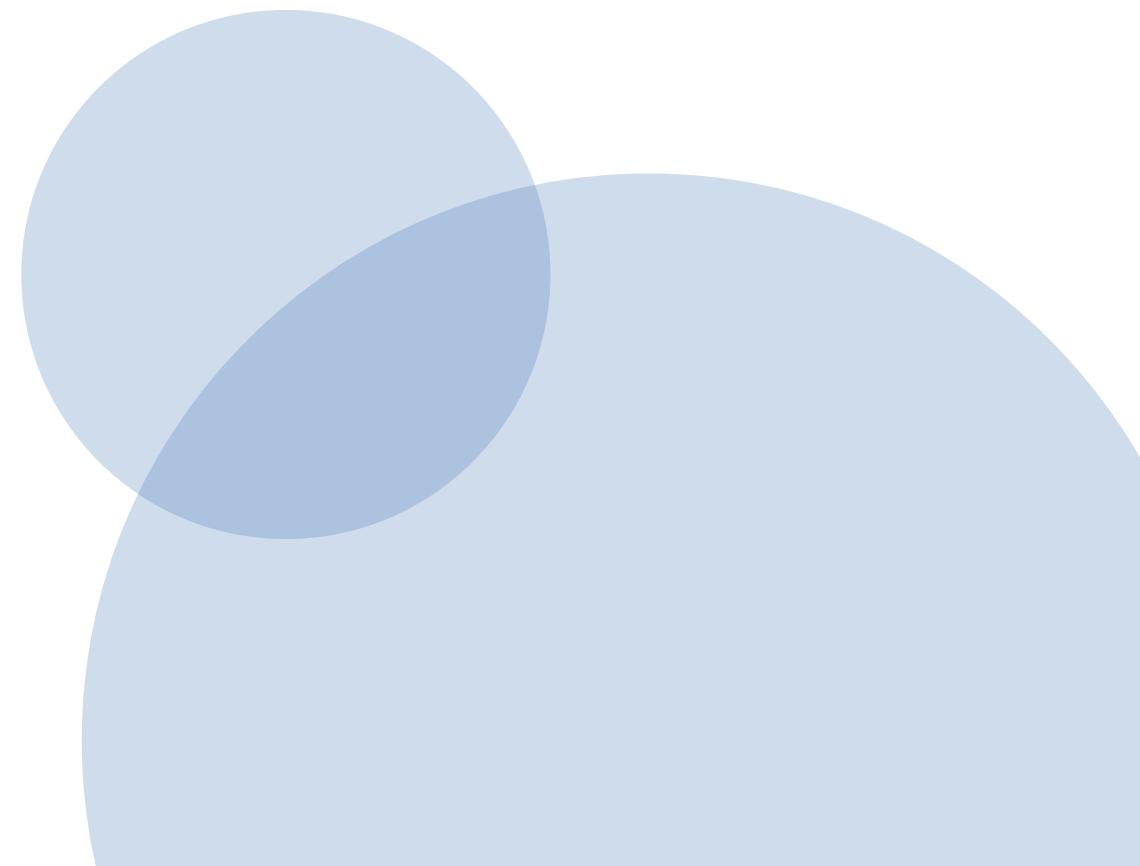


Bits & Bytes

Agenda:

- Hvad er forskellen på digitale og analoge signaler.
 - Hvordan bliver elektriske signaler til talværdier.
 - Det binære talsystem
 - Øvelse 1 – tæl binært
 - Det hexa-decimale talsystem. Hvad skal vi bruge det til.
 - Øvelse 2 – kodning og afkodning
 - Oktale talsystem
 - Opgaver i Codelabby
-
- Næste gang – data og filer



Analoge og digitale signaler

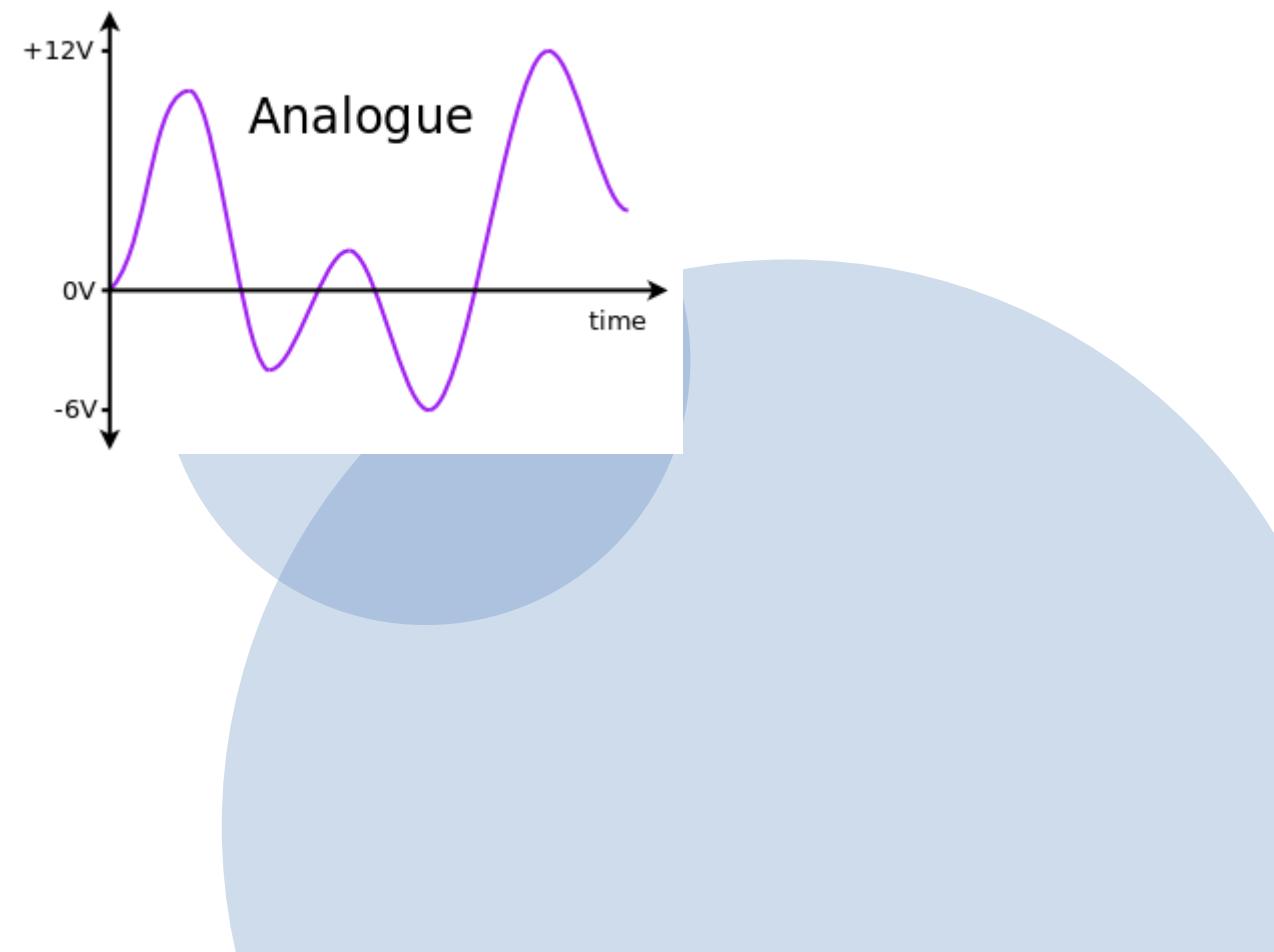
Analoge signaler:

Trinløse

Støjfølsomme

Er besværlige at skabe og læse elektronisk

Hvor er eksempler på analoge signaler?

