Remember, Record, Resist:

A Self-Contained QR-Code Archival System for Long-Term Information Preservation

a.k.a Lyra Phasma

Supervisor:

PLACEHOLDER: Supervisor

A thesis submitted in partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

PLACEHOLDER: University

College of Engineering and Information Technology

Department of Information Technology

Submission Date

2027

To my parents, who gave me life and strength;
to my beloved who brings stillness in my torrential storm;
and to those who have endured —
the fallen, the widowed, the forgotten,
whose stories deserve to be remembered and retold.
May this work stand as a small act of remembrance.

Table of Contents

List of Figures

List of Tables

Chapter 1
Mathematics

1.1 Set Theory

$\{ x : x \in \mathbb{R} \mid x > 0 \}$ is the set of all strictly positive real numbers.

S = letters of the alphabet.

$|S| = 26$

Power of Sets

$$S = \{a, b, c\}$$
$$\mathcal{P}(S) = \{\varnothing, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$$

Let $A = \{1, 2, 3\}$ and $B = \{3, 4, 5\}$:

Union: $A \cup B = \{1, 2, 3, 4, 5\}$
Intersection: $A \cap B = \{3\}$
Complement: $A \setminus B = \{1, 2\}$
Symmetric Difference: $A \triangle B = \{1, 2, 4, 5\}$

Union $A \cup B = \{1, 2, 3, 4, 5\}$

Intersection $A \cap B = \{3\}$

Complement $A \setminus B = \{1, 2\}$

Symmetric Difference $A \triangle B = \{1, 2, 4, 5\}$

Cartesian products

$$A \times B = \{(1, 3), (1, 4), (2, 3), (2, 4)\}$$

$$B \times A = \{(3, 1), (3, 2), (4, 1), (4, 2)\}$$

$$A \times B \neq B \times A$$

1.2 Discrete Mathematics

1.2.1 Key Concepts

Arguments Group of statements, one of which is claimed to follow from the others.

Proposition A statement that is either true or false, usually a declarative sentence.

1.2.2 Connectives

Connectives	Symbols	Meaning
Negation	$\sim$ / $\neg$	Not
Conjunction	$\wedge$	And
Disjunction	$\vee$	Or
Implication/ Conditional	$\rightarrow$	If
Biconditional	$\leftrightarrow$	If and Only If
NAND	$\uparrow$	Not And
NOR	$\downarrow$	Not Or
XOR	$\oplus$	Exclusive Or

1.2.2.1 Summary

Negation Inverts the truth value

Conjunction True when both statements are True

Disjunction True when at least one of the statements are True

Biconditional True when both statements have the same truth value

Implication False if First and Second Statement are True and False respectively, otherwise all configurations are True

NAND Negation of Conjunction

NOR Negation of Disjunction

XOR Negation of Biconditional

1.2.3 Truth Tables

1.2.3.1 Negation

P	$\sim$P
T	F
F	T

1.2.3.2 Conjuction

P	Q	P $\wedge$ Q
T	T	T
T	F	F
F	T	F
F	F	F

1.2.3.3 Disjunction

P	Q	P $\vee$ Q
T	T	T
T	F	T
F	T	T
F	F	F

1.2.3.4 Biconditional

P	Q	P $\leftrightarrow$ Q
T	T	T
T	F	F
F	T	F
F	F	T

1.2.3.5 Implication

P	Q	P $\rightarrow$ Q
T	T	T
T	F	F
F	T	T
F	F	T

1.2.3.6 NAND

P	Q	P $\uparrow$ Q
T	T	F
T	F	T
F	T	T
F	F	T

1.2.3.7 NOR

P	Q	P $\downarrow$ Q
T	T	F
T	F	F
F	T	F
F	F	T

1.2.3.8 XOR

P	Q	P $\oplus$ Q
T	T	F
T	F	T
F	T	T
F	F	F

Chapter 2
QR Code Standard

2.1 Front Matter

2.1.1 Foreword

This chapter is an independent compilation and reinterpretation of the QR Code standard. It is informed by publicly available resources such as research papers, technical books, developer documentation, and collaborative knowledge bases. While I did not have access to the complete ISO/IEC publications due to their cost and limited availability, this document draws upon secondary sources and open references to provide an accessible explanation of QR Code structure and encoding principles.

Where appropriate, annotations and commentary have been added to clarify complex topics and to make the material approachable for readers who may wish to implement their own encoder or decoder, or who are simply curious about how QR codes function.

This is not an official or authoritative standard. It should be read as a study guide and commentary rather than as a normative reference.

