAK, etc. The message used in one embodiment of the invention comprised first a start bit of one digit, an operand or operator code of seven digits, a called address vector of 24 bits, a data portion of either 40 or 1,024 bits depending upon the operator portion, a check field of 24 bits, and a response field of 8 bits. The 40-bit data field is utilized for control messages while the 1,024-bit data field is used for data messages. The operator register 354 is enabled from the OPSC input and the output is decoded via decode block 360 which provides an input to sequence control 352. The address register 356 is enabled by the output ADRSC for comparing the received address with the address supplied from the address strap input to address compare circuit 362 to determine whether or not the address is the same as the present station. The address strap is used so that the address of a station can be changed to any given call code.

The response register 358 continuously receives input data and the output is merely ignored until the proper time. This proper time is obtained by checking the operator to determine whether a call word is a data or control message and then using the bit count decode 372 and the operator counter 368 to determine the proper time to start checking the response decode and the ending of this check 8 bits later. Since the time at which parity check occurs depends upon the type of word, this circuit is also dependent upon the response obtained from operator decode 360. Thus, an input labeled parity shift delimit is utilized to enable the parity check circuit only during the proper time.

Returning to the address compare section it will be realized that if the station incorporating the receive control module is awaiting a call, the address should compare. However, if the station has just transmitted a call and is awaiting a return of that call to determine the response, the address will not compare.

The response register and its decode 364 are utilized to determine whether the call which is returned contains a busy or connect signal or an acknowledge or not acknowledge response. There will on occasion be a faulty return word wherein the response section is such that none of these responses is decoded. This bit of information is also provided to the sequence control unit. As will be ascertained from a study of the discussion in connection with FIG. 3, the sequence control utilizes this information also.

RECEIVE CONTROL MODULE FLOW DIAGRAM

FIG. 3

Since the block diagram of FIG. 2 can be implemented utilizing a plurality of logic circuits and registers and since these circuits can be implemented in such a variety of ways, it is believed that a flow diagram of the operation of FIG. 2 will provide more information to one skilled in the art. In this manner the apparatus can be easily programmed into a computer in practicing the invention. However, since the flow diagram is quite detailed, the physical circuit design will also be apparent from an understanding of the flow chart.

The flow charts of FIG. 3 are divided into five pages labeled 3A to 3E. Normally, a receive module is a passive unit although it is responsible for initiating a prepoll if it was the last device to react to a poll in its order of placement on the loop. A local-intermediate loop strap is utilized in a MGC to force a return to state 2 (RST2) whenever the device reaches a state indicated as AG as shown in FIGS. 3B, 3D, and 3E. This strapping operation is performed upon installation of the device and is included to illustrate a universal embodiment even though not required as part of the explanation of the SCU unit operation which has the inventive subject matter of this application.

Referring now to FIG. 3B, a POC or power on clear signal is supplied to state 44 after actuation of the device. The device will then set the tap enable to zero and pass to state 45.

In the present flow diagram and the remaining flow diagrams, it should be noted that the blocks for each state have upper and lower portions. The upper portion defines an immediate action upon state entry. The lower portion of the block defines a clocked action on exit from the state. The symbol ":" denotes an action which occurs upon entering a state, or immediately thereafter, and exists only during the existence of that state. Thus, such an action cannot, by definition, occur during the bottom portion of a flow chart block. The symbol "→" denotes a setting, which is often a flag or flip-flop setting, external to the flow diagram actions. This "→" symbol is a "permanent" setting, which remains until set otherwise and can occur either upon entering (top part) a state or upon leaving (bottom part) a state. Thus, the receive module remains in state 44 as long as power on clear (POC) is present. The first

clock occurring after POC is removed causes a change from state 44 to state 45. While the receive module remains in state 44, tap enable (TE) is set at logic "1." Upon completion of POC during state 44, the device is ready to enter state 45 upon receipt of the next clock at which time TE is set to logic "0."

As will be noted, state 45 receives three other inputs besides the one from state 44. The device remains in state 45 until the initialization sequence (logic "1") is obtained from the RGT. Prior to this time the device remains in an idle or recirculating condition since the output of the initialization sequence quadrangle continually indicates that the initialization sequence line is zero. On the simultaneous occurrence of a "1" and a clock signal to the sequence control, the device proceeds to state 1. As the device passes through state 1 the poll active signal is set to one to enable a poll delimit signal to be sent to the transmit module when a poll is detected. The opcode counter 368 is set to zero and the loop busy output is set to zero. The device then passes to state 2 and idles there until a start bit is received indicating the possible commencement of data.

The character which looks like a backward C in both the upper and lower portions of the state 2 block should be interpreted in this specification to mean "means the same as, or implies." Thus, upon each passage of the idling status through the data decision block the error check circuits are reset. Upon the occurrence of the start bit or first data bit the loop busy lead is set to a one, and the opcode register shift control and opcode counter enable are momentarily set to one.

In state 46 the second bit is examined at which time the count is set to zero. If the second bit is a zero it will proceed as a prepoll indication to state 47. If the second bit is a logic 1, the device will know this is not a prepoll and is either a poll or a call. Thus, it will proceed to state 3. During the existence of state 46 the poll delimit line will be set to a one if the poll active flag is a one and the opcode register shift control and the opcode counter enable are momentarily set to a one.

Assuming the second bit is a logic zero, the device in state 47 will return the opcode counter 368 to zero and the loop busy output to zero. Further, the poll active signal is set to one if not previously set there.

If the second bit is a logic '1' thereby indicating that the received message is not a prepoll, the device in state 3 will examine the third bit. If the third bit is a logic 1, the apparatus will realize that the received message is not a poll and will proceed to state 5. However, if the third bit is a zero, a poll is indicated and the device will go to state 4. In state 3, the opcode register shift control and opcode counter enable are again temporarily set to one. Further, if poll active is still a one from state 1 or is a one for some other reason, the poll delimit is set to a one.

Assuming the condition of the third data bit is a zero, the apparatus enters state 4 where the op count, and loop busy are again set to a zero while the poll active is set to "1" if zero or if already a "1" causes the poll delimit to be set to a one. The device then returns to state 2 and again waits for a start bit. However, if the third bit is a one, the device knows that this is not a poll and proceeds to state 5. It remains circulating in state 5 until the operator counter 368 provides a count of 7.

On each clock bit through state 5 the OPSC and OPCNE are set to a one temporarily. When the counter 368 reaches a count of 7, the device proceeds to state 6. This count of 7 is 7 plus the bit which allowed the apparatus to pass through the states 2 and 46. Thus, as the device reaches state 6, there are actually 8 bits or clock pulses since the first data bit was received.

