Gujarat Technological University

Chandkheda, Ahmedabad

Affiliated

Vishwakarma Government Engineering College

A Project Report On

ViFi – Vision Fidelity

(UDP) User Defined Project

Under subject of

Project – II

B.E. IV, Semester -- 8th

(Electronics and Communication Engineering)

Submitted By:

Alokendu Mazumder	150170111003
Zeal Patel	150170111078
Rahul Modi	150170111053

Guided By:

Prof. JagrutiMakwana

(Academic Year 2019-2020)

Certificate

This is to certify that the final year project under semester 8th, “**ViFi – Vision Fidelity**”, has been satisfactorily completed and submitted for the subject **Project – 2** by AlokenduMazumder (150170111003),within the four walls of Vishwakarma Government Engineering College, Ahmedabad. This project has been thus submitted satisfactorily by the following team of students in partial fulfilment of the degree of Bachelor of Engineering in Electronics and Communication Engineering.

Signature	**Signature**
Prof.JagrutiMakwana	*Dr*. Rajesh A. Thakkar
EC Department	Head of Department
VGEC	EC Department
(*Internal Project Guide*)	VGEC

Certificate

This is to certify that the final year project under semester 8th, "**ViFi – Vision Fidelity**", has been satisfactorily completed and submitted for the subject **Project – 2** by Zeal Patel (150170111078), within the four walls of Vishwakarma Government Engineering College, Ahmedabad. This project has been thus submitted satisfactorily by the following team of students in partial fulfilment of the degree of Bachelor of Engineering in Electronics and Communication Engineering.

Signature

Prof.JagrutiMakwana
EC Department
VGEC
(*Internal Project Guide*)

Signature

Dr. Rajesh A. Thakkar
Head of Department
EC Department
VGEC

Certificate

This is to certify that the final year project under semester 8th, "**ViFi – Vision Fidelity**", has been satisfactorily completed and submitted for the subject **Project – 2** by Rahul Modi (150170111053), within the four walls of Vishwakarma Government Engineering College, Ahmedabad. This project has been thus submitted satisfactorily by the following team of students in partial fulfilment of the degree of Bachelor of Engineering in Electronics and Communication Engineering.

Signature

Prof.JagrutiMakwana
EC Department
VGEC
(*Internal Project Guide*)

Signature

Dr. Rajesh A. Thakkar
Head of Department
EC Department
VGEC

Acknowledgement

We would like to express our special thanks of gratitude to our Project Guide ***Prof.JagrutiMakwana*** who gave us the golden opportunity to do this wonderful project on the topic of ***“ViFi – Vision Fidelity”***. Under his esteemed guidance, we have been able to research upon the current trends on the said project and it has benefitted us a lot in understanding the recent innovations in the field.

ABSTRACT

Nowadays people are always in hurry, whether it’s their workplace, home, railway stations, airports and etc. In the midst of all the other necessary activities people use their smartphones to share photos, important documents, apps and etc under the bound of hotspot and internet. Now assume a scenario where a person wants to share a file from his cellphone to his old PC and he’s out of option in terms of internet connectivity and USB sticks. Now in such cases the data transfer becomes a tedious process.

ViFi’s primary aim to resolve this kind of data transfer problems and it assures data transfer in absence of any internet, hotspot and Bluetooth connectivity and capable of working on any electronic devices. The ViFi shares a simple HTML UI which can act as both transmitter and receiver.

Index

Chapter 1: Background of ViFi

1.1) Background – I: Application oriented Problem Statements

Education Sector

India is country of approx. 140 crore people and advancing at a very high rate in major technological sectors. Almost all are using smartphones, laptops, high speed internet, in their day to day life as these stuffs makes life simpler to live. Apart from the population using these advance technologies, there resides the heart of true India, which is dubbed as “RURAL INDIA”. The government is trying their best to equip rural side of India with these technologies and hence launched several programs to spread awareness about it. Most of the programs launched by Government are recorded as huge success, like Digital India, BHIM app and etc. These programs aim to renovate the rural part and embed it with the latest technological advancement happening out there.

As of the statistics collected by the some leading media giants, it’ concluded that as of 2018, there’s only 18% of Mobile internet penetration in rural India. Also another report claims that over 900 million people in India can’t get online. In 2015, only 22% adults in our country has access to internet, according to the pew research center, that ranks India far behind other large developing countries like China and Brazil where adult’s internet access rates are 60% and 65% respectively.

The internet access and smartphone ownership in India has increased in recent years, but Asia’s third largest economy still lags behind most other emerging economies. As many as 54% of people across emerging and developing countries reported using internet at leastoccasionally or owing a smartphone in 2015 compared with 45% in 2013. Most of the increase came from emerging countries like Malaysia, Brazil and China. India still lags behind half of the survey countries in term of Internet usage and smartphone ownership.

Interestingly, India also reported one of the highest levels of responds who own a non-smartphone cellphone, at 61% Ukraine (64%), Senegal (63%), Nigeria (62%) and Tanzania (66%).

As a matter of fact, the digital divide also includes educational levels and income. In India, only 9% of those with lower education levels are online, as compared with 38% who have higher education levels. Similarly, the income gap too, is reflected in the survey. 11% of lower income group in India reported that they use Internet occasionally and owned Smartphone in contrast to 28% of those with higher incomes.

Also, from a survey conducted by The Hindu, it was found that Rural India has 17% more homes with TV sets than cities.

The above line formed the basis of possible application of Vi-Fi in almost every sector, which will be discussed in details later on.

By studying the above statistics and analyzing them we can conclude that still there are some places in Rural India where internet connectivity and smartphone

ownership is very less. Hence the digital resources are still somewhat inaccessible to these places.

Advertising Agencies

Imagine how great it would be if get any product or information about any product directly from television? Let say, an advertisement of HIKE app is going on at any of the TV channel during a regular show, and as a customer I wish to get the app directly from TV to my smartphone without even opening play store.

Hence, life would be easier if we are able to fetch any product or data or any kind of government forms directly from the TV during its commercial.

As ViFi is capable of transmitting any file extension (apk in this case) from one screen to another device. More detailed approach will be discussed earlier in this text.

Railway Stations and Airport Terminals

In railway station and in airport we can observe that they use large LCD screens to display the arrival and departure information of the ongoing and forthcoming trains and flights respectively. These large screens costs very high and required daily maintenance and services in order to run them smoothly. Although if they malfunction the whole place may turn up into chaos and the public will face difficulties.

Hence in order to reduce the costing spend on these large LCD displays and to maintain smooth records of arrival and departure timings and deliver it to passengers, ViFi can be used as it required a small screen which can handle a display of small QR codes. The passengers can directly fetch the data from the QR codes placed in such places without any kind of connectivity.

Schools, Colleges & Workplace

As discussed in the very first paragraph, mostly people don’t carry any additional USB sticks and drives with themselves at their workplace, whether it be school, college or any other office. And if the internet connectivity there gets compromised the data transfer from PC/Laptop or any other device become tedious in nature.

Hence to make sure that transfer of photos, files and other data should be smooth even in absence of USB drives, internet and hotspot, ViFi is orginated to overcome these problems. ViFi is proved to be easy to use and can run on almost any smartphone and PC without any additional requirement.

Chapter 2: Theory, Market Survey& Design Analysis

Vi-Fi has basically two portions, one is 'Sender ' and other is 'Receiver '. Both themodules are coded in the same HTML page so the user can easily use our project for data transmission and reception.

Sender portion consists of QR code generation part which flashes a series of QR codes generated by a file selected for the transmission. While Receiver portionuses QR scanner library 'Instascan' from GitHub. This library uses a camera and scan QR codes. In output, it gives data incorporated in QR codes. Further, this data is processed and combined to create an original file which was transmitted and downloaded in the receiver device.

All libraries and algorithm to transmit and receive file are included in a single HTML page of size 1.51 MB. As it is a HTML file, it can be used across any platform as long as they support JavaScript and have file access permissions. The user interface is simplest among other file transfers application outthere. It has got two big buttons as SENDER and RECEIVER. To send file just select file you want to transmit and to receive a file just point your camera towards flashing QR codes.

