

BMAC™ Device Software Design Guide

National Semiconductor
Application Note 678
David Brief
January 1990



Table of Contents

1.0 INTRODUCTION

2.0 CONTROL SERVICES

3.0 INITIALIZATION PROCEDURE

- 3.1 Put the BMAC Device in Stop Mode
- 3.2 Load the MAC Parameter RAM
- 3.3 Clear the MAC Event Counters
- 3.4 Clear the Event and Mask Registers—Optional
- 3.5 Modify the Timer Thresholds—Optional
- 3.6 Set the Option Register
- 3.7 Set the Mode Register
- 3.8 Loopback Testing—Optional

4.0 DURING OPERATION

- 4.1 Control Interface Event Registers
- 4.2 Event Control
- 4.3 Servicing Interrupts
- 4.4 Event Counters

5.0 EXAMPLE PROCEDURES

- 5.1 Getting the Ring Operational
- 5.2 Receiving and Copying a Frame
- 5.3 Initiating Claim

6.0 DIAGNOSTIC SCENARIOS

- 6.1 For (At Least) One Station
- 6.2 For Two or More Stations
- 6.3 Path Tests

1.0 INTRODUCTION

This application note describes how to initialize the BMAC device and what to do while it is inserted in a ring. The software support to implement the Media Access Control (MAC) level protocol and the necessary services to a Station Management (SMT) entity in an FDDI node are described. The necessary data service support for transmitting and copying frames is not covered in this application note.

The MAC protocol and all of the services required by SMT (except the SMT data services) are supported through the BMAC device's Control Interface. The processor running this software must have access to the Control Bus and have the ability to respond to interrupts.

2.0 CONTROL SERVICES

The Control Services provided by the BMAC device are accessed through the Control Interface. A more detailed description of the facilities and services provided by the BMAC device is given in the BMAC device datasheet.

The BMAC Control Bus address space is divided into 4 address ranges:

The Operation registers control the current mode of operation of the BMAC device (Run/Stop, Internal Loopback) including the options that are being used (Short/Long Addressing, MAC state machine options). In addition several functions may be initiated (Master Reset, MAC Reset, Claim, Beacon).

The Event registers record the occurrence of events which may cause interrupts (each event bit has a corresponding mask bit). These include Ring, Token and Counter Increment/Overflow Events.

The MAC Parameter RAM contains all of the MAC related parameters such as this station's long and short addresses. The MAC Counter/Timer Thresholds contain the event counters (Frame, Error, Lost, Copied, Not Copied, Transmitted, Token) in addition to the programmed thresholds for the various MAC timers such as TMAX and TVX.

The various ranges may be accessed for reading and/or writing either always or only in stop mode as shown below.

Address Range	Description	Read Cond	Write Cond
00–07	Operation Registers	always ²	always ²
08–2F	Event Registers	always ²	always (Cond) ²
40–7F	MAC Parameter RAM	stop ^{1,3}	stop ^{1,3}
80–BF	MAC Counters/Thresholds	always	stop ¹

Note 1: An attempt to access a currently inaccessible location because of the current mode or because it is a reserved address space will cause a command error (ESR.CCE set to One).

Note 2: Read and write accesses to reserved locations within the Operation, Event Address ranges cause a command error (ESR.CCE set to One).

Note 3: The MAC Parameter RAM is also accessible when:

- a) the MAC Transmitter is in states T0, T1 or T3;
- b) Option.ITC and Option.IRR are set
- c) Function.CLM and Function.BCN are not set otherwise accesses will cause a command error (ESR.CCE set to One).

Note 4: Reserved bits in registers are always read as 0 and are not writable.

3.0 INITIALIZATION

Before being inserted into a ring, the BMAC device must be initialized. To initialize the BMAC device the following steps should be followed. Each action is explained further below.