It should be noted at this point that the operator counter must have already triggered the enabling mechanism to a count of 7 before it can detect such a count. Thus, it returns to state 7 after the count indication is 7 and upon leaving for state 6 it adds one more pulse thereby resetting the modulo 8 counter 368 to a logic 0 for later counts such as the count of 7 following state 29 in FIG. 3D and other portions in the flow diagram.

In state 6 the decoded output from operator decode 360 is checked by the decision boxes. If the operator decode provides anything other than a connect-control, a connect-data, or a call-control and call-data, the device will proceed to state 45. The passage through state 6 will activate the address register shift control lead to a temporary one even though the opcode indicates improper operation.

If the output from the operator decode 360 indicates a connect-control, the device will proceed to state 7. The bit counter 370 was started counting at the same time that operator counter 368. The device will stay in state 7 until bit counter 370 reaches a count of 104. At this time the decode 372 provides an output which enables the device to change from state 7 to the decision block LISTP in FIG. 3D. If the unit is in a MGU, the device will proceed to state 43; and as it passes therethrough, will send a prepoll and return to state 1. However, in most installations the device will be strapped to a zero and the device will return to state 2 directly and await the reception of another data or start bit.

If the operator decode provided an indication of connect-data in the decision block of state 6, the device would proceed to state 8 where the device would remain in an idling or recycling state until the decode counter 372 reached a count of 1,086. The count of 1,086 is the length of a complete data message including the header, parity check, and response sections. Upon a count of 1,086 the device again proceeds to the decision block in FIG. 3D where it sends a prepoll or returns directly to state 2.

The reason for the MGU initiating a prepoll on the local exchange loop is to give it priority on sending messages. When the operator decode indicates that there is a CONNECT message on the loop, the MGU merely waits until the message has passed and sends a new prepoll so that further devices on the loop can respond with calls. If the MGU has a message from the intermediate exchange loop which it wishes to place on the local exchange loop, it merely withholds the prepoll bit and instead sends the message out to the appropriate device on the local exchange loop.

Later in the flow diagram the receive control module will react differently to a CONNECT message.

It will now be assumed, after returning to state 2, that a further message is decoded which is either a call-control or a call-data message. In this event the flow sequence leaves state 6 through the decision control

block and proceeds to state 9 in FIG. 3B. The apparatus reaches state 9 at the time of the first bit of the called address portion of the data word. Thus, the address register shift control lead is set to a one and the bit counter is incremented for 22 bits until the bit count decode 372 supplies an output indicating that count 22 is reached. Upon receipt of this signal, the device moves to state 17. The address register is no longer activated since it is only activated long enough to admit 24 data bits into the address register. The 24 bits are obtained by one bit being entered from state 6 and 22 bits being entered from state 9 before the apparatus makes its final loop through state 9 and finds that the 22nd count occurred as it left the decision block with a "no" indication on the previous recycle. In other words, the count is incremented after the decision is made whether or not the count equals the desired amount. Thus, 23 counts are obtained from state 9 and one count is obtained from state 6 to make a total of the 24 counts for the called address vector.

It should be noted at this point that while both the operator counter 368 and the bit counter 370 were set to zero in state 46, the operator counter is already incremented by the setting to one of the OPCNE lead in the upper portion of the various state blocks. The bit counter is already incremented by the $CNT = CNT + 1$ in the lower portion of the blocks. Thus, the count of 22 during state 9 had not been previously affected by the count indicated in state 5.

From the decision block indicating a count of 22 has been reached, the device now enters state 17 where it awaits an output from address compare 362. If the address compares, the device proceeds to state 26 on FIG. 3D. As it leaves state 17 the parity shift control delimit is temporarily set to one and since the address did compare the loop busy is set to a zero. If the address did not compare then, of course, the loop busy would be left alone. In state 26 the device idles or recycles for a number of bits while advancing the count register until on the 46th counted bit the device proceeds to state 27. As may be noted from FIG. 26 of the referenced application, a SELECT mode call may contain a parity check section for the 24 bits after the called address vector. Thus, an output is obtained from parity check circuit 366 and the sequence control determines whether the parity checked or did not check. If parity did not check the device proceeds directly to state 29. As may be ascertained from the lower portion of state 27 the occurrence of a parity not checking will reset the loop busy line to a logic "0." In state 29 the opcode counter enable circuit is temporarily set to one on each circle of the recycling device until a count of 7 is reached and then upon the next cycle the device proceeds to FIG. 3B and enters the strap decision block prior to state 42. If the device is anything other than a MGU, the apparatus will return to state 2 and again await an incoming data bit. However, if it is a MGU, the device will proceed to state 42 and wait for a count of 256. The error check circuits are reset on each cycle. If prior to count of 256 a data bit is detected, the device will proceed to state 3 of FIG. 2A. However, if no data bit is detected prior to count 256, upon reaching count 256 the device will proceed to state 43 on FIG. 3D where a prepoll will be sent and the device will return to state 2 to await the return of the prepoll, a poll, or a further data message.

Returning to state 27 it may be assumed that, instead of not obtaining a parity check, a parity check is obtained. If either RBSCC or RBSCD is a logic "1" along with an indication of parity check, the device will proceed to state 28. The raising of either of these lines indicates that the station is busy to any calls whether control or data.

In state 28 the response delimiter is raised to a "1" as well as raising the operational counter enable to a "1." The device then proceeds to state 29 and continues as previously described to state 42 to send a prepoll if it's a MGU or returns to await further data bits if the device is not a MGU.

If the parity checks and neither of the receive busy SELECT call lines are raised to a "1," the device proceeds to state 30. It cycles through state 30 for seven counts until on the 8th count it passes to state 31. As before the 8th count returns the operator counter to a logic zero condition so that it may be operational for future counts. During each cycle the response delimiter, the opcode counter enable, and the acknowledge/connect lines are placed in a logic "1" state.

The device then proceeds to state 31 where it awaits the receipt of additional logic bits. The raising of the acknowledge connect line in state 30 to a logic "1" supplied a signal to the transmit control block enabling it to supply the proper response in the final 8 bits of the SELECT mode call word to be returned to the calling device. The receive control block is now awaiting a CONNECT message from the calling device. When the device receives its first data bit the opcode counter enable and the opcode register shift control leads are placed at a logic "1" temporarily while the count is shifted to a "0." In state 32 the device recycles for seven counts during each of which the opcode register shift control and opcode counter enable leads are temporarily placed in a logic "1" condition. On the next clock after the 7th count the device proceeds to state 33 of FIG. 3D. As the device passes through state 33 the address clock enable is set to "1." Since 8 bits have passed since the first data bit was received, the operator decode 360 provides an output signal. If the output is a connect-control and the control-data line is a logic "0" or the output is a connect-data and the control-data line is a logic "1," the device will proceed to state 34. If neither of these conditions exist, the device will proceed to state 45 and await another initialization sequence. This procedure is taken because the arrival of the wrong CONNECT words at this time would indicate that either the calling or the called device is inoperative.

