VPI/VCI Translation and Cell Tagging in ATM With The MU9C1480A LANCAM®

INTRODUCTION

The MUSIC LANCAMs are content-addressable memories (CAM) originally intended for LAN bridge and router address filtering applications. However, LANCAMs function well, due to its flexibility, in a broad range of applications other than LANs. Asynchronous Transfer mode (ATM) is the basis of a new network generation that can transport data with a bursty nature (file transfer), digitized voice (continuous bit rate), and compressed digitized voice and video (variable bit rate). The promise of ATM is that it will unite telecom services with LAN services into a world-wide multi-use communications network. This Application Note describes the VPI/VCI transfer, using the MU9C1480A-90DC, a 1024 x 64 bit CAM as an example. Other LANCAMs, up to 4 K deep, are currently available from MUSIC Semiconductors and can be used in the same manner.

ATM

VPI/VCI Translation

ATM is a connection-oriented system. Every connection has to be established prior to the exchange of data. Each connection is identified by a Virtual Path identifier (VPI)/ Virtual Channel identifier (VCI) combination, which can be found in the header of all ATM cells carrying information about that specific connection.

At the User-Network interface (UNI), the VPI is an 8-bit field and the VCI is a 16-bit field in the cell header, while at the Network-Network interface (NNI) the VPI is a 12-bit field and the VCI a 16-bit field.

The VCI/VPI combination has a local significance. Each part of a total connection over several switches has its own unique VPI/VCI combination for that part of the connection, either from user to switch or from switch to switch. This implies that whenever an ATM cell travels through a switch, the VPI/VCI value has to be changed into the value used for the next part of the connection. This is called VPI/VCI translation.

One of the best solutions available today to perform the VPI/VCI translation is with the help of a MU9C1480A LANCAM, because there is:

No limitation in range of the VPI/VCI(s)
No distribution-dependent limitation in search speed
No limitation in search speed depending on length of list

Cell Tagging

In most switch implementations, the VPI/VCI translation is performed at the moment the ATM cells have been received by the switch port. In many designs, in addition to the VPI/VCI translation, the cells are tagged with information to route them through the switch. This tagging information can then be linked to the new VPI/VCI at the same time.

Policing

Besides VPI/VCI translation and cell tagging, a switch port has to perform more functions that can be assisted by the LANCAM. One of the functions is policing. With the help of the LANCAM, the switch is able to check whether the VPI/VCI is known and a VPI/VCI translation and tagging operation can take place or whether the cell should be discarded due to an unknown VPI/VCI.

The amount of cells that are allowed to come in during a certain time frame is limited and based upon the “contract” for that specific connection. The average bandwidth and burstiness of the incoming cells for a connection have to conform to certain rules. If these rules are violated the switch can decide to throw away cells from that specific connection.

All VPI/VCIs of incoming cells are checked by the LANCAM to see if they belong to existing connections. If the VPI/VCI does not belong to an existing connection, the cell can be thrown away. When a VPI/VCI belongs to an existing connection, a policing IC can check on bandwidth voliations for this specific VPI/VCI.

SYSTEM BLOCK DIAGRAM

A wide variety of interfaces between ICs is found in the market today, not only varying in speed and type of information carried, but also varying in width. To illustrate the feasibility of using a CAM in a 622 Mbps ATM application, a possible ATM system is depicted in Figure 1. In this example system, a 16-bit wide interface (DI15-DI0) is used for the incoming and outgoing ATM information, clocking is provided by CLKI for up to 622 Mbps (cell inter-arrival time of appr. 695 ns.), the start of the cells is indicated by C_STRI, at least one byte of extra information is carried between two cells, and idle cells are indicated by IDLE_CI.

The block diagram shown in Figure 1 consists of one or more MU9C1480A LANCAMs, containing the database of VPI/VCI combinations, a DRAM (in this example a OKI MSM514260A-70) containing the associated VPI/VCI values and possible TAG information, and a logic block which controls the LANCAM and DRAM to accomplish the translation and tagging functions.

