

電子發票證明聯一維及二維條碼規格說明

版本：1.9

財政部財政資訊中心

中華民國 111 年 5 月

修訂表：

版本	日期	修改摘要
Ver 1.0	2011/11/29	初版
Ver 1.1	2011/12/20	新增營業人申請流程，刪除 QR Code 部分規格限制
Ver 1.2	2012/02/01	新增常見問題、參考用原始碼、一二維條碼檢查表
Ver 1.3	2012/04/24	修訂常見問題
Ver 1.4	2012/05/09	修訂常見問題、修訂部分文字敘述
Ver 1.5	2016/07/04	修訂文件名稱、條碼規格、營業人申請流程、常見問答及部分文字敘述
Ver 1.6	2016/11/15	修訂加密元件，增加代表店、總公司統一編號說明及部分文字調整
Ver 1.7	2018/05/11	二維條碼規格部分文字及範例圖檔調整
Ver 1.8	2018/08/22	配合統一發票使用辦法第 9 條修正規定，因應境外電商營業人應記載事項調整，單價、金額及總計得以外幣列示。第貳章、二維條碼規格增加「三、境外電商營業人規格」
Ver 1.9	2022/05/10	配合電子發票實施作業要點就買受人為營業人時，交易明細應增列銷售額及稅額之規定，即買受人為營業人，自當無銷售系統無法分離稅項之情事，爰修正國內營業人二維條碼規格之銷售額，若買受人為非營業人且銷售系統無法順利將稅項分離計算者，始得以 00000000 記載銷售額。

目錄

第壹章、一維條碼規格	4
第貳章、二維條碼規格	5
第參章、營業人申請流程	10
第肆章、加密元件	11
第伍章、參考用原始碼	13
第陸章、常見問題	15
附錄、一維及二維條碼檢查表	16

第壹章、一維條碼規格

一、用途：

記載於電子發票證明聯，提供兌領獎時輸入使用。

二、規格：

一維條碼應以三九碼 (Code 39) 記載，記載事項應含發票年期別、發票字軌號碼及隨機碼總計 19 碼。詳細內容如下：

1. 發票年期別 (5)：記錄發票字軌所屬民國年份(3 碼)及期別之雙數月份(2 碼)，例如 104 年 3-4 月發票年期別記載為「10404」。
2. 發票字軌號碼 (10)：記錄發票完整 10 碼號碼。
3. 隨機碼 (4)：記錄發票 4 碼隨機碼。

三、範例：



圖 1: 一維條碼範例

本範例條碼資料為：
10404UZ176908720122

第貳章、二維條碼規格

一、用途：

記載於電子發票證明聯，提供行動應用時讀取發票資訊及資訊防偽用途。

二、國內營業人規格：

二維條碼應以 QR Code 記載，數量二個，以左右並列、水平(上緣)對齊、大小一致方式配置，編碼區長寬各 1.5 公分以上，其周圍應至少各保留 0.2 公分 ($\pm 10\%$) 以上之空白處。記載事項含發票字軌號碼、發票開立日期 (年月日)、4 碼隨機碼、銷售額、總計額、買方統一編號、賣方統一編號、加密驗證資訊、營業人自行使用區、二維條碼記載完整品目筆數、該張發票交易品目總筆數、中文編碼參數、品名、數量、單價、補充說明。詳細內容如下：

(一) 左方二維條碼記載事項：

1. 發票字軌號碼 (10)：記錄發票完整 10 碼字軌號碼。
2. 發票開立日期 (7)：記錄發票 3 碼民國年份 2 碼月份 2 碼日期共 7 碼。
3. 隨機碼 (4)：記錄發票 4 碼隨機碼。
4. 銷售額 (8)：記錄發票未稅總金額總計 8 碼，將金額轉換以十六進位方式記載。若買受人為非營業人且銷售系統無法順利將稅項分離計算，則以 00000000 記載，不足 8 碼左補 0。
5. 總計額 (8)：記錄發票含稅總金額總計 8 碼，將金額轉換以十六進位方式記載，不足 8 碼左補 0。
6. 買方統一編號 (8)：記錄發票買受人統一編號，若買受人為一般消費者則以 00000000 記載。
7. 賣方統一編號 (8)：記錄發票賣方統一編號。
8. 加密驗證資訊 (24)：將發票字軌號碼 10 碼及隨機碼 4 碼以字串方式合併後使用 AES 加密並採用 Base64 編碼轉換，AES 所採用之金鑰產生方式請參考第參、肆章及「加解密 API 使用說明書」。