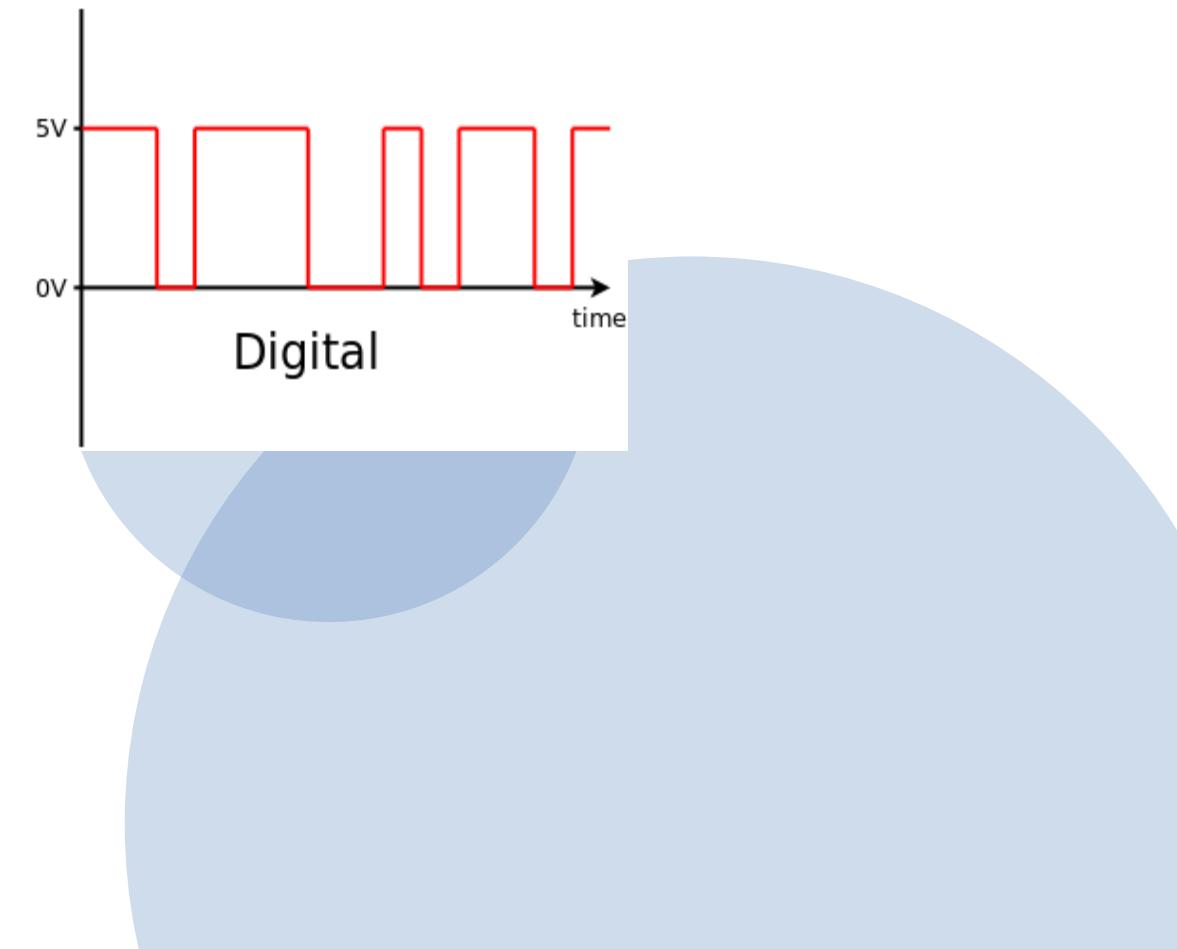


Analoge og digitale signaler

Digitale Signaler

Elektrisk robuste

Er billige at fremstille (transistor)