To be able to operate this system at high clock rates, it needs to work according to a time-sliced principle (also see the timing implications). Only 27 clock cycles of the CLKI are available for each incoming cell. Both the basic operations and part of the background operations need to be completed in this limited time frame. The basic operation to be performed for every incoming cell is VPI/VCI translation and tagging, which take 13 of the 27 CLKI cycles. This operation is started as soon as the complete header is received and found to be error free. As soon as this operation is completed, other background operations can be executed to keep the LANCAM database up to date. The amount of time that is available to perform background

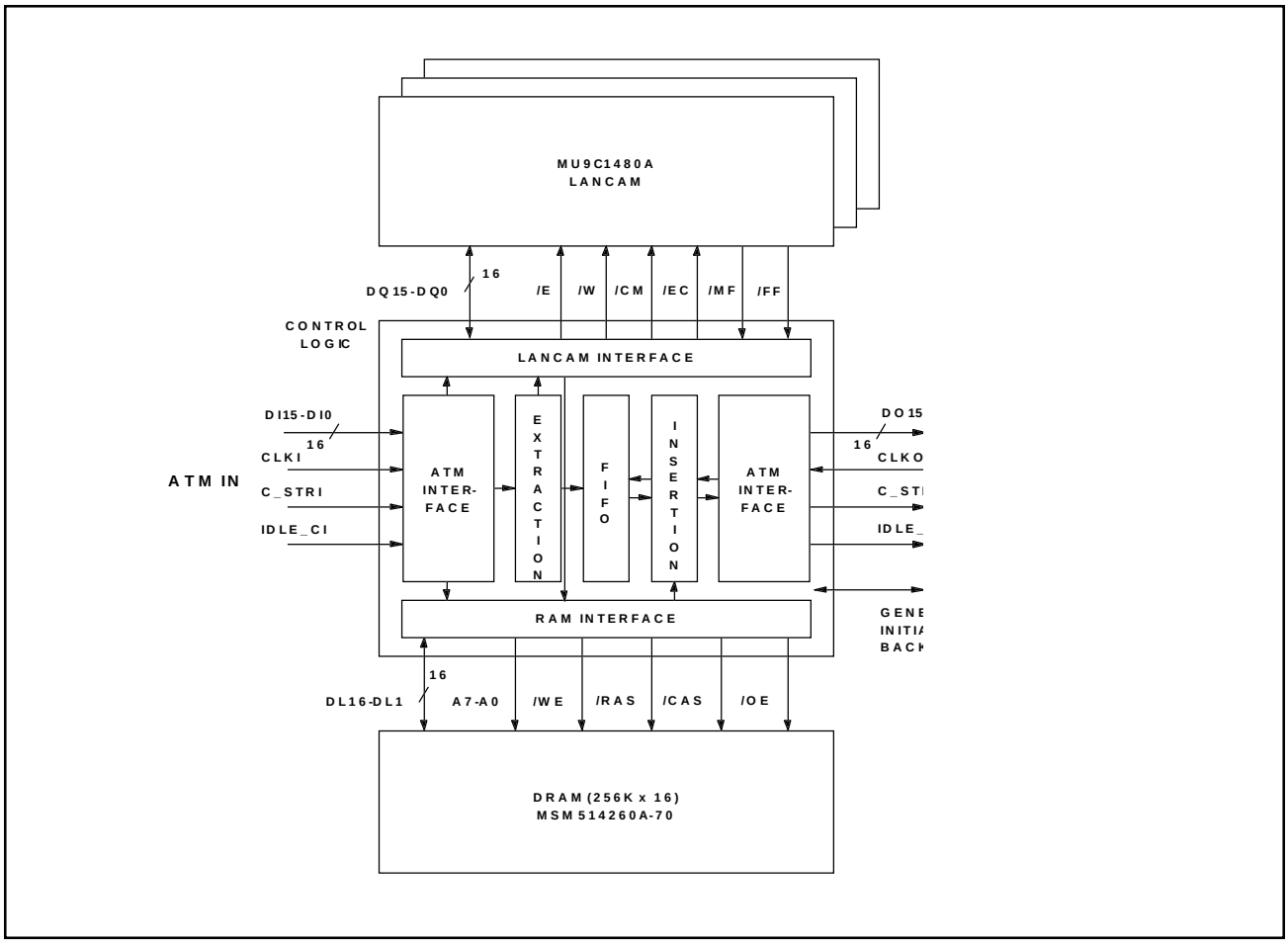

Figure 1: ATM VCI/VPI/ Translator Block Diagram

operations is limited to only 14 CLKI cycles per arriving cell. Feasible background operations during this time might include:

- VPI/VCI addition
- Associated data addition to a just added VPI/VCI
- VPI/VCI removal

In this example, due to timing restraints, the VPI/VCI translation, the VPI/VCI addition in the LANCAM database, and the associated data addition in the RAM combined with the check on duplicate VPI/VCI values can not be completed within the duration of one cell. The VPI/VCI addition will be done in one cell duration while the associated data addition is done one cell later. The VPI/VCI removal operation can be completed within one cell duration. This is not a severe limitation because the number of background operations per time interval will be much less than the amount of VPI/VCI translations performed in that same time interval. In the following section, the example code of the various operations is discussed, and the timing implications are shown.

EXAMPLE CODE

The code in this chapter are examples of four basic ATM operations in addition to the initialization of the LANCAMs:

- VPI/VCI lookup
