

BMAC™ Device Hardware Design Guide

National Semiconductor
Application Note 689
David Brief
February 1990



Introduction

The BMAC device provides the basic facilities required to implement the Media Access Control functions required by the ANSI X3T9.5 FDDI MAC standard. This document describes how to use the BMAC device to implement the MAC level functionality required by stations in an FDDI network.

A short overview of the BMAC device is given followed by a discussion of design considerations and tradeoffs for using the BMAC device.

Familiarity with the ANSI standard and BMAC Device Datasheet is recommended before using this document.

The state machines and timing shown in this document provide illustrative examples only. Should there be a discrepancy, the BMAC Device Datasheet is the overriding authority.

Table of Contents

- 1.0 OVERVIEW
- 2.0 CONTROL INTERFACE
- 3.0 PHY INTERFACE
- 4.0 MAC INTERFACE
- 5.0 BRIDGING/EXTERNAL MATCHING SUPPORT

1.0 Overview

The BMAC device interfaces to one or more PLAYER™ components, a Control Bus and to external logic that tunes the MAC interface to the requirements of the system.

The BMAC device is comprised of the Ring Engine, the PHY Interface, the Control Interface and the MAC Interface as shown in *Figure 1*.

The BMAC device utilizes a full duplex, byte wide (symbol pair) architecture. There are two bytes of delay in the transmit path, and three bytes of delay in the receive and repeat paths. Two bytes of delay are present in the loopback path.

1.1 RING ENGINE

The Ring Engine implements the FDDI MAC protocol for transmitting, receiving, repeating and stripping frames in a

ring of stations. CLAIM, BEACON and Void frames are generated by the Ring Engine when appropriate. The Ring Engine transmits, strips or repeats Protocol Data Units (PDUs, i.e., Tokens and Frames) and handles the token management functions required by the timed token protocol in accordance with the FDDI MAC standard.

On output (to the ring), interface logic prepares one or more frames for transmission and requests a service opportunity. Based on the requested service class and requested token type, the Ring Engine waits for a token meeting the requested criteria. When a token is captured, the Ring Engine signals the interface and soon thereafter transmission begins. After traversing the ring, frames are stripped based on the Source Address. Frames with a Source Address matching one of the station individual addresses are stripped by the Ring Engine. Status is available at the MAC interface for every transmitted frame.

On input, the Ring Engine sequences through the incoming byte stream, comparing against the station short or long addresses. The results of these comparisons are made available at the MAC Interface. Interface logic then decides how to handle the frame. In the normal case, a frame with a Destination Address matching one of the station addresses is copied and passed to the system.

1.2 CONTROL INTERFACE

The Control Interface implements the interface to the Control Bus by which to initialize, monitor and diagnose the operation of the BMAC device. The Control Interface of the BMAC device is identical to the control interface of the PLAYER device. Typically, all of the BMAC and PLAYER devices within a station will all be addressable on a shared Control Bus.

1.3 PHY INTERFACE

The PHY Interface provides a byte stream to the PLAYER device (PHY Request) and receives a byte stream from the PLAYER device (PHY Indication). The Configuration Switch in the PLAYER device allows the BMAC device to be switched into the Primary and Secondary rings as desired.

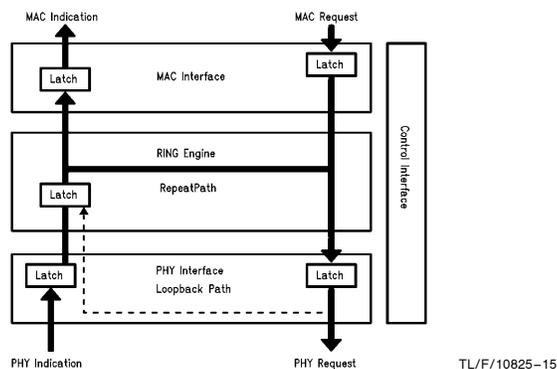


FIGURE 1. BMAC Device Block Diagram

TRI-STATE® is a registered trademark of National Semiconductor Corporation.
BMAC™ and PLAYER™ are trademarks of National Semiconductor Corporation.

1.4 MAC INTERFACE

The MAC Interface provides the interface to external buffering and control logic. A byte stream is provided to interface logic with appropriate control signals (MAC Indication), and a byte stream is provided to the BMAC device with appropriate handshake control signals (MAC Request).

It is the job of the external interface logic to transform the byte streams to and from the BMAC device to implement the data services required by the system interface.

2.0 Control Interface

The Control Interface provides an 8-bit asynchronous interface to the Control Bus. Through the Control Interface, access is provided to all internal registers. These registers control the operation of the BMAC device as described below. The interface to the Control Bus is identical to that of the PLAYER device.

The Control Interface provides the synchronization between the asynchronous Control Bus and the synchronous operation of the Ring Engine. The Ring Engine is a synchronous device running exclusively on the 12.5 MHz Local Byte Clock and the in phase 25 MHz Local Symbol Clock.

FDDI components within a station will typically share the same Control Bus. The processor on this bus may be a dedicated microcontroller that is running time critical portions of SMT (i.e., CMT), or it may be a more general purpose microprocessor which gets called on to handle interrupts.

The Control Interface is separated completely from the MAC Interface in order to allow the data paths to run at full speed and not be shared with the slower control transfers. Access to registers may occur simultaneously with the data transfer.

All communication that is not synchronized with the (high speed) data services uses the Control Interface. For example, error conditions are reported through the Control Interface, while frame reception status is reported at the MAC interface synchronized with the data stream. Likewise, options for frame transmission that can change with every frame are submitted to the interface with every frame.

All events are latched in condition latch registers, and may generate interrupts. Only enabled events may generate interrupts. Events are reported through a two level logical hierarchy. Events are grouped into classes according to their probable usage. Each event of a class may be enabled individually in the mask registers and event classes are enabled via mask registers at the Interrupt Register.

Conditions are used to signify that an event or series of events has occurred. The Interrupt signal becomes active to notify the managing entity that a condition or set of conditions exist.

All of the events suggested by the MAC standard are reported by the BMAC device. In addition, the BMAC device provides events on each counter increment and overflow.

The Control Interface also manages access to shared registers. Certain Status and Parameter Registers are not accessible while in Run mode because the Ring Engine may be accessing those locations. All Control Interface accesses are checked against the current operational mode to determine if the register is currently accessible. If not currently accessible, the Control Interface access is rejected (and re-

ported in an Event Register). This means that all Control Interface accesses complete in a deterministic amount of time. This can be useful in the design of certain interfaces to simplify synchronization circuitry. The Exceptional Status Register must be checked to insure that the operation terminated normally.

The registers accessible through the Control Interface maintain the Operation, Event, Status and MAC Parameters.

The Operation Registers are used to control the operation of the BMAC device. The Operation Registers include the Mode, Option and Function Registers. The Mode Register determines the current operational mode (run/stop, loop-back, etc.). The Option Register determines how the BMAC device will work while in run mode. The Function Register initiates functions and provides polled status on their completion. These include a Software Master Reset, a MAC reset and the CLAIM and BEACON requests.

The Event Registers are used to report the occurrence of events. The Event Registers are used to generate interrupts when selected conditions occur (under program control).

The Status Registers are used to access either current status or long term status from event counters.

The Parameter Registers are used to set up parameters used by the Ring Engine such as the station's address and the group address maps.

3.0 PHY Interface

The PHY Interface provides the data paths to connect the BMAC device to one or more PHY Layer devices. The interface is synchronous; with every rising edge of the Local Byte Clock ten bits are transferred in each direction—from the BMAC device to the PLAYER device on the PHY Request Interface and from the PLAYER device to the BMAC device on the PHY Indicate Interface. The elasticity buffer of the PLAYER device removes the asynchronous element from the data stream and allows the BMAC device to work as a synchronous device with synchronous interfaces.

One BMAC device may drive one or more PLAYER devices while only one PLAYER device should be driving the BMAC device at any given time. TRI-STATE® control and pullups are provided within the PLAYER device to allow several PLAYER devices to share one PHY Indicate Bus. This is very important in most Dual Attach and many Concentrator configurations.

FDDI uses a 4B/5B symbol encoding scheme that is normally byte aligned. The PLAYER device aligns starting delimiters of all PDUs to the byte boundary. This allows the BMAC device to detect the JK starting delimiter as a single code point and greatly simplifies the alignment issues. The BMAC device aligns delimiters of all transmitted or repeated PDUs to the byte boundary.

The 10 bits transferred with each clock consist of 8 bits of data and one bit of control to say if the data represents a data code point or a control code point. Mixed data/control symbol pairs are encoded to a set of control code points with the data being given as zero regardless of the actual data symbol value. Odd parity is also provided with every byte.

Through parity is important in this path because every station's data goes through this path. Parity errors detected in the MAC are treated as all other code violations (format errors).

4.0 MAC Interface

4.1 OVERVIEW

The BMAC device is partitioned at the MAC Interfaces to allow a full spectrum of system interface organizations. We discovered early on that it would be difficult, if not impossible to develop an interface that met cost and performance goals for all potential uses of FDDI. However, in crafting the MAC Interface, we did not just guess what a good interface would include. We modeled a complete System Interface Architecture including a real time multi-tasking operating system at the register transfer level on top of the MAC under the assumption that all lower performance interfaces would be some subset of a very high performance interface architecture.