Put the BMAC device in Stop Mode
Load the MAC Parameter RAM
 Individual Addresses (MLA, MSA)
 Group Addresses (GLA, GSA, MAP, SGM)
 Requested Token Rotation Time—TREQ
 Beacon Information—TBT
Clear the MAC Event Counters
Clear the Event and Mask Registers—Optional
Modify the Timer Thresholds—Optional
 TVX
 TMAX
 Asynchronous Priority
Set the Option Register
Set the Mode Register
Loopback Testing—Optional

3.1 Put the BMAC Device in Stop Mode

The Mode Register is programmed first to place the BMAC device into STOP mode so that all registers can be accessed.

The Parity for the different interfaces is enabled here as is the ability to be in MAC Loopback. At Initialization it doesn't matter if the part is configured in loopback, but since loopback testing will probably be done after initialization it could be set now as well.

In a system without parity checking on the Control Bus or the MAC Interface the Mode register would be set to 44h. This enables the parity checking on the PHY interface which is actually part of the Ring. (Parity is always generated by the BMAC device to the PLAYER™ device.)

3.2 Load the MAC Parameter RAM

Load RAM with values as indicated in the following passages.

Individual Addresses (MLA, MSA)

MLA—the 48-bit address
MSA—the 16-bit address—Optional

Group Addresses (GLA, GSA, MAP, SGM)

The same MAP is used for the short and long group addresses. To disable Group Addressing, GLA and/or GSA must be set to all ONE's.

Long Addresses—GLA plus MAP (Optional)

Short Addresses—GSA plus MAP (Optional)

Fixed Group Address—FGM(15:1) This is located at FF FF FF FF 0x where the last nibble is fanned out using the Fixed Group Map (SGM).

The Broadcast Address—FGM(0) **must** be set to One in order to participate properly in the Next Station Addressing protocols that rely on the Broadcast Address.

Requested Token Rotation Time—TREQ

This should be loaded with FF000000 unless this station is using/managing Synchronous Bandwidth.

When this station is using Synchronous Bandwidth and needs a faster average response time for its Synchronous Bandwidth, the value of TREQ is used in the Claim process to negotiate the target timer rotation time.

If this station wins the Claim process, every station will use this station's value of TREQ as TNEG.

Determining TREQ

TREQ is this MAC's requested value for the token rotation time, i.e., in the worst case, this station wants to "see" a token at least once every 2*TREQ. For example if a station wanted to be guaranteed to capture a token every 2 ms it would set TREQ to 1 ms.

1 ms is 12,500 ticks of the 80 ns clock or 30D4 hex. Subtracting this from 1-0000-0000 yields FF-FF-CF-2B for TREQ since TRT is an unsigned twos complement up counter.

Since the least significant byte of TREQ is transmitted as 0, the value should be rounded up in order to guarantee that this station will see the token as often as it needs to. In this case TREQ would be written to FF FF D0 00.

Beacon Information—TBT

TBT(31:0) should be loaded with 00 00 00 00.

This is only modified in the case of specialized Beacon Frames. Additional Beacon frames are being defined by the FDDI standards committee.

The BMAC device has two limitations on the Beacon Frames it can transmit. Firstly, only frames with a Null DA with four bytes of information are transmitted by the BMAC device. This precludes the use of the internally generated Beacon Frames for the directed Beacon because it can not be sent to the SMT multicast address. Secondly the size restrictions on Beacon Frames also preclude their use for conveying useful information. The Beacon Frame is the penultimate immediate transmission. (Blocking the MAC Indication Input with Option.IRPT will allow transmission of Beacon Frames in the presence of an upstream Beaconer.)

3.3 Clear the MAC Event Counters

The counters are 20-bit counters, but SMT requires 32-bit counters. This implies that the upper 12 bits are maintained by software.

In order to use the low order bits directly without having to calculate how much they have changed since the last time they were read, the counters should be cleared at initialization.

