

BAB II

TINJAUAN PUSTAKA

II.1. Definisi Citra

Citra adalah suatu *representasi* (gambaran), kemiripan, atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat *optik* berupa foto, bersifat analog berupa sinyal-sinyal *video* seperti gambar pada layar televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan.

Citra analog adalah citra yang bersifat kontinu, citra analog tidak dapat direprestasikan dalam komputer sehingga tidak bisa diproses di komputer secara langsung. Oleh sebab itu, agar citra dapat diproses di komputer, proses konversi analog ke digital harus dilakukan terlebih dahulu.

Citra digital adalah citra yang dapat diolah oleh komputer. *Monitor* akan menampilkan sebuah kotak kecil (piksel). Namun, yang disimpan dalam komputer hanyalah angka-angka yang menunjukkan besar intensitas pada masing-masing piksel tersebut (T. Sutoyo, dkk; 2009: 9).

Proses mengubah citra analog menjadi citra digital disebut digitalisasi citra. Ada dua hal yang harus dilakukan pada digitalisasi citra, yaitu digitalisasi spasial yang disebut juga sebagai *sampling* (penerokan) dan digitalisasi intensitas yang sering disebut sebagai kuantisasi.

Kita tahu bahwa citra yang dihasilkan dari peralatan digital (citra digital) langsung bisa diproses oleh komputer. Mengapa peralatan digital bisa

menghasilkan citra digital?. Sebenarnya di dalam peralatan digital sudah terdapat sistem sampling dan kuantitas. Sedangkan analog tidak dilengkapi kedua sistem tersebut. Kedua sistem inilah yang bertugas memotong-motong citra menjadi x kolom dan y baris (*process sample*), sekaligus menentukan besar intensitasnya yang terdapat pada titik tersebut (proses kuantisasi) sehingga menghasilkan resolusi citra yang diinginkan (T. Sutoyo, dkk; 2009: 12-13).

Teknologi dasar untuk menciptakan dan menampilkan warna pada citra digital berdasarkan pada penelitian bahwa sebuah warna merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau, dan biru (*Red, Green, Blue* - RGB).

Di dalam komputer, citra digital disimpan sebagai suatu *file* dengan format tertentu. Format citra tersebut menunjukkan cara sebuah citra digital disimpan, misalnya apakah dengan suatu kompresi atau tidak. Contoh format citra digital adalah .bmp, .jpg, .png, .tif dan sebagainya (<http://digilib.tes.telkomuniversity.ac.id/>).

II.2. Steganografi

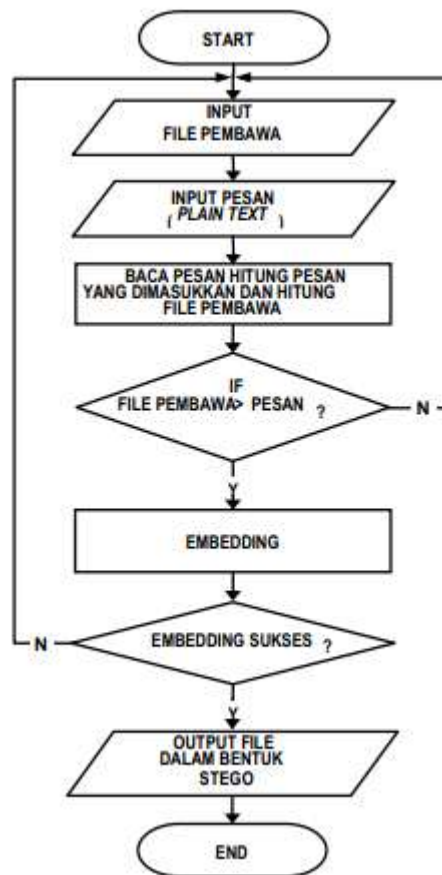
Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sehingga orang lain tidak menyadari ada sesuatu pesan di dalam media tersebut. Kata steganografi (*steganography*) berasal dari bahasa Yunani *steganos* yang artinya “tersembunyi/terselubung” dan *graphein* “menulis” sehingga kurang lebih artinya “menulis (tulisan) terselubung” (T. Sutoyo, dkk; 2009: 244).

Bangsa Romawi mengenal steganografi dengan menggunakan tinta tak tampak (*invisible ink*) untuk menuliskan pesan. Tinta tersebut dibuat dari campuran sari buah, susu, dan cuka. Jika tinta digunakan untuk menulis maka tulisannya tidak tampak. Tulisan di atas kertas dapat dibaca dengan cara memanaskan kertas tersebut Jurnal SAINTIKOM (Sulindawaty; 2011: 156).

