IMPLEMENTATION OF ENCODERS / DECODERS FOR CYCLIC CODES

CIRCUITS FOR DIVIDING POLYNOMIALS

THE DIVISION OF POLYNOMIALS PLAYS A KEY ROLE IN THE ENCODING AND DECODING CYCLIC CODES. FOR EXAMPLE DIVISION IS REQUIRED FOR:

* CYCLIC SHIFT OF THE CODEWORD POLYNOMIAL
* COMPUTATION OF PARITY POLYNOMIAL $R(x)$ $\left[= \text{rem.}\ \frac{x^r M(x)}{g(x)}\right]$

THE DIVISION CAN BE PERFORMED USING FEED-BACK SHIFT REGISTERS

CONSIDER TWO POLYNOMIALS $V(x)$ AND $g(x)$ GIVEN BY;

$$V(x) = V_m x^m \oplus V_{m-1} x^{m-1} \oplus \dots V_2 x^2 \oplus V_1 x \oplus V_0.$$

$$g(x) = g_r x^r \oplus g_{r-1} x^{r-1} \oplus \dots g_2 x^2 \oplus g_1 x \oplus g_0.$$

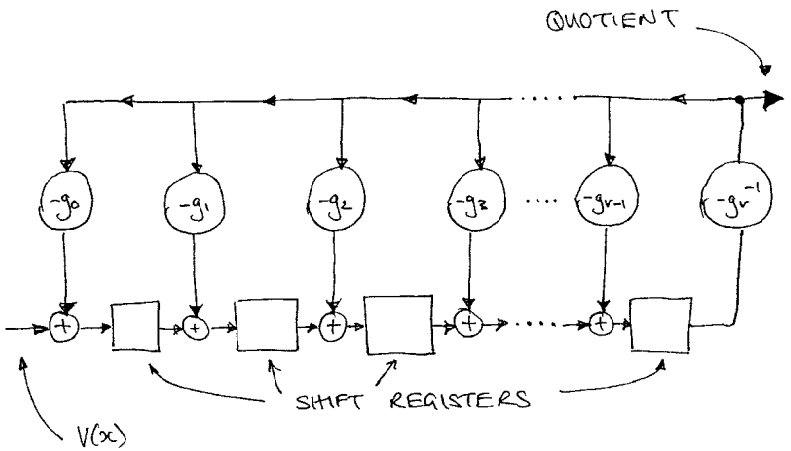
WHERE $m \geq r$.

DIVIDING $V(x)$ BY $g(x)$ YIELDS A QUOTIENT, $Q(x)$ AND A REMAINDER $R(x)$

HENCE:

$$\frac{V(x)}{g(x)} = Q(x) \oplus \frac{R(x)}{g(x)}$$

WE CAN PERFORM THIS DIVISION BY THE FOLLOWING CIRCUIT



$\dots V_{m-2}, V_{m-1}, V_m$ (HIGH-ORDER CO-EFFICIENT FIRST)

$g_i = 0$ OR 1. IF $g_i = 1$ CLOSED PATH

IF $g_i = 0$ OPEN PATH

OPERATION OF DIVIDER CIRCUIT

* SHIFT REGISTERS ARE INITIALIZED BY BEING FILLED WITH ZEROS.

* THE FIRST 'r' SHIFTS ENTER THE MOST SIGNIFICANT (HIGHEST-ORDER) COEFFICIENTS OF $V(x)$.

* AFTER THE r^{th} SHIFT, THE QUOTIENT OUTPUT IS $g_r^{-1} V_m$ - THE HIGHEST ORDER TERM IN THE QUOTIENT

* FOR EACH OF THE QUOTIENT COEFFICIENTS q_i THE POLYNOMIAL $q_i\, g(x)$ IS SUBTRACTED BY THE FEEDBACK ARRANGEMENT FROM THE DIVIDEND.

