ABEX TEST CD-ROM

TCDR-701

SCANNING VELOCITY 1.3m/sec.

ALMEDIO INC.

CONTENTS

1. The Structure of This Disc

1-1 Construction

Table 1 shows the overall structure of this disc TCDR-701

Table 1: The Structure of CD-ROM TEST DISC TCDR-701

Area	Type of content	User Data	Block address (length)	Number of blocks	Sequential Block Number	Mode	Sub code *	
							TNO	Index
Lead-in	Digital mute	—	M S B —	—	—	—	00	—
Gap	Type A	All zero	00 00 00 (00 02 00) 00 01 74	150	—	1	01	00
Data area	Type B	Specified	00 02 00 (60 00 00) 60 01 74	270,000	1 270,000			01
Lead-out	Type A	All zero	—	—	—		AA	

Where: Gap + Data area = Information area

TNO: Track number

Block address: Minute, Second, Block

1 Block = 2 K bytes = 2048 bytes

1 Second = 75 Blocks

1 Minute = 60 Seconds = 4500 Blocks

60 Minutes = 3600 Seconds = 270,000 Blocks

* Subcode is specified by Compact Disc format.

☆ 1.3m/sec was used as the scanning velocity of this disc.

1-2 Contents of Type A

Table 2 shows the contents of Type A blocks where Mode 1 was used.

Table 2: The contents of Type A blocks

<table>
<tr><td>Sync</td><td colspan="2">'00 FF FF FF FF FF FF FF FF FF FF 00'</td><td>12 bytes</td></tr>
<tr><td rowspan="4">Header</td><td rowspan="3">Block address</td><td>Minutes in BCD (1byte)</td><td rowspan="4">4 bytes</td></tr>
<tr><td>Seconds in BCD (1byte)</td></tr>
<tr><td>Blocks in BCD (1byte)</td></tr>
<tr><td colspan="2">Mode '01' (1byte)</td></tr>
<tr><td>User data</td><td colspan="2">All bytes are '00' (HEX notation)</td><td>2048 bytes</td></tr>
<tr><td rowspan="4">Auxiliary data</td><td colspan="2">Error detection code: EDC</td><td>4 bytes</td></tr>
<tr><td colspan="2">All bytes are '00' (HEX notation)</td><td>8 bytes</td></tr>
<tr><td rowspan="2">ECC</td><td>P parity (26, 24) Reed Solomon codes</td><td>172 bytes</td></tr>
<tr><td>Q parity (45, 43) Reed Solomon codes</td><td>104 bytes</td></tr>
</table>

ECC: Error correction codes

1-3 Contents of Type B

Table 3 shows the contents of Type B blocks where Mode 1 was used.

Table 3: The contents of Type B blocks

Sync	'00 FF FF FF FF FF FF FF FF FF FF 00'		12 bytes
Header	Block address	Minutes in BCD (1byte)	4 bytes
		Seconds in BCD (1byte)	
		Blocks in BCD (1byte)	
	Mode '01' (1byte)		
User data	'Test data'		2048 bytes
Auxiliary data	Error detection code		4a bytes
	All bytes are '00' (HEX notation)		8 bytes
	ECC	P parity (26, 24) Reed Solomon codes	172 bytes
		Q parity (45, 43) Reed Solomon codes	104 bytes

2. The Test Data

2-1 Contents of The Test Data

Table 6 on last page shows The Test Data in CD-ROM TEST DISC TCDR-701

Test Data can be grouped into the following three groups,

(a) Block number informations (b) M-sequence

(c) Check sum

24 Bytes Data from the first of User Data show Block number information. The relationship, in this disc, between the Sequential Block Number and the Block Address in decimal is shown in the next formula.

Sequential Block Number

$$= (MIN \times 60 + SEC) \times 75 + Block + 1 - 150 *$$

Where: MIN = Minute, SEC = Second

* Note: 150 is the Gap which is (2 seconds × 75 blocks).

2-2 M-Sequence Random Data

In order to generate the pseudo random sequence data, M-sequence (Maximum-length Linear Feed-back Shift Register Sequence) was used.

Primitive Polynomial = '1E0000401' (HEX notation)

Expression of Data

MSB : left LSB: right

MSB : The most significant bit

LSB : The least significant bit

Initial value: Sequential Block Number of the block

Direction of bit shift: Towards lower bit

《Generation Process》

(1) The primitive polynomial is shifted by one bit to the right, and the result is stored into IFED (32 bits data).
IFED = ' F0000200 ' (HEX notation)

(2) A 32 bit working register is stored with the Sequential Block Number.

(3) If the least significant bit of the working register is 1 then
flag LSBF = 1, else flag LSBF = 0.