However, assuming that the control data lines in the operator section of the CONNECT word correspond, the device will proceed to state 34. It will stay in state 34 for 22 counts until on the 23rd count the device proceeds to state 35. As will be ascertained, during each of the recycles in state 34, the address clock enable is set to a "1" while the count is incremented. Upon reaching state 35 the 24 address bits will have been entered into the register 356 of FIG. 2 and decoded in comparator 362. The device stays in state 35 and continuously increments the counter until the count reaches 62 or 1,046. During each of these cycles the received data delimiter signal is placed in a logic "1." If the control-data line is a logic "1," the apparatus counts to 1,046 to allow the passage of the full data

message. However, if the control data line is a logic "0," this indicates a control word and only a count of 62 is required. Upon the attainment of the required count, the device proceeds to state 36 where, upon passing to state 37, the parity shift control delimiter is set to a "1" temporarily while the loop busy and count leads are placed in a logic "0." The device stays in state 37 for 22 counts while incrementing the counter until on the 23rd count the device proceeds to state 38. The 23 counts necessary to go through state 37 plus the one count for state 36 allows a parity check after the passage of the message. Thus, a parity check is made in state 38. If parity does not check, the device proceeds to state 40 where a count of 7 is completed during which time the response delimiter and opcode counter enable are each set to a one. On the 8th count the device proceeds to state 41 on FIG. 3E. If the parity does check, the device proceeds from state 38 to state 39 whereby the acknowledge connect line is raised to a "1" in addition to the response delimiter and the opcode counter enable being set to "1." Again, this state idles for a count of 7 and the device proceeds to the strap prior to state 41.

As will be ascertained, if the parity did not check, no acknowledgement would be sent to the calling party. However, since the parity did check, an acknowledgement was sent so that a further message portion could be transmitted. If the device is anything other than a MGU on the local loop side, the device proceeds to state 2 and awaits a data input signal. If, however, the device is a MGU, the device proceeds to state 41 where the opcode counter enable is set to a "1" and the apparatus waits 7 more counts before proceeding to state 43 where a prepoll is sent and it now returns to state 2.

Previously, the apparatus was followed through to state 17, in FIG. 3B, where it was assumed that the address compared. However, if the address does not compare, the device proceeds to state 10. It stays in state 10 for 53 counts. If after the 53rd count the transmit active signal is a logic "0," the device proceeds to a strap decision block above state 42 on the same sheet. Again, depending upon whether the device is a MGU or not, the device may or may not supply a prepoll output before returning to state 2.

If, upon a count of 53, the transmit active signal is a logic "1," thereby indicating that the reason the address did not compare was that a call had just been sent out and it was now being received, the device proceeds to state 11. The response field is examined by decode network 364 and if a busy signal is supplied the device will proceed to state 12 where the response delimiter is set to a one temporarily and the poll active is set to a zero. The device then proceeds to state 2 to await a new data bit. If the decode indicates that the response is not busy and if further there is no connect signal, the device will proceed through state 13 to state 2. Again, the response delimiter is set to a one.

If the response decode indicates that a connect was received, the device proceeds to state 14 and as it passes therethrough sets the response delimiter to a "1" and the acknowledge-connect lead to a logic "1." The device then proceeds to state 15 where it awaits the reception of further data. When further data is received, the opcode register shift control is set to a "1" as well as the opcode counter enable lead. The

device proceeds to state 16 where it waits the 7 counts necessary to have the operator section of the work decoded. Each cycle around stage 16 the opcode counter enable and the opcode register shift control is temporarily set to a one. After the 7th passage the device proceeds to state 18 where the operator decode signals from 360 are examined. If the output is connect-data, the device proceeds to state 19 where a total count of 1,078 is obtained before proceeding to state 20. If the operator is decoded as a connect-control signal then the device proceeds to state 23 where it waits 94 counts before proceeding to state 20. In either case, by the indicated counts upon reaching state 20 the response field is in the response register. As indicated previously, the data continuously runs through the response register 358 and may be examined at any time via the response decode block 364 by the sequence control block 352. The response decode is now examined and if it indicates that there is a negative acknowledgement, the device proceeds to state 21 where the response delimit is set to a "1" temporarily and the poll active is set to a zero before proceeding to state 2 and awaiting a new data bit. However, if the response is an acknowledge, the response delimit is temporarily set to a "1" and the acknowledge connect is set to a "1" so that the transmit control unit knows that a positive response has been obtained. The device again returns to state 2 to await new data bits. If neither an acknowledge nor a negative acknowledgement is received, the device proceeds to state 25. As the apparatus passes through state 25 the response delimit is set to a "1" and the device proceeds to state 45 to await an initialization sequence thereby indicating that the present circuit apparatus is apparently inoperative.

After reaching state 18, if the operator decode block 360 indicates that the operator is neither a CONNECT data or CONNECT control, the apparatus proceeds to state 24 and counts 1,078 before passing to state 25 and thereafter to state 45 to await an initialization sequence. Again, receipt of either a signal indicating that it is not a CONNECT block in state 18 or an improper acknowledgement in state 20 indicates that the circuitry is inoperative and requires a return to the initialization sequence to ascertain if the circuitry can be made operative on a new attempt.

To summarize, the receive module awaits the start of a message; if it is a poll or a prepoll, it ignores it other than to inform the transmitter module and awaits an actual message. If the message decoded is anything other than a call, at initial conditions, the device returns to waiting for further messages after an appropriate time interval. When a call is received and the address does not compare, and the device further knows that it is its own call previously sent, it is examined to see if the response field indicates a connect. The next reception of data should again be a word transmitted by the transmit control portion of the associated device, the device awaits the CONNECT data or CONNECT control word which it examines to make sure that it is acknowledged before returning to its initial waiting state. If the address does compare, the device checks parity, the control/data lines and whether or not the station wishes to stay in a busy condition before proceeding to receive the CONNECT word. When the CONNECT word, either data or control, is received, its

parity is checked and an acknowledgement is supplied before returning to the initial state.