以上欄位總計 77 碼。下述資訊為接續以上資訊繼續延伸記錄，且每個欄位前皆以間隔符號“.”(冒號)區隔各記載事項，若左方二維條碼不敷記載，則繼續記載於右方二維條碼。

9. 營業人自行使用區 (10 碼)：提供營業人自行放置所需資訊，若不使用則以 10 個“*”符號呈現。
10. 二維條碼記載完整品目筆數：記錄左右兩個二維條碼記載消費品目筆數，以十進位方式記載。
11. 該張發票交易品目總筆數：記錄該張發票記載消費品目總筆數，以十進位方式記載。
12. 中文編碼參數 (1 碼)：定義後續資訊的編碼規格，若以：
 - (1) Big5 編碼，則此值為 0
 - (2) UTF-8 編碼，則此值為 1

(3) Base64 編碼，則此值為 2

- 13.品名：商品名稱，請避免使用間隔符號“(冒號)於品名。
- 14.數量：商品數量，以十進位方式記載。
- 15.單價：商品單價，以十進位方式記載。
- 16.補充說明：非必要使用資訊，營業人可自行選擇是否運用，於左右兩個二維條碼已記載所有品目資訊後，始可使用此空間。長度不限。

(二) 右方二維條碼記載事項：

1. 右方二維條碼前兩碼起始符號 (2 碼)：首 2 碼固定以 “**” 為起始符號，供未來讀取端辨識左方或右方二維條碼之用。
2. 接續左方二維條碼不敷記載之中文編碼後資訊。

左方二維條碼使用 QR Code V6 (41x41) (含)以上版本，並採用 Level L (容錯率 7%)以上之防錯標準。

三、境外電商營業人規格：

二維條碼應以 QR Code 記載，數量二個，以左右並列、水平(上緣)對齊、大小一致方式配置，編碼區長寬各 1.5 公分以上，其周圍應至少各保留 0.2 公分 ($\pm 10\%$)以上之空白處。記載事項含發票字軌號碼、發票開立日期 (年月日)、4 碼隨機碼、銷售額、總計額、買方統一編號、賣方統一編號、加密驗證資訊、營業人自行使用區、二維條碼記載完整品目筆數、該張發票交易品目總筆數、中文編碼參數、品名、數量、單價、補充說明。詳細內容如下：

(一) 左方二維條碼記載事項：

1. 發票字軌號碼 (10)：記錄發票完整 10 碼字軌號碼。
2. 發票開立日期 (7)：記錄發票 3 碼民國年份 2 碼月份 2 碼日期共 7 碼。
3. 隨機碼 (4)：記錄發票 4 碼隨機碼。
4. 銷售額 (8)：境外電商以 00000000 記載。
5. 總計額 (8)：境外電商以 00000000 記載。
6. 買方統一編號 (8)：記錄發票買受人統一編號，若買受人為一般消費者則以 00000000 記載。
7. 賣方統一編號 (8)：記錄發票賣方統一編號。
8. 加密驗證資訊 (24)：將發票字軌號碼 10 碼及隨機碼 4 碼以字串方式合併後使用 AES 加密並採用 Base64 編碼轉換，AES 所採用之金鑰產生方式請參考第參、肆章及「加解密 API 使用說明書」。

