

Digitalni formati zapisa podataka

Filip Marić

januar, 2008.

1 Digitalizacija

Prodor računara kao posledica tehnološkog napretka. U poslednje vreme smo svedoci izrazito snažnog prodora računara u većinu oblasti svakodnevnog života. Od mašina koje su pre svega služile za izvođenje matematičkih i naučnih proračunavanja, računari su danas postali sredstvo za kućnu zabavu (gledanje filmova, slušanje muzike), izvor informacija (internet, web) i nezaobilazno sredstvo u komunikaciji (e-mail, chat, instant messaging, video conferencing, voip). Ovako nagli razvoj i proširivanje osnovne namene računara je prouzrokovan velikim porastom količine informacija koje su zapisane u digitalnom formatu. Ovo je opet prouzrokovano tehnološkim napretkom koji je omogućio jednostavno i jeftino digitalizovanje signala, skladištenje velike količine digitalno zapisanih informacija kao i njihov brz prenos i obradu.

Tehnološka konvergencija. Takođe, svedoci smo razvoja velikog broja elektronskih uređaja, poput mobilnih telefona, PDA uređaja, CD/DVD/MP3 playera, digitalnih foto aparata i kamera, igračkih konzola i slično. Razvoj ovih sprava je tesno povezan sa razvojem digitalnih računara, a ono što je svima zajedničko je da su to uređaji koji manipulišu digitalno zapisanim informacijama. Svi ovi uređaji se opravdano mogu smatrati specijalizovanim malim računarima. U prilog ovoj činjenici može se navesti pojava poznata pod imenom *tehnološka konvergencija* (engl. technological convergence) koja dovodi do toga da postepeno nestaju oštre granice između raznih tipova uređaja i uređaji polako počinju da dobijaju sve više različitih funkcija. Tako npr. većina mobilnih telefona može da služi kao digitalni foto aparat, igračka konzola PlayStation 2 može da služi i kao DVD player i slično.

Kontinualna priroda signala. Većina podataka koje računari zapisuju nastaju zapisivanjem prirodnih signala. Najznačajnije primeri signala, svakako, predstavljaju zvuk i slika, ali se takođe pod signalima mogu podrazumevati i ultrazvučni signali, EKG signali, zračenja različite vrste itd.

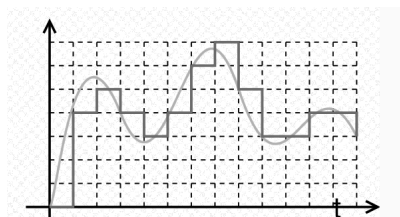
Signali koji nas okružuju u prirodi u većini slučajeva se prirodno mogu predstaviti neprekidnim funkcijama. Npr. zvučni signal predstavlja promenu pritiska vazduha u zadatoj tački posmatranja i to kao neprekidnu funkciju vremena. Slika predstavlja intenzitet svetlosti određene boje u datom vremenskom trenutku i to kao neprekidnu funkciju prostora.

Analogni zapis. Osnovna tehnika koja se primenjuje kod analognog zapisa signala je da se kontinualne promene signala koji se zapisuje opišu kontinualnim promenama određenog svojstva medijuma na kojem se signal zapisuje. Tako npr. promene pritiska vazduha koji predstavlja zvučni signal direktno odgovaraju promenama nivoa namagnetisanja na magnetnoj traci na kojoj se zvuk analogno zapisuje. Količina boje na papiru direktno odgovara intenzitetu svetlosti u vremenskom trenutku kada je fotografija bila snimljena. Dakle, kao što i samo ime kaže, analogni zapis uspostavlja *analogiju* između određenog svojstva medijuma na kome je signal zapisan i samog signala koji je zapisan.

Osnovna prednost analogne tehnologije je činjenica da su ulaganja potrebna da bi se dobio zapis veoma mala, ukoliko se zadovoljimo relativno niskim kvalitetom. Tako su npr. još drevni narodi mogli da naprave nekakav zapis zvuka uz pomoću jednostavne igle prikačene na trepereću membranu.

Osnovi problem analogne tehnologije je što je izrazito teško na medijumu napraviti skoro identičan zapis signala koji se zapisuje. Takođe, problem predstavlja i inherentna nestalnost medijuma, njegova promenljivost tokom vremena i podložnost spoljašnjim uticajima. Sa obzirom da male varijacije medijuma direktno dovode do varijacije zapisanog signala, vremenom neizbežno dolazi do pada kvaliteta analogno zapisanog signala. Obrada analogno zapisanih signala je izuzetno komplikovana i za svaku vrstu obrade signala, potrebno je da postoji uređaj koji je specijalizovan za tu vrste obrade.

Digitalni zapis. Osnovna tehnika koja se koristi kod digitalnog zapisa signala je da se vrednost signala izmeri u određenim vremenskim trenucima odnosno određenim tačkama prostora i da se onda na medijumu zapišu izmerene vrednosti. Ovim je svaki digitalno zapisani signal predstavljen nizom brojeva koji se nazivaju odbirci (engl. sample). Svaki od brojeva predstavlja vrednost signala u jednoj tački diskretizovanog domena. Sa obzirom da izmerene vrednosti takođe dolaze sa kontinualne skale, neophodno je izvršiti i diskretizaciju kodomena, odnosno dopustiti zapisivanje samo određenog broja nivoa različitih vrednosti.



Slika 1: Digitalizacija zvučnog signala

Problemi digitalnog zapisa. Ovakav zapis predstavlja diskretnu aproksimaciju polaznog signala. Pitanje koje se neizbežno postavlja je koliko često je potrebno vršiti merenje, kako bi se polazni kontinualni signal mogao verno rekonstruisati. Odgovor daje Nyquist-Shannon-ova teorema o odabiranju koja kaže da je signal dovoljno meriti dva puta češće od najveće frekvencije koja sa u njemu javlja. Npr. pošto čovekovo uho čuje frekvencije do 20KHz, dovoljno je da frekvencija odabiranja bude oko 40KHz.

Prednosti digitalnog zapisa. Osnovi problem implementacije digitalnog zapisa predstavlja činjenica da je neophodno imati veoma razvijenu tehnologiju da bi se uopšte stiglo do iole upotrebljivog zapisa. Izuzetno je komplikovano napraviti uređaj koji je u stanju da 40 hiljada puta izvrši merenje intenziteta zvuka. Jedna sekunda zvuka se predstavlja sa 40 hiljada brojeva, za čiji je zapis neophodna gotovo cela jedna sveska. Ovo je osnovni razlog zbog čega se digitalni zapis istorijski javio ovoliko kasno. Međutim, kada se došlo do tehnološkog nivoa koji omogućava digitalni zapis, prednosti koje je donela digitalna tehnologija su izuzetne.