3.4 Clear the Event and Mask Registers—Optional

The Event and Mask registers are actually cleared on a Master Reset. If you did not do a Master Reset before the initialization sequence it is good practice to clear these registers.

3.5 Modify the Timer Thresholds—Optional

At Master Reset the timer thresholds are set to the defaults recommended by the standard. In most applications there is little incentive to modify the defaults.

TVX

In most applications, the valid transmission timer, TVX, would remain at its default value. The value of TVX determines in how long a valid transmission should be seen. If a valid frame is not seen in this period of time the Claim process is started. This is one of the recovery required conditions.

TMAX

In most applications, TMAX would also remain at its default value. The value of TMAX is used to determine how long to stay in the Claim process before starting to Beacon.

Asynchronous Priority

If not using an Asynchronous priority, the threshold register THSH1 should be set to 0. If using one, the threshold registers should be set appropriately.

3.6 Set the Option Register

There are three types of options that are configurable via this register: the addressing modes, the state machine transitions, and the frame status indicators.

In a typical application only the Long Address would be enabled. In this case the Option Register would be programmed with 02h.

Additional detail on the use of the state machine Options and frame status options is given in the BMAC device data-sheet.

3.7 Set the Mode Register

The Mode Register is set last so that the RUN bit can be set after all other initialization is complete.

The Parity for the different interfaces is enabled here as is the ability to be in MAC Loopback.

During operation in a system without parity checking on the Control Bus or the MAC Interface, the Mode Register would be set to 05h. If Loopback testing is to be done following initialization the Mode register should be set to 45h.

3.8 Loopback Testing—Optional

Loopback testing can be considered an integral part of the initialization sequence. Because the BMAC device is full duplex, loopback testing can be used to check most of the operation of the BMAC device. In loopback it is possible to get an operational ring and check that the claim process is entered and that a token is issued. The token can then be captured and frames transmitted. This allows frames to be sent to oneself to check all of the frame handling logic, address comparison logic, etc. See the diagnostic section below for more details.

Even though the "ring" is very small (2 bytes in internal loopback) an operational ring can still be reached. The BMAC device transmits void frames between tokens since the ring is not big enough to hold a token (and its preamble).

4.0 DURING OPERATION

After the BMAC device is inserted into an operational ring, various SMT processes monitor the event counters and respond to error and exceptional events.

The Ring Management (RMT) process is responsible for getting and keeping the logical ring of MACs operational, thereby allowing the MACs to provide data services to their users. RMT uses the events generated by the BMAC device such as:

- Timer Expirations (TVX, TRT)
- Reception of MAC frames
- Capturing/Passing a Token
- Duplicate Token/Duplicate Address
- Losing a frame
- Not copying a frame

The ring changing operational states in order to monitor the ring and take the appropriate actions when the ring cannot provide data services.

To that end, RMT provides higher level recovery mechanisms than the MACs Claim and Beacon processes, including Directed Beacons. It has the ability to initiate Configuration Management (CMT) recovery including the CMT Trace process. RMT also can take advantage of some of the BMAC device state machine transition options, such as Inhibit Recovery Required, to prevent this station from entering and potentially winning the Claim process.

RMT is also responsible for resolving any duplicate address detection problems that prevent the ring from becoming operational and managing the use of restricted tokens on the ring.

Management Information Base

The station keeps track of all of its information in the Management Information Base (MIB) and updates the information in its database and in the MAC and PHY entities it controls when appropriate. SMT uses the MIB to generate frames conveying information about the station configuration and its "neighborhood". Other stations may optionally access and modify this station's MIB using the optional frame based parameter management protocol (sometimes referred to as the remote set/get protocol).