以上欄位總計 77 碼。下述資訊為接續以上資訊繼續延伸記錄，且每個欄位前皆以間隔符號“(冒號)區隔各記載事項，若左方二維條碼不敷記載，則繼續記載於右方二維條碼。

9. 營業人自行使用區 (10 碼)：提供營業人自行放置所需資訊，若不使用則以 10 個“*”符號呈現。
10. 二維條碼記載完整品目筆數：記錄左右兩個二維條碼記載消費品目筆數，以十進位方式記載。
11. 該張發票交易品目總筆數：記錄該張發票記載消費品目總筆數，以十進位方式記載。
12. 中文編碼參數 (1 碼)：定義後續資訊的編碼規格：境外電商 UTF-8 編碼，此值為 3
13. 境外電商銷售額(10)：記錄發票未稅總金額總計 10 碼，其中固定 8 碼整數與固定 2 碼小數，將 8 碼整數與 2 碼小數分別轉換以十六進位方式記載。因境外電商銷售電子勞務予境內自然人，應以訂價開立統一發票，以 0000000000 記載。
14. 境外電商總計額(10)：記錄發票含稅總金額總計 10 碼，其中固定 8 碼整數與固定 2 碼小數，將 8 碼整數與 2 碼小數分別轉換以十六進位方式記載。不足固定碼左補 0。

例如：總金額為 363.99

(1) 整數 363 以十六進位轉換為 16B，不足 8 碼左補 0 為 0000016B

(2) 小數 99 以十六進位轉換為 63

(3) 總金額記載內容為 0000016B63

15. 品名：商品名稱，請避免使用間隔符號 “:”(冒號)於品名。
16. 數量：商品數量，以十進位方式記載。
17. 單價：商品單價，以十進位方式記載。
18. 補充說明：非必要使用資訊，營業人可自行選擇是否運用，於左右兩個二維條碼已記載所有品目資訊後，始可使用此空間。長度不限。

(二) 右方二維條碼記載事項：

1. 右方二維條碼前兩碼起始符號 (2 碼)：首 2 碼固定以 “**”為起始符號，供未來讀取端辨識左方或右方二維條碼之用。
2. 接續左方二維條碼不敷記載之中文編碼後資訊。

左方二維條碼使用 QR Code V6 (41x41) (含)以上版本，並採用 Level L (容錯率 7%)以上之防錯標準。

四、範例：

- (一) 以 Big5 編碼為例：



二維條碼範例(Big5 編碼)

本範例左方二維條碼資料為：

AB112233441020523999900000144000001540000000001234567ydXZt4LAN1U
HN/j1juVcRA==:*****:3:3:0:乾電池:1:105:

右方二維條碼資料為：

**口罩:1:210:牛奶:1:25

(二)以 UTF-8 編碼為例：



圖 3: 二維條碼範例(UTF-8 編碼)

本範例左方二維條碼資料為：
AB112233441020523999900000144000001540000000001234567ydXZt4LAN1U
HN/j1juVcRA==:*****:3:3:1:乾電池:1:105:

右方二維條碼資料為：
**口罩:1:210:牛奶:1:25

(三) 以 Base64 編碼為例：



圖 4: 二維條碼範例(Base64 編碼)

本範例左方二維條碼資料為：
AB112233441020523999900000144000001540000000001234567ydXZt4LAN1U
HN/j1juVcRA==:*****:3:3:2:5Lm+6Zu75rGg

右方二維條碼資料為：
**OjE6MTA1OuWPo+e9qToxOjlxMDrniZvlpbY6MToyNQ==

五、補充說明：

1. 考慮行動器材鏡頭之判讀能力，須符合可自動對焦之三百萬畫素機型可辨識二維條碼為設計基準。
2. 左方二維條碼之加密驗證資訊為強化 4 碼隨機碼之安全機制，減低被偽造風險。
3. 針對消費品目明細在部分情況下（例如：折扣、點數扣抵…等）無[數量]資訊下，則該品目可不記錄於二維條碼中。
4. 品目數係指呈現商品的項目數，而非各個商品數量的加總。
5. 如需 QRcode 加密工具，請至電子發票整合服務平台\常用文件下載項下，下載「電子發票 QRCode 說明文件(含 JRE)」檔。
6. B2B 發票的隨機碼(4 位)請用 4 位空白。