Osnovni problem analogne tehnologije predstavlja činjenica da je za postizanje visokog kvaliteta zapisa potrebno imati medijume visokog kvaliteta. Kvalitet reprodukcije digitalnog zapisa ne zavisi od toga kakav je kvalitet medija na kome su podaci zapisani, dok god je medijum dovoljnog kvaliteta da se zapisani brojevi mogu razaznati.

Takodje, kvarljivost koja je inherentna za sve medije postaje nebitna. Npr. papir vremenom žuti što prouzrokuje pad kvaliteta analognih fotografija tokom vremena. Međutim, ukoliko bi papir sadržao zapis brojeva koji predstavljaju vrednosti boja u pikselima digitalno zapisane fotografije, činjenica da papir žuti ne bi predstavljala problem dok god se brojevi mogu razpoznati.

Digitalni zapis omogućava kreiranje apsolutno identičnih kopija što dalje omogućava prenos podataka na daljinu. Npr. ukoliko izvršimo fotokopiranje fotografije, napravljena fotokopija je neuporedivog lošijeg kvaliteta od originala. Međutim, ukoliko prekopiramo CD na kome su zapisani brojevi koji čine zapis neke fotografije, kvalitet ostaje apsolutno isti. Ukoliko bi se dva CD-a pogledala pod mikroskopom, oni bi izgledali prilično različito, međutim to ne predstavlja problem dok god se brojevi koji su na njima zapisani mogu raspoznati.

Obrada digitalno zapisanih podataka se svodi na matematičku manipulaciju brojevima i ne zahteva više korišćenje specijalizovanih mašina.

2 Zapis brojeva

Procesom digitalizacije se podaci pretvaraju u niz brojeva koji se unose u računar, tako da je veoma važno omogućiti zapis brojeva u računarima. Dekadni brojevni sistem koji ljudi koristi u svakodnevnom životu nije pogodan jer zahteva azbuku od 10 različitih simbola (cifara). Tehnologija izrade računara i medija za zapis podataka omogućava izgradnju elemenata koji imaju dva diskretna stanja što za zapis brojeva daje azbuku od samo 2 različita simbola. Tako npr. „žica” u računaru propuštaju električnu struju i nalaze se pod određenim naponom. Iako je napon kontinualna veličina, uspostavljanjem određenog praga je moguće je smatrati da postoje dva diskretna stanja i da žica može ili biti pod naponom i predstavljati cifru 1 ili ne biti pod naponom i predstavljati cifru 0. Takođe, npr. laserski zrak na površini kompakt diska buši rupice. Polje koje nije izbušeno može predstavljati cifru 1, a ono koje je izbušeno cifru 0. Videćemo da je azbuka od samo dva simbola u potpunosti dovoljna za predstavljanje svih vrsta brojeva, pa samim tim i svih digitalno zapisanih podataka. Ovo je moguće zahvaljujući binarnom brojevnom sistemu koji predstavlja osnovu digitalnih računara. U računarstvu se takođe koriste i heksadekadni brojevni sistem i ređe oktalni brojevni sistem zbog toga što omogućavaju izuzetno jednostavnu konverziju između njih i binarnog sistema.

2.1 Neoznačeni brojevi

Čitanje brojeva. Pretpostavimo da je data osnova b takva da je b prirodan broj veći od 1. Niz cifara $(a_n a_{n-1} \dots a_1 a_0)_b$ predstavlja zapis broja u osnovi b , pri čemu za svaku cifru a_i važi $0 \leq a_i < b$.

Klasična definicija zapisa broja u osnovi b :

$$(a_n a_{n-1} \dots a_1 a_0)_b = \sum_{i=0}^n a_i \cdot b^i = a_n \cdot b^n + a_{n-1} \cdot b^{n-1} + \dots + a_1 \cdot b + a_0$$

Npr. $(101101)_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2 + 1 = 32 + 8 + 4 + 1 = 45$.

Npr. $(3245)_8 = 3 \cdot 8^3 + 2 \cdot 8^2 + 4 \cdot 8 + 5 = 3 \cdot 512 + 2 \cdot 64 + 4 \cdot 8 + 5 = 1536 + 128 + 32 + 5 = 1701$.

Hornerova definicija zapisa broja u osnovi b :

$$(a_n a_{n-1} \dots a_1 a_0)_b = (\dots ((a_n \cdot b + a_{n-1}) \cdot b + a_{n-2}) \dots + a_1) \cdot b + a_0$$

Npr. $(3245)_8 = 1701$, jer

3	2	4	5
3	$3 \cdot 8 + 2 = 26$	$26 \cdot 8 + 4 = 212$	$212 \cdot 8 + 5 = 1701$

Hornerov algoritam za izračunavanje vrednosti broja se može zapisati kao:

```
int x = 0;
for (int i = n; i >= 0; i--)
    x = x * b + a[i];
```

Hornerova definicija je računski optimalnija, jer je u svakom koraku dovoljno izvršiti samo jedno množenje i jedno sabiranje.

Pisanje brojeva. Pošto za svaku cifru a_i u zapisu broja u osnovi b važi da je $0 \leq a_i < b$, na osnovu Hornerove definicije broja $x = (a_n a_{n-1} \dots a_1 a_0)_b$, jasno je da mora da važi da je $a_0 = x \bmod b$. Takođe, $x \operatorname{div} b = (a_n a_{n-1} \dots a_1)$. Dakle, izračunavanjem celobrojnog količnika i ostatka pri deljenju sa b , određena je poslednja cifra broja x i broj koji se dobije uklanjanjem poslednje cifre iz zapisa. Ukoliko se isti postupak primeni na dobijeni količnik, dobili smo rekurzivni algoritam koji omogućava da se odrede sve cifre u zapisu broja x . Postupak prestaje u trenutku kada količnik postane 0.

Npr. $1701 = (3245)_8$ jer je $1701 = 212 \cdot 8 + 5 = (26 \cdot 8 + 4) \cdot 8 + 5 = ((3 \cdot 8 + 2) \cdot 8 + 4) \cdot 8 + 5$. Ovaj postupak se može prikazati i tablično

1701	212	26	3	0
5	4	2	3	

Prva vrsta tablice sadrži celobrojne količnike, a druga ostatke pri deljenju sa osnovom b . Primetimo da se zapis formira tako što se dobijeni ostaci pročitaju unatrag.