- VPI/VCI addition associated data addition to a VPI/VCI
- VPI/VCI removal

The format of the routines shown in Figures 2 through 6 shows where CW, CR, DW, and DR indicate the cycle type. The next field shows the instruction or data to be passed to the LANCAM, and the following three fields show the state of the MI, FI, and EC pins. The initialization routine shown in Figure 2 is called INIT.DAT. It first programs a Page address in all LANCAMs. It programs the LANCAMs in 32-bit CAM/32-bit RAM mode and sets the Destination/Source counter limits accordingly.

The VPI/VCI lookup is done in routine LOOKUP.DAT, shown in Figure 3. This routine returns the address of the matching location, which can be used to get the associated VPI/VCI value and tag information out of the DRAM. The maximum amount of tag data that can be used is limited by the access time of the DRAM and the number of DRAMs used in parallel. In this example, the system has no DRAM accessed in parallel and has therefore only 64 bits of associated data, consisting of tag data and an associated VPI/VCI.

To add a VPI/VCI value to the database in the LANCAM, the routine ADD.DAT is used. This routine, shown in Figure 4, writes a new VPI/VCI into the Next Free address in the LANCAM. In the case that this VPI/VCI value already exists in the LANCAM database, an error situation will be created. The routine ASSOCIATED.DAT, illustrated in Figure 5, is used to check for the uniqueness of the VPI/VCI one cell duration later. If the VPI/VCI is unique, the associated data can be added. If the VPI/VCI is not unique, all VPI/VCIs with an equal value can be removed with the routine REMOVE.DAT.

To be able to add associated data, the address in which the new VPI/VCI was stored needs to be known. By performing a VPI/VCI lookup, the address where to store the associated data in the RAM is obtained together with the knowledge if the just added VPI/VCI value was unique. This extra lookup/check is done by routine ASSOCIATED.DAT. (Figure 5)

