

데이터 전송 효율을 고려한 4비트행x4비트열 2 바이트 문자 부호화 규칙에 관한 연구

Coding Rule of Characters by 2 bytes with 4x4 bits to Improve the Transmission Efficiency in Data Communications

홍완표*

Wan-Pyo Hong*

요 약

본 논문은 컴퓨터 등 정보기기에 입력되는 문자나 기호의 부호화 규칙을 제시하였다. 문자나 부호를 부호화할 때 본 논문에서 제시하는 규칙을 적용할 경우, 데이터를 전송로에 전송할 때 회선부호화과정에서 발생하는 스크램블링의 발생으로 인한 전송효율이 떨어지는 것을 감소시킬 수 있다. 본 논문에서는 4비트열 x 4비트행의 2 바이트 부호화에 대한 것을 고찰하였다. 본 논문에서는 2바이트(16비트) 원천부호체계로 KS X 1001에 규격화되어 있는 로마 문자용 부호집합에 대한 부호체계를 참고로 하였다.

Abstracts

This paper propose the rule of coding for the characters and symbol, etc which are used in computer, information devices, etc. When they use the rule of coding, they may improve the efficiency of transmission in data communications by reducing the number of scrambling during the line coding in the coder in the transmitter. The paper considered the codes of two bytes(16 bits) of 4bits column x 4bits rows. In the paper, we applied the code system of Roman characters in KS X 1001.

Key words : 원천부호화, 회선부호화, 전송효율, ASCII, HDB-3

Source coding, Line coding, Transmission efficiency, ASCII, HDB-3

I. 서 론

정보기기에 입력되는 문자나 기호는 정해진 부호로 부호화되어 저장되거나 프린터 등을 통하여 출력된다. 정보기기에 입력되는 문자나 기호를 일정한 체계의 부호로 변환하는 것을 원천 부호화(source coding)[1]라고 한다.

대표적인 원천부호체계는 ASCII[2][3][4], EBCDIC[5], Unicode[6], UTF[7] 등이다. 정보기기에 입력되어 정보기기에 저장되거나 입력되는 데이터는 인터넷 등 네트워크를 통해 원거리에 전송된다. 즉, 인터넷에 전송되기 위해 정보기기로부터 전송장치에 입력되는 정보의 2진 부호는 원거리 전송에 적합한 신호로 부호화된다. 이것을 회선부호화(line coding)라고 한다

* 한세대학교 IT학부 정보통신공학과(wphong@hansei.ac.kr)

· 제1저자 (First Author) : 홍완표

· 투고일자 : 2011년 9월 9일

· 심사(수정)일자 : 2011년 9월 12일 (수정일자 : 2011년 10월 25일)

· 게재일자 : 2011년 10월 30일

[8][9].

전송장치는 입력되는 데이터비트중에 일정한 개수 이상의 0비트가 입력되면, 데이터비트의 회선부호화과정 중에서 이 연속된 0의 비트들을 사전에 정해진 비트패턴을 가진 신호로 바꾸어 전송한다.

이것은 일정한 개수 이상의 0의 비트열 성분을 가진 신호가 수신기의 복호기에 입력될 경우, 각 비트에 대한 동기를 맞추지 못하여 데이터를 복호화하지 못하는 일이 발생되지 않도록 하기 위한 것이다.

이러한 과정을 스크램블링이라고 한다[8]. 따라서 정보기기에서 문자나 기호 등을 원천부호화 할 때, 일정한 개수 이상의 “0”의 비트로 구성된 부호가 많을 경우, 전송장치에서 스크램블링이 많이 발생하게 된다. 즉 문자나 기호 등에 대한 원천부호화의 방법에 따라 전송장치의 효율에 영향을 주게 되는 것이다.

전송장치에서 회선부호화 방식으로 AMI방식을 사용하는 경우에, 일반적으로 사용되는 스크램블링 방식으로 B8ZS(bipolar 8-zero substitution)와 HDB-3(high-density bipolar 3)[10]방식이 있다. 전자는 미국의 표준방식[10][11]이고 후자는 ITU-T[12]와 우리나라에서 사용하는 표준방식[13]이다.