The MAC Interface provides independent Indication (receive) and Request (transmit) Interfaces. Separate signals for control and data are presented at the interface to allow overlapping/pipelining of data and control (status/command) processing. A byte stream is transferred in each direction. On input, the MAC Indication Data byte stream (MID(7:0)) is handled by interface logic using the provided sequencing and status signals. On output, the MAC Request Data byte stream (MRD(7:0)) is generated by interface logic and passed through the BMAC device to the ring on a service opportunity.

The interface is synchronous using the 12.5 MHz Local Byte Clock (LBC). All Outputs change and all Inputs are latched on the rising edge of LBC.

The remainder of this section describes the structure of the MAC Interface.

4.2 MAC INDICATION INTERFACE

Overview

Every byte of all incoming frames is presented at the MAC Indication interface. The MID byte stream is effectively a three byte time delayed version of the PID byte stream. Sequencing signals and addressing flags are provided to help make the decision of whether or not to (continue to) copy the frame.

The Sequencing Signals are asserted at different points within the frame. They are asserted under the following conditions:

RCSTART	when the Starting Delimiter is present on MID
FCRCVD	when the Frame Control Field is on MID
DARCVD	when the last byte of the DA is on MID until the next JK
SARCVD	when the last byte of the SA is on MID until the next JK

INFORCVD when the fourth byte of the Info Field is on MID until the next JK

Not all of the signals would be used by a typical implementation. The sequencing signals are reset on a Master Reset.

One of these Termination Event signals is asserted at the end of every PDU as described below:

EDRCVD	when the Ending Delimiter of a frame is on MID until the end of the Frame Status (typically asserted for two byte times)
TKRCVD	when the Ending Delimiter of a token is on MID for one byte time
FRSTRP	when the first Idle byte of a stripped frame is on MID
FOERROR	when the byte with the format error is on MID
MACRST	when a MACRST occurs or BMAC device is in Stop Mode

The Flags provide the input for potential copy criteria and status breakpoints as follows:

AFLAG	Internal DA Match (Short/Long: FCSSL); Individual/Group: DIAG): Valid with DARCVD.
MFLAG	INTERNAL SA Match (Short/Long: FCSSL); Valid with SARCVD
SAMESA	SA Same as in previous frame: Valid with SARCVD on non-MAC frames.
SAMEINFO	First four bytes of Info same as in previous frame; Valid with INFORCVD on MAC frames.
VDL	Valid Data Length; Criteria—more than the minimum number of bytes and an integral number of symbol pairs: Valid with EDRCVD.
VFCS	Valid FCS: Criteria—received FCS matches with standard CRC polynomial: Valid with EDRCVD.

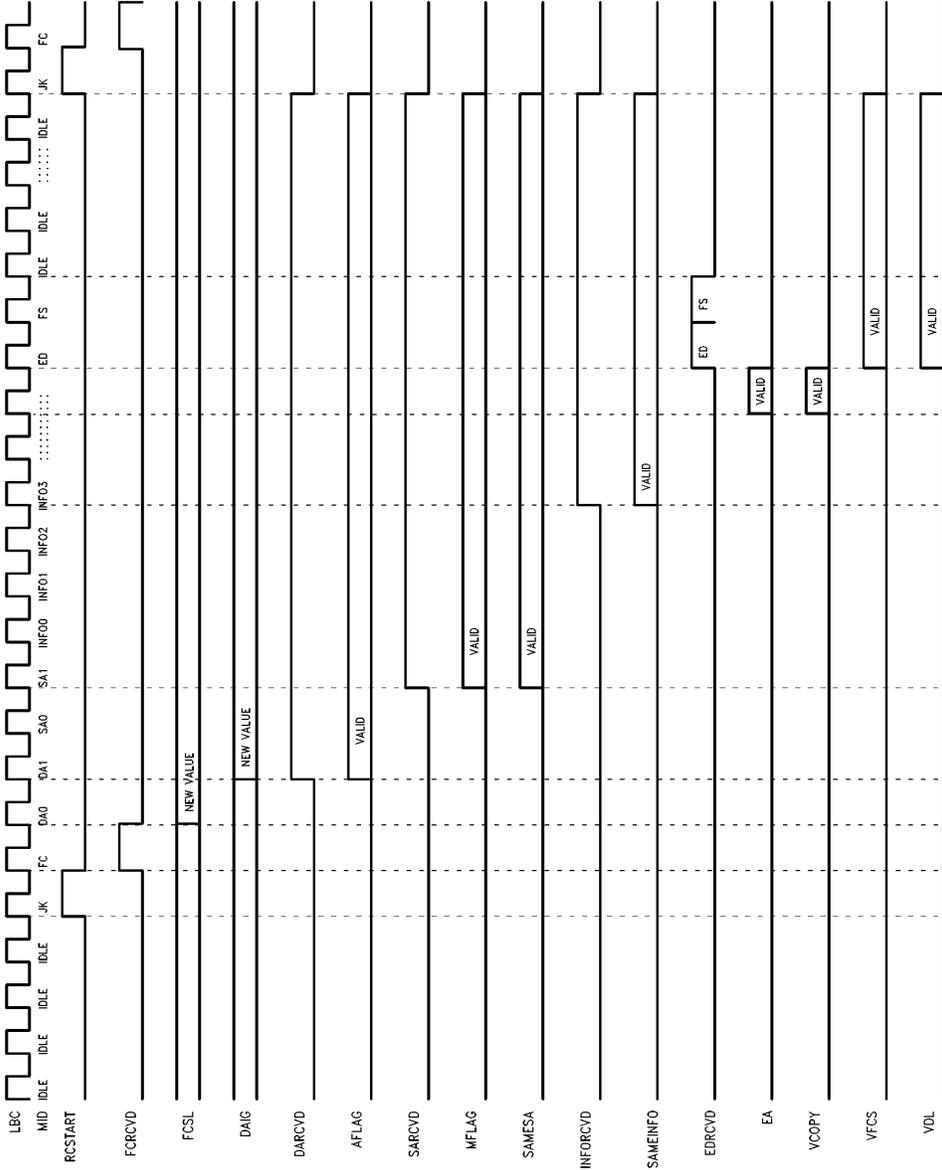
Note: The Flags are only valid when the corresponding sequencing signal is also set.

For setting the outgoing control indicators, the interface accepts the following:

EA	For external address matches for the setting of the A Indicator (Bridging Group addressing, Aliasing)
VCOPY	For the setting of the C Indicator

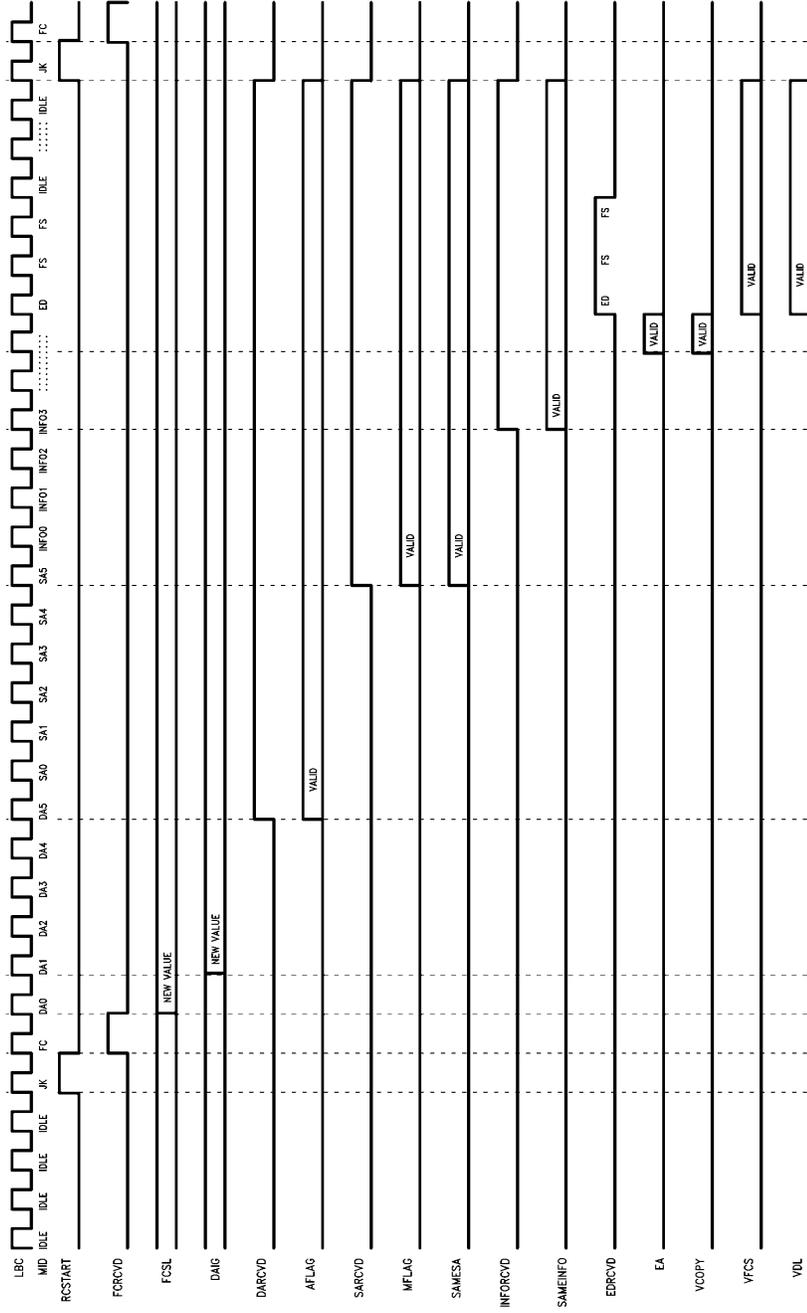
When AFLAG or EA is set and the EMIND bit of the Option Register is set, the frame copied counter or the frame not copied counter will be incremented depending on the value of VCOPY for all repeated frames.