Ovaj algoritam prikazan u pseudo kodu:

```
i = 0;
while (x > 0) {
    a[i++] = x % 10;
    x /= 10;
}
reverse(a, i); /* Obrćemo niz a */
```

Prikazani algoritam i Hornerov algoritam su međusobno komplementarni i može se smatrati da predstavljaju "dve strane iste medalje".

Direktno prevođenje između heksadekadnog i binarnog sistema. Osnovni razlog korišćenja heksadekadnog sistema je mogućnost jednostavnog prevođenja brojeva između binarnog i heksadekadnog sistema. Heksadekadni sistem pomaže da se binarni sadržaj memorije kompaktno zapiše sa mnogo manjim brojem cifara. Tako se npr. 32-bitni sadržaj može zapisati korišćenjem samo 8 heksadekadnih cifara.

$$(1011\ 0110\ 0111\ 1100\ 0010\ 1001\ 1111\ 0001)_2 = (B67C29F1)_{16}$$

Prevođenje se vrši tako što se grupišu četiri po četiri binarne cifre, krenuvši od pozadi i svaka četvorka se zasebno prevede u odgovarajuću heksadekadnu cifru na osnovu sledeće tabele:

heksa	binarno	heksa	binarno	heksa	binarno	heksa	binarno
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Zapisi fiksirane dužine U računarima se obično odvajaju fiksirani broj binarnih cifara za zapis svakog broja. Takve zapise označavamo sa $(\dots)_b^n$. Ukoliko je broj cifara potrebnih za zapis broja kraći od tražene dužine zapisa, broj se proširuje vodećim nulama. Npr. $55 = (0011\ 0111)_2^8$. Ograničavanjem broja cifara ograničava se i raspon brojeva koje je moguće zapisati i to na raspon od 0 do $2^n - 1$. U sledećoj tabeli su dati rasponi za najčešće korišćene binarne zapise:

broj bita	raspon
8	0 - 255
16	0 - 65535
32	0 - 4294967295

2.2 Označeni brojevi

Pod označenim brojevima se podrazumevaju celi brojevi kod kojih se u zapis uključuje i zapisivanje znaka broja (+ ili -). Sa obzirom da računari koriste binarni brojevni sistem, razmatraćemo samo zapise označenih brojeva u binarnom brojevnom sistemu. Postoji više tehnika zapisivanja označenih brojeva od kojih su najčešće u upotrebi *označena apsolutna vrednost* i *potpuni komplement*.

Označena apsolutna vrednost. Zapis se formira tako što se u registar unapred fiksirane dužine n , na prvu poziciju upiše znak broja, a na preostalim $n - 1$ poziciju upiše zapis apsolutne vrednosti broja. Pošto se za zapis koriste samo dva simbola (0 i 1), postuje se dogovor da se znak + zapisuje simbolom 0, a znak - se zapisuje simbolom 1. Ovim se postiže da pozitivni brojevi imaju identičan zapis kao da su u pitanju neoznačeni brojevi.

$$\text{Npr. } +100 = (0\ 1100100)_2^8, \quad -100 = (1\ 1100100)_2^8$$

Osnovni problem zapisa u obliku označene apsolutne vrednosti je činjenica da se osnovne aritmetičke operacije teško izvode ukoliko su brojevi zapisani na ovaj naćni.

Potpuni komplement. Ovaj zapis označenih brojeva zadovoljava sledeće uslove:

1. Nula i pozitivni brojevi se zapisuju na isti način kao da su u pitanju neoznačeni brojevi, pri čemu u njihovom zapisu prva cifra mora da bude 0.
2. Sabiranje se sprovodi na isti način kao da su u pitanju neoznačeni brojevi, pri čemu se prenos sa poslednje pozicije zanemaruje.

Tako se npr. broj $+100$ u potpunom komplementu zapisuje kao $(01100100)_2^8$. Nula se zapisuje kao $(00000000)_2^8$. Zapis broja -100 u obliku $(\dots)_2^8$ se može odrediti na sledeći način. Zbir brojeva -100 i $+100$ mora da bude 0.

	binarno	dekadno
	????????	-100
+	01100100	+100
	$\cancel{1}$ 00000000	0

Analizom traženog sabiranja cifru po cifru, počevši od poslednje, sledi da se -100 mora zapisati kao $(10011100)_2^8$.

Do ovoga je moguće doći i na sledeći način. Ukoliko je poznat zapis broja x , zapis njemu suprotnog broja je moguće odrediti iz uslova da je $x + (-x) = (100\dots00)_2^{n+1}$. Pošto je $(100\dots00)_2^{n+1} = (11\dots11)_2^n + 1$, zapis broja $(-x)$ je moguće odrediti tako što se izračuna $(11\dots11)_2^n - x + 1$. Izračunavanje razlike $(11\dots11)_2^n - x$ se svodi na *komplementiranje* svake pojedinačne cifre broja x . Tako se određivanje zapisa broja -100 može opisati na sledeći način:

	01100100	+100
	10011011	komplementiranje
+	1	
	10011100	

Kao što je traženo, zapisi svih pozitivnih brojeva i nule počinju cifrom 0 dok zapisi negativnih brojeva počinju sa 1.

Zapis broja $(100\dots00)_2^n$ je sam sebi komplementaran, a pošto počinje cifrom 1, uzima se da on predstavlja zapis najmanjeg negativnog broja. Tako dobijamo da je u zapisu potpunog komplementa $(\dots)_2^n$ moguće zapisati brojeve od -2^{n-1} do $2^{n-1} - 1$. U sledećoj tabeli su dati rasponi za najčešće korišćene zapise:

broj bita	raspon
8	-128 - +127
16	-32768 - +32767
32	-2147483648 - +2147483647

2.3 Razlomljeni brojevi

Za zapis razlomljenih brojeva se najčešće koristi i tzv. *pokretni zarez* (engl. floating point). Standard koji definiše zapis brojeva u pokretnom zarezu se naziva *IEEE 754*.

