International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 18 (2015) pp 39138-39144

Current ISBN System – Learning & Practicing via Simulation

A.AHMAD[1], M. A. K. RIZVI[1], N. MOHANAN[1] and S. AHMAD[2]

[1]Department of Electrical and Computer Engineering, College of Engineering,
Sultan Qaboos University, PO Box 33, Zip code 123, OMAN
[2]Departmentof Electrical and Computer Engineering, Cockrell School of Engineering,
University of Texas at Austin, TX, 78705, USA
afaq@squ.edu.om http://www.squ.edu.om/electrical/tabid/2673/language/en-US/Default.aspx

Abstract
International Standard Book Number (ISBN) is a numeric code which is used to uniquely identify books internationally. The International Organization for Standardization (ISO) initially used a 10-digit system which was updated to 13-digit system since 2007. The new code format led to the mass updating of the old code. It is important for the general public to understand the concept of ISBN. There is also a demand for tools for verification, prediction, generation, error detection and correction of the code. This paper explains the concept of ISBN and also demonstrates a tool for the functions mentioned.

Key-Words:-ISBN-10, ISBN-13, Error Check, Check Digit, ISO, Simulated, Library, Book Industry.

1 Introduction

Since the dawn of the digital age and globalization, it has become a non-deniable necessity to identify every major product uniquely on an international level. Since then, numerous coding techniques have been developed to implement in various fields. May it be the Universal Product Coding, Vehicle Identification Number, the credit card number or the International Standard Book Number(ISBN), coding has been employed to identify every item uniquely [1] – [5].

As computer technology started to flourish by the 1960s, publishers started implementing a new numbering system for better trade. W. H. Smith, the largest book retailer in Great Britain, created the Standard Book Numbering (SBN) system which was designed in 1966 and came into practice in 1967.Concurrently, The International Organization for Standardization (ISO) investigated and adapted the SBN system for international use, forming the International Standard Book Number (ISBN) as an ISO standard in 1970, which became ISO 2108 [6], [7].It is unique to each book which is printed above the bar code and on the copyright page. This helps the publishers and the book trade to associate a book to its bibliographic data, in recording and maintaining the stock levels and to electronically tracking the worldwide sales of a particular book.

Errors in reading, writing and transmission of codes are an integral part of the coding process and research around the world is continuously involved in finding various techniques to detect and correct these errors to produce a more reliable and fault tolerant system. It may create an interest among the users of published materials to understand, appreciate and apply the knowledge of ISBN in their daily lives. This paper takes the reader through the past and the present of ISBN. A simple tool for the users to validate, generate and convert ISBN codes is created and demonstrated.Furthermore, a vital goal hoping to be achieved by developing this ISBN tool is to add one more educational tool in the basket of our developed tools [3], [8]-[22] for the industry as well as for teaching purposes.

The ISBN has assigned unique codes for every member country and each of these countries, in a similar manner, assigns a unique code to each of its publishers. The publisher then follows wit assigning an exclusive code, depending on the digits available, to each kind of book it publishes. Thus, each version of a book gets its own unique ISBN, which may contain the information about its language, country of origin, publisher and the serial number.

1.1 ISBN-10

The ISBN has assigned unique codes for every member country and each of these countries, in a similar manner, assigns a unique code to each of its publishers. The publisher then follows wit assigning an exclusive code, depending on the digits available, to each kind of book it publishes. Thus, each version of a book gets its own unique ISBN, which may contain the information about its language, country of origin, publisher and the serial number.

The ISBN-10 code is structured into four parts [3], [5], [23]. They are:

i. Registration Group Element (RGE) – It denotes the language, the region or the country code of the book. Its length may be 1 to 5 digits long.
ii. Registrant Element (RE) – Denotes the Publisher code ad is generally assigned by the country in which the publisher resides. Its length may be 1 to 7 digits long.
iii. Publication Element (PE) – This is the book code assigned by the publisher. The range of this code depends on the digits allotted to that particular publisher. It may range from a single digit to 6 digits.
iv. Check Digit (CD) – It is the last digit of the ISBN code. It is used to verify the ISBN of the given book. It uses the concept of modulo 11.

The Tables 1, 2 and 3 below show a short list of codes of the three parts namely RGE, RE and PE which construct ISBN-10 [23]. Furthermore, Fig. 1 demonstrates a sample of the four components RGE, RE, PE and CD [3].