The MAC related information kept in the MIB includes:

- The current and previous values of the event counters
- Which PLAYER devices the MAC is connected to in the station
- The most recently determined upstream and downstream neighbors (this is useful for creating logical ring maps)
- The addresses that the MAC can interpret
- The current values of TMAX, TVX, TREQ
- Current Synchronous Bandwidth allocations
- Current Asynchronous priority threshold

The station's current estimate of the ring latency is also kept in the MIB. The BMAC device is well suited for this function since it contains a latency counter to measure the ring latency. This is necessary to calculate the ring load and to set meaningful asynchronous priorities. The ring latency is also useful for tuning default timer values that assume a maximum default size ring (TMAX for example).

In addition to the SMT processes occurring within a station, other performance monitoring processes and data service related processes may be active. An example performance monitoring process could measure the load on the network over a period of time by reading the token count periodically and deducing the load from that based on the ring latency and the number of tokens received over a period of time. The ring latency could also be used to optimize certain timing parameters in the station and for accurately setting the asynchronous priorities.

Additionally, the buffering capabilities of the station could be deduced by looking at the number of frames not copied compared with the number of frames that were copied.

4.1 Control Interface Event Registers

The event registers record the occurrence of events. Events are recorded in condition latch registers and contribute to the Interrupt when the bit in the corresponding mask register is enabled. See Figure 1.

4.2 Event Control

All events are latched in condition latch registers, and may generate interrupts. Events are grouped into classes according to their probable usage. Event classes are enabled via mask registers in groups at the Interrupt Register and individually at the Condition Registers. Enabled events may generate interrupts. Each condition latch register has a corresponding mask 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. Only enabled conditions contribute to the Interrupt signal.

The interrupt signal is an open drain signal to allow it to be combined with a similar signal from other BMAC and PLAY-ER devices in the station. In this case, software would determine which device is causing the interrupt and process it accordingly.

4.3 Servicing Interrupts

After an interrupt has occurred, the source of the interrupt must first be determined in order to decide how to service the interrupt. In the process of servicing an interrupt, a management entity may use one or both levels of condition masks to disable new interrupts while one is being serviced. Soon after the managing entity has processed the interrupt to some extent, it is ready to rearm the interrupt in order to be notified the next time the event occurs.

The Interrupt Condition Register (ICR) always contains the merged output of the masked condition registers. It is only possible to remove a condition by clearing the corresponding condition latch register bit. Condition latch registers are cleared by writing 0's to the appropriate bits. By storing the events on chip, and having the ability to selectively clear bits, the need for the software to maintain a copy of the event registers is alleviated.