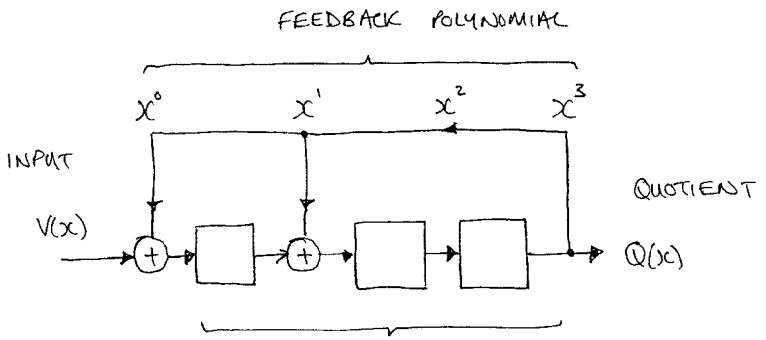
* THE DIFFERENCE AT EACH SHIFT APPEARS ON THE SHIFT REGISTERS.

* AFTER M+1 SHIFTS INTO THE REGISTER, $V(x)$ HAS BEEN SHIFTED IN, $Q(x)$ SERIALLY SHIFTED OUT, AND THE REMAINDER $R(x)$ RESIDES ON THE SHIFT-REGISTERS.

FOR EXAMPLE

SUPPOSE WE NEED TO DIVIDE $V(x)$ BY $g(x) = x^3 \oplus x \oplus 1$.

REQUIRED CIRCUIT LOOKS LIKE THIS;

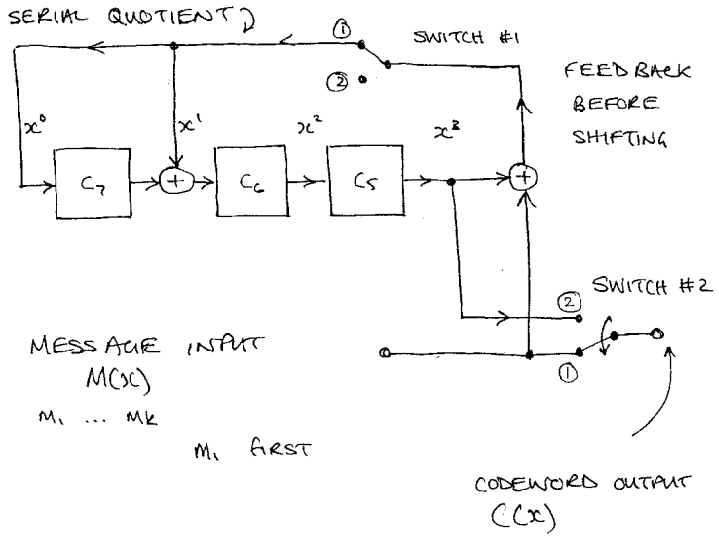


AGAIN NOTE THAT MOD-2 ADDITION IS THE SAME AS MOD-2 SUBTRACTION.

ENCODING USING (n-k) -STAGE SHIFT REGISTERS

CONSIDER OUR OLD FRIEND THE (7,4) CODE, WITH THE GENERATOR POLYNOMIAL GIVEN BY

$$g(x) = x^3 \oplus x \oplus 1$$



OPERATION

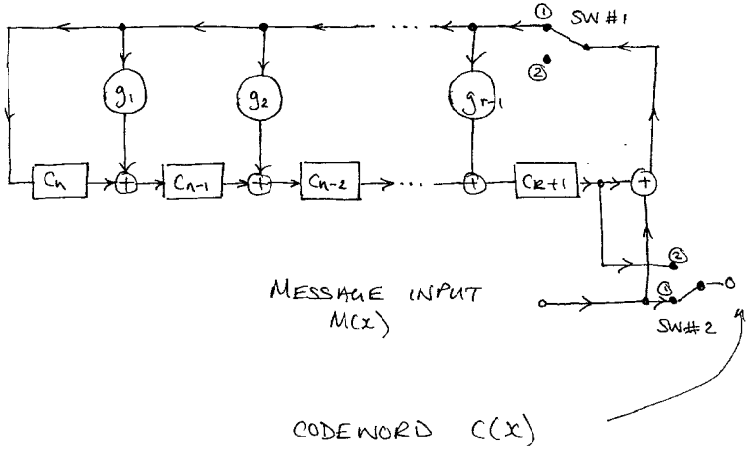
1) SWITCH #1, #2 POSITION ① MESSAGE BITS SHIFTED INTO CHANNEL AND INTO REGISTER STAGES

2) AFTER 'k' MESSAGE BITS, HAVE BEEN SHIFTED IN, THE REGISTERS CONTAIN THE PARITY BITS

3) SWITCHES #1 AND #2 TO POSITION ②. REGISTER CONTENTS SHIFTED INTO THE CHANNEL

IN GENERAL ...