3 Zapis teksta

ISO definiše tekst (ili dokument) kao "informaciju namenjenu ljudskom sporazumevanju koja može biti prikazana u dvodimenzionalnom obliku... Tekst se

sastoji od grafičkih elemenata kao što su karakteri, geometrijski ili fotografski elementi ili njihove kombinacije, koji čine sadržaj dokumenta.” Dakle, iako obično tekst zamišljamo kao dvodimenzioni objekat, u računarima se tekst predstavlja kao jednodimenzioni (linearni) niz karaktera. Potrebno je, dakle, uvesti specijalne karaktere koji označavaju prelazak u novi red, tabulator, kraj teksta i slično.

Osnovna ideja koja omogućava zapis teksta u računarima je da se svakom karakteru pridruži određeni (neoznačeni) ceo broj i to na unapred dogovoreni način. Ovi brojevi se nazivaju *kodovima karaktera* (engl. character codes).

Tokom razvoja i ekspanzije računarstva, broj karaktera koje je bilo poželjno kodirati je postajao sve veći i veći. Pošto je u početku razvoja englesko govorno područje bilo dominantno osnovno je bilo predstaviti sledeće karaktere:

- Mala slova engleskog alfabeta: a, b, ... , z
- Velika slova engleskog alfabeta: A, B, ... , Z
- Interpunkcijske znakove: Npr. , . : ; + * - _ () [] { }
- Specijalne znake: Npr. kraj reda, tabulator, ...

Specijalnim karakterima se najčešće ne pridružuje zaseban grafički lik.

Standardne tabele kodova ovih karaktera su se pojavile još tokom šezdesetih godina XX veka. Najpoznatije od njih su:

- EBCDIC - IBM-ov standard, korišćen uglavnom na mainframe računarima, pogodan za bušene kartice.
- ASCII - standard iz koga se razvila većina danas korišćenih standarda za zapis karaktera.

ASCII. ASCII (American Standard Code for Information Interchange) je standard uspostavljen 1968. godine od strane ANSI (American National Standard Institute) koji definiše sedmobitni zapis koda svakog karaktera što daje mogućnost zapisivanja ukupno 128 različitih karaktera, pri čemu nekoliko karaktera ima dozvoljeno slobodno korišćenje. ISO (International Standard Organization) takođe delimično definiše ASCII tablicu kao deo standarda ISO 646 (US).

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	STX	SOT	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

Slika 2: ASCII tablica

- Prva 32 karaktera $(00)_{16}$ do $(1F)_{16}$ su specijalni kontrolni karakteri.
- 95 karaktera ima pridružene grafičke likove (engl. printable characters)

- Cifre 0-9 su predstavljene kodovima $(30)_{16}$ do $(3F)_{16}$ tako da se njihov ASCII zapis jednostavno dobija dodavanjem prefiksa 011 na njihov binarni zapis.
- Velika i mala slova se razlikuju u samo jednom bitu. Npr. *A* se kodira kao $(41)_{16}$ odnosno $(100\ 0001)_2$, dok se *a* kodira kao $(61)_{16}$ odnosno $(110\ 0001)_2$. Ovo omogućava da se konverzija veličine slova u oba smera može vršiti efikasno.
- Slova su poređana u kolacionu sekvencu u skladu sa engleskim alfabetom.

Zanimljivo je napomenuti i da različiti operativni sistemi predviđaju različito kodiranje oznake za prelazak u novi red. Tako operativni sistem Windows podrazumeva da se prelazak u novi red kodira sa dva kontrolna karaktera i to *CR* tj. $(0D)_{16}$ i *LF* tj. $(0A)_{16}$, operativni sistem Unix i njegovi derivati podrazumevaju da se koristi samo karakter *LF*, dok MacOS podrazumeva korišćenje samo karaktera *CR*.

Izmene ASCII tablice. Tokom osamdesetih godina XX veka, Jugoslovenski zavod za standarde definiše YU-ASCII (YUSCII) tako što određene interpunkcijske znakove zamenjuje našim dijakriticima:

YUSCII	ASCII	kôd	YUSCII	ASCII	kôd
Ž	@	$(40)_{16}$	ž	'	$(60)_{16}$
Š	[$(5B)_{16}$	š	{	$(7B)_{16}$
Đ	\	$(5C)_{16}$	đ		$(7C)_{16}$
Ć]	$(5D)_{16}$	ć	}	$(7D)_{16}$
Č	^	$(5E)_{16}$	č	~	$(7E)_{16}$

YUSCII je takođe deo standarda ISO 646. Osnovne mane YUSCII kodiranja je činjenica da ne poštuje abecedni poredak, kao i da su važni interpunkcijski znaci izostavljeni.

8-bitna proširenja ASCII tablice. Podaci se u računaru obično zapisuju bajt po bajt. Sa obzirom da je ASCII sedmobitan standard, ASCII karakteri se zapisuju tako što se njihov sedmobitni kôd proširi vodećom nulom. Ovo znači da jednobajtni zapisi u kojima je vodeća cifra 1, tj. raspon od $(80)_{16}$ do $(FF)_{16}$ nisu iskorišćeni. Na žalost, ovih dodatnih 128 kodova nisu dovoljni kako bi se iskodirali svi karakteri koji su potrebni za zapis tekstova van engleskog govornog područja. Zbog toga je odlučeno da se umesto jedinstvene tabele koja bi proširivala ASCII na 256 karaktera napravi nekoliko tabela koje ovo čine, pri čemu svaka od tabela sadrži karaktere potrebne za zapis određenog jezika odnosno određene grupe jezika.

Problem je što postoji dvostruka standardizacija ovako kreiranih kodnih stranica i to od strane ISO (International Standard Organization) i od strane korporacije Microsoft.

ISO definiše familiju 8-bitnih kodnih stranica koje nose zajedničku oznaku ISO 8859. Kodovi $(00)_{16}$ do $(1F)_{16}$, $(7F)_{16}$ i od $(80)_{16}$ do $(9F)_{16}$ ostaju nedefinisani ovim standardom, iako se često u praksi popunjavaju određenim kontrolnim karakterima (često podeljenih u tzv. grupe C1 i C2).