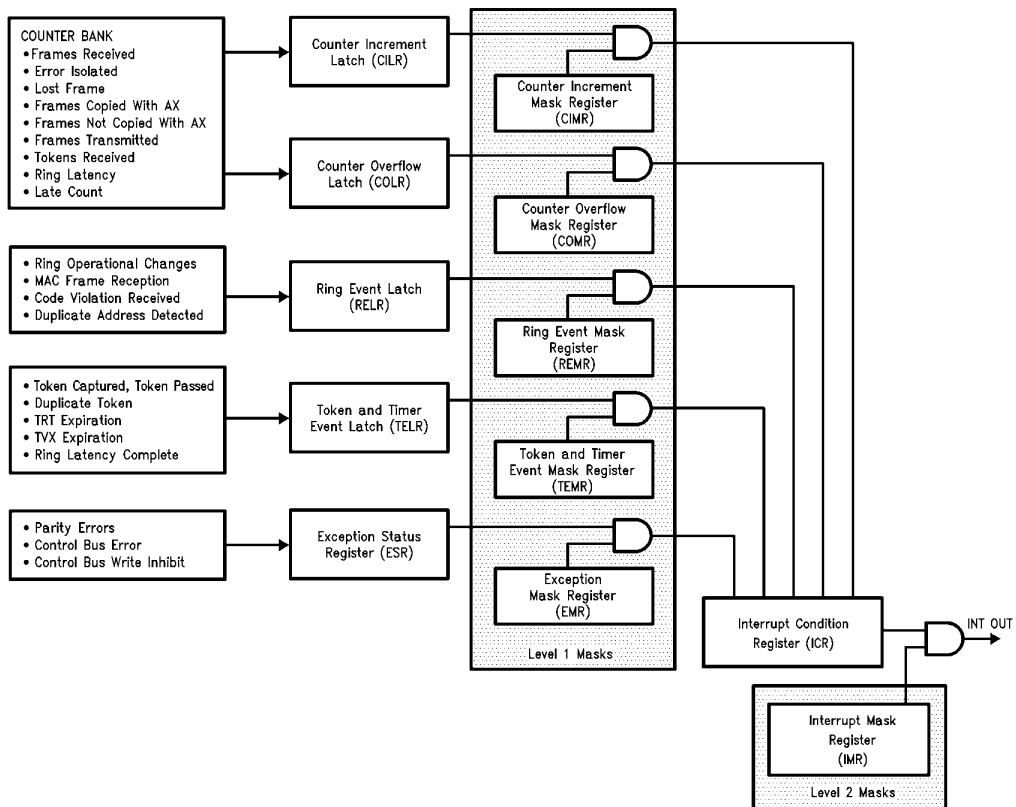


FIGURE 1. Control Interface Event Registers

TL/F/10796-1

Example Interrupt Service Routine

1. Disable Interrupts
2. Determine which event is triggering the interrupt
3. Determine which condition(s) exist that need(s) attention
 - a. Read ICR
 - b. Read appropriate Condition Register
4. Process event
 - a. Complete Processing for the event or
 - b. Queue a process to handle the event
5. Clear or Mask the condition
 - a. To Clear:
 - i. Will only clear conditions that have not changed since last read
 - ii. Make sure that last value read is in the Compare Register
 - b. To Mask:
 - i. Clear the appropriate Mask Bit
 - ii. Before the Mask Bit is set to reenable interrupts, the condition must be cleared as shown above.
6. Reenable interrupts

Additional Notes

1. Nesting of Interrupts:

Nesting of interrupts may be of use in driver level software. For example if an error condition occurs while “processing” a frame it may be prudent to stop processing the frame and handle the error condition. Alternatively, once processing of frames begins, it may not be necessary to reenale frame related interrupts until all copied frames have been processed. This is especially true with token ring protocols where bursts of frames between stations is common (or at least should be common to optimize performance of the media and the software. Software performance would be increased because the software performance can be related closely to the number of interrupts that need to be processed).

2. Conditional Writes:

In the period between the Read of a condition latch register, and the corresponding Write to reset the condition, additional events could occur. To prevent the overwriting and consequent missing of events, an interlock mechanism is used. Whenever a condition latch register (RELR0, RELR1, TELR, CILR, COLR, or ESLR) is Read, its contents are stored in the Compare Register.

Each bit of the Compare Register is compared with the current contents of the register that is to be written. For any bit that has not changed, the new value of the bit is written into the register. For any bit that has changed the writing of the bit is inhibited. This prevents the software from overwriting bits which have changed since the last read and losing interrupt events. The fact that an attempt was made to modify a changed bit in the register is latched in the Conditional Write Inhibit bit of the Exceptional Status Register (ESR.CWI). This bit is written unconditionally after each write to a conditional write register. This is different than in the PLAYER device.

The Compare Register may also be written unconditionally by software. There is a single compare register for all of the conditional write registers in the BMAC device. This is different than in the PLAYER device where each conditional write register has its own compare register.

After a conditional write register is read, if another conditional write register is read before a condition is cleared, the compare register will no longer have the appropriate value in it. In such cases the compare register should be written with the previously read value of the register.

The compare register may also be useful for software compare/update sequences and for diagnostic purposes.

4.4 Event Counters

The event counters are 20-bit counters, but the SMT MIB and SMT frames requires 32-bit counters. This implies that the upper 12 bits are maintained by software.

The counters may be read either periodically or upon an event. The fact that individual counters incremented or overflowed is reported as an event in the CILR or COLR event registers respectively.