(n,k) ENCODER $r = n-k$



DECODING : SYNDROME CALCULATION.

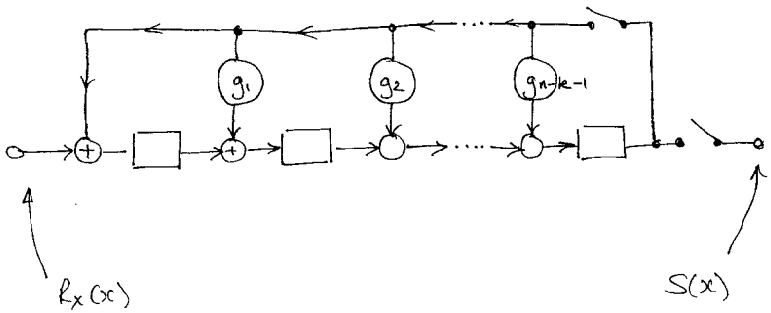
RECALL THAT THE SYNDROME $S(x)$ IS GIVEN BY ;

$$S(x) = \frac{R_x(x)}{g(x)}$$

WHERE $R_x(x)$ - RECEIVED POLY.
$g(x)$ - GENERATOR POLY.

THIS CAN BE REALIZED USING A DIVIDER CIRCUIT ;

TOPOLOGY DETERMINED BY $g(x)$.



RECEIVED CODEWORD

SYNDROME

WELL KNOWN BLOCK CODES

HAMMING CODES

HAMMING CODES ARE (n, k) CODES HAVING THE FOLLOWING PROPERTY

$$(n, k) = (2^m - 1,\ 2^m - 1 - m)$$

WHERE $m = 2, 3, \ldots$

e.g $(3, 1)$, $(7, 4)$, $(15, 11)$ etc.

HAMMING CODES HAVE $d_{min} = 3$. HENCE SINGLE ERROR CORRECTION OR UPTO DOUBLE ERROR DETECTION.

HAMMING CODES ARE PERFECT CODES

GOLAY AND EXTENDED GOLAY CODE

GOLAY CODE IS A LINEAR CYCLIC $(23, 12)$ CODE. $d_{min} = 7$. A PERFECT CODE CAPABLE OF CORRECTING ANY COMBINATION OF 3 OR FEWER ERRORS

$$g(x) = x^{11} \oplus x^9 \oplus x^7 \oplus x^6 \oplus x^5 \oplus x \oplus 1$$

REMEMBER :

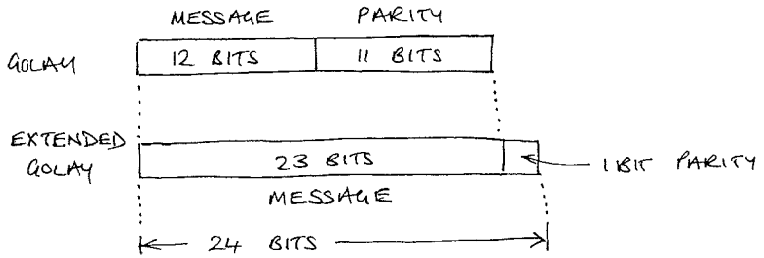
FOR A PERFECT (n,k) CODE CORRECTING UPTO 't' ERRORS WE REQUIRE

$$n - k = r.$$

$$2^r = \sum_{i=0}^{t} \binom{n}{i} = \sum_{i=1}^{t} \binom{n}{i} + 1$$

THE EXTENDED GOLAY CODE

FORMED BY ADDING AN OVERALL PARITY BIT TO THE (23, 12) GOLAY CODE TO MAKE IT A (24, 12) CODE



PRODUCES A ½ RATE CODE WHICH IS EASIER TO IMPLEMENT. CONSIDERABLY MORE POWERFUL THAN SIMPLE HAMMING CODES.

