White Paper
04-2017

The CIPURSE™ Terminal Library

How it can fast-track your CIPURSE™ Reader Implementation

Abstract

CIPURSE™ is an Open Standard for transport ticketing that is fast gaining adoption. It presents a great opportunity for reader and terminal equipment manufacturers. The CIPURSE™ Terminal Library (CTL) is a tool which can enable equipment manufacturers to quickly implement CIPURSE™ support in their products. It is a portable library distributed in ANSI C and Java sources which is available from Infineon.

In this paper, we examine what CIPURSE™ is and what capabilities the CTL offers. We describe the various reader architectures which can integrate the CTL and give examples of commercially available readers which have already done so in a short time. We explain why the CTL is the smart way to fast-track the implementation of any CIPURSE™ reader.

Infineon shares the conviction of the Open Standard for Public Transport (OSPT) Alliance that the future belongs to open standards. Having interoperable equipment compliant to open standards is essential to realizing this vision. The adoption of the CTL in reader and terminal equipment embraces openness and ensures compliance.

Table of contents

1 What is CIPURSE™?

CIPURSE™ is a protocol defined by the Open Standard for Public Transport (OSPT) Alliance, which promotes interoperability of transport ticketing standards. It addresses the problem of single-vendor proprietary standards used in transport ticketing systems, which limits the competition in all system component suppliers.

CIPURSE™ offers best in class security based on Advanced Encryption Standard (AES) 128-bit. Furthermore, its advanced cryptographic protocol is inherently resistant to side-channel attacks such as differential power analysis and differential fault analysis. Therefore, it does not need dedicated hardware measures against these attacks and can be realized on low-cost silicon implementations.

CIPURSE™ employs open standards including ISO/IEC 7816 and ISO/IEC 14443. This makes it compatible with most existing smart card infrastructure. With a software upgrade to integrate CIPURSE™, existing readers and terminals can be used in a CIPURSE™ based fare collection system.

The OSPT Alliance is a nonprofit industry organization founded in 2010 by 4 companies, including Infineon. As of March 2017, it has more than 60 full members and associate members (comprising chip suppliers, smart card manufacturers, reader manufacturers, system integrators, transit operators and more).

For more information, visit www.osptalliance.org

1.1 Why choose Open Standards?

Imagine that you live in a city where the buses and trains use a proprietary legacy fare collection system with contactless cards and readers. Due to inherent security weaknesses of the legacy system, hackers have succeeded in compromising the system and made clone cards. The transport operator has been kept busy patching the system and applying a blacklist to contain the breach. Then the operator has to deal with another teething problem: the 4-byte card unique identifier (UID) has reached its maximum value and rolled over.

If cards may not have unique identifiers, how can the transport operator effectively apply the blacklist? If unsavoury characters can board buses and trains with fake cards, how can law enforcement protect its citizens?

Eventually, the transport operator must consider an expeditious overhaul of the aging system. It can take one of two possible paths.

1. To install a new system based on proprietary technology.
2. To install a new system based on an Open Standard (such as CIPURSE™).

Our proposition is the second option makes better sense from both **Security** and **Cost** perspectives.

First, an Open Standard necessitates the public availability of its specifications for scrutiny and rigorous debate by anyone, including industry experts. Notably, CIPURSE™ employs an advanced security concept inherently resistant to side channel attacks, which won the **German Prize for IT Security** in 2012. Besides, CIPURSE™ cards can have UIDs up to **10 bytes long**.

Second, an Open Standard which is steered by an industry alliance, whose membership is open to all industry players in a non-discriminating way, **secures its own future** by giving members the power to influence its specifications and keep its relevance. Notably, CIPURSE™ is steered by the OSPT Alliance, which has more than **60 full members and associate members** from across the industry.

Third, an Open Standard which is devoid of proprietary technology licensing costs and discriminatory pricing behavior brings **substantial long-term cost savings** to practitioners and end-users. Notably, there is no upfront licensing cost to use CIPURSE™ and only a small royalty payable per CIPURSE™ product sold (lower of US$0.01, or two percent of each product's commercial selling price). Furthermore, vendors can sell up to 10,000 units of CIPURSE™ products per year royalty-free.