TRANSMIT CONTROL MODULE

FIG. 4

The transmit control module normally remains in an idle condition and allows the passage therethrough of loop data as received from the receive control module. As previously indicated, the transmit control module delays the data by a half bit time period. This is accomplished in a J-K flip-flop 380 constructed very similar to that found in the receive control module of FIG. 2. The incoming data to the transmit control module is received on lead 382 to a sequence control No. 1, block 384. This data leaves control block 384 on a lead 386 to the J-K flip-flop 380 and leaves from the true output of the J-K flip-flop. A further input lead 388 to a sequence control unit No. 2 designated as 390 provides an indication that a buffer has been filled with data to be transmitted. Upon completing a sequence of logic steps, the control block 390 supplies an input on a lead 392 to the control block 384 that data is available to be transmitted by the transmit control module. A block counter and bit count decode block 394 is illustrated with clock, reset, and count enable inputs and with a plurality of outputs providing an output decode of count 0, 30, 32, 53, 72, 95, 100, 1,056, 1,079, and 1,084. All of these decoded outputs are supplied back to the control sequence block 384. A further output of the sequence control block 384 is BUFRS supplied to a response buffer 358 in FIG. 2. There is a response counter and generator 398 which receives an input of response count enable from block 384 and an acknowledge/connect input from the receive module of FIG. 2. The counter 398 supplies outputs of response and response count 7 to control block 384. A limit counter 400 receives inputs of limit count 1 enable and reset limit count 1 while providing an output of limit count 1. A further limit counter 2 is designated as 402 and it receives inputs of limit count 2 enable and reset limit count 2 while again providing an output indicative of limit count 2. A parity generator block generally designated as 404 receives as inputs a clock, a parity generator input, a parity generator register reset, and a parity clock enable. The parity generator holds 23 bits and provides an output to the sequence control block 384 of parity data. An opcode generator block 406 receives an input of transmit control/data (TXCD) from the control block for the data buffer or from the station interface buffer and receives further inputs of parallel load 1 or parallel load 2. The output of the opcode generator 406 is supplied in parallel to an opcode register 408 of 6 bits. The parallel load inputs to generator 406 enables the loading in parallel to register 408 of SELECT or CONNECT opcodes. An address register 410 is illustrated having several serial inputs such as a third party input, an address data input, and a gated clock input. The register 410 contains three 8-bit areas of subscriber, exchange, and area code data bit messages. Each 8-bit area or field contains a 6-bit address and a 2-bit prefix. The area code prefix is -01, the exchange prefix is -10, and the subscriber prefix is -00. Where the device being addressed is a processor rather than a subscriber, the address is -11. Although not per-

At the present time, the system may be designed to have a party line with a plurality of stations connected to a single SCU; and if so, an extension number would be included as an 8-bit address in the first CONNECT message sent to the called station as shown in the referenced application. The three portions of block 410 and a similar register 414 are drawn in different sizes for convenience although in actuality each contains an 8-bit register section.

The sequence control block 384 also supplies an input of register clock enable to 410 along with register in (REGIN). A final serial input to block 410 is a normal clock input. Further register 414 which is utilized as a third party address register, receives the same serial inputs as did register 410 and additionally receives a parallel input from intercept station address block 412. Each of the registers 410 and 414 provide an output through an OR gate 416 to the opcode register 408. The intercept address block 412 receives a serial input from control block 384 labeled intercept station address enable (ISADEN). When this lead is raised to a logic "1," the intercept station address is parallel loaded into the address register 414.

Most of the inputs to control block 384 have already been listed; but there are several more such as the MGU strap input (MGUSTP) which may be set upon production of the device. Further straps are the local/intermediate strap (LISTP) and (Red) priority strap. If the transmit control module is to be placed on the local loop side of a MGU, the LISTP is set to be a "1"; and it is set to be a "0" if it is on the intermediate loop side of the MGU. The Red strap is set to a logic "1" to provide a priority SCU unit. A further input is poll in (POLIN) for indicating to the control block not to send a poll. The control block 384 supplies an output labeled alarm (if used in a SCU) to the station interface that the set limit number of busy or NAK responses were received and indicated by limit counters 400 and 402, respectively. Another output is a transmit delimit (XMTDEL) which output is supplied to the station interface or to the data buffer control for obtaining the data to be transmitted via line 382. A buffer reset (BUFSR) output is also utilized to reset the buffer full lead 388 to its logic "0" condition. A final output of block 384 is transmit active (XMTAC) which is supplied to the receive module of FIG. 2 to indicate that a message is being transmitted and that the device is thus busy.

A further input to block 384 is data which is the input from the receive control module of FIG. 2 delayed by one-half bit in time. This data, during normal idle conditions of the transmit control module, is merely transmitted through block 384 to the data-out line 386. Another input is poll delimit (POLDEL) received from the receive module and indicating that a poll has been received from the loop. Response delimit and acknowledge-connect are two further input leads to block 384. The response delimit informs control block 384 when it should look at or send a response as indicated by the acknowledge connect line. A final input is the transmit control/data line which was previously indicated as being supplied to opcode generator 406. It should be noted that, although not previously mentioned, several of the other blocks such as 380, 390, and 394 receive clock inputs.

As indicated, messages are sent in the referenced system application from node to node or station to station along a closed loop path. In order to determine whether or not the next station is busy, a SELECT mode call is sent; and if the response section of this word is changed to a connect, the data or control word is then supplied in a CONNECT mode word. The device being called may already be connected to a given party. However, from the format used in the present embodiment of the invention, the party receiving the SELECT mode word does not know whether the SELECT call word is coming from the party to which he is connected or from a third party. Thus, he must reply with a connect so that he can either receive the data or control word. If the word is from a third party, it will be a branch call control message of a CONNECT word block. Thus, it will contain the address of the calling party. The called device will decode the branch call portion of the word and determine that it must now send a SELECT call and CONNECT mode control message back to the third party indicating that, although it accepted the SELECT word, in actuality it is busy to third party messages. Thus, the return message has the calling party's address in the called address vector portion thereof, and the reply word is the busy control message word as shown in FIG. 26 of the referenced application.

Applying this information to FIG. 4 it will be noted that the data normally is supplied directly through control block 1 (384). When the device is not connected to anyone, it responds to a call with a connect and then when it receives the control branch message the calling party's address is placed in the register 414. This information is retained so that the device can return a disconnect or supply CLCK messages back to its connected station. At the time that the calling address is stored in register 414, the response section is changed to a connect so that the calling buffer will determine upon examining the response section that the two parties may be connected. The called party then sends out a SELECT call and CONNECT CLCK to the calling party indicating a "connection."

If the device now receives a third party SELECT call, the device, not knowing that the calling party is a third party, will change the response section to a connect and await the CONNECT mode word. If by some inadvertence this CONNECT mode word were a data message, the device would, of course, accept the message as coming from the connected party rather than a third party. However, the apparatus is designed such that this will not occur. Rather, any message from a third party which was not connected to the present device would have as its CONNECT mode word the branch call control message. The address information from the branch call control message is supplied to block 414, where it will be available for sending a reply of a busy control message to the party in the next available instance.