2.1) Block Diagram

1) Transmitter

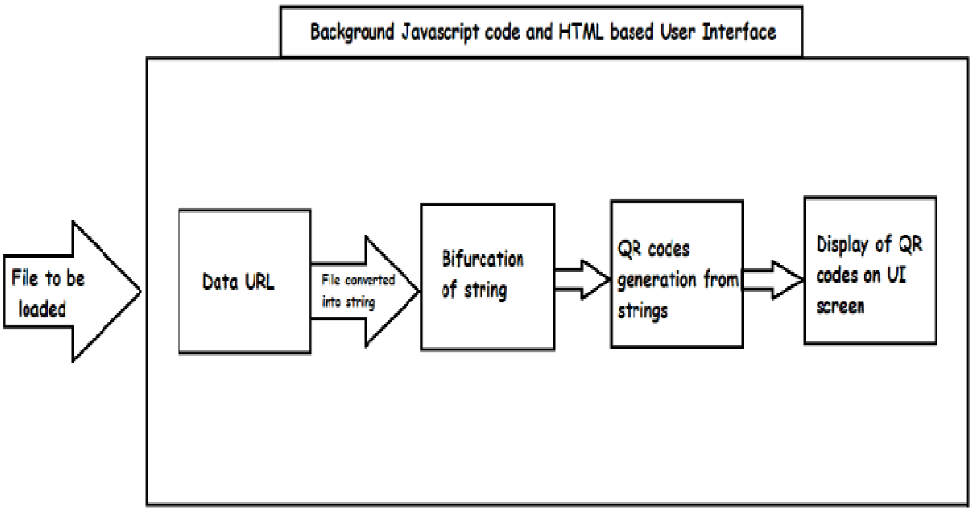

Background Javascript code and HTML based User Interface
File to be loaded
Data URL
File converted into string
Bifurcation of string
QR codes generation from strings
Display of QR codes on UI screen

Fig. 2.1.1 – Transmitter Block Diagram

2) Receiver

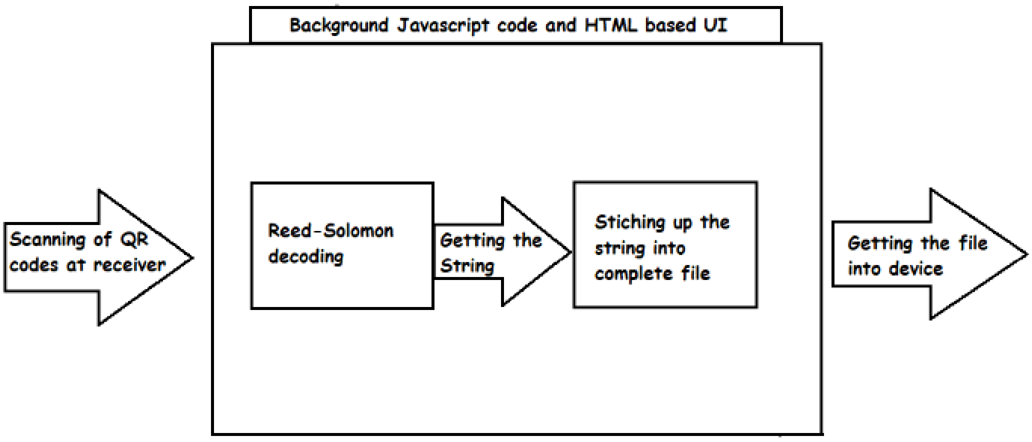

Background Javascript code and HTML based UI
Scanning of QR codes at receiver
Reed-Solomon decoding
Getting the String
Stiching up the string into complete file
Getting the file into device

Fig. 2.1.2 – Receiver Block Diagram

2.2) Working Algorithm

- ✓ File which is to be transmitted is converted into DATA URL
- ✓ That DATA URL in converted into string
- ✓ The length of string is then calculated and that string is divided in smallest possible set which is required to make a complete QR code.
- ✓ Hence depending on the size of files, number of QR codes are generated, in simple ways a complete file is divided uniformly among several QR codes.
- ✓ Another special QR code is generated which contains information like, name of file, size of file, the number of QR codes generated and etc.
- ✓ Each QR code contains a unique header which is essential to specify its position among all QR codes.
- ✓ As the file is completely transformed into QR codes, the number of QR codes will start flashing at the sender's screen.
- ✓ On the other hand the receiver will flash device's camera to the flashing QR codes to receive the incoming data.
- ✓ The receiver will perform the reverse action and will stich all the data of QR code into a single predefined file format.

2.3)Novelity of ViFi

Our algorithm doesn’t needs any internet connectivity and external circuitry or drivers (like used in LiFi) for data transfer. It is simple protocol where any type of data can be encoded into a stream of QR codes and user can fetch the data by simply putting his/her cell phone’s camera.

As discussed earlier, a receiving device with 5 MP or above can fetch the data directly just by pointing at the flashing QR codes. The receiver algorithm will fetch the data using Reed Solomon decoding technique and convert it into string, and finally stitching it into a complete file.

ViFi is purely hardware independent and it doesn’t require any extra hardware circuitry and can run easily on any existing technology and device.

Any size of data can be converted into stream of QR codes.

Also, during TV commercials like government related acts and schemes, if the QR codes are displayed during the ads, the user can fetch the data directly from using the device, living in any corner of the world even where internet connectivity is not available.

Hence its can used for mass broadcasting and smart education programs, government schemes in rural areas where smartphone ownership and internet connectivity is very low.

By the same idea, eBooks can also be downloaded in similar way from TV directly.

2.4) Current National Status of Idea

Currently the QR codes are ONLY used in transfer of money from one bank account/wallet to another. App’s like Paytm uses QR technology for money transfer, that too in presence of internet.

In International Research Journal of Engineering and technology, similar looking idea was proposed, BUT they are using advance image processing techniques for data transfer also, their algorithm is restricted to generate ONLY ONE single QR code, hence it applied an upper limit to the size of data file to be transmitted, whereas our algorithm has no upper limit to transfer data, as it can generate multiple QR codes for data transfer.

2.5) Current International status of Idea

A technology named ThruGlassXfer exists to transfer data in form of QR codes, BUT this technology required an Arduino based keyboard and mouse in order to upload the data and convert it into QR code streams, hence additional circuitry is required in this protocol and carrying keyboard and mouse everywhere you go is not preferable for everyone. Plus the QR code streams that are being generated are directly displayed to YOUTUBE, from there user can download the data by simply flashing their cell phone’s camera to the QR codes. Hence this technology can’t work in remote areas. Again this data transfer is completely dependent on additional circuitry and internet. For high speed transfer, they are using additional HDMI capture technology which again required bulky hardware parts.

Another US patent is found where data is transfer via QR codes, but the data is limited to URL only. In order to avoid the attack of malware and viruses, the URL of website is first converted into QR code and that code is checked for potential threats, if it’s clear then it allows user to visit the site by scanning the QRQ code. Again, dependency on internet came to picture and also the data type is restricted to ORL only.

2.6) Design Survey:

1. Is a product related to your project work available in the market? (YES/NO):

- ✓ NO

2. Which features are you trying to improve if the product is already available in the market?

- ✓ NIL

3. Will you be using a better technology? If YES, which technology?

- ✓ WE are trying to develop easy file transmission system on existing technology

4. Are you trying to reduce the cost? (YES/NO):

- ✓ Our product is not much related to cost, it is just a web page.

Chapter 3: Building Blocks of ViFi

Basically this project has two main pillars: Javascript and HTML. The full transfer and receiving algorithm is incorporated with the help of JavaScript based libraries and HTML based UI. We have used different JavaScript based libraries for QR transfer and reception and combined all the components of the ViFi in a single HTML file which acts as its User Interface.

3.1) QR codes

The QR Code is a two-dimensional version of the barcode, known from product packaging in the supermarket. Originally developed for process optimization in the logistics of the automotive industry, the QR Code has found its way into mobile marketing with the widespread adoption of smartphones. "QR" stands for "Quick Response", which refers to the instant access to the information hidden in the Code. QR Codes are gaining popularity because the technology is **"open source",** i.e. available for everyone. Significant advantages of QR Codes over conventional barcodes are larger data capacity and high fault tolerance.

The black and white checkered pixel patterns appear at first glance to be a small crossword puzzle and seem to be composed at random. But if you look closely, certain structures can be identified. For the scanner to recognize a QR Code as such, the Code must always be square. A number of additional elements ensure that the information is read correctly.

A barcode is a machine-readable optical label that contains information about the item to which it is attached. In practice, QR codes often contain data for a **locator, identifier, or tracker that points to a website or application**. A QR code uses four standardized encoding modes to store data efficiently; extensions may also be used.

The Quick Response system became popular outside the automotive industry due to its fast readability and **greater storage capacity compared to standard UPC barcodes**. Applications include product **tracking, item identification, time tracking, document management, and general marketing.**

A QR code consists of black squares arranged in a square grid on a white background, which can be read by an imaging device such as a camera, **and processed using Reed–Solomon error correction until the image can be appropriately interpreted**. The required data is then extracted from patterns that are present in both horizontal and vertical components of the image.