Ten MAC Indication Timing Examples are shown in *Figures 2–11*.



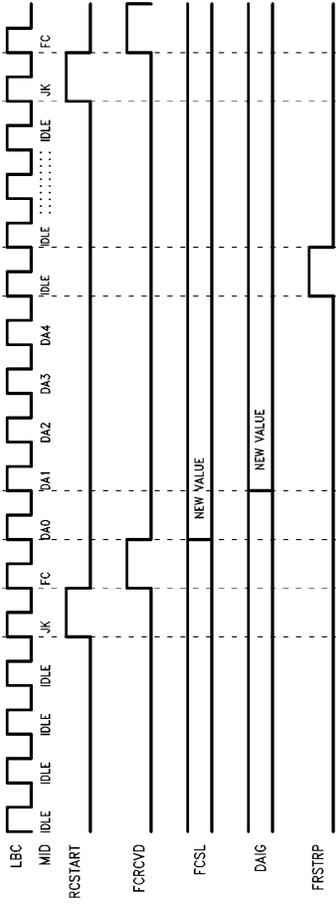
TL/F/10825-1

FIGURE 2. MAC Indication Timing Example—Short Frame



TL/F/10825-2

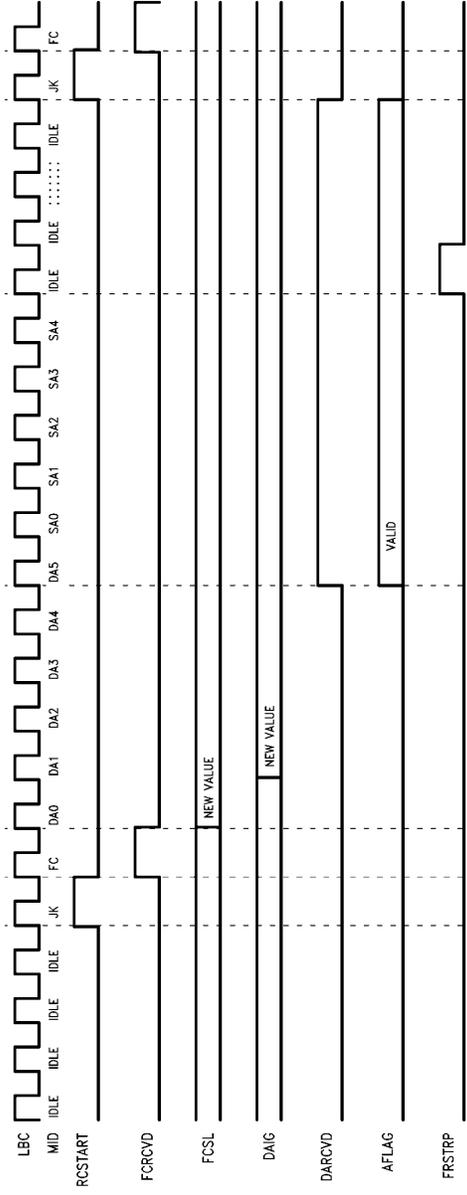
FIGURE 3. MAC Indication Timing Example—Long Frame



TL/F/10825-3

Note: FRSTRP may be asserted any time after RCSTART; All signals are held at their current value until the next RCSTART.

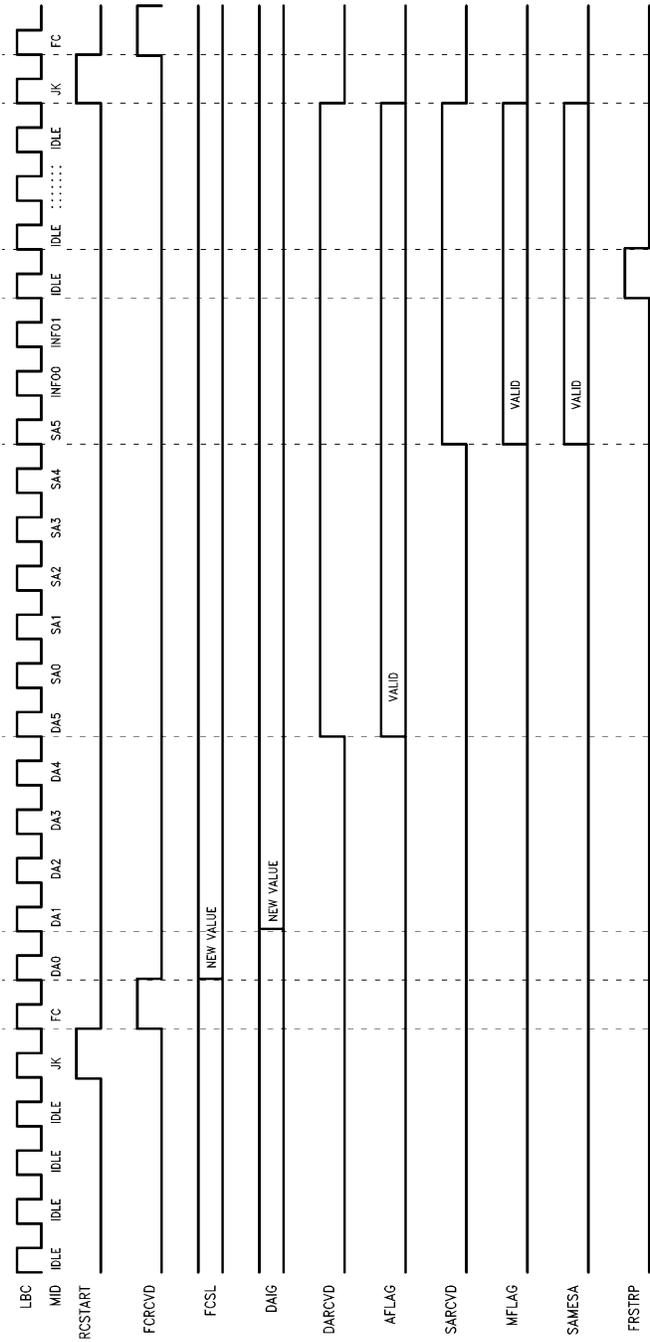
FIGURE 4. MAC Indication Timing Example—Stripped Frame before SA



TL/F/10825-4

Note: FRSTRP may be asserted any time after RCSTART; All signals are held at their value until the next RCSTART.

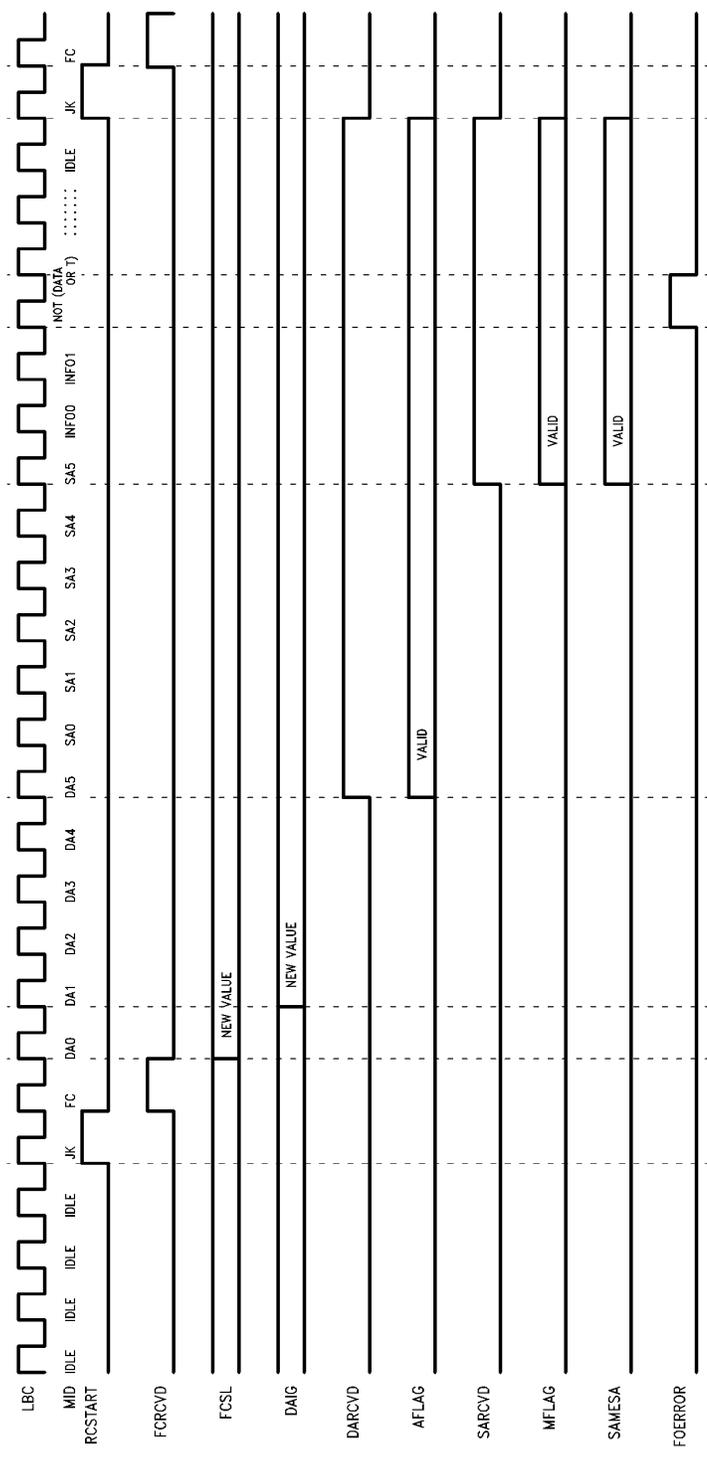
FIGURE 5. MAC Indication Timing Example—Stripped Frame during SA



