

國立成功大學

電信管理研究所

碩士論文

以區塊鏈技術實作動態頻譜管理及接取平台

**Implementation of Blockchain-Based Dynamic
Spectrum Management and Access Platform**

研究生：黃瀚緯

指導教授：陳文字 博士

中華民國一百零八年六月

國立成功大學

碩士論文

以區塊鏈技術實作動態頻譜管理及接取平台
Implementation of Blockchain-Based Dynamic Spectrum
Management and Access Platform

研究生：黃瀚緯

本論文業經審查及口試合格特此證明

論文考試委員：

林易泉 林福林

沈宗麟

陳文字

指導教授：陳文字

系(所)主管：陳勁甫

中華民國 108 年 6 月 19 日

摘要

隨著資通訊產業的日趨進步與快速發展，越來越多裝置設備都有連接上網路的需求，因此對於頻譜資源的需求有增無減，然而頻譜資源是有限的，要如何有效利用目前就已經十分壅塞的頻譜資源，是不論電信業者、政府機關、資通訊產業業者都面臨到的難題。為了能夠讓頻譜資源的使用更加有效率、頻譜資源更易管理，本研究利用擁有環境偵測能力的感知無線電 CR (cognitive radio) 技術，在軟體定義無線電平台 GNU Radio 下進行開發，實作一個動態頻譜接取系統，並建立一個符合 IEEE 國際標準之動態頻譜接取系統，同時也將區塊鏈技術融合至動態頻譜接取平台中的頻譜交易機制當中。本篇研究以感知無線電裝置結合軟體定義無線電技術，並將裝置連接上建立在樹莓派 (Raspberry Pi3) 之動態頻譜接取資料庫，讓感知無線電裝置能夠在發射資料前先藉由自身環境感測能力偵測周遭頻譜使用狀況，透過軟體定義無線電設備 USRP (universal software radio peripheral)，並調整自身發射的參數，連接資料庫後由資料庫進行最終的干擾判斷進而同意裝置之發射請求。本研究同時在動態頻譜接取系統中實作頻譜交易的功能，並將能夠確保交易安全的區塊鏈技術融入頻譜交易當中，利用區塊鏈分散式帳本的架構驗證交易的安全性，讓整個動態頻譜接取系統更加完善且安全，以符合未來頻譜資源使用的需求。

關鍵字：動態頻譜接取系統、區塊鏈、感知無線電、軟體定義無線電、GNU Radio、USRP、頻譜資源資料庫、頻譜交易

Implementation of Blockchain-Based Dynamic Spectrum Management and Access Platform

Hang-Wei, Huang

Wen-Tzu, Chen

Institute of Telecommunication Management,
National Cheng Kung University

SUMMARY

With the rapid development of the information communication technology, more and more devices have their demands on the spectrum resource to access the internet service. However, spectrum resource is limited. It's a big challenge for mobile operator, manufacture, and government to make spectrum usage more efficient. In order to manage the spectrum resource efficiently, this study aims to build a dynamic spectrum management and access platform according to the IEEE 1900 standards. Also, this study includes spectrum transaction which is implemented with blockchain technology into the spectrum management platform. Raspberry Pi 3 Model B equipped with USRP (universal software radio peripheral) is used in this study as a CR (cognitive radio) device. In dynamic spectrum access scheme, a CR device would sense the radio environment before transmitting data in order to avoid possible interference. If the device concludes that other devices nearby are using the certain spectrum resource, the device itself will dynamically adjust its key parameters such as carry frequency and modulation scheme. Spectrum management database plays another significant row in this study. When the database receives a transmitting request from CR device, it must determine whether interference happens. To make the spectrum transaction system safe and sound, blockchain technology is employed in this study. Blochchain is resistant to modification of each transaction record, so it can assure the safety of whole system without any malicious attack and simultaneously verify every transaction in the dynamic spectrum management and access platform.

KeyWords: Dynamic Spectrum Access, Software-Defined Radio, GNU Radio, Cognitive Radio, Blockchain, Database, Spectrum Transaction

INTRODUCTION

With the development of the information communication technology, Internet of things comes in reality and brings humanity a whole new world. Every device can access the Internet and has a variety of applications which only depend on our imagination. Nevertheless, spectrum resource is so crowded for each device to access. Spectrum resource is no doubt limited. No one can deny the importance of making a much more efficient usage of precious spectrum resource. Practically, spectrum resource is highly governed by the related authority. Using spectrum resource without permission is illegal. However, there are still a lot of so called spectrum holes due to the inefficient use of the spectrum resource. As a result, dynamic spectrum access plays a significant role in improving the efficiency of spectrum access. A CR (cognitive radio) device can dynamically adjust its communication parameter to meet the local radio environment without any interference. Furthermore, a complete dynamic spectrum access platform must support a robust mechanism for the user who doesn't own the license to have an opportunity to access the spectrum resource via transactions. In this scenario, blockchain-based transaction mechanism can provide the users in need with a hack-free spectrum transaction platform. Users can be able to easily transact their spectrum license with other users on this platform. In this way, the spectrum authority can lower their cost to maintain the dynamic spectrum management and access platform due to the blockchain technology.

MATERIALS AND METHODS

According to the IEEE 1900 standards, plenty of CR devices which are connected to spectrum database continuously seek for opportunity to access the spectrum resource. In this study, several Raspberry Pi3s are used to implement the whole dynamic spectrum management and access platform. Raspberry Pi3 is a single-board computer based on Linux operating system. We can make a lot of things with it. As shown in figure 1, a cognitive radio device is implemented with software-defined radio hardware, USRP B200, driven by a Raspberry Pi3. A development environment for software defined radio, GNU Radio, is used in this study. GNU Radio is a user-friendly platform where designers can adjust the cognitive radio device dynamically and easily. At the same time, this study uses another Raspberry Pi3 as a database server where the dynamic spectrum platform is established. Apache server, phpMyAdmin and MariaDB are used to build up the spectrum database server. Every data concerning the spectrum usage as well as spectrum transaction data is recorded in the database. CR device in this study can connect to the spectrum database we build via Internet. In this way, a dynamic spectrum management and access platform where we conduct our two experiments is implemented.

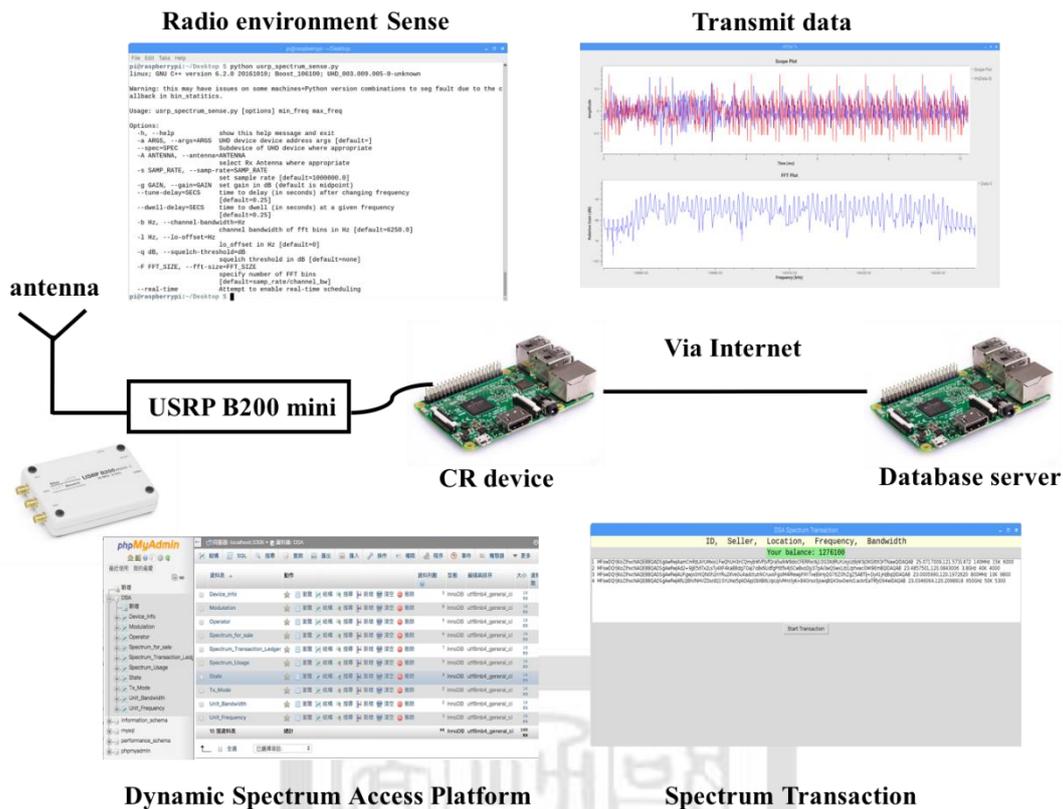


Figure 1. Architecture of the dynamic spectrum management and access platform

To implement a dynamic spectrum management and access platform, this study also adopts a graphical user interface to connect the cognitive radio device within the platform. In this study, we adopt 140 MHz as central frequency for our CR device. Moreover, there are two modulation scheme we can select in our experiment, GMSK and OFDM-QPSK. Before transmitting the data, cognitive radio device will sense radio environment to detect the radio waves nearby. If the power of the possible radio waves is relatively higher than the noise power, the device will not transmit radio wave. Otherwise, a transmission request will be sent to the database to obtain an approval. No device can transmit data without approval from database or it would be regarded as an illegal user. Besides the spectrum access scenario, spectrum transaction is also implemented with graphical user interface. User can sell or buy the spectrum resource on the platform without any safety concern under the protection of blockchain technology. When the transaction button is clicked, transaction occurs immediately and the hash value which is hashed by the SHA-256 function will be uploaded to the database. The hash value mentioned above is composed of the transaction detail, such as seller, buyer, previous hash value, central frequency, bandwidth, and timestamp. Timestamps and the previous hash value are critical information of a transaction due to their uniqueness. The transaction ledger can be easily certificated by checking the hash value of each transaction.

RESULTS AND DISCUSSION

Based on the research results, the CR device can dynamically adjust its communication parameters,

for example, central frequency, bandwidth, gain value, and modulation scheme to coordinate with radio environment. Through the adjustment, the CR device can successfully transmit or receive data according to the real demand. However, there is still a hidden or exposed terminal problem if we simply appeal to the ability of sensing environment of cognitive radio device. It's not enough to make the spectrum resource harmonious. Therefore, it is quite necessary to have a spectrum resource database to coordinate device to avoid any potential interference. On the other hand, distributed blockchain ledger can authenticate every spectrum transaction. Each transaction record is saved in the database and can be hashed by SHA-256 function which can be easily verified but hardly decoded. If its hash value doesn't match to the previous hash value or others ledger, we can conclude that there must be something wrong in the ledger, and then we must deny the ledger.

CONCLUSION

In this thesis, we implement the blockchain-based dynamic spectrum management and access platform by using Raspberry Pi3 which is connected to USRP B200 via USB port as a CR device. In addition to CR device, we implement a spectrum database with another Raspberry Pi3. CR device can dynamically adjust its communication parameter to transmit or receive data according to the current radio environment. We can conclude that CR device cannot use the spectrum resource in harmony with other devices without the central spectrum database. Spectrum resource can be comprehended with three dimensions which are individually frequency, time, and geographical location. And that is exactly the reason why there are so many spectrum holes in the current spectrum allocation scheme. Consequently, dynamic spectrum access is the solution to optimize current spectrum usage to fulfill the potential spectrum need in the near future. Spectrum resource database is a critical element in dynamic spectrum access scheme. We can regard the database as the brain of whole platform. With the help of this platform, related regulator can supervise the radio environment easily than before with less effort. Also, with the aid of blockchain technology, spectrum transaction or even spectrum sharing scheme would be much easier and safer than ever.

誌謝

台南是我的家，我愛台南，我愛成大，從成大交通管理科學系現在的電信管理研究所，我在這裡學到了很多，有太多太多的話想說，有太多太多的人需要感謝，我想大概只能謝天了吧，謝謝我遇到的一切，人生的酸甜苦辣，讓我不在原地踏步，讓我成為更棒的人，生而為人，我很幸福。

如果說我生而為成大電信管理研究所的學生，那陳文字老師便是我存在的原因，感謝我的指導教授陳文字老師，從大四錄取研究所的那一刻起，到最後口是完成這段期間，沒有陳文字老師便沒有現在的我，陳文字老師就好比學業上的父母一般，傳授我一切該學的知識，不論是學業上的各種理論，或者人生道路上的哲學，老師都傾囊相授，鄙人將終生難忘碩士這兩年與老師相處，與老師討論報告的日子。

除了陳文字老師，我還要感謝我的女友芳仔，在這兩年的碩士生涯當中讓我學到了很多，有歡笑，有淚水，也在我最需要陪伴的時候，在身旁陪伴著我，讓我克服諸多難關，迎向璀璨光明的未來，未來不知道還會遇到什麼挑戰，共勉之。同時也要感謝我同研究室的同學，棋宏及凱翔，幫我排除許多學業上的疑難雜症，當我這文組同學面臨障礙時，事實提點我，拉我一把，我想，我會懷念那些一起在研究室看日出的日子。

說到致謝，怎能少到了給我靈魂，給我軀體的父母呢，請原諒我的任性，原諒我的忤逆，原諒我的幼稚，沒有您們的辛勞付出，就沒有今日的我，如今的我已長大成人，您倆老可以放心的休息了，我會讓您們少擔心我一點，有我養，別怕。

瀚緯 謹誌

成功大學電信管理研究所

中華民國一零八年七月

目錄

表目錄.....	ix
圖目錄.....	x
第一章 緒論.....	1
1.1 研究背景.....	1
1.2 研究動機與目的.....	5
1.3 論文組織.....	7
第二章 文獻回顧探討.....	10
2.1 動態頻譜接取.....	10
2.1.1 軟體定義無線電.....	12
2.1.2 感知無線電.....	13
2.1.3 頻譜資源資料庫.....	14
2.2 區塊鏈概念及其特性.....	15
2.3 區塊鏈技術實作動態頻譜接取系統文獻回顧.....	19
2.4 文獻歸納小結.....	20
第三章 研究架構.....	23
3.1 軟體架構.....	23
3.1.1 GNU Radio.....	23
3.2 硬體裝置介紹.....	25
3.2.1 USRP 技術規格.....	25
3.2.2 樹莓派技術規格.....	26
3.3 感知無線電模組設計.....	28
3.3.1 傳送端設計.....	28
3.3.2 接收端設計.....	29
3.4 區塊鏈實作去中心化動態頻譜接取平台.....	30
3.4.1 頻譜租賃交易資訊.....	31
3.4.2 共享頻段地理資訊.....	32

3.4.3 感知無線電裝置資訊.....	33
3.5 系統整合架構.....	34
第四章 研究結果.....	35
4.1 平台架構與實驗流程.....	35
4.2 資料庫實作與設計.....	38
4.2.1 硬體及軟體使用.....	38
4.2.2 資料欄位設計.....	39
4.3 頻譜交易機制.....	42
4.3.1 區塊鏈概念實作.....	42
4.3.2 交易驗證.....	44
4.4 動態頻譜接取概念設計與實作.....	46
4.4.1 通道偵測.....	46
4.4.2 偵測通道後發射資料.....	50
4.4.3 資料庫干擾判斷.....	53
4.5 研究結果探討.....	56
第五章 結論.....	58
5.1 結論.....	58
5.2 未來研究建議.....	59
參考文獻.....	60

表目錄

表 1、USRP B200 mini 規格.....	26
表 2、Raspberry Pi 3 規格.....	27
表 3、頻譜租賃交易資訊.....	31
表 4、頻譜共享資訊.....	32
表 5、感知無線電設備裝置資訊	33
表 6、DSA 資料庫之資料表名稱及功能	40



圖目錄

圖 1、中華民國頻率分配表(特高頻、超高頻、極高頻).....	3
圖 2、動態頻譜接取系統整體架構圖	6
圖 3、動態頻譜接取模型(以使用頻段區分) [8].....	11
圖 4、動態頻譜接取模型(以發射功率區分) [8].....	11
圖 5、軟體定義無線電架構圖[4].....	12
圖 6、感知無線電系統架構[9].....	13
圖 7、分頻、分時之頻譜使用狀況示意圖 [10].....	14
圖 8、比特幣交易紀錄(擷取自區塊 No. 438995) [14]	17
圖 9、比特幣區塊資料(擷取自區塊 No. 438995 Hash 值部分)[14].....	18
圖 10、區塊鏈概念示意圖 [14].....	18
圖 11、GNU Radio 開發環境.....	24
圖 12、USRP B200 Mini [12]	26
圖 13、樹莓派實體照片.....	27
圖 14、收發之車載資訊.....	28
圖 15、感知無線電傳送端設計	29
圖 16、感知無線電接收端設計	30
圖 17、系統整合架構圖.....	34
圖 18、實驗硬體設備.....	35
圖 19、頻譜交易流程圖.....	36
圖 20、隱藏節點問題.....	37
圖 21、動態頻譜接取實驗流程	38
圖 22、phpMyAdmin 管理者介面.....	39
圖 23、主要使用之資料表.....	41
圖 24、主鍵所連接之資料表	41

圖 25、頻譜購買介面.....	42
圖 26、頻譜購買介面(購買確認彈跳視窗).....	43
圖 27、頻譜交易介面(交易成功).....	43
圖 28、驗證交易介面(正常情況).....	44
圖 29、驗證交易介面(異常狀況).....	45
圖 30、通道偵測程式之參數調整	47
圖 31、實際進行通道偵測資料(無偵測到訊號).....	48
圖 32、實際進行通道偵測資料(無偵測到訊號)頻譜圖	49
圖 33、實際進行通道偵測資料(有偵測到訊號).....	49
圖 34、實際進行通道偵測資料(有偵測到訊號)頻譜圖	50
圖 35、調變方式選擇之圖形化介面(準備畫面).....	51
圖 36、調變方式選擇之圖形化介面(發射後).....	52
圖 37、裝置發出傳送請求(GMSK)-同意	52
圖 38、裝置發出傳送請求(OFDM)-拒絕	53
圖 39、GMSK 方塊圖	54
圖 40、OFDM 方塊圖(1)	54
圖 41、OFDM 方塊圖(2)	55
圖 42、GMSK 訊號發射圖	55
圖 43、OFDM 訊號發射圖	56

第一章 緒論

1.1 研究背景

資通訊科技發展至今，從最初的第一代行動通訊到時至今日的第四代行動通訊，而第五代行動通訊在世界各地也如火如荼的發展當中，科技日新月異的發展，每天都可能有不同的新科技新技術被發展出來，多到數不清的科技產品以及創新應用也如雨後春筍般冒出來，將我們人類的生活昇華到前所未有的層次。就以智慧型手機來說，裡面的每一個應用程式不正代表著其取代我們傳統生活的某一項物品嗎，例如手機裡面的鬧鐘取代了傳統實體的鬧鐘、手機的時間取代傳統的時鐘或者手錶、手機裡的音樂功能取代了收音機以及 MP3 等音樂播放器、手機備忘錄功能取代了便條紙及記事本……等族繁不及備載，資通訊科技儼然已經滲透進我們生活當中。而要能夠成就如此資通訊科技的蓬勃發展，除了硬體設備如晶片、天線模組、儲存裝置……等，以及軟體開發如應用程式、驅動程式、作業系統……等，絕對少不了一大功臣，那便是「網際網路」，網際網路將原本人與人之間獨立的節點連接起來，讓這個世界更加多采多姿，網路讓我們可以縮短人與人之間的距離，讓國界消失，讓我們得以輕易的獲取許多資訊以及知識，越來越多人已經離不開有網際網路的生活，出門在外少了網際網路的陪伴便會覺得渾身不自在。

網際網路依照接取方式主要可以分為無線以及有線兩種，其中有線網路是指其利用實體物理介質透過銅軸纜線、雙絞線或者光纖等方式傳輸封包資訊。而無線網路並非是完全沒有實體有線網路，無線網路是透過無線通訊的方式讓有線實體網路得以延伸，利用電磁波傳遞資訊，一般而言生活中常見的無線通訊為手機以及基地台的溝通，或者手機透過行動熱點

Wi-Fi 連結上網際網路。若提及無線網路或者無線通訊，絕對不能不提及「頻譜」資源的重要性。

除了常見的網際網路服務可以透過無線通訊的方式傳送以外，其他許多服務如無線電接收器或者發收器、數位無線電視、或者是現今第四第五代行動通訊最熱門車物聯網的應用等，都仰賴著無線頻譜資源傳送封包資訊。頻譜資源並非任何人想用就可以用的，目前各國的頻譜規劃主要是參考國際電信聯盟(International Telecommunication Union, ITU)之無線電分配規則訂定並配合該國資通訊產業的需求，發放頻譜使用執照，將各個頻段分配給不同業者使用，以發揮頻譜使用之最大效益，頻譜的資源非常有限，但是人對於頻譜資源使用的慾望卻日益增加，有增無減。

根據國家通訊傳播委員會的頻譜資料庫查詢系統[1]，我們可以查詢到中華民國的頻率分配表，如圖 1 所示，以特高頻(VHF)、超高頻(UHF)、及極高頻(SHF)為例，不難發現現有的頻譜配置已經相當擁擠，氣象輔助、太空研究與作業、衛星系統、水上無線電助航以及航空無線電助航等，不論是政府軍事用途還是民間商業用途都需要使用到頻譜資源，因此，如何有效的管理頻譜，並將頻譜資源做最有效率以及最佳化的運用，同時能夠達到妥善地分配頻譜資源，並監控頻譜使用者以維持電波秩序，避免沒有執照的頻譜使用者非法接取該頻段傳送資訊，一直以來是各國政府通訊相關機關頭痛的問題。