Table 1: Registration Group Element (RGE)

Category	Group	RGE
Language	English	0, 1
	French	2
	German	3
Region	Caribbean Community	976
	International NGO Publishers & EC Organizations	92
	Russian Federation and former USSR	5
Country	India	81, 93
	Japan	4
	Oman	99969
	South Korea	11, 89

1.2 ISBN-13

ISBN-13 is an extension from its 10-digit counterpart, the ISBN-10 format. As the implementation of ISBN increased for more than three decades, the ISBN-10 was approaching its maximum limit of allocation. To avoid this overflow, the ISBN code had to be upgraded to a 13-digit system. The check-digit is calculated using modulo 10. Every ISBN-10 code can be converted to ISBN-13 by just adding a prefix of 978 and recalculating the check-digit. This system has been implemented since January 1, 2007 [3], [24]. The prefix of 979 is being used for some new publications and they cannot be converted to ISBN-10 anymore [3], [25], [26]. Other combinations of prefixes are reserved for later use.

Figure 2 is meant to depict and demonstrate a sample of the of ISBN-13 code to explain each of components of the code [3].

Table 2: SomeRegistrant Elements (RE)

Group	RGE	Available (RE)
English Language	0	00-09 100-399 4000-5499 55000-86979 869800-998999 9990000-9999999
Caribbean Community	976	0-3 40-59 600-799 8000-9499 95000-99999
India	81	00-19 200-699 7000-8499 85000-89999 900000-999999

Table 3: SomeRegistrant Elements (RE)

Group	One of the Prefix (RGE + RE)	PE per Publisher Availability
English Language	0-9990000	10
Caribbean Community	976-0	10,0000
India	81-7000	1000

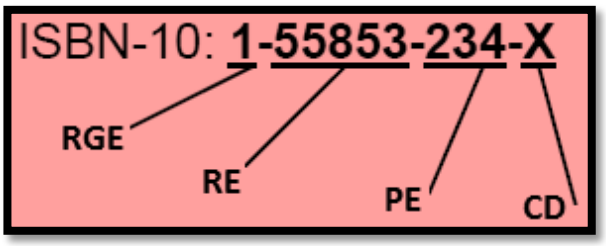

Figure 1: Anatomy of ISBN-10

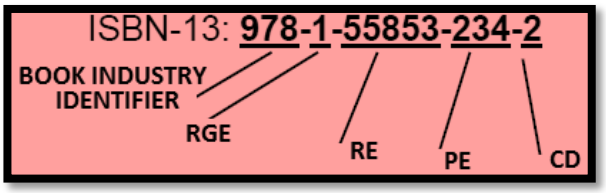

Figure 2: Anatomy of ISBN-13

2 Modular Arithmetic& Check Digit

2.1 Modular Arithmetic

There are many ways to calculate CDs. The simplest and commonly used CD schemes exploit the code number itself as part of a modular arithmetic operation [3].

Modular arithmetic involves a simple representation of a division process. Mainly it works a lot with the remainders generated by division.

For example, if number37 (N) is divided by number 8 (D), the remainder is 5 (R). Using modular arithmetic notation, this can be written 37 = 5(mod 8). Similarly, 49 = 4(mod 9) and 63 = 8(mod 11) and in general, if N-R is a multiple of the integer D,then this division process can be expressed as such given by Eq. (1).

$$N = R \ (mod \ D) \quad (1)$$

This means that 37=5(mod 8) =-3(mod 8) since 37-(-3) = 40 = 8 × 5.

The simplest CD schemeuses the code number itself as part of a modular arithmetic operation. For example, take a code number, N, create a CD, R, for this number when N is divided by the number D such that as given in Eq. (1).

If we wanted to use modulus 9 (mod 9) and we have the number 23456, our CD would be 2 and the number would from then be written as 234562.

Therefore, if we were to write the code number down incorrectly, we might discover that this incorrect number, N', did not fit the criterion that:

$$N' = R(mod \ D) \quad (2)$$

Or in other way round if the CD was written incorrectly, then R' would not fit the criterion that:

$N = R'(mod\ D)$ (3)

and so it would be detected that an error had occurred. However, since our number might be several digits long, trying to work out where the error was would probably be impossible. Thus this simple scheme cannot be utilized in error correction.