Dalam bidang keamanan komputer, steganografi digunakan untuk menyembunyikan data rahasia, saat *enkripsi* tidak dapat dilakukan atau bersamaan dengan *enkripsi*. Walaupun *enkripsi* berhasil dipecahkan (*decipher*), pesan atau data rahasia tetap tidak terlihat. Pada *cryptography*, pesan disembunyikan dengan “diacak” sehingga pada kasus-kasus tertentu dapat dengan mudah mengundang kecurigaan, sedangkan pada steganografi pesan “disamarkan” dalam bentuk yang relatif “aman” sehingga tidak terjadi kecurigaan itu.

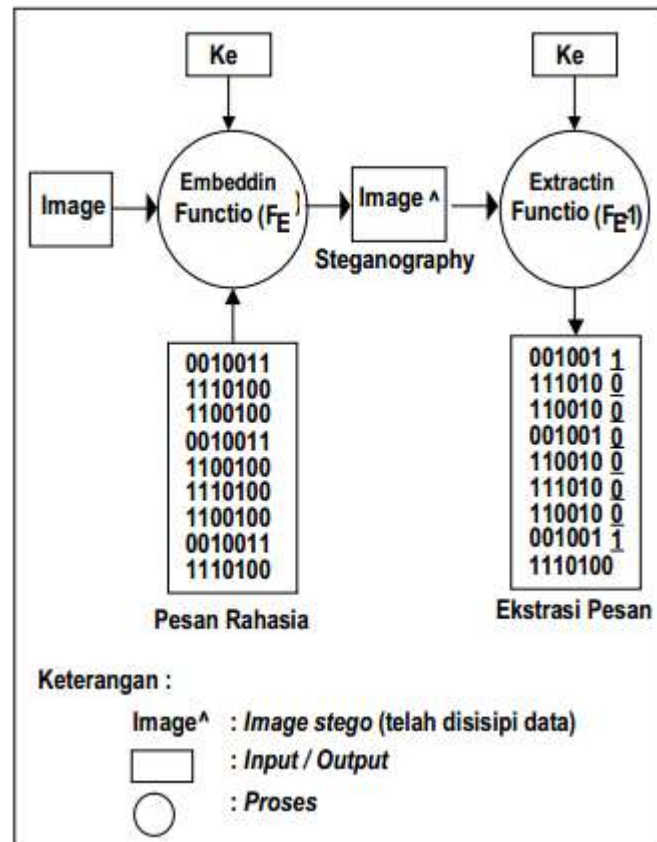
Steganografi membutuhkan dua properti, yaitu wadah penampung dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, *audio*, teks, dan *video*. Data rahasia yang disembunyikan juga dapat berupa citra, *audio*, teks, atau *video*. ” (T. Sutoyo, dkk ; 2009: 244).

Secara umum proses *Embedding* (proses penyisipan pesan) pada citra digital digambarkan dengan *Flowchart* pada gambar II.1. *Flowchart* pada gambar II.1. menunjukkan proses fungsi *embedding* pada teknik steganografi yang dalam hal ini dilakukan oleh bagian pengirim (*sender*). Dalam proses *embedding* pesan rahasia disisipkan ke file pembawa (citra digital), yang akan menghasilkan *output file* dalam bentuk *stego* (gambar steganografi).



Gambar II.1 *Flowchart Proses Embedding Function* secara umum
 Sumber : Sulindawaty; 2011: 159

Setelah proses *embedding* dilakukan maka di bagian penerima pesan (*recipient*), dilakukan proses *extracting* pada *stego* untuk memisahkan pesan rahasia (*emb*) dan data penyimpan (*cover*). Untuk dapat melihat gambaran proses *embedding* dan *extracting* dapat dilihat pada gambar II.2 dibawah ini :



Gambar II.2 Diagram Proses *Embedding* dan *Extracting*
 Sumber : Sulindawaty; 2011: 160

Gambar II.2 merupakan diagram proses *embedding function* dimana setelah proses ekstraksi dilakukan dengan mengekstrak *bit* terkecil dari masing-masing piksel pada *stego* secara berurutan. Terlihat adanya perubahan *bit-bit* pada setiap piksel. Perubahan *bit-bit* terkecil yang terjadi hanya 1 *bit* saja dari setiap piksel, dimana perubahan yang terjadi terlalu kecil untuk dapat terdeteksi oleh mata manusia, sehingga pesan dapat disembunyikan secara efektif

Jurnal SAINTIKOM (Sulindawaty; 2011: 159-160).