BCH - (BOSE-CHAUDHURI-HOCQUENGHEM)

ONE OF THE MOST POWERFUL CLASSES OF CYCLIC BLOCK CODES

(HAMMING CODES ARE A SUBSET OF BCH CODES)

BCH CODES CAN PROVIDE ERROR CORRECTION FOR A WIDE VARIETY OF BLOCK LENGTHS, CODE RATES, AND ALPHABET SIZES.

BCH CODES GUARANTEE THE CORRECTION OF 't' RANDOM ERRORS. AS 't' INCREASES, THE REDUNDANCY REQUIRED INCREASES DRAMATICALLY (SEE LAST PAGE)

THE ALGEBRAIC STRUCTURE OF BCH CODES LEADS TO QUITE STRAIGHT FORWARD DECODERS, WE WILL NOT CONSIDER IT HERE. THOUGH.

REED-SOLOMON CODES

A SPECIAL CLASS OF BCH CODES (WHICH ACTUALLY PRE-DATE BCH) THAT OPERATE ON SYMBOLS OF MORE THAN 1 BIT.

FOR EXAMPLE: BINARY = 1 BIT PER SYMBOL
ASCII CHARS = 8 BITS PER SYMBOL

R-S CODES ARE PARTICULARLY VALUABLE WHEN AS USUAL DATA IS REPRESENTED IN CHARACTER OR BYTE FORM. IN THIS CASE THE POLYNOMIALS HAVE MULTI-LEVEL COEFFICIENTS.

R-S CODES ARE WIDELY USED EVEN THE DATA-RATES ARE OF THE ORDER OF $Mb\ sec^{-1}$.

EXAMPLES OF R-S CODES IN USE;

(63, 47) CODE USED IN THE U.S. CELLULAR DIGITAL PACKET DATA (CDPD) STANDARD.
$m = 6$ BITS PER SYMBOL.

COMPACT DISC SYSTEM USES TWO INTERLEAVED R-S CODES; (32, 28) AND (28, 24)
- IN THEORY THE ERROR CORRECTION CAPABILITY OF THIS CODING SCHEME PERMITS 2.5MM HOLES TO BE MADE IN A CD WITHOUT ANY ILL-EFFECT!

MAXIMUM-LENGTH SHIFT-REGISTER CODES

A CLASS OF CODES CHARACTERIZED BY

$$(n, k) = (2^m - 1, m)$$

WHERE $m \in \mathbb{Z}^+$ (A POSITIVE INTEGER)

MAXIMUM LENGTH CODES ARE OFTEN USED TO GENERATE PERIODIC BINARY SEQUENCES WITH PERIOD $= 2^m - 1$

THESE SEQUENCES ARE CALLED PSEUDO-NOISE (PN) SEQUENCES AND ARE USED IN THE GENERATION OF SPREAD-SPECTRUM SIGNALS

⇒ WE WILL REVISIT THEM LATER IN THIS COURSE WHEN WE LOOK AT DIRECT SEQUENCE SPREAD SPECTRUM SYSTEMS

CYCLIC REDUNDANCY CHECK CODES

THESE CODES ARE USED FOR ERROR DETECTION IN LONG FRAMES OF DATA.

TYPICAL GENERATOR FUNCTIONS USED IN PRACTICE ARE;

CRC 16: $g(x) = x^{16} \oplus x^{15} \oplus x^2 \oplus 1$

CRC16-CCITT: $g(x) = x^{16} \oplus x^{12} \oplus x^5 \oplus 1$

THE FORM OF ERROR POLYNOMIAL FOR AN ERROR BURST OF LENGTH 'b' BITS IS;

$$E(x) = x^i E_2(x)$$

WHERE

$$E_2(x) = x^{b-1} \oplus x^{b-2} \oplus \dots x^0$$

WHERE x^i INDICATES THE POSITION OF THE ERROR BURST WITHIN THE DATA BLOCK.