TL/F/10825-5

Note: FRSTRP may be asserted any time after RCSTART; All signals are held at their value until the next RCSTART.

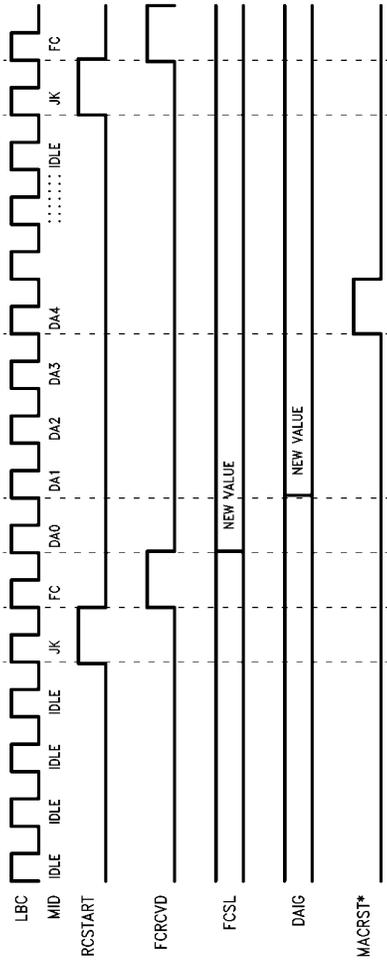
FIGURE 6. MAC Indication Timing Example—Stripped Frame after SA



TL/F10825-6

Note: FOERROR may be asserted at any time.

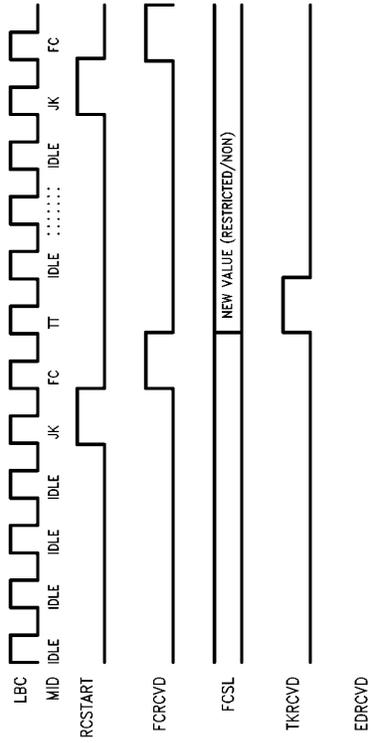
FIGURE 7. MAC Indication Timing Example—Format Error



TL/F/10825-7

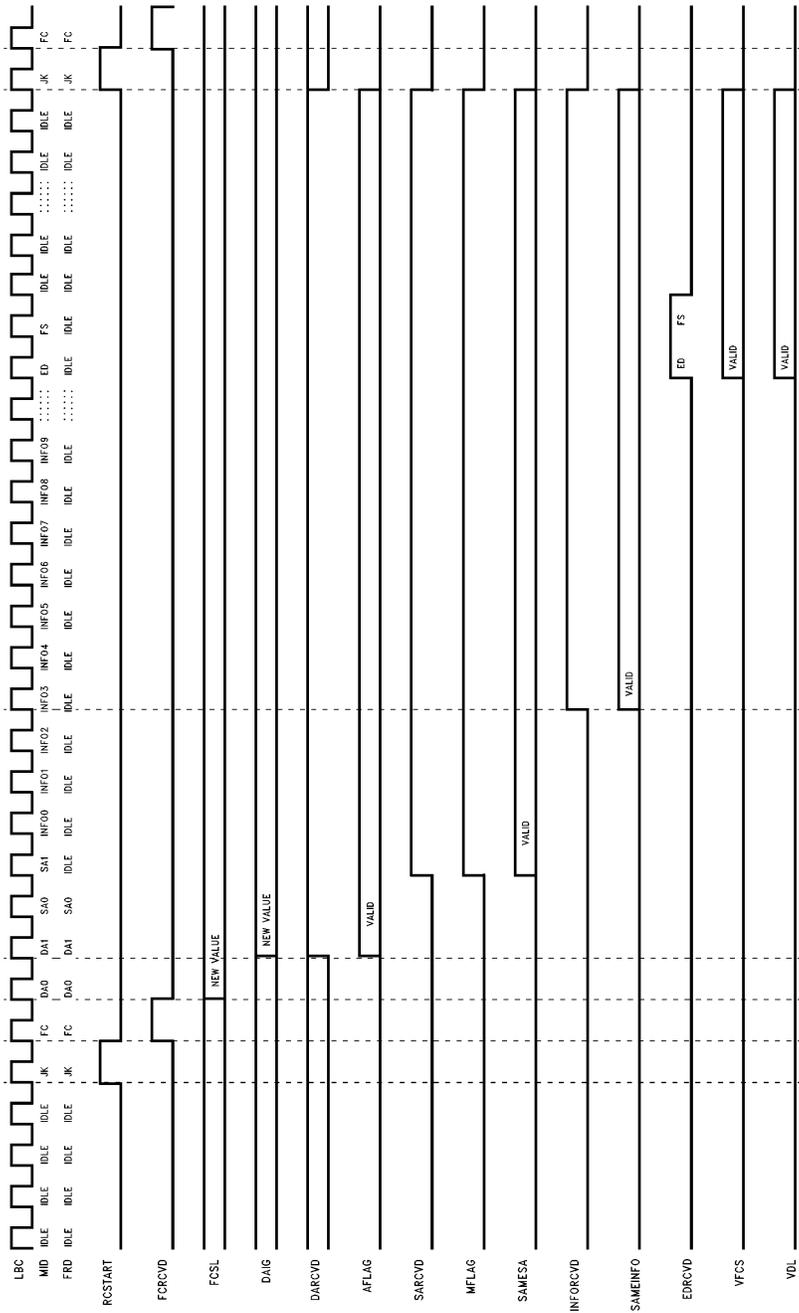
Note: MACRST may be asserted if Mode Run = 0.