(4) The working register is shifted by one bit to the right bringing 0 into the most significant bit.

(5) If LSBF = 1, the working register is Exclusive-ORed with the IFED and replaced by the result.
If LSBF = 0, the working register will be left unchanged,

(6) The working register is ANDed with the ' FFFF ' (HEX notation), in order to get the lower 16 bits as the two bytes of the result.
The lower (higher) byte of the result is stored into the lower (higher) address.

(7) Keeping the working register unchanged, return to process (3) for the next address value.
This process is repeated 1009 times to generate the data of bytes 24 to 2043 of the user data in the block.

☆Example of Sequential Block Number=1

(1) IFED=' F0000200 '

IFED	F0	00	02	00

(2) Set' 00000001 ' in working register: WR.

WR	00	00	00	01

(3) Check LSB, LSB=1 then LSBF=1.

WR	00	00	00	01

↑ 1 ⟶ LSBF=1
0 ⟶ LSBF=0

(4) Shift data, then WR=' 00000000 '.

0 ⟶ | 00 | 00 | 00 | 00 | ⟶

(5) LSBF=1, then WR is Exclusive-ORed with IFED.

WR	F0	00	02	00

(6) Data (24)=' 00 ' Data(25)=' 02 '

WR	F0	00	02	00

⟶ DATA (2×I)
⟶ DATA (2×I+1)

(7) Keeping WR=' F0000200 ', return to process(3).

2-3 Check Sum Data

In Order to check data within the User Data, Check Sum was recorded in the last two bytes (16 bits) of this area.

The Check Sum is achieved by considering 16 bits as 1 word in the User Data and accumulating all the words besides the Check Sum bytes, and taking the lower 16 bits (2 bytes) as the result.
The lower byte of this result stored into byte number 2046 of the User Data, and the higher byte into 2047.

* The translation rule from Byte values into Word values is;

$$\text{Word}(N) = \text{Byte}(2\times N) + 256\times \text{Byte}(2\times N+1)$$
$$(N = 0, \cdots\cdots\cdots, 1023)$$

* Example program for Check Sum generation

```
      INTEGER*2  IWORD (1024)
      ISUM=0
      DO 10  N=1,  1023
      ISUM=IWORD (N)+ISUM
10    CONTINUE
      ISUM=ISUM  .AND. 'FFFF' (HEX notation)
      IWORD (1024)=ISUM
      STOP
      END
```

3. Sample Data

3-1 Example of Type A block

Table 4: Data of Block address 00 MIN 00 SEC 00 Block

Address	Data (HEX notation)															
	〈Sync and Header〉															
	Sync												Header			
0000	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	00	00	01
	〈User Data〉															
0016	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0032	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0048	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0064	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0096	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0112	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0128	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0144	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
.																
.																
.																
2032	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
2048	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

3-2 Example of Type B block

Table 5: Data of Block address 00 MIN 02 SEC 00 Block
Sequential Block Number=1

Data (HEX notation)

Address	〈Sync and Header〉 Sync											Header			
0000	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	02	00	01
	〈User Data〉															
0016	01	00	00	20	01	00	00	20	20	00	02	00	20	30	30	6D
0032	30	32	73	30	30	66	20	20	00	02	00	01	80	00	40	00
0048	20	00	10	00	08	00	04	00	02	00	01	00	00	02	00	01
0064	80	00	40	80	20	C0	10	E0	08	F0	04	78	02	3C	01	1E
0080	00	0D	80	06	40	03	A0	81	D0	C0	68	E0	34	F0	1A	78
0096	0D	3C	06	1C	03	0E	01	05	80	00	40	80	20	C0	10	E0
0112	08	F0	04	78	02	3C	01	1E	00	0D	80	06	40	83	A0	C1
0128	D0	60	68	B0	34	58	1A	2C	0D	96	06	49	83	24	41	10
0144	20	0A	10	85	88	C2	44	E1	A2	F0	51	78	28	3E	14	1F
.																
.																
.																
2016	F4	9E	7A	4F	BD	27	DE	91	EF	C8	77	E6	3B	F1	9D	7A
2032	4E	BF	A7	5F	D3	AD	E9	D4	74	68	3A	34	1D	1A	0E	0F
2048	87	87	C3	C1	E1	62	70	B3	B8	D9	DC	6C	20	20	CD	6B
															Check Sum	

1. テストディスクの概要

1-1 テストディスクの内容

テストディスクTCDR-701はTable 1 (3頁) の構成で記録されています。
＊Sub codeは、コンパクトディスクのフォーマットで定められているものです。
☆走査速度は1.3m/secで記録されています。