If a VPI/VCI needs to be removed from the database, because either the VPI/VCI value is stored more than once or the connection is released, the routine REMOVE.DAT is used. This routine, illustrated in Figure 6, removes all valid entries in the database equal to a specific VPI/VCI.

```
CW      0X0000      MI_HI    FI_LO    EC_HI

CW      TCO_DS      MI_HI    FI_LO    EC_HI
CW      0XFFFF      MI_HI    FI_LO    EC_HI
! Select all LANCAMs

CW      TCO_CT      MI_HI    FI_LO    EC_HI
CW      0X0000      MI_HI    FI_LO    EC_HI
! Reset all LANCAMs

CW      TCO_PA      MI_HI    FI_LO    EC_HI
CW      0XABCD      MI_HI    FI_LO    EC_HI
! Set page address first LANCAM

! Optional:  program the page addresses of the
other LANCAMs

CW      TCO_CT      MI_HI    FI_LO    EC_HI
CW      0X8080      MI_HI    FI_LO    EC_HI
! Use LANCAM in 32 bit CAM/ 32 bit RAM mode

CW      TCO_SC      MI_HI    FI_LO    EC_HI
CW      0X5850      MI_HI    FI_LO    EC_HI
! Set Destination Counter Limits 10B-11B
! Set Destination Counter to 10B
! Set Source Counter Limits 00B-01B
! Set Source Counter to 00B

! Default destination for the VPI/VCI is the
Comprand register
```

Figure 2: Init.dat Code Listing

```
DW      0XCCCC      MI_HI    FI_LO    EC_HI
DW      0XFFFF      MI_HI    FI_LO    EC_LO
! Write the VPI/VCI

CR      0X0000      MI_HI    FI_LO    EC_LO
CR      0X0000      MI_HI    FI_LO    EC_HI
! Read the Status register to get the matching
! address, which can be used as a RA< pointer, and
! check whether there are NO multiple matching
! locations (which indicates that the VPI/VCI
! combination exists multiple times)
```

Figure 3: Lookup.dat Code Listing

```
DW      0X1111      MI_HI    FI_LO    EC_HI
DW      0X3333      MI_HI    FI_LO    EC_HI
! Write the VPI/VCI
CW      MOV_NF,CR,V MI_HI    FI_LO    EC_HI
! Move the VPI/VCI, associated VPI/VCI
! combination into the Next Free address
```

Figure 4: Add.dat Code Listing

```
DW      0XCCCC      MI_HI    FI_LO    EC_HI
DW      0XFFFF      MI_HI    FI_LO    EC_LO
! Write the VPI/VCI

CW      0X0000      MI_HI    FI_LO    EC_LO
CR      0X0000      MI_HI    FI_LO    EC_HI
! Read the Status register to get the matching
! address, which can be used as a RAM pointer, and
! check whether there are NO multiple matching
! locations (which indicates that the VPI/VCI
! combination exists multiple times)
```

Figure 5: Associated.dat Code Listing

```
DW      0X0000      MI_HI    FI_LO    EC_HI
DW      0X0000      MI_HI    FI_LO    EC_HI
! Write the VPI/VCI

CW      VBC_ALM,E   MI_HI    FI_LO    EC_HI
! Make all locations, which contain thisVPI/VCI,
! empty
```

Figure 6: Remove.dat Code Listing

TIMING IMPLICATIONS

The time budget for LANCAM operations is limited. A VPI/VCI translation, which is performed for each received cell, consists of a LANCAM lookup operation done by LOOKUP.DAT (Figure 3 on page 4) to see whether the VPI/VCI value is known and to get a pointer to a RAM address where the associated VPI/VCI and tag data are stored. This LANCAM access is followed by a RAM access to find the associated and tag data. If no background operations are necessary, the system is idle for the rest of the cell duration. An illustration of this action is found in Figure 7. The VPI/VCI translate routine is started at the moment the complete header of the cell is received and the HEC indicates that the header is error free. If there is no matching location found in the LANCAM, the cell currently being received is discarded and replaced by an idle cell. If there is a matching location found, the address of that matching location is used as a pointer in the RAM to get the associated VPI/VCI and tag data. If there is a multiple match found, the VPI/VCI value exists more than once in the database, which is an erroneous situation. In that case, the VPI/VCI values have to be removed (by starting a REMOVE.DAT routine).

