

TD

Radio Frequency Identification

Préambule

LES MATERIELS ET LOGICIELS MIS A VOTRE DISPOSITION SONT LA PROPRIETE DE
POLYTECH NICE – SOPHIA ET FONT L'OBJET D'ACCORDS SPECIFIQUES AVEC
LA SOCIETE ASK.

CES MATERIELS ET LOGICIELS NE PEUVENT EN AUCUN CAS ETRE UTILISES ET/OU CONSERVES
EN DEHORS DE CETTE SEANCE DE TRAVAUX DIRIGES.

TOUT LOGICIEL QUI AURA ETE INSTALLE SUR VOTRE MACHINE DE TRAVAIL DEVRA ETRE EF-
FACE APRES LA SEANCE. AUCUNE COPIE NE DEVRA ETRE CONSERVEE.

1 Mise en œuvre du RFID

Les travaux dirigés sur le RFIDs s'appuient sur une plate-forme professionnelle distribuée par la société ASK. A ce titre, des droits particuliers d'utilisation ont été signés entre Polytech'Nice – Sophia et ASK. Il est impératif de respecter les instructions données en préambule.

Le kit de développement fourni inclut un lecteur USB de RFID ASK RDR417, un SDK, un ensemble de cartes aux standards RFID (CTS512A, CTS256B, CTS512B & CTM512B, ISO15693, etc.).



1.1 Installation et Tests des drivers des applications de contrôle

1.1.1 Installation et Tests du matériel

Avant branchement à votre ordinateur du lecteur qui vous a été fourni, vous devrez installer les drivers pour les lecteurs ASK 4X7 (dossiers Drivers\RDR4X7_VCP-USB_Driver_v2.12.06).

Un certain nombre d'applications de démonstration sont disponibles pour vérifier si votre matériel fonctionne bien (dossier Demos). Vous pourrez commencer par tester les étiquettes RFID qui vous ont été fournies à l'aide de l'application « Demos\Polling » qui vous permettra de lire différents formats de cartes. Ces applications sont aussi à lancer en mode Administrateur pour que celles-ci fonctionnent et soient bien capables de se connecter au lecteur.

Question 1 : Vous identifierez bien le type des cartes qui vous ont été attribuées. Vous retrouverez les caractéristiques de la carte dans le tableau ci-dessous.

Question 2 : Quel est le type de RFID présent dans votre carte étudiant ?

TD

Radio Frequency Identification

	CTS-512-B	CTS-512-A Mifare-UL	SRI 512/ SRT 512	CTM-512-B	CTM1568A Mifare-UL C	CTM-8k-A Mifare-1k
RF Interface	ISO 14443 B	ISO14443 A	ISO 14443 B	ISO14443 B	ISO 14443 A	ISO14443 A
Compatibility	Calypso	Mifare*	Calypso	Calypso	Ultra Light	Mifare*
EEProm	512 bits	512 bits	512 bits	512 bits	1536 bits	8196 bits
Security						
Unique S/N	64 bits	56 bits	64 bits	64 bits	56 bits	32 bits
OTP area	up to 128 bits	32 bits	Sri : 5 x 32 bits Srt : -	Variable	-	Variable
Memory Write protection	Yes per sector	Yes per sector or block	Yes per 32 bits	Yes per sector	Yes	Yes per sector
Authentication	Simple static	Simple static	-	Simple dynamic	3 DES	Mutual dynamic
Key length	-	-	-	80 bits diversified	112 bits	48 bits
Other features			2 x 32 bits, binary counter	One way counter	16 bits counter	Random generator
SAM	Optional	Optional	-	Yes	Yes	MFRC500
Miscellaneous						
Anticollision	Yes	Yes	Yes	Yes	Yes	Yes
Typical transaction time	< 100 ms	< 100 ms	< 100 ms	< 200 ms	< 200 ms	< 200 ms
Typical communication distance	10 cm	10 cm	10 cm	10 cm	10 cm	10 cm
Write endurance	100 000 cycles	10 000 cycles	1 million	100 000 cycles	10 000 cycles	100 000 cycles

1.1.2 Envoi et réception de données au lecteur via l'application Monitor

Pour un premier contact avec la plate-forme, installez l'application « Monitor » qui se trouve dans « ASK/Monitor » (application disponible après installation dans « ASK Tools » du menu « Démarrer »).

Dans « Monitor », ouvrez le port sur lequel votre lecteur est connecté (« USB x » ou « ASK RDR4x7 0 »). En activant le bouton « Reset CSC » vous devriez obtenir un message du type « Reset CSC OK ».

Vous pouvez alors commencer à envoyer et recevoir des commandes au et du lecteur (Ouvrez pour cela le Panneau « Send and Receive » ).

Envoyez la commande suivante : 01 03 00 00 00 10 10 01 01 77, et lisez la réponse qui est retournée.

La liste et le format des commandes selon les cartes RFID utilisées sont documentées dans [ASK/CouplerSoftwareInterface/RD-ST-05077-18_ASK CSC - Coupler Software Interface _Gen4XX.pdf](#)

Question 3 : D'après la documentation, trouvez la signification de la commande précédente « 01 03 00 00 00 ... »

Question 4 : Comparez la commande que vous avez tapée avec la commande qui a été réellement envoyée au dispositif (ligne « -Send » du log en bas de la fenêtre). Proposez une décomposition fonctionnelle des valeurs réellement envoyées.

