Section 1-5 Binary Codes

Thursday, January 21, 2021 12:52 PM

Suppose you want to inventory the contents of your fridge, and there are four things in there you want to track. You have an old computer that has 2-bit registers (memory) so you can store

00 → you decide this represents milk
01 → eggs 00
10 → rotting leftovers
→ ketchup

It's your code. If you write "00" to memory, you know it "means" milk. In this situation, "00" is not a number, it is "code" for milk.

If you buy more groceries, you need more memory.

000 → milk
001 eggs
010 rotting leftovers
011 ketchup
100 whipping cream
101 precious jewels
110 apples
110 cheese

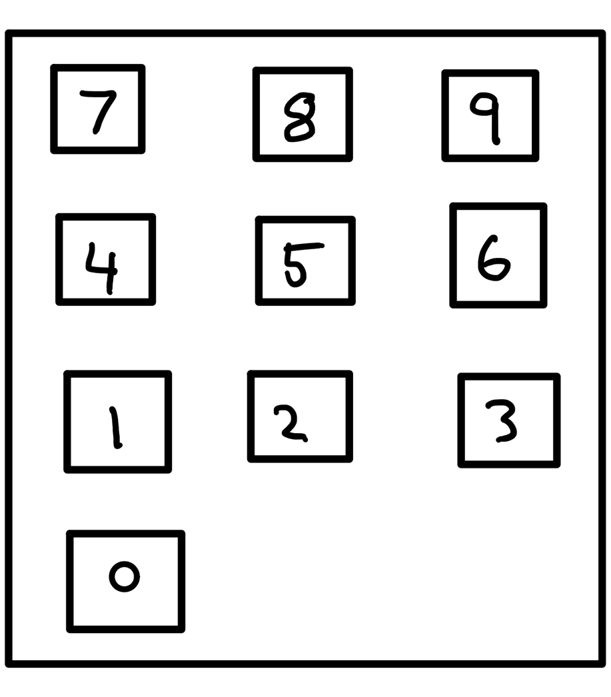
not numbers

We will examine some common codes

Computers: do everything in binary
Humans: used to base 10 and also letters

Consider the number pad on your device

We have 10 buttons to represent - we'll send a "code" for any button that's pressed
How many bits will we need?



Let's choose a code for when we press "3"
how about 0011, why not?

Binary Coded Decimal (BCD)

Pick a code for each number key

key	code
0:	0000
1:	0001
2:	0010
3:	0011
4:	0100
5:	0101
6:	0110
7:	0111
8:	1000
9:	1001
	1010
	1011
	1100
	1101
	1110
	1111

(1010 – 1111: not used in this code; no meaning)

These are digital CODES for each key, not binary numbers.
If you press 2 5 7

Keypad sends: 0010 0101 0111

They are binary-coded digits, not numbers - they mean "someone just pressed the "2" key," etc

Example if you read that sequence as bits (a binary number) you'd have:

$(001001010111)_2 = 599_{10}$
it is NOT 257

Your circuitry has to "know" what format it should be expecting - in other words, you design it to decode and act on BCD formatted data

Say you were transmitting BCD but wanted to add the numbers you recieve, say "16" and "3"

16 = 0001 0110   3 = 0011

  0001 0110
+   0011
  ???

Computers do math in base 2!

You would have to convert each digit to base 10 - forget it
Remember, in BCD these are not numbers anyway!

OK, so, maybe you have an entire keyboard.
Now you need a "code" for each key

26 upper case letters + 26 lower case letters + 9 digits + assorted punctuation

We have a standard code for this: 7-bit ASCII (American Standard Code for Information Interchange)
$2^7 = 128$ So can represent 128 unique symbols
punctuation
cartoon cuss-words (@#!&)
Some are non-printing like
form feed
space
bell
etc from days of paper


1:26 PM Thu Jan 21
Done ASCII-Table-wide.svg 100%

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	|
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

From Wikipedia

OK, suppose you want to include Chinese characters?

UNICODE → enough bits to represent every character in every language, including emojis

Other codes

Some codes are designed to have certain mathematical properties.

Example "2 out of 5" this code represents numbers but each code only has two "1's" in 5 bits:

0 00011
1 00101
2 00110
3 01001
etc

Usefor for error-checking

Example Gray Code

Repesents numbers, but only one bit changes from one number to the next:

0 0000
1 0001
2 0011
3 0010
4 0110
5 1110
6 1010
7 1011
8 1001
9 1000

Why? CMOS transistors

Counting from 0 to 7: binary (14 bits have to flip)
Gray Code (7 bit have to flip)