

FCC Ethernet Classifiers using Microcode

Author: Oded Gabbay

DoGav Systems Ltd.

December, 2005

Contents

1.	References.....	4
2.	Abbreviations.....	4
3.	Introduction.....	5
4.	Host – CPM Interface Definitions	6
4.1.	FCC Parameter RAM.....	6
4.2.	Classifier Array.....	6
4.3.	RxBD Rings Array.....	7
4.4.	uCode Structure	7
4.5.	Interrupt.....	7
4.6.	FPSMR.....	7
5.	Installation.....	8
5.1.	Package contents.....	8
5.2.	Trap usage.....	8

Revision history

<i>Version</i>	<i>Date</i>	<i>Modified by</i>	<i>Status</i>	<i>Approver</i>
1.0	20-Dec-05	Oded Gabbay	Draft	David Gabbay
1.1	21-Dec-05	Oded Gabbay	Draft	David Gabbay
1.2	26-Feb-06	Amy Dyson	Final	David Gabbay

List of Tables

Table 1. Abbreviations used in this document.....	4
Table 2. FCC Parameter RAM.....	6
Table 3. Classifier Structure	6
Table 4. RxBD Ring Structure.....	7
Table 5. uCode Structure	7

1. References

1. MPC8280 Reference Manual, Rev. 0, 3/2004.

2. Abbreviations

Table 1. Abbreviations used in this document.

Abbr.	Description
uCode	PowerQUICC's CPM Microcode
BD	Buffer Descriptor
RxBD	Receive Buffer Descriptor

3. Introduction

This uCode utility extends Freescale's PQII Ethernet protocol capabilities. It gives a performance boost by letting the CPM classify the incoming Ethernet frames according to defined patterns and masks. The classifier inserts the incoming frame into the corresponding RxBD ring and, if instructed to, generates an event. Frames that fail the match are discarded. The user can define a default ring for non-classified frames by setting one of the ring masks to all zero.

In order to optimize the uCode and to minimize the number of instructions, alignment and size limitations were applied. These limitations have no impact on the application, since the mask can be set up per bit. The classifier can be applied to the first 64 bytes of the frame. The current uCode version supports a classifier for a single FCC; the other FCCs operate normally.

The API consists of two structures: the classifier structure and the RxBD structure.

The classifier structure defines: the pattern to be matched; the mask which defines the "don't care" bits in the match operation; the offset that defines the first skipped bytes in the compare operation; the length of the compare pattern; and the index to the corresponding entry in the RxBD structure.

In case of match, the operation can be either accept or discard.

In the classifier structure array, the pattern that's expected to be most frequent should be on top, and the pattern that's expected to be least frequent should be on the bottom.

Currently, this utility is verified for a single buffer frame. The application should set up the MRBLR parameter to the largest expected received buffer size.

4. Host – CPM Interface Definitions

4.1. FCC Parameter RAM

Table 2 describes the changes to the FCC parameter RAM, which are required by the uCode module.

Table 2. FCC Parameter RAM

Offset	Name	Width	Description
0x00	RIPTR	Hword	Same function as in mpc8280 manual (Ref. 1). However, must point to a 64-byte area and the address must be divisible by 64.
0x02 – 0x0C	-	-	Same as in mpc8280 manual (Ref. 1).
0x0C	RBASE	Word	Initialize to the base of the first RxB ring.
0x10 – 0xFC	-	-	Same as in mpc8280 manual (Ref. 1).
0xFC	uCode_Strcut_Offset	Hword	Offset of the classifier array from start of DPRAM. Must be 128-byte aligned!
0xFE – 0xFF	-	-	Same as in mpc8280 manual (Ref. 1).

4.2. Classifier Array

The Classifier Array supports up to 8 classifiers. The array must be located in the DPRAM. Each entry in the array is 128 bytes long. The actual number of classifiers is the desired number + 1, since the last classifier signals the end of the array and is not taken into consideration. Table 3 describes the entry's structure.

Table 3. Classifier Structure

Offset	Name	Width	Description
0x00	queueNum	Hword	The queue for matched packets. If the value is equal or larger than 8, then discard the packet.
0x02	Reserved	Hword	Reserved.
0x04	offset	Hword	Start checking the pattern from beginning of packet + this offset. Must be divisible by 4.
0x06	length	Hword	How many bytes to compare with pattern. Must be divisible by 4. If this value is 0 (zero), then this classifier entry is not taken into consideration and signals the end of the classifier array.
0x08	RxFramCnt	Word	Received frames counter for this classifier.
0x0C – 0x20	Reserved	20 Bytes	Reserved.
0x20	Pattern	48 Bytes	The pattern to compare the packet with. Must be in little-endian form.
0x50	Mask	48 Bytes	The mask for comparison. Compare bit only if it is binary 1. If binary 0, ignore it.

4.3. RxBD Rings Array

The RxBD Rings Array supports up to 8 RxBD rings. The array must be located in the DPRAM. Each array entry is 8 bytes long. The first entry is indexed as “ring 0”, the second entry is indexed as “ring 1” and so forth. Table 4 describes the entry’s structure.

Table 4. RxBD Ring Structure

Offset	Name	Width	Description
0x00	RBASE	Word	RxBD base address.
0x04	RBPTR	Word	Points to the current BD inside the RxBD ring. Initialize to RBASE.

4.4. uCode Structure

The 16-byte uCode Structure is located at offset 0x8A00 from the DPRAM start. Table 4 describes the uCode structure.

Table 5. uCode Structure

Offset	Name	Width	Description
0x00	RxBD_Array_Base	Hword	Offset of the RxBD rings array from start of DPRAM. Must be 8-byte aligned!
0x02	LastQueueIndex	Hword	Microcode managed parameter; initialize to zero.
0x04	TmpReg1	Word	Microcode managed parameter; initialize to zero.
0x08	TmpReg2	Word	Microcode managed parameter; initialize to zero.
0x0C	Reserved	Word	Reserved.

4.5. Interrupt

Although the FCCE register is 16 bits, only its lower byte is used by the PQ2. This uCode utility uses the higher byte to notify the user of the RxBD rings’ incoming frames. It’s mapped as follows:

0	1	2	3	4	5	6	7	8	15
R7	R6	R5	R4	R3	R2	R1	R0	As defined in mpc8280 manual (Ref. 1)	

4.6. FPSMR

Once a uCode utility has been installed, it automatically works on all channels configured in the same mode – for example, all FCC channels that are configured to Ethernet. Therefore, we added a special bit to the FPSMR register to tell the uCode if it should run or not. We used bit number 7 (indicated as reserved in mpc8280 manual (Ref. 1)). The application should turn on this bit for a single FCC.

5. Installation

5.1. Package contents

This package comes as a ZIP compressed file containing the following files:

1. *PQ2_FCC_Eth_Classifiers.o* – This file contains the installation function and the uCode utility. Compiled using GNU compiler.
2. *InstallUcode.h* – This file contains function prototypes.
3. *FCC Ethernet Classifiers.pdf* – This document.

5.2. Trap usage

This uCode package occupies three traps.