Figure 1 OSPT CIPURSE™

2 What is the CIPURSE™ Terminal Library (CTL)?

The CIPURSE™ Terminal Library (CTL) is a piece of software developed by Infineon for reader manufacturers to quickly add CIPURSE™ functionality to readers, or terminals which are connected to the readers. It provides a comprehensive list of commands to personalize and operate CIPURSE™ contactless cards and Secure Access Modules (SAM).

The library is available as ANSI C source code (compliant to C89/C90 standard) and Java source code (requiring Java Runtime Environment 1.6 or later). The footprint of the CTL (C implementation) is 20kB flash and 2kB Random Access Memory (RAM) when compiled using the Keil MDK-ARM compiler.

Let's take a closer look at the CTL's modular design. The core comprises 3 modules: Presentation, Framework and Crypto Services. These modules implement the platform-independent CIPURSE™ functionality which does not have to be modified for each target platform. Adjacent to the core sits the Platform Abstraction Layer (PAL). The PAL has platform-specific code which may need some modifications to fit a target platform.

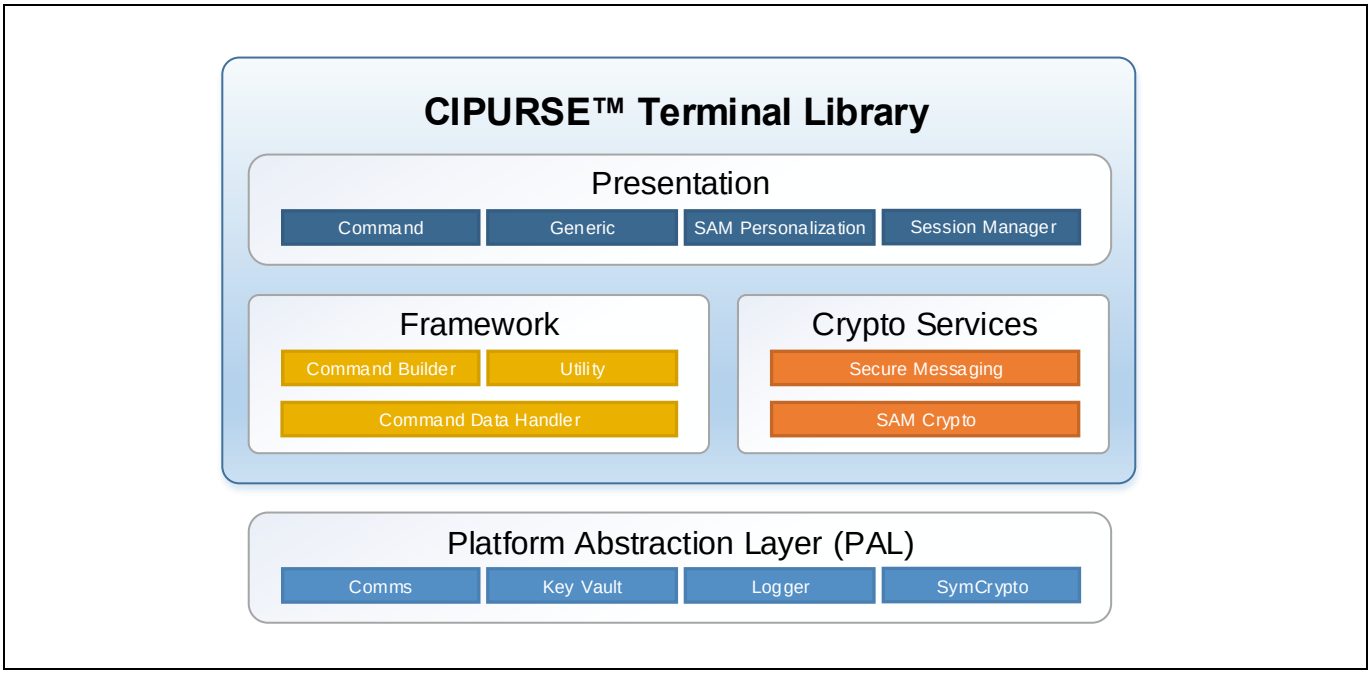

Figure 2 Design of the CIPURSE™ Terminal Library

The Presentation module is the CTL's façade to a reader application. It provides 4 groups of Application Programming Interface (API).