FIGURE 8. MAC Indication Timing Example—MAC Reset



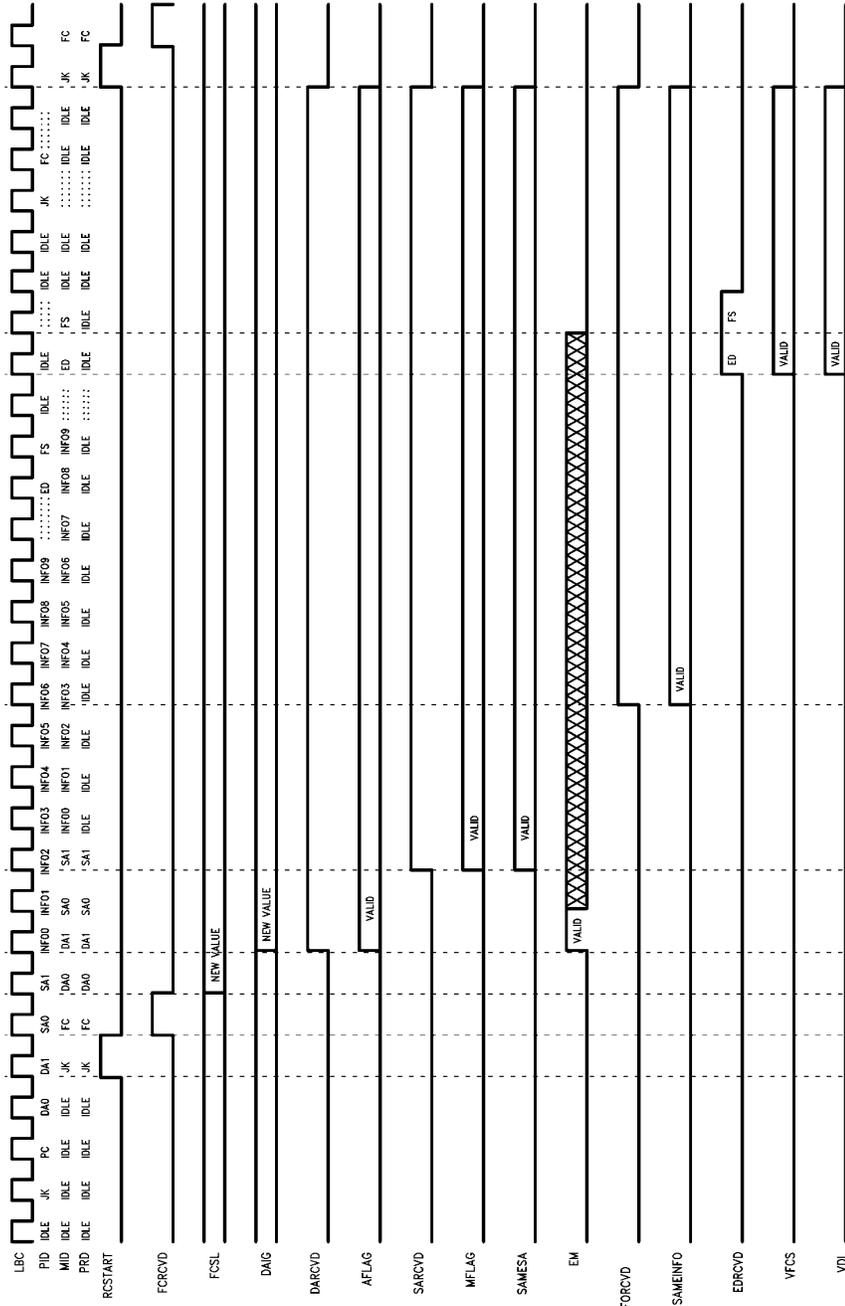
TL/F/10825-8

FIGURE 9. MAC Indication Timing Example—Token Reception



TL/F/10825-9

FIGURE 10. MAC Indication Timing Example—Internal Stripping based on MFLAG



TL/F/10825-10

Note: EM may be asserted at any time; Stripping begins 3 byte times after EM is asserted.
FIGURE 11. MAC Indication Timing Example—External Stripping based on External MFLAG

COPY CRITERIA

The decision to copy or not copy a frame is made by external hardware that looks at flags from the BMAC device qualified by the sequencing signals. The Copy decision may also use inputs from additional address matching logic. The Copy Criteria may be different for frames with different FC values. For example, SMT frames might only be copied on the basis of internal address matches while LLC frames might be copied on the basis of external matches as well. When using different frame copying criteria for different FC values, the copy logic must latch the relevant bits of the FC, or alternatively map the FC, or other fields within the frame, to one of several groups, each of which share the same copy criteria. The frame copying criteria might also be programmable in each group. The BMAC device thus provides the mechanisms by which to create very complicated or simple copy criteria.

A simple copy criteria might be as follows:

All Frames AFLAG and not MFLAG (so that you don't copy frames that you sent, i.e., broadcast and multicast)

A more sophisticated copy criteria might have different copy criteria for each of several groups. In this example, groups are differentiated by the FC.

MAC	Not (SAMEINFO and SAMEFC)
SMT	AFLAG or MFLAG
LLC Synch	Short and AFLAG and Not MFLAG
LLC Asynch	Long and AFLAG and Not MFLAG
Reserved	Do not copy
Implementer	Short and external address match

The copy criteria for MAC frames allows a station to skip copying MAC frames with the same (first four bytes of) information and the same FC value.

RECEIVE SEQUENCER

An example receive sequencer is shown in *Figure 12* for the Indicate byte stream.

RS0: Idle

Remain in the Idle state while not copying any data or writing any status. Leave Idle when there is a place to put an incoming frame and the beginning of a frame (RCSTART) is received. At this point the Stage state is entered.

RS1: Stage

Remain in the Stage state while deciding whether or not to copy a frame. The decision is made at the commit point. The commit point occurs any time after both the relevant address comparisons are complete, and after the point in the frame where it is not considered a legal fragment (4 bytes into the Information Field, when INFORCVD is asserted). At the commit point the results of the address comparisons as indicated by the Flags from the BMAC device are compared against the copy criteria. If the criteria matches and no frame termination event has occurred, the decision is made to continue copying the frame and the Copy state is entered. Otherwise, the Idle state is entered and the rest of the frame is not copied.

RS2: Copy

Remain in the Copy state while copying a frame into memory or temporary storage. Once committed to copying, continue copying until a Termination Event (TE) occurs. Upon a TE, status is written. For every frame or partial frame copied, there should be a place to record status.

RS3: Status

Remain in the Status State while Status is being written. After Status has been written, return to the Idle state.

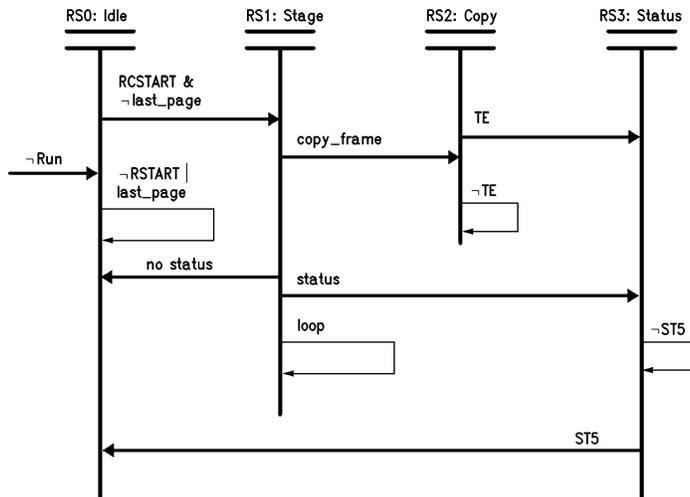


FIGURE 12. Example Receive Sequencer

TL/F/10825-16

CONDITIONS

- $TE = \neg TE \ \& \ (OutofSpace \ | \ EDRCVD \ | \ FRSTRP \ | \ FOERROR \ | \ MACRST \ | \ TKRCVD)$ {this is the termination event}
- $copy = \neg TE$ and copydecision and commit_point {copydecision is result of copy criteria}
- $status = TE$ and copydecision
- $nostatus = \neg copydecision$ and $(TE \ | \ \neg TE$ and commit_point)
- $write_status = TE$ and $((RS1$ and copydecision)|RS2 | RS3)
- $last_page = \{nospace$ condition}
- $SCMP = \{status$ recording is complete}

COMMIT LOGIC

The decision to copy or not is made early in the frame in order to minimize the bus bandwidth used and to minimize the amount of temporary buffering (FIFOing) required.

Status is written at the end of every committed frame, even if an error occurs late in the frame. Since FDDI provides a very reliable transmission system, almost all frames (99.99%+) will be received without errors. In this way all committed frames are handled in exactly the same way.

With this scheme the hardware doesn't handle any exceptions in this time critical and performance sensitive area. It could be a real challenge to recover space on the fly by backing up pointers as when trying to guarantee that back to back frames with minimum preamble can be copied.

RECEPTION STATUS

For each frame received, the following status could be written.

Frame Status [8-Bits]

- Values of received control indicators E, A and C_ind, for each report values of [R, S, T, other] {two bits per indicator}
- Value of FCS check at this station
- Result of Valid Data Length check

Frame Attributes [8-Bits]

- DA Match—Short/Long Ind/Group
- SA Match—Short/Long
- External Match Criteria
- Terminating Condition [2]
 - MAC Reset
 - Valid ED
 - Format Error
 - Frame Stripped by another station

Location and size of the frame (each Data Unit)

The location and size of the frame is either as one contiguous Data Unit or as multiple separate Data Units.

INDICATION BREAKPOINT PROCESSING

Since the performance of a station is proportional to the number of interrupts that must be taken and the amount of status that must be processed, it is desirable to limit the number of interrupts and the amount of status that must be processed (and written).

In one approach, status is always written, but interrupts are only generated at defined breakpoints. In a more sophisticated approach, status is only generated at breakpoints, and interrupts at an even less frequent interval.

As a burst of frames is being received, several options are available for reporting/generating status. A status breakpoint could be requested whenever a frame is received or whenever a burst of frames is received. When status is requested on a burst of frames, the burst is delimited by either events that happen during a burst as a result of a single token opportunity (at the transmitting stations), or events that happen as a result of more than one token opportunity.

Within a single token opportunity, status might be generated as a result of the end of the opportunity e.g., a token or MAC frame is received) or if an FC change is detected.

An Indication threshold provides a method of requesting status every n frames. In a large burst of frames status could be requested every n frames. Status could also be requested if a frame is received with unexpected frame status.

Across more than one token opportunity, status can be generated on operations relative to a SAP (Service Access Point). Such status might be generated relative to the last FC, DA or SA received on that SAP as opposed to just the last FC, DA or SA received.