In order to use the low order bits directly without having to calculate how much they have changed since the last time they were read, the counters should be cleared at initialization.

Some uses of the counter may require that a consistent value be obtained across two counters. Since the event that a counter incremented is stored, software can tell if a consistent reading was obtained.

When reading individual counters, the upper 12 bits of the counter are latched when the low order 8 bits are read. This allows consistent readings of a single counter and implies that the low order byte must be read first.

At least one of the event counters is incremented for every Starting Delimiter (JK) received.

After a Starting Delimiter (JK) is detected:

If Token Ending Delimiter—Increment Token Count

{or} **If** Format Error—Increment Lost Count

{or} **If** Frame Ending Delimiter—Increment Frame Count

If Er = R and FCS error detected—Increment Error Isolated Count

Else If AFLAG and VCOPY—Increment Frame Copied Count

Else If AFLAG and not VCOPY—Increment Frame Not Copied Count

5.0 EXAMPLE PROCEDURES

5.1 Getting the Ring Operational

To get the ring operational requires setting the Run bit in the Mode Register to a ONE. Once TVX expires, the Claim process will be entered, and if a single token path exists, the Claim process will quickly complete, a token will be issued and the ring will become operational. This occurs even when in internal loopback.

e.g.:

- Set Mode.Run = 1.
- TVX will soon expire causing entrance to Claim.
- Claim will resolve and a token will be issued.
- The reception of a valid token causes the ring to become operational.
- Once the ring is operational the station should check to make sure that $TNEG > T_{min}$ to ensure that it can operate on the ring as an equal station (if this is not true it may be denied service for excessively long periods of time).

5.2 Receiving and Copying a Frame

All frames are received by the BMAC device, but only frames addressed to this station are copied. Every frame received by this station causes the frame received counter to be incremented. In addition for frames addressed to this station (when the A flag is set) either the frame copied (FRCOP) or frame not copied (FRNCOP) bit is set and the appropriate counter is incremented.

e.g.:

- CILR.FRCOP is set indicating that a frame was copied by external_logic
- Can be used to wake up driver software
- Software that receives this interrupt should
 - Process the Frame Status
 - Process some of the frame
 - Pass remainder of frame to another process to be processed

5.3 Initiating Claim

- Enter stop mode (this breaks the ring)
- Load a new value of TREQ into the parameter RAM.
- Load TNEG with TMAX.
- Clear the events related to claim in RELR0 and RELR1
- Enter run mode.
- Initiate the claim process by writing the Function Register with 14.
- Wait until Function Register is zero.
- Check that $TNEG \geq TREQ$ when claim completes.
- See if this station won claim:
 - See if RELR1.MYCLM is set
 - If set see that $TNEG = TREQ$.

6.0 DIAGNOSTIC SCENARIOS

6.1 For (At Least) One Station

- Control Interface Checkout
- Internal Frame Generation Tests
- State Machine Sequencing Tests
- External Frame Transmissions
- Test of Token Timers
- Test of Transmission Options
- Full Duplex Operation

6.2 For Two or More Stations

- Beacon Scenarios
- Claim Scenarios (Need Three To Do Complete Testing)
- Duplicate Token Conditions
- Duplicate Address Conditions
- Abnormal Frame Termination (Format Errors, FCS Errors, etc.)

6.3 Path Tests

A path test is performed to determine that everything is working in that path. By doing successive path tests on parts of the same path, the Fault Domain can more accurately be determined (i.e., which chip/connector is broken).

The fact that the chip set is full duplex greatly aids the path tests. This allows identical tests to be run at all levels.

Paths that can be tested in a node:

- Through the BMAC Device
- Through All of the Station Paths
- Through the CRD device for SAS and through Each CRD device for DAS and Concentrators
- Through Every Station on the (Logical) Ring

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: (408) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-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,
Chiba 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