When the transmit control module supplies output data, this information is supplied through the parity generator 404 so that the parity count can be checked and a parity code can be generated. At the appropriate time, the parity information is supplied to the message and transmitted as part of the overall message. This section of the message is referred to as a "check."

The transmit control device will receive, on occasion, a negative acknowledgement or a busy signal from a party which is being called. Each consecutive occurrence of negative acknowledgement or busy is recorded in the limit counters 400 and 402. If more than the set number of busy signals are consecutively received, the alarm lead is raised if the transmit control block is in a SCU. If the transmit control module is in a MGU, the MGU will merely issue a poll. If the transmit control module is on the local loop side nothing further will be done with the message and the calling party on another local loop will, after a predetermined amount of time, attempt to send the message again. If the transmit control section is on the intermediate loop side of the MGU, the intercept station address will be loaded into address register 414 and the message will be sent to an intercept station (not shown specifically).

The response counter 398 is utilized to count the number of bits in the response section of the received SELECT or CONNECT mode words and to generate a new response if indicated to do so.

TRANSMIT CONTROL MODULE FLOW DIAGRAM

FIG. 5

A flow diagram is presented in FIG. 5 of the block diagram of the transmit control module of FIG. 4. As may be ascertained from the following step-by-step description of the flow path of FIG. 5, the transmit control module awaits the receipt of a start bit at the data-in line and a ready-to-call indication. If the start bit is anything other than a poll or a prepoll when the device is strapped as a RED priority unit, or if there is no ready-to-call indication, the device returns to its idle state. Two exceptions to returning to the idle state immediately are if it has been asked to send a poll or if it has been asked to send a response. If it has been asked to send a poll and it is ready to call, it will send the call rather than the poll. This, of course, only happens to the transmit control module which is on the local loop side of a MGU. The call will be sent and the device will return to its idle state awaiting a return of the message to the receive control module for examination of the response return.

Returning to the situation of the device being polled and being ready to call, the transmit control module will send out the operator code and the address from the registers 408 and 410 of FIG. 4. The address register 410 was previously filled with an address by a previous CONNECT call. The parity bits are now transmitted and then zeros are transmitted while the device awaits receipt of the SELECT mode call from the next node. When the message is returned with the appropriate indication in the response section, the transmit control module either notes that it is busy and increments the busy limit counter and then, depending upon whether the transmit control module is part of a MGU, sends a poll or awaits further instructions. By awaiting further instructions it is meant that the device will await the next poll (or prepoll if RED) while still containing the ready to call indication so that upon the next poll a further SELECT call can be transmitted. If the response is a connect signal, the busy counter is set to "0" and the CONNECT mode block is assembled by

sending the data from the address and opcode registers 408 and 414 onto the output line preparatory to sending the data or control bits and then sending the parity bits. If the device is a MGU, a poll will be sent thereafter. However, for other units, a response will be awaited from the called station to see if it is negative or positive acknowledgement. If it is a negative acknowledgement the appropriate limit counter 402 is incremented. If the number of negative acknowledgements exceeds the prescribed maximum the message is sent to the intercept station if on the intermediate exchange loop and the message is dumped if on the local exchange loop. If the allowable number of negative acknowledgements has not been exceeded, the device returns to its initial state. If the response is a positive acknowledgement, the buffer full line is reset as well as the negative acknowledgement counter and it now returns to the initial state.

Keeping the above information in mind and further noting that there are two sequence control blocks 384 and 390, the flow diagrams will be described. Since the flow diagram for 390 is very short it will first be described from FIG. 5D. Upon receipt of the power-on-clear signal, control 390 awaits the receipt of a buffer full indication from the station interface module or from the buffer control unit of the MGU. When this indication is received, the apparatus proceeds from state 21 to state 23 wherein either a SELECT control call or SELECT data call opcode is parallel loaded into the opcode register 408 by setting PALDE 1 to a logic "1" condition. The device then proceeds to state 23 and places a logic "1" on the ready call lead 392 to sequence control block 384. The device then remains in this state and recirculates until the buffer full lead is changed from a logic "1" to a logic "0" by the buffer reset (BUFRS) output lead of control block 384. Upon the buffer full lead being returned to zero, block 390 returns from state 23 to state 21.

Block 384 awaits the power-on-clear signal from other circuitry in the device and when it is received goes to state 1. It stays in state 1 as long as the input conditions correspond to conditions TT. TT is expanded upon at the bottom of FIG. 5A as being the conditions of no response delimit signal and no poll delimit or no ready call signal or last signal and no send poll or no ready call signal or a poll in and no poll delimit or no last signal. As will be noted by the upper portion of the state 1 block, the TT detection keeps the data-out line directly connected to the data-in line and there is merely a direct connection through block 384 of data received from the receive module. On each circulation through state 1, the parity generator register is reset. It should be remembered as mentioned in conjunction with FIGS. 3 that the setting of any logic in the bottom portion of a state causes that indication to stay until it is changed by another state. The top portions are temporary changes and only occur upon each passage through the state. Thus, on the first passage the count in counter 394 is set to zero. However, the device will immediately start counting when it leaves state one and encounters CNTEN in later states.

When a response delimit signal is received by the device it will proceed to state 5. The response delimit output causes the connection of the output of the response generator 398 to be connected to the data

output line and for the response counter in 398 to be incremented by one. The device stays in state 5 until the counter equals seven and then returns to state 1. Basically the response delimit is to add a response to a received message for the station signifying that the device acknowledges the received message whether it is a SELECT or CONNECT message.

While the response delimit also occurs if a word is received back which has previously been transmitted, this sequence of events occurs later in the flow chart.

If the transmit control module receives both a ready call signal and a send prepoll signal while simultaneously having a logic "0" on the POLIN lead, the device will go to state 2. It should be noted at this point that the device will proceed from state 1 to state 2 only when the transmit control module is on the local loop side of a MGU. Further, it should be noted that this will occur only when an input bit is not being or has not been received. In other words, the transmit control normally has received 1 logic bit at the same time that the receive control module receives a bit and thus the device will not leave state 1 to proceed to states 4 or 24 until the occurrence of the second logic bit. The receive control module cannot provide the appropriate logic level changes on the poll delimit lines until the second logic bit. Therefore, the device will enter state 4 or 24 after the second logic bit and in preparation for working with the third logic bit. However, states 2 and 5 will be entered after the first logic bit so that the second logic bit will be operated upon in states 2 and 5. For this reason state 2 is required in the SENPPOL and RDYCAL and POLIN path in order to provide the second logic "1" bit before rejoining the POLDEL and RDYCAL and LAST branch in state 3.

