+---------------+
! d i g i t a l ! I N T E R O F F I C E M E M O R A N D U M
+---------------+

TO: Dolphin Project List

DATE: 03 Nov 78
FROM: Alan Kotok
DEPT: LCEG
DTN #: 231-6381
LOC: MR1-2/E47
FILE: ECC-SPEC

SUBJ: ECC MCA Specification

Dolphin ECC MCA Functional Specification Revision 3

1.0 INTRODUCTION

The ECC MCA is a Macro Cell Array which is intended to be generally applied throughout the Dolphin System for the purpose of implementing a single error correcting, double error detecting, Hamming code on a 36 bit data word. The ECC chip has two basic modes: generation and checking.

In the generation mode, the ECC chip supplies the 7 check bits needed for "SEC/DED" (single error correction, double error detection). In the checking mode it takes in all 43 bits, and supplies error detection signals, as well as locating any single error. The ECC chip does not, however, correct a detected error. That function must be implemented in associated hardware.

Provision is also made for generating and checking parity on 18 bit half words and 9 bit quarter words, as ECC is stripped off or applied, respectively.

2.0 GOALS

1. To provide "SEC/DED" for all RAMs and Busses in the Dolphin.
2. To use a single code for all applications of ECC.
3. To use a single chip design for all applications of ECC.
4. To implement a code assignment for the data bits which provides the simplest possible decoding of the syndrome (location of an erroneous bit) in bit sliced data structures.
5. To perform the checks required as fast as possible.
6. To allow the "syndrome" outputs to drive a bus of syndromes and other data.

3.0 THEORY

The ECC chip performs SEC/DED by use of the Hamming Code with an overall parity bit appended for double error detection. In short, the Hamming code locates a single error by having each respective bit included in a well-chosen and unique set of parity checks. The resulting parity bits are sent along or stored in addition to the data bits. When the resulting elongated word is checked later, the same parity checks are recomputed. A single error will cause "parity errors" to appear in the same "parity checks" as were originally computed. Since no two bits are included in the same set of parity checks, the pattern of parity

errors locates the offending bit, which can be corrected by complementing it.

The Hamming Code further prescribes a simple assignment of the parity checks such that the pattern of parity errors (called the "syndrome") can be treated as a binary number to locate the offending bit. Thus, each data bit is given a binary number, and is included in the set of parity checks where ones exist in the value of its bit number. For instance, bit 15 (octal) would be included in parity check 10, 4 and 1. This can be seen to be a unique assignment. If the bit numbers in the resulting code which have a single one in them (1, 2, 4, 10, etc.) are reserved for the check bits themselves, then the check bits are not a function of any other check bits, and are easily computed.

There is, however, a degree of freedom in the assignment of bit numbers. Namely, it is not either neccessary, desirable, or possible to use the "natural" bit numbers (0 to 35 in our case) as the code assignments. Therefore, the ECC chip implements an arbitrary assignment chosen for other reasons, but still consistent with the goals.

Double error detection is provided by the inclusion of an overall parity bit covering both the data and check bits. Any single error will perforce cause the overall parity to be wrong. By avoiding assigning code slot 0, any single error will also cause a non-zero syndrome. The absence of errors will cause a zero syndrome and correct overall parity. A double error will cause "correct" overall parity but a non-zero syndrome. This is because two errors cannot occur on the same bit, so the exclusive-or of the two bit numbers cannot be zero.

4.0 CODE ASSIGNMENT

4.1 Slice Constraints

4.1.1 Four And Eight Bit Slices -

We want to be able to handle the correction of erroneous bits in the "bit sliced" data path of the Dolphin. Slice sizes of 4 and 8 bits seem necessary, along with provision for handling half the bits from each half word. For example a slice might have bits 0, 1, 18 and 19. It is therefore desirable to make the logic on each slice be able to compare its slice number against some part of the syndrome, and, if equal, apply the correction to one of its inputs uniformly as decoded from the remaining bits of the syndrome.

Furthermore, the ability to apply a correction to a check bit is frequently necessary, so that the full 43 bit word may be propagated correctly to the next logic stage. As discussed previously, the check bits have values of integer powers of 2. Thus, the bit slice scheme mentioned above would require

complicated logic to detect these special cases. Therfore, a mode is provided in the ECC chip to detect the unusual cases, and "re-map" the syndrome into a more easily decoded form, including errors in the check bits.