Happy scanning, bestie :3

2.1.2 Disclaimer

This chapter is an independent, non-official reinterpretation of the QR Code standard, intended for educational and reference purposes. It is not affiliated with, endorsed by, or a substitute for any publication by the International Organization for Standardization (ISO), the International Electrotechnical Commission (IEC), or any related standards body.

All technical descriptions herein are based on information from publicly accessible and legally obtainable sources. Care has been taken to avoid reproducing verbatim text from proprietary publications. Commentary and interpretation are original contributions of the author, provided to improve accessibility and understanding.

This work is offered under principles of fair use and fair dealing for scholarly and educational purposes. Readers requiring normative or compliance-related guidance should consult the official ISO/IEC 18004 standard or its national equivalents.

2.1.3 Introduction

2.1.4 Introduction

A QR Code (Quick Response Code) is a two-dimensional matrix barcode invented in 1994 by Masahiro Hara of the Japanese company Denso Wave, originally for tracking automobile parts. QR Codes are recognizable by their square modules arranged on a contrasting background, typically with position-detecting fiducial markers in three corners.

Data encoded within a QR Code is interpreted by imaging devices such as cameras, and recovered using error—correction algorithms—most notably Reed–Solomon error correction—to ensure robustness against noise and distortion. The encoded information is embedded in both the horizontal and vertical dimensions of the symbol, enabling compact and efficient data storage and retrieval.[1]

2.2 Definition of Terms

2.2.1 Abbreviations

2D	two-dimensional [2]
QR code	quick-response code [1]
API	Application Programming Interface [3]

2.3 Symbol Description

QR code is a 2D barcode with the following characteristics:

1. Formats:
   a. QR code, the full version; and
   b. Micro QR code, a smaller version of the QR code standard for applications where symbol size is limited.
2. Character Set[1]:
   a. Numeric: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
   b. Alphanumeric: 0–9, A–Z (upper-case only), space, $ (dollar sign), % (percent symbol), * (asterisk), + (plus symbol), - (dash), . (period), / (slash), : (colon)
   c. Binary/byte: ISO/IEC 8859-1 as described below.
   d. Kanji/kana: Shift JIS X 0208 as described below.

2.4 Character Sets

2.4.1 ISO/IEC 8859-1

According to Wikipedia, ISO/IEC 8859-1:1998, *Information technology—8-bit single-byte coded graphic character sets—Part 1: Latin alphabet No. 1*, is part of the ISO/IEC 8859 series of ASCII-based standard character encodings, first edition published in 1987. ISO/IEC 8859-1 encodes what it refers to as "Latin alphabet no. 1", consisting of 191 characters from the Latin script[4].

2.4.1.1 Decode (Code → Character)

1. Start with the hex code. Example: **`0x2F`**
2. Remove the prefix **`0x`**.
3. Split into two nibbles:
   a. High nibble = first hex digit (**`2`**).
   b. Low nibble = second hex digit (**`F`**).
4. Convert the high nibble into the row by multiplying it by 16 (hex shift). Example: **`2`** → **`20`**
5. The low nibble is the column. Example: **`F`**.
6. Locate the character at row **`20`**, column **`F`** in the table.

1. To get the code of a character—for example, the exclamation mark—one must:
   a. Find the character of interest in the table. If it is not in the table, then it means that the character set does not contain said character.
   b. After finding said character, get its row and column. In this example, the row and column is 2 and 1 respectively.
   c. Then, combine '0x' hex prefix, the row and column digit of said character. In this example, it should result to '0x21'.

Table 2.1: ISO/IEC 8859-1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20	[SP][1]	!	”	#	$	%	&	'	(	)	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	‹	=	›	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{	l	}	~	
80																
90																
A0	[NBSP][2]	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	[SHY][3]	®	¯
B0	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Symbols and Punctuations: 0x21–2F, 0x3A–3F, 0x40, 0x5B–60, 0x7B–7E, 0xA1–A9, 0xAB–AC, 0xAE–B1, 0xB4–B8, 0xBB, and 0xBF

2.4.2 Shift JIS X 0208

2.5 Versions

2.5.1 Explanation

According to Denso Wave Incorporated, QR Code versions determine the data capacity and structure of a code[5].

The following tables are sourced from the official QR Code documentation provided by Denso Wave.

The figures in the Numeral, Alphanumeric, Binary and Kanji columns indicate the maximum allowable number of respective characters, including those for the

[1] Space character
[2] Non-breaking space
[3] Soft hyphen

data bit number.

For example, when using the Version 1 QR Code with correction level L, the maximum allowable characters for data bit number, numerals, binaries, and Kanji are 152, 25, 17, and 10 respectively.