For example, if our number 234562, was wrongly written down as 235462, it would be seen that 23546=/=2(mod 9) but that does not tell us where the error has occurred because even if we know that only one error has occurred, we do not know if the penultimate or final (check) digit was incorrect.

2.2 ISBN-10

ISBN-10 uses a mod-11 technique for its CD calculation. The number consists of 10 digits, 9 of which are the code for the book and the last digit is the check digit. Consider the 9 digits of an ISBN code to be $d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9$. The following algorithmic procedure calculates the CD [3]. The multiplying weights with respective digits d_i are shown in Table 4.

Table 4: Multiplying factors for digit d_i

Digit	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9
x	x	x	x	x	x	x	x	x	x
Weight	1	2	3	4	5	6	7	8	9

$$Sum = (1xd_1 + 2xd_2 + 3xd_3 + 4xd_4 + 5xd_5 + 6xd_6 + 7xd_7 + 8xd_8 + 9xd_9) \quad (4)$$

$$sum = c\ mod(11) \quad (5)$$

$$Check\ Digit = c \quad (6)$$

As it is observed above, each digit in the code is given differing weights. Thus ISBN-10 can detect all single errors.

2.3 ISBN-13

ISBN-13 is an extension from its 10-digit counterpart, the ISBN-10 format. As the implementation of ISBN increased for more than three decades, the ISBN-10 was approaching its maximum limit of allocation. To avoid this overflow, the ISBN code had to be upgraded to a 13-digit system. The check-digit is calculated using modulo 10. Every ISBN-10 code can be converted to ISBN-13 by just adding a prefix of 978 and recalculating the check-digit. This system has been implemented since January 1, 2007 [3], [9]. The prefix of 979 is being used for some new publications and they cannot be converted to ISBN-10 anymore [10], [11]. Other combinations of prefixes are reserved for later use.

Consider an ISBN-13 code in which the 12 digits are represented using $d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 d_{10} d_{11} d_{12}$.

The algorithm used to calculate the check-digit isdescribed below. The multiplying weights with respective digits d_i are shown in Table 5.

Table 5: Multiplying factors for digit d_i

d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}
X	x	x	x	x	x	x	x	x	x	x	x
1	3	1	3	1	3	1	3	1	3	1	3

$$\text{Sum} = (1d_1 + 3d_2 + 1d_3 + 3d_4 + 1d_5 + 3d_6 + 1d_7 + 3d_8 + 1d_9 + 3d_{10} + 1d_{11} + 3d_{12}) \quad (7)$$

$$\text{sum} = 10 - c \bmod(10) \quad (8)$$

$$\text{Check Digit} = 10 - c \quad (9)$$

As it is observed in the relations above, there are several similarities to the modulo-11 algorithm discussed previously. Each digit in the code is given a different weight. Thus ISBN-13, just like ISBN-10, can detect all single errors.

3 Simulated Tools

3.1 ISBN-10 to ISBN-13 Converter

ISBN-13 was created by keeping in mind its compatibility to the ISBN-10 format. As mentioned in the earlier section, ISBN-13 is just an extension to the ISBN-10 system. As an example, consider the ISBN-10 code, 81-7450-494-X (Chemistry Part I: Textbook for Class XI, NCERT). The following steps may be followed to convert it to the ISBN-13 format [3], [9]. Figure 3 depicts the developed and implemented simulation model for converting ISBN-10 to ISBN-13.

- The CD 'X' is discarded and the remaining 9 digits are kept intact.
- The prefix of 978 is added to the 9 digit code to obtain the new 12 digit code. i.e. 978-81-7450-494.
- The check-digit is calculated from the new 12 digit code using the algorithm defined through Equations (7) – (9)..

$$\text{Sum} = 1*9 + 3*7 + 1*8 + 3*8 + 1*1 + 3*7 + 1*4 + 3*5 + 1*0 + 3*4 + 1*9 + 3*4 \quad (10)$$

$$\text{Sum} = 136 \quad (11)$$

$$\text{Sum} = 136 = 6 \bmod(10) \quad (12)$$

$$\text{Check Digit} = 10 - 6 = 4 \quad (13)$$

Thus the complete ISBN-13 code for ISBN-10, 81-7450-494-X is now formed by adding the check-digit at the end i.e. 978-81-7450-494-4.