Salah satu algoritma steganografi yang paling populer dan sering digunakan untuk menyembunyikan informasi dalam citra digital metode penyisipan *Least Significant Bit* (LSB). LSB adalah algoritma sederhana yang menukar *bit* yang paling kecil ke dalam beberapa *byte* media penyembunyiannya secara berurutan SNIKOM (Yudhi Andrian; 2013: 274).

II.2.1 Metode *Least Significant Bit* (LSB)

Pendekatan paling sederhana untuk menyembunyikan data dalam file citra disebut penyisipan *Least Significant Bit* (LSB). Penyisipan *Least significant bit* (LSB) adalah pendekatan yang umum untuk menanamkan informasi dalam media citra. *Least significant bit* (dengan kata lain, *bit* ke-8) sebagian atau seluruh dari *byte* dalam sebuah gambar diubah menjadi sebuah *bit* dari pesan rahasia SNIKOM (Yudhi Andrian; 2013: 275).

Metode ini menggunakan citra digital sebagai *covertext*. Pada susunan *bit* di dalam sebuah *byte* (1 *byte*= 8 *bit*), ada *bit* yang paling berarti (*Most Significant Bit* atau MSB) dan *bit* yang paling kurang berarti (*Least Significant Bit* atau LSB). Sebagai contoh *byte* 11010010, angka *bit* 1 (pertama, digaris-bawahi) adalah *bit* MSB, dan angka *bit* 0 (terakhir, digaris-bawahi) adalah *bit* LSB. *Bit* yang cocok untuk diganti adalah *bit* LSB, sebab perubahan tersebut hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya.

Misalkan segmen piksel-piksel citra/gambar sebelum penambahan *bit-bit* adalah:

00110011 10100010 11100010 10101011 00100110

10010110 11001001 11111001 10001000 10100011

Pesan rahasia (yang telah dikonversi ke sistem *biner*) misalkan '1110010111', maka setiap *bit* dari pesan tersebut menggantikan posisi LSB dari segmen piksel-piksel citra menjadi:

00110011 10100011 11100011 10101010 00100110

10010111 11001000 11111001 10001001 10100011

Misalkan *byte* tersebut menyatakan warna merah, maka perubahan satu *bit* LSB tidak mengubah warna merah tersebut secara berarti. Mata manusia tidak dapat membedakan perubahan kecil tersebut. Jurnal Dinamika Informatika (Basuki Rakhmat, dkk; 2010: 6).

Bila menggunakan gambar 24-*bit*, *bit* dari masing-masing komponen warna merah, hijau dan biru dapat digunakan, karena masing-masing ditampilkan dalam bentuk *byte*. Dengan kata lain, seseorang dapat menyimpan 3 *bit* di setiap pixel. Citra dengan piksel 800×600 , dapat menyimpan total Jumlah 1,440,000 *bit* atau 180.000 *byte* data yang disisipkan SNIKOM (Yudhi Andrian; 2013: 275).

II.3 Kriptografi

Kriptografi berasal dari bahasa Yunani, menurut bahasa dibagi menjadi dua *kripto* dan *graphia*, *kripto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ketempat yang lain.

Kriptografi mempunyai sejarah yang sangat menarik dan panjang. Kriptografi sudah digunakan 4000 tahun yang lalu yang diperkenalkan oleh orang-orang Mesir untuk mengirim pesan ke pasukan militer yang berada di lapangan dan supaya pesan tersebut tidak terbaca oleh pihak musuh walaupun kurir pembawa pesan tersebut tertangkap oleh musuh.

Algoritma kriptografi dibagi menjadi dua :

1. Kriptografi Klasik

Kriptografi klasik merupakan suatu algoritma yang menggunakan satu kunci untuk mengamankan data, teknik ini sudah digunakan beberapa abad yang lalu. Dua teknik dasar yang biasa digunakan pada algoritma jenis ini, diantaranya adalah :

a. Teknik Substitusi:

Teknik substitusi merupakan penggantian setiap karakter dari plaintext dengan karakter lainnya. Substitusi memiliki beberapa algoritma diantaranya : *Caesar cipher*, *Playfair cipher*, *Shift cipher*, *Hill cipher*, dan *Vigenere cipher*.

b. Teknik Transposisi

Teknik transposisi merupakan suatu teknik yang menggunakan permutasi karakter, yang mana dengan menggunakan teknik ini pesan yang asli tidak dapat dibaca kecuali memiliki kunci untuk mengembalikan pesan tersebut ke bentuk semula atau disebut dengan *deskripsi*. Transposisi memiliki beberapa algoritma diantaranya :

Rahasia yang sempurna, *One Time Pad* (OTP), dan Rotor mesin (*Dony Ariyus; 2006: 9*).