ISO 8859-1	Latin 1	većina zapadno evropskih jezika
ISO 8859-2	Latin 2	centralno i istočno evropski jezici
ISO 8859-3	Latin 3	južno evropski jezici
ISO 8859-4	Latin 4	severno evropski jezici
ISO 8859-5	Latin/Cyrillic	ćirilica većine slovenskih jezika
ISO 8859-6	Latin/Arabic	najčešće korišćeni arapski
ISO 8859-7	Latin/Greek	moderni grčki alfabet
ISO 8859-8	Latin/Hebrew	moderni hebrejski alfabet

Microsoft definiše familju 8-bitnih stranica koje se označavaju kao windows-125x. Na nekim mestima se neopravdano windows kodne stranice nazivaju ANSI. Za srpski jezik, značajne su kodne stranice:

windows-1250	centralno i istočno evropski jezici
windows-1251	ćirilica većine slovenskih jezika

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	ı	ç	£	¤	¥	¦	§	¨	©	ª	«	¬	SHY	®	—
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	

Slika 3: ISO-8859-1 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	À	Á	Â	Ã	Ä	Å	Æ	Ç	¨	Š	Š	Ž	Ž	SHY	Ž
B	°	à	á	â	ã	ä	å	æ	ç	¸	š	š	ž	ž	~	ž
C	Á	À	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	á	à	â	ã	ä	å	æ	ç	ø	é	ê	ë	ì	í	î	ï
F	đ	ñ	ò	ó	ô	õ	ö	÷	ř	ů	ú	û	ü	ý	ť	

Slika 4: ISO-8859-2 tablica

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€	?	/	?	„	…	†	‡	?	‰	Š	<	Š	Ť	Ž	Ž
9	?	‘	/	”	”	•	–	–	?	‰	š	>	š	ť	ž	ž
A	°	˘	˘	Ł	ł	À	á	§	¨	©	ª	«	¬	®	Ž	
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	Á	À	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	á	à	â	ã	ä	å	æ	ç	ø	é	ê	ë	ì	í	î	ï
F	đ	ñ	ò	ó	ô	õ	ö	÷	ř	ů	ú	û	ü	ý	ť	

Slika 5: Windows-1250 tablica

UNICODE. Iako navedene kodne strane omogućuju kodiranje tekstova koji nisu na engleskom jeziku nije moguće npr. u istom tekstu mešati ćirilicu i

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8																
9																
A	NBSP	Ё	Ѣ	Ѓ	Є	Ѕ	І	Ї	Ј	Љ	Њ	Ѧ	Њ	Њ	SHY	Љ
B	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
C	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
D	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
E	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F	№	ё	ђ	ѓ	є	ѕ	і	ї	ј	љ	њ	ѧ	ѧ	ѧ	Љ	Љ

Slika 6: ISO-8859-5 tablica

našu latinicu. Takođe, azijskim jezicima nije dovoljno 256 mesta za zapis svih karaktera. Pošto je kapacitet računara narastao, postepeno se krenulo sa standardizacijom skupova karaktera koji karaktere kodiraju sa više od jednog bajta. Kasnih osamdesetih godina XX veka, dve velike organizacije su pokušale standardizaciju tzv. Univerzalnog skupa karaktera (engl. Universal Character Set - UCS). To su bili ISO, kroz standard 10646 i projekat UNICODE organizovan i finansiran uglavnom od strane američkih firmi koje su se bavile proizvodnjom višejezičkog softvera (Xerox Parc, Apple, ...).

ISO 10646 je zamišljen kao četvorobajtni standard. Pri tome se prvih 65536 karaktera koriste kao osnovni višejezični skup karaktera, dok je ostali prostor ostavljen kao proširenje za drevne jezike, celokupnu naučnu notaciju i slično.

UNICODE je za cilj imao da bude:

- Univerzalan (UNIversal) - sadrži sve savremene jezike sa pismom
- Jedinstven (UNIque) - bez dupliranja karaktera - kodiraju se pisma, a ne jezici
- Uniforman (UNIform) - svaki karakter sa istim brojem bitova - 16

Vremenom su se pomenuta dva projekta združila i nastao je jedinstven standard koji se jednostavno naziva UNICODE. UNICODE svakom karakteru dodeljuje dvobajtni kod (tako da kôd svakog karaktera sadrži tačno 4 heksadekadne cifre). Ovim je moguće dodeliti kodove za ukupno 65536 različita karaktera.

U sledećoj tabeli je naveden raspored određenih grupa karaktera:

0020-007E	ASCII printable
00A0-00FF	Latin-1
0100-017F	Latin extended A (osnovno proširenje latinice sadrži sve naše dijakritike)
0180-027F	Latin extended B
...	
0370-03FF	Grčki
0400-04FF	Ćirilica
...	
2000-2FFF	specijalni karakteri
3000-3FFF	CJK (Chinese-Japanese-Korean) simboli
...	

UTF-8. Unicode standard u suštini predstavlja veliku tabelu koja svakom karakteru dodeljuje broj. Standardi koji opisuju kako se niske karaktera pre-

vode u nizove bajtova se dodadno definišu. ISO definie UCS-2 standard koji jednostavno svaki UNICODE karakter prevodi u odgovarajuća dva bajta.

Latinični tekstovi kodirani preko UCS-2 standarda sadrže veliki broj nula, koje obično u operativnim sistemima poput UNIX-a i u programskom jeziku C imaju specijalno značenje. Iz istog razloga softver koji je razvijen za rad sa dokumentima u ASCII formatu ne može da radi bez izmena nad dokumentima kodiranim preko UCS-2 standarda.

Unicode transformation format (UTF-8) je algoritam koji svakom dvobajtnom UNICODE karakteru dodeljuje određeni niz bajtova čija dužina varira od 1 do najviše 3. UTF je ASCII kompatibilan, što znači da se ASCII karakteri zapisuju pomoću jednog bajta, na standardni način. Konverzija se vrši na osnovu sledećih pravila:

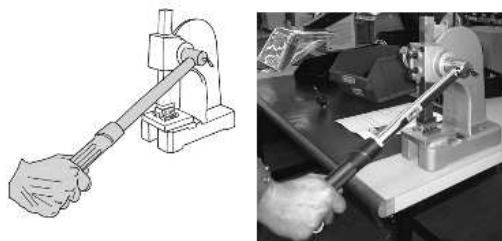
raspon	binarno zapisan UNICODE kôd	binarno zapisan UTF-8 kôd
0000-007F	00000000 0xxxxxxx	0xxxxxxx
0080-07FF	00000yyy yyxxxxxx	110yyyyy 10xxxxxx
0800-FFFF	zzzzyyyy yyxxxxxx	1110zzzz 10yyyyyy 10xxxxxx

Npr. karakter A koji se nalazi u ASCII tabeli ima UNICODE kôd $(0041)_{16} = (0000\ 0000\ 0100\ 0001)_2$, tako da se na osnovu prvog reda prethodne tabele zapisuje kao $(01000001)_2 = (41)_{16}$ u UTF-8 kodiranju. Karakter Š ima UNICODE kôd $(0160)_{16} = (0000\ 0001\ 0110\ 0000)_2$. Na njega se primenjuje drugi red prethodne tablice i dobija se da je njegov UTF-8 zapis $(1100\ 0101\ 1010\ 0000)_2 = (C5A0)_{16}$.