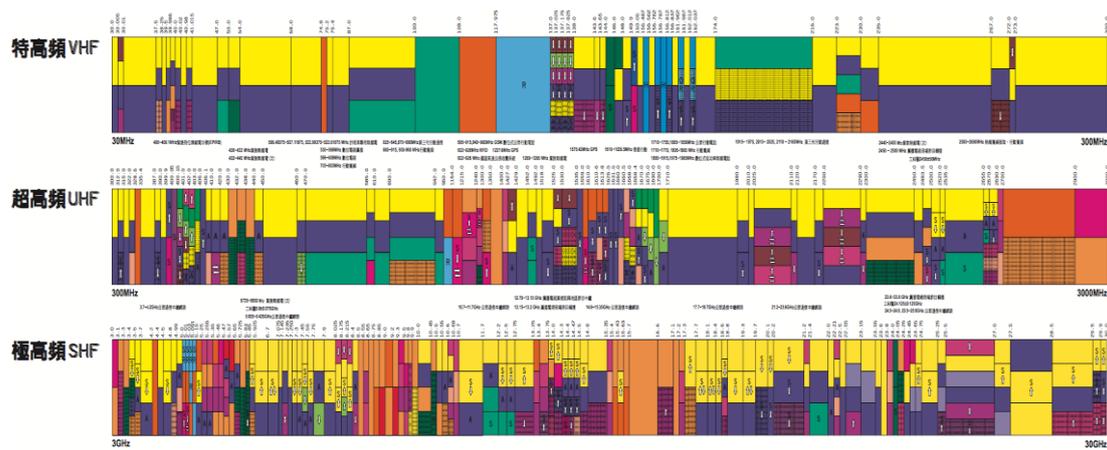


圖 1、中華民國頻率分配表(特高頻、超高頻、極高頻)

為了解決頻譜資源不足的問題，感知無線電[2][3](Cognitive Radio, CR)以及動態頻譜接取(Dynamic Spectrum Access, DSA)的概念於 1999 年由 Mitola 提出來，感知無線電是以軟體定義無線電[4][5](Software Defined Radio, SDR)的概念為基礎發展。軟體定義無線電讓傳統無線電設備只需保留前端天線收發設備，其餘元件如調變器、放大器、濾波器等都可以藉由軟體的方式實作其功能，收發的頻率、增益值、頻寬大小等都可以藉由軟體設定，感知無線電更進一步來說，可以透過內建的感知器，感測週遭的頻譜環境以調整自身的參數，動態頻譜接取正是取自感知無線電的核心概念，藉由感知無線電能夠感知環境的能力，持續偵測環境當中可以用的頻譜資源，當偵測到某頻段閒置未被使用時，感知無線電便可以調整其發射頻段到該閒置頻段已傳送或者接受資料，畢竟，某頻段執照擁有者不一定每分每秒都在傳送或接收資料，多數時間可能都是閒置的，無形之中造成頻譜資源的浪費，動態頻譜接取可以讓頻譜的使用更有效率，可以服務更多的使用者及用戶，動態頻譜接取的技術便能有效解決頻譜資源缺乏的問題，讓閒置的頻譜得以有效利用。

在頻譜的使用者當中，一般來說主要分為持有執照的合法使用者(Licensed User)，以及未持有執照的使用者(Unlicensed User)，沒有執照的

使用者在未經允許的情況下是不能使用頻譜資源的，未經授權使用及干擾是違法的行為，會受到主管機關裁罰。然而目前並沒有一個完善的頻譜監管機制以及一個公開透明的平台供頻段擁有者或者有頻譜需求者可以在上面買賣頻段的使用權，始終停留在只有執照擁有者可以使用頻譜區塊的時代，合法使用者同時也面臨著擁有某頻段使用權卻可能不常使用導致閒置的窘境。

區塊鏈(Blockchain)去中心化、保障資料安全、時間戳記等概念在 1990 年代便已經被提出來[6]，然而時至今日，其核心概念的實際應用非常多元，其中最著名的應用為虛擬貨幣「比特幣」(Bitcoin)的應用，比特幣這個詞可說是比區塊鏈一詞還熱門，若沒有區塊鏈的概念，絕不會有比特幣的誕生。區塊鏈的特性可以保障金錢交易在沒有第三方介入的情況且同時能保障交易安全，讓互相不信任的買賣雙方可以放心的進行貨幣交易，同時，區塊鏈可以保障上面資料的真實性以及透明性，所以資料一但上傳區塊鏈後便絕對無法竄改，區塊鏈上每一筆資料都是經過 RSA-256 演算法加密的，保證安全，且資料公開透明，任何人都可以看到區塊鏈上所有資料，不需特別跟某一個第三方中心化機構申請帳號密碼，任何人都可以簡單地進入區塊鏈的系統使用上面的功能。區塊鏈上面的一切是由所以使用者共同維護的，不受任何第三方中心化機構控制，因此也不必擔心該中心化機構受到惡意駭客攻擊導致資料憑空消失，甚至進階一點，我們可以在區塊鏈上部署「智能合約」，事先擬定一個規則，使用者只要遵守合約內容及規則，便可以安全放心的在區塊鏈上作任何交易。

隨著第五代行動通訊的發展，車物聯網的時代來臨，越來越多設備需要透過無線通訊的方式連結上網路，現在生活周遭看到的每一個物件未來都有可能加上感測器並配合無線通訊模組，將感測資料傳至雲端進行運算，同時，許多創新服務對於頻譜的需求非常大，往往需要一個完整且無干擾

的頻譜區塊，而頻譜的資源並不會憑空增加，頻譜資源勢必會更加壅擠。

除了使用大於 6Hz 的高頻段的頻譜進行通訊，如何在現有低於 6GHz 的頻段上做更有效率的分配及應用，避免頻譜資源被閒置，讓頻段持有者可以在沒使用頻段時可以將頻譜資源釋放出去，讓有需求但沒執照的使用者可以不必花大量資本購買執照便可花較低成本租賃頻譜，是資通訊發展至關重要的問題之一。從頻段頻率本身的角度、頻段使用時段的角度以及頻段使用的地理位置角度，透過動態頻譜接取技術，站在頻譜交易或者頻譜共享的觀點，設法將頻譜資源做最有效率的配置。而為了讓頻譜交易及頻譜共享的機制更加安全且更加公開透明同時保障交易的真實性，區塊鏈技術將是讓現有動態頻譜接取更加安全且更加有效率的關鍵技術及重要突破，區塊鏈配合動態頻譜接取勢必能促使對於頻譜資源有需求的資通訊科技業者蓬勃發展。

1.2 研究動機與目的

由於我國目前缺乏一個關於頻譜共享以及頻譜買賣租賃使用的一個平台，因此光靠動態頻譜接取技術也難以達成頻譜資源最有效率及最佳化應用的願景，而為了將頻譜資源最有效應用，需要一個平台提供給有多餘或者閒置頻譜資源可以出租以獲取額外利潤的頻段執照持有者，或者是提供給有頻譜需求但沒有執照的使用者，讓頻譜的資訊可以在平台上一目瞭然，分頻、分時或分地的頻段資訊，在不擾亂電波秩序的前提下達到雙贏的局面。

動態頻譜接取系統架構圖如圖 2 所示，其架構為 IEEE 1900.x 系列之標準，圖片左半部為軟體定義無線電架構下之感知無線電裝置，裝置可以透過其環境監測感知能力偵測環境頻譜使用狀況(Sense Environment)，再

進一步調整自身感知無線電的參數，以調整發送、接送頻率以及調整發射訊號的強度大小，避免干擾到原本的合法使用者。圖片右半部則為動態頻譜接取系統平台之架構，平台上需要有中心化的策略控制(Policy Control)，在確保不互相干擾、確保安全、確保連線品質的情況下，合理分配頻譜資源，以及維護電波秩序。

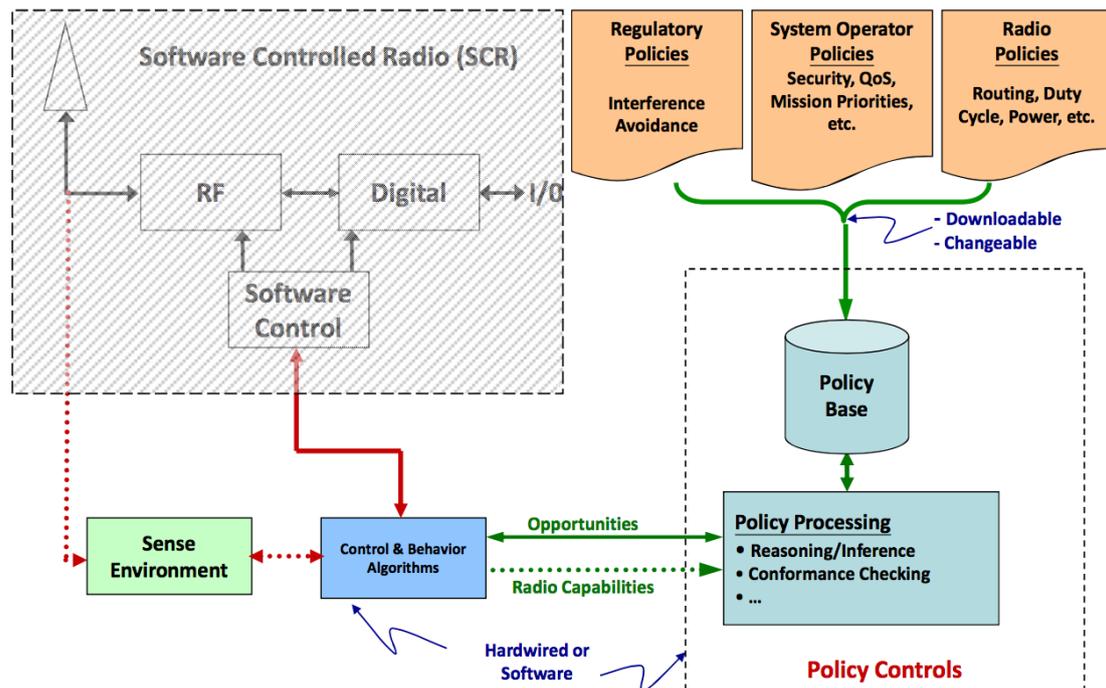


圖 2、動態頻譜接取系統整體架構圖

本研究將實作一符合 IEEE 1900.x 系列標準之動態頻譜接取系統，動態頻譜接取系統主要分為兩個部分，其一為如圖 2 左半部之感知無線電裝置，具備環境感知能力並且能夠依照需求動態調整發射參數，感知無線電裝置為連接到動態頻譜接取資料庫，也就是本研究實作的第二個部分，如上圖 2 右半部所示，動態頻譜接取資料庫會紀錄所有區域之頻譜使用狀況，區域內裝置基本資訊、裝置持有者、頻段擁有者、頻譜交易資訊、頻譜交易紀錄等頻譜資源相關資料，除了紀錄頻譜相關狀況外，動態頻譜接取資料庫也扮演著維持電波秩序和諧的角色，進行通盤的干擾分析，避免不同裝置之間相互干擾，影響合法使用者之權益。

除了以感知無線電實作上述之動態頻譜接取標準外，本研究也將 IEEE 1900.x 系列沒有提及之頻譜交易機制融入到動態頻譜接取平台當中，讓裝置可以透過此平台進行頻段的交易買賣，讓閒置的頻譜資源得以有效利用，並將區塊鏈的技術與概念融入到頻譜交易當中，使所有的頻譜交易資訊公開透明、交易成本降低的同時，保障頻譜交易的安全，不像許多傳統沒有納入區塊鏈概念的交易系統有著被駭客惡意侵入破壞的風險，以區塊鏈為核心技術的交易系統有著分散式帳本去中心化的特性，除非能一次入侵百分之五十以上的裝置，否則我們可以說此頻譜交易平台是絕對安全的，本研究目的為以下幾點：

- 分析現有動態頻譜接取系統之優缺點。
- 在軟體定義無線電平台上實作感知無線電設備，實現頻譜資源相關資訊的發佈、更新及設備參數的即時動態調整。
- 建立以區塊鏈核心概念為基礎之頻譜交易平台。
- 以感知無線電技術結合軟體定義無線電平台，實作一符合 IEEE 1900.x 系列標準架構之動態頻譜接取平台，並建立一頻譜資料庫。

1.3 論文組織

本研究著重在以軟體定義無線電實作一符合現有標準之動態頻譜接取平台，並將融入區塊鏈核心觀念及技術之頻譜交易機制納入在平台當中，讓所有參與在平台中的使用者可以透過平台進行頻譜交易，同時配合軟體定義無線電以及感知無線電的技術達到實際上透過上述之平台來調整自身設備參數以接取頻譜資源，並在不影響電波秩序的情況下，由頻譜資料庫來協調各個裝置使用平台，使頻譜資源使用更加有效有效率。

本研究第一章針對現行動態頻譜接取技術、軟體定義無線電技術、感知無線電技術以及區塊鏈技術及核心概念加以介紹，分析若以區塊鏈公開透明、保障交易安全、去中心化等核心概念實作以感知無線電為基礎之動態頻譜接取系統，同時並建立一個頻譜共享及租賃之平台，讓所有有需求之使用者能放心的在平台上獲取頻譜相關資訊、共享或者交易頻譜相關資源，使設備本身之感知無線電裝置能夠以軟體定義無線電修改自身參數，在不影響其他使用者的情況下接取頻譜資源，以提升頻譜資源使用之效率。目前我國現行之頻譜資源分配缺乏一個平台加以整合，因此本研究不只提出一個有效之平台以利動態頻譜接取，更提出將區塊鏈的核心技術及概念融合之頻譜交易機制納入至動態頻譜接取平台中，希冀能解決目前頻譜資源使用效率不彰之困境，最後歸納出本研究之研究目的。

本研究第二章針對過去有軟體定義無線電、感知無線電、動態頻譜接取系統及區塊鏈做相關文獻探討，並說明本研究與相關文獻差異之處。建構一符合國際標準之動態頻譜接取平台，並將區塊鏈核心技術及概念融入到頻譜買賣交易的機制當中，並將其買賣交易的功能加入到平台當中，分析現行動態頻譜接取系統之優缺點，同時補足目前動態頻譜系統中缺乏頻譜交易機制，並探討如何利用區塊鏈核心概念改善未來會出現在動態頻譜接取系統當中之頻譜交易機制。

本研究第三章將說明本篇研究之研究架構，敘述本研究使用之硬體設備以及軟體裝置，在軟體定義無線電開源平台 GNU Radio 上搭載 USRP B200 mini 裝置並利用其 UHD 模組進行相關參數設定以利實作以感知無線電技術為基礎之動態頻譜接取系統，並以樹莓派(Raspberry Pi 3)作為平台網路伺服器，實作一個以區塊鏈為核心之頻譜共享及租賃交易平台，同時連接平台及感知無線電裝置，使之能即時更新平台頻譜相關資訊。

本研究第四章為研究結果，說明本篇研究之環境配置，以 IEEE 1900.x 動態頻譜接取標準系列為主，參考國外動態頻譜接取資料庫或者相關平台進行平台實作，並利用軟體定義無線電技術及其平台實作感知無線電系統，即時串接上述實作之頻譜動態接取平台，以及敘述如何以 Python 程式實作以區塊鏈為底層核心技術之頻譜交易機制，交易發生當下將交易紀錄即時上傳到頻譜資訊資料庫中，交易紀錄也透過程式進行驗證，確保交易資料的正確性。本篇研究重點為將區塊鏈公開透明、去中心化及保障交易安全的特性融入在此平台當中，並對其優缺點做相關探討。紀錄本研究實作之過程，列出相關感知無線電之參數、平台頻譜資訊之假設，並解釋其效益以及優缺點。

本研究第五章為結論，說明感知無線電實作之動態頻譜接取平台、區塊鏈核心概念之頻譜交易機制的貢獻，以及頻譜資料庫本身對整個動態頻譜接取系統架構的效益及其必要性，針對動態頻譜平台優缺點做討論，以及對缺點提出相關改善措施，探討為何頻譜資料庫較難以去中心化的概念實作，為何只針對頻譜交易的部分做區塊鏈概念的實作，並對未來研究提出建議。

第二章 文獻回顧探討

本章節將針對本篇研究所使用之技術、硬體設備以及使用之開發平台做一個詳細的介紹，內容包含區塊鏈特性及其相關技術、動態頻譜接取技術、軟體定義無線電開發平台，最後再進行目前現有區塊鏈融合動態頻譜接取技術之相關文獻探討統整，並整理出小結。

2.1 動態頻譜接取

隨著資通訊科技不斷的發展，第五代行通訊的逐步成熟，萬物聯網時代的來臨，不只人對於無線通訊的需求日趨增加，越來越多的設備裝置也將會連上網路，或者進行裝置對裝置的溝通，然而頻譜資源有限，並不會憑空增加，因此動態頻譜接取(Dynamic Spectrum Access, DSA)技術成為了讓頻譜資源更做更佳有效率利用的解方。

動態頻譜接取[7]主要是給沒有頻譜執照的次要使用者(Secondary User, SU)接取頻譜資源所使用之技術，在避免影響到合法持有頻譜執照的主要使用者(Primary User, PU)之權益的情況下接取頻譜資源。一般來說，頻譜資源的接取可以以「使用頻率」、「使用地點」、「使用時間」三個面向來區分，動態頻譜接取技術可以有彈性且即時的使用頻譜資源，避開正在被使用中的頻段、在閒置的時段使用某特定頻段以及調整其發射功率使其不會與其他訊號相互干擾。

圖 3 及圖 4 為主要使用者(PU)及次要使用者使用(SU)頻率的示意圖，圖 3 代表次要使用者只能在主要使用者閒置時使用[8]，並在下一個時框時即時且動態地改變其使用頻段，讓閒置的頻率也能被有效利用，以圖 3 來說，在時間為 t_1 時，主要使用者使用 B1 到 B3 頻段，因此次要使用者使用閒置的 B7 頻段，而下一個時刻 t_2 時，原本使用中的 B1 頻段由使用中

轉為閒置狀態，因此次要使用者便可以使用 B1 頻段，讓閒置的頻段資源得以被有效使用。圖 4 則代表次要使用者在不干擾主要使用者訊號的情況下，利用較低的功率發送訊號，達到頻譜共享的效益，縱軸為功率大小，橫軸則為頻段，由圖 4 我們可以看到，雖然主要使用者及次要使用者使用的是同一個頻段，但次要使用者裝置的功率較低，不會干擾到主要使用者的權益，因此可以和諧的共享此頻段。

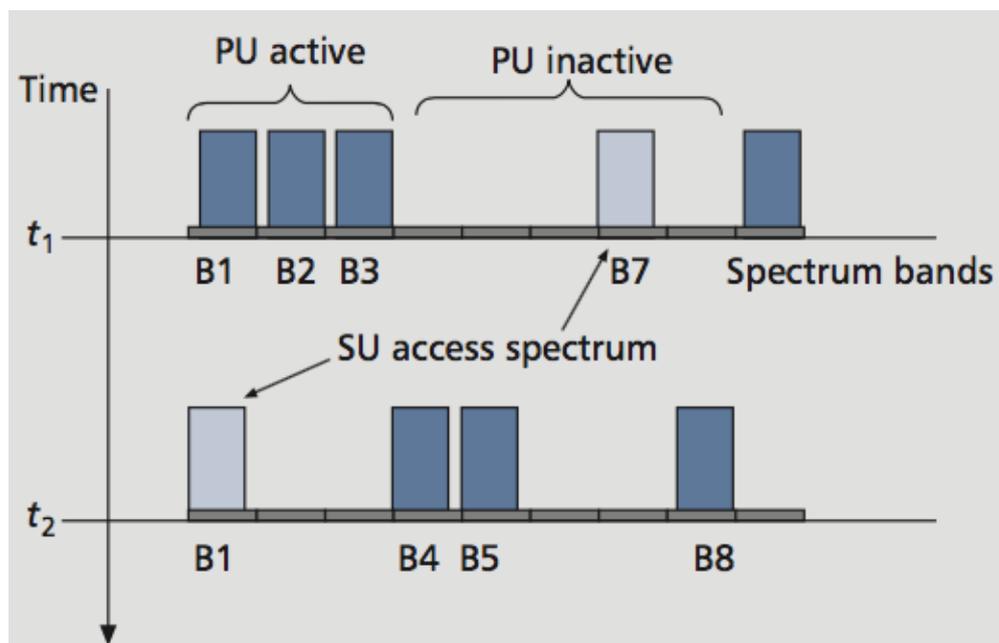


圖 3、動態頻譜接取模型(以使用頻段區分) [8]

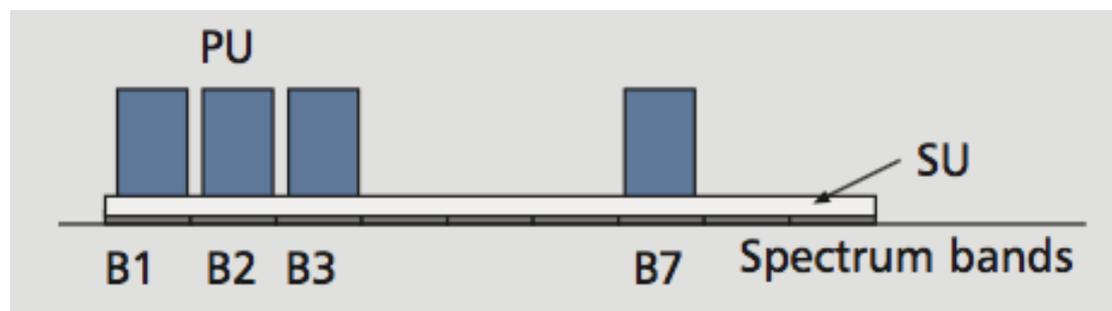


圖 4、動態頻譜接取模型(以發射功率區分) [8]

而軟體定義無線電技術以及感知無線電技術為實現動態頻譜接取之重要技術，使其能夠感測環境頻譜相關使用資訊，進而動態的調整其設備參數，將在以下兩小節分別詳述之。

2.1.1 軟體定義無線電

軟體定義無線電(Software Defined Radio, SDR)最早於 1990 年代由 Mitola 提出，其概念為讓許多傳統硬體裝置裡面的功率放大器、濾波器等電子元件能夠透過電腦軟體取代，且許多參數如發射頻率、發射頻寬、天線增益值、訊號調變方式……等都能夠輕鬆透過軟體介面加以操作調整，如此以來不必大費周章，耗費較高成本更換不同硬體裝置便可達到客製化通訊規格及功能的效果，由讓電腦軟體來處理訊號。

軟體定義無線電架構圖如圖 5 所示，軟體定義無線電並非讓電腦軟體完全取代傳統硬體裝置，軟體無線電還是得仰賴前端射頻裝置(RF Front-End)透過外接天線接收或者發送訊號，訊號透過類比數位轉換器(Analog to Digital Converter, ADC)、數位類比訊號轉換器(Digital to Analog Converter, DAC)、數位類比訊號轉換器(Digital to Analog Converter, DAC)，以及可程式邏輯閘陣列(Field Programmable Gate Array, FPGA)做訊號處理，再利用電腦通用序列匯流排(Universal Serial Bus, USB)接口作為溝通硬體裝置以及電腦軟體的平台，讓電腦程式軟體進行後續訊號處理，使用者可以依據其不同的需求，透過軟體定義無線電平台實作符合其需求的通訊系統。