4.3 MAC Request Interface

Overview

The MAC Request Interface is used to gain access to the ring and to transmit data into the ring according to the requested service class. The interface utilizes a handshake that separates token capture from frame transmission. Upon gaining a service opportunity, a byte stream is passed from interface logic to the MAC Request Interface. The data is passed through the Ring Engine and appears at the PHY Request Interface two byte times later.

While a frame is being transmitted, the Request parameters for the next Request (if different) may be presented to the interface. At the end of the current frame transmission, a decision is made to continue or cancel the current service opportunity based on the new Request parameters.

The MAC Request Interface allows several options on a frame by frame basis. The Request options provide the ability for Source Address transparency and FCS transparency. In both cases, data from the Request stream is transmitted in place of data from the Ring Engine. The use of Source Address transparency has no effect on the sequencing of the interface. When Source Address transparency is not used, the SA from the internal parameter RAM is substituted for the SA bytes in the request stream, which must still be present. As a separate and independent option, the most significant bit of the SA, which is the escape bit for the routing information fields in long SAs, may also come transparently from the data stream, either with a transparent SA or the internal SA.

Since the FCS is appended to the frame, when FCS transparency is not used, no FCS bytes are present in the request stream.

The strip option allows frames with SAs other than this station's to be stripped based on a Void stripping. At the end of a service opportunity where any frame was transmitted with the strip option, two My_Void frames are transmitted. Stripping continues until one of them returns. In this way all frames transmitted on the service opportunity will be stripped. The second My_Void is then stripped based on the SA. The second Void frame is included just in case the first Void frame is lost. This mechanism has the desirable property that the probability of understripping is extremely low. For LANs, frames should not have any possibility of either arriving out of order or being delivered twice.

Reporting Status related to a transmission is one of the most challenging aspects of designing an interface to the BMAC device. During a transmission several errors can occur. A transmission may be terminated unsuccessfully because of external buffering or interface parity errors, internal Ring Engine errors, a MAC reset, or reception of a MAC frame. When a transmission is aborted due to an external error (and Option.IRPT is not set), a Void frame is transmitted to reset the TVX timers in all stations in the ring.

The BMAC device also guarantees that a valid frame is sent with at most L_MAX preamble (3.20 μs of preamble). This alleviates the requirement from being handled by the interface logic. During a service opportunity when data is not ready to be transmitted, Void frames are transmitted to re-

set the TVX timers in all stations. During an immediate request from the CLAIM or BEACON states, when no CLAIM or BEACON frames are ready, the internally generated CLAIM or BEACON frames are transmitted.

SERVICE CLASSES

A token capture class of "non-rstr" indicates that the transmitter token class must be non-restricted to begin servicing the request. A token capture class of "rstr" indicates that the transmitter token class must be restricted to begin servicing the request. A token issue class of "non-rstr" means that the transmitter token class will be non-restricted upon completion of the request. A token issue class of "rstr" means that the transmitter token class will be restricted upon completion of the request. (See Figure 13.)

RQRCLS (3:0)	Name	Class	THT	Token Capture	Token Issue	Notes
0000	None	None				
0001	April_1	Async thsh1	enabled	non-rstr	non-rstr	
0010	April_2	Async thsh2	enabled	non-rstr	non-rstr	
0011	April_3	Async thsh3	enabled	non-rstr	non-rstr	
0100	Syn	Synch	disabled	any	captured	1
0101	Imm	Immediate	disabled	none	none	4
0110	ImmN	Immediate	disabled	none	non-rstr	4
0111	ImmR	Immediate	disabled	none	rstr	4
1000	Asyn	Async	enabled	non-rstr	non-rstr	
1001	Rbeg	Restricted	enabled	non-rstr	rstr	2, 3
1010	Rend	Restricted	enabled	rstr	non-rstr	2
1011	Rcnt	Restricted	enabled	rstr	rstr	
1100	AsynD	Async	disabled	non-rstr	non-rstr	2
1101	RbegD	Restricted	disabled	non-rstr	rstr	2, 3
1110	RendD	Restricted	disabled	rstr	non-rstr	2
1111	RcntD	Restricted	disabled	rstr	rstr	2

Note 1: Synchronous requests are not serviced when RELR.BCNR is set.

Note 2: Restricted requests are not serviced when RELR.BCNR or RELR.CLMR are set.

Note 3: Restricted Dialogues only begin when a non-restricted token has been received and transmitted.

Note 4: Immediate Requests are serviced when the ring is non-operational. These requests are serviced from the Data state if neither RQCLM nor RQBCN are asserted. If RQCLM is asserted, Immediate requests are serviced from the CLAIM state and if RQBCN is asserted, Immediate requests are serviced from the BEACON state. RQCLM and RQBCN do not cause transitions to the CLAIM and BEACON states. Function.CLM and Function.BCN cause these transitions.

FIGURE 13. Request Service Classes

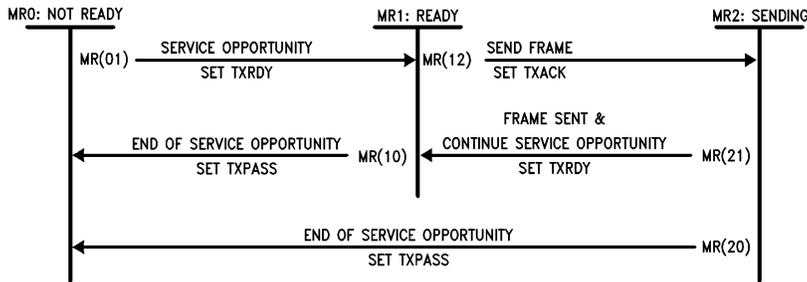


FIGURE 14. MAC Request Interface State Diagrams

TL/F/10825-12

The Logical States of the MAC Request Interface

The MAC Request Interface has three logical states: either the Ring Engine is not ready to service a request, the Ring Engine is ready for the next frame from the interface, or the Ring Engine is sending a frame from the interface. See *Figure 14*.

The values of TXRDY and TXPASS associated with these three states are shown below.

State	TXRDY	TXPASS
Not ready	= 0	= 1
Ready	= 1	= 0
Sending	= 0	= 0
Internal error	= 1	= 1

Conditions

Service Opportunity: A service opportunity occurs when it is possible to service the current request, as defined by the current service parameters (RQRCLS, RQCLM and RQBCN).

The last latched version of RQRCLS is used when RQRDY is asserted.

Send Frame: A frame can be sent from the interface when at least 8 bytes of preamble have been transmitted, TXRDY,

RQRDY and RQSEND are asserted, and RQFINAL has not yet been asserted for this request.

Continue Service Opportunity: The Service Opportunity is continued after the current frame if valid service parameters continue to be presented during the frame, and the timer(s) used for the (next) requested service class have not reached their threshold.

The table below shows the timer thresholds used for each service class:

Service Class	Threshold
All Requests	TRT Expiration
All Requests with THT Enabled	THT Expiration
Priority Asynchronous Requests	Asynchronous Priority Threshold

End of Service Opportunity: The end of a service opportunity occurs when it is no longer necessary or possible to continue the service opportunity. The service parameters are continuously compared with the current state of the Transmitter. If an unserviceable request is presented or any time threshold is reached, the service opportunity will not continue after the current frame (if any).

Two MAC Request Timing Examples are shown in *Figures 15 and 16*.