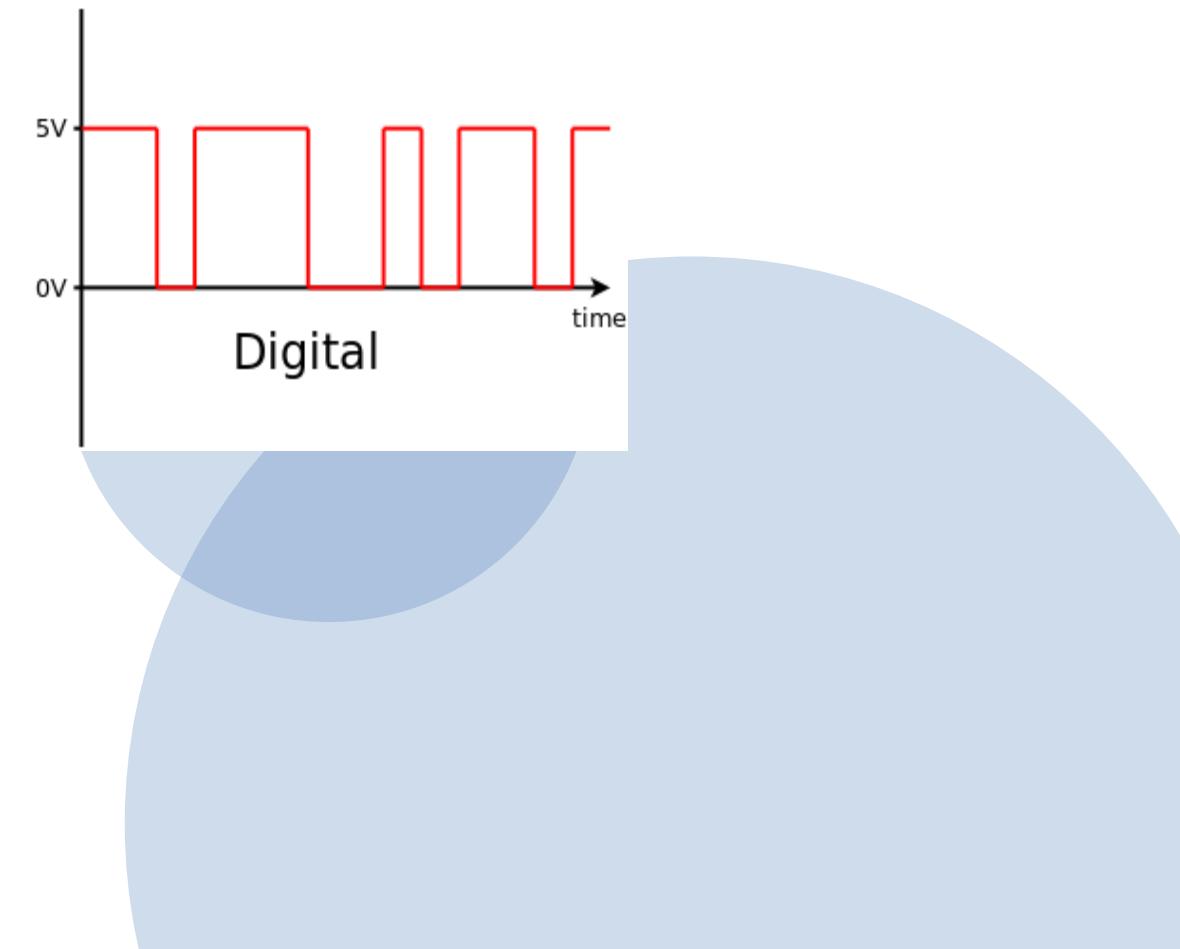


Analoge og digitale signaler

Digitale Signaler

Elektrisk robuste

Er billige at fremstille (transistor)



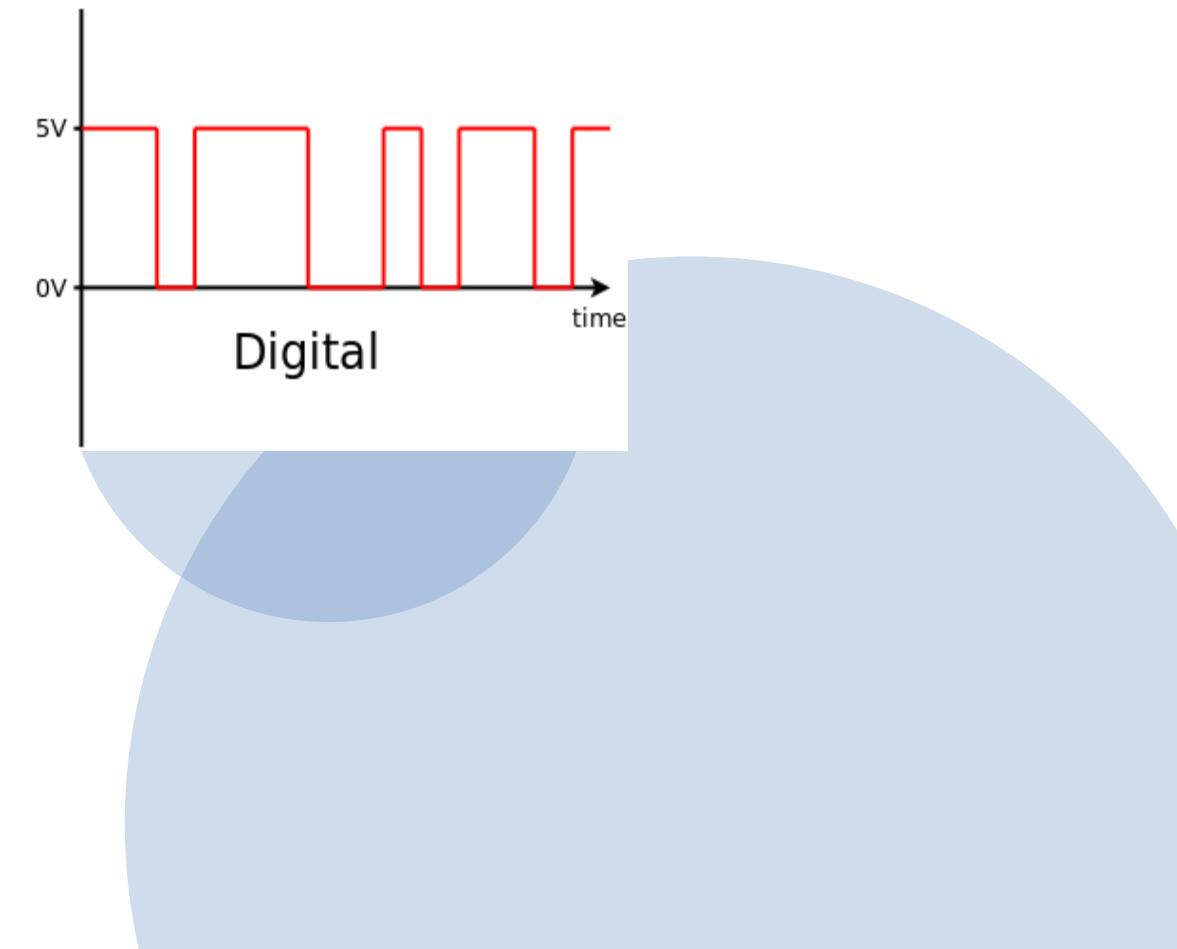
Analoge og digitale signaler

Digitale Signaler

Elektrisk robuste

Er billige at fremstille (transistor)

Hvor findes de digitale signaler.