4 Zapis slika

Slike se u računaru zapisuju koristeći *vektorski zapis*, *rasterski zapis* ili *kombinovani zapis*.

Vektorski zapis slika podrazumeva da se slika sastoji od konačnog broja geometrijskih oblika (tačaka, linija, krivih, poligona), pri čemu se svaki oblik predstavlja svojim koordinatama. Slike koje računari generišu (CGI - Computer Generated Imagery) obično koristi vektorsku grafiku. Vektorski zapisane slike često zauzimaju manje prostora, dozvoljavaju uvećavanje (engl. zooming) bez gubitaka na kvalitetu prikaza i mogu se lakše preuređivati, obzirom da se objekti mogu nezavisno jedan od drugoga pomerati, menjati, dodavati i uklanjati.



Slika 7: Odnos vektorske i rasterske grafike

Rasterski zapis slika podrazumeva da je slika predstavljena pravougaonom

matricom sitnih delova koji se nazivaju pikseli (engl. pixel - PICture ELeMent). Svaki piksel je opisan isključivo oznakom njegove boje. Raster nastaje kao rezultat digitalizacije slike. Rasterska grafika se još naziva i bitmapirana grafika. Uređaji za prikaz (monitori, projektori), kao i uređaji za digitalno snimanje slika (fotaparati, skeneri) koriste rasterski zapis.

Modeli boja Za predstavljanje crno-belih slika, dovoljno je boju predstaviti isključivo količinom svetlosti. Različite količine svetlosti se diskretizuju u konačan broj nivoa nivoa osvetljenja i se dobija time odgovarajući broj nijansi sive boje. Ovakav model se naziva *Grayscale*. Ukoliko se za zapis informacije o količini svetlosti koristi 1 bajt, ukupan broj nijansi sive boje je 256.

U slučaju da se slika predstavlja isključivo sa dve boje (npr. skenirani tekst nekog dokumenta) koristi se model pod nazivom *Duotone*. Boja se tada predstavlja sa jednim bitom.

Mešanjem crvene (R), zelene (G) i plave (B) svetlosti se dobijaju sve ostale boje. Tako se npr. mešanjem crvene i zelene svetlosti dobija žuta svetlost. Bela svetlost se dobija mešanjem sve tri osnovne komponente, dok crna boja predstavlja odsustvo svetlosti. Imajući ovo u vidu, informacija o boji se dobija beleženjem informacije o količini crvene, plave i zelene svetlosti. Ovaj model se naziva *RGB model (aditivni model)*. RGB model boja se koristi kod uređaja koji boje prikazuju mešanjem svetlosti (monitori, projektori, ...). Ukoliko se za informaciju o svakoj komponenti pojedinačno koristi 1 bajt, ukupan broj bajtova za zapis informacije o boji je 3 što daje $2^{24} = 16M = 16777216$ različitih boja. Ovaj model se često naziva *TrueColor* model boja.

Nasuprot ovome, prilikom štampanja, koristi se *CMYK model (subtraktivni model)* boja. Osnovne komponente su ticitjan (C), magenta (M) i žuta (Y). Boje se nanose na papir i upijaju određenu komponentnu svetlosti. Tako ticitjan boja upija crvenu svetlost (a reflektuje zelenu i plavu), magenta upija zelenu (a reflektuje crvenu i plavu), a žuta upija plavu svetlost (a reflektuje crvenu i zelenu). Pošto se podrazumeva da je papir na koji se štampa nepigmentisan (beo), bela boja se dobija tako što se ne nanese ni jedna pigmentna komponenta. Crna boja bi trebalo da se dobije nanošenjem sve tri pigmentne komponente, međutim, vrlo je teško hemijski realizovati tako kvalitetne pigmente koji bi dali apsolutno crnu boju. Zbog toga se koristi i posebna crna (K) pigmenta komponenta.

Za obradu slike, pogodan je *HSB model* boja. Svaka boja se predstavlja preko Hue (H) komponente koja predstavlja ton boje na osnovu kojega i imenujemo boje, zatim Saturation (S) komponente koja predstavlja zasićenost boje odnosno njenu „jarkost” i Brightness (B) koja predstavlja osvetljenost boje. HSB je jedan od nekoliko različitih modela koji boju predstavlja razdvajanjem obojenosti i osvetljenosti (engl. chrominance/luminance). Tako npr. televizori u boji obično imaju kontrolu color kojom se određuje zasićenost boja i kontrolu brightness kojom se određuje osvetljenost boja. Ton boja je fiksiran i njega obično nije moguće prilagođavati na televizorima.

Formati zapisa rasterskih slika Rasterske slike su predstavljene matricom piksela, pri čemu se za svaki piksel čuva informacija o boji. Dimenzije ove matrice predstavljaju tzv. *apsolutnu rezoluciju* slike. Apsolutna rezolucija i model boja koji se koristi određuju broj bajtova pomoću kojih je moguće sliku

predstaviti. Tako npr. ukoliko je apsolutna rezolucija slike 800x600 piksela, pri čemu se koristi RGB model boje sa 3 bajta po pikselu, potrebno je ukupno $800 \cdot 600 \cdot 3B = 1440000B \approx 1.373MB$ za predstavljanje slike.

Prilikom pripreme slike za štampu važan pojam je *relativna rezolucija* koja se izražava u dpi (dots per inch) i predstavlja broj tačkica po jednom dužnom inču (2.54 cm). Relativna rezolucija direktno određuje dimenzije slike prilikom njenog štampanja.

Ukoliko se u okviru slike javlja samo manji broj različitih boja (pri čemu boje mogu da se odaberu proizvoljno, iz šireg modela boja), određene uštede se mogu postići korišćenjem *indeksiranog zapisa slike*. Recimo da na raspolaganju imamo svih 16M boja iz *TrueColor* modela, ali da naša slika koristi najviše 256, različitih boja ovog modela. U tom slučaju slika će zapisuje tako što se posebno zapiše indeksirana paleta boja koji svakom broju od 0 do 255 pridružuje 3 bajta koji opisuju boju u RGB modelu, dok se svakom pikselu slike pridružuje broj između 0 i 255 (za čije zapisivanje je dovoljan 1 bajt) koji predstavlja indeks boje tog piksela u okviru indeksirane palete boja.