그림1은 회선부호화방식인 AMI방식[16]과 스크램블링방식인 B8ZS 및 HDB-3방식을 보여주는 것이다. 본 논문은 HDB-3 스크램블링 방식을 고려하여 문자나 기호 등의 원천부호화 규칙을 제시하였다. 현재 문자나 기호 등의 원천부호화 체계는 일반적으로 3비트열 x 4비트행의 1바이트[14] 또는 4비트열 x 4비트행의 2바이트[14]를 따르고 있다.

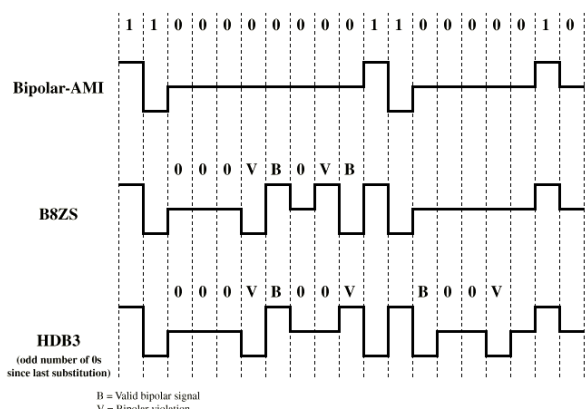


그림 1. AMI방식과 B8ZS 및 HDB-3방식
Fig.1. B8ZS, HDB-3 Scrambling and AMI line coding

ASCII, EBDIC, Unicode, UTF 등이 대표적인 원천부호화 체계이다. 본 논문은 KS X 1001 정보교환용 부호계 중에서 4비트열 x 4비트행의 2바이트의 부호화체계에 대한 부호화규칙을 제시하였다. 본 논문에서는 2 바이트 부호체계에 대한 모델로 KS X1001[14]에 규격화되어 있는 로마문자용 부호집합을 참고로 하였다.

본 논문에서는 원천부호화에 사용되는 4x4비트 2 바이트중에서 연속된 “0”의 비트가 4개 이상 발생하는 부호체계를 찾아내었다. 이것은 문자나 기호를 원천부호화하는 과정중에서 연속된 4개이상의 “0”이 발생하는 부호의 사용을 피하도록 하고자 하는 것이다. 그러므로 본 논문에서 제시한 규칙을 적용할 경우 회선부호화과정에서 발생하는 스크램블링을 최소화할 수 있고 결과적으로 전송장치의 운용효율과 데이터의 전송효율을 향상시킬 수 있도록 하였다.

II. 원천부호화 표준화

2-1 개요

한국기술표준원은 2004년도에 정보교환용 부호계(한글 및 한자)에 대한 표준규격 KS X 1001을 개정 공표하였다[14]. 이 규격은 정보 처리 및 데이터를 전송하는 시스템에서 정보 교환에 사용하는 부호계의 표현 형식에 대한 것을 규정하고 있다. 이 규격에서는 문자, 도형문자, 특수문자, 숫자, 한글낱자, 한글글자마디, 로마문자, 한자, 그리스문자 그리고 바이트 등에 대한 정의를 하고 있다. 한글 낱자는 한글 글자마디를 이루는 닿소리, 홀소리 글자 또는 첫소리, 가운데소리, 끝소리 글자(“?”이나 “과” 같은 겹글자도 하나의 낱자로 여긴다.)를 가리킨다. 한글 낱자는 닿소리 글자 30자, 홀소리 글자 21자, “채움”(끝소리 글자 없음 표시) 1문자와 고어 42문자, 총 94문자로 한다.

한글 글자 마디는 첫소리와 가운데소리 글자, 또는 첫소리, 가운데소리, 끝소리 글자로 이루어진 한글의 단위이다. 한글 글자 마디는 사용 빈도에 의하여 선정된 한글 글자 마디 2,350자로 하고 있다. 원천부호체계는 행과 열로 구성된 2바이트조합으로 구성된다. 바이트는 정보 교환의 편의상 한 단위로 취급

되는 8개의 비트를 의미한다. 행은 2바이트 부호 계에서 첫째 바이트에 의하여 구별되는 부호 값의 집합이다. 열은 지정된 행 안에서 둘째 바이트에 의하여 구별되는 각각의 부호 값을 의미한다[14].

2-2 문자 부호화 규칙

이 규격은 부호계의 단위를 2바이트로 하고 있다. 각각의 바이트는 정보 교환용 부호계의 확장법(ISO/IEC2022)에 따라 7비트 또는 8비트로 한다. 두 개의 바이트 중에서 앞의 바이트가 첫째 바이트, 뒤의 바이트가 둘째 바이트이다.