Thus, when the SENPPOL and RDYCAL lines are set to a logic 1 and the POLIN path line is set to a logic 0, the device will leave state 1 while setting the data output to a logic 1 and enter state 2. As further shown in the lower portion of state 1, the count is set equal to zero and if response delimit is equal to zero the response counter is also reset to zero. The second data output bit is also placed in a logic "1" in state 2, and the device passes to state 3. In state 3 the count enable is set to a "1" temporarily as well as connecting the output of opcode generator 408 to the data output lead and also connecting the output of 408 to the REGIN for returning or recirculating the address back to register 414. In addition the parity check enable is set to a logic "1" along with the register clock enable for shifting the address and opcode out of the appropriate registers. A final setting in the top part of state 3 is connecting the parity generator to the output so that parity generation can proceed. As will be noted, the transmit active line is set equal to one in the lower part of state 3 and the device idles in state 3 until a count of 30 is obtained. When a count of 30 is reached, the device proceeds to state 7. State 2 only occurs in the transmit control module in the local loop side of a MGU. In all other units the device proceeds to state 4 upon simultaneous reception of a poll delimit and a ready call and LAST. If the device in which the transmit control module is a priority or RED unit, the output line is set equal to one. However, if it is not a priority unit the output line is connected to the input line. Thus, if the signal being received is a prepoll and the device is not a

priority unit, the second bit will remain a logic 0. If the unit is a priority unit and it has a message ready to send, it will change the second bit to a logic 1 so that a call may be consummated. If the word coming in is a call, the non-priority unit will merely pass this logic bit while the RED priority unit will write over the logic 1 with a second logic 1.

In state 4 the various settings of the top portion of state 3 are initiated if the decision block of state 4 determines that there is either a second poll delimit or a priority unit without a poll delimit. If the device is a non-priority unit and there is no poll delimit there is a return to state 1. This causes the output line to be connected to the input. Thus, a received prepoll to a non-priority unit remains as a prepoll and is passed to following devices. A priority unit upon the determination of no second poll delimit will proceed to state 3 so that a call may be made. The transfer to state 3 connects the output to the opcode generator as indicated in state 4.

If the decision block of state 4 determines there is a second poll delimit, the device proceeds to state 6. The device will exit from state 6 on the occurrence of the 4th data bit. If it exits to state 1 due to the poll delimit line being down, it will be realized that this is a call and that the output line should remain connected to the input. However, if there is still a poll delimit the device will realize that a poll has been received since the receive module produces three poll delimits in response to a poll. In exiting to state 3 from state 6 the LASTS line will be set to a 1 if the device is a RED priority unit. The setting of the LAST flag is provided so that in the event of an abortion from state 8 to state 9 due to a busy reply, the poll will still be forwarded upon the receipt of the next prepoll. If a non-priority unit exits to state 9, it will set LAST at this time and lose its turn. As will be ascertained from the description infra, the LAST indication is set after normal completion of a message only by non-priority units. The setting is not used when the priority unit is activated by a prepoll since the RED unit has utilized the system out of turn and the use of the LAST flag to change the next prepoll to a poll would prevent normal system operation.

Either a priority unit or a non-priority (RED) may be included on the path from state 6 to state 3. The device idles in state 3 until a count of 30 is decoded at which time the device proceeds to state 7. At the time that the transmit control unit reaches state 7, the 6 bits of the opcode plus the 24 bits of the address have been transmitted. In state 7 the data output lead 386 is connected to the parity data output lead of parity generator 404. The count enable is set to "1" and the parity check enable is set to "1" while the parity generator input is set to a "0." This condition remains for 23 more counts until the count reaches 53. On the 24th count the complete parity check word has been transmitted and the device proceeds to state 8. In state 8 the data output lead is placed at zero while the parity generator register is reset and the opcode register is parallel loaded from the opcode generator 406 by the setting of PALDE 2 to a logic "1." The opcode is loaded with either a CONNECT control or CONNECT data opcode. The device remains in state 8 until the response delimit line is raised to a logic "1" indicating that the receive control module has received the message back,

and it is now time for the device to determine whether the response from the call sent out is a connect or busy. If the acknowledgement connect line is a zero from the receive control module, the device proceeds to state 9 where the limit counter 400 is incremented by one indicating that another attempt has resulted in a busy. The transmit active line is returned to zero so that the device can receive further calls. If the LCNTI lead is a one, thereby indicating that the limit of busy signals has been exceeded, the device will proceed to the MGU strap decision block prior to state 18 in FIG. 5E. If it is not a MGU it proceeds to state 20 where the buffer reset is placed in a logic "1" condition, the LAST flag is set to a logic "1" and the alarm is set equal to one and the two limit counters are set to zero. The device then awaits the time necessary for the second sequence control unit 390 to return its output of ready call to a zero condition. When this occurs the device returns to state 1. If the device is a MGU when it leaves the decision block above state 18, it proceeds to state 18 where the data output lead is set to a logic "1." It then enters the decision block LISTP where a one is strapped if the transmit control unit is on the local loop side of the MGU, and a zero is strapped if it is on the intermediate loop side of the MGU. Assuming it is on the local loop side, the alarm is not raised since it is in a MGU and the device eventually returns to state 1. However, if it is on the intermediate loop side (indicated by a logic "0") the device proceeds to state 19 where the intercept station address is parallel loaded from 412 to unit 414 and the two limit counters are set to zero so that the device is now ready to send the message to the intercept station where all non-delivered messages on the intermediate exchange loop are delivered.

Returning now to the decision block below state 9 where it may be assumed that the limit of busy signals has not occurred and the device proceeds to the MGUSTP decision block. If it is not a MGU the device returns to state 1 and attempts to send the message on the next appropriate prepoll or poll. If it is a MGU, it proceeds to state 10 where a one is placed on the output lead for the purpose of sending a prepoll before returning to state 1.

Returning even further back to the decision block AKCON prior to state 9, it may be assumed that a positive acknowledgement was received from the select word previously sent. In this instance the device will proceed to state 11. In state 11 the new opcode is placed in register 408 and transmission and parity check is commenced. Leaving state 8 had placed the count at zero so the upper portion of state 11 indicates that the count zero and count one conditions places a one on the output lead and the output of the lead from register 408 is not connected for the first two counts. However, the count enable is set to a one for each of the 32 circulations. The register clock enable is set for the counts 2-32. The output lead is connected to the input of the parity generator during each of the 32 recycling operations and after a count of one, the parity check enable is set to a logic "1" so that the parity generator will receive the output message.