Kako bi se smanjila količina informacija potrebnih za zapis slike, pribegava se tehnikama kompresije. Tehnike kompresije se mogu podeliti na:

1. kompresije bez gubitka (engl. lossless)
2. kompresije sa gubitkom (engl. lossy)

Neke od osnovnih tehnike kompresije bez gubitka su:

- *RLE (engl. Run Length Encoding)* - Uzastopni nizovi piksela iste boje se komprimuju zapisujući njihov broj uzastopnih pojavljivanja.

Npr. `wwwwwwwwwwwwBwwwwwwwwwwwwBwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwBwwwwwwwwwwww` se zapisuje kao `12W1B12W3B24W1B14W`.

Koristi se za zapis slika u PCX formatu, a moguće ga je koristiti i u okviru BMP, TGA, TIFF.

- *Entropijsko kodiranje* - Osnovna ideja entropijskih algoritama kodiranja je da umesto da svaki piksel kodiraju sa istim brojem bita, pikselima koji se češće javljaju dodeli kraće kodove, a da pikselima koji se ređe javljaju dopusti i duže kodove. Jedan od često korišćenih algoritama entropijskog kodiranja je Huffman-ovo kodiranje.
- *Dinamički izgrađeni rečnik* - Brzi algoritmi kompresije koji se zasnivaju na izgradnji rečnika koji dužim sekvencama dodeljuje kraće kodove, pri čemu se proces izgradnje rečnika vrši uporedo sa vršenjem kompresije kako bi se dobilo na efikasnosti algoritma. Najkorišćeniji algoritam ovog tipa je Lempel-Ziv-Welch (LZW) algoritam. Koristi se u okviru GIF i TIFF formata.
- *DEFLATE* - često korišćeni algoritam, zbog činjenice da nije zaštićen patentima i slobodan je za upotrebu. Algoritam predstavlja kombinaciju Lempel-Ziv algoritma i Huffman-ovog kodiranja. Koristi se u okviru PNG i TIFF formata.

Neke od osnovnih tehnika zapisa sa gubitkom su:

- *Smanjenje broja boja* - Slične nijanse boje se izjednačavaju kako bi se prešlo na indeksirani zapis.
- *Chroma subsampling* - Pošto je ljudsko oko osetljivije na razlike u nivou osvetljenosti nego u nivou obojenosti, slika se komprimuje tako što se podaci o obojenosti zapisuju sa manjim brojem bita.
- *Transformacijsko kodiranje* - slika se iz prostornog prebaci u frekvencijski domen korišćenjem najčešće Diskretne kosinusne transformacije (DCT) ili eventualno naprednije Wavelet transformacije. Oba ova metoda predstavljaju određenu vrsta analogona Fourier-ove transformacije. Zatim se uklone one frekvencije na koje ljudsko oko nije osetljivo. Ova tehnika leži u osnovu JPEG algoritma.
- *Fraktalno kodiranje* - kompresija se zasniva korišćenjem fraktala i fraktalne dimenzije.

Osnovna greška koju ljudi prave prilikom baratanja sa slikama je pogrešan izbor formata u kome se slika snima. BMP format zapisa slike najčešće podrazumeva da je slika nekomprimovana i samim tim nepotrebno zauzima previše prostora i nije pogodna za prenos preko mreže. GIF i PNG formati su naročito pogodni za kompresiju slika koje sadrže dijagrame, grafikone logotipove i slične veštačke oblike koji sadrže velike površine iste boje. JPEG (Joint Photographic Experts Group) kako samo ime kaže je namenjen isključivo za kompresiju digitalnih fotografija. Prilikom kompresije, slika se deli na grupe piksela (najčešće veličine 8x8 piksela) i kompresija se primenjuje posebno na svaku grupu piksela. Ukoliko je stepen kompresije jako veliki, ove grupe postaju okom vidljive i slika značajno gubi na kvalitetu. Ove pojave se nazivaju *artifakti*.



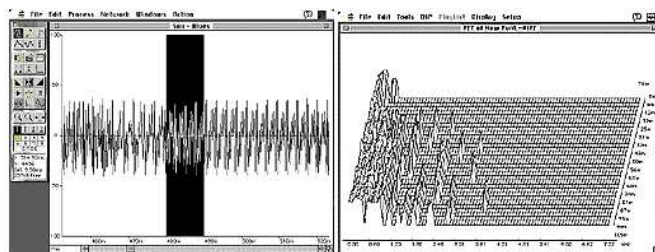
Slika 8: Artifakti prilikom JPEG kompresije

U nekim slučajevima je bitno da fotografije budu sačuvane tako da ne dođe do gubitka kvaliteta (npr. medicinski snimci, digitalizovani arhivski sadržaji i slično). U tom slučaju se najčešće primenjuje TIFF format.

Ukoliko se koristi neki od metoda kompresije sa gubitkom (npr. JPEG), prilikom svakog novog snimanja slike, dodatno se gubi na kvalitetu. Zbog ovoga se preporučuje da se svi međurezultati obrade slike snime u nekom od formata koji ne uključuje gubitak kvaliteta, a da se kompresija sa gubitkom primeni tek kao poslednji korak, kada je obrada završena.

5 Zapis zvuka

Osnovni parametri koji opisuju zvučni signal su njegova amplituda (koja odgovara „glasnoći“) i frekvencija (koja odgovara „visini“). Pošto ljudsko uho čuje raspon frekvencija od nekih 20Hz do 20KHz (mada je ovo individualno), na osnovu Nyquist-Shannon-ove teoreme, dovoljno je izvršiti odabiranje oko 40 000 puta u sekundi. Npr. AudioCD standard koji se koristi prilikom snimanja običnih audio CD-ova, propisuje frekvenciju odabiranja (engl. *sampling rate*) 44.1KHz. Pored ovoga, za postizanje još većeg kvaliteta neki standardi (miniDV, DVD, digital TV) propisuju odabiranje na frekvenciji 48KHz. Ukoliko se snima samo ljudski govor (npr. u mobilnoj telefoniji) frekvencije odabiranja mogu biti i znatno manje. Drugi važan parametar digitalizacije je broj bita sa kojim se zapisuje svaki pojedinačni odabirak (engl. *sampling resolution*). Najčešće se koristi 2 bajta po odbirku (16 bita) čime se dobija mogućnost zapisa 64K=65536 različitih nivoa amplitude.