There are several standards that cover the encoding of data as QR codes:

- ✓ October 1997 – AIM **(Association for Automatic Identification and Mobility)** International
- ✓ January 1999 – JIS X 0510
- ✓ June 2000 – ISO/IEC 18004:2000 ***Information technology – Automatic identification and data capture techniques – Bar code symbology** – QR code* (now withdrawn)
  Defines QR code models 1 and 2 symbols.

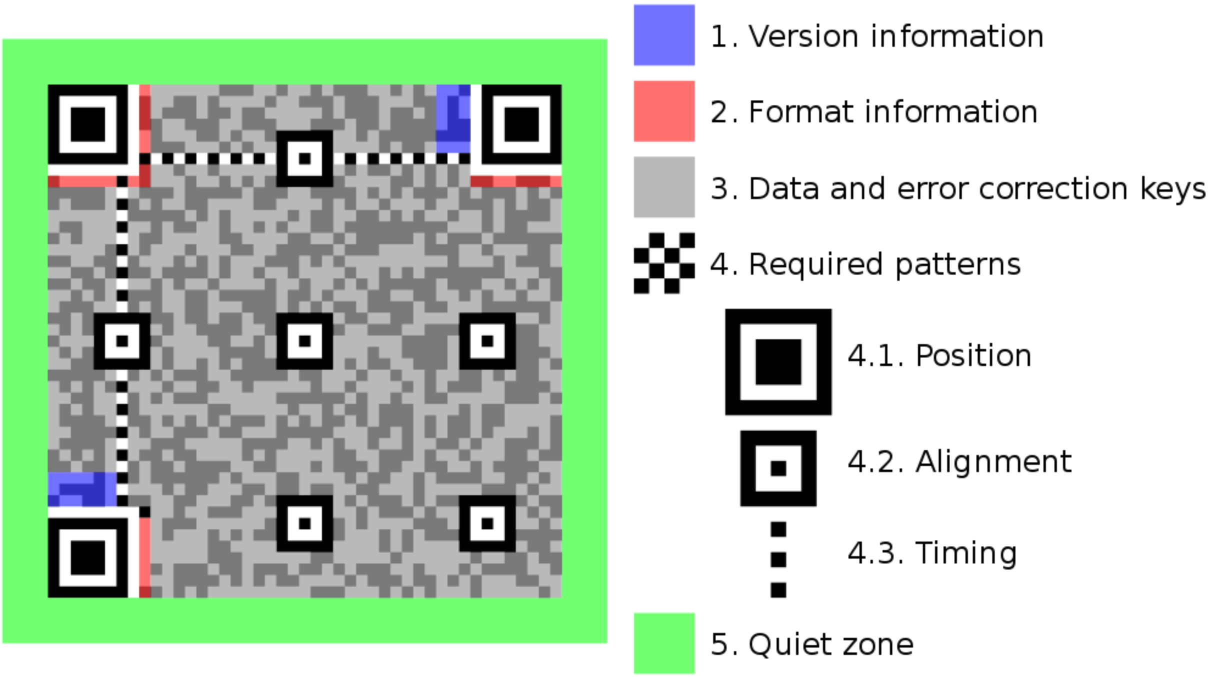
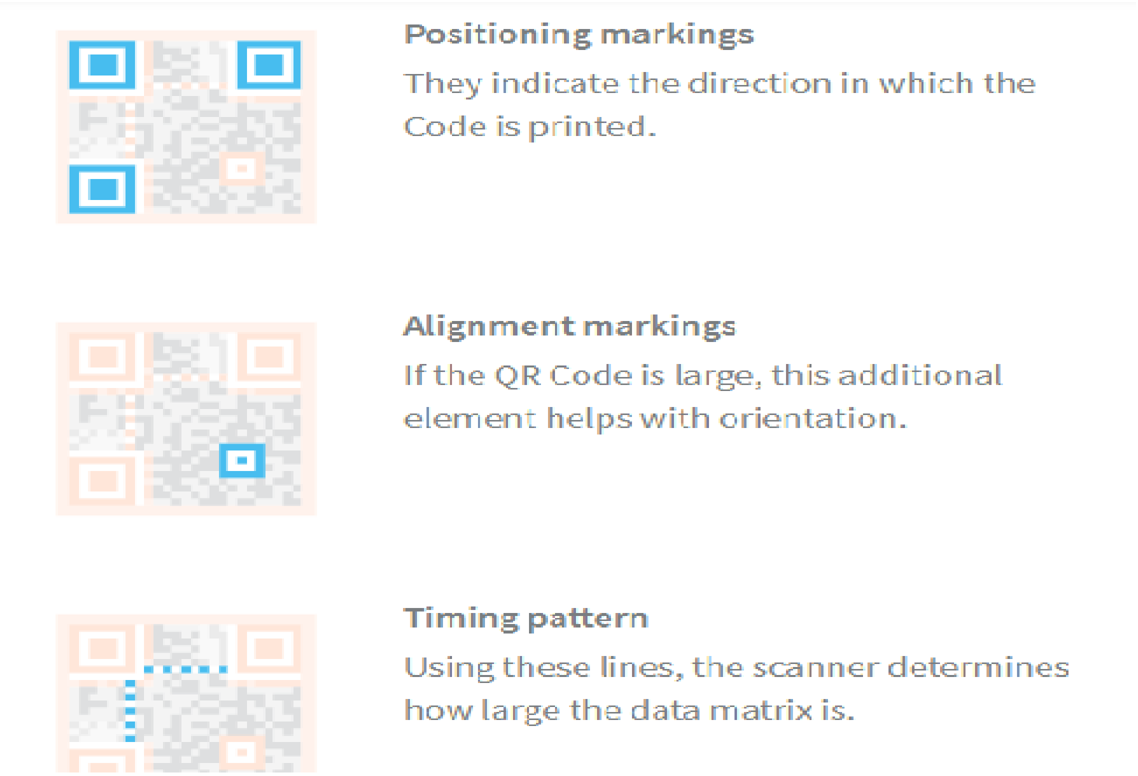


Fig. 3.1.1 - A basic QR code



Positioning markings

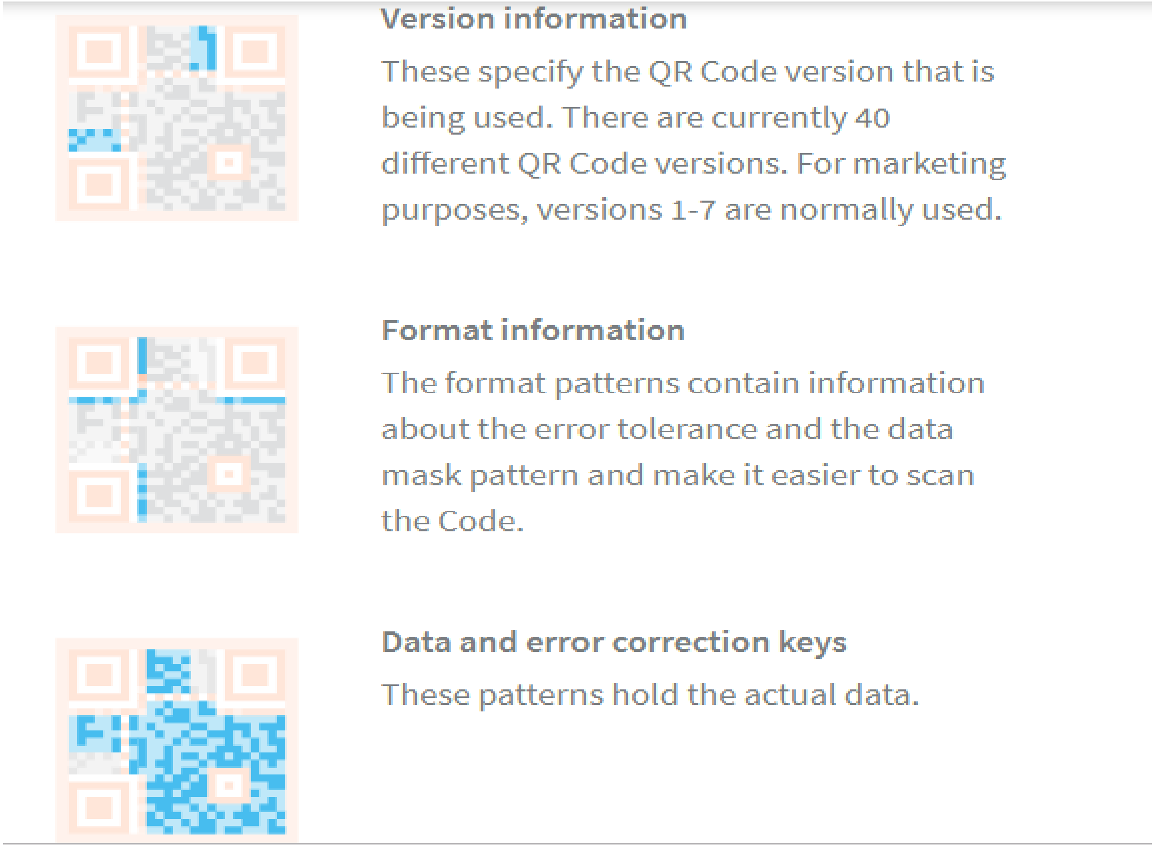
They indicate the direction in which the Code is printed.