第參章、營業人申請流程

- 一、開立電子發票之賣方營業人管理者連線至電子發票整合服務平台，於「密碼及種子管理(QRCode)」功能下使用憑證或種子密碼登入設定加密種子密碼；接著，營業人於終端透過AES金鑰產生工具將加密種子密碼輸入，產出營業人加密金鑰。透過此金鑰，搭配公開之AES演算法即可將隨機碼及發票字軌號碼產製正確的QRCode加密驗證資訊共24碼。
- 二、營業人應至電子發票整合服務平台「QRCode解密驗證」功能下，輸入QRCode字串線上解密驗證，以確保其產製加密驗證資訊之正確性。
- 三、以此做法加密資料能確保每家營業人能管理自己的金鑰，且演算法是公開的；兼具安全性以及管理彈性。
- 四、詳細平台操作流程及金鑰產生工具之使用說明，請參閱「加解密API使用說明書」。

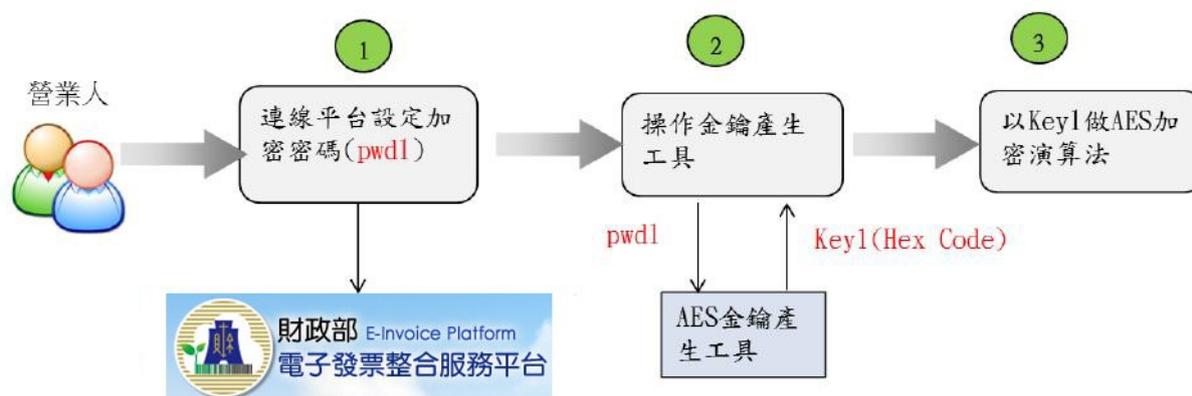


圖 1: 營業人產生 QRCode 加密元件之作業流程

第肆章、加密元件

一、做法：

考量營業人資訊能力以及應用廣度需求，提供多組加密元件(VB、.NET、Windows Visual C/C++、JAVA 以及 Linux 元件) 協助營業人進行系統建置，API 介面為求未來 QR Code 規格進行調整時，營業人僅需更換元件檔，不需針對程式重寫，在設計時介面要求所有發票中資訊皆予載入。

二、函式原型：

String QRCodeINV(String InvoiceNumber, String InvoiceDate, String InvoiceTime, String RandomNumber, int SalesAmount, int TaxAmount, int TotalAmount, String BuyerIdentifier, String RepresentIdentifier, String SellerIdentifier, String BusinessIdentifier, String** ProductArrays, String AESKey)