- Command API – constructs the syntax for most CIPURSE™ commands, including support for all file types: Binary File, Value Record Files, Linear and Cyclic Record Files.
- Generic API – sends miscellaneous commands.
- SAM Personalization API – constructs the syntax for SAM personalization commands.
- Session Manager API – manages a session (which is an instance of the CTL used by an application).

The Framework module implements common routines used by other modules.

- Command Builder – constructs the syntax of complex commands.
- Common Data Handler – stores global data structures and buffers.
- Utility – works with data buffers and construct commands.

The Crypto Services module implements CIPURSE™ Secure Messaging (SM) and cryptography.

- Secure Messaging API – manages a secure channel used by secure messaging.
- SAM Crypto API – implements SAM cryptographic operations (which can be hardware or software based).

The PAL module implements platform-specific functionality.

- Comms API – manages the connection to a contactless card or SAM, and sends / receives data. It defaults to a PC/SC reader implementation but can be modified to suit other platforms.
- Key Vault API – provides key storage in software if a SAM is not used.
- Logger API – sends trace messages to an output console.
- SymCrypto API – implements AES encryption and decryption. It can be modified to use the hardware accelerators on a target platform.

2.1 Which Application Examples are available?

Instructive examples are distributed with the CTL to illustrate how the Presentation module APIs can be used by an Application. These include:

- Binary Files example – reads / updates binary files.
- Record Files example – reads / updates linear and cyclic record files.
- Value Record Files example – reads / updates value record files.
- Limited Refund example – performs limited value increase / decrease of value record files.
- Admin Operations example – creates and deletes Elementary File (EF) / Application Dedicated File (ADF), reads / updates file attributes.
- SAM Personalization example – creates key files and loads keys on Master / Field SAMs.
- Cryptographic Operations example – generates and verifies Message Authentication Code (MAC).
- Error Handling example – recovers from a card communication error.

2.2 How does the CTL hide complexity from the Application?

Per design intent, an application should only have to interact with the CTL Presentation module, since it has the APIs to invoke any CIPURSE™ functionality. These APIs serve as high-level instructions that can be assembled to accomplish a task, such as deducting a fare from a card.

Suppose an application needs to update sensitive data (e.g. purse value) stored in a Value Record File on a CIPURSE™ card, while guarding against the possibility of someone eavesdropping on the Radio Frequency (RF) communication.