For ECC on the Dolphin Bus, it is desired to include the "TAG" and "ID" bits in the same checking word as the data. Since there are ample unused slots in the code assignments, 4 of these have been committed to the TAG, and 6 to the ID. These 10 bits are not included in the checking done by the ECC chip, but are added in later by an auxilliary chip on generation of the check bits, and are subtracted out during error detection.

4.1.2 Six Bit Slices -

The IBOX and EBOX use 6 bit slices, and carry half word parity, but no check bits. It is therefore desireable to provide syndrome decoding to allow easy correction on 6 bit slices. It is also desireable to supply 2 parity bits on the corrected data for data bits 0-17 and 18-35. When generating check bits, the half word parities are computed, including the parity bits. The 6 bit slice coding is shown below.

4.2 Basic Code

The basic code assignment is shown first below. The coordinates are the bit numbering in the ECC code. The overall parity bit is arbitrarily assigned as code slot zero, since it is not included in any of the normal parity checks. The "C"s are the check bits, numbered in octal, the "D"s are the PDP-10 data bits numbered in decimal, the "I"s are the ID field numbered by binary weight in decimal, and the "T"s are the TAG field, similarly numbered.

	0	1	2	3	4	5	6	7
00	C0	C1	C2		C4			
10	C10	T2	T4	T8	D00	D01		
20	C20				D18	D19		
30	D02	D03	D04	D05	D06	D07	D08	D09
40	C40	I1	I2	I4	I8	I16	I32	T1
50	D10	D11	D12	D13	D14	D15	D16	D17
60	D20	D21	D22	D23	D24	D25	D26	D27
70	D28	D29	D30	D31	D32	D33	D34	D35

The check bits supplied are normally the mod-2 sums of the data bits whose code slot includes the binary weight of the check bit. Thus a data word of all zeroes would yield all zero check bits. (Check bit C0 is the overall parity, and hence would also be zero). In order to avoid a null word checking as good, it is desireable to complement some of the check bits, in order to force an all zeroes received word to be a double error. It is similarly

desireable to have all ones be detected as a double error. In order to force all zeroes to be a double error, an even number of check bits must be complemented, but since there are an odd number of check bits, the all ones case will have an odd number of errors. It was therefore necessary to carefully choose which check bits to complement, so that the syndrome of an all ones word is not one of the assigned code slots. Since each check bit includes an even number of the 36 data bits, both all ones and all zeroes would give all zero check bits. The return of all ones, including check bits would yield a syndrome of 77, and appear as a single error. By complementing, say, C40 and C10, the syndrome for all ones is 27, which is not otherwise used.

The result of all this is that the ECC MCA complements C40 and C10 on output, and re-complements them when received. It also specifically tests for syndrome 27, and flags it as a double error.

Since it is not desired to have to drive the Dolphin Bus to a non-zero value during idle cycles, and there is a confirmation mechanism to notice an all zeroes word, the Bus Interface chipset will re-compelment C40 and C10 back to their even-parity state.

4.3 Syndromes

4.3.1 8 Bit Mode -

The 4 and 8 bit sliced "re-mapped" syndrome groups together the check bits in row 00. Under diagnostic mode control, bits D18 and D19 are movable from code slots 24 and 25 to 16 and 17 to allow easier 8 bit slicing.

	0	1	2	3	4	5	6	7
00		$C\overline{1}$	$C\overline{2}$	$C\overline{10}$	$C\overline{4}$	$C\overline{20}$	$C\overline{40}$	$C\overline{0}$
10		T2	T4	T8	D00	D01	<D18>	<D19>
20					(D18)	(D19)		
30	D02	D03	D04	D05	D06	D07	D08	D09
40		I1	I2	I4	I8	I16	I32	T1
50	D10	D11	D12	D13	D14	D15	D16	D17
60	D20	D21	D22	D23	D24	D25	D26	D27
70	D28	D29	D30	D31	D32	D33	D34	D35

4.3.2 6 Bit Mode -

The 6 bit "re-mapped" syndrome causes all check bit errors to yield zero syndromes. This means that check bits cannot be corrected when using 6 bit slices.