三、函式回應：

以字串方式回應 77 碼 QRCode 字串，營業人僅需將此 QRCode 結果送至印表機。

四、函式參數說明：

1. InvoiceNumber：以字串方式載入發票字軌號碼共 10 碼。
2. InvoiceDate：以字串載入發票開立年月日(中華民國年份月份日期)共 7 碼。
3. InvoiceTime：發票開立時間 (24 小時制) 共 6 碼，以時時分分秒秒方式之字串載入。
4. RandomNumber：以字串方式載入 4 碼隨機碼。
5. SalesAmount：以整數方式載入銷售額 (未稅)，若無法分離稅項則記載為 0。
* 請注意不得開立零元發票或負數發票
6. TaxAmount：以整數方式載入稅額，若無法分離稅項則記載為 0。
7. TotalAmount：以整數方式載入總計金額(含稅)。
8. BuyerIdentifier：以字串方式載入買受人統一編號，若買受人為一般消費者，請填入 00000000 8 碼字串。

9. RepresentIdentifier : 以字串方式載入代表店統一編號，電子發票證明聯二維條碼規格已不使用代表店，請填入 00000000 8 碼字串。

10. SellerIdentifier : 以字串方式載入銷售店統一編號。

11. BusinessIdentifier : 以字串方式載入總機構統一編號，如無總機構請填入銷售店統一編號。

12. ProductArrays : 單項商品資訊

ProductArrays 中包含產品的陣列 (ProductArray)，此產品陣列應包含：

i. Product Code : 以字串方式記載透過條碼槍所掃出之條碼資訊。

ii. Product Name : 以字串方式記載商品名稱。

iii. ProductQty : 以字串方式記載商品數量。

iv. ProductSaleAmount : 以字串方式載入商品銷售額 (整數未稅)，若無法分離稅項則記載為字串 0。

v. ProductTaxAmount : 以字串方式載入商品稅額(整數)，若無法分離稅項則記載為字串 0。

vi. ProductAmount : 以字串方式載入商品金額(整數含稅)。

13. AESKey : 以字串方式記載加密金鑰之 HEX 值。

第五章、參考用原始碼

檔案: tw\gov\nat\invoice\qrutil\QREncrypter.cs

```
namespace tw.gov.nat.invoice.qrutil {
using System ; using System.IO ; using System.Security.Cryptography ; using System.Text ;
public class QREncrypter {
public string AESEncrypt(string plainText , string AESKey) {
byte[] bytes = Encoding.Default.GetBytes(plainText) ;
ICryptoTransform transform = new RijndaelManaged { KeySize = 0x80 , Key =
this.convertHexToByte(AESKey) , BlockSize = 0x80 , IV =
Convert.FromBase64String("Dt8lyToo17X/XkXaQvihuA==") }.CreateEncryptor() ;
MemoryStream stream = new MemoryStream() ; CryptoStream stream2 = new CryptoStream(stream ,
transform , CryptoStreamMode.Write) ; stream2.Write(bytes , 0 , bytes.Length) ;
stream2.FlushFinalBlock() ; stream2.Close() ; return Convert.ToBase64String(stream.ToArray()) ;
}
private byte[] convertHexToByte(string hexString) {
byte[] buffer = new byte[hexString.Length / 2] ; int index = 0 ; for (int i = 0 ; i < hexString.Length ; i += 2) {
int num3 = Convert.ToInt32(hexString.Substring(i , 2) , 0x10) ; buffer[index] = BitConverter.GetBytes(num3)
[0] ; index++ ;
}
return buffer ;
}
private void inputValidate(string InvoiceNumber , string InvoiceDate , string InvoiceTime , string
RandomNumber , decimal SalesAmount , decimal TaxAmount , decimal TotalAmount , string
BuyerIdentifier , string RepresentIdentifier , string SellerIdentifier , string BusinessIdentifier , Array[]
productArray , string AESKey)
{ if (string.IsNullOrEmpty(InvoiceNumber) || (InvoiceNumber.Length != 10)) {
throw new Exception("Invalld InvoiceNumber: " + InvoiceNumber) ; }
if (string.IsNullOrEmpty(InvoiceDate) || (InvoiceDate.Length != 7)) {
throw new Exception("Invalld InvoiceDate: " + InvoiceDate) ; }
try {
long num = long.Parse(InvoiceDate) ; int num2 = int.Parse(InvoiceDate.Substring(3 , 2)) ; int num3 =