Alignment markings

If the QR Code is large, this additional element helps with orientation.

Timing pattern

Using these lines, the scanner determines how large the data matrix is.



Version information

These specify the QR Code version that is being used. There are currently 40 different QR Code versions. For marketing purposes, versions 1-7 are normally used.

Format information

The format patterns contain information about the error tolerance and the data mask pattern and make it easier to scan the Code.

Data and error correction keys

These patterns hold the actual data.

Fig 3.1.2 - QR codes labelling

Capacity of QR codes

Up to 7089 digits or 4296 characters, including punctuation marks and special characters, can be entered in one Code. In addition to numbers and characters, words and phrases (e.g. Internet addresses) can be encoded as well. As more data is added to the QR Code, the Code size increases and the Code structure becomes more complex.

The amount of data that can be stored in the QR code symbol depends on the datatype (*mode*, or input character set), version (1, ..., 40, indicating the overall dimensions of the symbol, i.e. 4 × version number + 17 dots on each side), and error correction level. The maximum storage capacities occur for version 40 and error correction level L (low), denoted by 40-L.

Table 3.1.1

Maximum Character Storage Capacity (40 – L)

Input Mode	Maximum Characters	Bits/Char	Possible Characters and Default Encoding
Numeric only	7,089	3.5	0,1,2,3,4,5,6,7,8,9
Alphanumeric	4,296	5.5	0-9, A-Z and $, %, *, +, -
Binary/Byte	2,953	8	ISO 8859-I
Kanji/Kana	1,817	13	Shift JIS X 0280

Error Correction

Codewords are 8 bits long and use the **Reed–Solomon error correction algorithm** with four error correction levels. The higher the error correction level, the less storage capacity. The following table lists the approximate error correction capability at each of the four levels:

Level L (Low)	7% ofcodewords can be restored.
Level M (Medium)	15% of codewords can be restored.
Level Q (Quartile)	25% of codewords can be restored.
Level H (High)	30% of codewords can be restored.

In larger QR symbols, the message is broken up into several Reed–Solomon code blocks. The block size is chosen so that at most 15 errors can be corrected in each block; this limits the complexity of the decoding algorithm. The code blocks are then interleaved together, making it less likely that localized damage to a QR symbol will overwhelm the capacity of any single block.

Due to error correction, it is possible to create artistic QR codes that still scan correctly, but contain intentional errors to make them more readable or attractive to the human eye, as well as to incorporate colors, logos, and other features into the QR code block.

It is also possible to design artistic QR codes without reducing the error correction capacity by manipulating the underlying mathematical constructs.

QR Encoding techniques

The format information records two things: the error correction level and the mask pattern used for the symbol. Masking is used to break up patterns in the data area that might confuse a scanner, such as large blank areas or misleading features that look like the locator marks. The mask patterns are defined on a grid that is repeated as necessary to cover the whole symbol. Modules corresponding to the dark areas of the mask are inverted. The format information is protected from errors with a BCH code, and two complete copies are included in each QR symbol.

Table 3.1.2: QR Encoding Modes

Indicator	**Meaning**
0001	Numeric encoding (10 bits per 3 digits)
0010	Alphanumeric encoding (11 bits per 2 characters)
0100	Byte encoding (8 bits per character)

Table 3.1.3: QR character count and Size

Mode indicator	Description	Typical structure [Type: size in bits]
0001	Numeric	[0001 : 4] [Character Count Indicator : variable] [Data Bit Stream : 10 × charcount]
0010	Alphanumeric	[0010 : 4] [Character Count Indicator : variable] [Data Bit Stream : 11 × charcount]
0100	Byte encoding	[0100 : 4] [Character Count Indicator : variable] [Data Bit Stream : 8 × charcount]
1000	Kanji encoding	[1000 : 4] [Character Count Indicator : variable] [Data Bit Stream : 13 × charcount]
0011	Structured append	[0011 : 4] [Symbol Position : 4] [Total Symbols: 4] [Parity : 8]
0111	ECI	[0111 : 4] [ECI Assignment number : variable]
0101	FNC1 in first position	[0101 : 4] [Numeric/Alphanumeric/Byte/Kanji payload : variable]
1001	FNC1 in second position	[1001 : 4] [Application Indicator : 8] [Numeric/Alphanumeric/Byte/Kanji payload : variable]
0000	End of message	[0000 : 4]

Types of QR codes

a) Micro QR

Micro QR code is a smaller version of the QR code standard for applications where symbol size is limited. There are four different versions (sizes) of Micro QR codes: the smallest is 11×11 modules; the largest can hold 35 numeric characters.

b) IQR code

IQR Code is an alternative to existing QR codes developed by Denso Wave. IQR codes can be created in square or rectangular formations; this is intended for situations where a rectangular barcode would otherwise be more appropriate, such as cylindrical objects. IQR codes can fit the same amount of information in 30% less space. There are 61 versions of square IQR codes, and 15 versions of rectangular codes. For squares, the minimum size is 9x9 modules; rectangles have a minimum of 19x5 modules. IQR codes add error correction level S, which allows for 50% error correction.[55] IQR Codes have not yet been given an ISO specification, and only proprietary Denso Wave products can create or read IQR codes

c) SQRC code

Secure Quick Response code (SQRC) is a type of QR code that contains a "private data" segment after the terminator instead of the specified filler bytes "ec 11".[57] This private data segment must be deciphered with an encryption key. This can be used to store private information and to manage company's internal information

d) Frame QR

FrameQR is a QR code with a “canvas area” that can be flexibly used. In the center of this code is the canvas area, where graphics, letters, and more can be flexibly arranged, making it possible to lay out the code without losing the design of illustrations, photos, etc

Extension of QR

Researchers have proposed a new High Capacity Colored 2-Dimensional (HCC2D) Code, which builds upon a QR code basis for preserving the QR robustness to distortions and uses colors for increasing data density (at this stage it is still in prototyping phase). The HCC2D code specification is described in details in Querini *et al.* (2014) while techniques for color classification of HCC2D code cells are described in detail in Querini and Italiano (2014), which is an extended version of Querini and Italiano (2013).

Introducing colors into QR codes requires addressing additional issues. In particular, during QR code reading only the brightness information is taken into account, while HCC2D codes have to cope with chromatic distortions during the decoding phase. In order to ensure adaptation to chromatic distortions which arise in each scanned code, HCC2D codes make use of an additional field: the Color Palette Pattern. This is because color cells of a Color Palette Pattern are supposed to be distorted in the same way as color cells of the Encoding Region. Replicated color palettes are used for training machine learning classifiers.

3.2) JavaScript

JavaScript, often abbreviated as **JS**, is a high-level, interpreted programming language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.

The terms *Vanilla JavaScript* and *Vanilla JS* refer to JavaScript not extended by any frameworks or additional libraries. Scripts written in Vanilla JS are plain JavaScript code.

Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design. JavaScript was influenced by programming languages such as Self and Scheme.

Use in Webpages

As of May 2017 94.5% of 10 million most popular web pages used JavaScript. The most common use of JavaScript is to add client-side behavior to HTML pages, also known as Dynamic HTML (DHTML). Scripts are embedded in or included from HTML pages and interact with the Document Object Model (DOM) of the page. Some simple examples of this usage are:

- ✓ Loading new page content or submitting data to the server via Ajax without reloading the page (for example, a social network might allow the user to post status updates without leaving the page).
- ✓ Animation of page elements, fading them in and out, resizing them, moving them, etc.
- ✓ Interactive content, for example games, and playing audio and video.
- ✓ Validating input values of a Web form to make sure that they are acceptable before being submitted to the server.

- ✓ Transmitting information about the user's reading habits and browsing activities to various websites. Web pages frequently do this for Web analytics, ad tracking, personalization or other purposes.