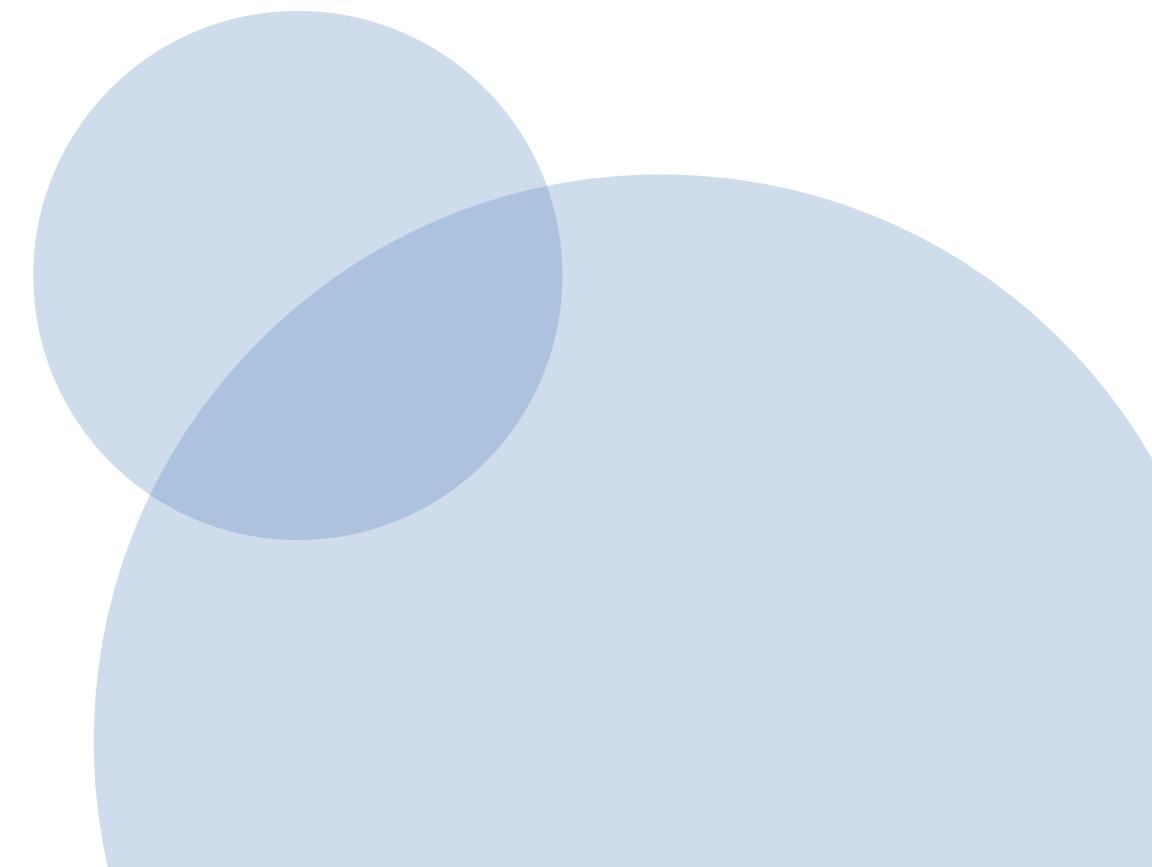


Analoge og digitale signaler

Bit – Binary Digit

2 tilstande:

Tændt/slukket
Sand /falsk
0/1
Høj/Lav (spænding)



Bits

Flere bits

1 bit = 2 forskellige kombinationer

2 bits = 4 forskellige kombinationer

3 bits = 8 forskellige kombinationer

4 bits = 16 forskellige kombinationer

antal kombinationer = $2^{\text{antal-bits}}$

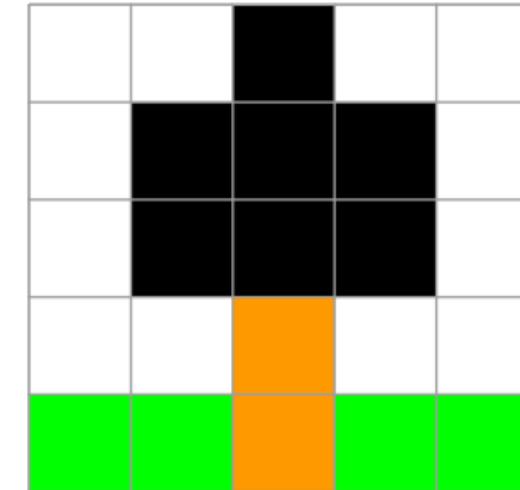
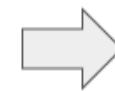
hvor mange bits skal der til for at repræsentere hele ASCII karaktersættet?

dec	hex	symbol									
000	00	(nul)	032	20	~	064	40	@	096	60	'
001	01	(soh)	033	21	!	065	41	A	097	61	a
002	02	(stx)	034	22	"	066	42	B	098	62	b
003	03	(etx)	035	23	#	067	43	C	099	63	c
004	04	(eot)	036	24	\$	068	44	D	100	64	d
005	05	(enq)	037	25	%	069	45	E	101	65	e
006	06	(ack)	038	26	&	070	46	F	102	66	f
007	07	(bel)	039	27	,	071	47	G	103	67	g
008	08	(bs)	040	28	(072	48	H	104	68	h
009	09	(tab)	041	29)	073	49	I	105	69	i
010	0A	(lf)	042	2A	*	074	4A	J	106	6A	j
011	0B	(vt)	043	2B	+	075	4B	K	107	6B	k
012	0C	(np)	044	2C	,	076	4C	L	108	6C	l
013	0D	(cr)	045	2D	-	077	4D	M	109	6D	m
014	0E	(so)	046	2E	.	078	4E	N	110	6E	n
015	0F	(si)	047	2F	/	079	4F	O	111	6F	o
016	10	(dle)	048	30	0	080	50	P	112	70	p
017	11	(dc1)	049	31	1	081	51	Q	113	71	q
018	12	(dc2)	050	32	2	082	52	R	114	72	r
019	13	(dc3)	051	33	3	083	53	S	115	73	s
020	14	(dc4)	052	34	4	084	54	T	116	74	t
021	15	(nak)	053	35	5	085	55	U	117	75	u
022	16	(syn)	054	36	6	086	56	V	118	76	v
023	17	(etb)	055	37	7	087	57	W	119	77	w
024	18	(can)	056	38	8	088	58	X	120	78	x
025	19	(em)	057	39	9	089	59	Y	121	79	y
026	1A	(eof)	058	3A	:	090	5A	Z	122	7A	z
027	1B	(esc)	059	3B	;	091	5B	[123	7B	{
028	1C	(fs)	060	3C	<	092	5C	\	124	7C	
029	1D	(gs)	061	3D	=	093	5D]	125	7D	}
030	1E	(rs)	062	3E	>	094	5E	^	126	7E	~
031	1F	(us)	063	3F	?	095	5F	_	127	7F	(del)