ERROR BURSTS

AN ERROR BURST IS NOT NECESSARILY A STREAM OF CONTINUOUS ERRORS.

THE DEFINITION OF AN ERROR BURST IS THAT THE FIRST AND LAST BITS OF ANY INTERMEDIATE BITS ARE IN ERROR

FOR EXAMPLE; SUPPOSE $E(x)$ =

0000011010111000000

$\Rightarrow E_2(x) = x^7 \oplus x^6 \oplus x^4 \oplus x^2 \oplus x \oplus 1 \quad i = 5.$

* CRC CODES WILL <u>DETECT</u> ALL BURST ERRORS OF LENGTH $b \leq r$.

* BURSTS OF LENGTH $b > r$ WILL GO UNDETECTED.

* THE SUNDROME FOR THE <u>ERROR BURST</u> IS THE REMAINDER OF;

$$S(x) = \frac{E_2(x)}{g(x)}$$

$E_2(x)$ - OF DEGREE $b-1$

$g(x)$ - OF DEGREE $n-k=r$

HENCE, IF $b \leq r$ A REMAINDER WILL ALWAYS BE GENERATED AND THESE BURSTS WILL ALWAYS BE DETECTED.

* WITH $b > r$ SOME BURST PROBLEMS REMAIN UNDETECTED e.g. THOSE THAT GIVE ZERO ERROR SYNDROME $S(x)$.

$S(x)$ IS OF DEGREE $(r-1)$ AND HAS 'r' BINARY COEFFICIENTS, ONLY ONE OF WHICH IS ZERO THE PROBABILITY OF ERROR BURSTS WHICH GIVE ZERO REMAINDER IS THEREFORE;

$$\frac{1}{2^r} = 2^{-r}.$$

BLOCK INTERLEAVING

INTERLEAVING IS OFTEN USED AS A METHOD TO COMBAT BURST ERRORS.

INTERLEAVING THE CODED MESSAGE BEFORE TRANSMISSION AND DE-INTERLEAVING AFTER RECEPTION CAUSES BURSTS OF CHANNEL ERRORS TO BE SPREAD OUT IN TIME, AND THUS HANDLED BY THE DECODER AS IF THEY WERE RANDOM ERRORS

THE INTERLEAVER SHUFFLES THE CODE SYMBOLS OVER SEVERAL BLOCK LENGTHS. THE SPAN REQUIRED IS DETERMINED BY THE BURST STATISTICS OF THE CHANNEL e.g BURST PERIOD.

A BLOCK INTERLEAVER TAKES THE CODEWORDS, PERMUTES THEM AND FEEDS THE PERMUTED CODEWORDS TO THE MODULATOR.

INTERLEAVING IS USUALLY DONE BY AN MxN MATRIX. THE CODEWORDS ARE PLACED INTO THE COLUMNS AND THE DATA READ OUT FROM THE ROWS.

FOR EXAMPLE;

CONSIDER THE FOLLOWING

INPUT SEQUENCE: $C_1, C_2, C_3, C_4 \ldots C_{24}$

$N = 6$ COLUMNS

$M = 4$ ROWS

$$\begin{bmatrix} C_1 & C_5 & C_9 & C_{13} & C_{17} & C_{21} \\ C_2 & C_6 & C_{10} & C_{14} & C_{18} & C_{22} \\ C_3 & C_7 & C_{11} & C_{15} & C_{19} & C_{23} \\ C_4 & C_8 & C_{12} & C_{16} & C_{20} & C_{24} \end{bmatrix}$$

OUTPUT SEQUENCE: $C_1, C_5, C_9, C_{13}, C_{17}, \ldots$ etc.