JavaScript code can run locally in a user's browser (rather than on a remote server), increasing the application's overall responsiveness to user actions. JavaScript code can also detect user actions that HTML alone cannot, such as individual keystrokes. Applications such as Gmail take advantage of this: much of the user-interface logic is written in JavaScript, and JavaScript dispatches requests for information (such as the content of an e-mail message) to the server. The wider trend of Ajax programming similarly exploits this strength.

Chapter 4: Implementation and Application

4.1) Implementations and Experimental Results

ViFi is successfully implemented and it’s in developing stage currently. We have tested this project first at DAIICT Gandhinagar during a Hackathon and we won it later for the idea behind the use of QR codes. The possible application that we presented at DAIICT was to use QR codes to transmit digital data and resources to extreme remote places. Later on ViFi also fetched second prize in GTU Hackathon where we proposedit’s another possible application in the field of advertisement and railway and airports. We proposed its possible use for better advertising strategy and to cut down the costing of installing big LCD screens in airport/railway station respectively.

Right now we are more concerned about the flashing interval between two QR codes. The script which we are running allows us to adjust the time interval between the changes of QR code. If the time interval become very small, then QR codes will flash at very high rate and the receiving camera may miss out one or few QR codes, hence landing at incomplete transfer. If the time is very big, then flashing will become very slow and it will make the transfer slow and sluggish.

So, to check out this problem, we have selected an optimum speed based on our several testing on a single smartphone camera and we concluded that 400-600 milliseconds suits best for the optimum time interval of changing the QR codes. Within this interval the receiverdoesn’t misses out a single QR code and also the speed is near about acceptable when transferring small amount of data.

Implementation Table 4.1.1

S.No	**Time interval between two QR codes (in millisecond)**	**Number of QR missed by camera (average value for max 30 QR codes)**
1	50-100	12
2	100-150	10
3	150-200	9
4	200-250	6
5	250-300	3
6	300-400	1
7	400-500	0
8	500-600	0

***The camera used in this experiment belongs to Moto G5 standards.**

From the above experimental data, it is concluded that for optimum and lossless trafer of data, flashing rate should lie between 400-600 milliseconds.

Also we have calculated the data handling capacity of QR codes used in ViFi which is approximately 2.88 KB. Hence a single QR code is capable of holding 2.88 KB worth data independent of file being loaded.

Calculations:

Let’s assume that the flashing rate is 500 milliseconds.

Transmitting speed of data achieved: 2.88 KB/0.5 sec = 5.76 KBps.

With this much of transmitting speed, any camera with lower resolution can also fetch the data completely.

Also, it’s observed that as the data file size increases, the number of QR codes increases very drastically. The number of QR codes depends on several factors like, number of characters in file, file extension and the richness of data in it.

Implementation Table 4.1.2

S.No	File Extension	File Size	Number of QR codes generated
1	Pdf	178 KB	4875
2	Pdf	204KB	5575
3	Pdf	354 KB	9653
4	Pdf	525 Kb	14328
5	Docx	11 KB	324
6	Docx	13 KB	327
7	Docx	14.6 KB	403
8	Docx	16.5Kb	460
9	Jpeg	1.57 MB	44117
10	Jpeg	1.82 MB	50999
11	Jpeg	1.86 MB	52171
12	Jpeg	6.8 MB	190572

It’s observed that as the size of JPEG files varies, the number of QR codes increased by 27000-29000 by per MB increase in jpeg format.

Calculation:

Case - I

From 1.82 MB – 1.57 MB = with increase in 0.25 MB: – 50999-44117 = 6882 QR codes are increased.

Case - II

From 1.86 MB – 1.82 MB = with increase in 0.04 MB: – 52171-50999 = 1172 QR codes increased.

Hence, normalizing the above two cases it’s observed that 27500-29000 QR codes increases in response to increase in 1 MB of data in jpeg format.

These calculations can also be done for different format and then the nature of increase of QR codes can be predicted, and after that suitable steps can be taken to control it.

Right now we can’t predict the exact relation between size of file and number of QR generated as it depends on several other characteristic of file.

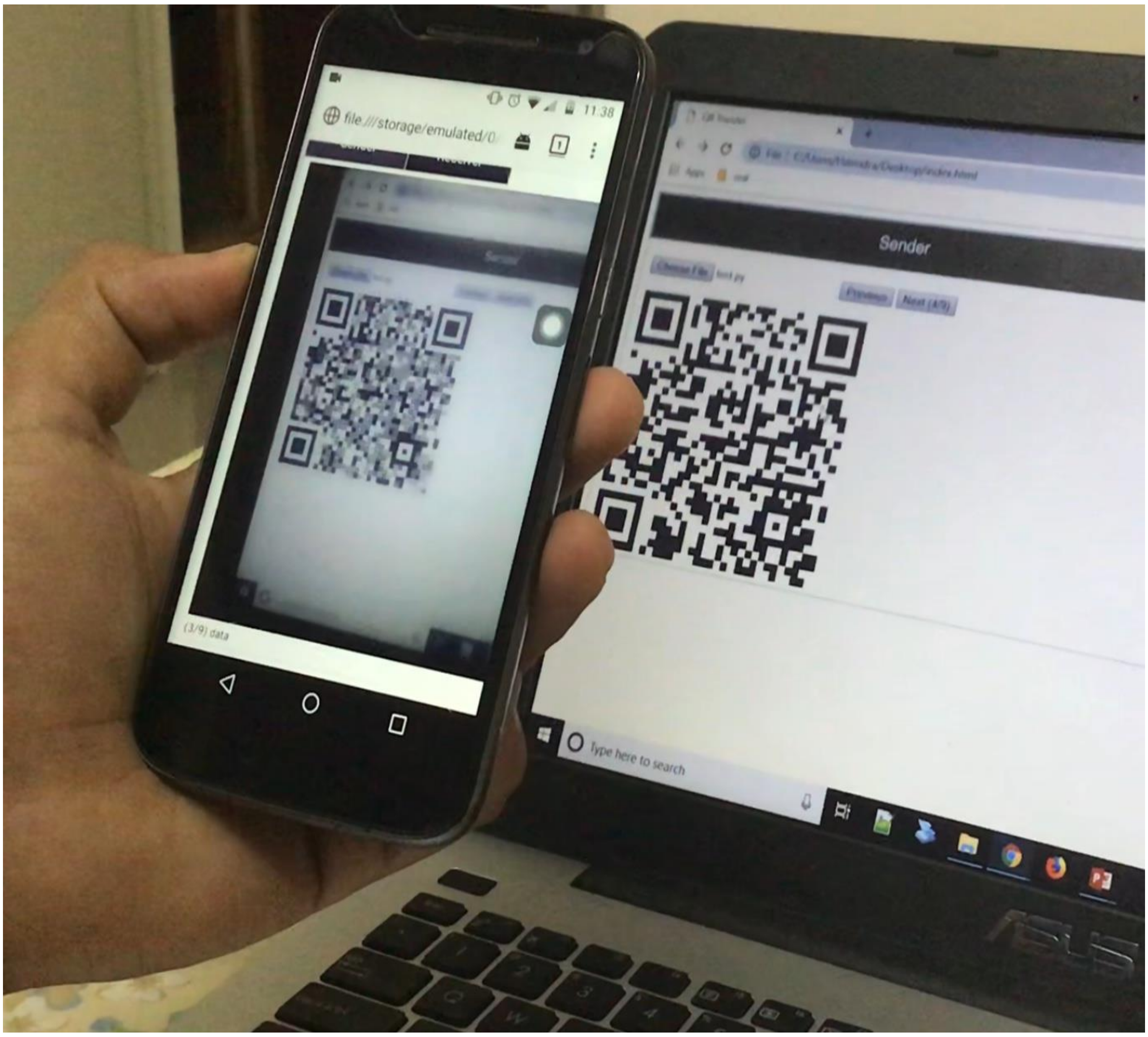
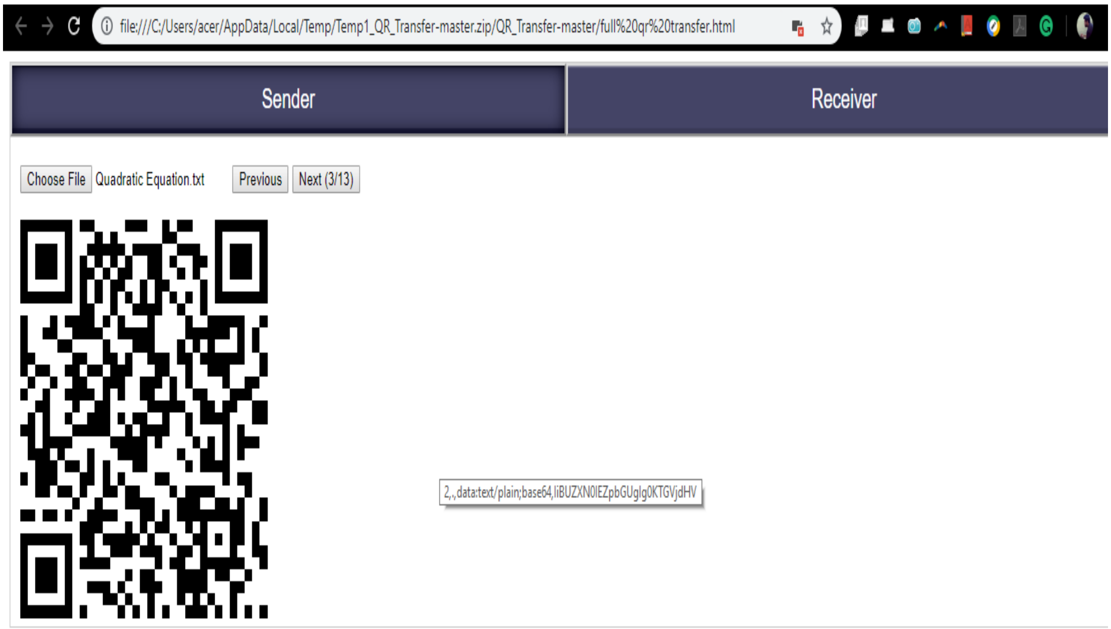