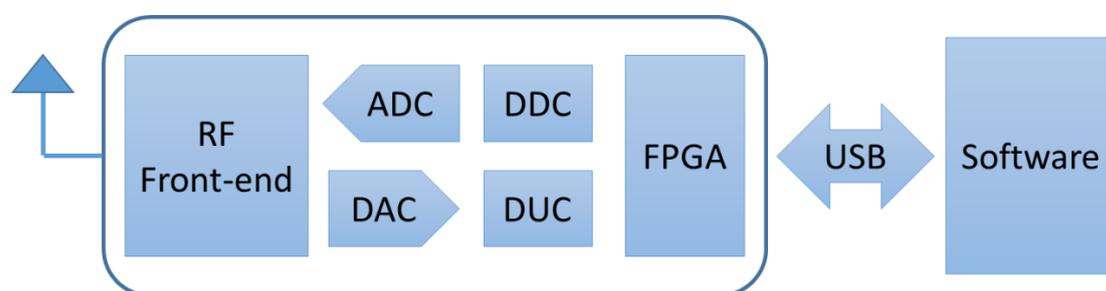


圖 5、軟體定義無線電架構圖[4]

2.1.2 感知無線電

感知無線電以軟體定義無線電技術為基礎出發，並同時有著環境感知的能力以及重新配置設備參數的能力[9]，感知無線電之示意圖如下圖 6 所示。感知無線電裝置可以有智慧地偵測目前所在地的環境相關資訊，監聽頻譜使用狀況，辨別出目前什麼頻率閒置中，什麼頻率正在被使用，亦或者感知無線電裝置可以根據頻譜共享及交易租賃的平台資料庫知道哪個頻率目前可以使用、什麼時間可以使用、以及在什麼樣的發射功率下可以不干擾到正在合法使用頻段的使用者，依照平台的頻譜相關規範調整其發射頻率、發射功率、訊號調變方式等相關發射參數，以維持該地電波秩序和諧。

動態頻譜接取技術正是利用軟體定義無線電技術及感知無線電技術，可以主動偵測環境中可使用的頻段調整其參數並使用之，或者可以透過頻譜共享及交易租賃平台，當所有感知無線電裝置可以一起納入平台的規範時，所有裝置皆可以動態且和諧合法的使用頻譜資源，即時配置其參數，達到頻譜資源有效率之應用。

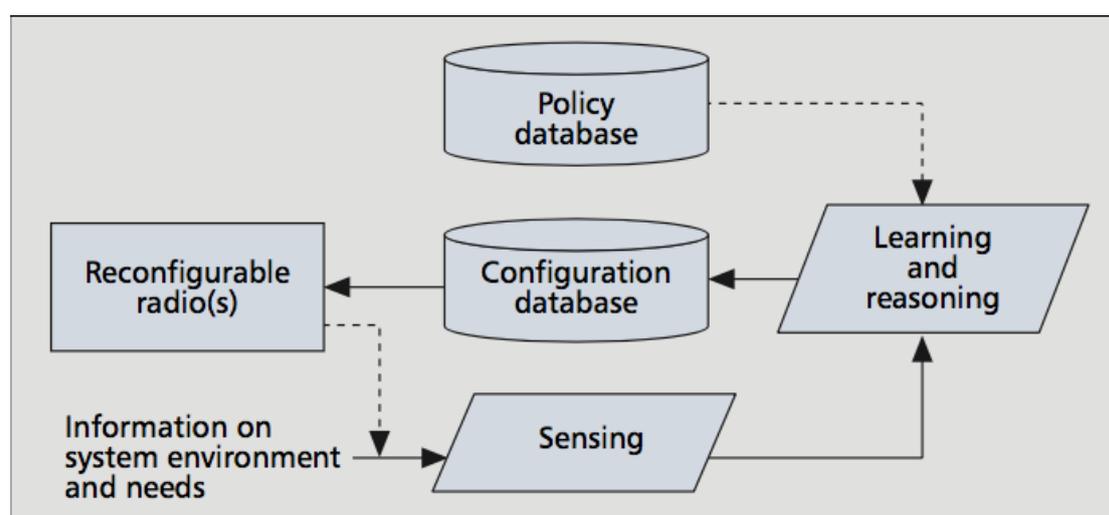


圖 6、感知無線電系統架構[9]

2.1.3 頻譜資源資料庫

頻譜資源除了如圖 7 所示的可以以頻率、以時間來區分頻譜是否閒置未被使用(閒置的頻段資源被稱為頻譜空洞, Spectrum hole)或正在被使用 [10], 頻譜資料庫可以判斷裝置可以在何時使用何種頻率來收發訊號, 進而在動態地接取使用頻譜資源時, 確保裝置之間不會相互干擾。頻譜資源還可以依照地理位置來區分頻譜資源能否被使用, 畢竟, 某發送或接收頻段的訊號影響範圍一定有其極限, 在影響範圍外的裝置若使用該頻段做無線應用是不會影響到原有使用者的。例如各國之電視頻段共享[11](TV White Space, TVWS)模式便是利用地理位置來區分頻譜資源, 電視頻段共享為目前各國最普遍常見的頻譜共享模式, 其為廣播電視常見之 VHF/UHF 頻段, 約在 470MHz 到 800MHz 之間, 各國無線電視數位化後, 可以使用的頻譜資源提升, 因此給了不論是頻段持照使用者還是未持有頻譜執照的使用者更大的發揮空間做資通訊技術運用, 但同時還是要避免既有合法持照使用者如地方電視台廣播業者等受到其他未持有執照的使用者干擾, 保障既有持有執照業者之權益。

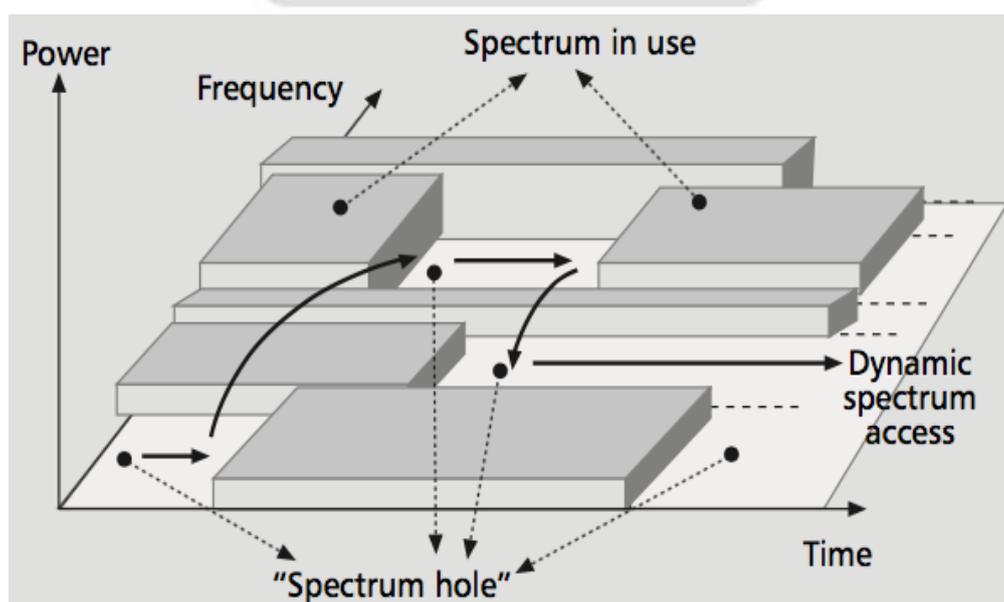


圖 7、分頻、分時之頻譜使用狀況示意圖 [10]

各國在運作電視頻譜共享機制時，多數採用地空空白頻譜資料庫系統 (Geo-location white space spectrum databases, GL-WSDBs)來協調主要使用者及次要使用者使用頻譜資源[12]，資料庫均須符合各地主管機關規範。當設備有使用需求時，需先連線至地空空白頻譜資料庫查詢該地頻譜相關使用資訊，該資料庫會回傳該裝置所在地附近的頻譜使用狀況，進而告知該裝置目前合適以何種頻率及何種功率大小收發訊號，一但使用頻譜資源後，會將自身裝置所處位置、收發頻率以及發射功率大小等參數回傳至資料庫，避免其他使用者使用該通道而相互影響。除了分頻、分時看到頻譜使用資訊外，可以得知目前全國各地電視空白頻段使用狀況及分佈，並且即時更新以合理分配頻譜資源，保護所有使用者之權益，而任何需要使用電視空白頻段的使用者，無論是否持有持照，裝置都需通過該地主管機關之型式認證，合乎發射功率等規範，避免裝置之間互相干擾，由資料庫進行最終的干擾確認。

2.2 區塊鏈概念及其特性

區塊鏈的許多概念早在 1990 年代就被提出來了，但到了 2008 年中本聰(Satoshi Nakamoto)發表了比特幣的白皮書[13]，才將區塊鏈的應用帶到了嶄新的境界，並造就了家喻戶曉的虛擬加密貨幣「比特幣」，讓全球各地掀起了一股「挖礦潮」，區塊鏈也不僅僅只侷限在虛擬貨幣的應用，只要牽涉到資料的儲存以及保護，區塊鏈都能夠派上用場，儼然只要區塊鏈概念融合某個已有的應用就可以成為下一個破壞式的創新。

區塊鏈的核心概念十分簡單，講白了只是一個記帳的軟體亦或者可以說是一個每個人手上都有一份的帳本而已，雖說區塊鏈演變至今還有分成「公有鏈」、「私有鏈」、「聯盟鏈」等，但其核心觀念始終如一，區塊鏈概念及技術有以下特點：

- 去中心化之分散式帳本技術

區塊鏈最一開始的應用為虛擬貨幣如比特幣、以太幣等的交易，為了避免重複交易、重複扣款、駭客入侵等傳統網路交易可能會發生的問題，區塊鏈採用分散式帳本技術，簡言之，就是將所有網路上的每一筆交易紀錄都儲存在每一個用戶的裝置當中，而非由某一中心化金融銀行機構控制，一方面可以在交易時節省被銀行機構被收取之手續費，不經過中心機構便可以達成點對點(Peer to Peer, P2P)的交易，另一方面可以保障每一用戶的帳戶安全，若以舊有金融機構中心化模式來說，如果今天有駭客入侵資料庫惡意更改帳戶餘額，那將會造成不論是金融機構還是其用戶莫大的損失，錢財將會憑空消失，而因為區塊鏈分散式帳本的技術，所有交易資料都存在每個用戶端，除非駭客能夠入侵每一台裝置竄改資料，不然戶頭裡的餘額是不會輕易被盜竊的。

- 公開透明

因為上述分散式帳本且去中心化特性，每一筆交易資料都被儲存在每個人的裝置上，因此訊息公開且透明，圖 8 所示為比特幣區塊鏈中所擷取之某幾筆交易，由圖 8 我們可以看到每筆交易的細節，何年何月何日何時，由哪一個用戶轉入多少比特幣至哪個帳戶中，一目瞭然，任何人都能夠輕易存取區塊鏈上每一筆資料，不一定要加入到區塊鏈當中我們也可以獲得區塊鏈上的所有資料，而也因為其公開透明的特點，降低人與人之間的不信任成本，為了彌平點對點間的不信任，傳統上採用第三方機構的方法，由第三方來驗證雙方的交易是否有效，但第三方機構也有著收取收續費及其依然為中心化機構的問題，因此區塊鏈的公開透明特性可以保障雙方權益，交易內容也可以受到其他節點的監督，也無法否認交易內容的真實性。

Transactions		
fc06e8933105bf179d91c5b26c17c405a07229fe519063003a2622995c91	2016-11-15 08:26:59	
No Inputs (Newly Generated Coins)	→ 1KuWLoZuoJgz3N6sLoAvGth5XGm8YufTGT	12.91251118 BTC
		12.91251118 BTC
6ebad12706b60350d93ba2c42ce6d47cece8289447508e1a685857d289134cd	2016-11-15 08:26:58	
1YmVfLU347vMh65TNZE3hugCJgTmomLW	→ 1YmVfLU347vMh65TNZE3hugCJgTmomLW 19ycWQ2z2gJ1kZbp2z2knq3GKANcBMSZ1r	0.14734068 BTC 1.707 BTC
		1.85434068 BTC
86733d9ad44ba3d576cd5284ac879e50634e600467b6d8c80fa8d5a8atae5254	2016-11-15 08:23:52	
1ALNTyDtmLzNoZVW5MNMVKWb8Leigdk3Tj	→ 1MnTMW2iMUaxeGUJQzokWdJSHwC4tTjQ8 1KpXeWbX79MUauZmLeLRGVWFdXPK3EJqT	64.464 BTC 5.067 BTC
		69.531 BTC

圖 8、比特幣交易紀錄(擷取自區塊 No. 438995) [14]

- 加密技術保障交易安全

區塊鏈上的每一筆交易都是透過 RSA 加密演算法加密的，此加密法有著非對稱的特性，需要一把公鑰及一把私鑰，公鑰為公開資訊，私鑰則由使用者保存且不公開，加密方式非常簡單，將訊息內容透過對方之公鑰彙整成一定長度的亂碼，對方透過其私鑰可以輕易破譯訊息內容，倘若沒有私鑰則解密難度極高，甚至無法被破解，若沒有私鑰是無法偽裝成該使用者進行交易的，因此我們可以說區塊鏈採用之加密技術是能保障交易安全，不必擔心傳統網路銀行帳號密碼儲存在金融中心機構資料庫會有資料竊取的問題，自己的私鑰自己保管，而若私鑰因某種原因遺失，則整組鑰匙將失效，無法僅透過公鑰進行交易。

- 資料無法被輕易竄改

區塊鏈一詞可以分為「區塊」加上「鏈」兩字，將不同「區塊」透過「鏈」串接在一起成為區塊鏈，每個區塊內包含數筆交易資訊，每筆交易資料透過安全雜湊演算法運算(Secure Hash Algorithm, SHA-256)，SHA 安全雜湊演算法的特性為不管原本的資料內容多寡，產出的資料皆為一固定位元長度的亂碼，哪怕只是更改原本資料的一個符號也會讓產出的亂碼完

全不一樣，因此非常容易被驗證是否被竄改過。一個區塊的多筆交易資料最後透過名為 Merkle Tree 之樹形資料結構生成獨一無二的雜湊函數值，而每一個區塊的雜湊函數值都內含前一個區塊所生成之雜湊函數值如圖 9 所示，因此行成一個鏈接的概念，讓每一個區塊透過雜湊函數串連起來，如圖 10 所示，也就是說，如果竄改整個區塊鏈當中任一筆資料，就會導致區塊鏈無法被串接起來，因為區塊前後的雜湊值會對不上，除非竄改並重新演算每一個區塊的雜湊值，才可以讓每個區塊重新被鏈上，但同時也因為分散式帳本特性的關係，每個使用者手上都有一份區塊鏈的備份，只要某人的區塊鏈帳本與其他大部分人的不相同，我們就能輕易斷定該使用者是有問題的。

Hashes	
Hash	0000000000000000042b6b0a7bfa7f43b648167a5c0547d6b3a676e23d4c226
Previous Block	0000000000000000023522e6372109dd28f97a86eef46a54cf15199c005e6a85
Next Block(s)	00000000000000000257616ab8ed39e0e7bca056d13fa11652075a34e335e7ed
Merkle Root	4a6be460a67526fd576788cf37713ae614003989706a0675e9d946ec58ca030

圖 9、比特幣區塊資料(擷取自區塊 No. 438995 Hash 值部分)[14]

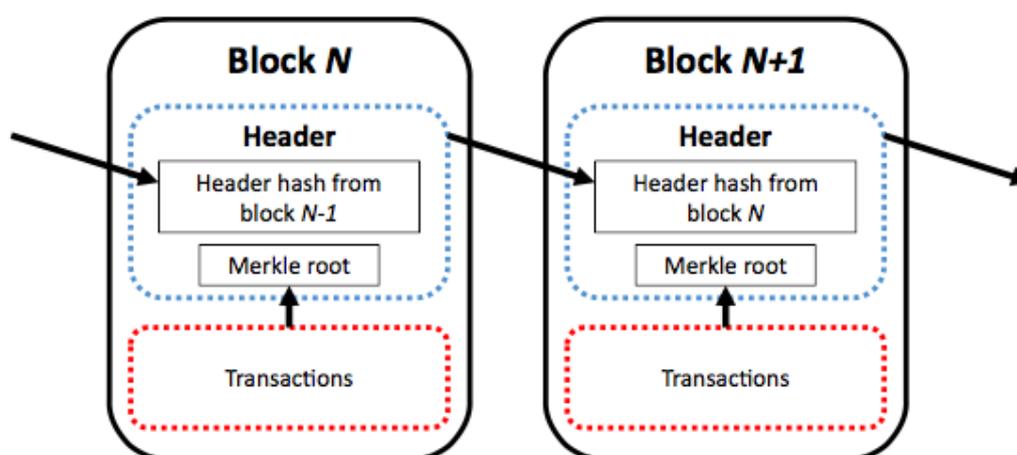


圖 10、區塊鏈概念示意圖 [14]

2.3 區塊鏈技術實作動態頻譜接取系統文獻回顧

因為區塊鏈特性的關係，其應用十分廣泛且多元，從虛擬貨幣交易到食品安全認證甚至到能源運用[14]都能見其身影，文獻[15]指出區塊鏈技術結合物聯網應用潛力無窮，因為萬物聯網時代下每個裝置都能視為一個節點，將不必透過任何一個中心裝置控制所有裝置，節點與結點間可以互相傳送資料，不必擔心整個智慧工廠、智慧家庭等受到攻擊，進而失去控制權導致安全漏洞門戶大開。

文獻[16]指出在車聯網的架構下，安全將是最重要的議題，若車輛遭到駭客惡意入侵，可能將導致嚴重的車禍事故，因此研究中將區塊鏈的技術結合動態頻譜接取系統當中，將每一台車輛的感測器、控制晶片、制動器等元件都視為一個感知無線電裝置，其接取頻譜的主要方式為透過一虛擬貨幣「Specoins」進行頻譜交易，主要使用者若有閒置的頻譜資源可以透過公告拍賣頻譜的方式出租或販售頻譜資源，若次要使用者要使用頻譜資源就必須花費 Specoins 購買頻譜使用權，而 Specoins 的取得則可以透過以下兩點方式取得：

- 透過挖礦的方式維護區塊鏈以賺取 Specoins
- 透過實體現金貨幣購買 Specoins

此文獻透過頻譜交易讓主要使用者可以透過出租頻譜的方式獲得額外利潤，次要使用者則可以透過較低成本接取頻譜資源，且裝置不用一直處於偵測模式以提升使用效率，並用區塊鏈技術來保障交易的安全，任何想使用頻譜資源的裝置都可以透過拍賣機制獲得頻譜使用的權限。文獻最後指出此利用區塊鏈技術實作動態頻譜接取拍賣的模型在效能上是比傳統隨機網路接取方式還更有效率的，而在能耗上則比傳統接取方式還耗電。

2.4 文獻歸納小結

透過區塊鏈及動態頻譜接取相關文獻研讀，顯示出在隨著下世代行動通訊的發展如火如荼的同時，我們勢必得正視頻譜資源缺乏的問題，而透過頻譜動態接取系統可以較有效的改善頻譜使用效率，區塊鏈技術也可以讓頻譜交易更佳有保障且更加有效率，然而文獻[16]僅對於頻譜交易機制做討論，讓感知無線電通過區塊鏈交易頻譜使用權，主要使用者透過主動廣播的方式拍賣頻譜使用權，因此需要一個專用的拍賣頻道來搜集這些拍賣資訊，其缺乏一個交易平台的概念來發布這些頻譜拍賣資訊，且文獻並未針對頻譜共享機制提出一個更可靠的模型。

目前國外許多頻譜共享機制以及許多文獻如[11][12]皆提出一個共享頻譜資料庫系統的概念，分頻、分時以及分區地搜集頻譜相關資訊，讓次要使用者能夠友善地共享頻譜資源，避免影響主要使用者權益，然而文獻中也提及這類頻譜資料庫系統有著安全上的疑慮，目前資料庫大多由政府機關中心化地管控，一方面一一認證平台上頻譜資訊及認證所有次要使用者的裝置的合法性十分困難，另一方面維護資料庫安全需要大量成本，若是資料庫遭到有心人士侵入並不當存取破壞資料將可能導致電波秩序大亂。且文獻皆是針對頻譜共享機制做討論，資料庫也是針對頻譜共享頻段做設置，並未對頻譜租賃交易機制做探討。

在區塊鏈技術的概念下，我們可以確保各式各樣的資料真實性、交易的安全性及整體架構的穩定性，雖說以實際情況而言，動態頻譜接取資料庫沒辦法真的去中心化，其中牽涉到的監理法規部分，終究是要有中心監理機關制定法規並約束之，但若能將此概念融入至動態頻譜接取系統中頻譜交易的環節當中，並實作一個平台讓所有有頻譜需求的使用者可以放心地再上面共享及交易租賃頻譜資源，相信將能使未來的頻譜交易機制更加

安全，頻譜資源的利用比以往更加有效率，且可以受到所有節點裝置的監督，讓頻譜交易更有保障，以下列點本研究對於區塊鏈融合動態頻譜接取的想法及概念：

- 沒有任何中心化機構能夠掌控頻譜共享及交易租賃，交易不需經第三方認證(去中心化)
- 所有頻譜共享及租賃交易內容公開透明且交易紀錄被所有人維護(分散式帳本技術)
- 任何感知無線電裝置都能輕鬆在上面做動態頻譜接取(幾乎零成本交易、進入障礙低)
- 每筆頻譜交易資料及頻譜共享紀錄皆無法被竄改(時間戳記、雜湊函數演算法)
- 每筆頻譜交易資料及頻譜共享紀錄能夠輕易被主管機關驗證(透過雜湊函數驗證，不必耗費大量運算空間及時間一筆一筆交易比對，某筆資料一經查證有問題，必能推得其後所有交易亦有問題)