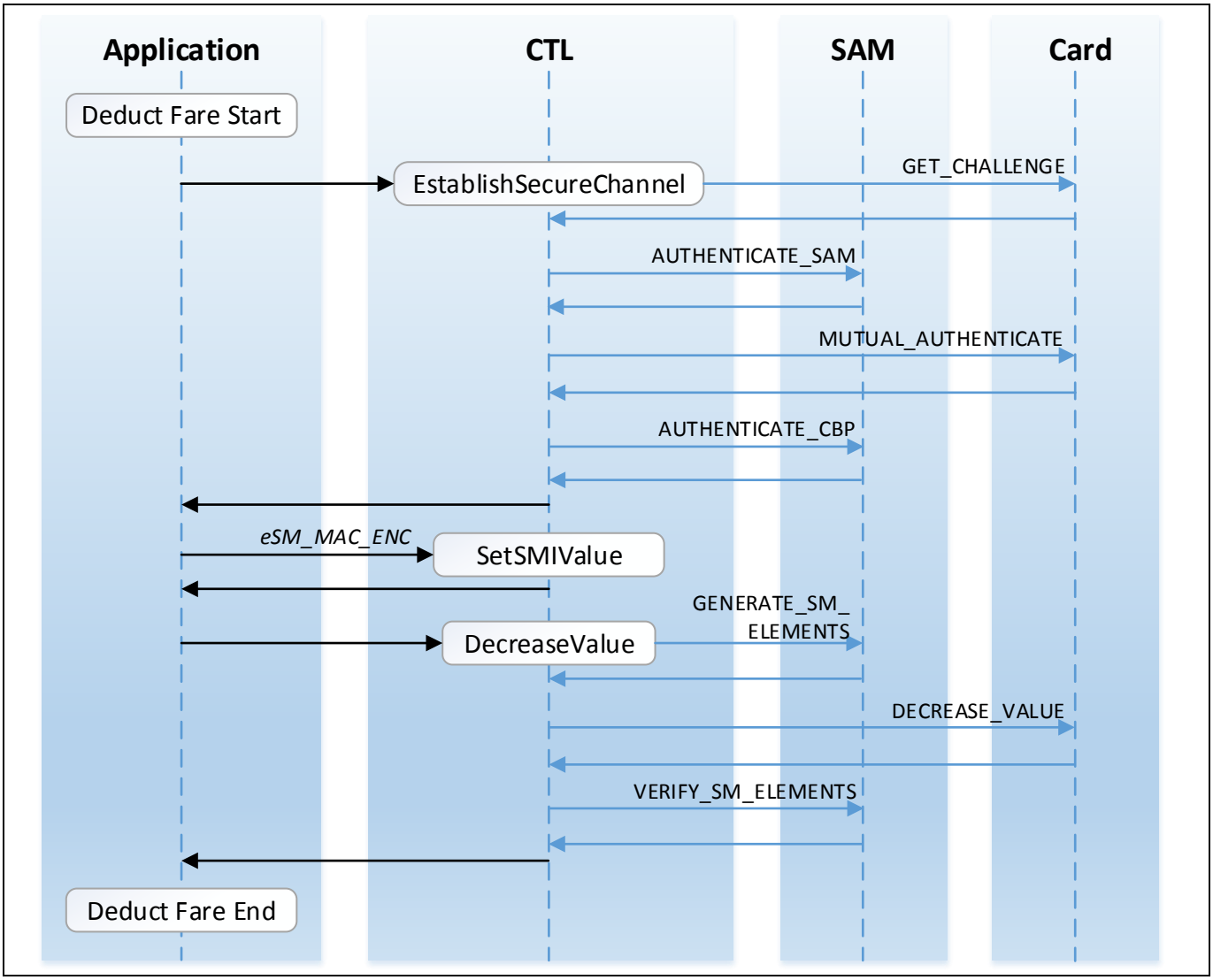

Figure 3 Simplified view of a Fare Deduction Operation

1. When the application calls the Command API **EstablishSecureChannel**, the CTL performs a mutual authentication with the card by generating the relevant command Application Protocol Data Units (APDU) to setup the secure session with the card and the SAM.

2. Then the application calls the **SetSMIValue** with the *eSM_MAC_ENC* parameter, indicating that it wants to send subsequent commands to the card with Message Authentication Code (MAC) enabled for integrity-checking and wants the card's responses to be Encrypted (ENC) for privacy.

3. When the application invokes **DecreaseValue**, the CTL uses the SAM to generate the necessary cryptogram (MAC code) for the command and sends the APDU to the card. Upon receiving the card's response, the CTL sends it to the SAM to verify its cryptograms and perform decryption, before it returns the decrypted result to the application.

In this way, the application is able to perform an operation securely on the card without having to deal with the complexity of the CIPURSE™ Secure Messaging (SM) protocol and underlying cryptographic operations.

Furthermore, if the application developer wishes to try a different security level (for e.g. sending a command in PLAIN format and receiving the response with MAC code enabled), the application code change needed is minimal – specify *eSM_PLAIN_MAC* instead of *eSM_MAC_ENC* in call to **SetSMIValue**.