	0	1	2	3	4	5	6	7
00								
10	D00	D01	D02		D03	D04	D05	
20	D06	D07	D08		D09	D10	D11	
30	D12	D13	D14		D15	D16	D17	
40	D18	D19	D20		D21	D22	D23	
50	D24	D25	D26		D27	D28	D29	
60	D30	D31	D32		D33	D34	D35	
70								

5.0 FUNCTIONS

The ECC MCA performs a variety of related functions, controlled by the Diagnostic Status Register. It is thus necessary to load this register before using the ECC MCA. The definitions of the bits in the Status Register are as follows:

Bit(s) Value Function

0 (MSB) 0 NORMAL MODE: Causes bits 1-7 to control functions.

1 DIAGNOSTIC MODE: Causes Status Register bits 1 thru 7 to be output as Check or Syndrome bits as follows:

Status Bit	Output
1	C40
2	C20
3	C10
4	C4
5	C2
6	C1
7	C0

Also, input latches are forced into the LOAD mode, and the CONTROL input pin has no effect.

1-2 10 CHECK BIT MODE: Check bits are generated from the 36 DATA inputs, and output on lines C40 thru C0. Parities of partial words (as selected below) are computed and appropriate error indications are given on the parity output lines. (See pinout section for details.)

11 CHECK OR SYNDROME MODE: Check bits are output if the CONTROL input pin is LOW, or if a double error is detected. Syndromes are output if the CONTROL input is HIGH and no double error is detected. The parity lines will be as described above if in Check Bit mode, and as below if in Syndrome mode. The Single and Double error lines will be meaningless if the CONTROL signal is low.

	00	GATED SYNDROME MODE: Syndromes are always generated, but are only output on lines C40 thru C1 if the CONTROL input is HIGH. Single and Double error indications are always given. The parity output lines will be the mod-2 sum of the ones on the selected parital data words, and will be correct even in the face of a single error.
	01	SYNDROME LATCH MODE: Syndromes are always generated and output. Error and Parity lines are as described in GATED SYNDROME MODE. The input DATA and CHECK BIT lines are held latched if the CONTROL line is LOW, and are passed thru if the CONTROL line is HIGH. The CONTROL line is intended to be connected to the system clock.
3-4	00	(UNUSED, reserved for future use)
	01	SIX BIT SLICE MODE: If a syndrome mode has been selected, the syndrome will be re-mapped in accordance with the 6 bit slice mapping shown above.
	10	EIGHT BIT SLICE (BITS 18 and 19 MOVED) MODE: If a syndrome mode has been selected, the syndrome will be remapped in accordance with the 8 bit slice map as shown above. Code slots 24 and 25, corresponding to data bits 18 and 19 will be moved to code slots 16 and 17.
	11	EIGHT BIT SLICE MODE: If a syndrome mode has been selected, the syndrome will be rempaped in accordance with the 8 bits slice map as shown above. Code slots 24 and 25, corresponding to data bits 18 and 19 will remain in that position.
5	0	HALF WORD PARITY MODE: In check bit mode, parity is computed on the two half words of input data, and XORed with the input half word parity lines. The results are output on the output parity lines. Also, a test is made on input data bits 0 and 6 thru 17 to test for "positive non-zero left half" to detect a GLOBAL index word. This signal is output on one of the "parity" lines. In syndrome mode, the half word parity output signals are the mod-2 sum of the half words of data, complemented if the syndrome identifies an error in that half word such that the parity will be correct when the data is corrected.
	1	QUARTER WORD PARITY MODE: In check bit mode, parity is computed on the four quarter words of input data, and XORed with the input quarter word

parity lines. The OR of these signals is output on one of the output parity lines. In syndrome mode, the quarter word parity output signals are the mod-2 sum of the quarter words of data, complemented if the syndrome identifies an error in that quarter word such that the parity will be correct when the data is corrected.

6	0	ENABLE ERROR MODE: Allows single and double error signals to be generated. This is the normal mode.
	1	DISABLE ERROR MODE: Suppresses both single and double error detection. Forces the Single and Double Error lines false.
7		(UNUSED) Reserved for the future.