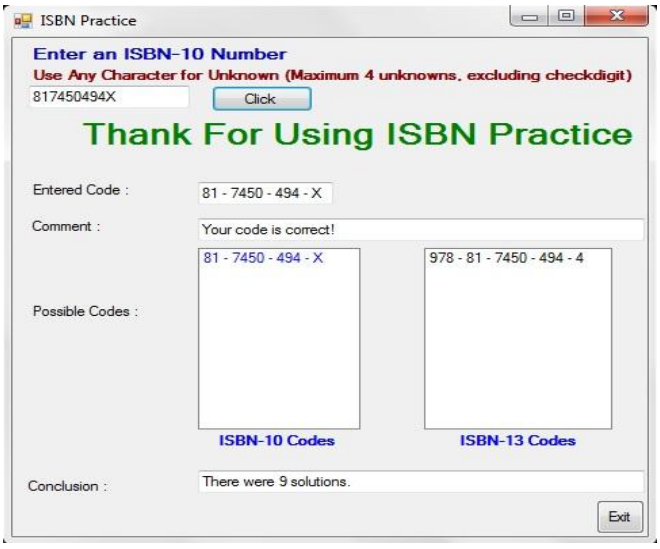

Fig 3: ISBN-10 code ISBN-13 code conversion and code verification tool

3.1.1 Examples

A few more examples are verified and demonstrated below.

Example 1:

An ISBN code for a book shown in Fig. 4 is verified as follows.

Sum = 84

CD = 10 – 84 (mod 10) = 6

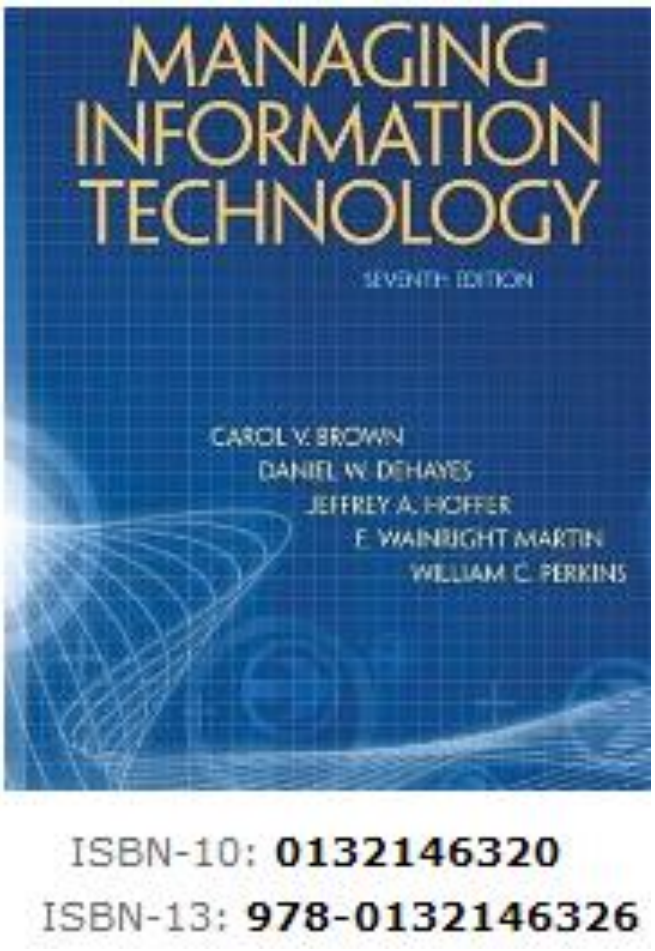

Fig 4: Example 1

Example 2:

An ISBN code for a book shown in Fig. 5 is verified as follows.

Sum = 73

CD = 10 – 73 (mod 10) = 7

Fig 5: Example 2

3.2 Code Verification Tool

Verification of the ISBN code is done by calculating the Sum of products and then finding the check digit,as explained above through Equations (4)-(6) for ISBN-10 and Equations (7)-(9) for ISBN-13. This check digit can be compared to the received check digit to verify the code. Thus error can easily be detected. The simulated code verification tool is shown in Fig.3.