2. Kriptografi Modern

Kriptografi modern merupakan suatu algoritma yang digunakan pada saat sekarang ini, yang mana kriptografi modern mempunyai kerumitan yang sangat kompleks, karena dalam pengoperasinya menggunakan komputer. Secara umum berdasarkan kesamaan kuncinya, algoritma sandi dibedakan menjadi:

a. Kunci Simetris

Skema algoritma sandi akan disebut kunci-simetris apabila untuk setiap proses *enkripsi* maupun *dekripsi* data secara keseluruhan digunakan kunci yang sama. Skema ini berdasarkan jumlah data per proses dan alur pengolahan data didalamnya dibedakan menjadi dua kelas, yaitu block-cipher dan stream-cipher. Didalam algoritma simetris terdapat beberapa contoh yang menggunakan kunci simetris seperti: *Data Encryption Standard* (DES), *Blowfish*, *Twofish*, *MARS*, *IDEA*, *3DES*, dan *Advanced Encryption Standard* (AES).

b. Kunci Asimetris

Skema ini adalah algoritma yang menggunakan kunci yang berbeda untuk proses *enkripsi* dan *dekripsi*. Skema ini disebut juga sebagai sistem kriptografi kunci publik karena

kunci untuk *enkripsi* dibuat untuk diketahui oleh umum (*public-key*) atau dapat diketahui siapa saja, tapi untuk proses dekripsinya hanya dapat dilakukan oleh yang berwenang yang memiliki kunci rahasia untuk mendekripsinya, disebut *private-key*. Dapat dianalogikan seperti kotak pos yang hanya dapat dibuka oleh tukang pos yang memiliki kunci tapi setiap orang dapat memasukkan surat ke dalam kotak tersebut. Keuntungan algoritma model ini, untuk berkorespondensi secara rahasia dengan banyak pihak tidak diperlukan kunci rahasia sebanyak jumlah pihak tersebut, cukup membuat dua buah kunci, yaitu kunci publik bagi para koresponden untuk mengenkripsi pesan, dan kunci privat untuk mendekripsi pesan. Berbeda dengan skema kunci-simetris, jumlah kunci yang dibuat adalah sebanyak jumlah pihak yang diajak berkorespondensi. Didalam algoritma asimetris terdapat beberapa contoh yang menggunakan kunci Asimetris seperti: *Knapsack*, *Rivest-Shamir-Adelman* (RSA), dan *Diffie-Hellman* (http://id.wikipedia.org/wiki/Kriptografi#Era_komputer).

Kriptografi pada awalnya dijabarkan sebagai ilmu yang mempelajari bagaimana menyembunyikan pesan. Namun pada pengertian modern kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas. Jadi

pengertian kriptografi modern adalah tidak saja berurusan hanya dengan penyembunyian pesan namun lebih pada sekumpulan teknik yang menyediakan keamanan informasi (Rifki Sadikin; 2012: 9).

Sistem kriptografi terdiri dari 5 bagian yaitu:

1. *Plaintext*: pesan atau data dalam bentuk aslinya yang dapat terbaca. *Plaintext* adalah masukan bagi algoritma *enkripsi*. Untuk selanjutnya digunakan istilah teks asli sebagai sinonim kata *Plaintext*.
2. *Secret key*: *secret key* yang juga merupakan masukan bagi algoritma *enkripsi* merupakan nilai yang bebas terhadap teks asli dan menentukan hasil keluaran algoritma *enkripsi*. Untuk selanjutnya digunakan istilah kunci rahasia sebagai sinonim kata *secret key*.
3. *Ciphertext*: *ciphertext* adalah keluaran algoritma *enkripsi*. *Ciphertext* dapat dianggap sebagai pesan dalam bentuk tersembunyi. Algoritma *enkripsi* yang baik akan menghasilkan *ciphertext* yang terlihat acak. Untuk selanjutnya digunakan istilah teks sandi sebagai sinonim kata *ciphertext*.
4. Algoritma *Enkripsi*: algoritma *enkripsi* memiliki 2 masukan, yaitu teks asli dan kunci rahasia. Algoritma *enkripsi* melakukan transformasi terhadap teks asli sehingga menghasilkan teks sandi.
5. Algoritma Deskripsi: algoritma deskripsi memiliki 2 masukan, yaitu teks sandi dan kunci rahasia. Algoritma deskripsi memulihkan kembali teks sandi menjadi teks asli bila kunci rahasia yang dipakai algoritma

deskripsi sama dengan kunci rahasia yang dipakai algoritma *enkripsi* (Rifki Sadikin; 2012: 15-16).