As will be noted, upon reaching a count of 32 the transmit delimit lead is placed in a logic "1" condition immediately before proceeding to state 12.

The device remains in state 12 until the count reaches 1,056 or 72 depending, respectively, upon whether the message to be transmitted is a data or a control word. This information is obtained by checking the transmit connect data line to see if it is a logic "1" meaning that the CONNECT block is data or a logic "0" meaning that it is a control word. On each circulation through state 12, the output line is connected to the parity generator as well as to the transmit data line 382 which is connected to the buffer control unit or the station interface buffer. Additionally, the parity check enable and the count enable leads are set to a logic "1" so that counter 394 can count the data bits out and the parity generator 404 can generate parity.

Upon reaching the proper state as determined by line TXCD and the count in bit count decode block 394, the device proceeds to state 13. Upon leaving state 12 the transmit delimit is placed in a zero condition.

In state 13 the parity bits are transmitted by connecting the output of parity generator 404 to the data output lead and placing the parity generator lead to a zero while continuing the count enable and parity clock enable leads at one. By placing the parity generator lead at a zero, the parity generator is automatically cleared by receiving zeros at the input. When the count reaches 95 or 1,079, depending upon the lead TXCD, the device proceeds to the MGUSTP decision block immediately below state 13. If the device is not a MGU, it proceeds directly to state 15 where it awaits return of a transmitted CONNECT block to determine the response. If it is a MGU it proceeds to state 14 where zeros are transmitted until a count of 100 or 1,084 is reached at which time a prepoll is sent out by placing data out to a one. As will be noted, each circulation thereof the count enable is set to a one. Upon reaching either of the prescribed counts, the device will proceed to state 15 where it awaits the return of the transmitted CONNECT block to determine the response.

The reason for sending out the prepoll from the MGU in state 14 is to allow the next device in the loop on the intermediate loop side to receive a poll and transmit a word or on the local loop side to allow the next appropriate device on the local loop to receive a prepoll and begin the process of transmitting a word.

The device stays in state 15 until the response delimit is set to a one. During this time the data out is placed at a zero and the parity reset enable is placed at a one. Further, the counter is set to a zero. When the response delimit is raised to a one, the device proceeds to the decision block AKCON. If the response is negative acknowledgement, the limit counter 402 is incremented and the transmit active is set to zero in state 17. The device then checks to see if the predetermined limit has been reached. If it has been reached it proceeds to the MGUSTP strap in FIG. 5E prior to state 19; and if it is not a MGU, proceeds to state 20 where it proceeds as previously indicated. If it is a MGU it proceeds to state 19 and continues as previously indicated by sending the message to the intercept station. If the predetermined limit has been reached by counter 402, the unit proceeds to state 1. If, however, the acknowledgement is positive, the limit counter 402 is set to zero in state 16. Transmit active is also set to a zero and the buffer reset along with LASTS are placed in a logic "1" condition. As soon as the sequence con-

trol block 390 changes ready call to a zero, the device returns to state 1 so that it may respond to other start bits for transmitting data.

From the above flow diagram description it should be apparent that the transmit control module supplies prepoll and polling bits, response sections of messages acknowledging receipt of a word from a calling buffer or node and supplies SELECT and CONNECT messages in responding to a "connected" device. Additionally, it supplies busy responses to third party calls after determining the calling party's address.

MNEMONIC LIST FOR TRANSMIT MODULE OF FIGS. 4 AND 5

ADDATA Address data — Data received from the receive control module containing third party addresses or received from station interface buffer for address of party being called.

ALARM Alarm — Informs station interface that more than the predetermined limit number of busy or NAK responses were received; this is an output signal.

BUFFUL Buffer full — Informs transmit control module to place a SELECT call MESSAGE on the loop. This is an input from buffer control or station interface.

BUFRS Buffer reset — Instructs the buffer control or station interface to reset BUFFUL to logic "0."

CNTEN Count enable — Line used to enable block counter 394.

CNT(No.) Count decoded output — The count number is a specific lead from the bit decode unit 394 indicating that a specific numerical count has been reached. This is an output from 394 to sequence control 384.

DATOUT Data out — An output from control unit 384 to flip-flop 389 containing the data to be provided back to the loop.

DATIN Data in — Input data from the receive module to sequence control 384.

GTDCCLK Gated clock — Clock signals from the receive control module for shifting an address into the address register 410. (NOTE: The clock from the receive module may be slightly different from the clock used to send data to the loop and thus two different clocks are used in the transmit module.)

ISADEN Intercept station address enable — The line used to parallel load the intercept station address from register 412 to register 410.

LAST Flag set by last non-priority unit to successfully send a data message.

LASTR Line leading to LAST flip-flop for resetting the flag to a logic 0 condition.

LASTS Line leading to LAST flip-flop for setting the flag to a logic 1 condition.

LCNT Limit count 1 or 2 — The lines from limit counters 400 and 402 to the sequence control 384 for indicating that a predetermined set limit for busy signals or negative acknowledgement (NAK), respectively, have been reached.

LC1EN Limit count 1 enable — Enabling lead from control unit 384 to the busy counter 400.

LC2EN Limit count 2 enable — Enabling lead from 384 to negative acknowledgement counter 402.

LISTP Local-intermediate strap — This is an input supplied to control block 384 whose value is dependent upon the use for the transmit control module. The input is a logic "1" if the control module is used for the local loop side of a MGU and is a logic "0" if used for the intermediate loop side of a MGU.

MGUSTP Master group unit strap — This is an input to control unit 384 which remains constant for a particular transmit control module and is placed at a logic "1" if the control module is used in a MGU and is placed at a logic "0" if it is used in any other device.

POLIN Poll in — This is an input from the LDI indicating that a poll has just been received.

PARRSE Parity generator register reset — Lead from 384 to parity generator 404 to reset the generator to zero.

PARCKE Parity clock enable — Input from 384 to 404 containing the clock utilized by the parity generator for shifting data inputs.

PARGEN Parity generator — This line provides data input to the parity generator. When operative this lead is connected to the DATOUT line 386.

PALDE1 Parallel load enable 1 — Lead for directing opcode generator 406 to parallel load a SELECT control call or a SELECT data call opcode into the register 408.

PALDE2 Parallel load enable 2 — Line to generator 406 for parallel loading a CONNECT control or CONNECT data opcode into register 408. The determination whether the above two inputs will provide control or data opcodes depends on input TXCD also supplied to generator 406.

RED A strapped line on the unit which sets a line to a logic 1 for obtaining priority (a non-priority unit would have a logic 0 on this line and would be defined as RED).

RDYCAL Ready call — Input from control 390 to control 384 indicating that the data buffer is full and that the control unit 390 has provided the proper opcode to register 408.