int.Parse(InvoiceDate.Substring(5)) ; if ((num2 < 1) || (num2 > 12)) {
throw new Exception() ;
}
if ((num3 < 1) || (num3 > 0x1f)) {
throw new Exception() ;
}
}
catch (Exception)
{
throw new Exception("Invalld InvoiceDate: " + InvoiceDate) ; }
if (string.IsNullOrEmpty(InvoiceTime)) {
throw new Exception("Invalld InvoiceTime: " + InvoiceTime) ; }
if (string.IsNullOrEmpty(RandomNumber) || (RandomNumber.Length != 4)) {
throw new Exception("Invalld RandomNumber: " + RandomNumber) ; }
if (SalesAmount < 0M)
```

```

{
throw new Exception("Invaild SalesAmount: " + SalesAmount) ; }
if (TotalAmount < 0M)
{
throw new Exception("Invaild TotalAmount: " + TotalAmount) ; }
if (string.IsNullOrEmpty(BuyerIdentifier) || (BuyerIdentifier.Length != 8)) {
throw new Exception("Invaild BuyerIdentifier: " + BuyerIdentifier) ; }
if (string.IsNullOrEmpty(RepresentIdentifier)) {
throw new Exception("Invaild RepresentIdentifier: " + RepresentIdentifier) ; }
if (string.IsNullOrEmpty(SellerIdentifier) || (SellerIdentifier.Length != 8)) {
throw new Exception("Invaild SellerIdentifier: " + SellerIdentifier) ; }
if (string.IsNullOrEmpty(BusinessIdentifier)) {
throw new Exception("Invaild BusinessIdentifier: " + BusinessIdentifier) ; }
if ((productArray == null) || (productArray.Length == 0)) {
throw new Exception("Invaild ProductArray") ; }
if (string.IsNullOrEmpty(AESKey))
{
throw new Exception("Invaild AESKey") ; }
}
public string QRCodeINV(string InvoiceNumber , string InvoiceDate , string InvoiceTime , string
RandomNumber , decimal SalesAmount , decimal TaxAmount , decimal TotalAmount , string
BuyerIdentifier , string RepresentIdentifier , string SellerIdentifier , string BusinessIdentifier , string[][]
productArray , string AESKey)
{ try
{ this.inputValidate(InvoiceNumber , InvoiceDate , InvoiceTime , RandomNumber , SalesAmount ,
TaxAmount , TotalAmount , BuyerIdentifier , RepresentIdentifier , SellerIdentifier , BusinessIdentifier ,
productArray , AESKey) ;
}
catch (Exception exception)
{
throw exception ; }
return ((InvoiceNumber + InvoiceDate + RandomNumber + Convert.ToInt32(SalesAmount).ToString("x8") +
Convert.ToInt32(TotalAmount).ToString("x8") + BuyerIdentifier + SellerIdentifier) +
this.AESEncrypt(InvoiceNumber + RandomNumber , AESKey).PadRight(0x18)) ;
}}
}

```

第陸章、常見問題

一、QR Code 加密元件提供包含那些版本?提供之版本無法適用如何處理?

答: QR Code 加密元件版本包含 VB、Windows Visual C/C++、.NET dll、JAVA 或 Linux 元件。若業者之設備不支援上述元件，另已提供元件之 C#/C++ 原始碼，業者可依此規格撰寫對應程式符合自身需求。

二、承第一題，若選擇自行開發程式，該注意些甚麼?如何驗證程式的正確性?

答: 業者自行開發程式，請務必符合第五章中函式原型規定，以避免未來條碼規格修改，造成程式修改困難。

驗證程式正確性主要分為三部分，第一部分是程式碼正確，第二部分是金鑰正確，第三部分是電子發票整合服務平台可正確解析；驗證程式碼正確性應與提供之範例程式進行比較(提供 .NET、JAVA 及 Windows Visual C/C++ Sample Project)，金鑰正確性應至平台進行解析。業者完成程式後，請產生四組 QR Code 字串(77 碼)，至平台進行驗測。

三、QR Code 除大小限制外有何規定?是否有建議方式?