2.3 What if I don't want to use the CTL?

The CTL is certainly not the only avenue to creating a CIPURSE™ reader. The gratifying thing about CIPURSE™ being an Open Standard is that the **CIPURSE™V2** specifications can be freely downloaded from the OSPT website (www.osptalliance.org), and one can develop a reader from them. The documents are:

- CIPURSE™V2 Operation and Interface Specification Revision 2.0
- CIPURSE™V2 Cryptographic Protocol Specification Revision 1.0
- CIPURSE™V2 T Profile Specification Revision 2.0
- CIPURSE™V2 S Profile Specification Revision 2.0
- CIPURSE™V2 L Profile Specification Revision 2.0
- CIPURSE™V2 SAM Specification Revision 2.0

The **Operation and Interface Specification** and **Cryptographic Protocol Specification** specify the CIPURSE™V2 core functionality (data objects, command set and security mechanism). The **Profile Specifications** specify the 3 application profiles (T for Transaction, S for Standard and L for Lite). The **SAM Specification** specifies the interface and cryptographic services of a CIPURSE™ SAM.

Although developing from specifications is a plausible approach, it may take much longer than if one is using the CTL. There can be a significant development and qualification effort to attain a high degree of compliance to the specifications, which is important for cross-vendor interoperability. Therefore, we highly recommend using the CTL as the smart way to fast-track any CIPURSE™ reader implementation.

2.4 Top Reasons for using the CTL

1. Compliance to CIPURSE™V2 specification (including core functionality, card profiles and SAM).
2. Includes working examples with self-checking code.
3. Written in portable ANSI C and Java code.
4. Comprises fully tested code.
5. Developed and maintained by Infineon – OSPT founding member and key contributor.
6. Modular design – easy to understand and extend (e.g. PAL module).
7. Saves development and verification time.

3 Which Reader Architectures can use the CTL?

3.1 Windows or Linux Terminals

In a typical Windows or Linux terminal which is connected to USB **Personal Computer / Smart Card Interface (PC/SC)** readers with ISO/IEC 14443 Contactless interface and ISO/IEC 7816 SAM interface, the CTL can be readily used as a middleware layer to support a CIPURSE™ application. Since the default PAL module in the CTL already works with PC/SC readers, no further adaptation work is needed. The PC/SC readers do not need any firmware upgrade.

In this case, depending on the language used to develop the terminal application, either the CTL C or Java implementation may be employed. If the terminal application is written in a programming language such as C# or Python, the CTL (C implementation) can be compiled as a Dynamic Link Library or an Extension module to be loaded by the application.