도형문자에 대한 부호계열은 7/8비트열 바이트 모두 b7-b1비트열로 구성된다. 따라서 8비트열 바이트의 경우에는 최상위 비트인 b8비트가 항상 “1”이 되게 한다. 이 규격에 포함되지 않는 한글 글자 마디는 첫소리, 가운뎃소리, 끝소리 글자로 분류하고 각 부분에 해당되는 특수 문자 영역의 한글 낱자 부호 값을 정보 교환에 사용한다. 한글 낱자만의 정보 교환과 구분하기 위하여 한글 글자 마디마다 맨 앞에 “채움” 문자를 1개씩 추가하도록 하고 있다.

2-3 문자 부호의 분류와 배열

도형문자의 분류와 배열은 표1과 같다[14]. 사용빈도가 높은 특수 문자, 숫자, 한글 낱자, 외국 문자 및 패선 조각은 제1행~제12행에 배열하고 있다.

표 1. 문자부호 배열표

Table 1. Assignment of Characters Code

문자 종류	행 번호
특수 문자	1~2
로마 문자용 부호 집합	3
한글 낱자	4
로마 숫자 그리스문자	5
패선 조각	6
각종 단위 기호	7
확장 로마문자 및 특수 문자	8~9
가나 문자	10~11
키릴 문자	12
정의 유보	13~15
한글 글자 마디	16~40
사용자 정의 영역-영역 A	41
한 자	42~93
사용자 정의 영역-영역 B	94

한글글자마디는 총2350자를 가나다순으로 제16행~제40행에 배열하고 있다.

2-4 2바이트 조합형 부호계 구성규격

이 규격 부속서3[14]에는 기본 부호계인 2바이트 완성형 부호계의 보조 부호계로서 2바이트 조합형 부호계를 규정하고 있다. 한글은 부속서 3 표 2에 규정된 첫소리 글자 19자, 가운뎃소리 글자 21자, 끝소리 글자 27자로 조합 가능한, 모든 현대 한글 글자 마디(11,172자) 및 현대 한글 낱자(67자)에 대한 것이다. 한글글자마디에 대한 코드배치영역은 첫 번째 바이트는 84-D3, 두 번째 바이트는 41-7E, 81-FE로 규정하고 있다. 한글글자마디에 대한 부호 값은 2바이트 내에 첫소리 글자 5비트, 가운뎃소리 글자 5비트, 끝소리 글자 5비트로 하여 한글 낱자를 조합하여 표현한 값으로 하고 있다. 각 한글 낱자의 순서는 최상위 비트(MSB)를 1로 하고 첫소리, 가운뎃소리, 끝소리 글자가 순서대로 나오도록 구성하고 있다. 표2는 이것에 대한 것이다.

표 2. 한글글자마디 구성 규칙

Table 2. Hanguk Syllable Code Rule

첫째 바이트								둘째 바이트													
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0						
“1”	초 성							중 성							종 성						

2-5 7비트 한글 낱자 부호계 규격

이 규격 부속서 4[14]에는 한글낱자 7비트 부호계에 대하여 규정하고 있다. 이 규격은 권장규격으로 원칙적으로 정보교환용으로는 사용하지 않는 것으로 규정하고 있다. 7비트 한글 낱자 부호값은 한글 낱자 1자를 표현한다. 즉, 한개의 글자마디는 첫소리 바이트, 가운뎃소리 바이트, 끝소리 바이트의 세개의 바이트로 조합되어 나타낸다. 이 표 3에서 4/0의 채움 문자에 대한 용법은 첫소리, 가운뎃소리, 끝소리 글자가 없을 때 각각 그 자리에 넣는 부호이다.