When a new VPI/VCI value has to be added to the database, the ADD.DAT routine is used, as shown in Figure 8. This routine adds a VPI/VCI value to the database, but because of timing restraints, it does not check for conflicts with existing VPI/VCI values and does not update the associated data relating to this VPI/VCI value. The installation of the associated data takes place in the next cell duration by running the routine ASSOCIATED.DAT.

The ASSOCIATED.DAT routine checks for VPI/VCI uniqueness and performs the loading of associated data. This exercise is illustrated in Figure 9 on page 7. To summarize, to add a new VPI/VCI value including associated data takes two cell times.

To remove a VPI/VCI value out of the database, routine REMOVE.DAT is executed. This clears the VPI/VCI value out of the LANCAM database and with that the pointer to the associated data in the RAM. This action is illustrated in Figure 10 on page 7.

GENERAL

In general it can be concluded that the MU9C1480A LANCAM is very well suited for various system functions in ATM systems up to 622 Mbps. It is a solution that does not suffer from limitiation in the range of VPI/VCIs used, distribution-dependent limitation in search speed or limitation in search speed depending on the number of VPI/VCI active in the system. The MU9C1480A LANCAM is also a very flexible part that can be used in most of the ATM systems, not necessary being limited by the internal interfaces used in the ATM system.