Fig 4.1.1 - ViFi in action


file:///C:/Users/acer/AppData/Local/Temp/Temp1_QR_Transfer-master.zip/QR_Transfer-master/full%20qr%20transfer.html
Sender
Receiver
Choose File
Quadratic Equation.txt
Previous
Next (3/13)

```
2,.,data:text/plain;base64,IiBUZXN0IEZpbGUgIg0KTGVjdHV
```

Fig 4.1.2 – Transmitting UI

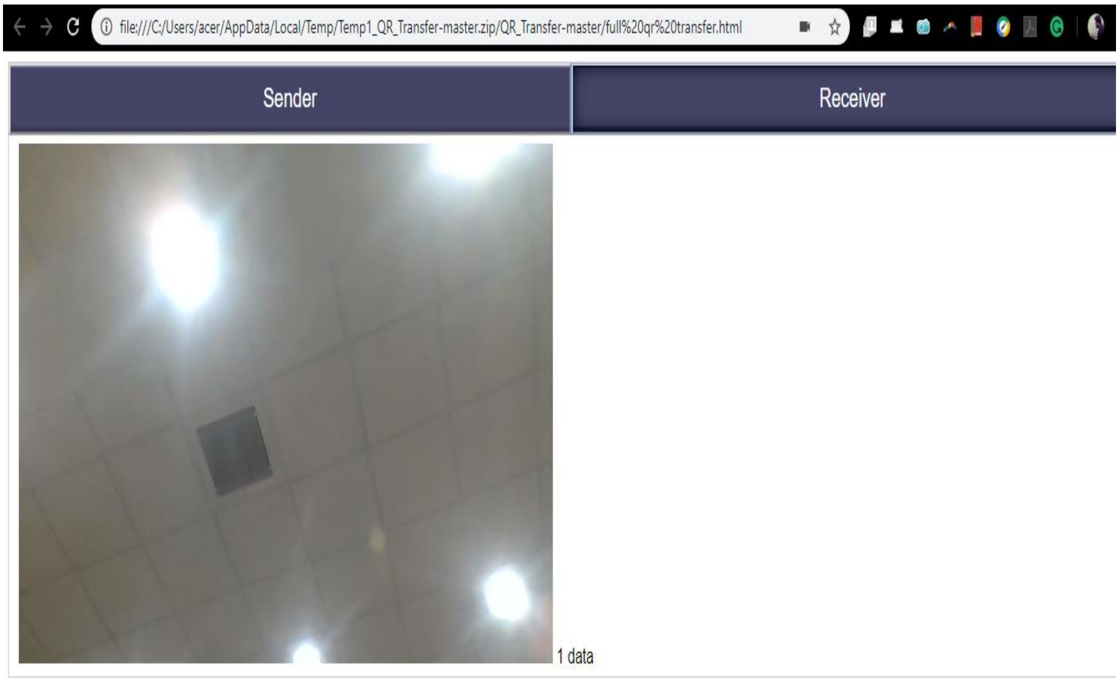

file:///C:/Users/acer/AppData/Local/Temp/Temp1_QR_Transfer-master.zip/QR_Transfer-master/full%20qr%20transfer.html
Sender
Receiver
1 data

Fig 4.1.2 – Receiving UI

4.2) Application:

a) In Education sector

Recalling the statistics from chapter one, we have already witnessed that the smartphone ownership and internet connectivity is very less as compared to other debeloping country.

These are some places in India where internet connectivity is almost zero.

- ✓ **Changlang- Arunachal Pradesh**
- ✓ **Zanskar- J & K**
- ✓ **Valley of Flowers- Uttarakhand**
- ✓ **Kheerganga-Himachal Pradesh**
- ✓ **Remote areas of North East & Sikkim**

Hence it becomes difficult for students to get digital learning resources at the doorsteps. But with the help of ViFi and broadcast ministry spreading of digital resources will become very easy at these places also.

The resources which is to transmitted shall be given to the broadcast ministry. The ministry will then use the transmitting code of ViFi to convert all the files into QR codes series and the flashing codes will be telecasted during any ongoing commercial at TV. Now the student will fetch the data from flashing QR codes on TV using his/her simple cell phone and utilize it. Hence, in this way a mass education outreach program can be launched to distribute digital materials at remote areas.

Not only in extreme rural areas, but as we roam through YouTube for educational videos of NPTEL and etc, ViFi can also be used there. While playing the video, same QR codes can be flashed at any corner of frame which contains important highlights of the video. Student can then download those files cum important notes directly in his cellphone without getting much distracted from video.