표 3. 7비트 한글낱자 부호계
Table 3. Hangeul Characters 7-bits code

				b7	0	0	0	0	1	1	1	1	
				b6	0	0	1	1	0	0	1	1	
				b5	0	1	0	1	0	1	0	1	
b4	b3	b2	b1	행	0	1	2	3	4	5	6	7	
0	0	0	0		0				(채움) ㄹ				
0	0	0	1	1				ㄱ	口				
0	0	1	0	2				ㄲ	ㅂ	ㅅ	ㅇ		
0	0	1	1	3				ㄳ	ㅃ	ㅆ	ㅈ		
0	1	0	0	4				ㄴ	ㅅ	ㅈ	ㅊ		
0	1	0	1	5	기 능 문 자			ㄷ	ㅅ	ㅆ	ㅈ	ㅊ	
0	1	1	0	6					ㄹ	ㅅ	ㅈ	ㅊ	
0	1	1	1	7					ㅁ	ㅇ	ㅈ	ㅊ	
1	0	0	0	8					ㄷ	ㅅ			
1	0	0	1	9					ㄹ	ㅅ	ㅈ	ㅊ	
1	0	1	0	10					ㄷ	ㅅ	ㅈ	ㅊ	ㅌ
1	0	1	1	11					ㄹ	ㅅ	ㅈ	ㅊ	ㅌ
1	1	0	0	12					ㄷ	ㅅ	ㅈ	ㅊ	ㅌ
1	1	0	1	13					ㄹ	ㅅ	ㅈ	ㅊ	ㅌ
1	1	1	0	14					ㄷ	ㅅ	ㅈ	ㅊ	ㅌ
1	1	1	1	15					ㄹ	ㅅ	ㅈ	ㅊ	ㅌ

이상에서와 같이 정보교환용 부호계는 국제 표준화 기구 (ISO)의 정보 교환용 부호계의 확장 방법 (ISO/IEC 2022)을 따르는 2바이트 부호 계로 규정하고 있다[14]. 부호값의 범위는 2121H-7E7EH까지 8836문자를 수용할 수 있도록 구성하고 있다. 2바이트 조합형 부호계에서 모아 쓴 한글의 부호값은 KS X 1003(KSC 5636)에 규정되어 있는 7비트 부호계와 8비트 부호계의 기능 문자와 중복되지 않도록 하고 있다. 첫소리와 가운뎃소리 글자 자리에도 “채움” 문자부호를 배열하여 2바이트 조합형 부호계에서 한글 낱자 부호 값도 한글 글자 마디와 같은 방법으로 만들 수 있도록 하고 있다. 또한 정보 교환용 한글 부호계에 포함되지 않은 한글 글자 마디는 그 출현 빈도가 매우 낮을 것으로 보고 보통 한글 낱자와 구분하기 위하여 각 한글 글자 마디마다 맨 앞에 “채움” 문자 1개씩 덧붙여 교환하도록 하고 있다. 도형문자의 한글 글자마디 부호 표는 총2350 자를 가나다순으로 배열하고 있다. 이와 같이 현재의 한글 부호계 규격은 국제표준 부호배열 규격에 맞추어 가나다 또는

ㄱ, ㄴ, ㄷ 등의 순서로 단순 배열하고 있음을 알 수 있다[14].

2-6 로마문자용 부호집합

정보 교환용 부호계 (한글 및 한자)(KS X 1001 : 2004) 표1[14]에는 다음 표 4와 같은 로마문자용 부호 집합이 규격화되어 있다.

표 4. 로마문자용 부호집합
Table 4. Code sets of Roman Characters

	A3A	A3B	A3C	A3D	A3E	A3F
0		0 A3B0	@ A3C0	P A3D0	~ A3E0	p A3F0
1	! A3A1	1 A3B1	A A3C1	Q A3D1	a A3E1	q A3F1
2	" A3A2	2 A3B2	B A3C2	R A3D2	b A3E2	r A3F2
3	# A3A3	3 A3B3	C A3C3	S A3D3	c A3E3	s A3F3
4	\$ A3A4	4 A3B4	D A3C4	T A3D4	d A3E4	t A3F4
5	% A3A5	5 A3B5	E A3C5	U A3D5	e A3E5	u A3F5
6	& A3A6	6 A3B6	F A3C6	V A3D6	f A3E6	v A3F6
7	' A3A7	7 A3B7	G A3C7	W A3D7	g A3E7	w A3F7
8	(A3A8	8 A3B8	H A3C8	X A3D8	h A3E8	x A3F8
9) A3A9	9 A3B9	I A3C9	Y A3D9	i A3E9	y A3F9
A	* A3AA	: A3BA	J A3CA	Z A3DA	j A3EA	z A3FA
B	+ A3AB	; A3BB	K A3CB	[A3DB	k A3EB	{ A3FB
C	, A3AC	< A3BC	L A3CC	W A3DC	l A3EC	 A3FC
D	— A3AD	= A3BD	M A3CD] A3DD	m A3ED	} A3FD
E	. A3AE	> A3BE	N A3CE	^ A3DE	n A3EE	— A3FE
F	/ A3AF	? A3BF	O A3CF	— A3DF	o A3EF	

이 표 4의 규칙에 의해 영어단어 “apple”를 원천부호로 나타내면 다음과 같다.