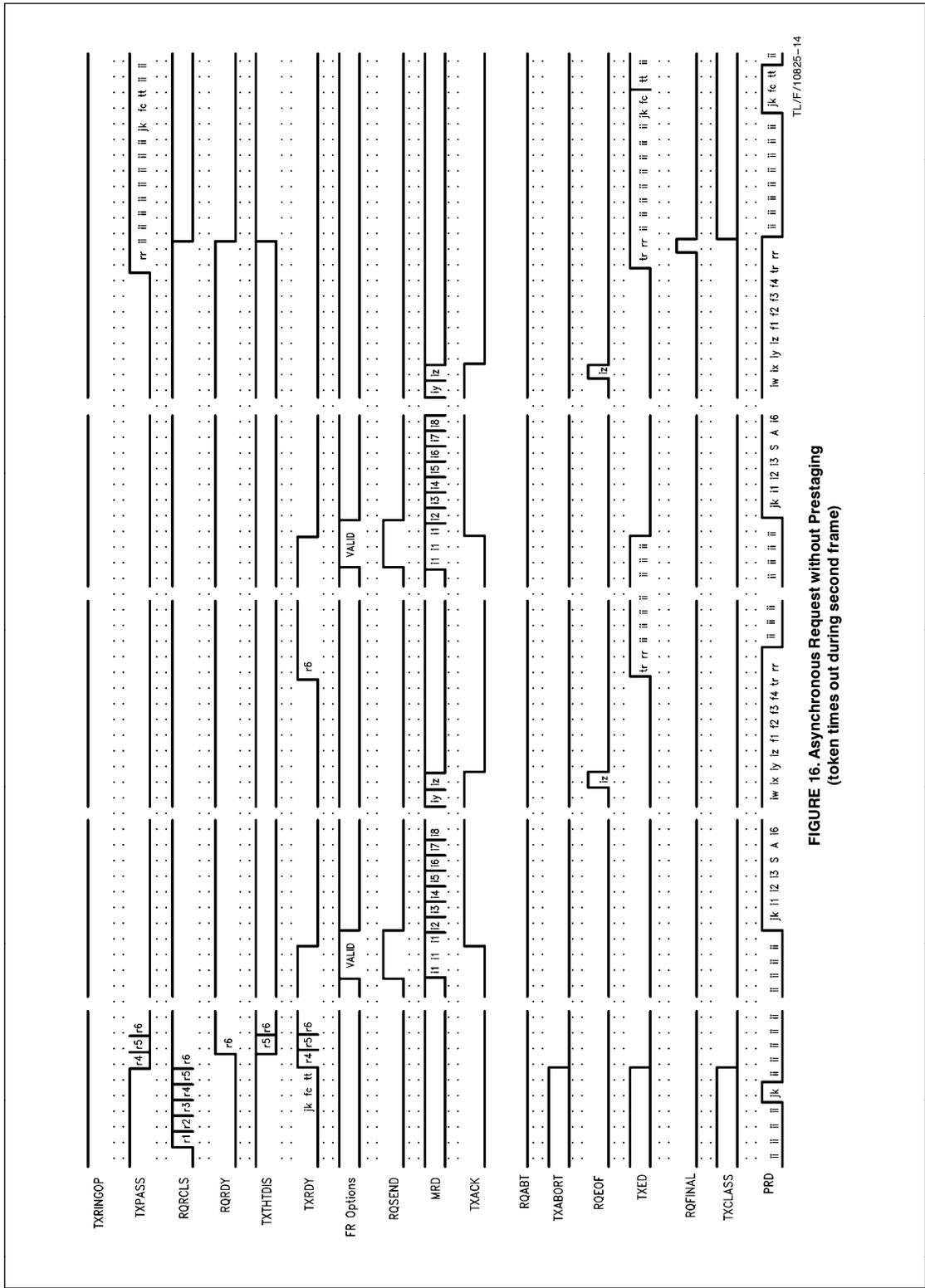


FIGURE 16. Asynchronous Request without Prestaging
(token times out during second frame)

REQUEST MACHINE

The Request Machine shown in *Figure 17* is a simplified version of a Transmit Sequencer. This could be used as the basis for many designs.

Note that all signals that begin with TX come from the BMAC Device Transmitter, and all signals that begin with RQ come from the PAL Sequencer (the Request Machine or Transmit Sequencer).

RQ0: IDLE

While in the Idle State, there are no frames to be sent and all of the status from previous frames has been reported. When a frame is to be sent, it is loaded into the temporary buffer. Once a frame is ready to be sent RQRCLS is loaded with the appropriate value. This requests a service opportunity as described by the 4-bit RQRCLS field in addition to the RQCLM and RQBCN which are used with Immediate transmissions. A transition to the Ready State then occurs.

RQ1: READY

RQRDY is asserted to indicate that the interface has or will soon have a frame that is ready to be sent. RQRDY causes the RQRCLS to be latched into the BMAC device and used while RQRDY is asserted. Once the token is captured a transition to the Send State occurs. Alternatively, enough data could be loaded into temporary storage to keep it not empty over transmission of the frame.

RQ2: SEND

Upon entry to the Send State RQSEND is asserted and the Frame Options are latched by the MAC interface. TXRDY is deasserted and TXACK asserted when the beginning of the frame is sent. A flag is set that causes us to remain in the Send State until the end of the frame as denoted by the

rising edge of TXPASS or TXRDY. While TXACK is asserted, data is transmitted (TXACK could be used as an enable for a byte counter). On the last byte of the frame, RQEOF is asserted and in response TXACK is deasserted. TXACK is used by the frame level handshake and RQEOF is returned from the frame level handshake. The Request Machine does not use TXACK or provide RQEOF.

The Last Frame and Last Request signals help control the sequencing for multiframe requests or multirequest service opportunities. Only between requests can the Requested Service Class change. The frame options may change on each frame.

RQ3: STATUS

After the frame is sent, status related to that frame is written or accumulated as appropriate either on the transmitted frame only or the returning frame as well. In short rings, this should be several byte times after the frame is transmitted, but in large rings there could be a several microsecond delay.

Conditions

Frame_Ready

- Frame_Ready is True when a frame is ready to be sent
- The Frame options (SAT, SAIGT, STRIP, FCST) must be valid at this time.

Instead of loading the entire frame into the temporary buffer before telling the MAC Interface, it would be possible to load in enough of the frame to guarantee that the temporary buffering will have enough data to keep up with the ring over the duration of the frame. An implementation like this could employ a FIFO or an equivalent speed matching device to match or smooth the bandwidth characteristics of the hardware feeding the temporary buffering.

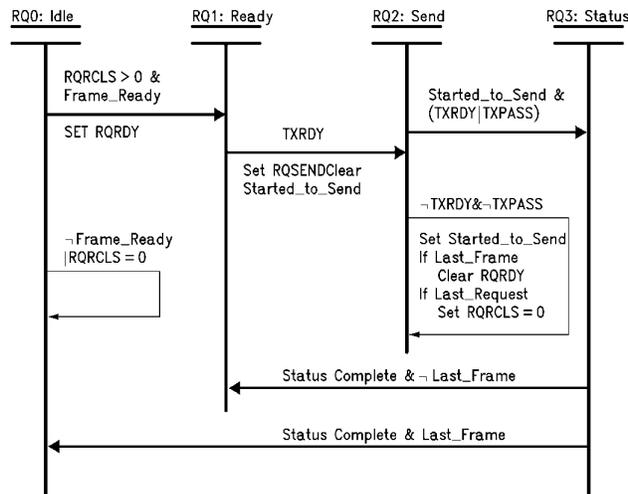


FIGURE 17. Example Transmit Sequencer

TL/F/10825-17

Notice that on the transition to the Ready State $RQRCLS > 0$ is also used as a condition. This implies that the token could be captured before the data is actually ready. In other words, from the word go a race takes place between the data becoming ready in the temporary buffering and the MAC capturing a useable token.

Started__to__Send

- This is true any time after the MAC transmitter has started to send a frame.

Status__Complete

- This is true when status for the transmitted frame has been recorded.
- This may be a result of the transmitted frame, and optionally the returning frame.

Notice that in the case where status is written for every frame, additional frames may not be transmitted until after status is written. A further alternative approach would allow status on frames of a request to be stored in a temporary location and then only write status between requests. An extension to this idea would be to store temporary status separately for each individual request. This last approach would allow both of the Status Complete transitions to occur immediately. At some point however the interlock with the actual writing of status would have to occur. For example, even in the case where status is written into temporary storage for each request, a new request could not be started until status for the previous request has been written.

A pipelined multiple request machine could even take the status__complete and last__frame transition before the end of the frame and begin to load in the new parameters for the next frame. This would allow frames of the next Request to be transmitted very soon after the frames of the previous Request.

TRANSMISSION STATUS

Transmission status concerning the previous transmission is valid in all cases one cycle after TXRDY or TXPASS is asserted.

Transmission Completion

If an Ending Delimiter was transmitted TXED will be active one cycle after TXRDY or TXPASS is asserted until the starting delimiter of the next frame is transmitted. If the frame was aborted TXABORT is active one cycle after TXRDY or TXPASS is asserted.

The frame could have been aborted for several reasons:

- a MAC Reset
- the ring going non operational
- an internal BMAC device error
- an RQABORT during the frame

TRT Expiration

If TXPASS is asserted and THT was disabled during the last frame that was transmitted (THTDIS is asserted), TRT has expired. This is a serious error and indicates that there was an overallocation of synchronous bandwidth or a station used more than it was allocated. The ring will likely be in the CLAIM state when this occurs.

External CLAIM/BEACON Transmission

When transmitting CLAIM/BEACON frames from the CLAIM/BEACON state, if TXPASS is asserted the CLAIM/BEACON process is complete. In this case TXABORT indicates if this station won (TXABORT=0) or lost (TXABORT=1) the CLAIM/BEACON process.

Holding the Token

At end of the frame, TXRDY is asserted if possible (THT or TRT has not expired) or desirable (there are more frames to transmit on this service opportunity). Otherwise TXPASS is asserted.

Issue Token Class

If RQRCLS goes to zero at any time during a service opportunity a token will be issued.

TXCLASS indicates the class of token that was issued, in the case of TXPASS, or would have been issued in the case of TXRDY. Since the issue token class is given in the RQRCLS encoding, this is a sanity check, especially in a non-pipelined implementation.

For certain SMT frames (NSA frames), the returning status is of considerable interest. In these cases, it may be easier to just turn on copying of SMT frames based on MFLAG (i.e., AFLAG or MFLAG instead of the normal AFLAG and not MFLAG). The status can be the same as the reception status.

PROVIDING A TRANSMISSION CONFIRMATION SERVICE

Providing a Transmission Confirmation Service is one of the most challenging aspects of designing an interface to the BMAC device. Some useful hints for providing such a service are given below.

Overview

A transmission confirmation service returns ending status for a request to transmit one or more frames. As frames of a request are transmitted, the status on the returning frames is accumulated. When all of the frames of the request return around the ring, a positive status is given. Should anything other than the expected frames (with the correct FC, SA, FCS, and frame status values) return before all of the transmitted frames return, a negative status is generated. Status is thus generated when either all of the frames return, or an abnormal condition is detected.