本研究探討過去相關文獻後列出以下不同之處：

- 過去相關頻譜資料庫的文獻皆是針對頻譜「共享」，缺乏頻譜交易及租賃的平台，且過去文獻資料庫的安全問題並未有一個有效的解方，因此本研究將實作一含有頻譜共享及交易租賃之動態頻譜接取平台，並提出將區塊鏈概念技術融入至平台中。
- 頻譜的交易不再透過主要使用者主動廣播的方式拍賣，而是直接將拍賣資訊放上平台，所有頻譜交易租賃資訊公開透明，讓次要使用

者可以透過虛擬貨幣購買頻譜資源，並用區塊鏈技術保障交易安全。

- 先前研究並未詳細說明如何以區塊鏈技術實現動態頻譜接取，僅說明透過區塊鏈交易頻譜資源相比於傳統頻譜接取方式更有效率，而本研究將透過軟體定義無線電平台，實作感知無線電裝置，連接上區塊鏈平台，並分析其優缺點，進而提出改善方式。
- 本研究實作之動態頻譜接取平台，由中心化的頻譜資料庫儲存不同區域內的頻譜狀況，並進行通盤的干擾判斷，避免裝置之間相互影響，並將頻譜交易機制實作在平台當中，每一個感知無線電設備節點中都有者平台上所有資訊的交易紀錄備份，因此較無交易安全上的疑慮，不必擔心遭到惡意入侵濫用頻譜，同時可以讓每一個節點感知無線電裝置發揮頻譜監控的功能，並共同維護此平台以獲取接取頻譜資源的獎勵籌碼。

第三章 研究架構

本章節將介紹本研究之架構，同時介紹所有實作以感知無線電技術為基礎之區塊鏈動態頻譜接取平台會使用到之軟硬體設備。第一部分將介紹用來開發軟體定義無線電之開發平台 GNU Radio，第二部分將介紹實作感知無線電裝置會使用到之軟體無線電前端射頻設備 USRP、用來作為平台網頁伺服器之微型電腦樹莓派(Raspberry Pi 3)，第三部分則介紹如何在軟體定義無線電平台下實作感知無線電收發系統及其架構，第四部份介紹區塊鏈實作去中心化動態頻譜接取平台之設計架構，將敘述平台上的資料內容及操作邏輯，如何將區塊鏈概念融入當中，最後一部分則介紹將感知無線電裝置融入至平台之系統整合架構，以達到即時更改感知無線電參數之動態頻譜接取系統。

3.1 軟體架構

軟體定義無線電技術發展至今有許多開發平台，有些平台需要綁定特定前端裝置才能使用，有些平台則需要付費才得以使用且不少功能介面都被軟體開發者鎖定住，使用者較無法在平台上彈性的依據自身需求開發相關應用，因此本研究選定在 Linux 系統下之開源軟體定義無線電開發平台 GNU Radio 作為本研究軟體平台。

3.1.1 GNU Radio

GNU Radio [17]為一款免費且開源之軟體定義無線電開發平台，其只要連接外部射頻裝置便可以透過內建的模組進行開發，GNU Radio 的環境圖形化介面本身是由程式語言 Python 撰寫之，而真實的底層訊號處理則由程式語言 C++做處理，透過 Python 之圖型化介面及其腳本程式操控底層訊

號處理。

GUN Radio 的開發環境如下圖 11 所示，與傳統我們認真程式開發環境堆疊著一行又一行程式碼的刻板印象不同，GNU Radio 開發介面十分友善且非常直覺，其運用「區塊」的概念做軟體定義無線電開發，圖上每一個區塊都代表訊號處理的一個過程，使用者只需要有基本訊號處理的流程概念即可，點擊兩下方塊便可以調整參數，不必真的寫一行一行的程式碼，僅需透過箭頭將圖片右方需要用到之訊號處理方塊如調變方式、濾波器……等拉到左方並串連起來便可以完成軟體定義無線電開發。

而若是遇到「拉方塊」辦不到的事情，我們則可以透過修改拉完方塊後所產生之 Python 原始碼達到我們需求，若再遇到修改 Python 原始碼辦不到的事情，則必須更深層的修改訊號處理的 C++原始碼。有別於其他軟體定義無線電開發平台，GNU Radio 基本上將所有控制權交給使用者，而不是綁死在平台開發者手上，十分有彈性，幾乎能實作各式各樣之通訊系統。

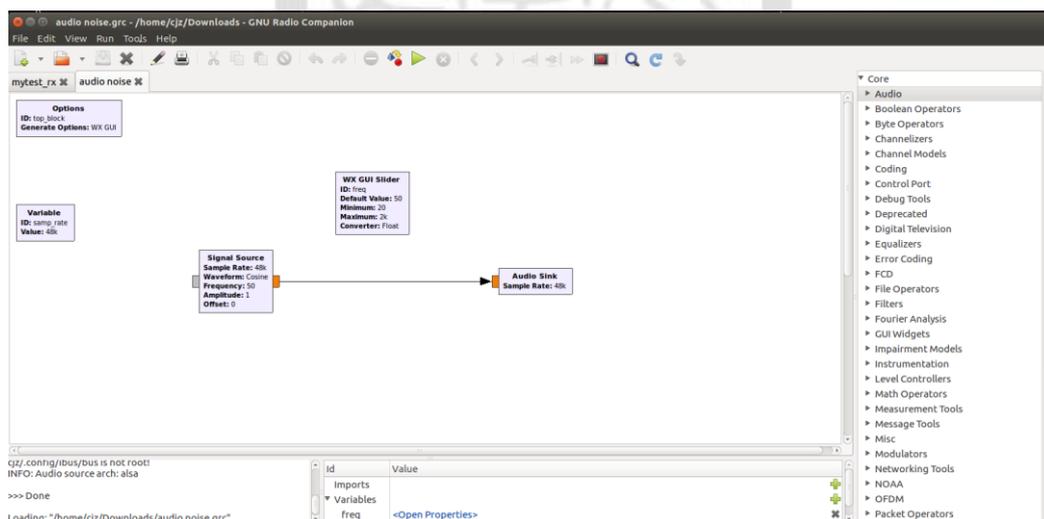


圖 11、GNU Radio 開發環境

3.2 硬體裝置介紹

軟體定義無線電裝置的訊號的發射及接收來自於前端硬體射頻裝置，收到訊號後才能交給電腦軟體處理後續內容，或者先由軟體無線電平台處理訊號後再經由射頻裝置發射訊號。

本研究所使用之前端射頻裝置為 Ettus Research 公司所開發的 USRP B200 mini [18]，並將軟體定義無線電平台安裝至微型小電腦樹莓派中，讓樹莓派成為一感知無線電裝置，同時將本研究欲實作之頻譜共享及交易租賃動態頻譜接取平台設置在樹莓派當中，作為區塊鏈當中的節點，共同維護平台穩定性。

3.2.1 USRP 技術規格

USRP B200 mini 規格如表 1 所示，裝置圖如圖 12 所示，USRP B200 mini 不需另外接電源供電，透過電腦 USB3.0 接孔直接供電並傳入訊號自軟體無線電平台中。其頻率範圍非常廣，搭配不同頻率之天線，可以從較低頻之廣播電台頻率到高频之車聯網 VANET 所使用之頻率，並可以同時做收送及發送資料的動作。其開發模組非常容易取得，在 GNU Radio 中僅需幾行指令安裝 USRP 之 UHD 模組便可以使用 USRP 作為軟體定義無線電開發之前端射頻裝置，一樣以滑鼠點擊 USRP 的模組方塊便可以調整其發射頻率、發射頻寬、天線增益值等參數，是一款開發軟體定義無線電應用的利器。

表 1、USRP B200 mini 規格

發送/接收頻率範圍	70MHz-6GHz
頻寬	最高可達 56MHz
電腦連結接口	USB3.0
支援模式	Full Duplex(一個接收一個發送)
最大取樣率	61.44MSample/second(MS/s)



圖 12、USRP B200 Mini [12]

3.2.2 樹莓派技術規格

本研究使用由英國樹莓派基金會開發之微型電腦樹莓派(Raspberry Pi)，最初的開發目的以普及電腦科學教育為主，其有著低成本且易攜帶的特性，入門門檻非常低，以開源之 Linux 做為電腦之開機作業系統，目前發展至今已演變到第三代樹莓派(Raspberry Pi 3)，系統規格越做越好，實體網路介面、Wi-Fi 介面、藍芽傳輸等功能一應俱全，其規格如下表 2 所示。

樹莓派實體圖片如圖 13 所示，除了作為電腦的應用以外，上面還有 40 根 pin 腳的 GPIO 孔，可以外接許多如溫濕度感測器、計步器、七段顯示器等物聯網裝置，配合軟體無線電技術相信其發展潛力無窮。本研究將上述之軟體定義無線電前端射頻裝置連接自樹莓派上，並將動態頻譜接取資料庫平台伺服器實作於此。

表 2、Raspberry Pi 3 規格

CPU	1.2 GHz 64-bit quad-core ARM Cortex-A53
記憶體	1GB LPDDR2
視訊輸出	HDMI
儲存裝置	microSD
USB	USB2.0 x 4
藍牙連接	Bluetooth 4.1; Bluetooth Low Energy(BLE)
乙太網路	10/100 RJ45
無線網路	802.11n
重量/尺寸	42g/85mm x 56mm x 17mm
GPIO	40-pin 2.54 mm (100 mil) expansion header: 2×20 strip



圖 13、樹莓派實體照片

3.3 感知無線電模組設計

本研究使用 Linux 作業系統下開源的 GNU Radio 作為軟體定義無線電開發平台，前端硬體裝置為 USRP B200 mini，接收及發射的頻率範圍非常廣，約在 70MHz 到 6GHz 之間，本節將分成「傳送」端以及「接收」端詳述本研究感知無線電模組設計，預計接收及發送的資料為簡單的文字.txt 檔案，傳送資料前會將感知無線電裝置本身的相關資訊傳送到頻譜資料庫，讓頻譜資訊資料庫進行傳送請求的裁定。本研究傳送及接收的檔案如下圖 14 為例，若今天某感知無線電裝置為車輛裝置，在車聯網的架構下，要接收或者發車輛數據的相關資料，例如車輛名稱、時間、引擎轉速、時速等，以利與其他車輛裝置或者路側設備裝置做資料交換，此時便需要使用某頻段作為收發資料之頻段。

```
Car1  
Time:9:20:29.807336  
RPM:0  
MPH:0.0  
Throttle:5.88235294118  
Temperature:188  
Engine Time:29:20:29.950159
```

圖 14、收發之車載資訊

3.3.1 傳送端設計

感知無線電裝置傳送端在 GNU Radio 平台下的流程圖如下圖 15 所示，首先一開始將要發送的車載資訊資料文字檔放入「File Source」的方塊當中，然後經過「Packet Encoder」方塊將數據編碼後經由「GMSK」調變方式透過「USRP Sink」方塊將資料傳送出去，本研究除了 GMSK 調變方式

外，亦會加入 OFDM 的傳送方式(子載波調變方式為 QPSK)，透過更改 Python 程式碼，不必更換硬體裝置便可以更換調變方式，即是所為軟體定義無線電。

在最一開始資料要送出前需要先設定 USRP B200 mini 的裝置參數如發送中心頻率、天線增益值、發送頻寬等，因此為了確定參數配置，裝置須先連結上動態頻譜接取之資料庫，利用共享或者交易租賃的方式取得頻道的合法使用權，以避免干擾到其他使用者，而在參數都調整完後需要先進行「通道偵測」，偵測所選定之發送頻率有沒有人被其他人使用，若確定沒有其他人正在使用時便可以開始傳送資料。

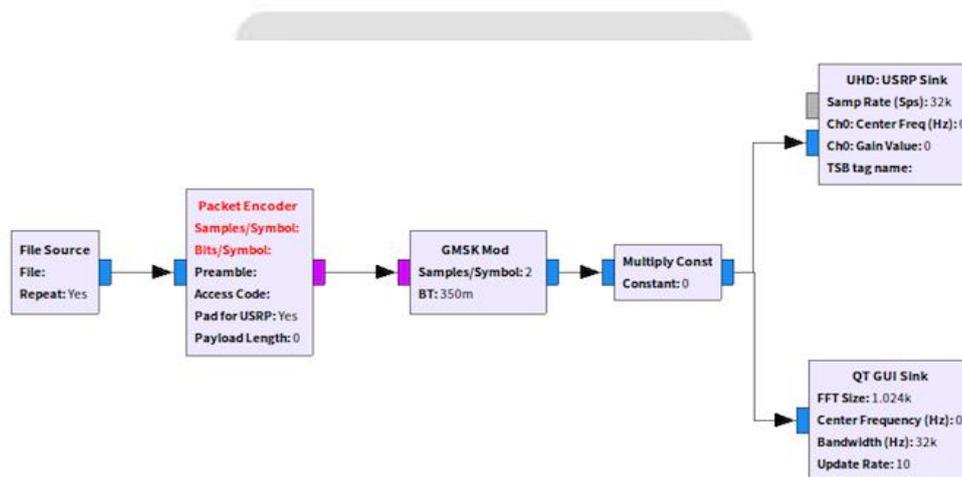


圖 15、感知無線電傳送端設計

3.3.2 接收端設計

感知無線電裝置之接收端裝置則與傳送端相反，如下圖 16 所示，首先可以從動態頻譜接取資料庫得知發送端的中心頻率，調整成與發送端相同的中心頻率後，透過「USRP Source」接收到訊號，經由低通濾波器「Low Pass Filter」後再將資料透過「GMSK Demod」及「Packet Decoder」兩個模組解碼，便可接收到原始的資料內容，訊號強度大小則由「QT GUI Sink」產生 FFT 圖呈現。

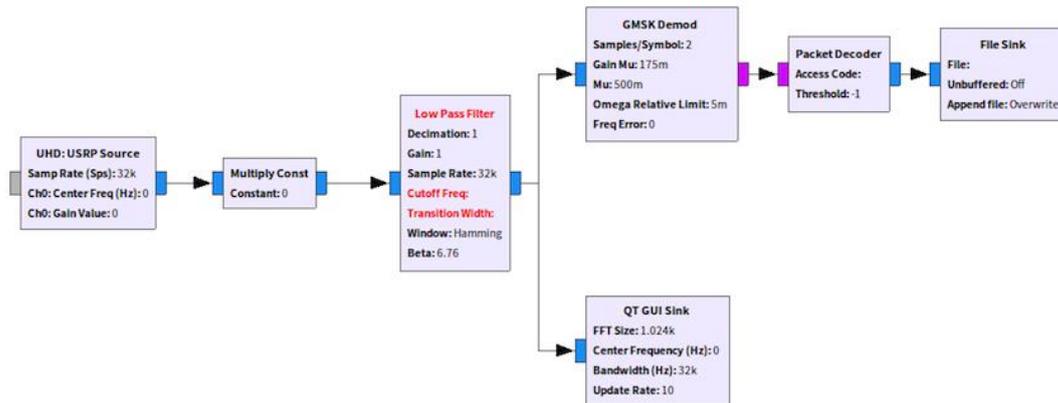


圖 16、感知無線電接收端設計

3.4 區塊鏈實作去中心化動態頻譜接取平台

本研究將實作一可以讓主要使用者及次要使用者和諧使用頻譜資源的頻譜共享及交易租賃之動態頻譜接取平台，使用上述提及之樹莓派作為本平台之網頁伺服器及資料庫伺服器，平台上將用擬定之虛擬貨幣作為租賃或者購買頻譜資源的媒介。

動態頻譜接取資料庫平台伺服器後端系統以程式語言 Python 作為網站開發的語言，同時以 Python 撰寫區塊鏈的核心概念邏輯，將每筆交易資料區塊彙整成一雜湊函數值，並與前一個區塊做鏈接。其中，Python 尤其適用於網站平台後端的開發，能夠非常容易地將 Python 程式碼嵌入到資料庫當中，與前端網站圖形資料庫管理介面連接，讓本地端端的資料透過 Python 程式語言處理後展示在終端機上，方便使用者閱讀。

頻譜資料庫則使用語 Python 相容性非常高且開源的 MySQL 資料庫管理系統，頻譜資料庫內預計將儲存感知無線電裝置資訊、分頻分時分地之頻譜資訊及共享或者交易頻譜的紀錄，並將資料庫設計為關連式資料庫，讓資料表內資料可以互相連結。

3.4.1 頻譜租賃交易資訊

透過本研究實作之動態頻譜接取平台，任何有頻譜使用需求的用戶都可以透過此平台進行頻譜共享及交易租賃，頻譜租賃交易資訊的示意圖如下表3所示，若有次要使用者欲使用160MHz作為中心頻率發送資料的話，其可以利用1000虛擬貨幣購買頻譜使用權，並與頻段執照持有者訂定一「智能合約」[15]，將租賃頻譜的合約配置在區塊鏈當中，由區塊鏈本身自動執行合約內容，確認購買者是否有足夠餘額購買頻段，並檢驗其設備使否符合合約規範，一但頻譜租賃出去後，區塊鏈便會自動記錄此筆交易紀錄，並即時更新資料庫之資料，同時公告給所有節點以維護區塊鏈，其中，在區塊鏈平台上，所有使用者都有自己的公私鑰，私鑰儲存在自身裝置上，而為了可以方便進行頻譜交易及維護區塊鏈平台上的匿名特性，所有裝置的名稱都以公鑰表示，只有主管機關能夠透過其關聯式資料庫得知不同公鑰所代表的裝置或者業者資料。

表 3、頻譜租賃交易資訊

編號	2
頻段持有者	1AWEupx2U34mKvM3p6uvmuWpd8eydZcWYk
所在地(以經緯度表示)	(120.244, 23.402)
頻段	160MHz
頻寬	25KHz
功率大小	10dBm
使用時間	12-18 時
價格	1000
表示單位	M

3.4.2 共享頻段地理資訊

除了上述之頻譜租賃交易資訊，本動態頻譜接取平台也會以地理資訊為基礎提供頻譜共享頻段資訊，如下表 4 所示，透過平台可以看到所在地的共享頻段目前使用狀況及使用資訊，當裝置要使用該頻段做無線通訊應用時，需先像動態頻譜接取平台發送收發資料的請求，頻譜共享的智能合約會自動檢驗發送請求之感知無線電裝置技術規格是否合乎法規標準，認證透過後，該裝置便可以合法使用該共享頻段，同時即時更新如表 4 之頻譜共享資訊，更新正在使用之裝置，並重新計算該地理區域內可以使用共享頻段之相關參數。

除了頻譜交易及租賃會紀錄每一筆交易紀錄外，頻譜共享的使用紀錄依然會被區塊鏈技術登記在每一個節點裝置的分散式帳本當中，且頻譜使用歷史紀錄公開透明，如此一來若有惡意裝置濫用共享頻段作非法應用也可以被輕易定位出來，提供給監理機關進行行政裁罰，並固定以程式碼進行驗證，隨時維護頻譜秩序。

表 4、頻譜共享資訊

所在區域 (以經緯度表示)	(122.402, 25.123)方圓 10 公里內
區域內主要使用者	• 1CS6DgWSnqyc1CTh1CHMid64nmah6rDfWV
正在使用之裝置	• 1Po1oWkD2LmodfkBYiAktwh76vkF93LKnh • 1CL5TbB2MaR4mrFjtYQ5GyA3cP2bSmPxAn
頻段	488MHz
頻寬	30KHz
功率大小	<10dBm
閒置時間	00-08 時

3.4.3 感知無線電裝置資訊

在本研究之動態頻譜接取系統當中，亦會紀錄每一個感知無線電裝置節點的資訊，資訊如表 5 所示，若任何一個沒有納入本平台中的裝置將被視為非法裝置，是不能享有共享及租賃交易頻譜的權益的，對監理機構來說，裝置資訊非常便於監理，所有資訊都在平台上一目瞭然，也因為區塊鏈無法輕易被竄改及造假的關係，監理單位也不必擔心裝置資訊遭到惡意竄改，不論是主要使用者還是次要使用者也不必擔心帳戶內的虛擬貨幣遭到盜用或者憑空消失。

感知無線電裝置的每一筆紀錄都被區塊鏈誠實地記錄在平台上，使用紀錄公開透明，且每一筆紀錄的有其備份在每一個感知無線電裝置上，裝置發射訊號之前也必須先連上資料庫確認是否會造成干擾。

表 5、感知無線電設備裝置資訊

所在地 (以經緯度表示)	(119.402, 21.689)
裝置名稱	1CS6DgWSnqyc1CTh1CHMid64nmah6rDfWV
設備規格	<ul style="list-style-type: none"> • 發送/接收頻率範圍：70MHz-6GHz • 最大取樣率：10.66MS/s • 取樣解析度：12 bits • 支援模式：TX/RX
頻段使用紀錄	<ul style="list-style-type: none"> • 時間：2018-06-07 11:38:00 – 2018-06-07 13:38:21 • 使用頻段：3.8GHz • 功率大小：1.2dBm

3.5 系統整合架構

本研究系統整合的架構圖如下圖 17 所示，將 3.3 小節之感知無線電模組與 3.4 小節與區塊鏈為基礎之動態頻譜接取平台結合，使本研究所設計之感知無線電裝置能夠及時的獲取去中心化動態頻譜接取平台頻譜資訊，即時動態地調整自身裝置的無線電參數，同時將此用資料回饋給平台。

每一個感知無線電裝置都被視為一個節點，內有區塊鏈上所有頻譜共享及交易租賃資訊和所有感知無線電裝置的資訊備份，以達到區塊鏈去中心化及分散式帳本的概念，除非駭客能侵入一半以上的感知無線電裝置，不然沒有任何人能掌控這個動態頻譜接取系統，就算一個節點被入侵破壞，還有其他千千萬萬個節點有著資料的備份，不怕資料流失以及被竄改，平台區塊鏈系統的維護則有每一個節點上的感知無線電裝置共同維護，並可以透過維護區塊鏈賺取虛擬貨幣，可以透過虛擬貨幣租賃購買頻譜使用權。