答: 業者採用 QR Code 編碼區長寬各 1.5 公分以上，其周圍應至少各保留 0.2 公分 ($\pm 10\%$) 以上之空白處。且左方二維條碼應使用 QR Code V6 (41x41) (含) 以上版本，並採用 Level L (容錯率 7%) 以上之防錯標準。

四、QR Code 加密元件中，金鑰(AESKey) 及加密種子密碼(pwd1) 關係為何?

答: 為避免 AES 金鑰於網路上傳遞，增加外流風險，業者至電子發票整合服務平台登入後，輸入加密種子密碼，平台即能知悉業者之金鑰；業者在本地端透過財政資訊中心提供之軟體輸入加密種子密碼，亦能取得此金鑰，平台與業者即可不透過網路傳輸之模式下取得金鑰。

五、同一營業人所使用之 AESKey 是否每張發票都一樣，各分店是否亦是一樣?

答: AES 金鑰為同一業者使用一組，不同分店得使用總機構所產生之金鑰或自行產製金鑰，同一營業人開立發票應使用同一組，請業者務必小心保存避免外流，建議可每 6 個月更換 QRCode 種子密碼並重新產製 AES 金鑰。

六、QR Code 加密元件中，函式參數 RepresentIdentifier(代表店統編)是什麼?該填入甚麼資訊?

答: 電子發票證明聯二維條碼規格已不使用代表店，請填入 00000000 字串。

七、提供之元件該如何產生 QR Code 圖檔?

答: 僅提供資訊加密程式，QR Code 圖檔須由業者自行負責。

八、業者上線前該進行什麼檢查?

答: 業者上線前，應對一、二維條碼可讀性進行自我檢查，自我檢查方法請參考附錄一維及二維條碼檢查表。

附錄、一維及二維條碼檢查表

一、營業人自我檢查一維及二維條碼正確性:

- 連線至網址: <http://goo.gl/1jllS>
- 上傳電子發票證明聯掃描檔，Barcode Reader Settings 選擇「Linear」，並點選「Decode Image」，確認一維條碼掃描結果為：
 - Code 39
 - 19 碼含年期別/字軌號碼/4 碼隨機碼

Barcode Reader & Decoder Software

Home > OnLine Barcode Decoder

■ OnLine Barcode Decoder

Show result as ▼

Barcode Image file 1 張照片

File limit: 2Mb
Files: jpg, gif, png, bmp, tiff (including multipage)

Barcode Reader Settings

Linear QRCode DataMatrix PDF417 AztecCode

Code-128 Code-39 Codabar Interleaved 2/5 Industrial 2/5 Code 93 Code 11

EAN-13 EAN-8 UPC-A UPC-E DataBar (RSS-14) DataBar Limited

DataBar Stacked DataBar Expanded DataBar Expanded Stacked

Linear Unrecognized

Quantity: Speed: ▼

圖：一維條碼檢查頁面

- 營業人於 Barcode Reader Settings 選擇「QRCode」，並點選「Decode Image」，確認回應於瀏覽器前 77 碼字元類似於
QQ00081524100080139660000014000000141234567828433892qk90D8qgCwuEvOngCZ
aEdaw=

- 營業人採用財政資訊中心提供之 dll 檔案，僅須確認字軌是否正確對應，其他部分不必另行檢查。若自行開發則應依規格進行確認。

三、營業人自我檢查資料可讀性:

- 若營業人本身擁有解析度足夠之行動工具，建議可安裝 Quickmark 軟體進行行動載具讀取 QR Code。
- Android 程式: <http://goo.gl/aHw9q>

➤ iPhone 程式: <http://goo.gl/jzb3F>

- 另營業人應至少透過自身擁有之條碼設備，進行一維條碼檢測。