Bits

2 Bits

00	00	10	00	00
00	10	10	10	00
00	10	10	10	00
00	00	11	00	00
01	01	11	01	01

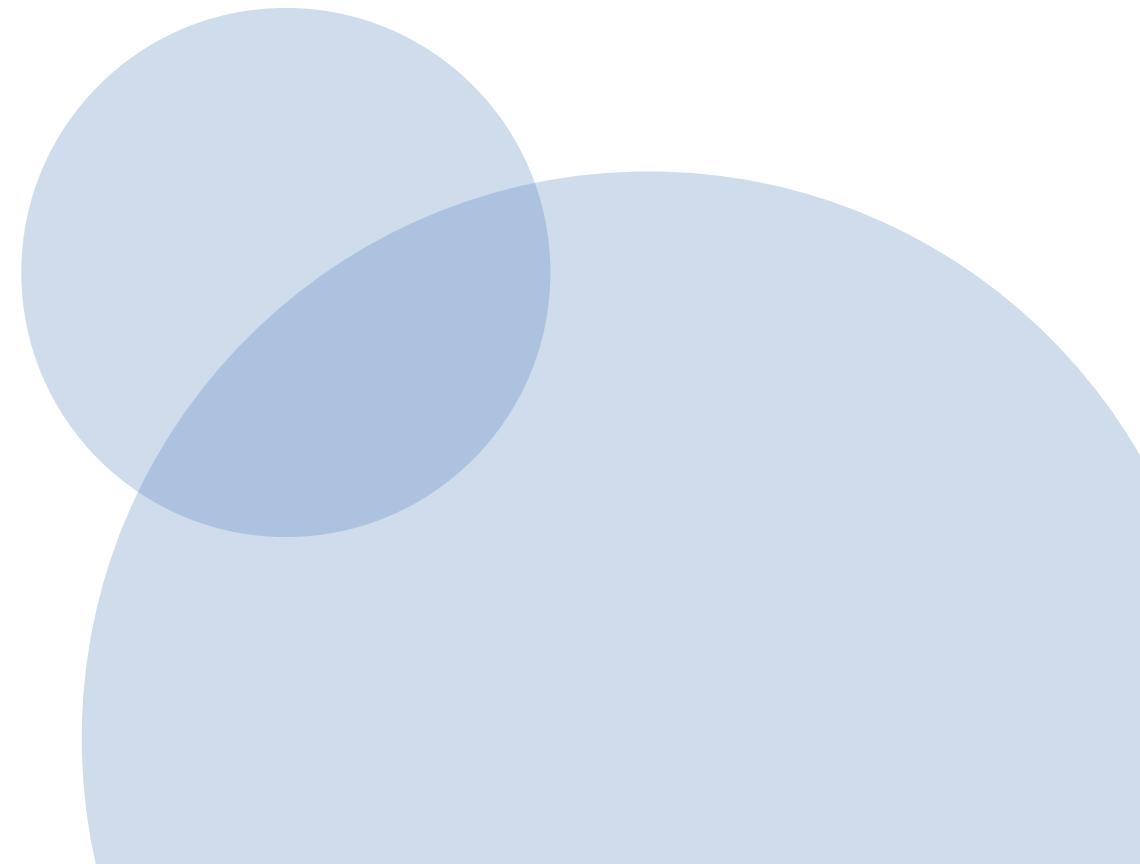


Byte

8 Bits -> 1 Byte -> 256 Værdier.

Nok til at kunne repræsentere det engelske alfabet, talrække og symboler.

Derfor er computeren oprindeligt bygget på et 8 bit system.



Bit vægte

Bit	Vægt/Værdi
1	2^0 1
2	2^1 2
3	2^2 4
4	2^3 8
5	2^4 16
6	2^5 32
7	2^6 64
8	2^7 128

decimalværdi = summen af vægtene af de aktive/tændte bits



Øvelse – binær tæller

Brug din hånd som binær tæller.

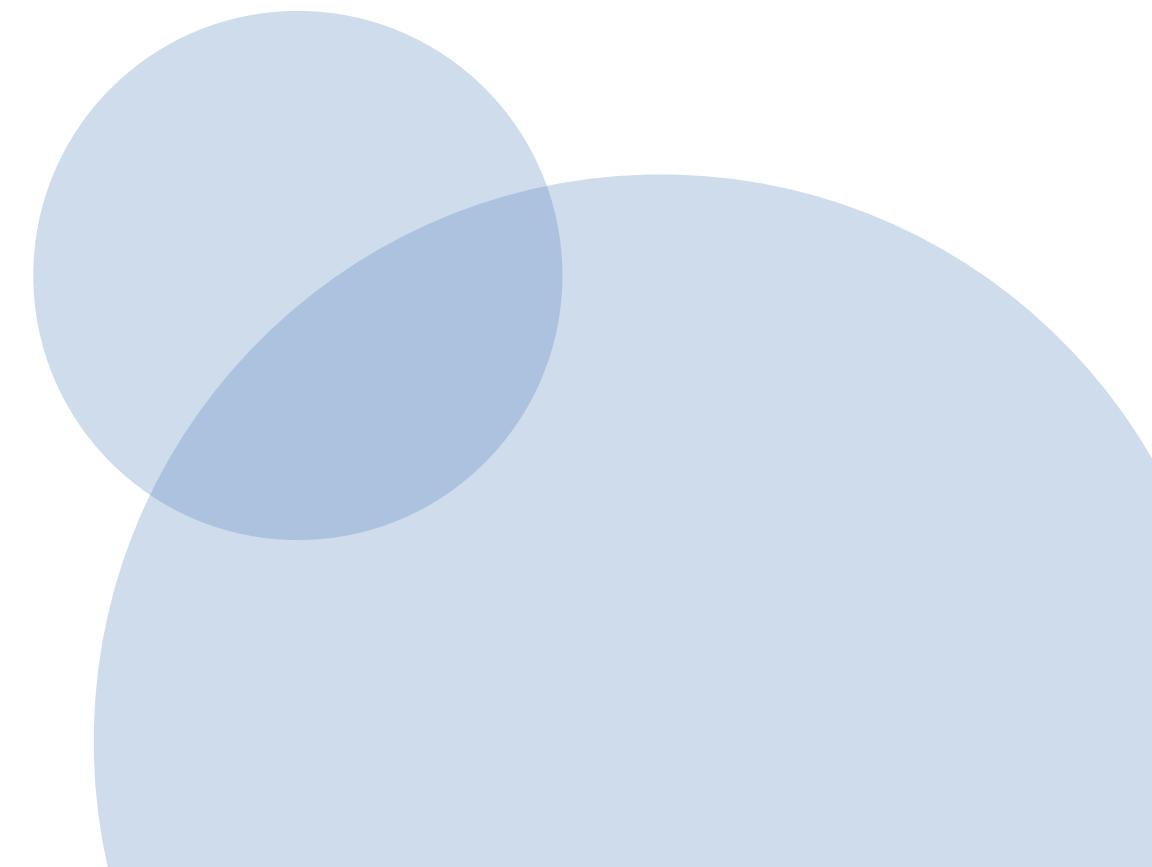
Tommelfinger = bit 0

Pegefinger = bit 1

.....

...

Tæl fra 0 → så høj en værdi som det er muligt med en hånd.