Slika 9: Slika zvučnog signala. Levo: u fiksiranom vremenskom trenutku, Desno: kroz vreme

Kako bi se dobio prostorni osećaj zvuka, primenjuje se tehnika višekanalnog snimanja zvuka. U ovom slučaju, svaki kanal se nezavisno snima sa posebnim mikrofonom i reprodukuje na posebnom zvučniku. *Stereo* zvuk podrazumeva snimanje zvuka sa dva kanala. *Surround* sistemi podrazumevaju snimanje sa više od dva kanala (od 3 pa čak i do 10) pri čemu se često jedan poseban kanal izdvoji za specijalno snimanje niskofrekvencijskih komponenti zvuka (tzv. bas).

Formati zapisa zvuka Kao i slika, nekomprimovan zvuk zauzima puno prostora. Npr. jedan minut stereo zvuka snimljenog u AudioCD formatu zauzima $2 \cdot 44100 \frac{\text{sample}}{\text{sec}} \cdot 60 \text{sec} \cdot 2 \frac{B}{\text{sample}} = 10584000B \approx 10.1MB$.

Nekomprimovani zvuk se naziva nekada i PCM (Pulse Code Modulation) i snima se u WAV (Wave) kontejnerski format datoteke.

Kao i u slučaju slike, osnovni metodi kompresije se dele na

1. kompresije bez gubitka (engl. lossless)
2. kompresije sa gubitkom (engl. lossy)

Algoritam koji vrši kompresiju i dekompresiju zvučnog zapisa se naziva kodek (engl. codec).

Za razliku od kompresije slike, gde su formati kompresije bez gubitka imale značajnu ulogu, kod kompresije zvuka se one ređe primenjuju. Najznačajniji predstavnik danas je FLAC format.

Od formata zapisa sa gubitkom, najpopularniji format je svakako MP3 (MPEG-1 Audio-Layer 3). MP3 kompresija se zasniva na tzv. psiho-akustičkim efektima. Psiho akustika proučava koje komponente zvuka je moguće ukloniti iz zvučnog signala, a da ljudsko uho ne oseti promenu. Najznačajniji psihoakustički efekat je efekat maskiranja koji dovodi do toga da se u prisustvu glasnog zvuka određene frekvencije zvukovi koji su tiši, a imaju blisku frekvenciju glasnom zvuku ne mogu jasno raspoznati. Takođe uhu je potrebno određeno vreme da se adaptira na promenu intenziteta zvuka, tako da se nakon prestanka glasnog zvuka neki kratak vremenski period tihi zvukovi ne mogu razaznati. Ove nečujne komponente onda bivaju uklonjene, odnosno kodirane sa manje bitova što dovodi do kompresije. Matematički aparat koji leži u pozadini MP3 algoritma se opet svodi na prevođenje u frekvencijski domen primenom modifikovane diskretne kosinusne transformacije i na naknadnu analizu signala u frekvencijskom domenu. MPEG-1 standard, čiji je deo MP3 ne definiše precizno algoritam kodiranja, već samo format zapisa datoteke i algoritam dekodiranja. Ovo je dovelo do toga da je zbog boljeg razumevanja psihoakustičkih efekata vremenom nastajao niz sve boljih i boljih MP3 kodača. Kvalitet kompresije je određen vrednošću tzv. *bitrate*-a. Naime, za razliku od kompresije slike gde se obično ne zadaje precizno željena veličina komprimovane datoteke već se zadaje samo neki parametar koji određuje kvalitet kompresije, kod kompresije zvuka je neophodno da svaka sekunda muzike zauzima potpuno jednak broj bitova kako bi dekodiranje i puštanje zvuka bilo jednostavnije. Bitrate se izražava u kbit/s. Najčešće vrednosti za MP3 kompresiju su 128 kbit/s i 192kbit/s.

Sledeća tablica rezimira neke od najčešće korišćenih formata zapisa zvuka:

wav	Kontejnerski format. U najvećem broj slučajeva sadrži nekomprimovani zapis zvuka.
flac	Format kompresije zvuka bez gubitaka.
mp3	MPEG 1 Audio Layer 3.
wma	Windows Media Audio - Microsoft-ov format zapisa komprimovanog zvuka.
ra	Real Audio - pogodan za streaming audio (npr. slušanje radija preko interneta).
aac	Advance Audio Codec - format zasnovan na MPEG 2 i MPEG 4 standardima - poboljšanje MP3 formata.
ogg	slobodan, open source kontejnerski format koji najčešće sadrži zvuk komprimovan Vorbis kodekom.
ac-3	surround zapis zvuka koji se obično koristi kod zapisa zvuka iz filmova u DVD zapisu.

6 Zapis filma

Filmovi se obično zapisuju tako što se posebno opiše zvuk, a posebno video zapis i sve se zajedno spakuje u neku od kontejnerskih datoteka. Osnovni parametri koji određuju kvalitet kompresije videa su rezolucija slike, i bitrate koji uključuje količinu informacije potrebnih za zapis jedne sekunde i video i audio materijala. MPEG (Moving Picture Experts Group) je radna grupa ISO koja se bavi standardizacijom kodiranja video i audio zapisa.

Za razliku od zvuka i slika, zapis filmova bez kompresije je nezamisliv. Već same digitalne kamere vrše kompresiju tzv. DV (Digital video) format koji ima

fiksirani bitrate od 25 Mbits/sec.

Sledeća tablica rezimira neke od najčešće korišćenih formata zapisa videa:

MPEG-1, part 2	Koristi se za video CD-ove (VCD).
MPEG-2, part 2	Koristi se za DVD, SVCD i digitalnu televiziju.
MPEG-4, part 2	Koristi se za internet, i skladištenje velikih količina video materijala jer definiše načine kompresije.
DivX, XviD	Različite implementacije MPEG-4, part 2 standarda
WMV	Windows Media Video - Microsoft-ov format zapisa videa
RealVideo	Format prevasnodno namenjen za video streaming. Polako nestaje iz upotrebe.

Česta zabuna nastaje zbog činjenice da se neki kontejnerski formati datoteke (npr. AVI, MOV, VOB, ...) mešaju sa algoritmima kompresije.