3.3 Error Detecting andCorrecting Tool

Error is the wandering or straying away from the actual data to create a mistake. During transmission of data, errors are very common. The general transmission errors and their abundance are given in Table 6 [4,5]. A check digit is included with the data to detect an error and correct the received data. Error detection is the technique which implements this using various algorithms. Modular Arithmetic technique is the check digit techniques used in ISBN.

Table 6: Transmission errors and their abundance

Error Type	Error	Percentage
Single bit error	x → y	60 – 90 %
Adding a digit	xy → xxy	10 – 20 %
Deleting or Omitting a digit	xy → xy	10 – 20 %
Transposition error	xy → yx	10 – 20 %
Twin error	xx → yy	0.5 – 1.5 %
Jump train error	wyz → xyw	< 1 %
Phonetic error	13 → 30	0.5 – 1.5 %

Hence let us also consider an ISBN code that has been received, but may have one or more errors mentioned in Table 6. So a tool is required to calculate and match the check digit and then validate the code after correction.

3.3.1 Transposition Error

Now consider an ISBN-10 code in which a transposition error has occurred [6]. Let us consider the scenario where the fourth and the fifth digits have been exchanged. Hence the code word is –

$$d_1 d_2 d_3 d_5 d_4 d_6 d_7 d_8 d_9 \quad (14)$$

Let the sum of the actual code word be denoted by R and the sum of the received code word is C.

$$R = (1 * d_1 + 2 * d_2 + 3 * d_3 + 4 * d_4 + 5 * d_5 + 6 * d_6 + 7 * d_7 + 8 * d_8 + 9 * d_9) \quad (15)$$

$$C = (1 * d_1 + 2 * d_2 + 3 * d_3 + 4 * d_5 + 5 * d_4 + 6 * d_6 + 7 * d_7 + 8 * d_8 + 9 * d_9) \quad (16)$$

The two are subtracted to obtain the difference between the two. If there was no transposition, the difference would give zero.

$$R - C = (4 * d_4 + 5 * d_5) - (4 * d_5 + 5 * d_4) \quad (17)$$

$$R - C = (d_5 - d_4) \quad (18)$$

As each number here is a single digit, the range of the result is between-10 and 10. In this range, the only multiple of 11 is 0. Thus,

$$(d_5 - d_4) = 0 \quad (19)$$

$$d_5 = d_4 \quad (20)$$

This shows that a transposition error is undetected only when both digits are equal, else the transposition error can always be detected [4].

Detection of other errors like double errors is almost impossible in the ISBN scheme. For a single digit miss, the digit can be computed using the coding logic. If more than one digit is missed in the code then, using the coding logic, the range of the corrected code can be predicted.

3.3.2 Single Bit Error and Computing Missing Digit

The ISBN system is capable of detecting single errors in its code. It can also regenerate a code if one of its digits is missing. An example is considered below and Figure 6 shows a demonstration of using the developed simulated tool to check and correct the error.

Consider the code

$$d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 d_{10} \quad (21)$$

$$\text{Sum} = d_{10} \bmod (11) \quad (22)$$

Let the digit at the n^{th} position be an unknown. Then,

$$\text{Sum}_1 + d_n * m_n = d_{10} \bmod (11) \quad (23)$$

where,

Sum_1 = Sum of the products of known terms

m_n = multiplier for n^{th} term (24)

Since Sum_1 and m_n are known, d_n can be calculated and the ISBN code can be reconstructed.

Example 3:

As an example, consider the code 81-7450-494-X. Let's consider the case where the 5^{th} digit is missing. i.e. the code is 81-74□0-494-X.Here, the unknown digit is calculated using the sum of the known product terms.

$$\text{Sum} = 183 + 5x = 10 \bmod (11) \quad (25)$$

$$7 \bmod (11) + 5x = 10 \bmod (11) \quad (26)$$

$$5x = 3 \bmod (11) \quad (27)$$

Thus,

$$x = 5 \quad (28)$$

Figure 6 depicts the developed simulated tool for handling single bit error, missing bit recovery and possible transposition error correction.

3.3.3 Handling of Multi-bitMissing Digits

Let there be two digits which areunknown in the code. An example which is considered below is demonstrated through the developed tool depicted in Fig. 7.