II.3.1 Algoritma *Hill Cipher*

Hill cipher termasuk dalam salah satu kriptosistem polialfabetik, artinya setiap karakter alfabet bisa dipetakan ke lebih dari satu macam karakter alfabet. Cipher tersebut ditemukan pada tahun 1929 oleh Lester S. Hill. Ide dari *Hill Cipher* adalah misalkan m adalah bilangan bulat positif. Dengan cara mengambil m kombinasi linier dari m karakter alfabet dalam satu elemen *plaintext*. Misalkan $m=2$, maka dapat ditulis suatu elemen *plaintext* sebagai $x = (x_1, x_2)$ dan suatu elemen ciphertext sebagai $y = (y_1, y_2)$. Disini (y_1, y_2) adalah kombinasi linier dari x_1 dan x_2 . Misalkan:

$$\begin{array}{l} Y_1 = 1x_1 + 5x_2 \\ Y_2 = 9x_1 + 8x_2 \end{array} \quad \dots\dots (1)$$

Sehingga dapat dituliskan kedalam bentuk matrik sebagai berikut:

$$\begin{array}{l} Y_1, Y_2 = X_1, X_2 = \begin{array}{cc} 1 & 5 \\ 9 & 8 \end{array} \end{array} \quad \dots\dots (2)$$

Secara umum, algoritma *Hill Cipher* akan menggunakan matriks K $m \times m$ sebagai kunci untuk mengacak pesannya. Jika elemen pada baris i dan kolom j dari matriks K_{ij} , maka dapat dituliskan sebagai berikut:

Dikatakan bahwa *ciphertext* diperoleh dari *plaintext* dengan cara transformasi linier. Untuk melakukan *dekripsi*, akan digunakan matrik invers K^{-1} . Jadi, *dekripsi* dilakukan jika matrik tersebut memiliki nilai invers dengan rumus:

1. Perkalian matrik memiliki sifat asosiatif, yaitu $(AB)C = A(BC)$.
2. Matriks invers dari A adalah A^{-1} dimana $AA^{-1} = A^{-1}A = I_m$.
3. Matrik identitas $m \times m$ yang ditulis dengan I_m , adalah matrik yang berisi 1 pada diagonal utama dan berisi 0 pada elemen lainnya.

II.4 Kode ASCII

ASCII singkatan dari *American Standard Code for Information Interchange*. Ini adalah kode karakter 7-bit di mana setiap bit tunggal mewakili karakter yang unik. Pada halaman web ini Anda akan menemukan 8 bit, 256 karakter, sesuai dengan ISO 8859-1 dan Microsoft® Windows Latin-1 meningkat karakter, yang tersedia dalam program tertentu seperti *Microsoft Word*.

Karakter kontrol ASCII (kode karakter 0-31). Yang pertama 32 karakter dalam ASCII-tabel adalah kode kontrol tak patut ditulis dan digunakan untuk mengendalikan peripheral seperti printer.

Tabel II.1 ASCII kode karakter (0-31)

DEC	BIN	Simbol	Deskripsi	DEC	BIN	Simbol	Deskripsi
0	00000000	NUL	<i>Null char</i>	16	00010000	DLE	<i>Data Line Escape</i>
1	00000001	SOH	<i>Start of Heading</i>	17	00010001	DC1	<i>Device Control 1 (oft. XON)</i>
2	00000010	STX	<i>Start of Text</i>	18	00010010	DC2	<i>Device Control 2</i>
3	00000011	ETX	<i>End of Text</i>	19	00010011	DC3	<i>Device Control 3 (oft. XOFF)</i>
4	00000100	EOT	<i>End of Transmission</i>	20	00010100	DC4	<i>Device Control 4</i>