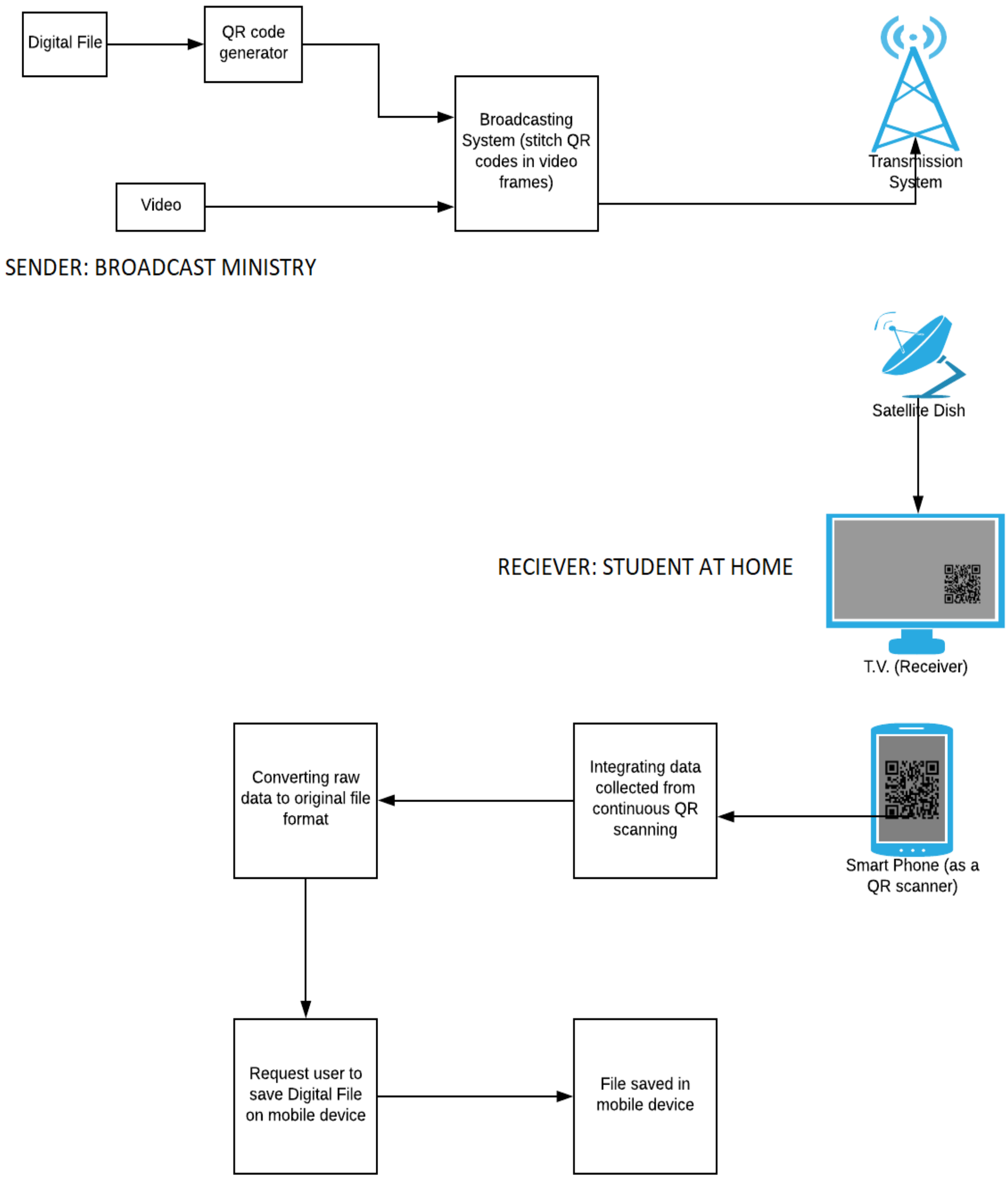

Digital File
QR code generator
Broadcasting System (stitch QR codes in video frames)
Video
Transmission System
SENDER: BROADCAST MINISTRY
Satellite Dish
RECIEVER: STUDENT AT HOME
T.V. (Receiver)
Converting raw data to original file format
Integrating data collected from continuous QR scanning
Smart Phone (as a QR scanner)
Request user to save Digital File on mobile device
File saved in mobile device

Figure 4.1.4– Flow of Education Application

b) Advertising:

Wouldn't it be easy if we get digital products like apps, posters and other important documents like recruitment forms, circulars directly from TV? With the application ViFi it is possible. Assume an advertisement of HIKE app is going on, and a series of QR codes flashed on a corner of TV, juts scan the codes with your smartphone, and you will get the APK downloaded without even moving a finger. Also, if we are viewing any product on TV, just scan the QR codes and the product will get ordered. If we want to choose packages for our channel set. Scan the QR code and it will redirect you to the selection page. Hence in advertising sector, it has millions of application. **The bottom line is, it reduces the effort of searching for any particular entity on internet**.

c) In Railway Stations & Airports

We have seen large LCD screens on airport and railway stations sowing the arrival and departure of trains and planes respectively. Such large screens required continuous and heavy maintenance to run smooth every day and also cost very high. Hence in order to cut down the expenditure towards these bulky and costly screen we can replace these with small screen where QR codes will be flashed and those QR codes will contain real data regarding the timings of transport. Hence a passenger will just look for QR codes in airports/stations, he/she will just scan it and will get all the necessary details.

d) In colleges and workplace

This project was initiated from the very fact that in colleges mostly students don't carry USB sticks and struggle to fetch or transfer data like programming files, project report from their smartphone to PC's and vice versa in absence of internet and other network oriented connections. Now, if in lab or workplace we wish to fetch data from any PC to our PC/Smartphone which is not locally connected and don't have internet access, it becomes difficult for us to deal with. Hence ViFi can be also implemented there. Just load the file in HTML UI and scan up the QR codes with your electronic device, you will get the file in less efforts and no time.

Apart from these above mentioned, other applications are also possible as ViFi is a very application diversified project and can be implemented everywhere.

4.3) Advantages of ViFi:

- ✓ Transfer of data doesn't require internet, hotspot or Bluetooth connectivity.
- ✓ Can be accessed from extreme rural areas.
- ✓ Can transfer any extension of data.
- ✓ Only requirement is, User must have color TV and cellphone having camera 5.0MP and can support docx or pdf or txt format files.
- ✓ Easy data transfer.
- ✓ Can cut down costing of big LCD displays in airport and railway stations.

- ✓ Can run on existing hardware.
- ✓ Doesn’t require any additional circuitry.
- ✓ Very simple UI and purely based on HTML and JavaScript.

4.4) Expected Outcome:

Data can be easily shared at remote areas where smartphone ownership and internet connectivity is not available. E.G during any TV commercial or in any government channel a constant QR code will be flashed continually, the QR code will contain data like Government schemes, eBooks and etc., so the students living there can simply fetch the data from TV directly without any medium. Hence it can be used as tool for mass smart education programs.

Also in Airports and Stations, costing for the large LCD can be cut down to a great extension by replacing them with small screens with flashing QR codes.

Also, less efforts shall be contributed towards finding product in internet or anywhere in web, we can get product directly from TV screen. Also less efforts are given in finding notes to a particular educational lecture in YouTube, just scan away the QR codes flashing during lecture and you will have notes in your pocket!

For commercial uses people are not expected to carry USB, data cables and other hardware’s for data exchange purpose everywhere.

To sum up, implementation of ViFi in various sector will reduce the effort of finding and accompany easy and smooth data transfer from anywhere between two or multiple parties.

Conclusion

ViFi project concludes with the fact that, with its implementation it will not only make the data sharing process much easier at individual as well as at multiuser level it also reduce the efforts of ordinary people to search for any **ENTITY** at internet. Not only that it can also be used to save money and cut down the expenditure of Railway stations and Airports and other public places where it can be implanted.

With its additional application it may someday dominate the advertising world as it enables the user to get the product directly from TV and other screens.

Also it will prove a real asset in the educational sector and mission Digital India, with the aid of broadcasting technologies it will spread the resources to each and every corner of the earth, hence ***nullifying the dependencies on smartphone and internet.***

Future Goals

✓ To increase the transfer and reception rate without compromising the fact that it ca be run on low hardware oriented devices.

✓ To embed error detection and correction codes in it so that large files can be transfer without getting corrupted.

✓ As of now, we are also planning to develop a color based QR code to store more data in a single QR code.

Right now the QR codes have only black and white color, hence the receiver interprets them as binary 0 and 1. Hence there are only two possible symbol in a single QR code. So in order to increase the amount of information in it, we are planning to use color QR codes where each color indicates a unique symbol and in this way more amount of data can be accommodate in a single QR. Also we can apply QR codes with different greyscale levels, like with 256 possible greyscale squares in QR codes, each code will now contain 256 different kind of data in it, hence 256 unique symbols are now possible.

Also, Reed Solomon correction codes can be applied for damage free data transfer. Also in order to detect the mismatch and tampering of data in midway we are planning to use the cryptographic hash code.

(i) Reed Solomon Codes

Reed–Solomon codes are a group of error-correcting codes that were introduced by Irving S. Reed and Gustave Solomon in 1960.They have many applications, the most prominent of which include consumer technologies such as CDs, DVDs, Blu-ray Discs, QR Codes, data transmission technologies such as DSL and WiMAX, broadcast systems such as satellite communications, DVB and ATSC, and storage systems such as RAID 6.

Reed–Solomon codes operate on a block of data treated as a set of finite field elements called symbols. For example, a block of 4096 bytes (32,768 bits) could be treated as a set of 2731 12-bit symbols, where each symbol is a finite field element of GF(2^{12}), the last symbol padded with four 0 bits. Reed–Solomon codes are able to detect and correct multiple symbol errors. By adding *t* check symbols to the data, a Reed–Solomon code can detect any combination of up to and including *t* erroneous symbols, or correct up to and including $\lfloor t/2 \rfloor$ symbols. As an erasure code, it can correct up to and including *t* known erasures, or it can detect and correct combinations of errors and erasures. Reed–Solomon codes are also suitable as multiple-burst bit-error correcting codes, since a sequence of *b* + 1 consecutive bit errors can affect at most two symbols of size *b*. The choice of *t* is up to the designer of the code, and may be selected within wide limits.

Construction of Reed Solomon family of Codes

The Reed–Solomon code is actually a family of codes, where every code is characterized by three parameters: **an alphabet size *q*, a block length *n*, and**

a message length *k*, with *k* < *n* ≤ *q*. The set of alphabet symbols is interpreted as the **finite field of order *q*,** and thus, *q* has to be a prime power. In the most useful parameterizations of the Reed–Solomon code, the block length is usually some constant multiple of the message length, that is, **the rate *R* = *k*/*n* is some constant, and furthermore, the block length is equal to or one less than the alphabet size, that is, *n* = *q* or *n* = *q* – 1.**

Reed and Solomon's Orginal view: the Codeword as sequence of values

There are different encoding procedures for the Reed–Solomon code, and thus, there are different ways to describe the set of all codewords. In the original view of Reed & Solomon in early 1960's, every codeword of the Reed–Solomon code is a sequence of function values of a polynomial of degree less than *k*. In order to obtain a codeword of the Reed–Solomon code, the message is interpreted as the description of a polynomial *p* of degree less than *k* over the finite field *F* with *q* elements. In turn, the polynomial *p* is evaluated at *n<q distinct points* like *{a1, a2, a3……an} of field F.*

Common choices for a set of evaluation points include {0, 1, 2, ..., $n-1$}, {0, α, α^2, ..., α^{n-2}, 1}, {1, α, α^2, ..., α^{n-2}}, ... , where α is a primitive element of *F*.