Binær til decimal

7	6	5	4	3	2	1	0
128	64	32	16	8	4	2	1
0	0	1	0	1	1	0	1

45

Binær til decimal

7	6	5	4	3	2	1	0
128	64	32	16	8	4	2	1
0	1	1	1	1	0	1	0

122

Binær til decimal

7	6	5	4	3	2	1	0
128	64	32	16	8	4	2	1
1	1	1	1	1	1	0	0

252

Decimal til binær

39

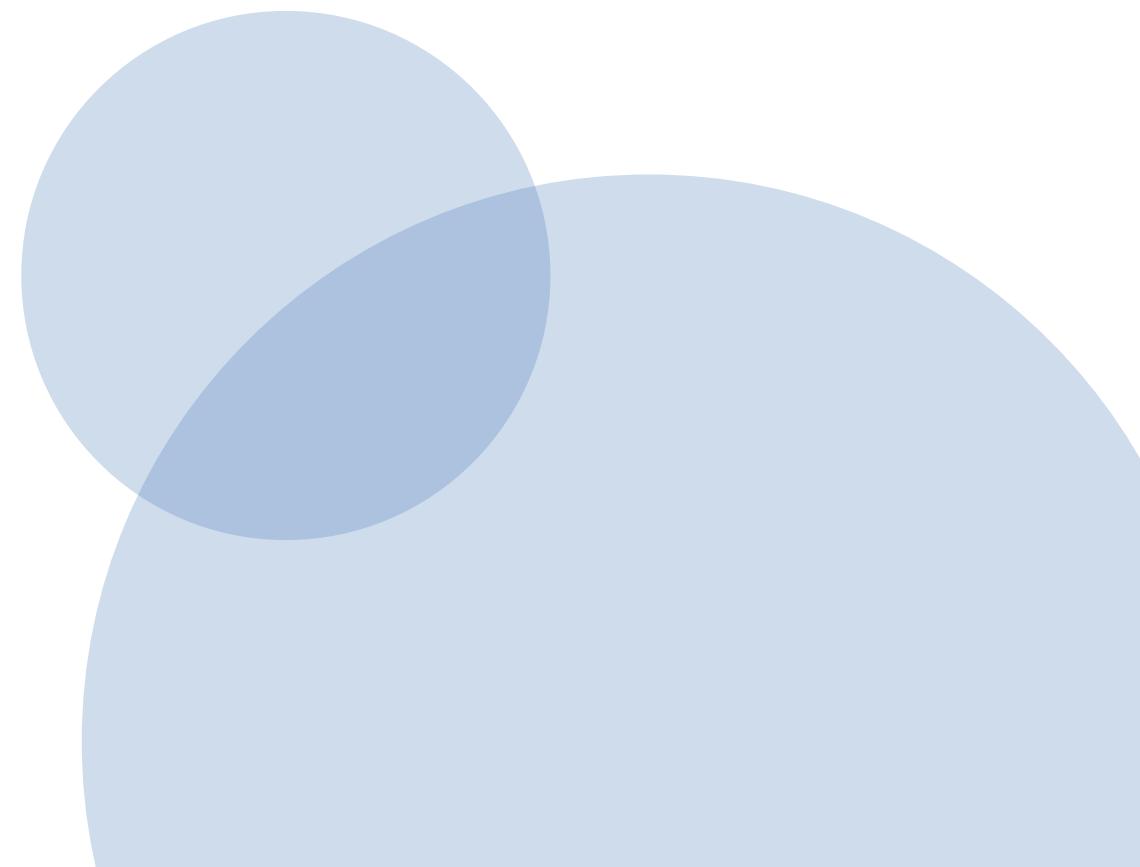
7	6	5	4	3	2	1	0
128	64	32	16	8	4	2	1
0	0	1	0	0	1	1	1

Øvelse – kodning og afkodning

I skal i par sende en data-pakke til hinanden, som skal sendes i binær form.

Senderen skal konvertere et kort tekst besked via ASCII tabellen til binær format.

Modtageren skal konvertere den binære besked til tekst via ASCII tabellen.



Hexadecimal

Farver i et computersystem repræsenteres også som binære data.

Hver pixel på en skærm har en farve.

Hver farve repræsenteres af et antal bits.

På jeres egne computere er det normal med en 24-bit farvedybde.

8 bits til rød farve, 8 bits til grøn farve og 8 bits til blå.

Hvor mange farver er det så muligt at bruge?

16.777.216

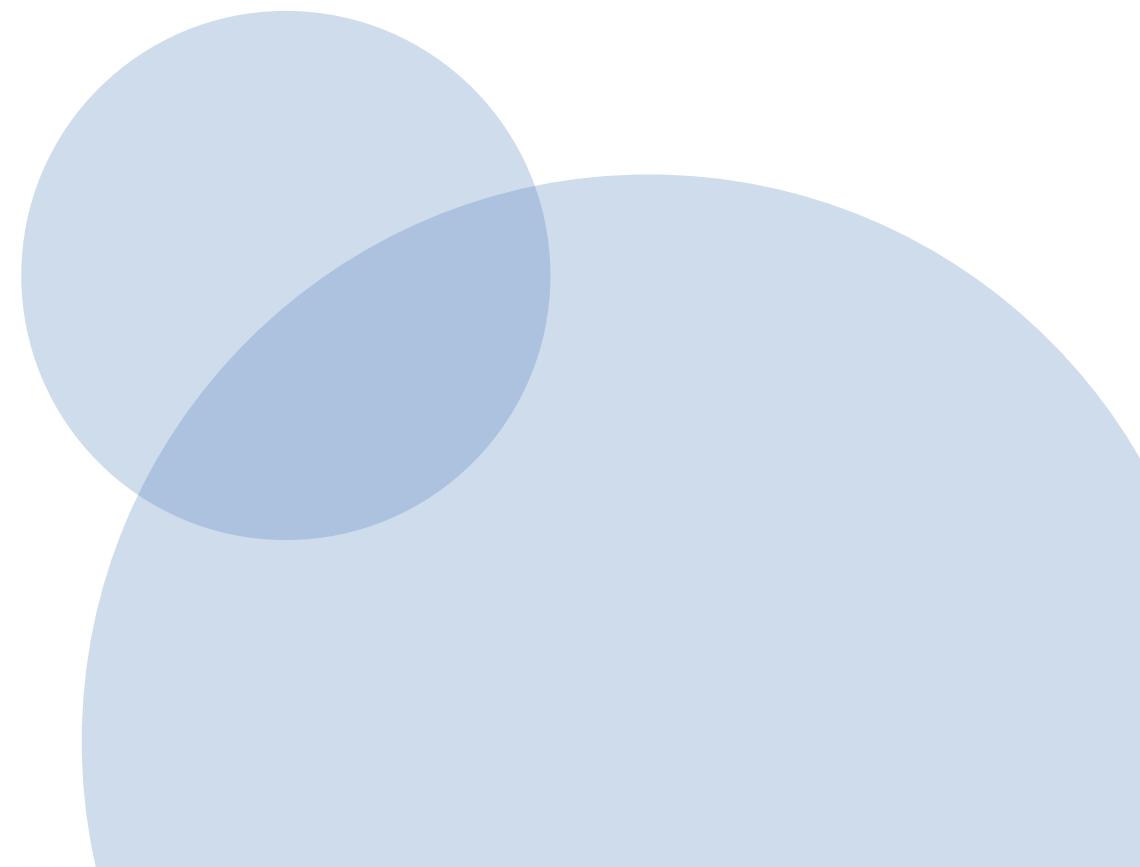
Farven "plum" har den binære værdi: 100001000011000101111001

For at gøre det lettere at læse og skrive multi-byte værdier bruger vi HEX notation.