5	00000101	ENQ	<i>Enquiry</i>	21	00010101	NAK	<i>Negative Acknowledgement</i>
6	00000110	ACK	<i>Acknowledgment</i>	22	00010110	SYN	<i>Synchronous Idle</i>
7	00000111	BEL	<i>Bell</i>	23	00010111	ETB	<i>End of Transmit Block</i>
8	00001000	BS	<i>Back Space</i>	24	00011000	CAN	<i>Cancel</i>
9	00001001	HT	<i>Horizontal Tab</i>	25	00011001	EM	<i>End of Medium</i>
10	00001010	LF	<i>Line Feed</i>	26	00011010	SUB	<i>Substitute</i>
11	00001011	VT	<i>Vertical Tab</i>	27	00011011	ESC	<i>Escape</i>
12	00001100	FF	<i>Form Feed</i>	28	00011100	FS	<i>File Separator</i>
13	00001101	CR	<i>Carriage Return</i>	29	00011101	GS	<i>Group Separator</i>
14	00001110	SO	<i>Shift Out / X-On</i>	30	00011110	RS	<i>Record Separator</i>
15	00001111	SI	<i>Shift In / X-Off</i>	31	00011111	US	<i>Unit Separator</i>

Sumber: <http://www.ascii-code.com/>

ASCII karakter yang dapat dicetak (kode karakter 32-127). Kode 32-127 biasa digunakan untuk semua variasi yang berbeda dari tabel ASCII, mereka disebut karakter yang dapat dicetak, mewakili huruf, angka, tanda baca, dan simbol-simbol lain. Anda akan menemukan hampir setiap karakter pada keyboard Anda. Karakter 127 merupakan perintah DEL.

Tabel II.2 ASCII kode karakter (31-127)

DEC	BIN	Simbol	Deskripsi	DEC	BIN	Simbol	Deskripsi
32	00100000		<i>Space</i>	80	01010000	P	<i>Uppercase P</i>
33	00100001	!	<i>Exclamation mark</i>	81	01010001	Q	<i>Uppercase Q</i>
34	00100010	"	<i>Double quotes (or speech marks)</i>	82	01010010	R	<i>Uppercase R</i>
35	00100011	#	<i>Number</i>	83	01010011	S	<i>Uppercase S</i>
36	00100100	\$	<i>Dollar</i>	84	01010100	T	<i>Uppercase T</i>
37	00100101	%	<i>Procenttecken</i>	85	01010101	U	<i>Uppercase U</i>
38	00100110	&	<i>Ampersand</i>	86	01010110	V	<i>Uppercase V</i>
39	00100111	'	<i>Single quote</i>	87	01010111	W	<i>Uppercase W</i>
40	00101000	(	<i>Open parenthesis (or open bracket)</i>	88	01011000	X	<i>Uppercase X</i>
41	00101001	)	<i>Close parenthesis (or close bracket)</i>	89	01011001	Y	<i>Uppercase Y</i>
42	00101010	*	<i>Asterisk</i>	90	01011010	Z	<i>Uppercase Z</i>
43	00101011	+	<i>Plus</i>	91	01011011	[	<i>Opening bracket</i>

44	00101100	,	Comma	92	01011100	\	Backslash
45	00101101	-	Hyphen	93	01011101	]	Closing bracket
46	00101110	.	Period, dot or full stop	94	01011110	^	Caret - circumflex
47	00101111	/	Slash or divide	95	01011111	_	Underscore
48	00110000	0	Zero	96	01100000	`	Grave accent
49	00110001	1	One	97	01100001	a	Lowercase a
50	00110010	2	Two	98	01100010	b	Lowercase b
51	00110011	3	Three	99	01100011	c	Lowercase c
52	00110100	4	Four	100	01100100	d	Lowercase d
53	00110101	5	Five	101	01100101	e	Lowercase e
54	00110110	6	Six	102	01100110	f	Lowercase f
55	00110111	7	Seven	103	01100111	g	Lowercase g
56	00111000	8	Eight	104	01101000	h	Lowercase h
57	00111001	9	Nine	105	01101001	i	Lowercase i
58	00111010	:	Colon	106	01101010	j	Lowercase j
59	00111011	;	Semicolon	107	01101011	k	Lowercase k
60	00111100	<	Less than (or open angled bracket)	108	01101100	l	Lowercase l
61	00111101	=	Equals	109	01101101	m	Lowercase m
62	00111110	>	Greater than (or close angled bracket)	110	01101110	n	Lowercase n
63	00111111	?	Question mark	111	01101111	o	Lowercase o
64	01000000	@	At symbol	112	01110000	p	Lowercase p
65	01000001	A	Uppercase A	113	01110001	q	Lowercase q
66	01000010	B	Uppercase B	114	01110010	r	Lowercase r
67	01000011	C	Uppercase C	115	01110011	s	Lowercase s
68	01000100	D	Uppercase D	116	01110100	t	Lowercase t
69	01000101	E	Uppercase E	117	01110101	u	Lowercase u
70	01000110	F	Uppercase F	118	01110110	v	Lowercase v
71	01000111	G	Uppercase G	119	01110111	w	Lowercase w
72	01001000	H	Uppercase H	120	01111000	x	Lowercase x
73	01001001	I	Uppercase I	121	01111001	y	Lowercase y
74	01001010	J	Uppercase J	122	01111010	z	Lowercase z
75	01001011	K	Uppercase K	123	01111011	{	Opening brace
76	01001100	L	Uppercase L	124	01111100		Vertical bar
77	01001101	M	Uppercase M	125	01111101	}	Closing brace
78	01001110	N	Uppercase N	126	01111110	~	Equivalency sign - tilde
79	01001111	O	Uppercase O	127	01111111		Delete