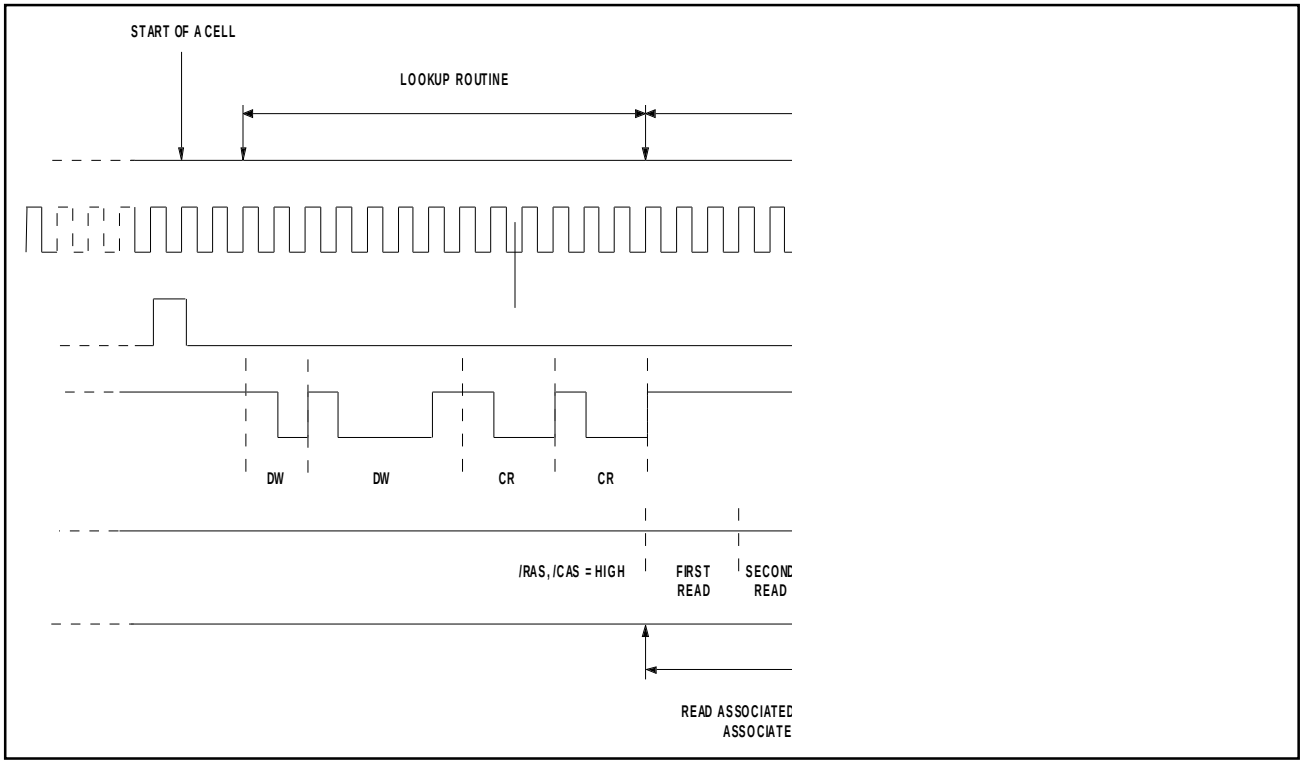

Figure 7: Lookup Timing Diagram

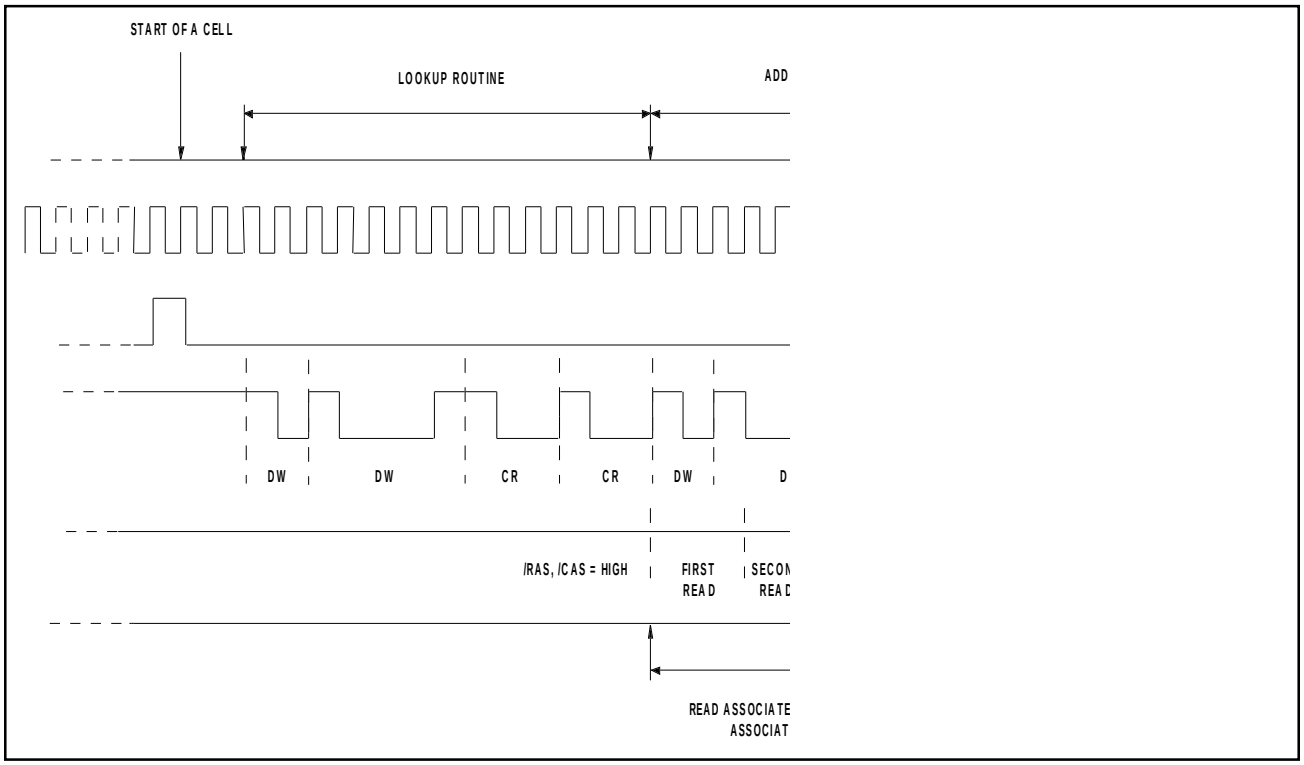

Figure 8: Lookup and Add Timing Diagram

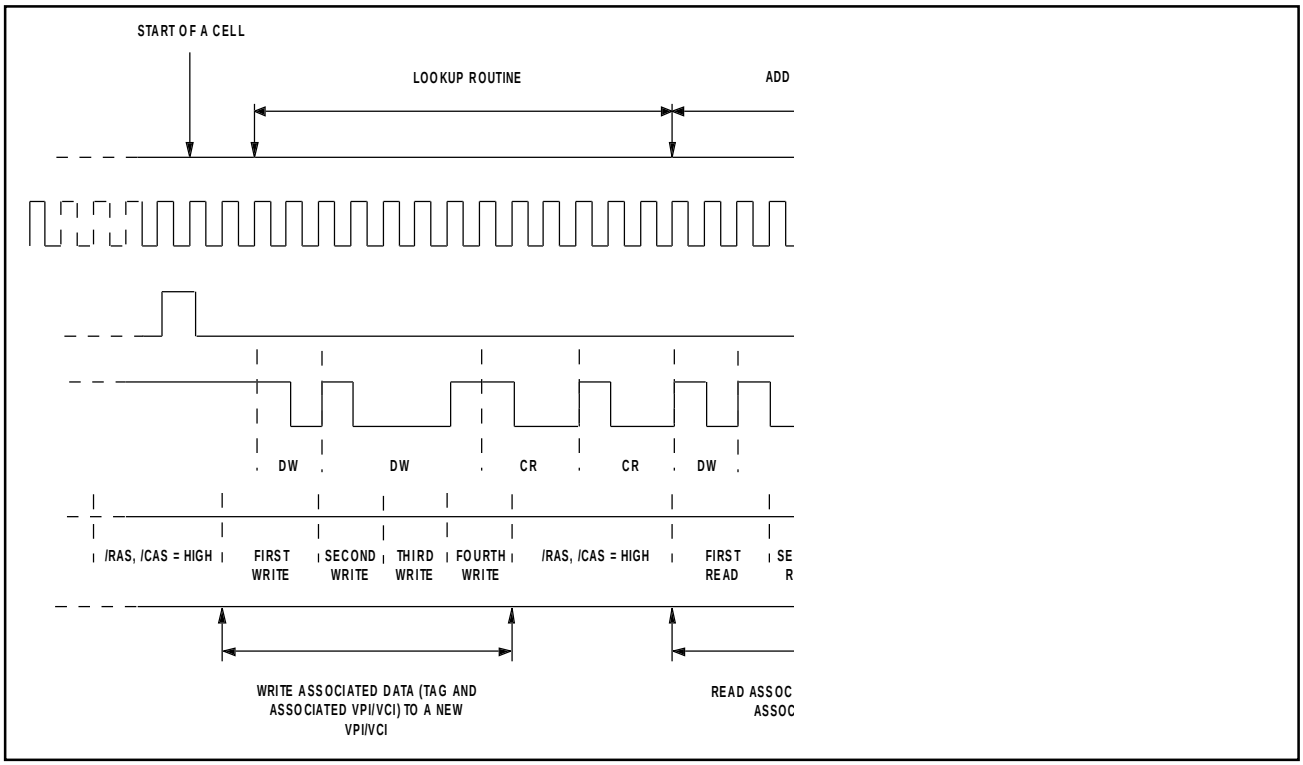

Figure 9: Load Associated Data Timing Diagram

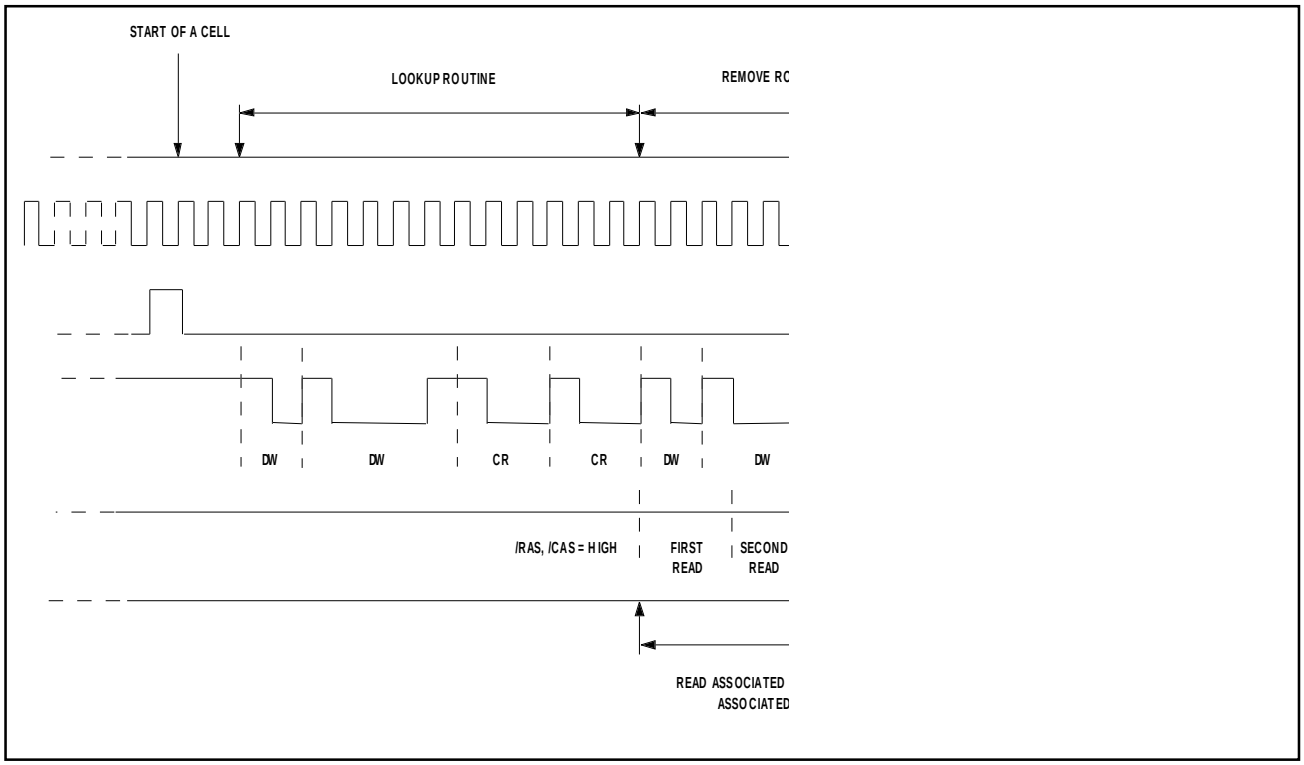

Figure10: Remove VCI/VPI Timing Diagram

NOTES

MUSIC Semiconductors Agent or Distributor:

MUSIC SEMICONDUCTORS

http://www.music-ic.com
email: info@music-ic.com

USA Headquarters
MUSIC Semiconductors
254 B Mountain Avenue
Hackettstown, New Jersey 07840
USA
Tel: 908/979-1010
Fax: 908/979-1035
USA Only: 800/933-1550 Tech. Support
888/226-6874 Product Info.

Asian Headquarters
MUSIC Semiconductors
Special Export Processing Zone 1
Carmelray Industrial Park
Canlubang, Calamba, Laguna
Philippines
Tel: +63 49 549 1480
Fax: +63 49 549 1023
Sales Tel/Fax: +632 723 62 15

European Headquarters
MUSIC Semiconductors
Torenstraat 28
6471 JX Eygelshoven
Netherlands
Tel: +31 45 5462177
Fax: +31 45 5463663