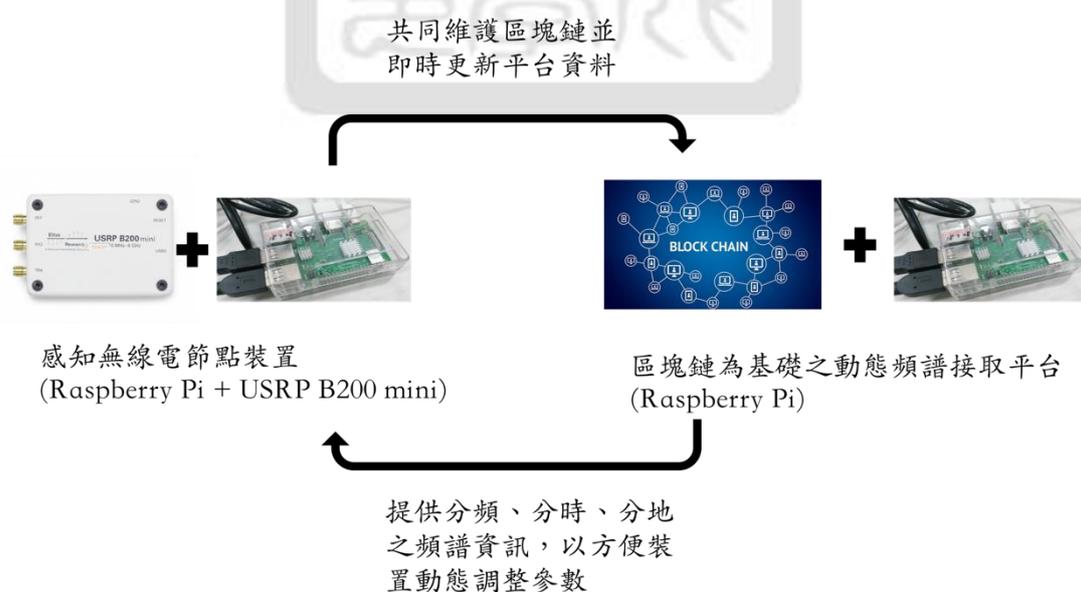


圖 17、系統整合架構圖

第四章 研究結果

4.1 平台架構與實驗流程

本研究目的為透過軟體定義無線電平台實作一符合 IEEE 1900.x 系列國際標準之動態頻譜接取系統，同時設計、實作頻譜資料庫平台，以儲存紀錄不同頻譜資訊及不同裝置資訊。並以區塊鏈概念實作頻譜交易相關機制，以因應未來動態頻譜接取系統下所衍伸之頻譜交易安全、認證的問題及日益漸增之頻譜交易需求。

本研究所使用之硬體裝備設施如下圖 18 所示，微型電腦樹莓派作為動態頻譜接取資料庫平台及驅動軟體無線電之平台，並接上圖片右下角之軟體無線電裝置，USRP B200 mini，用以做為訊號收發及偵測之裝置，圖片上之行動電源供應動態頻譜接取平台所需要之電源，網路接取部分則使用樹莓派內建之無線通訊模組，透過 Wi-Fi 連接上動態頻譜接取資料庫。



圖 18、實驗硬體設備

軟體部分則使用開源之軟體無線電平台，GNU Radio，用以驅動軟體無線電裝置 USRP B200 mini，並以 Python 程式語言實作本研究用到之程式，其中包含切換不同偵測訊號之參數、切換不同發射訊號之參數及調變方式等、以及實作區塊鏈概念的頻譜交易機制。

本研究實作一共分為兩個部分，分別為區塊鏈技術實作頻譜交易機制以及動態頻譜接取系統之實作，頻譜交易機制設計流程如下圖 19 所示。

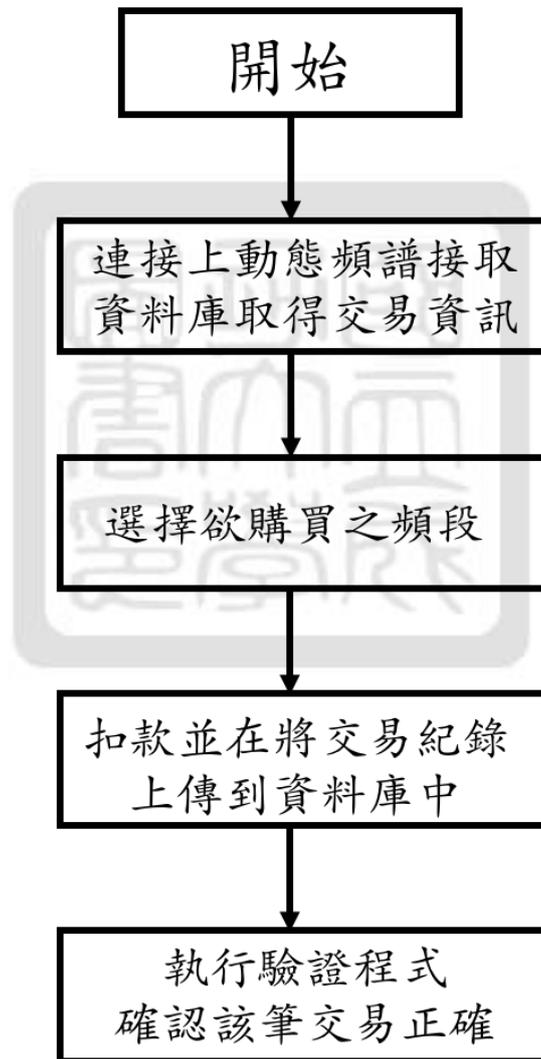


圖 19、頻譜交易流程圖

裝置連接上資料庫後，會使用系統所發放之非對稱加密公私鑰進行頻譜交易，選擇欲購買之頻段後，按下確認即可完成頻譜交易，並在動態頻譜接取資料庫內之頻譜交易紀錄資料表紀錄該筆交易相關資訊，並產生一 Hash 值作為下一筆交易資料之參考，如此一筆資料串一筆資料，行成「鏈」的概念。為了驗證交易正確性以確保交易資料沒有被惡意竄改，會再執行判斷程式，去判斷每筆資料所產生之 Hash 值是否符合資料庫上的資料，詳細程式部分會在後面小節詳述之。

本研究之動態頻譜接取實驗流程如下圖 21 所示，裝置若有傳送的需求，則先針對欲傳輸之頻段進行通道偵測，偵測該頻段附近是否有其他裝置使用中，避免造成不必要的干擾及碰撞，影響其他裝置的權益，若進行通道偵測後確認該裝置附近無其他裝置使用，則發送傳輸請求及裝置相關資訊到動態頻譜接取資料庫進行第二次確認，進行更通盤更完善的干擾判斷，避免裝置通道偵測之盲點，確認沒有如圖 20 所示之「隱藏節點」的問題，圖 20 中，裝置 A 跟裝置 C 都想發射資料給 B，但是 A 跟 C 都互相不在對方的偵測範圍內，若同時發射則為產生碰撞，若過程中有任何干擾的情形發生，則重新進入裝置參數調整階段，重新選擇發射之頻率、頻寬、功率大小、調變方式等。

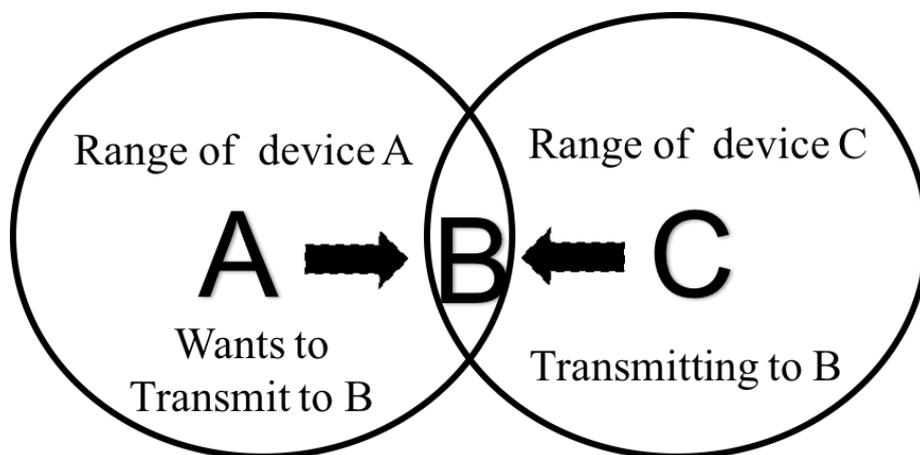


圖 20、隱藏節點問題

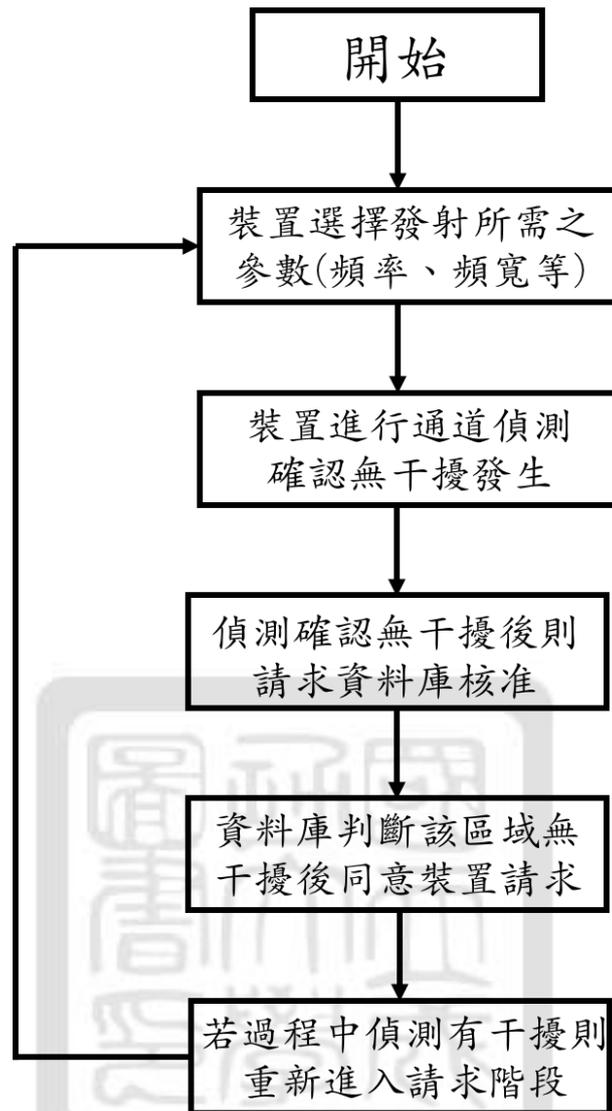


圖 21、動態頻譜接取實驗流程

4.2 資料庫實作與設計

4.2.1 硬體及軟體使用

本研究使用之作業系統為執行在樹莓派上之 Linux 系統，網頁伺服器則使用 Apache HTTP Server，本研究之動態頻譜接取資料庫則使用開源資料庫 MariaDB，使用的程式語言為 Python，Python 程式驅動軟體無線電裝置以及進行區塊鏈頻譜交易的程式邏輯設計。

本研究設計之動態頻譜接取資料庫建置在微型電腦樹莓派上，並讓資料庫運行在有線網路固定 IP 底下（本研究使用之固定 IP 地址為 140.116.54.153），方便其他裝置透過網際網路進行登入資料庫以進行資料接取，同時使用 phpMyAdmin 作為圖形化資料庫管理介面，方便管理者以網站的方式使用與操作資料庫。

4.2.2 資料欄位設計

本研究動態頻譜接取平台實際網頁管理介面如下圖 22 所示，圖 22 為上述之圖形化資料庫管理介面 phpMyAdmin，所設計之資料庫名為 DSA，代表動態頻譜接取的縮寫，本研究一共設計十個資料表，儲存本研究動態頻譜接取平台所用到的資料，使用者可以透過設定之帳號密碼，透過網際網路連線操作，介面操作為圖形化介面，對使用者來說十分友善，也方便未來若監理機構採用相關圖形化介面進行頻譜資源的監管。

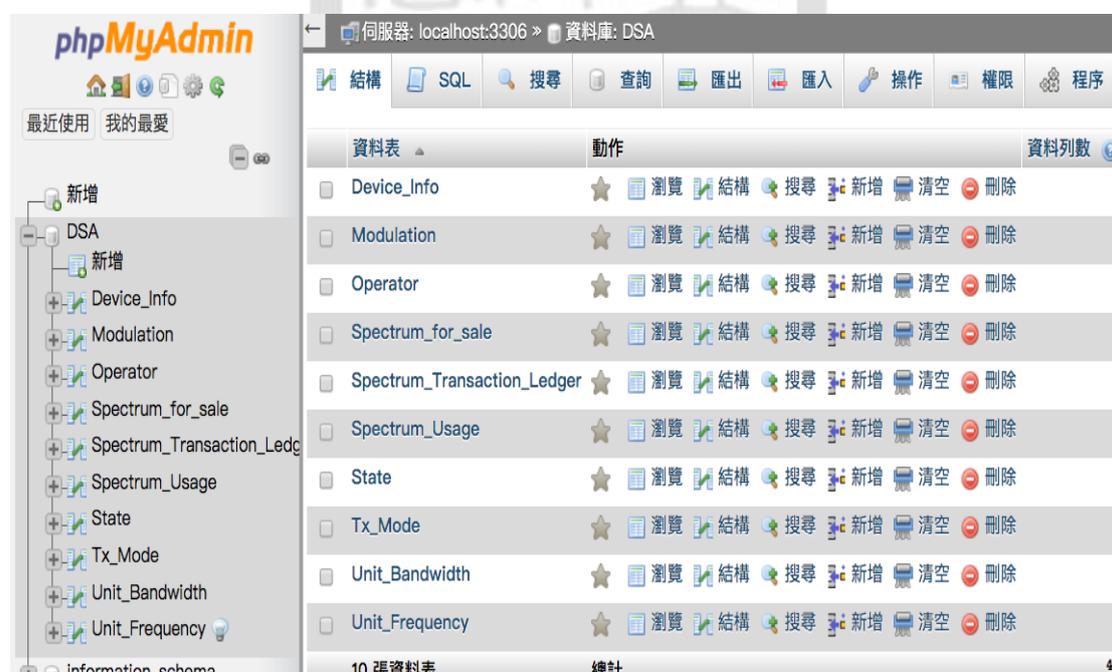


圖 22、phpMyAdmin 管理者介面

資料表之名稱以及資料表的功能如下表 6 所示，資料表儲存軟體無線

電裝置資訊、業者名稱、頻譜交易資訊、頻譜交易紀錄、頻譜即時使用狀況等資訊。

表 6、DSA 資料庫之資料表名稱及功能

資料表名稱	功能
Device_Info	儲存各個軟體無線電裝置的資訊，包含公鑰、裝置擁有者、經緯度資訊、目前狀態、餘額、偵測資訊及裝置參數。
Spectrum_for_sale	存放所有欲販售之頻段的相關交易資訊，如頻率、頻寬、價格、頻段地點等。
Spectrum_Transaction_Ledger	儲存每一筆頻譜交易紀錄，紀錄前一筆交易 Hash 值、買家賣家、頻段、頻寬、時間戳記等，最後會生成該筆資料的 Hash 值。
Spectrum_Usage	紀錄目前各個頻段之即時使用資訊，包含頻率、使用地點經緯度、頻率擁有者、頻寬、功率大小、調變方式、傳輸模式等。
Operator	紀錄各家業者的資訊，如公司中英文名稱、負責人名稱、負責電話、營業地址、統一編號等。
Modulation	儲存調變方式(Ex. FM、FSK、GMSK、QAM、OFDM 等)
Tx_Mode	儲存傳輸模式(Ex. 單方向發射、廣播等)
State	儲存裝置目前狀態(Ex. 發射中、閒置中等)
Unit_Bandwidth	儲存頻寬單位(K、M、G)
Unit_Frequency	儲存頻率單位(K、M、G)

資料表之細部欄位設計如下圖 23 以及圖 24 所示，圖 23 所代表的資料表為本研究設計之動態頻譜接取資料庫主要使用要之資料，圖上鑰匙圖案代表為主鍵，主鍵所連接之資料表則如圖 24 表示。

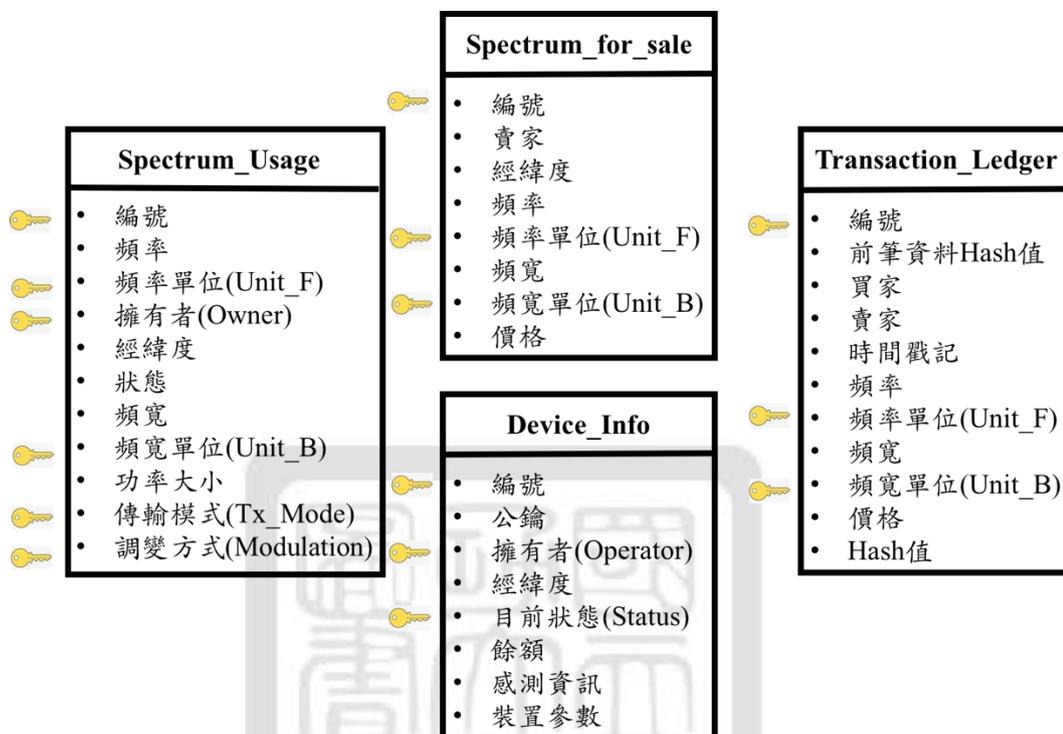


圖 23、主要使用之資料表

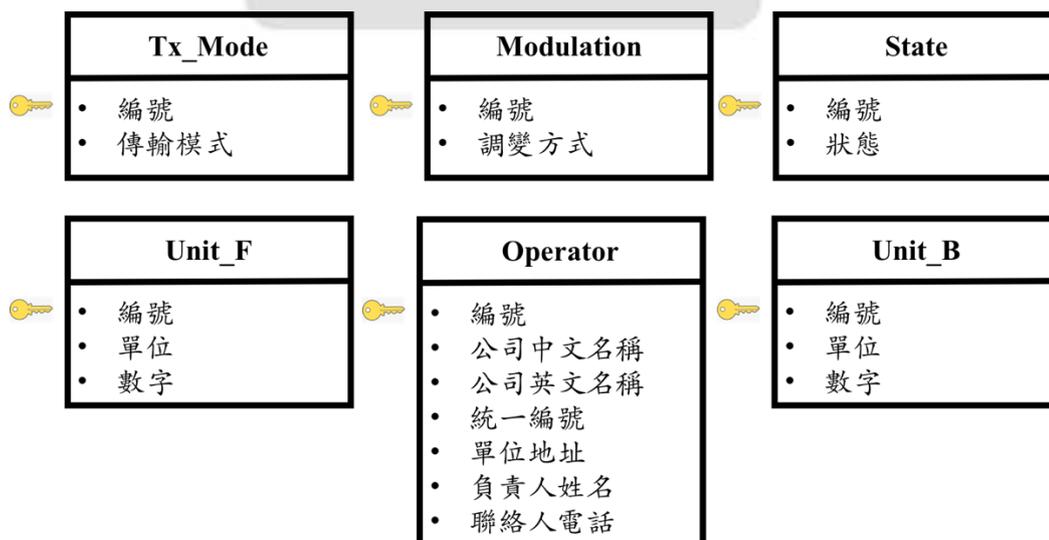


圖 24、主鍵所連接之資料表

4.3 頻譜交易機制

本節將介紹本研究所設計之以區塊鏈概念為核心之頻譜交易機制，將細分為兩部分，第一部分將介紹如何實作區塊鏈相關概念，第二部分將敘述如何利用 Hash 值去驗證每筆交易的真實性。

4.3.1 區塊鏈概念實作

頻譜交易區塊鏈概念本研究以 Python 程式語言撰寫，裝置若有頻譜購買需求，則須先連到本研究設計之頻譜資料庫，讀取頻譜交易資訊資料，以便得知有何頻譜資源可供購買，連上資料庫後會如圖 25 之圖形化介面所示，本研究之圖形化介面是由 Python 內建之圖形化介面模組 tkinter 所編寫，圖 25 上方列出頻譜交易資料庫當中每一筆欲出售之頻譜資料，每筆交易的資料包含交易編號、販售者、頻段使用地點、頻段之中心頻率、頻寬以及價格，以方便使用者進行閱覽及選取，同時在視窗上方顯示使用者目前所擁有之餘額，選取欲購買之頻段資源後便可進行交易。