Sumber: <http://www.ascii-code.com/>

Kode ASCII diperpanjang (kode karakter 128-255). Ada beberapa variasi yang berbeda dari tabel ASCII 8-bit. Tabel di bawah ini sesuai dengan ISO 8859-1, juga disebut ISO Latin-1. Codes 129-159 berisi *Microsoft® Windows Latin-1* karakter diperpanjang. Terjemahan (<http://www.ascii-code.com/>)

Tabel II.3 ASCII kode karakter (128-255)

DEC	BIN	Simbol	Deskripsi	DEC	BIN	Simbol	Deskripsi
128	10000000	€	Euro sign	192	11000000	À	Latin capital letter A with grave
129	10000001			193	11000001	Á	Latin capital letter A with acute
130	10000010	,	Single low-9 quotation mark	194	11000010	Â	Latin capital letter A with circumflex
131	10000011	ƒ	Latin small letter f with hook	195	11000011	Ã	Latin capital letter A with tilde
132	10000100	„	Double low-9 quotation mark	196	11000100	Ä	Latin capital letter A with diaeresis
133	10000101	...	Horizontal ellipsis	197	11000101	Å	Latin capital letter A with ring above
134	10000110	†	Dagger	198	11000110	Æ	Latin capital letter AE
135	10000111	‡	Double dagger	199	11000111	Ç	Latin capital letter C with cedilla
136	10001000	^	Modifier letter circumflex accent	200	11001000	È	Latin capital letter E with grave
137	10001001	‰	Per mille sign	201	11001001	É	Latin capital letter E with acute
138	10001010	Š	Latin capital letter S with caron	202	11001010	Ê	Latin capital letter E with circumflex
139	10001011	‹	Single left-pointing angle quotation	203	11001011	Ë	Latin capital letter E with diaeresis
140	10001100	Œ	Latin capital ligature OE	204	11001100	Ì	Latin capital letter I with grave
141	10001101			205	11001101	Í	Latin capital letter I with acute
142	10001110	Ž	Latin capital letter Z with caron	206	11001110	Î	Latin capital letter I with circumflex
143	10001111			207	11001111	Ï	Latin capital letter I with diaeresis
144	10010000			208	11010000	Ð	Latin capital letter ETH
145	10010001	‘	Left single quotation mark	209	11010001	Ñ	Latin capital letter N with tilde