IF A BURST OF 5 ERRORS OCCUR IN SYMBOLS $C_{14}, C_{18}, C_{22}, C_3, C_7$, AFTER DE-INTERLEAVING AT THE RECEIVER THE ERRORS ARE SPREAD OUT;

$$\begin{bmatrix} C_1 & C_5 & C_9 & C_{13} & C_{17} & C_{21} \\ C_2 & C_6 & C_{10} & (C_{14}) & (C_{18}) & (C_{22}) \\ (C_3) & (C_7) & C_{11} & C_{15} & C_{19} & C_{23} \\ C_4 & C_8 & C_{12} & C_{16} & C_{20} & C_{24} \end{bmatrix}$$

SYMBOL IN ERROR

DE-INTERLEAVED OUTPUT SEQUENCE;

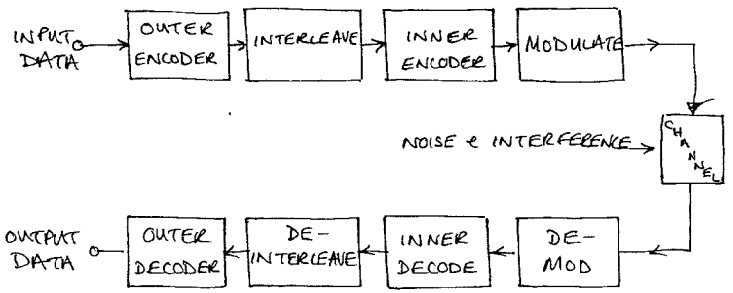
$C_1, C_2, (C_3), C_4, C_5, C_6, (C_7), C_8 \ldots$ etc

CONCATENATED CODES

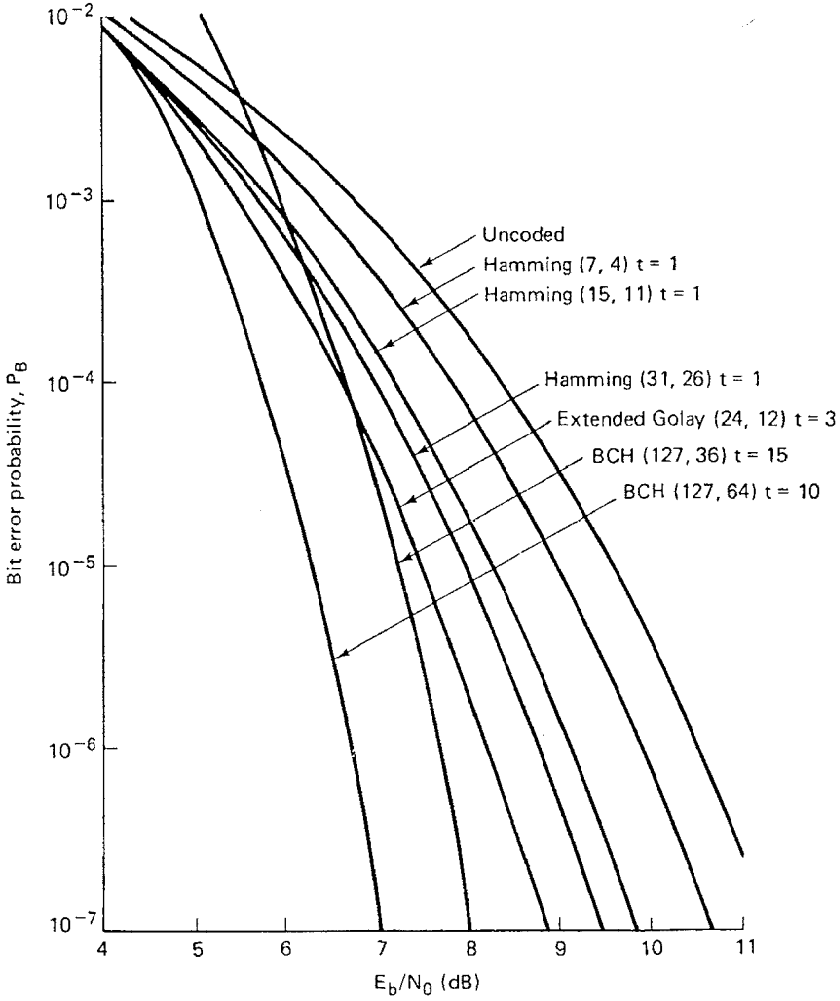
A CONCATENATED CODE USES TWO CODES (AN INNER CODE AND AN OUTER CODE) TO ACHIEVE THE DESIRED ERROR PERFORMANCE.