Formally the set of codewords of Reed-Solomon can be defined as:

$$\mathbf{C} = \Big\{ \, \big(p(a_1), p(a_2), \ldots, p(a_n)\big) \, \Big| \, p \text{ is a polynomial over } F \text{ of degree } < k \, \Big\} .$$

Where **C** is the set of possible codewords.

While the number of different polynomials of degree less than *k* and the number of different messages are both equal to *q^k*, and thus every message can be uniquely mapped to such a polynomial, there are different ways of doing this encoding. The original construction of Reed & Solomon interprets the message *x* as the *coefficients* of the polynomial *p*, whereas subsequent constructions interpret the message as the *values* of the polynomial at the first *k* points *a1, a2, a3……ak* and obtain the polynomial *p* by interpolating these values with a polynomial of degree less than *k*. The latter encoding procedure, while being slightly less efficient, has the advantage that it gives rise to a systematic code, that is, the original message is always contained as a subsequence of the codeword.

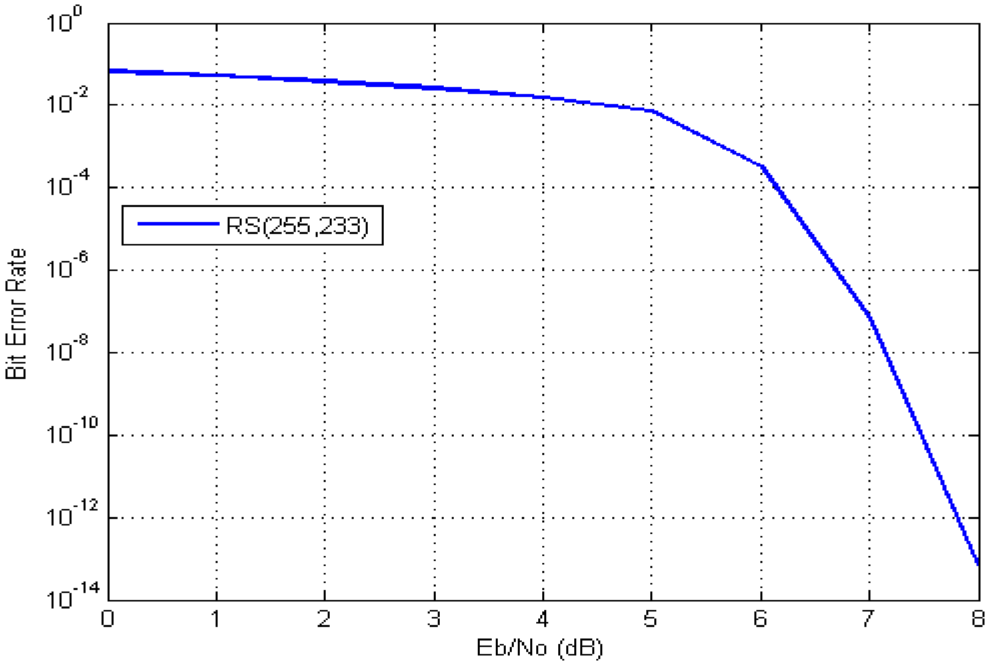


Fig (i): Theoritical BER of Reed-Solomon Codes

(ii)Cryptographic Hash Function

An illustration of the potential use of a cryptographic hash is as follows: Alice poses a tough math problem to Bob and claims she has solved it. Bob would like to try it himself, but would yet like to be sure that Alice is not bluffing. Therefore, Alice writes down her solution, computes its hash and tells Bob the hash value (whilst keeping the solution secret). Then, when Bob comes up with the solution himself a few days later, Alice can prove that she had the solution earlier by revealing it and having Bob hash it and check that it matches the hash value given to him before. (This is an example of a simple commitment scheme; in actual practice, Alice and Bob will often be computer programs, and the secret would be something less easily spoofed than a claimed puzzle solution).

The ideal cryptographic hash function has five main properties:

- ✓ It is deterministic so the same message always results in the same hash
- ✓ It is quick to compute the hash value for any given message
- ✓ It is infeasible to generate a message from its hash value except by trying all possible messages
- ✓ A small change to a message should change the hash value so extensively that the new hash value appears uncorrelated with the old hash value
- ✓ It is infeasible to find two different messages with the same hash value

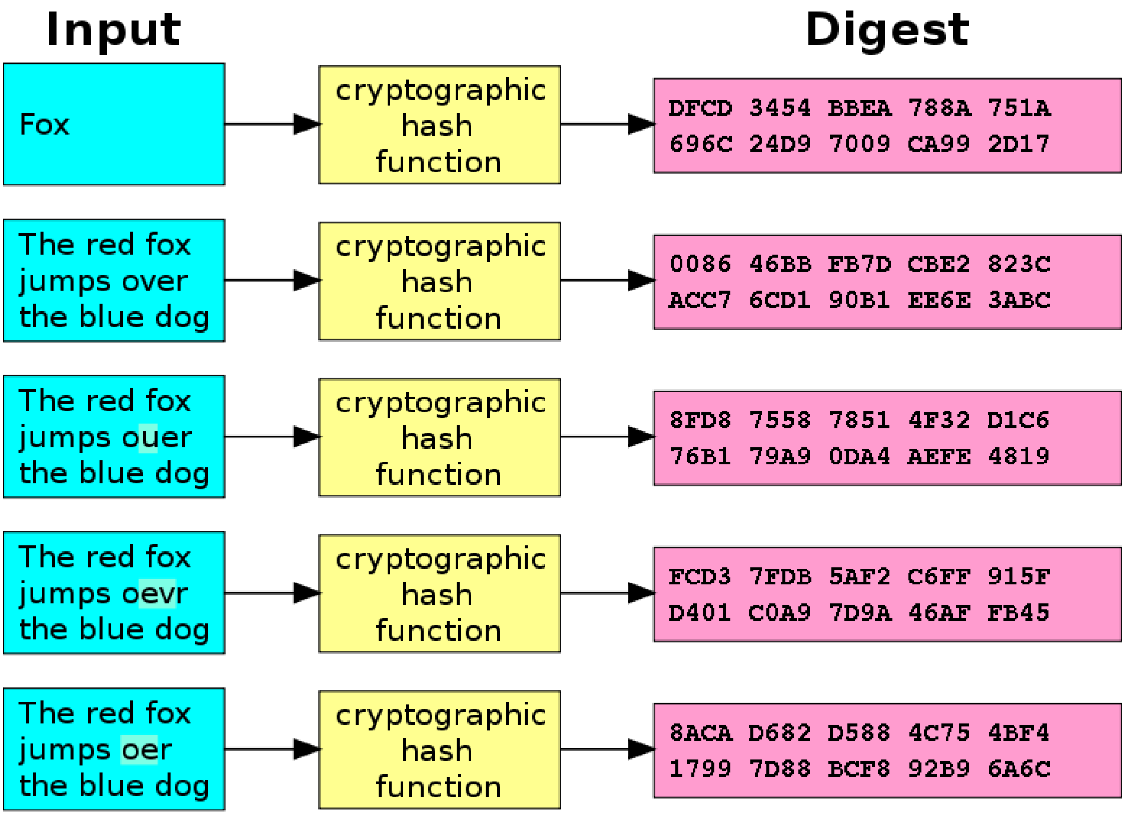

Input
Digest
Fox
cryptographic hash function
DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17
The red fox jumps over the blue dog
cryptographic hash function
0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC
The red fox jumps ouer the blue dog
cryptographic hash function
8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819
The red fox jumps oevr the blue dog
cryptographic hash function
FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45
The red fox jumps oer the blue dog
cryptographic hash function
8ACA D682 D588 4C75 4BF4 1799 7D88 BCF8 92B9 6A6C

Fig (ii): Cryptographic Hash Function block diagram

The above figure illustrates the avalanche phenomena, that a small change in input will drastically effect the output.

References

[1]Beginning JavaScript, Book by Paul Wilton and Jeremy McPeak

[2]Introducing the QR Code, Book by Sansweet Judith

[3]https://github.com/schmich/instascan

[4]https ://www.tutorialspoint.com/javascript/

[5] http://thruglassxfer.com/