Example 4:

Consider the ISBN code 81-7450-494-X. Let us consider the case where the 5^{th} and the 7^{th} digits are missing. i.e. the code is 81-74□0-□94-X.

$$\text{Sum} = 155 + 5x + 7y = 10 \bmod (11) \quad (29)$$

$$1 \bmod (11) + 5x + 7y = 10 \bmod (11) \quad (30)$$

$$5x + 7y = 9 \, mod (11) \quad (31)$$

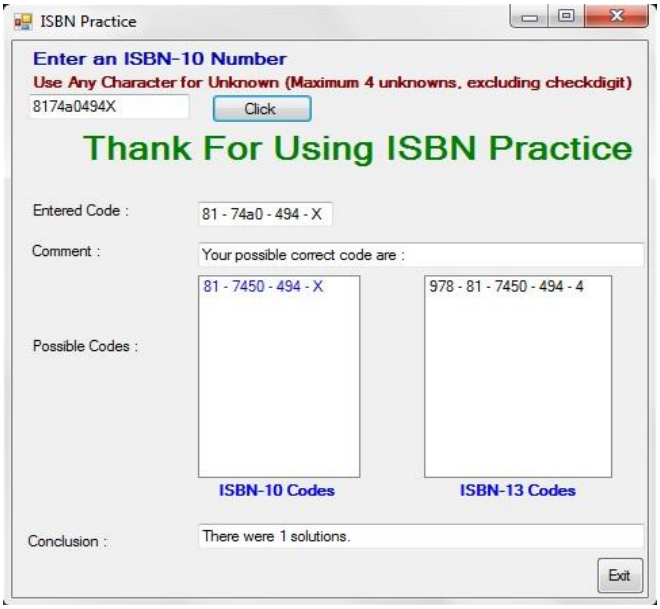

Fig 6: Single digit error, missing digit recovery and possible transposition error correction tool

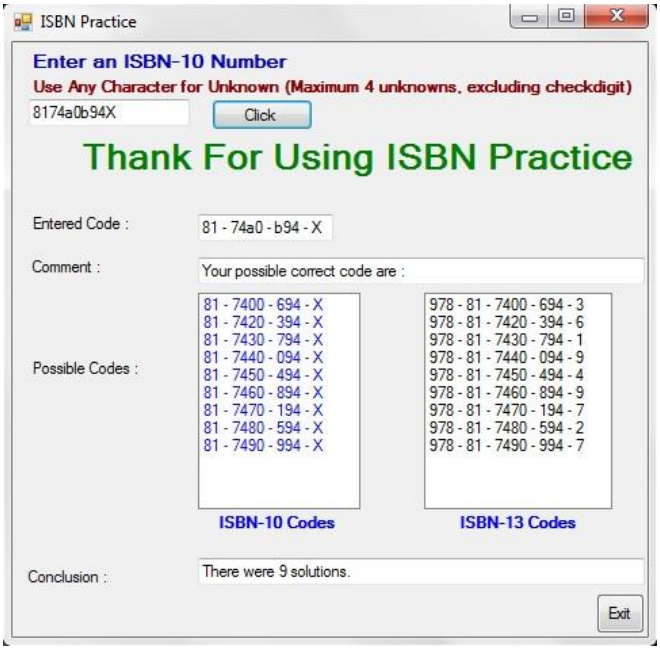

Fig 7: 2-digit missing code recovery tool

As the relationship implies in the ISBN code, many different combinations of digits are possible for the given unknown digits. Hence it is not possible to find out the exact single code word, but it is possible to predict all possible codes under the given known conditions. As explained in the sections above, similar algorithms can be developed in order to predict 3, 4 or even 5 digits in the case there are 5 unknowns (missing digits). Shown in Fig. 8 is the tool developed to demonstrate how to recover the 5 missing digits.

It is an important task for the publisher to know what combinations of the ISBN code it uses for the various books under its publications. Thus there is a requirement for a tool which can generate all the possible combinations of codes once the limits are provided to it.

The technique described in the section above and the developed tool depicted in Fig. 8 can be implemented by publishers to generate with ease new ISBN codes for their new publications' series. The publisher can simply keep the allotted code digits as known and then generate all possible code digits for theunknown code digits which would satisfy the ISBN concept and algorithms.