Question 5 : A l'aide de la documentation, analysez la réponse envoyée par le dispositif (ligne « -Receive » du log).

Question 6 : Refaite l'exécution de la même commande avec une carte posée sur le lecteur. Quelles réponses obtenez-vous ?

Question 7 : Trouvez dans la documentation les commandes permettant de contrôler les LEDs du lecteur de cartes. Tester l'allumage et l'extinction de chacune des LEDs séparément puis toutes simultanément.

TD

Radio Frequency Identification

1.2 Création de séquences de commandes

Il est possible de créer des séquences de commandes afin d'enchaîner celles-ci. . Vous avez des exemples de fichiers dans le dossier « C:\Program Files (x86)\ASK Tools\ASK CRC Monitor ».

Pour cela, vous pouvez créer des fichiers .ini ayant la syntaxe suivante :

```
[Sequence1]
Name= Leds

Name1= Rouge
Command1=XX XX XX XX
Sleep1=200

Name2=...
```

Pour charger un fichier de ce type vous utiliserez le bouton « SEQ » de l'interface de Monitor.

1.2.1 Séquence de commandes pour le contrôle des LEDs

Vous créez un fichier leds.ini contenant une suite de commandes qui allumera successivement les LEDs du dispositif, puis toutes les LEDs simultanément avant de tout éteindre.

1.2.2 Séquence de commandes pour la lecture et l'écriture des cartes

Nous allons maintenant utiliser des cartes CTS512B ou « Mifare UltraLight » (toutes les informations sur ces cartes se trouvent entre les pages 105 et suivantes de la documentation). La carte RFID CTS512B, conforme au standard ISO 14443 B est une mémoire EEPROM de 512 Bits paginées.

Ouvrez le fichier « C:\Program Files (x86)\ASK Tools\ASK CSC Monitor\Sequences.ini », vous aurez alors accès à des séquences de commandes, comme « Read/Write CTx512B or MFUL » pour votre carte Mifare UltraLight (MFUL).

Question 8 : Exécutez « Read/Write CTx512B or MFUL ». A quoi correspondent les commandes de la fonction ? En identifiant ces commandes, nous allons pouvoir les mettre en œuvre dans un programme écrit classiquement dans l'espace utilisateur.

2 Développement

Des bibliothèques DLL (Win32 utilisable en C, C++, C# depuis Visual Studio) sont fournies dans la distribution logicielle ASK. Vous trouverez des fichiers sources d'exemple d'utilisation.

Dans cette seconde partie, il s'agit de mettre en œuvre ce Kit de Développement afin de réaliser une simple application de lecture / écriture sur des TAGs type tickets Papier CTS512B. Nous réaliserons celle-ci en C#, même si la bibliothèque fournie n'est pas en code managé.

Pour utiliser une fonction d'une bibliothèque en code non managé (C ou C++) en C#, il suffit d'importer la fonction de la bibliothèque grâce à DllImport et de redéfinir le prototype de la fonction que vous souhaitez en C# (avec les paramètres correspondants).

```
[DllImport("Askcsc.dll", EntryPoint = "CSC_SearchCSC")]
public static extern Int32 SearchCSC();

[DllImport("Askcsc.dll", EntryPoint = "CSC_SendReceive")]
public static extern Int32 SendReceive(Int32 timeout, byte[] Commande,
Int32 commandeLen, byte[] Response, ref Int32 responseLen);
```

TD

Radio Frequency Identification

L'ensemble des fonctions de l'API est documenté dans ASK\ASKCSC_DLL\Spec ASKCSC DLL uk.pdf

Question 9 : Créer une application de stockage du nom d'un utilisateur sur une carte et la lecture pour l'identification associée.

En C#, afin d'encoder et de décoder vos chaînes de caractères en hexa, vous pourrez utiliser les fonctions suivantes :

```

public static string ByteArrayToHexString(byte[] Bytes)
{
    StringBuilder Result = new StringBuilder();
    string HexAlphabet = "0123456789ABCDEF";

    foreach (byte B in Bytes)
    {
        Result.Append(HexAlphabet[(int)(B >> 4)]);
        Result.Append(HexAlphabet[(int)(B & 0xF)]);
    }

    return Result.ToString();
}

public static byte[] HexStringToByteArray(string Hex)
{
    byte[] Bytes = new byte[Hex.Length / 2];
    int[] HexValue = new int[] { 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06,
                                0x07, 0x08, 0x09, 0x00, 0x00, 0x00, 0x00,
                                0x00, 0x00, 0x00, 0x0A, 0x0B, 0x0C, 0x0D,
                                0x0E, 0x0F };

    for (int x = 0, i = 0; i < Hex.Length; i += 2, x += 1)
    {
        Bytes[x] = (byte)(HexValue[Char.ToUpper(Hex[i + 0]) - '0'] << 4 |
                          HexValue[Char.ToUpper(Hex[i + 1]) - '0']);
    }

    return Bytes;
}
  
```

3 Pour aller plus loin...

Maintenant que vous avez travaillé sur une carte CTS-512-B, vous pouvez maintenant essayer de hacker votre carte d'étudiant, mais c'est à vos risques et périls !... 😊