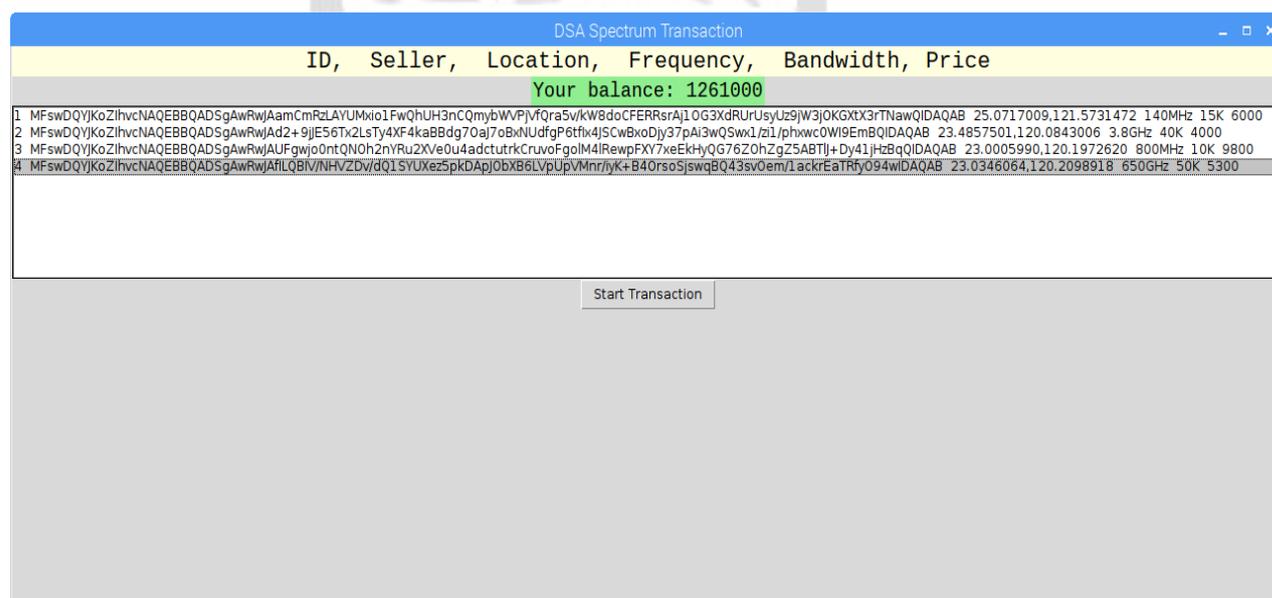


圖 25、頻譜購買介面

如圖 26 所示，當使用者選取欲購買之頻段後(以圖片 26 為例為編號 4 之頻譜資源)，按下圖 25 下方之開始交易的按鈕後，便會跳出提示視窗，詢問使用者是否確認購買該編號之頻譜資源，若使用者按下確認按鈕後便會進行交易，如圖 27 所示，產生交易成功的提示訊息，餘額也會更新成交易後的餘額，而在交易完成的同時，頻譜交易系統也將該筆交易資料訊息記錄在頻譜交易記錄中，並將所有買賣雙方的交易資料融入上一筆交易資料的 Hash 值及發生交易當下的時間戳記後進行雜湊運算，產生一 Hash 值，此 Hash 值及代表本筆交易的所有內容，獨一無二。

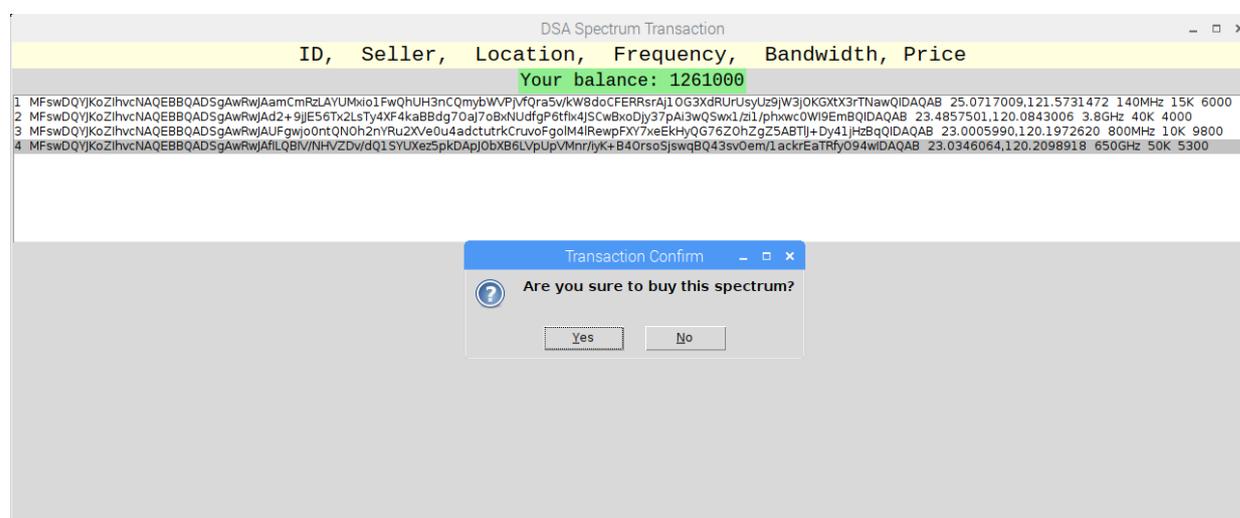


圖 26、頻譜購買介面(購買確認彈跳視窗)

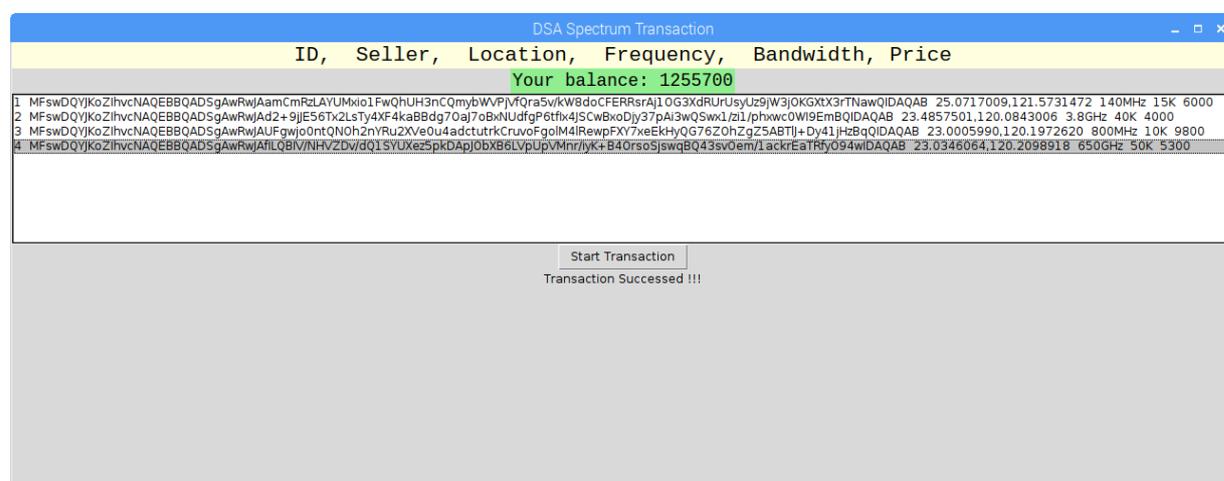
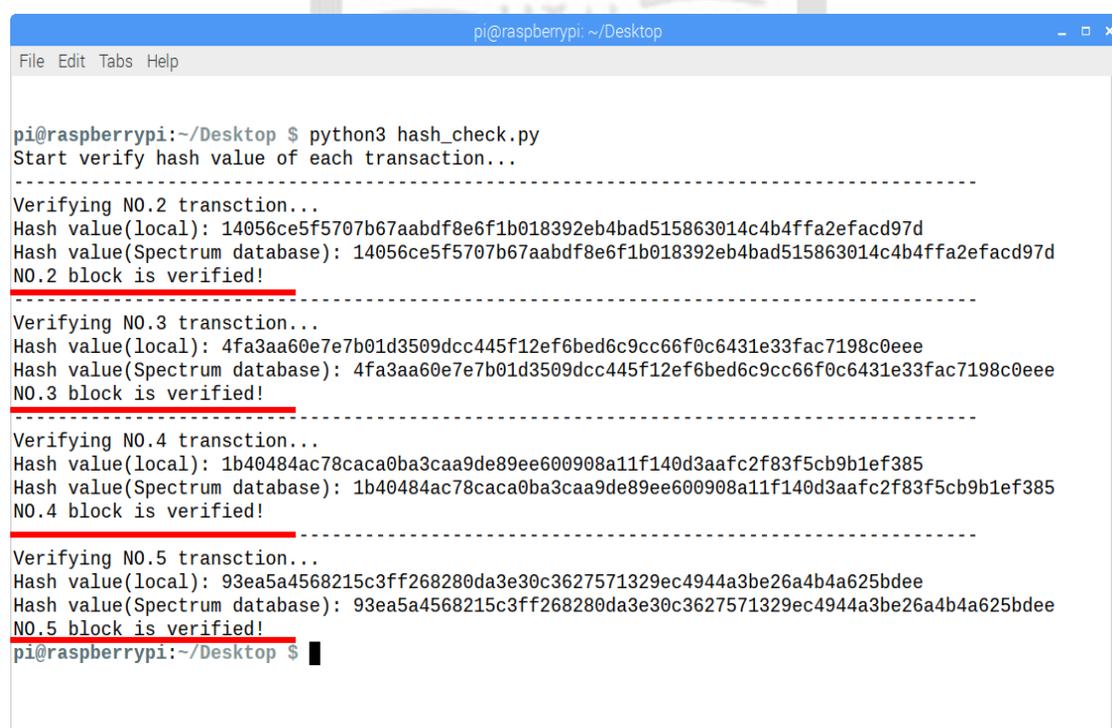


圖 27、頻譜交易介面(交易成功)

4.3.2 交易驗證

當交易完成後，會將該筆交易所有資訊彙整成一個 Hash 值，所匯入資訊包含，編號、前一筆資料之 Hash 值、買家、賣家、時間戳記、頻率、頻寬、價格等，一筆一筆按照時間先後順序上傳到資料表中將每筆資料互相串鏈。

而為了驗證交易的真實性，每台裝置都可以透過分散式帳本來驗證交易的正確，透過將自己所持有之帳本與資料庫上的帳本相互驗證，來確保交易的真實性，如圖 28 所示，程式會先抓取資料庫中每筆交易的 Hash 值進行比對，若與自己裝置內所生成之 Hash 值不同則代表該筆交易有問題，有被竄改過之疑慮，而因為區塊鏈特性的關係，每一筆資料的 Hash 值都引用參考上一筆資料的 Hash 值，因此能輕易比對出問題，若要竄改內部資料就等於要連前面幾筆資料一起竄改，所耗費之成本極高。



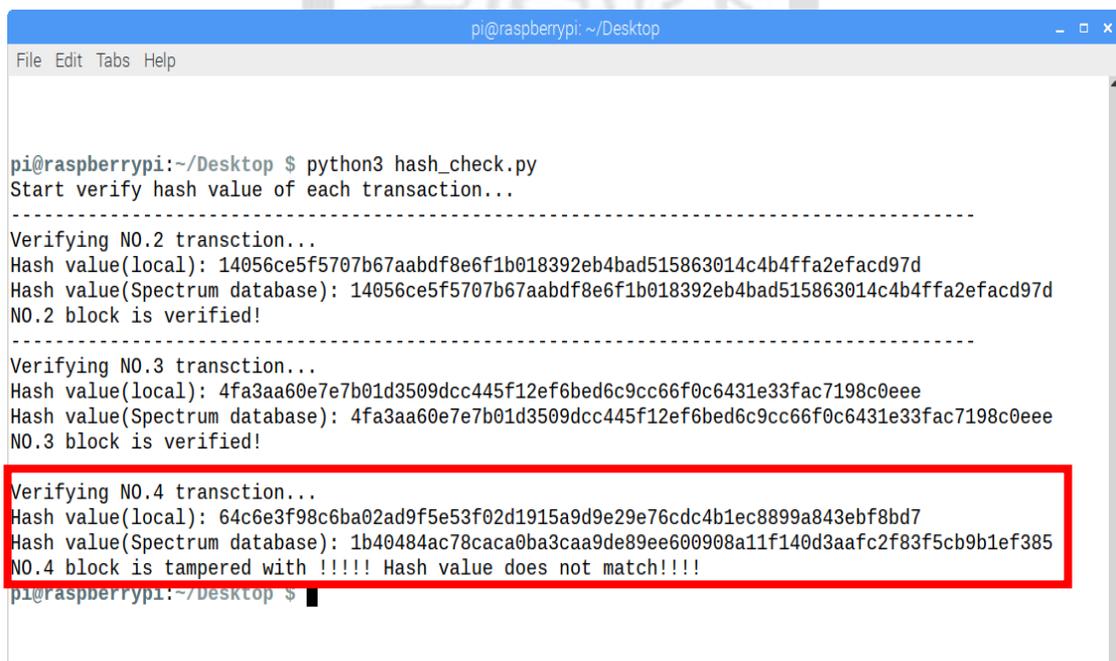
```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help

pi@raspberrypi:~/Desktop $ python3 hash_check.py
Start verify hash value of each transaction...
-----
Verifying NO.2 transction...
Hash value(local): 14056ce5f5707b67aabdf8e6f1b018392eb4bad515863014c4b4ffa2efacd97d
Hash value(Spectrum database): 14056ce5f5707b67aabdf8e6f1b018392eb4bad515863014c4b4ffa2efacd97d
NO.2 block is verified!
-----
Verifying NO.3 transction...
Hash value(local): 4fa3aa60e7e7b01d3509dcc445f12ef6bed6c9cc66f0c6431e33fac7198c0eee
Hash value(Spectrum database): 4fa3aa60e7e7b01d3509dcc445f12ef6bed6c9cc66f0c6431e33fac7198c0eee
NO.3 block is verified!
-----
Verifying NO.4 transction...
Hash value(local): 1b40484ac78caca0ba3caa9de89ee600908a11f140d3aafc2f83f5cb9b1ef385
Hash value(Spectrum database): 1b40484ac78caca0ba3caa9de89ee600908a11f140d3aafc2f83f5cb9b1ef385
NO.4 block is verified!
-----
Verifying NO.5 transction...
Hash value(local): 93ea5a4568215c3ff268280da3e30c3627571329ec4944a3be26a4b4a625bdee
Hash value(Spectrum database): 93ea5a4568215c3ff268280da3e30c3627571329ec4944a3be26a4b4a625bdee
NO.5 block is verified!
pi@raspberrypi:~/Desktop $
```

圖 28、驗證交易介面(正常情況)

若將資料庫某筆資料更動後執行驗證程式檔，經由程式驗證後會發現本地資料庫內的 Hash 值與頻譜資料庫上該筆交易的 Hash 值有所出入，因此判斷出該筆交易有被竄改過，則驗證結果會如圖 29 所顯示，在終端機介面顯示 Hash 值不相吻合的訊息，程式也會中斷，因為區塊鏈特性的關係，只要一筆資料有問題，後面每一筆資料也一定會有問題，不必特別花費電腦效能及時間去驗證後面的交易，這也正是區塊鏈所謂「鏈」的特性，讓每筆交易紀錄相互連結，以保障全部裝置的權益。

同時也因為 Hash 雜湊函數的特性，只要資料內容稍微更動過，哪怕只是多一個空白符號，Hash 值就會差異非常大，肉眼就可以輕易辨識出來，讓驗證交易變得非常輕鬆，只需要比對 Hash 值即可，不需要特別比對每一筆交易的細部資料，而若真的每一筆資料都遭竄改，每一筆 Hash 值都經過重新計算，在能夠確保 51% 以上的感知無線電裝置節點不受到入侵的情況下，區塊鏈分散式帳本的特性讓每一台裝置都有交易紀錄的備份，因此能夠確保整個區塊鏈頻譜交易的安全性。



```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help

pi@raspberrypi:~/Desktop $ python3 hash_check.py
Start verify hash value of each transaction...
-----
Verifying NO.2 transaction...
Hash value(local): 14056ce5f5707b67aabdf8e6f1b018392eb4bad515863014c4b4ffa2efacd97d
Hash value(Spectrum database): 14056ce5f5707b67aabdf8e6f1b018392eb4bad515863014c4b4ffa2efacd97d
NO.2 block is verified!
-----
Verifying NO.3 transaction...
Hash value(local): 4fa3aa60e7e7b01d3509dcc445f12ef6bed6c9cc66f0c6431e33fac7198c0eee
Hash value(Spectrum database): 4fa3aa60e7e7b01d3509dcc445f12ef6bed6c9cc66f0c6431e33fac7198c0eee
NO.3 block is verified!
-----
Verifying NO.4 transaction...
Hash value(local): 64c6e3f98c6ba02ad9f5e53f02d1915a9d9e29e76cdc4b1ec8899a843ebf8bd7
Hash value(Spectrum database): 1b40484ac78caca0ba3caa9de89ee600908a11f140d3aaafc2f83f5cb9b1ef385
NO.4 block is tampered with !!!!! Hash value does not match!!!!
pi@raspberrypi:~/Desktop $
```

圖 29、驗證交易介面(異常狀況)

4.4 動態頻譜接取概念設計與實作

本研究第二個實作的部分則為應用軟體定義無線電實作動態頻譜接取系統，本研究中將樹莓派接上軟體無線電裝置後成為感知無線電裝置，透過其能夠偵測環境頻譜狀態的功能，動態調整自身參數，同時連上頻譜資料庫進行傳送請求，由資料庫做更全面更通盤的認證，避免裝置之間的偵測盲點，造成干擾的情形，希冀能夠過動態頻譜接取平台達到頻譜資源使用的最佳化。

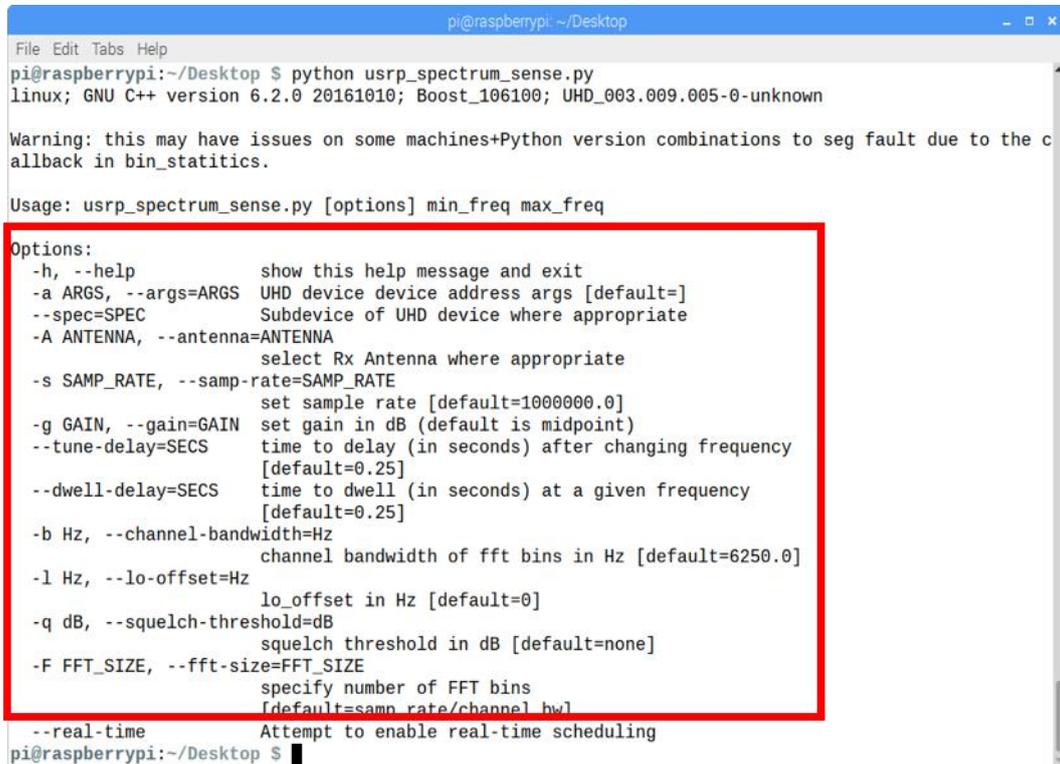
以下小節將詳述本研究根據 IEEE 1900.x 系列標準所實作之動台頻譜接取系統，並透過前一小節所設計之動態頻譜資料庫做介接，達到軟體無線電裝置連上資料庫確認後做資料即時發射的效果。

4.4.1 通道偵測

任無線電裝置發射訊號前都應先偵測是否會干擾到其他使用者，試想若民航頻道或者警消頻道受到不明訊號干擾，將可能會造成多大的傷害，是所有人都不想樂見的，為了避免碰撞，無線電資源使用是有相關法規規範的，需要持有主管機關的執照才能進行訊號發射，因此通道偵測的角色至關重要。

通道偵測部分，本研究使用 USRP B200 mini 作為軟體定義無線電之收發裝置，並接上樹莓派的 USB 接口，以進行裝置的驅動以及供電，程式則使用 USRP 官方所提供的通道偵測 Python 程式，因次這部分並沒有在第三章介紹 GNU Radio 時看到的圖形化介面，而是透過直接修改程式碼進行參數修改，如圖 30 所示，可以看到此通道偵測程式所詳列之可以調整參數，以本研究所會調整的參數舉例，參數 `-s` 所代表的意涵為取樣頻率 (Sampling Rate)、參數 `-g` 所代表的意涵為天線增益值 (Gain)、參數 `-b` 所

代表的意涵為頻寬(Bandwidth)，若執行程式時不特別在後面加上設定參數，則以上所述的參數都會是預設值，唯獨偵測範圍的最小頻率以及最大頻率需要在執行程式時就先輸入，否則通道偵測的程式將無法被啟動。



```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python usrp_spectrum_sense.py
linux; GNU C++ version 6.2.0 20161010; Boost_106100; UHD_003.009.005-0-unknown

Warning: this may have issues on some machines+Python version combinations to seg fault due to the c
allback in bin_statistics.

Usage: usrp_spectrum_sense.py [options] min_freq max_freq

Options:
-h, --help                show this help message and exit
-a ARGS, --args=ARGS      UHD device device address args [default=]
--spec=SPEC              Subdevice of UHD device where appropriate
-A ANTENNA, --antenna=ANTENNA
                          select Rx Antenna where appropriate
-s SAMP_RATE, --samp-rate=SAMP_RATE
                          set sample rate [default=1000000.0]
-g GAIN, --gain=GAIN      set gain in dB (default is midpoint)
--tune-delay=SECS        time to delay (in seconds) after changing frequency
                          [default=0.25]
--dwell-delay=SECS       time to dwell (in seconds) at a given frequency
                          [default=0.25]
-b Hz, --channel-bandwidth=Hz
                          channel bandwidth of fft bins in Hz [default=6250.0]
-l Hz, --lo-offset=Hz    lo_offset in Hz [default=0]
-q dB, --squelch-threshold=dB
                          squelch threshold in dB [default=none]
-F FFT_SIZE, --fft-size=FFT_SIZE
                          specify number of FFT bins
                          [default=samp_rate/channel_bw]
--real-time              Attempt to enable real-time scheduling
pi@raspberrypi:~/Desktop $
```