As explained by Denso Wave in the aforementioned documentation, to determine the version of the QR code to use, suppose the data to be input consists of 100-digit numerals. This can be achieved by following the steps described below[5]:

1. In the QR Code table, choose "Numeric" as the type of input data.
2. Choose an error correction level from the options of L, M, Q and H. For example, the error correction level that will be used is M.
3. Find a figure in the table, 100 or over and the closest to 100 that is at the intersection with a correction level M row. The number of the version row that contains this figure is the most appropriate version number. We can see at the tables below that the version closest to that is Version 3 with capacity for up to 101 numeric characters at error correction M.

2.5.2 Micro QR Code

Version	Modules ({w}×{h})	ECC Level	Numeric	Alphanumeric	Binary	Kanji
M1	11	-	5	-	-	-
M2	13	L	10	6	-	-
		M	8	5	-	-
M3	15	L	23	14	9	6
		M	18	11	7	4
M4	17	L	35	21	15	9
		M	30	18	13	8
		Q	21	13	9	5

2.5.3 QR Code

Version	Modules ({w}×{h})	ECC Level	Data bits (mixed)	Numeric	Alpha-numeric	Binary	Kanji
1	21	L	152	41	25	17	10
		M	128	34	20	14	8
		Q	104	27	16	11	7
		H	72	17	10	7	4
2	25	L	272	77	47	32	20
		M	224	63	38	26	16
		Q	176	48	29	20	12
		H	128	34	20	14	8
3	29	L	440	127	77	53	32
		M	352	101	61	42	26
		Q	272	77	47	32	20
		H	208	58	35	24	15
4	33	L	640	187	114	78	48
		M	512	149	90	62	38
		Q	384	111	67	46	28
		H	288	82	50	34	21
5	37	L	864	255	154	106	65
		M	688	202	122	84	52
		Q	496	144	87	60	37
		H	368	106	64	44	27
6	41	L	1088	322	195	134	82
		M	864	255	154	106	65
		Q	608	178	108	74	45
		H	480	139	84	58	36
7	45	L	1248	370	224	154	95
		M	992	293	178	122	75
		Q	704	207	125	86	53
		H	528	154	93	64	39
8	49	L	1552	461	279	192	118
		M	1232	365	221	152	93
		Q	880	259	157	108	66
		H	688	202	122	84	52
9	53	L	1856	552	335	230	141
		M	1456	432	262	180	111
		Q	1056	312	189	130	80
		H	800	235	143	98	60

Version	Modules ({w}×{h})	ECC Level	Data bits (mixed)	Numeric	Alpha-numeric	Binary	Kanji
10	57	L	2192	652	395	271	167
		M	1728	513	311	213	131
		Q	1232	364	221	151	93
		H	976	288	174	119	74
11	61	L	2592	772	468	321	198
		M	2032	604	366	251	155
		Q	1440	427	259	177	109
		H	1120	331	200	137	85
12	65	L	2960	883	535	367	226
		M	2320	691	419	287	177
		Q	1648	489	296	203	125
		H	1264	374	227	155	96
13	69	L	3424	1022	619	425	262
		M	2672	796	483	331	204
		Q	1952	580	352	241	149
		H	1440	427	259	177	109
14	73	L	3688	1101	667	458	282
		M	2920	871	528	362	223
		Q	2088	621	376	258	159
		H	1576	468	283	194	120
15	77	L	4184	1250	758	520	320
		M	3320	991	600	412	254
		Q	2360	703	426	292	180
		H	1784	530	321	220	136
16	81	L	4712	1408	854	586	361
		M	3624	1082	656	450	277
		Q	2600	775	470	322	198
		H	2024	602	365	250	154
17	85	L	5176	1548	938	644	397
		M	4056	1212	734	504	310
		Q	2936	876	531	364	224
		H	2264	674	408	280	173
18	89	L	5768	1725	1046	718	442
		M	4504	1346	816	560	345
		Q	3176	948	574	394	243
		H	2504	746	452	310	191