A3E1 A3F0 A3F0 A3EC A3E5

이 부호에서 앞의 바이트 A3가 행으로서 첫번째 바이트이고 뒤의 바이트 E1, FO 등이 두 번째 바이트이다. 첫 번째 바이트의 맨 왼쪽비트, b16이 최상위비

트가 되고 두 번째 바이트의 맨 오른쪽 비트, b1이 최하위비트가 된다. 그러므로 이 원천부호가 전송장치에 입력될 때는 다음과 같은 순서로 입력된다.

A3E5 A3EC A3F0 A3F0 A3E1 → 출력방향
 이것을 비트열로 나타내면 다음과 같다.
 10100011 11100101 10100011 11101100
 10100011 11110000 10100010 11110000 10100011
 11100001 → 출력방향

III. 회선부호화(Line coding)

3-1 회선부호화 목적

컴퓨터나 정보기기에서 문자를 입력하여 저장하고 표현하기 위해 2진 데이터로 변환한다. 즉 문자를 일정한 비트부호로 표현하는 것이다. 이것을 문자부호(Character code)라고 하며 ASCII부호와 EBCDIC부호가 이러한 것 들이다. 이러한 문자부호를 전송로를 통하여 원격지에 전송하기 위해서는 회선부호화(Line coding)이라는 과정을 거치게 된다. 즉 정보기기내의 2진 부호를 통신에 적합한 신호로 바꾸어 주는 것이다. 회선부호화를 하는 목적은 크게 두 가지 있다. 첫째는 정보기기내의 2진부호는 직류성분이다. 그러므로 2진부호를 그대로 전송로를 통하여 원거리에 전송하는 것은 적합하지 않다. 그러므로 이 직류성분의 2진 부호를 교류성분의 신호로 바꾸어 주는 것이다. 둘째는 정보기기내의 문자부호를 구성하는 비트열이 연속적인 “0”의 비트 열로 구성될 때 발생하는 문제를 방지하기 위한 것이다. 문자부호를 구성하는 “0”비트를 회선 부호화 할 때 + 또는 -준위를 갖도록 하지 않을 경우가 있다. 즉 비트“0”의 준위를 0으로 하는 것이다. 이 경우에 연속적인 “0”의 비트열이 수신될 경우 비트간의 동기를 맞출 수가 없게 된다. 즉 데이터를 디코딩(decoding)할 수 없게 된다[15].

그러므로 이러한 문제를 방지하기 위해 회선부호화과정을 거쳐 연속되는 “0”의 비트열을 비연속 “0”의 비트열로 인위적으로 바꾸어 주는 것이다. 결과적으로 정보기기내에서 만들어진 2진 문자부호의 비트

열 구성형태에 따라 시스템의 전송효율에 영향을 미치게 되는 것이다.

3-2 회선부호화 표준

가. 연속 “0” 비트열의 변경

2진 문자부호에서 연속된 “0”이 생성될 때 이 “0”의 연속 비트열을 비연속 “0”의 비트열로 변경시키는 방법에는 크게 두 가지 국제 표준이 있다. 북미 규격과 ITU-T규격이 이에 해당된다. 국내규격은 한국정보통신기술협회(TTA) 에서 제정한 규격이 있다. 본 논문에서는 TTA에서 제정한 국내 규격[13]과 ITU-T 규격[12]을 중심으로 연구결과를 도출하였다. 이 방법들에는 2진 문자부호를 교류적 전송 신호 성분으로 바꾸는 방법인 AMI방식을 함께 사용하고 있다.