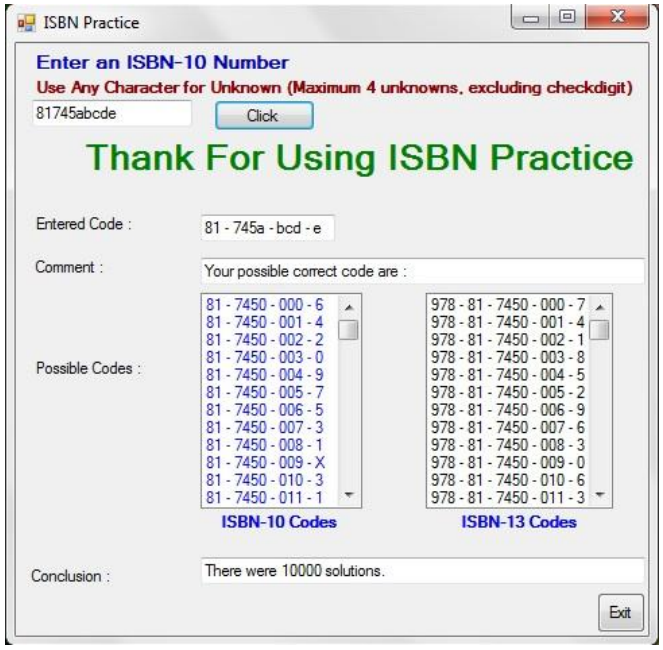

Fig 8: 5-digit missing code recovery tool

4 Conclusion

ISBN identifies every published material uniquely and has helped the concerned people with identification, organization, trade and tracking of books since four decades now. Initially, the system used for three decades had to be finally updated before an overflow problem surfaced. This led to mass conversions of all earlier versions. Just like every other coding technique, ISBN may also be susceptible to various errors and thus the check digit using modulo technique is applied. In this paper the background algorithms have been developed and programmed in Visual C++ to create a simple tool to verify, rectify, predict and generate ISBN-10 and ISBN-13 code. This tool can potentially help teachers, students, enthusiasts and publishers to practice, learn, understand and apply the ISBN knowledge.

The results of many ISBN codes have been verified. This tool can be used by publishers to generate codes and libraries andbook traders can utilize it to verify the codes. Additionally it can aid educational institutes to learn and teach the ISBN system which is of minimal complexity, the level of which enables the general public (or anyone interested) to easily understand it also. The tool can be developed further by adding the algorithm to break the code into its four parts and identify the language, country, publisher details etc. It is capable of predicting and/or generating up to 5 unknown code digits. This could further be developed to generate code even when more than 5 code digits are unknown.

Acknowledgement:

The authors would like to express their great appreciations and gratitude to Sultan Qaboos University, Sultanate of Oman for providing research facilities, technical supports and research environment.

References:

[1] A. Ahmad, S. Ahmad., A. Al-Habsi and M. Al-Hinai, Understanding Universal Product Code-A Study and Simulation Experiments, Indian Journal of Industrial and Applied Mathematics, vol. 5, no. 1, 2014, pp. 25-34.

[2] A. Ahmad, How To Learn And Simulate Universal Product Code-An Information Redundancy Technique, Information Systems for Sustainable Business Development (Book Chapter), Universal publisher (Brown Walker Press, USA), 2013, pp.185-188.

[3] A. Ahmad, Are We Equipped To Update Our Knowledge Due To Change of the New ISBN System, Proceedings International Conference of Education Technology 2008 (ICOET2008), Oman, March 02-05, 2008, pp. 1-9.

[4] H. Parks, G. Musser, L. Trimpe, V. Maurer, R. MaurerLaung-Terng, A Mathematical View of Our World, Thomson Brooks/Cole Publishing Company, 2007.

[5] A. Ahmad, Secrets Of Error Detection Scheme Of ISBN System, Journal of Engineering, vol. 1, no. 1, 2003, pp. 10-14.

[6] ISO 2108:2005 Information and documentation-International standard book number (ISBN), (accessedAugust 18, 2015). http://www.capub.cn/shtmsl/bzgf/images/2011/04/06/41E0FD94737B892B7FA50E3263B3D682.pdf

[7] ISBN History, (accessedAugust 18, 2015). http://www.isbn.org/ISBN_history

[8] A. Ahmad, D. Ruelens, and S. Ahmad, Development of verification tool for minimal Boolean equation, IEEE Technology and Engineering Education, vol. 8, no. 4, pp. 29-34, 2013.