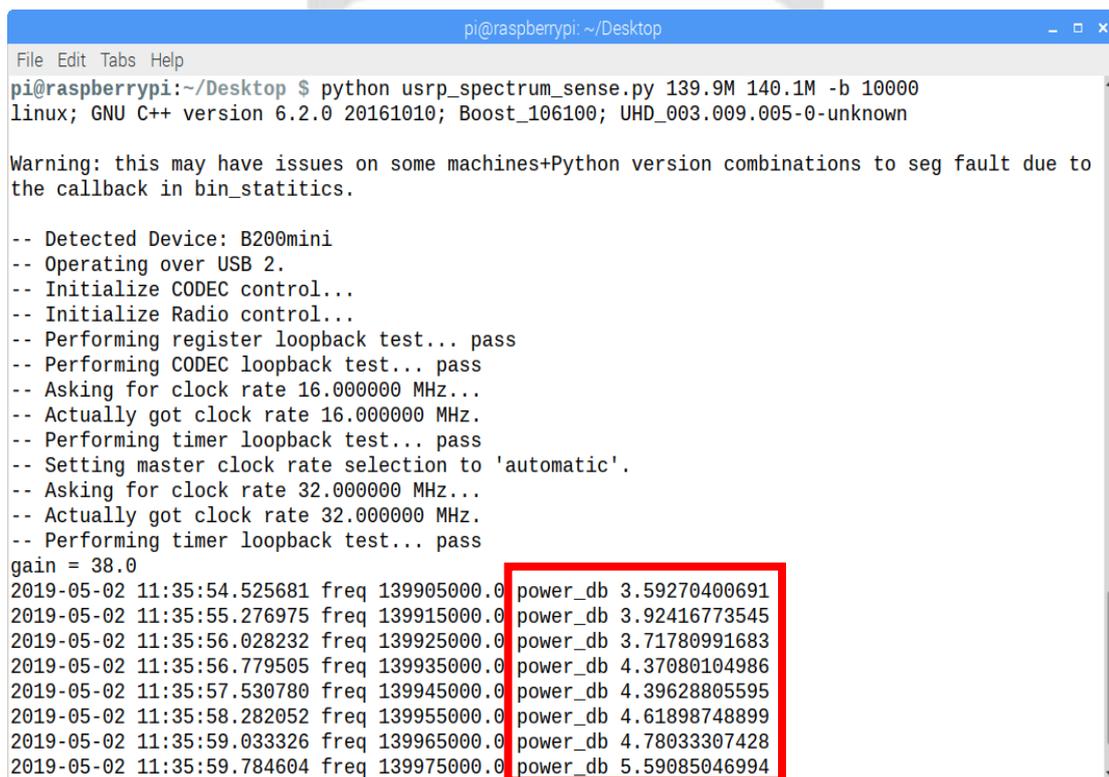
圖 30、通道偵測程式之參數調整

本研究所選定之發射中心頻率為 140MHz，因此發射訊號前須先針對中心頻率 140MHz 進行通道偵測，以確保該頻道無人使用，避免裝置之間的相互干擾，因此調整參數為讓通道偵測程式在 139.9MHz 到 140.1MHz 範圍之間進行通道偵測，頻寬大小則設定為 10KHz，偵測間隔則設定為每 0.75 秒偵測一次，偵測功率大小為預設值 38dB，取樣頻率亦為預設值 1MHz，偵測所選定頻率範圍之內的收到訊號的功率大小。

如圖 31、32 所示，執行程式後便可以在終端機介面看到偵測到的值 (power_db)，以圖 31、32 為例，偵測的當下周遭並沒有其他裝置正在發射訊號，因此偵測到的功率大小值 (power_db) 普遍落在 10 以下，並在

裝置旁邊以另一台筆電執行 GNU Radio 的接收程式以相同的中心頻率進行訊號接收，以驗證所執行的通道偵測程式是正常的，以圖 32 之接收頻譜圖驗證。

而若偵測到有訊號，則所偵測到的功率大小值皆高於 25 以上，如圖 33、34 所示，偵測過程中使用另一台軟體無線電裝置以中心頻率 140MHz 發射訊號，用以模擬裝置旁邊真的其他裝置正在使用相同頻率進行訊號的發射，並以圖 32 之頻譜圖驗證訊號是真的有被發射，因此裝置所執行之通道偵測程式便會偵測到訊號，功率值則明顯比沒有發射訊號時高，此時若裝置發射訊號則容易造成不同裝置間干擾碰撞的問題。



```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python usrp_spectrum_sense.py 139.9M 140.1M -b 10000
linux; GNU C++ version 6.2.0 20161010; Boost_106100; UHD_003.009.005-0-unknown

Warning: this may have issues on some machines+Python version combinations to seg fault due to
the callback in bin_statitics.

-- Detected Device: B200mini
-- Operating over USB 2.
-- Initialize CODEC control...
-- Initialize Radio control...
-- Performing register loopback test... pass
-- Performing CODEC loopback test... pass
-- Asking for clock rate 16.000000 MHz...
-- Actually got clock rate 16.000000 MHz.
-- Performing timer loopback test... pass
-- Setting master clock rate selection to 'automatic'.
-- Asking for clock rate 32.000000 MHz...
-- Actually got clock rate 32.000000 MHz.
-- Performing timer loopback test... pass
gain = 38.0
2019-05-02 11:35:54.525681 freq 139905000.0 power_db 3.59270400691
2019-05-02 11:35:55.276975 freq 139915000.0 power_db 3.92416773545
2019-05-02 11:35:56.028232 freq 139925000.0 power_db 3.71780991683
2019-05-02 11:35:56.779505 freq 139935000.0 power_db 4.37080104986
2019-05-02 11:35:57.530780 freq 139945000.0 power_db 4.39628805595
2019-05-02 11:35:58.282052 freq 139955000.0 power_db 4.61898748899
2019-05-02 11:35:59.033326 freq 139965000.0 power_db 4.78033307428
2019-05-02 11:35:59.784604 freq 139975000.0 power_db 5.59085046994
```

圖 31、實際進行通道偵測資料(無偵測到訊號)

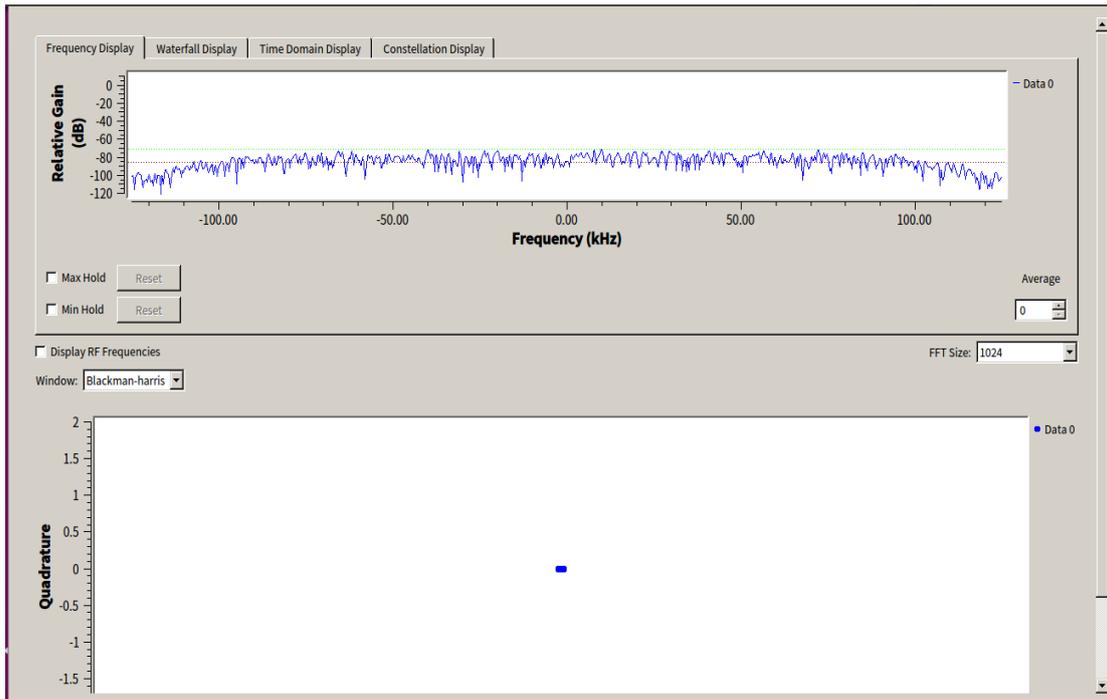


圖 32、實際進行通道偵測資料(無偵測到訊號)頻譜圖

```

pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop $ python usrp_spectrum_sense.py 139.9M 140.1M -b 10000
linux; GNU C++ version 6.2.0 20161010; Boost_106100; UHD_003.009.005-0-unknown

Warning: this may have issues on some machines+Python version combinations to seg fault due to
the callback in bin_statistics.

-- Detected Device: B200mini
-- Operating over USB 2.
-- Initialize CODEC control...
-- Initialize Radio control...
-- Performing register loopback test... pass
-- Performing CODEC loopback test... pass
-- Asking for clock rate 16.000000 MHz...
-- Actually got clock rate 16.000000 MHz.
-- Performing timer loopback test... pass
-- Setting master clock rate selection to 'automatic'.
-- Asking for clock rate 32.000000 MHz...
-- Actually got clock rate 32.000000 MHz.
-- Performing timer loopback test... pass
gain = 38.0
2019-05-02 11:37:49.672188 freq 139905000.0 power_db 2.9095978526
2019-05-02 11:37:50.423475 freq 139915000.0 power_db 3.3716916485
2019-05-02 11:37:51.174760 freq 139925000.0 power_db 3.08791726876
2019-05-02 11:37:51.926045 freq 139935000.0 power_db 7.94910503625
2019-05-02 11:37:52.676633 freq 139945000.0 power_db 17.906202304
2019-05-02 11:37:53.428183 freq 139955000.0 power_db 26.9284990841
2019-05-02 11:37:54.179532 freq 139965000.0 power_db 30.9093616709
2019-05-02 11:37:54.930786 freq 139975000.0 power_db 32.2796204491

```

圖 33、實際進行通道偵測資料(有偵測到訊號)

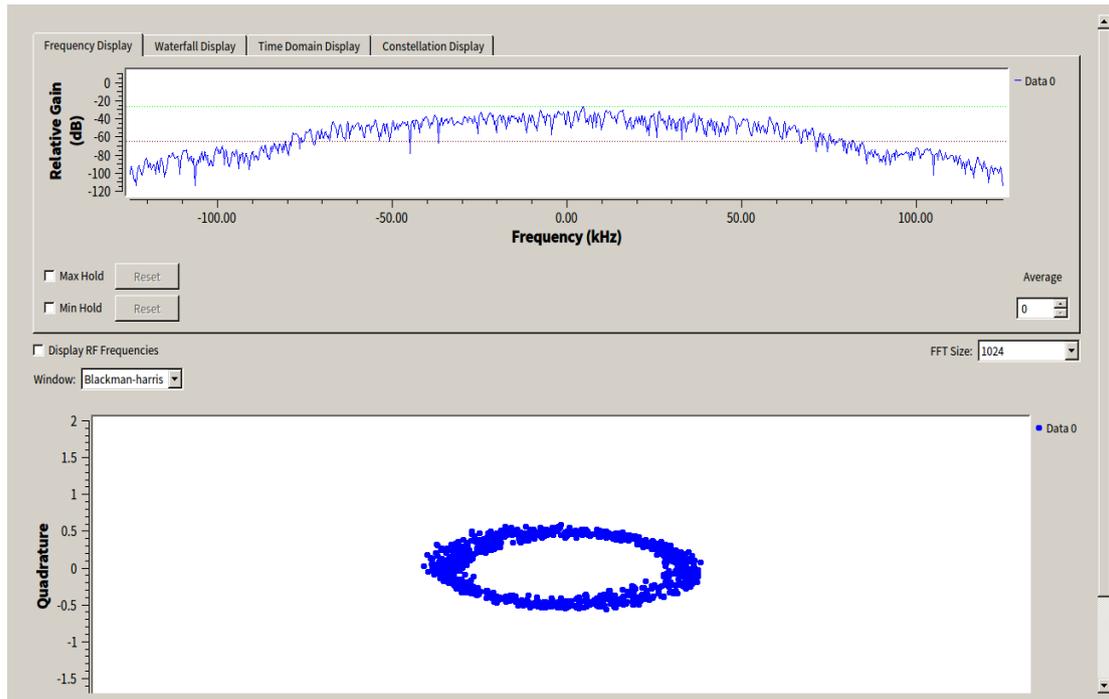


圖 34、實際進行通道偵測資料(有偵測到訊號)頻譜圖

4.4.2 偵測通道後發射資料

在進行前一小節所述之某選定頻率之通道偵測後，軟體定義無線電裝置便可以選定該頻率進行資料發射，並可以依需求選擇調變方式，以發揮感知無線電的精神，偵測環境參數，依據偵測到的參數自身調整參數。本研究判斷環境是否有干擾的依據為所偵測到的功率大小值是否有高於 25 dBm，若大於 25 dBm 則判斷周遭有其他使用者，因此終止發射訊號的程式，反之偵測到低於 25dBm，則判斷附近無其他使用者，不會造成干擾，因此發出傳送的請求的資料庫，傳送請求包含裝置本身所在的經緯度資訊、欲發射的中心頻率、頻寬大小、發射功率、取樣頻率、發射模式、調變方式等，交由資料庫進行最後的確認，若資料庫認證沒有干擾則進行發射，裝置便可以進行訊號發射，反之若資料庫判斷有干擾則駁回請求，程式終止，感知無線電裝置進入原待機狀態。

本研究亦以前述之 Python 圖形化介面作為使用者選擇調變方式介面之設計與實作，如圖 35、圖 36 所示，使用者可以選擇欲使用之調變方式，按下傳送按鈕後便可以進行環境干擾偵測，偵測沒有干擾後便會將請求上傳至頻譜資源資料庫，由頻譜資源資料庫進行最終之干擾確認，若確認裝置發射之訊號不會造成干擾，便同意該裝置之發射請求，裝置便可以馬上進行訊號發射，當訊號發射完成後，使用者可以在如圖下方看到上次發射之記錄，以供使用者參照，若感知無線電裝置的訊號發送請求因為被判斷為會造成干擾而拒絕發射請求，則程式回到如圖最初之調變方式選擇畫面，需重新選擇其他種不會造成干擾的調變方式或者其他發射參數。

圖 37 為裝置請求以 GMSK 調變方式進行訊號發射，圖 38 則為裝置請求以 OFDM 方式進行訊號發射，並將參數資料送進資料庫進行最終干擾判斷，確認是否可以進行訊號發射。

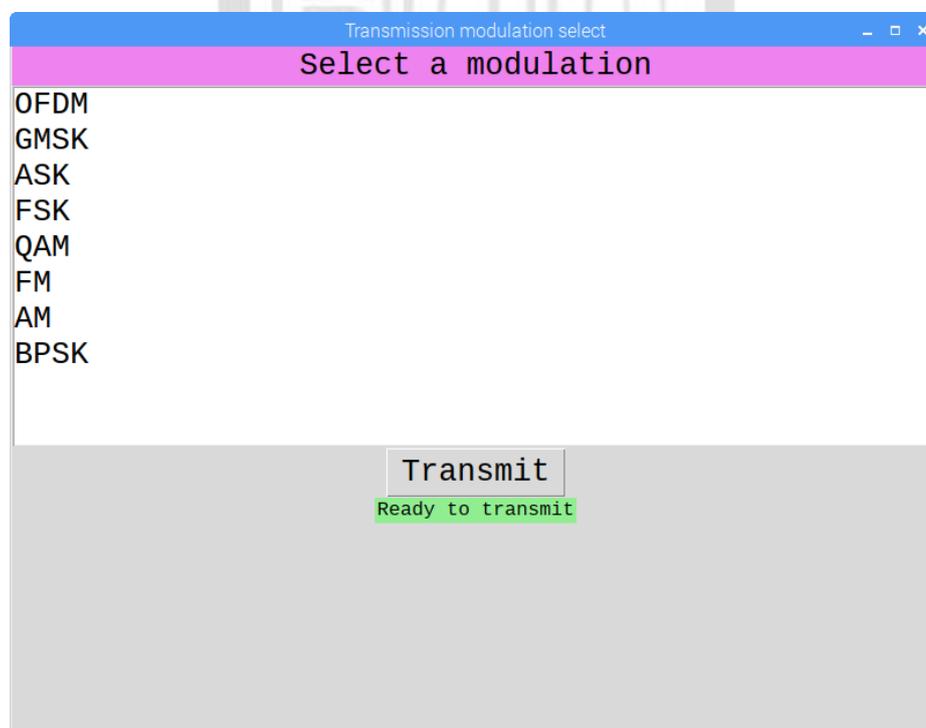


圖 35、調變方式選擇之圖形化介面(準備畫面)

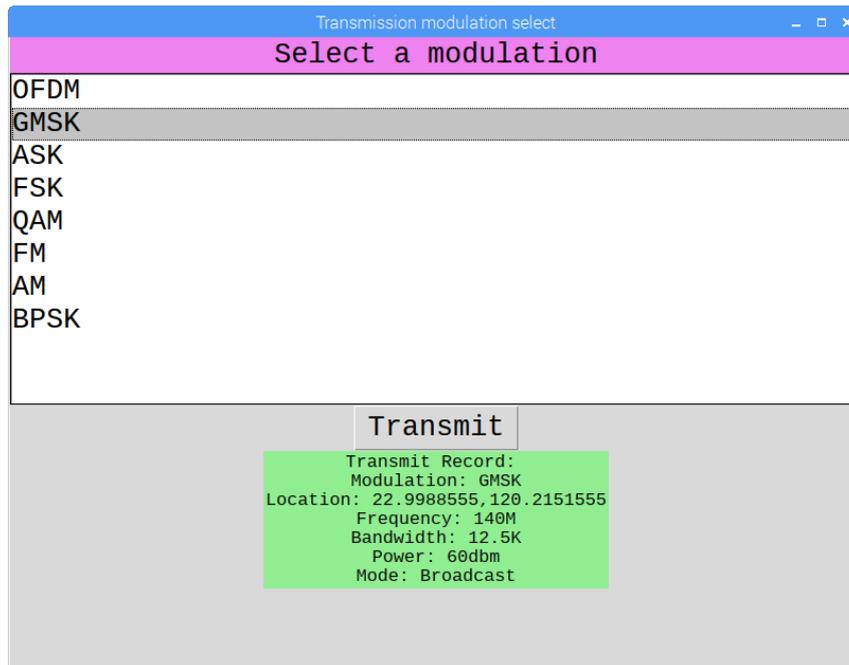


圖 36、調變方式選擇之圖形化介面(發射後)

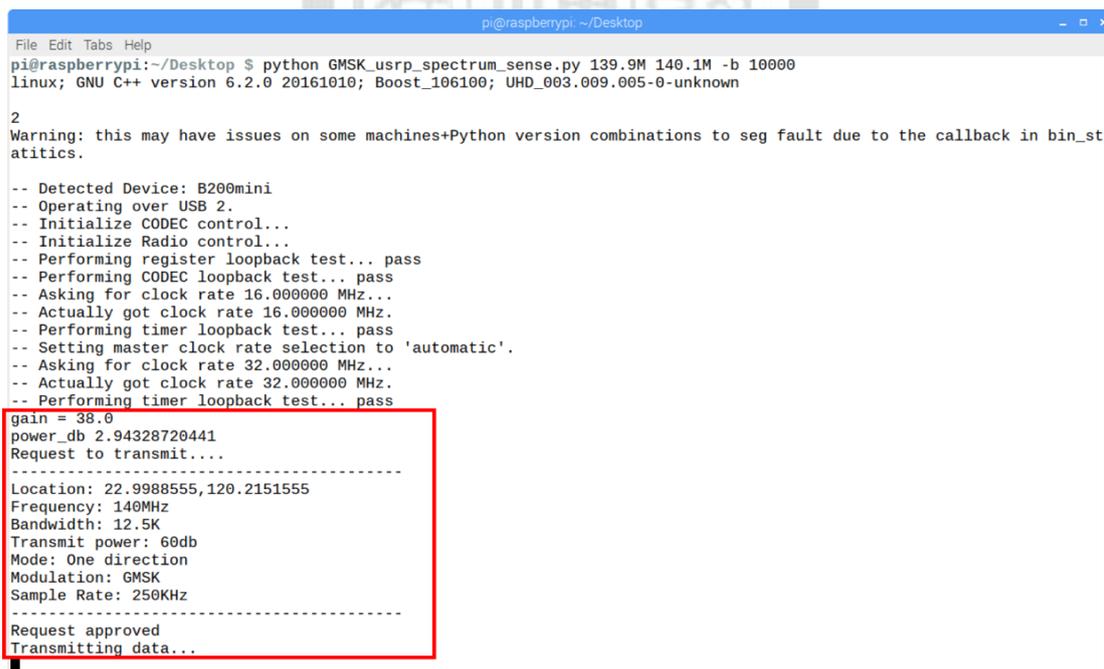


圖 37、裝置發出傳送請求(GMSK)-同意

```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help

pi@raspberrypi:~/Desktop $ python OFDM_usrp_spectrum_sense.py 139.9M 140.1M -b 10000
linux; GNU C++ version 6.2.0 20161010; Boost_106100; UHD_003.009.005-0-unknown

Warning: this may have issues on some machines+Python version combinations to seg fault due to the callback in bin_st
atictics.

-- Detected Device: B200mini
-- Operating over USB 2.
-- Initialize CODEC control...
-- Initialize Radio control...
-- Performing register loopback test... pass
-- Performing CODEC loopback test... pass
-- Asking for clock rate 16.000000 MHz...
-- Actually got clock rate 16.000000 MHz.
-- Performing timer loopback test... pass
-- Setting master clock rate selection to 'automatic'.
-- Asking for clock rate 32.000000 MHz...
-- Actually got clock rate 32.000000 MHz.
-- Performing timer loopback test... pass

gain = 38.0
power_db 4.26718791557
Request to transmit....
-----
Location: 22.9988555,120.2151555
Frequency: 140MHz
Bandwidth: 12.5K
Transmit power: 60db
Mode: Broadcast
Modulation: OFDM
-----
Request deny!!!
pi@raspberrypi:~/Desktop $
```