#843179

0x843179

Det kan ses som en slags kompression af lang binære tal.



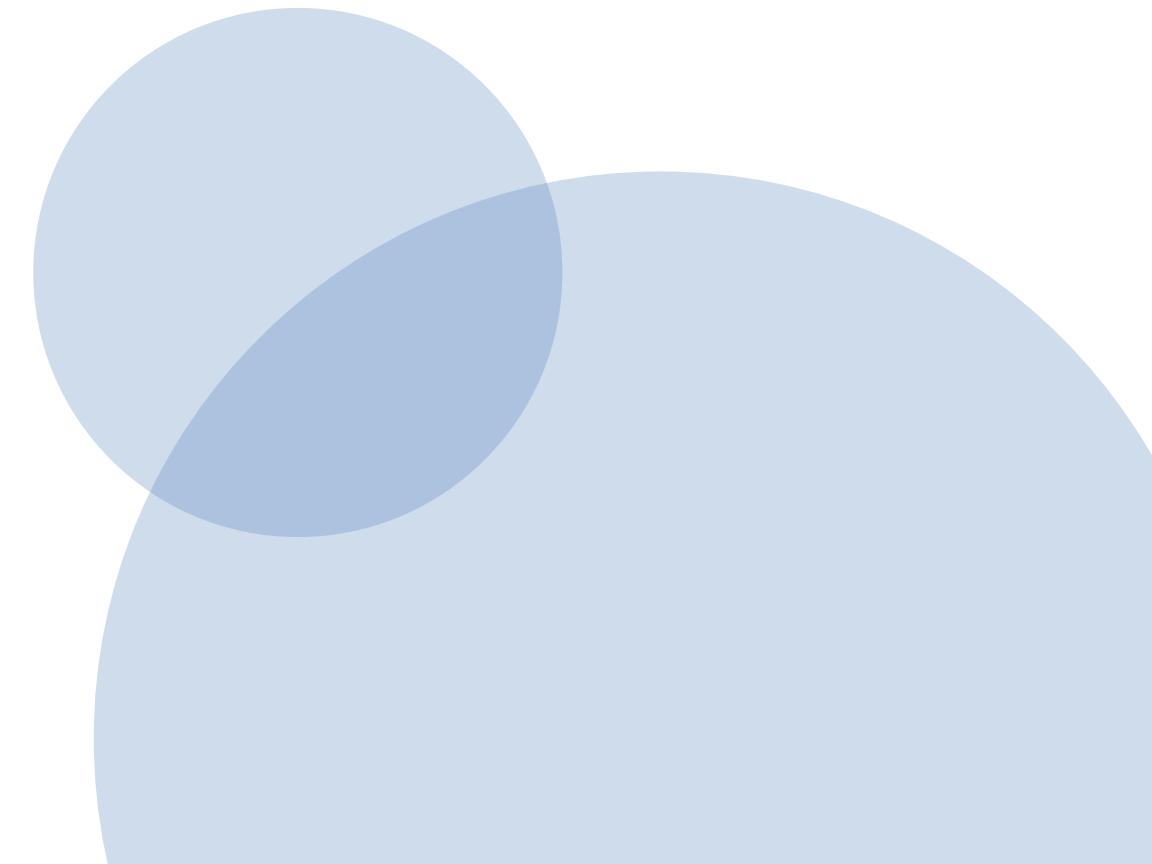
Hexadecimal

Hex er base 16.

0 1 2 3 4 5 6 7 8 9 A B C D E F

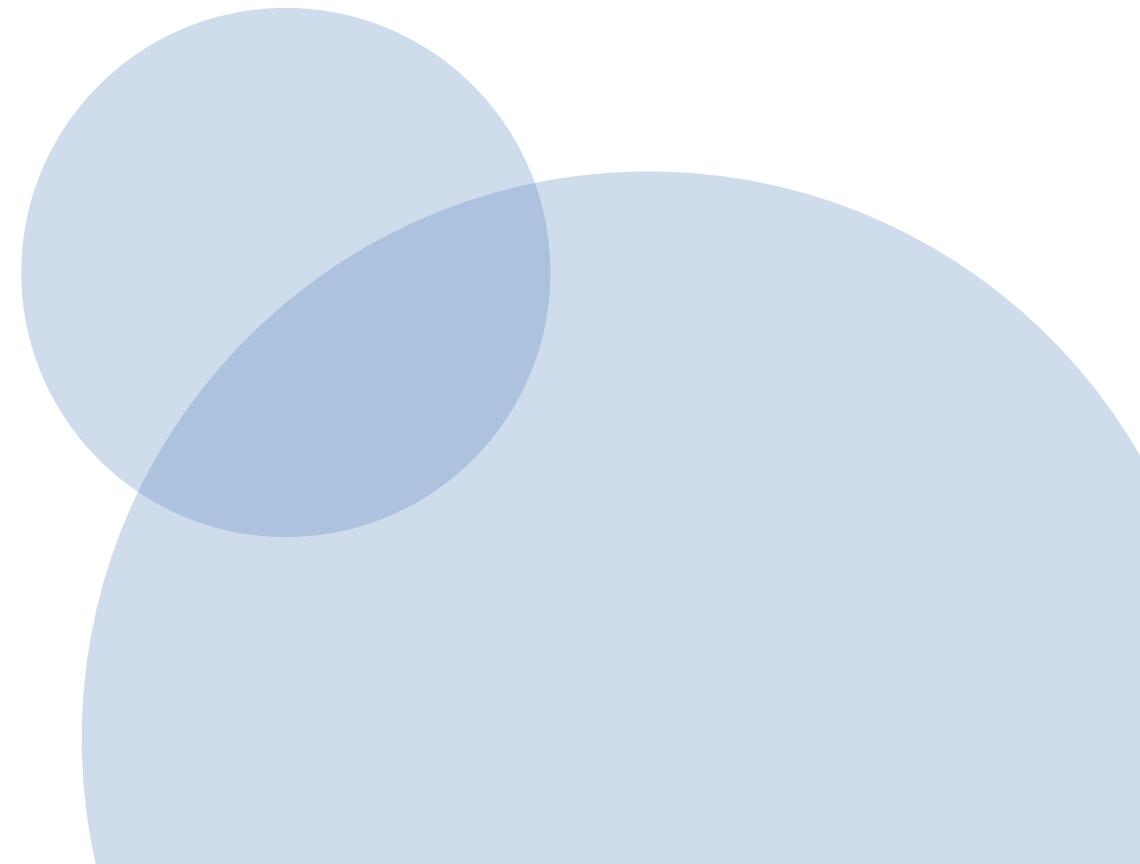
Hver halve byte udgør et hex ciffer med værdien 0-F

1100	0011
C	3
1111	1111
F	F



Øvelse

- Brug Cyberchef til at konvertere en længere besked til binær eller hexadeicmal.
- Send den med mail eller andet til naboen.
- Naboen skal afkode beskeden.