1) 북미규격

일반적으로 북미에서는 B8ZS(Bipolar with 8 zero substitution) 규격을 사용한다[16]. 이 규격은 연속 “0”의 비트가 8개 이상 입력될 때 8개의 “0”비트를 정해진 비트 열로 바꾸어 주는 방법이다. 바꾸어 주는 비트열은 000VB0VB가 된다. 이 방식은 AMI방식으로 인한 단점을 보완하기 위한 것이다. 즉 연속된 “0”으로 인한 동기를 잃어버리는 문제를 해결하기 위한 기술이다. V(Violation)비트는 AMI규칙을 따르지 않는 강제로 삽입 되는 준위로서 바로 이전 “1”의 극성과 같은 극성을 갖게 한다. B(Bipolar)는 AMI규칙을 적용한 준위이다. V와 B에 대한 극성의 결정은 연속되는 “0”의 바로 이전의 “1”의 비트에 대한 극성에 따라 결정된다.

2) ITU-T 및 국내규격

원천데이터 부호화에 대한 국내규격은 HDB3(High Density Bipolar 3zero)방식을 적용하고 있다 [10]. 이 방식은 원천 데이터로부터 연속 “0”의 비트열이 4개 이상일 때 이 4개의 “0”비트를 정해진 비트열로 바꾸어 주는 방식이다. 이 방식도 AMI방식의 단점을 보완하기 위한 방법이다. 연속된 4개의 “0”비트 열은 000V 또는 B00V중 하나로 바뀐다. V와 B에 대한 정의와 적용방법은 B8ZS의 경우와 동일하다. 두 개 중 하나를 결정하는 기준은 이전 비트 열 중에

서 “0”이 연속인 비트가 정해진 비트 열로 대체된 후에 발생된 “1”의 개수에 의해 결정한다. “1”의 개수가 홀수인 경우에는 000V로 대체되고 “1”의 개수가 짝수일 때는 B00V로 대체된다. 즉, 송신장치의 부호기에서는 “0”의 연속도 검색하여야 하며 정해진 비트 열로 “0”의 연속 비트열이 대체된 후의 “1”의 개수도 몇 개인지 계수하여 그 계수정보를 실시간 업데이트 하여 저장하고 있어야 한다. 수신기의 역부호기에서는 연속되어 입력되는 비트열로부터 대체되어 있는 비트열을 찾아내어 원래 비트열로 복원시켜주어야 한다.

IV. 문자원천부호화 규칙

4-1 부호화연구 범위와 조건

본 논문에서는 정보기기내에서 문자를 2진 부호화 하는 규칙을 논하는데 위에서 언급한 HDB-3 회선 (line) 부호화 방식을 기준으로 하였다. 본 논문에서는 정보기기 내에서 문자를 원천 부호화하는 규칙으로 다음과 같은 범위와 조건을 적용하였다.

첫째, 1개의 조합단위를 구성하는 4비트 열 또는 단위조합 4,4비트열과 바이트 8,8비트열간 및 바이트의 조합 16,16비트열의 조합에 있어서 4개 이상의 연속된 “0”비트열이 발생하는 조합을 찾아낸다.

둘째, 1바이트는 4비트 행 x 4비트 열=8 비트로 한다.

셋째, 단위조합간의 결합은 4,4행x4,4행=16비트 또는 4,4,4행 x4열=16비트로 한다.

넷째, 문자부호의 단위바이트 또는 바이트의 조합에 있어서 “0”이 연속하여 4개 이상 발생하는 조합을 찾아 제시한다.

4-2 4 x 4비트 단위조합

표4는 이러한 원칙을 적용한 4비트 x 4비트 단위 조합에 대한 각 비트열 별 조합이 가능한 것과 조합이 제한되는 조합조건을 보여 주는 것이다.

이러한 조건으로 볼 때 16진수 0의 경우가 조합제한 개수가 8개로 가장 많고 16진수 7이 5개, 16진수

4와 C가 각각 4개, 16진수 2, 6, A, E가 각각 2개이다. 기타 1, 3, 5, 9, B, D, F는 16진수 0을 제외한 모든 비트열과 조합이 가능하다.

결과적으로 이러한 방식으로 조합할 경우 조합이 가능한 16x16=총256개에서 52개의 경우에는 조합이 되었을 때 회선부호화 과정에서 스크램블링이 발생되게 됨을 나타내고 있다. 즉, 52개의 부호는 사용빈도가 낮은 문자부호나 디스플레이 장치로 출력되지 않는 비표현 문자부호에 부여한다.