The transmission confirmation service is similar to the (SM_)MA_UNITDATA_STATUS.indication (formerly called (SM_)MA_DATA.confirmation) defined in the ANSI FDDI MAC standard.

Ring Assumptions

- Returning frames can be associated with transmitted frames.

At most one request can be serviced (for each SAP) during a given token opportunity. Since the order of SAP serviced is fixed, the order for matching returning frames with their corresponding requests is also known. The returning FC, SA and FCS could be used to distinguish a frame transmitted by this station and match it with its corresponding SAP. Additional fields may be used to increase confidence in the association.
- Frames return in the order transmitted.

We assume that no frames are maliciously inserted into the ring.
- After a token is captured by a station for transmission, the next returning frames will be those transmitted by this station.

Since there is a single token, the next frame received by this station should be the (first) one it transmitted unless another station has failed to strip its frame(s).

- Synchronization points:

At these synchronization points, the number of frames transmitted and the number of frames received should be equal.

- Token received
- Frame with a different SA is received, after receipt of a frame with my SA.

Error Conditions

- Frame not recognized by addressed station
No station recognizes the Destination Address, and as a result the A indicator is not set. Either the station is not present or a line hit corrupted the Destination Address.

- Frame not copied by addressed station
The addressed station runs out of buffering resources and as a result does not set the C indicator. This is probably the most likely of the error conditions.

- Frame Error due to line hit
For example when a line hit causes an FCS error.

- Frame Lost due to line hit
For example the starting or ending delimiter is destroyed, or a data symbol becomes a non-data symbol. This may have different effects when the lost frame is:

- first frame of request
- a middle frame of a request
- the last frame of a request

- Frame not recognized by source station
A line hit corrupted the FC or the Source Address.

- Frame incorrectly recognized by source station
Frame insertion or misdirection can occur during reconfiguration, but CMT will remove such frames by scrubbing. An alias could be caused by a line hit on the FC or SA, but it is highly unlikely to contain the proper FCS value (this would require multiple errors or an unfortunate pattern on a transparent FCS transmission). Indiscriminate use of transparent FCS transmission could diminish the integrity of the confirmation service.

Error Handling

For each request, an expected status parameter is given either explicitly with the request or implicitly (it was given explicitly previously and has not changed). The expected status contains the expected value of the returning FCS, E, A and C indicators. If the expected value of these indicators is not returned, a negative confirmation is generated.

- Frame not recognized by addressed station
This condition can be handled using the expected status. The returning A indicator is not set, so the expected frame status is not returned. A negative confirmation is generated and the request is optionally terminated.

- Frame not copied by addressed station
This condition can be handled using the expected status. The returning C indicator is not set, so the expected frame status is not returned. A negative confirmation is generated and the request is optionally terminated.

- Frame Error due to line hit
This condition can be handled using the expected status. The returning E indicator is not reset or the FCS is invalid, so the expected frame status is not returned. A negative confirmation is generated and the request is optionally terminated.

- Frame Lost due to line hit

This type of error is more difficult to handle because it cannot be detected until the next synchronization point. At the next synchronization point the received and transmitted counts for the current request will not be equal. A negative confirmation is generated and the request is optionally terminated.

- Frame not recognized by source station
This type of error is more difficult to handle because it creates a false synchronization point. At this point the received and transmitted counts for the current request will not be equal. A negative confirmation is generated and the request is optionally terminated.

- Frame incorrectly recognized by source station
This condition will be detected unless the line hit caused a formerly invalid FCS to become valid. When detected, this condition may produce a false negative confirmation, but it will not produce a false positive confirmation.

5.0 Bridging/External Matching Support

There are two major considerations for MAC level Bridging products on FDDI:

1. How stripping will be accomplished and
2. How the control indicators will be handled.

The FDDI standard does not have a recommended Bridging algorithm like 802.3's Transparent Bridging and 802.5's Source Routing Bridging.

The issue with stripping is that every station needs to strip every frame it transmits. Typically this is accomplished by stripping based only on the Source Address. However, in bridging applications where the SA is not necessarily of this station, this station needs to either watch out for all of its frames (and use CAM technology to assist the strip decision) or use some other mechanism. Currently, the FDDI X3T9.5 Committee has agreed that the strip points of all frames is before the fourth byte of the INFO field. That implies that any fragment with more than three bytes of INFO is an illegal fragment.

The BMAC device provides an alternate stripping mechanism to accomplish the stripping by sending out two My_Void frames before the token and stripping everything until the first My_Void returns.

The definition of a bridgeable frame also has impact in bridging applications. Only LLC frames are bridgeable. Management frames (MAC and SMT frames) are not bridgeable. BEACON frames, CLAIM frames and SMT frames are local to a single ring. Likewise, synchronous and restricted dialogs are local to a single ring.

5.1 EXTERNAL MATCHING LOGIC

External Matching Logic is used to allow this station to recognize additional addresses. These additional addresses may be aliases (additional addresses that are not this station's management address) or additional group addresses not recognized by the BMAC device. In addition, more perfect filtering or routing could be done on certain classes of frames. Certain bridges and routers (not 802.1d transparent bridges since these address matches do not set the A_ind) can be viewed as having a set of aliases.

The result of the address comparison is then fed into the copy criteria and used to set the A indicator (the BMAC device EA input). The C indicator should be set as usual unless the frames recognized by the external matching logic are sent to a different buffer memory and generate a separate VCOPY signal.

There are two choices of where the external matching logic may tap into the data stream, either at the PHY Indicate Interface or at the MAC Indicate Interface. We recommend using the PHY Indicate Interface between the PLAYER device and BMAC device, because it provides a three byte time advantage over the alternative.

When using alias addressing it is also possible to strip frames on the basis of these additional aliased source addresses if the STRIP option is not used or desired. Three byte times after the EM signal is asserted, Idle symbols are transmitted into the ring and the frame is stripped.

5.2 SA, SAIG, FCS TRANSPARENCY OPTIONS

These transparency options are particularly useful in bridging applications. The SA transparency options are useful in certain Transparent Bridging algorithms and in Source Routing Protocols. The SA and SAIG transparency options allow the SA to be sent transparently from the data stream as opposed to being sent from the MAC Parameter RAM. The FCS transparency is particularly useful between bridged rings (end to end FCS checking with FDDI to FDDI bridges). The FCS transparency could also be useful for diagnostic purposes and with implementer defined FCSs on implementer frames.

5.3 IMPLEMENTING THE BRIDGING COMMITTEE RECOMMENDATIONS

The meaning of the control indicators also is impacted in bridging applications. The ANSI committee decided that a

bridge will only set the A_IND if the frame is explicitly addressed to the bridge and may optionally set the C_IND for all frames copied with the intent to be forwarded. If an end station receives a frame addressed to it with the A indicator reset, the C indicator set, and it does not copy the frame, the station sends it out with the A indicator set but the C indicator reset.

END STATIONS

For end stations and bridges which also act as end stations, the BMAC device always implements the option recommended by the Bridging Working Committee to X3T9.5. For frames with A not set and C set for which the station intends to set the A indicator but not the C, the A indicator is transmitted as set and the C indicator is transmitted as Reset.

BRIDGES

For stations acting as bridges, the committee recommends that for frames copied with the intent to forward (for which a station address or alias match did not occur), only the C indicator should be set, not the A indicator.

To accomplish this, since the BMAC device will not set the C indicator without setting the A indicator as well, it is necessary to intercept the byte stream between the BMAC device and the PLAYER device. Fortunately, the difference between the R and S symbol is only a single bit. Thus, when a frame is copied with the intent to forward it, the received A indicator is not set, and the station's AFLAG is not set, the C indicator must be changed from an R to an S. This occurs one byte time after EDRDVD is asserted and requires that PRD(0) be changed from a 0 to a 1.

This easily fits into a small slice of a PAL and is only required in bridges implementing this option.

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.



National Semiconductor Corporation
 2900 Semiconductor Drive
 P.O. Box 58090
 Santa Clara, CA 95052-8090
 Tel: 1(800) 272-9959
 TWX: (910) 339-9240

National Semiconductor GmbH
 Livny-Gargan-Str. 10
 D-82256 Fürstenfeldbruck
 Germany
 Tel: (81-41) 35-0
 Telex: 527849
 Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
 Sumitomo Chemical
 Engineering Center
 Bldg. 7F
 1-7-1, Nakase, Mihama-Ku
 Chiba-City,
 Ciba Prefecture 261
 Tel: (043) 299-2300
 Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
 13th Floor, Straight Block,
 Ocean Centre, 5 Canton Rd.
 Tsimshatsui, Kowloon
 Hong Kong
 Tel: (852) 2737-1600
 Fax: (852) 2736-9960

National Semicondutores Do Brazil Ltda.
 Rue Deputado Lacorda Franco
 120-3A
 Sao Paulo-SP
 Brazil 05418-000
 Tel: (55-11) 212-5066
 Telex: 391-1131931 NSBR BR
 Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
 Building 16
 Business Park Drive
 Monash Business Park
 Nottingham, Melbourne
 Victoria 3168 Australia
 Tel: (3) 558-9999
 Fax: (3) 558-9998

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.