THE PRIMARY REASON FOR USING A CONCATENATED CODE IS TO ACHIEVE A LOW ERROR RATE WITH AN OVERALL IMPLEMENTATION COMPLEXITY THAN WOULD BE REQUIRED BY A SINGLE (LONGER) CODE.

CONCATENATED CODES ARE USUALLY USED IN CONJUNCTION WITH INTERLEAVING. OPERATING WITH R-S CODES SUCH SYSTEMS CAN EASILY ACHIEVE $E_b/N_0 = 2dB$ FOR $P_b = 10^{-5}$.



CD AUDIO USES TWO INTERLEAVERS.



Nº CORRECTABLE ERRORS

TABLE 5.2 Generators of Primitive BCH Codes

n	k	t	g(x)	n	k	t	g(x)
7	4	1	13	255	171	11	154162142123423560770616306 37
15	11	1	23		163	12	75004155100756025515747245146 01
	7	2	721		155	13	37575130054076650157225064646776 33
	5	3	2467		147	14	16421301735371655253041653054410 11
31	26	1	45		139	15	46140173206017556157072273024745 35
	21	2	3551		131	18	21571333147151015126125027744214 20
	16	3	107657				65471
	11	5	5423325		123	19	12061405224206600371721032651614 12
	6	7	313365047				72506267
63	57	1	103		115	21	60526665572100247263636404600276 35
	51	2	12471				6313472737
	45	3	1701317		107	22	22205772322066256312417300235347 42
	39	4	166623567				6574750154441
	36	5	1033500423		99	23	10656667253473174222741416201574 33
	30	6	157464165547				2411076432303431
	24	7	17323260404441		91	25	67502650303274441727236317247325 11
	18	10	1363026512351725				550762720724344561
	16	11	6331141367235453		87	26	11013676341474323643523163430717 20
	10	13	472622305527250155				067225452733117213 17
	7	15	5231045543503271737		79	27	66700035637657500020270344207366 17
							10153267117665413423 55

↑ OCTAL GENERATOR

POLYNOMIAL

127	120	1	211	71	29	24024710520644321515554172112331 16
	113	2	41567			5444250362557643221706035
	106	3	11554743	63	30	10754475055163544325315217357707 00
	99	4	3447023271			61117264552676136567025433 01
	92	5	624730022327	55	31	73154252035011001330152753060320 54
	85	6	130704476322273			41432675501055704442603547361 7
	78	7	26230002166130115	47	42	25335420170626465630330413774062 33
	71	9	6255010713253127753			12333414544604500506602455254317
	64	10	120653402557077310 0045	45	43	15202056055234161131101346376423 70
	57	11	335265252505705053517721			36700244707623730332021570250 515
	50	13	544465125233140124215 01421	37	45	51363302550670074141774472454375 30
	43	14	17721772213651227521220574343			73570617432343234764435473740 30
	36	15	31460746665220750447645747217 35	29	47	30257155366730714655270640123613 77
	29	21	40311446136767060366753014117 6155			34224232420117411406025475741 04
	22	23	12337607040472252243544562663764 7043			5037
	15	27	22057042445604554770523013762217 604353	21	55	12562152570603326560017731536076 11
	8	31	70472640527510306514762242715677331 30217			22734140565307454252115312161 44
255	247	1	435			3473725
	239	2	267543	13	59	46417320050525645444265737142500 6
	231	3	156720665			33067744547656140317467721357 02
	223	4	75626641375			460500547
	215	5	23157564726421	9	63	15726025217472463201031043255355 11
	207	6	16176560567636227			41623672120440745451127661155 47
	199	7	7633031270420722341			61677516057
	191	8	2663470176115333714567			
	187	9	52755313540001322236351			
	179	10	22624710717340432416300455			

Source: Reprinted with permission from "Table of Generators for BCH Codes," *IEEE Trans. Inf. Theory*, vol. IT10, no. 4, Oct. 1964, p. 1964 IEEE.

TAKEN FROM SKLAR pp 302-305