표 5. 원천 부호화 규칙 ; 4 x 4비트
Table 5. Source Coding Rule; 4 x 4-bits

HEXA	상위 비트열	하위 비트열	
		조합제한	조합가능
0	0000	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F	X
1	0001	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
2	0010	0,1	2,3,4,5,6,7,8,9,A,B,C,D, E,F
3	0011	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
4	0100	0,1,2,3	4,5,6,7,8,9,A,B,C,D,E,F
5	0101	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
6	0110	0,1	2,3,4,5,6,7,8,9,A,B,C,D, E,F
7	0111	0,C,D,E,F	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
8	1000	0,1,2,3,4,5,6,7	8,9,A,B,C,D,E,F
9	1001	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
A	1010	0,1	2,3,4,5,6,7,8,9,A,B,C,D, E,F
B	1011	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
C	1100	0,1,2,3	4,5,6,7,8,9,A,B,C,D,E,F
D	1101	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F
E	1110	0,1	2,3,4,5,6,7,8,9,A,B,C,D, E,F
F	1111	0	1,2,3,4,5,6,7,8,9,A,B,C, D,E,F

표 6은 단위조합 4비트열 16진수 조합으로 구성된 8비트-1바이트간의 조합규칙에 대한 것이다. 첫 번째 출력된 바이트의 최상위 비트열과 두 번째 출력된 바이트의 최하위 비트열과 조합되는 것이다. 예를 들어 첫 번째 바이트의 최상위 비트열이 16진수 1인 경우 최상위 비트열은 0001이 된다. 두 번째 출력된 바이트의 최하위 비트열이 16진수 2인 경우 비트열이 0010이 된다. 따라서 두 바이트의 조합은 0010 0001이 된다. 결과적으로 두 바이트간의 조합이 “0”이 연속4개로 이루어지게 된다. 그러므로 전체 비트열 조합에 의한 바이트가 구성되면 각 바이트간의 결합에 대한 것을 고려하여야 한다. 표 6의 분석결과에 의하면 처음 출력되는 바이트의 최상위 4개의 비트열이 16진수 1인 경우, 즉 “0001”비트열인 경우에 두 번째 출력되는 바이트의 최하위 4개의 비트열 16진수 중에서 0, 2, 4, 6, 8, 10, 12, 14, 15의 10개가 결합이 부적합한 것으로 나타났다. 처음 바이트의 최상위 비트열의 16진수가 2와 3인 경우, 두 번째 바이트의 최하위 비트 열과 결합이 부적합한 16진수는 0, 4, 10, 14의 4개로 나타났다. 처음 바이트의 최상위 비트열의 16진수가 4, 5, 6 및 7인 경우, 두 번째 바이트의 최하

위 비트 열과 결합이 부적합한 16진수는 0, 8의 2개로 나타났다. 처음 바이트의 최상위 비트열의 16진수가 8~F인 경우, 두 번째 바이트의 최하위 비트 열과 결합이 부적합한 16진수는 0, F의 2개로 나타났다.

V. 결 론

이상에서 살펴본 바와 같이 컴퓨터 등 정보 기기 내에서 문자를 어떠한 규칙에 의해 부호화하여 통신망에 전송하는가에 따라 전송장치의 운용효율에 영향을 주게 된다. 본 논문은 컴퓨터 등 정보기기로부터 생성되는 문자 부호를 전송로를 통하여 원거리에 전송할 때 전송효율을 높일 수 있는 문자 부호화 규칙을 제시하였다. 현재 ITU-T[12]와 국내[13]의 회선부호화 (Line coding) 방식은 HDB-3 스크램블 방식[3]을 채택하고 있다. 이 방식은 입력데이터의 회선부호화 과정에서 연속되는 “0”의 비트가 4개 이상 발생되지 않도록 하는 방식이다. 이 방식을 채택한 목적은 AMI회선부호화를 적용한 이후에 연속”0”비트로 인한 동기의 소실을 막기 위한 것이다. 따라서 사용빈

표 6. 2 바이트(16비트)간 조합규칙
Table 6. Coding rule between Bytes

16진수	첫번째 바이트 최상위 비트열	두 번째 바이트 최하위비트열	
		조합제한	조합가능
0	0000	모두	-
1	0001	0, 2, 4, 6, 8, 10, 12, 14	1, 3, 5, 7, 9, B, D, F
2	0010	0, 4, 10, 14	1, 2, 3, 5, 6, 7, 8, 9, B, C, D, F
3	0011	0, 4, 10, 14	1, 2, 3, 5, 6, 7, 8, 9, B, C, D, F
4	0100	0, 8	1, 2, 3, 4, 5, 6, 7, 9, A, B, C, D, E, F
5	0101	0, 8	1, 2, 3, 4, 5, 6, 7, 9, A, B, C, D, E, F
6	0110	0, 8	1, 2, 3, 4, 5, 6, 7, 9, A, B, C, D, E, F
7	0111	0, 8	1, 2, 3, 4, 5, 6, 7, 9, A, B, C, D, E, F
8	1000	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
9	1001	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
A	1010	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
B	1011	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
C	1100	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
D	1101	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
E	1110	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
F	1111	0	1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