REGDAT Register data — Output line from register 408 to control unit 384 for supplying thereto the opcode and address.

REGCKE Register clock enable — Output from control 384 to the address registers 410 and 412 for enabling these registers to accept an address from the receive module, the buffer control, or the station interface.

REGIN Register in — Input from control module 384 for circulating the opcode and address register from the output REGDAT of this portion of the transmit control circuitry and back to the input to maintain the proper address therein.

RESDEL Response delimit — Informs block 384 when to look at or send a response as indicated by the AKCON line.

RELCT Reset limit counter 1 or 2 — An input for resetting the respective limit counter after the transmit control makes a successful call or takes the procedural steps of setting an alarm or sending information to an intercept register when the predetermined limit count is reached.

RESP Response — Response register line from 398 containing the proper response data for insertion into the response section of a select or control word.

RESNTE Response counter enable — Line from control block 384 to response counter 398 to enable this block for commencing the count.

SENPOL Send poll — Input from the receive control module to the sequence control 384 for informing the transmit control module to send the poll on the loop.

TXDAT Transmit data — Output from flip-flop 380 for transmission to the loop connected to this particular control module.

TXDATA Transmit data — Data received from the buffer control or the station interface buffer by the control unit 384. In most instances this data is passed through control unit 384 but on occasions of transmit this data (on lead 382) is open-ended.

TXCD Transmit control/data — This line is a logic "1" if the CONNECT block is data and is a logic "0" if the CONNECT block is control. This line also controls whether the select block is a call or a poll.

THDPTY Third party — This line is obtained from the receive control module and indicates if the call is to be placed to a third party. Basically, this line activates receipt of an address by register 414 and deactivates register 410 from receipt of an address on the ADDATA line.

XMTDEL Transmit delimit — A delimit line from control block 384 to the station interface or buffer control for obtaining the data on line 382.

In summary of the information presented supra, it will be observed that the present inventive concept is designed for use with a communication system wherein a master unit is connected to a closed loop communication line containing a plurality of slave units, the design of the system being such that it is necessary that the control unit be able to regain control of message transmission to units on the loop in short order, it being further desirable that the slave units be required to transmit only in response to poll signals from the master unit and transmit messages in consecutive order according to the serial connection of the slave units in the loop.

The above is accomplished by having the master unit send out a prepoll signal initially and when this is returned as a prepoll, the master unit sends out a poll. The slave units will only respond to a poll signal in the transmission of messages. Thus, upon receipt of a poll, a slave having a message to transmit will so transmit its message. The slave units are designed such that an internal flag is set upon transmission of a message. This flag will cause it to change the next prepoll received to a poll so that at this time only slave units situated further down the loop may receive the poll signal and send a message. The master unit will note the end of a message and if it has no messages to transmit will send a new prepoll. Thus, the last slave to transmit will change this to a poll and it will be forwarded to further slaves which have messages to transmit. Any further slave so responding will again set its integral flag so that the process may be continued after transmission of a message.

If the master, upon noting the end of a transmission in the loop, has a message to send, it will send the message (or messages) and at the end of such time will then send new prepoll signals.

The inventive concept also includes the possibility that it may be desirable that some slaves have priority over other slaves. This may be accomplished by allowing some of the slaves to respond to either a prepoll or a poll in the transmission of messages. If the priority slave responds to a prepoll, the internal flag is not set since it has not reacted in consecutive order. However, if it has transmitted in response to a poll, the internal flag must be set so that the poll can be allowed to proceed around the loop according to system design. In order that the design concept be complied with, it is desirable that priority slave units have a means for preventing reaction to a prepoll in the form of transmitting a message immediately after transmitting a message initiated by a poll. Rather, its reaction in this instance must be merely to change the prepoll to a poll so that further units on the loop will be allowed to transmit messages. However, later prepolis may be responded to normally by the priority slave units.

While a specific embodiment of the present invention has been disclosed, it is to be realized by those skilled in the art that various other implementations may be originated to accomplish the design concept of a master unit and consecutively operated slave units and the additional idea of allowing some of the slave units to have priority. Thus, we wish to be limited only by the scope of the appended claims.

We claim:

1. The method of providing orderly data communication on a series connected data loop having a master station and a plurality of slave stations which slave stations may only send data messages upon receiving a poll signal comprising, the steps of:

giving priority to data messages originating from said master station immediately after any pending data transaction;

sending a prepoll signal around said loop from said master station after each data transaction; and

changing the prepoll signal to a poll signal in the last station to respond to a poll signal so that the poll signal will be forwarded to the next succeeding station on the loop.

2. The method of claim 1 comprising the additional step of changing any prepoll signals received at said master station to poll signals.

3. The method of providing orderly data communication on a series connected data loop having a master station and a plurality of slave stations wherein said slave stations comprise priority and non-priority units and further wherein said non-priority slave station units may only send data messages upon receiving a poll signal and wherein said priority slave station units may send data upon receiving either a prepoll signal or a poll signal comprising, the steps of:

giving priority to data messages originating from said master station immediately after any pending data transaction;

sending a prepoll signal around said loop from said master station after each data transaction; and

changing the next received prepoll signal to a poll signal in any station responding to a poll signal in

attempting to transmit a data message so that the non-priority stations are allowed to send data messages only in their serial connected order on the loop.

4. Slave station apparatus for connection in a transmission loop over which prepoll, poll and call and data messages are sent, said loop including a master control station, said slave station comprising, in combination:

first means for detecting and distinguishing between received prepoll signals, poll signals, messages for the slave station, and messages for other stations;

second means for acknowledging receipt of messages for the slave station;

third means for sending messages upon receipt of a poll signal; and

fourth means for sending a poll signal only in response to the prepoll signal immediately following a poll to which the slave station reacted by sending a message.

5. Apparatus as claimed in claim 4 wherein said slave station permits unchanged passage therethrough of messages destined for other stations and wherein said

third means includes means for also sending a message upon receipt of a prepoll signal.

6. Message transmission apparatus comprising, in combination:

master means for transmitting and receiving messages, for transmitting prepoll signals after reception of each message when no messages are present for transmission, and for transmitting poll signals after reception of prepoll signals;

a plurality of slave means, each for receiving messages, for transmitting messages only in response to received poll signals, and for changing the next received prepoll signal to a poll signal after transmission of a message in response to a received poll signal; and

means for serially connecting said master means and said plurality of slave means in a closed loop.

7. Apparatus as claimed in claim 6 wherein:

at least one of said plurality of slave means additionally responds to prepoll signals by transmitting messages on the closed loop and thereby has priority over other slave means.

* * * * *

25

30

35

40

45

50

55

60

65