Version	Modules ({w}×{h})	ECC Level	Data bits (mixed)	Numeric	Alpha-numeric	Binary	Kanji
19	93	L	6360	1903	1153	792	488
		M	5016	1500	909	624	384
		Q	3560	1063	644	442	272
		H	2728	813	493	338	208
20	97	L	6888	2061	1249	858	528
		M	5352	1600	970	666	410
		Q	3880	1159	702	482	297
		H	3080	919	557	382	235
21	101	L	7456	2232	1352	929	572
		M	5712	1708	1035	711	438
		Q	4096	1224	742	509	314
		H	3248	969	587	403	248
22	105	L	8048	2409	1460	1003	618
		M	6256	1872	1134	779	480
		Q	4544	1358	823	565	348
		H	3536	1056	640	439	270
23	109	L	8752	2620	1588	1091	672
		M	6880	2059	1248	857	528
		Q	4912	1468	890	611	376
		H	3712	1108	672	461	284
24	113	L	9392	2812	1704	1171	721
		M	7312	2188	1326	911	561
		Q	5312	1588	963	661	407
		H	4112	1228	744	511	315
25	117	L	10208	3057	1853	1273	784
		M	8000	2395	1451	997	614
		Q	5744	1718	1041	715	440
		H	4304	1286	779	535	330
26	121	L	10960	3283	1990	1367	842
		M	8496	2544	1542	1059	652
		Q	6032	1804	1094	751	462
		H	4768	1425	864	593	365
27	125	L	11744	3517	2132	1465	902
		M	9024	2701	1637	1125	692
		Q	6464	1933	1172	805	496
		H	5024	1501	910	625	385

Version	Modules ({w}×{h})	ECC Level	Data bits (mixed)	Numeric	Alpha-numeric	Binary	Kanji
28	129	L	12248	3669	2223	1528	940
		M	9544	2857	1732	1190	732
		Q	6968	2085	1263	868	534
		H	5288	1581	958	658	405
29	133	L	13048	3909	2369	1628	1002
		M	10136	3035	1839	1264	778
		Q	7288	2181	1322	908	559
		H	5608	1677	1016	698	430
30	137	L	13880	4158	2520	1732	1066
		M	10984	3289	1994	1370	843
		Q	7880	2358	1429	982	604
		H	5960	1782	1080	742	457
31	141	L	14744	4417	2677	1840	1132
		M	11640	3486	2113	1452	894
		Q	8264	2473	1499	1030	634
		H	6344	1897	1150	790	486
32	145	L	15640	4686	2840	1952	1201
		M	12328	3693	2238	1538	947
		Q	8920	2670	1618	1112	684
		H	6760	2022	1226	842	518
33	149	L	16568	4965	3009	2068	1273
		M	13048	3909	2369	1628	1002
		Q	9368	2805	1700	1168	719
		H	7208	2157	1307	898	553
34	153	L	17528	5253	3183	2188	1347
		M	13800	4134	2506	1722	1060
		Q	9848	2949	1787	1228	756
		H	7688	2301	1394	958	590
35	157	L	18448	5529	3351	2303	1417
		M	14496	4343	2632	1809	1113
		Q	10288	3081	1867	1283	790
		H	7888	2361	1431	983	605
36	161	L	19472	5836	3537	2431	1496
		M	15312	4588	2780	1911	1176
		Q	10832	3244	1966	1351	832
		H	8432	2524	1530	1051	647

Version	Modules (w×h)	ECC Level	Data bits (mixed)	Numeric	Alpha-numeric	Binary	Kanji
37	165	L	20528	6153	3729	2563	1577
		M	15936	4775	2894	1989	1224
		Q	11408	3417	2071	1423	876
		H	8768	2625	1591	1093	673
38	169	L	21616	6479	3927	2699	1661
		M	16816	5039	3054	2099	1292
		Q	12016	3599	2181	1499	923
		H	9136	2735	1658	1139	701
39	173	L	22496	6743	4087	2809	1729
		M	17728	5313	3220	2213	1362
		Q	12656	3791	2298	1579	972
		H	9776	2927	1774	1219	750
40	177	L	23648	7089	4296	2953	1817
		M	18672	5596	3391	2331	1435
		Q	13328	3993	2420	1663	1024
		H	10208	3057	1852	1273	784

https://en.wikiversity.org/wiki/Reed

https://www.thonky.com/qr-code-tutorial/module-placement-matrix

Chapter 3
Sample Title

In conclusion: give me a degree

Bibliography

[1] Wikipedia contributors. *QR code*. May 2025. URL: **https://en.wikipedia.org/wiki/QR_code** (Retrieved June 6, 2025).

[2] Merriam-Webster. *2D*. URL: **https://www.merriam-webster.com/dictionary/2D** (Retrieved June 12, 2025).

[3] Wikipedia contributors. *API*. June 2025. URL: **https://en.wikipedia.org/wiki/API** (Retrieved June 12, 2025).

[4] Wikipedia contributors. *ISO/IEC 8859-1*. May 2025. URL: **https://en.wikipedia.org/wiki/ISO/IEC_8859-1**.

[5] DENSO WAVE INCORPORATED. *Information capacity and versions of QR Code*. URL: **https://www.qrcode.com/en/about/version.html** (Retrieved June 12, 2025).