[9] A. Ahmad, A.,Development Realization of a Better Signature Analysis Scheme by Adding a Bit to the Size of 8k", International Journal of Information Engineering, vol. 3, no. 4, pp. 122-128, 2013.

[10] A. Ahmad, S. S. Al-Busaidi, M. J. Al-Mushrafi, On Properties of PN Sequences Generated by LFSR-a Generalized Study and Simulation Modeling, Indian Journal of Science and Technology, vol. 10, no. 10, pp. 5351-5358, 2013.

[11] A. Ahmad and D. Ruelens, Development of digital logic design teaching tool using MATLAB & SIMULINK" IEEE Technology and Engineering Education, vol. 8, no.1, pp. 7-11, 2013.

[12] A. Ahmad, D. Abri, and S. S. Al-Busaidi. Adding Pseudo-Random Test Sequence Generator in the Test Simulator for DFT Approach, Computer Technology and Applications, vol. 3, no. 7, pp. 463-470, 2012.

[13] A. Ahmad, and D. Al-Abri, Design of a Pseudo-Random Binary Code Generator via a Developed Simulation Model, ACEEE International Journal on Information Technology, vol. 2, no. 1, pp. 33-36, 2012.

[14] A. Ahmad, How to Learn and Simulate Universal Product Code-an Information Redundancy Technique, Proceedings of The International Information Systems Conference iiSC (2011), held at Sultan Qaboos University, Muscat, Oman, October 11-12, 2011, pp. 200-203,

[15] A. Ahmad, A Simulation Experiment on a Built-In Self Test Equipped with Pseudorandom Test Pattern Generator and Multi-Input Shift Register (MISR), ACEEE International Journal of VLSI Design & Communication Systems vol. 1, no. 4, pp. 1-12, 2010.

[16] A. Ahmad, D. Al-Abri, Design of an Optimal Test Simulator for Built-In Self Test Environment, Journal of Engineering Research, vol. 7, no. 2, pp. 69–79, 2010.

[17] A. Ahmad, D. Al-Abri and M. M. Al-Ramhi, Design of an E-Learning Process in the Area of Digital System Testing, Proceedings International Conference on Distance Education (ICODE2006), Oman, March 27 – 31, 2006, pp 174-184

[18] A. Ahmad, A. M. J. Al-Lawati, and A. M. Al-Naamany, Identification of test point insertion location via comprehensive knowledge of digital system's nodal controllability through a simulated tool, Asian Journal of Information Technology, vol. 3, no. 3, pp. 142–147, 2004

[19] A. Al-Lawati and A. Ahmad, Realization of a simplified controllability computation procedure – A MATLAB-SIMULINK based tool, Journal for Scientific Research-Science and Technology, vol. 8, pp. 131–143, 2003

[20] A. Ahmad, M. J. Al-Musharafi and S. Al-Busaidi, Study and implementation of properties of m-sequences in MATLAB-SIMULINK – A pass / fail test tool for designs of random generators, Journal of Scientific Research – Science and Technology, vol. 7, part 1, pp. 147–156, 2002

[21] A. Ahmad, M. J. Al-Musharafi and S. Al-Busaidi, Design and study of a strong stream crypto-system model for e-commerce, International Council for Computer Communication Publishers, Washington DC, USA, vol. 1, pp. 619–630, 2002

[22] A. Ahmad, M. J. Al-Musharafi and S. Al-Busaidi, Study and implementation of properties of m-sequences in MATLAB-SIMULINK – A pass / fail test tool for designs of random generators,IEEE / IEE International conference on communication, computer and power, ICCCP'01, Oman, pp. 191-196

[23] ISBN Ranges, (accessedAugust 18, 2015). http://www.isbn-international.org/page/ranges

[24] Z. Wykes, ISBN-13 For Dummies – Special Edition, Wiley Publishing Inc., 2005.

[25] Biloxi,The Transition To ISBN-13, State Instructional Materials Review Association (FormerlyNational Association of State Textbook Administrators), 2005, (accessed August 18, 2015). www.bmibook.org/wp-content/uploads/.../ISBN-13-NASTA-1-3.ppt

[26] About the ISBN Standard, (accessed August 18, 2015). http://www.isbn.org/about_ISBN_standard