146	10010010	´	Right single quotation mark	210	11010010	Ò	Latin capital letter O with grave
147	10010011	“	Left double quotation mark	211	11010011	Ó	Latin capital letter O with acute
148	10010100	”	Right double quotation mark	212	11010100	Ô	Latin capital letter O with circumflex
149	10010101	•	Bullet	213	11010101	Õ	Latin capital letter O with tilde
150	10010110	–	En dash	214	11010110	Ö	Latin capital letter O with diaeresis
151	10010111	—	Em dash	215	11010111	×	Multiplication sign
152	10011000	˘	Small tilde	216	11011000	Ø	Latin capital letter O with slash
153	10011001	™	Trade mark sign	217	11011001	Û	Latin capital letter U with grave
154	10011010	Š	Latin small letter S with caron	218	11011010	Ú	Latin capital letter U with acute
155	10011011	›	Single right-pointing angle quotation mark	219	11011011	Û	Latin capital letter U with circumflex
156	10011100	Œ	Latin small ligature oe	220	11011100	Ü	Latin capital letter U with diaeresis
157	10011101			221	11011101	Ý	Latin capital letter Y with acute
158	10011110	Ž	Latin small letter z with caron	222	11011110	Þ	Latin capital letter THORN
159	10011111	Ÿ	Latin capital letter Y with diaeresis	223	11011111	ß	Latin small letter sharp s - ess-zed
160	10100000		Non-breaking space	224	11100000	À	Latin small letter a with grave
161	10100001	¡	Inverted exclamation mark	225	11100001	Á	Latin small letter a with acute
162	10100010	¢	Cent sign	226	11100010	Â	Latin small letter a with circumflex
163	10100011	£	Pound sign	227	11100011	Ã	Latin small letter a with tilde
164	10100100	¤	Currency sign	228	11100100	Ä	Latin small letter a with diaeresis
165	10100101	¥	Yen sign	229	11100101	Å	Latin small letter a with ring above
166	10100110	¦	Pipe, Broken vertical bar	230	11100110	Æ	Latin small letter ae
167	10100111	§	Section sign	231	11100111	Ç	Latin small letter c with cedilla
168	10101000	¨	Spacing diaeresis – umlaut	232	11101000	È	Latin small letter e with grave
169	10101001	©	Copyright sign	233	11101001	É	Latin small letter e with acute
170	10101010	ª	Feminine ordinal	234	11101010	Ê	Latin small letter e with

			<i>indicator</i>				<i>circumflex</i>
171	10101011	«	<i>Left double angle quotes</i>	235	11101011	Ë	<i>Latin small letter e with diaeresis</i>
172	10101100	¬	<i>Not sign</i>	236	11101100	ì	<i>Latin small letter i with grave</i>
173	10101101		<i>Soft hyphen</i>	237	11101101	í	<i>Latin small letter i with acute</i>
174	10101110	®	<i>Registered trade mark sign</i>	238	11101110	î	<i>Latin small letter i with circumflex</i>
175	10101111	–	<i>Spacing macron – overline</i>	239	11101111	ï	<i>Latin small letter i with diaeresis</i>
176	10110000	°	<i>Degree sign</i>	240	11110000	Ð	<i>Latin small letter eth</i>
177	10110001	±	<i>Plus-or-minus sign</i>	241	11110001	Ñ	<i>Latin small letter n with tilde</i>
178	10110010	²	<i>Superscript two – squared</i>	242	11110010	Ò	<i>Latin small letter o with grave</i>
179	10110011	³	<i>Superscript three – cubed</i>	243	11110011	Ó	<i>Latin small letter o with acute</i>
180	10110100	´	<i>Acute accent - spacing acute</i>	244	11110100	Ô	<i>Latin small letter o with circumflex</i>
181	10110101	µ	<i>Micro sign</i>	245	11110101	Õ	<i>Latin small letter o with tilde</i>
182	10110110	¶	<i>Pilcrow sign - paragraph sign</i>	246	11110110	Ö	<i>Latin small letter o with diaeresis</i>
183	10110111	·	<i>Middle dot - Georgian comma</i>	247	11110111	÷	<i>Division sign</i>
184	10111000	¸	<i>Spacing cedilla</i>	248	11111000	Ø	<i>Latin small letter o with slash</i>
185	10111001	¹	<i>Superscript one</i>	249	11111001	Ù	<i>Latin small letter u with grave</i>
186	10111010	º	<i>Masculine ordinal indicator</i>	250	11111010	Ú	<i>Latin small letter u with acute</i>
187	10111011	»	<i>Right double angle quotes</i>	251	11111011	Û	<i>Latin small letter u with circumflex</i>
188	10111100	¼	<i>Fraction one quarter</i>	252	11111100	Ü	<i>Latin small letter u with diaeresis</i>
189	10111101	½	<i>Fraction one half</i>	253	11111101	Ý	<i>Latin small letter y with acute</i>
190	10111110	¾	<i>Fraction three quarters</i>	254	11111110	Þ	<i>Latin small letter thorn</i>
191	10111111	¿	<i>Inverted question mark</i>	255	11111111	ÿ	<i>Latin small letter y with diaeresis</i>

Sumber: <http://www.ascii-code.com/>

II.5 Visual Studio

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi *console*, aplikasi *Windows*, ataupun aplikasi *Web*. Visual Studio mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi *Windows Mobile* (yang berjalan di atas .NET Compact Framework).

(http://id.wikipedia.org/wiki/Microsoft_Visual_Studio)