Oktaler

Oktal = 3 bits = 2^3 = 8 kombinationer

000 = 0

001 = 1

010 = 2

011 = 3

100 = 4

101 = 5

110 = 6

111 = 7

Oktaler bruges bl.a. til at tildele rettigheder til filer og mapper i Linux.

Hver fil eller mappe kan tildeles en kombination af rettigheder til at Læse, Skrive, Eksekvere.
Dette er for ejer, brugere, og andre.

Altså her bruges $3 \text{ bit} \times 3 = 3$ oktale værdier.

Alt efter hvilke bit der tændes i hver oktale, gives der rettigheder.

Oktale talværdier

Tilladelse	Symbol	Værdi (binær)	Oktal værdi
læse	r	100	4
skrive	w	010	2
eksekver	x	001	1

Der er et oktalt tal for hver rettighedsområde (ejer, bruger, resten)

Så hvis man tildeler de tre oktale tal 777 til en fil eller en mappe, er der fulde rettigheder for alle.

777 = 111 111 111

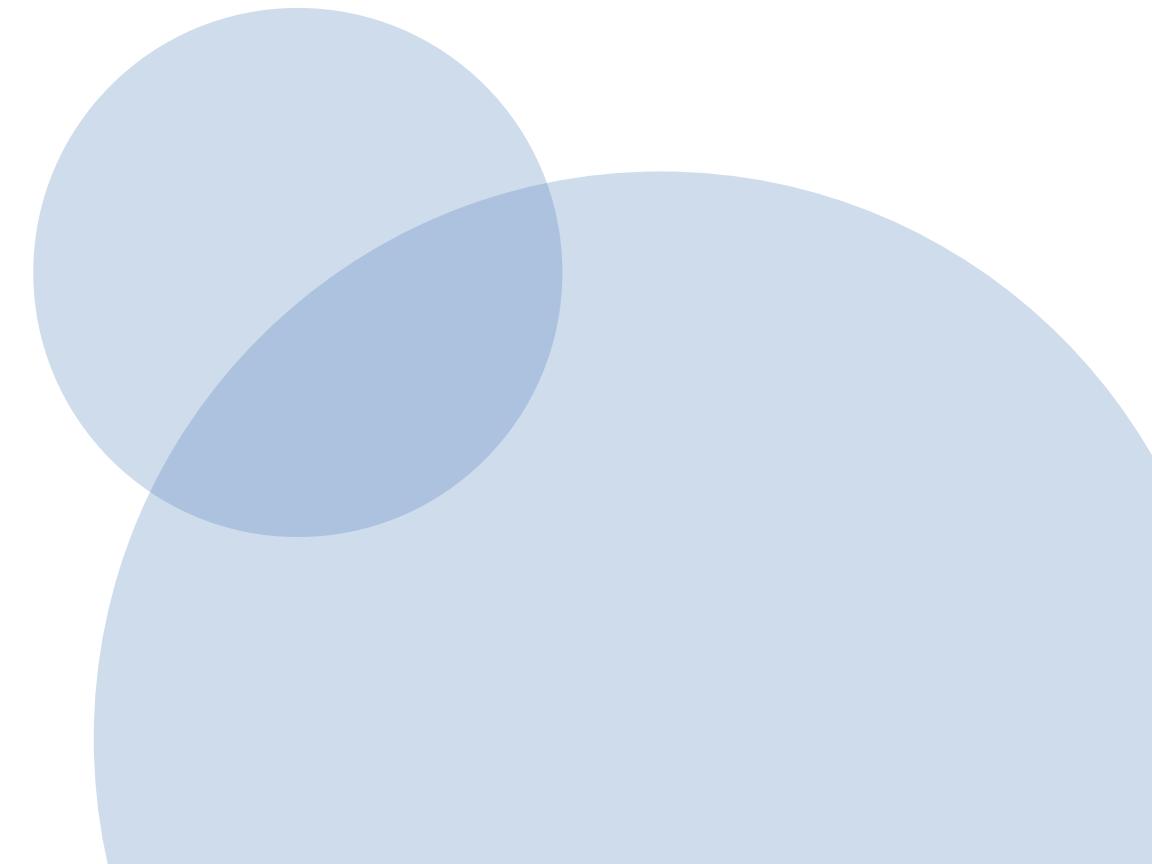
Der bruges 9 bits her.

Så det passer ikke med 1 eller to bytes, så man låner en bit i en byte.

0b00000001 11111111

Eller

0b11111111 10000000



ASCII

dec	hex	symbol									
000	00	(nul)	032	20	„	064	40	@	096	60	‘
001	01	(soh)	033	21	!	065	41	A	097	61	a
002	02	(stx)	034	22	”	066	42	B	098	62	b
003	03	(etx)	035	23	#	067	43	C	099	63	c
004	04	(eot)	036	24	\$	068	44	D	100	64	d
005	05	(enq)	037	25	%	069	45	E	101	65	e
006	06	(ack)	038	26	&	070	46	F	102	66	f
007	07	(bel)	039	27	,	071	47	G	103	67	g
008	08	(bs)	040	28	(072	48	H	104	68	h
009	09	(tab)	041	29)	073	49	I	105	69	i
010	0A	(lf)	042	2A	*	074	4A	J	106	6A	j
011	0B	(vt)	043	2B	+	075	4B	K	107	6B	k
012	0C	(np)	044	2C	,	076	4C	L	108	6C	l
013	0D	(cr)	045	2D	-	077	4D	M	109	6D	m
014	0E	(so)	046	2E	.	078	4E	N	110	6E	n
015	0F	(si)	047	2F	/	079	4F	O	111	6F	o
016	10	(dle)	048	30	0	080	50	P	112	70	p
017	11	(dc1)	049	31	1	081	51	Q	113	71	q
018	12	(dc2)	050	32	2	082	52	R	114	72	r
019	13	(dc3)	051	33	3	083	53	S	115	73	s
020	14	(dc4)	052	34	4	084	54	T	116	74	t
021	15	(nak)	053	35	5	085	55	U	117	75	u
022	16	(syn)	054	36	6	086	56	V	118	76	v
023	17	(etb)	055	37	7	087	57	W	119	77	w
024	18	(can)	056	38	8	088	58	X	120	78	x
025	19	(em)	057	39	9	089	59	Y	121	79	y
026	1A	(eof)	058	3A	:	090	5A	Z	122	7A	z
027	1B	(esc)	059	3B	;	091	5B	[123	7B	{
028	1C	(fs)	060	3C	<	092	5C	\	124	7C	
029	1D	(gs)	061	3D	=	093	5D]	125	7D	}
030	1E	(rs)	062	3E	>	094	5E	^	126	7E	~
031	1F	(us)	063	3F	?	095	5F	-	127	7F	(del)