圖 38、裝置發出傳送請求(OFDM)-拒絕

4.4.3 資料庫干擾判斷

延續前一小節所述，光是裝置本身判斷周遭是否有干擾是不夠的，須透過一個儲存所有地區裝置資訊的資料庫進行全面通盤的確認，避免隱藏節點的問題，在本研究所設計的動態頻譜接取資料庫中，其中一個資料表為 Spectrum_Usage，該資料表內紀錄所有頻段即使的使用狀況，包含經緯度位置、中心頻率、頻寬、功率大小、傳輸模式、調變方式、該頻段目前使用狀態等，以提供資料庫進行通盤的干擾判斷。

若資料庫判斷裝置的傳送並不會造成干擾，則會同意裝置的傳送請求，裝置收到同意訊息後，便可以發送訊號，如圖 37 所示。反之，若資料庫判斷裝置的傳送會造成干擾，影響其他合法使用者則會拒絕裝置的傳送請求，裝置收到資料庫的否定訊息後則中斷程式，如圖 38 所示。

圖 39、圖 40 以及圖 41，為本研究所設定之發射資料的 GNURadio 方塊圖，圖 39 為 GMSK 調變方式、圖 40 以及圖 41 則為 OFDM 傳送(子載波調變方式為 QPSK 調變)，發射的中心頻率皆為 140MHz，發射裝置皆為 USRP B200 mini，軟體皆運行在樹莓派當中。

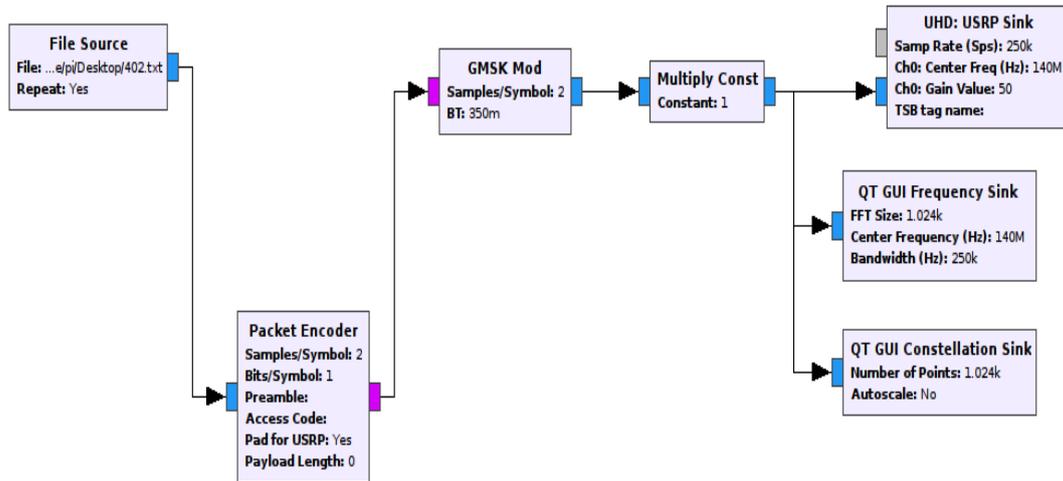


圖 39、GMSK 方塊圖

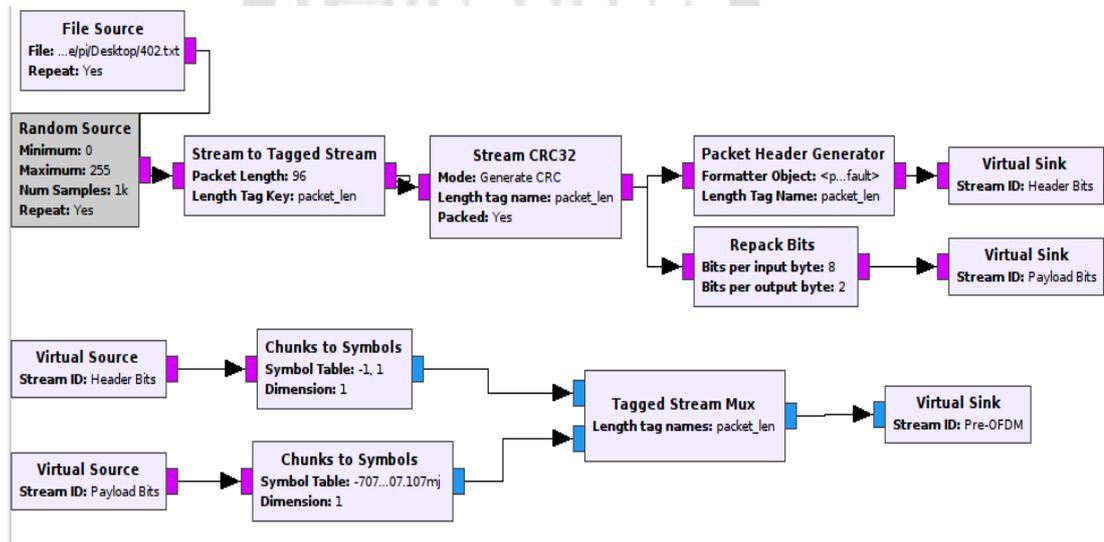


圖 40、OFDM 方塊圖(1)

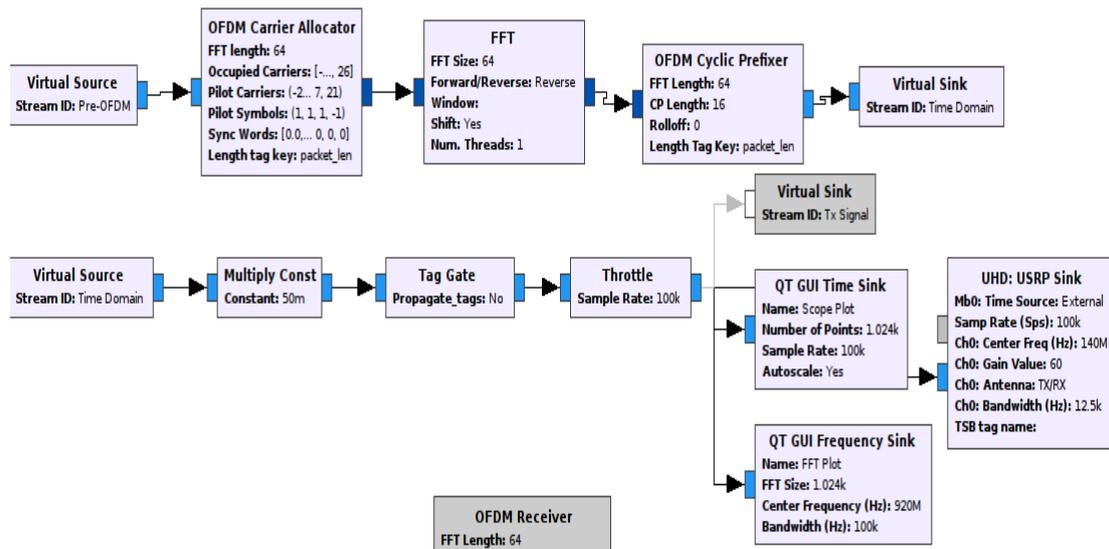


圖 41、OFDM 方塊圖(2)

若裝置的發射請求經由資料庫確認後，認證可以進行發射，則會開始發射訊號，如圖 42 以及圖 43 所示，經由頻譜儀實測後確認訊號有確實發射出去，達到動態頻譜接取的效果。

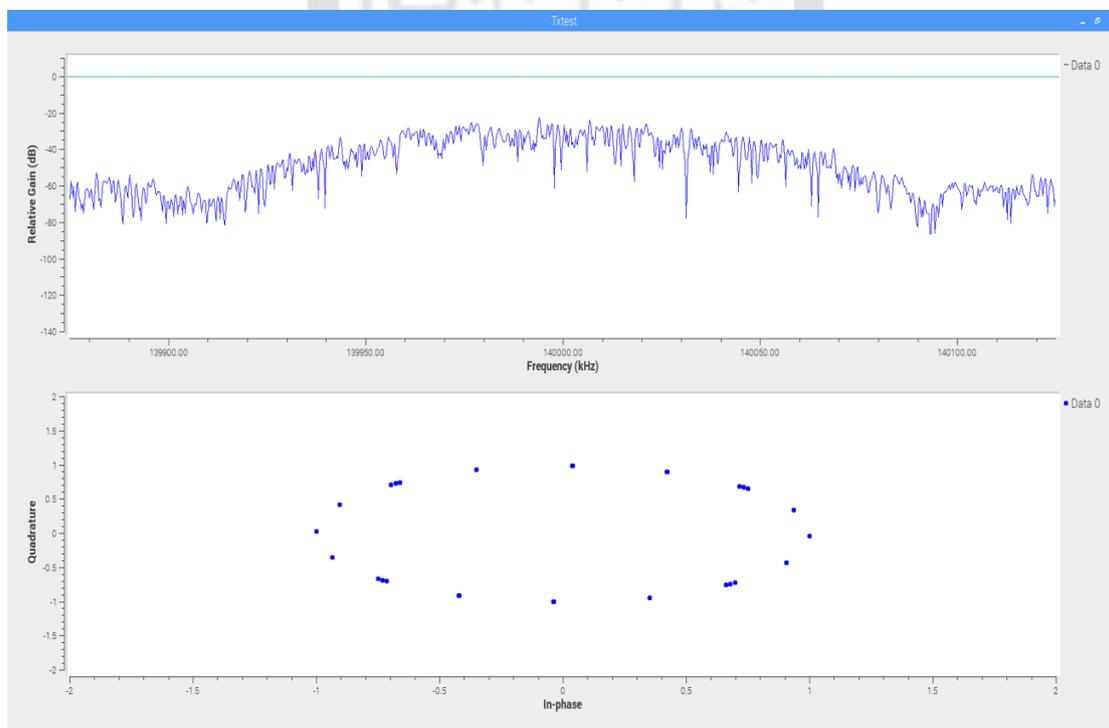


圖 42、GMSK 訊號發射圖

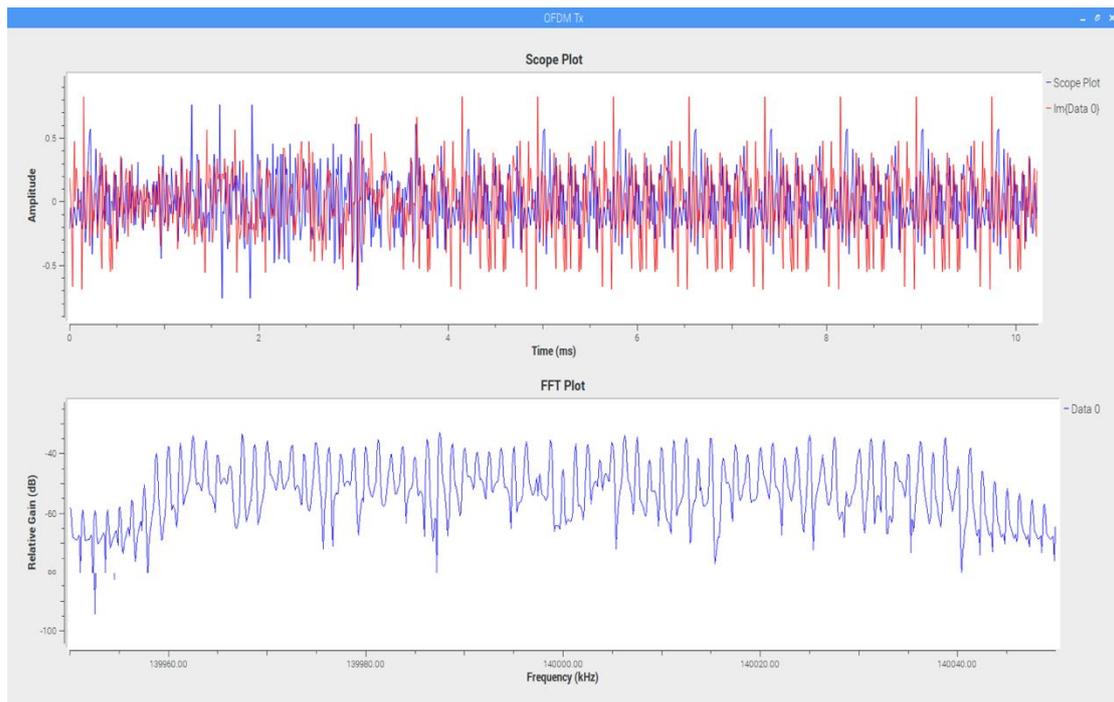


圖 43、OFDM 訊號發射圖

4.5 研究結果探討

本研究探討在未來日益倍增的頻譜需求下，以動態頻譜接取系統及以區塊鏈為概念的頻譜交易作為提出之解決方案，希冀能將日益壅塞的頻譜資源做最有效率的運應以及配置。

而動態頻譜接取系統又與關聯式資料庫有著密不可分的關係，在 4.1 小節中有提到隱藏節點的問題，雖然說感知無線電裝置具備著感測周遭環境的能力，能夠在其感測範圍內偵測到有沒其他裝置正在以相同的中心頻率進行訊號發射，但是其感測能力若沒有配合頻譜資料庫，則容易出現誤判的情形，因此需要一個能夠紀錄所有環境中所有裝置目前使用狀況等資料庫，方能最客觀的進行干擾判斷，因此在本研究當中也設計並實作動態頻譜資料庫，用以驗證感知無線電裝置的發送請求。實際實作資料庫時也發現，若感知無線電裝置沒有頻譜資料庫的輔助，將有一定的機率會造成

不必要個干擾，更進一步地說，若有頻譜資料庫的輔助，裝置可以透過資料庫得知目前該地有什麼頻率可以使用，進而透過軟體定義無線電平台，動態地調整參數並發射資料，是以可以推斷，有著頻譜資料庫的相輔相成，能夠使動態頻譜接取系統更加完善，頻譜利用更加有效率，不會有過多的閒置頻段。

而若所請求之發射請求因為中心頻率正被其他裝置使用中，或者所發射之功率大小、發射方向、發射頻寬等會影響到其他裝置發射權益，因而發送請求被頻譜資料庫拒絕，此時感知無線電裝置便可以有效發揮其軟體定義之功能，在被頻譜資料庫拒絕發送請求後可以重新調整感知無線電裝置的發射參數，進而避免干擾，只需要從軟體介面上用程式調整參數，便可以達到即時調整參數再進行訊號發射。

在頻譜交易的部分，透過第二章的文獻探討有提到頻譜資料庫安全的問題，若未來台灣開放頻譜交易，交易安全的問題需要受到重視，若頻譜交易受到竄改，一方面對於頻譜的購買及持有者的權益造成損害，另一方面則容易造成頻譜秩序紊亂，因此本研究透過區塊鏈公開透明、不可篡改、交易安全的特性，實作頻譜交易機制，並利用區塊鏈的 Hash 雜湊函數驗證每一筆交易的真實性，本研究在資料庫更動篡改交易資料後執行驗證程式，而也因為雜湊函數容易驗證特性的關係，程式便能馬上偵測到交易遭到竄改，該筆異常資料後的所有交易資料也必然為異常，除非惡意使用者一筆一筆將資料更動，才能躲過區塊鏈驗證。

而就算真的每筆資料都竄改躲過程式驗證，分散式帳本的特性讓每個裝置都有一份帳本的備份，只改一份帳本是沒意義的，得竄改整個區塊鏈架構下至少一半以上的裝置，耗費成本極大，不符合經濟效益，由此得知區塊鏈在驗證交易安全上功不可沒。

第五章 結論

5.1 結論

隨著第五代行動通訊的到來，物聯網 IOT 的應用也越來越廣泛，未來每一台裝置不論是車輛、穿戴式裝置、家電用品，生活中食衣住行每個環節都有可能需要連上網路，而頻譜資源終究是有限的，我們無法憑空生出頻譜資源，只能在現有頻譜資源的狹縫中求生存，因此動態頻譜接取的標準被提出來，頻譜資料庫記錄分時、分地、分頻的頻譜資料，裝置不必持有頻譜執照便可以透過購買閒置中的頻譜資源進行訊號的收發，而原頻段的持有者可以透過租賃其用不到的頻譜資源獲利，達到雙贏的局面。

在過去，許多無線電裝置的參數配置都是在出廠時就寫死燒錄在硬體當中，無法有彈性的調整參數，非常不符合動態頻譜接取的本意，因此軟體定義無線電以及感知無線電的問世，讓動態頻譜接取得以輕鬆實現，感知無線電裝置只要透過軟體無線電平台（以本研究來說便是 Linux 系統下開源的 GNU Radio），即能動態地調整無線電參數，同時可以實作感知無線電的感知功能，透過偵測環境頻譜狀態，來改變自己發射訊號的策略，並透過由主管機管制訂的頻譜資源資料庫進行確認，確認發送過程不干擾合法的執照擁有者使用頻段資源，維持頻譜秩序和諧。

區塊鏈的問世，解決了以往許多金錢交易系統內所存在的交易安全問題，區塊鏈的應用十分廣泛，舉凡金融交易、食安認證、旅館訂房系統等，都可以看見區塊鏈的身影，因此本研究認為，將區塊鏈概念融入頻譜交易機制當中，可以有效保障買賣雙方的權益，並且公開透明，對主管機關來說也可以明確看到交易情形，保障交易不被竄改，也不必擔心資料庫遭到惡意入侵，便於監理監管頻譜秩序。

5.2 未來研究建議

本研究在動態頻譜接取資料庫的建置上，除了頻譜交易是以分散式帳本儲存外，其他部分主要還是以中心化的結構為主，方便主管機管對於頻譜資源進行管理與監理，因此依然有著遭到惡意入侵或者破壞資料庫的風險存在，畢竟，建置完善的頻譜資料庫耗時費工，百密總有一疏，任何看似安全的中心資料庫都是可能有漏洞存在的，因此本研究建議未來在設置頻譜資源資料庫時應該考慮完全分散式的架構，不應該把雞蛋都放在同一個籃子當中，理應分散風險，讓每個有能力有空間容量能夠儲存資料庫的裝置都能夠有一份資料庫的備份，並互相驗證對方的資料庫是否正確，就分散式的架構而已，只要任何一份資料庫備份與他人不同，就代表某個環節出錯。

動態頻譜接取部分，本研究目前只進行兩到三台的裝置進行驗證，以一台裝置進行發射資料，另一台裝置偵測訊號，若判斷無產生干擾則可以發射訊號，反之若判斷干擾則進入待機的階段，未來研究應以更多更密集的裝置進行驗證，以符合未來連網裝置倍增的需求，而並非每一台裝置都有環境感知能力，因此也須考量裝置沒有感知環境能力的狀況，裝置該如何進行資料的發射與接收等。

頻譜資料庫判斷干擾的部分，本研究目前以判斷資料表內頻譜使用狀況來核准裝置的傳送請求，然而實務上干擾判斷的因素及演算法十分複雜，未來研究應該對於干擾的問題有更多的著墨，而非單純以資料表內的內容判斷，必須透過資料庫做更縝密嚴謹的計算，甚至可以做出頻譜資源分配決策的最佳化，資料庫內也需要做更細部的分區、分時、分頻資料，以符合動態頻譜接取所需。

參考文獻

- [1] 國家通訊傳播委員會，“頻率資料庫查詢系統，” [Online]. Available: <https://freqdbo.ncc.gov.tw/portal/index.aspx>.
- [2] J. Mitola III and G. Q. Maguire Jr., “Cognitive radio: Making software radios more personal”, IEEE Personal Communications, vol. 6, pp. 13-18, 1999.
- [3] J. Mitola III, “Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio”, Ph. D. diss., Royal Institute of Technology, Sweden, 8 May 2000.
- [4] ITU, “Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS),” Report ITU-R SM.2152, 2009.
- [5] J. Mitola, “Software radio architecture,” IEEE Communications Magazine, vol. 33, no. 5, pp. 26-38, May 1995.
- [6] S. Haber and W. S. Stornetta, “How to time-stamp a digital document,” Theory and Application of Cryptography, pp. 437-455, 1990.
- [7] E. Hossain D. Niyato Z. Han Dynamic Spectrum Access and Management in Cognitive Radio Networks Cambridge Univ. Press 2009.
- [8] M. Song C. Xin Y. Zhao X. Cheng "Dynamic spectrum access: From cognitive radio to network radio" IEEE Wireless Commun. vol. 19 no. 1 pp. 23-29 2012.
- [9] M. Sherman A. N. Mody R. Martinez C. Rodriguez R. Reddy "IEEE standards supporting cognitive radio and networks dynamic spectrum access and coexistence" IEEE Commun. Mag. vol. 46 no. 7 pp. 72-79 Jul. 2008.
- [10] I. F. Akyildiz W.-Y. Lee M. C. Vuran S. Mohanty "A survey on spectrum management in cognitive radio networks" IEEE Commun. Mag. vol. 46 no. 4 pp. 40-48 Apr. 2008.
- [11] A. B. Flores R. Guerra E. W. Knightly P. Ecclesine S. Pandey "IEEE 802.11af: A standard for TV white space spectrum sharing" IEEE Commun. Mag. vol. 51 no. 10 pp. 92-100 Oct. 2013.
- [12] D. Gurney "Geo-Location Database Techniques for Incumbent Protection in TV White Space" 3rd IEEE Int'l. Symp. New Frontiers in Dynamic Spectrum Access Networks pp. 1-9 2008-Oct.
- [13] S. Nakamoto. (2008, Oct.). Bitcoin: A peer-to-peer electronic cash system. [Online]. Available: <http://www.en.bitcoindr.cz/includes/img/bitcoin.pdf>

- [14] J.J. Sikorski, J. Haughton, M. Kraft "Blockchain technology in the chemical industry: machine-to-machine electricity market" *Appl Energy*, 195 (2017), pp. 234-246
- [15] K. Christidis M. Devetsikiotis "Blockchains and Smart Contracts for the Internet of Things" *IEEE Access* pp. 2292-2303 May 2016.
- [16] K. Kotobi and S. G. Bilén, "Blockchain-enabled spectrum access in cognitive radio networks," in *Proc. Wireless Telecommunications Symp. (WTS)*, 2017, pp. 1–6.
- [17] GNU Radio, "GNU Radio," [Online]. Available: <http://gnuradio.org/redmine/projects/show/gnuradio>.
- [18] Ettus Research, "USRP B200mini," [Online]. Available: <https://www.ettus.com/product/details/USRP-B200mini>.