6.0 PINOUTS

At this point, actual pins have not been assigned, and hence "U" numbers will be shown to allow reference to the drawings.

Pin	Function
U1-36	DATA INPUTS: These are the 36 data input lines. It is, of course, necessary to precisely follow the bit numbering shown. These inputs are latched when appropriate.
U37	CHECK BIT 0 INPUT: Used only when computing syndromes. This is the overall parity bit of the 43 bit word. This input is latched when appropriate.
U38	In syndrome mode, this is CHECK BIT 1 INPUT. In check bit mode, this is PARITY 00-08 INPUT used in QUARTER WORD PARITY MODE. This input is latched when appropriate.
U39	In syndrome mode, this is CHECK BIT 2 INPUT. In check bit mode, this is PARITY 09-17 INPUT used in QUARTER WORD PARITY MODE. This input is latched when appropriate.
U40	In syndrome mode, this is CHECK BIT 4 INPUT. In check bit mode, this is PARITY 18-26 INPUT used in QUARTER WORD PARITY MODE. This input is latched when appropriate.
U41	In syndrome mode, this is CHECK BIT 10 INPUT. In check bit mode, this is PARITY 27-35 INPUT used in QUARTER WORD PARITY MODE. This input is latched when appropriate.
U42	In syndrome mode, this is CHECK BIT 20 INPUT. In check bit mode, this is PARITY 00-17 INPUT used in HALF WORD PARITY MODE. This input is latched when appropriate.

U43 In syndrome mode, this is CHECK BIT 40 INPUT. In check bit mode, this is PARITY 18-35 INPUT used in HALF WORD PARITY MODE. This input is latched when appropriate.

U44 In syndrome mode, this is SYNDROME BIT 40 OUTPUT. In check bit mode, this is CHECK BIT 40 OUTPUT.

U45 In syndrome mode, this is SYNDROME BIT 20 OUTPUT. In check bit mode, this is CHECK BIT 20 OUTPUT.

U46 In syndrome mode, this is SYNDROME BIT 10 OUTPUT. In check bit mode, this is CHECK BIT 10 OUTPUT.

U47 In syndrome mode, this is SYNDROME BIT 4 OUTPUT. In check bit mode, this is CHECK BIT 4 OUTPUT.

U48 In syndrome mode, this is SYNDROME BIT 2 OUTPUT. In check bit mode, this is CHECK BIT 2 OUTPUT.

U49 In syndrome mode, this is SYNDROME BIT 1 OUTPUT. In check bit mode, this is CHECK BIT 1 OUTPUT.

U50 In syndrome quarter word mode, this is PAR OUT 00-08. In syndrome half word mode, this is PAR OUT 00-17. In check bit quarter word mode, this pin is unused. In check bit half word mode, this is PAR ERR 00-17.

U51 In syndrome quarter word mode, this is PAR OUT 09-17. In syndrome half word mode, this is PAR OUT 18-35. In check bit quarter word mode, this pin is unused. In check bit half word mode, this is PAR ERR 18-35.

U52 In syndrome quarter word mode, this is PAR OUT 18-26. In syndrome half word mode, this pin is unused. In check bit quarter word mode, this pin is true on ANY QUARTER WORD PARITY ERROR. In check bit half word mode, this pin is true if the data is a GLOBAL index word.

U53 In syndrome quarter word mode, this is PAR OUT 27-35. In syndrome half word mode, this pin is unused. In check bit mode, this pin is C0 OUT.

U54 In syndrome mode, this is DOUBLE ERROR. In check bit mode, this pin is unused.

U55 In syndrome mode, this is SINGLE ERROR. In check bit mode, this pin is unused.

U56 This is the CONTROL input. In CHECK OR SYNDROME MODE, a HIGH indicates syndrome mode, and LOW indicates check bit mode. In GATED SYNDROME MODE, a HIGH enables the 6 syndrome output lines, and a LOW disables them. In SYNDROME LATCH MODE, a HIGH allows the data to pass through the latches, and a LOW hold the data in the latches.

U57 Spare.

U58-60 Diagnostic lines. Will be the standard pins and functions.