1-2 タイプAの内容

データ　タイプAはMODE 1で記録されており、そのブロック内容を
Table 2 (4頁) に示します。

1-3 タイプBの内容

データ　タイプBはMODE 1で記録されており、そのブロック内容を
Table 3 (5頁) に示します。

2. テストデータの概要

2-1 テストデータの構成

CD-ROMテストディスクTCDR-701のデータの内容を最終ページのTable 6に示します。
テストデータは大別して、(a)ブロック番号情報、(b)M系列、ランダムデータ、(c)チェックサムデータからなっています。

ブロック番号情報はUser Dataの先頭から24バイトの中にあります。
この情報は、以下のコードで記録されています。

Sequential Block Number : binary, ASCII, BCD
Block Address : MIN, SEC, Block

このディスクでいう、Sequential Block Number とBlock Addressとの関係をつぎに示します。

Sequential Block Number
$= (MIN \times 60 + SEC) \times 75 + Block + 1 - 150$ *

* Note: 150はgapの2秒×75 Blockによるものです。

MIN=Minute, SEC=Second

2-2 M系列 ランダムデータ

M系列ランダムデータを発生する際の生成多項式は、次の関数を使用しています。

Polynomial='1E0000401' (HEX notation)

データ表現は、最上位ビットを左にし、最下位ビットを右にする表示方法を採用しています。
M系列の初期値は、各ブロックのSequential Block Numberを用いています。
ビットシフトの方向は下位ビット側へシフトするようにしています。

MSB : The most significant bit
LSB : The least significant bit

《発生プロセス》

(1) 最初に、生成多項式 (Polynomial) を、1ビット 下位ビット側へシフトした値、IFED (32 bits data)を作ります。

IFED='F0000200' (HEX notation)

(2) 32ビットワークレジスタに、Sequential Block Numberをセットします。

(3) ワークレジスタの最下位ビットをチェックし、もし1ならLSBF=1のフラグを立て、0ならLSBF=0とします。

(4) レジスター内のデータを、1ビット 下位ビット側へシフトし、最上位ビットには0をセットします。
(最下位ビットのデータは捨てることになります。)

(5) LSBF=1の場合は、シフト後のデータとIFEDとの排他的論理和をとり、その結果をワークレジスタにセットします。LSBF=0なら、内容は変更しません。

(6) (5)のデータを、'FFFF' (HEX notation)で論理積をとり、下位ビット側の16ビットのみを、2バイトのデータとして利用し、下位バイトを
lower addressにセットします。

(7) (5)で演算した結果をもとにして、つぎのデータを発生させるために(3)に戻ります。
この演算を1009回繰り返し、1 block内のデータを完成させます。

☆Sequential Block Number=1の例を8頁に示します。

2-3 チェックサムデータ

User data内のデータエラーを確認するために、その最後の2バイト (16ビット) にチェックサムを記録しています。
チェックサムの仕方は、チェックサムエリアを除くuser data全域に対し、16ビットを1ワードとして考え、16ビットの累積加算を行ないます。
その結果の下位16ビットの内、下位1バイトをbyte number2046に、上位1バイトをbyte number 2047にセットしています。

＊Byte valueをWord valueに変換する式は次の通りです。

$$\text{Word}\ (N) = \text{byte}\ (2\times N) + 256\times \text{byte}\ (2\times N+1)$$
$$(N=0, \cdots\cdots, 1023)$$

＊チェックサム生成のプログラム例を9頁に示します。

3. サンプル データ

3-1 タイプAの一例

00分00秒00ブロックのデータをTable 4 (10頁) に示します。

3-2 タイプBの一例

00分02秒00ブロックのデータをTable 5 (11頁) に示します。

Table 6: THE TEST DATA in CD-ROM TEST DISC TCDR-701

Group	Byte number in user data	Contents	Code
(a)	0 1 2	Lower byte Mid byte Higher byte — of Sequential Block Number	binary
	3	Space code (HEX notation 20)	ASCII
	4 5 6	2 lower digits 2 mid digits 2 higher digits — of Sequential Block Number	BCD
	7, 8	Space code (HEX notation 20)	ASCII
	9 10 11	MIN (2 digits) SEC (2 digits) Block (2 digits) — of the block address	BCD
	12	Space code (HEX notation 20)	ASCII
	13 14	Higher digit Lower digit — MIN of the block address	
	15	Character "m" code	
	16 17	Higher digit Lower digit — SEC of the block address	
	18	Character "s" code	
	19 20	Higher digit Lower digit — Block of the block address	
	21	Character "f" code	
	22, 23	Space code (HEX notation 20)	
(b)	24 · · 2043	M-sequence ($2^{32}-1$) starting from the Sequential Block Number	binary
(c)	2044, 2045 2046 2047	Space code (HEX notation 20) Lower byte Higher byte — of Check Sum	ASCII binary