도가 높거나 통신용으로 사용되지 않는 문자에 연속하여 4개 이상의 "0"의 비트가 존재하지 않는 부호를 부여하는 것이다. 본 논문에서는 문자를 부호화 할 때 사용빈도가 높은 문자의 경우에 "0"의 연속이 4개 이상 발생 되는 부호를 갖지 않도록 하는 문자부호화 규칙을 제시하였다.

본 연구와 관련된 3x4 비트열에 대한 문자부호화 규칙을 ASCII부호에 적용한 결과 전송효율이 향상되는 것으로 나타났다.[17]

향후 본 연구에서 제시한 4X4비트열에 대한 문자부호화 규칙을 4x4비트열을 토대로 하고 있는 Unicode, UTF, EBCDIC, KS X 1001의 각 문자부호집합 등에 적용하여 데이터 전송효율에 미치는 영향을 입증할 필요가 있다.

참 고 문 헌

- [1] <http://cnx.org/content/m0091/latest/>
- [2] RFC 20 ASCII format for Network Interchange ctober 1969(<http://tools.ietf.org/html/rfc20>)
- [3] American Standards Association, "American Code (July 6, 1999). for Information Interchange", ASA X3.4-1963, 17 June 1963
- [4] American National Standards Institute, "American National Standard for Information Systems-Coded Character Sets 7-Bit American National Standard Code for Information Interchange (7-Bit ASCII)", ANSI X3.4-1986, Inc., 26 March 1986
- [5] <http://en.wikipedia.org/wiki/EBCDIC>
- [6] <http://en.wikipedia.org/wiki/Unicode>
- [7] http://en.wikipedia.org/wiki/Universal_Character_Set
- [8] <http://www.infocellar.com/networks/Telecom101/T1-Line-Coding.htm>
- [9] http://en.wikipedia.org/wiki/Line_coding#Line_coding
- [10] <http://searchnetworking.techtarget.com/definition/B8ZS>

- [11] http://www.sqa.org.uk/e-learning/NetTechDC01ECD/page_13.htm
- [12] TU-T Recommendation G.703, "Physical/ electrical characteristics of hierarchical digital interfaces" pp24-41, Oct. 1998
- [13] TTA Standard, "Test Method for Telecommunication Terminal Equipment" TTAS. KO-05.0028/R1 pp306-451, Revised on 23 Dec. 2004
- [14] 산업자원부 기술표준원. "정보 교환용 부호계 (한글 및 한자) 부속서 4. 7비트 한글 낱자 부계" KS X 1001 : 2004, 2004. 12. 28.
- [15] Behrouz A. Forouzan, "Data communications" McGraw Hill Korea, pp132-134. Jan, 2008.
- [16] http://www.sqa.org.uk/e-learning/NetTechDC01ECD/page_12.htm
- [17] 홍완표, "데이터 전송효율을 고려한 3x4비트 1바이트 문자부호화 규칙에 관한 연구" 한국전자통신학회논문지, 제6권, 제4호, .2011.08.12

홍 완 표 (洪完杓)



1991년 서울과학기술대학교 전자공학과 졸업(공학사)
 1994년 연세대학교대학교 공학대학원 산업공학과 졸업(공학석사)
 1999년 광운대학교 대학원 전자공학과 졸업(공학박사)
 1990년 전기통신기술사합격
 1991년 정보통신부 5급특별임용고시합격 본부 통신정책실, 전파방송관리국, 정보화기획실
 1997년 삼성전자(주) 통신사업부 전송영업그룹장
 1999년 광운대학교 연구전담교수
 2000년 한국정보통신기술협회장
 2002년 한세대학교 IT학부 정보통신공학전공 교수
 한세대학교 정보통신연구소장
 관심분야 : 위성통신방송, 문자코딩, 통신정책