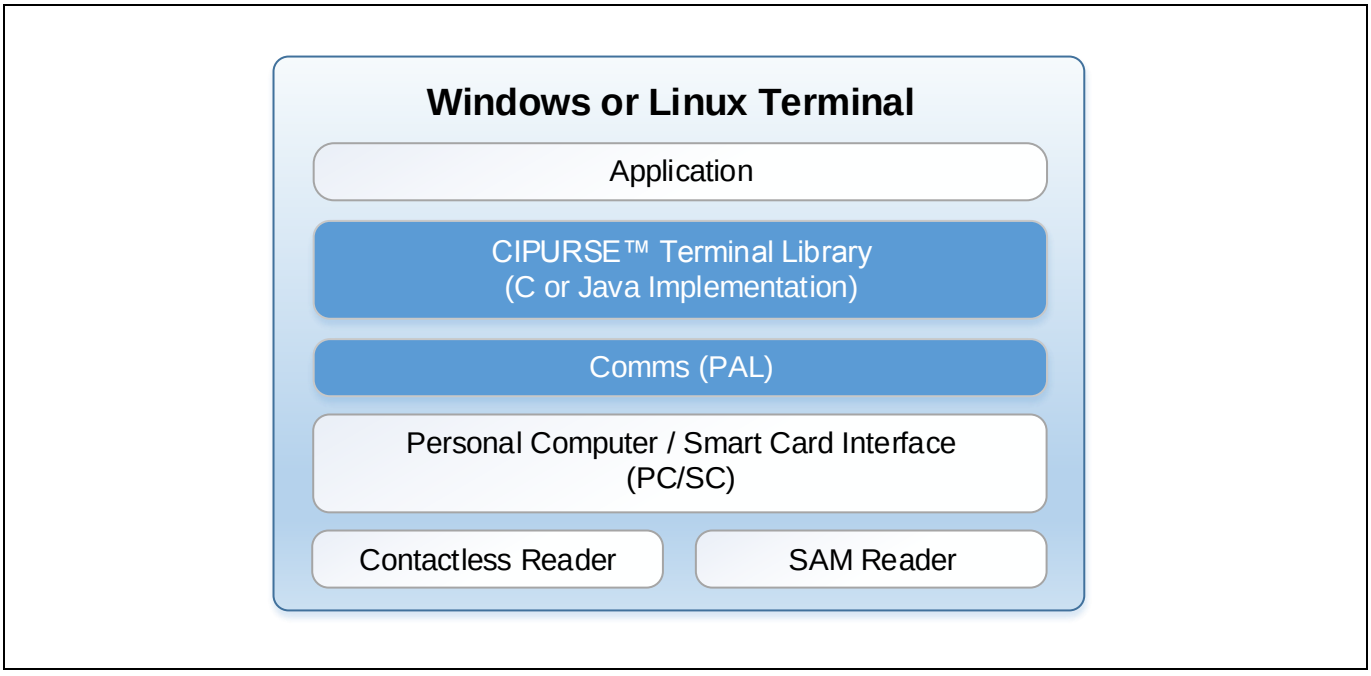

Figure 4 Using the CTL on a Windows or Linux Terminal

3.2 Embedded Systems

In a typical reader with embedded software written in C or C++, the CTL (C implementation) can be used as a middleware layer to support a CIPURSE™ application. In this case, the PAL module must be adapted to use the APIs supplied by the reader's ISO/IEC 14443 and ISO/IEC 7816 stacks. This is the minimum level of work needed to integrate the CTL.

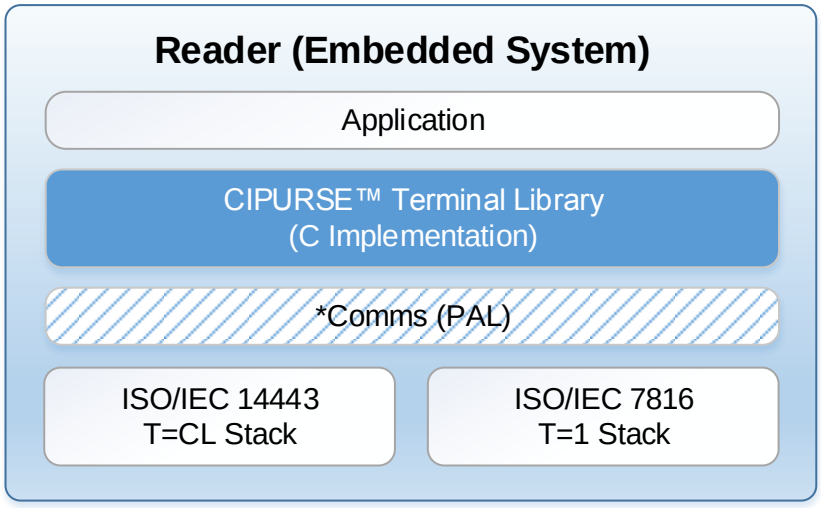

Figure 5 Using the CTL on a Reader based on an Embedded System

3.3 Android Devices

In a typical Android mobile terminal, the CTL (Java implementation) can be used as a middleware layer to support a CIPURSE™ application. In this case, the PAL module must be adapted to use the APIs supplied by Android's standard NFC stack and the terminal's ISO/IEC 7816 stack. The latter is usually a module of a custom Software Development Kit (SDK) supplied by the terminal manufacturer.

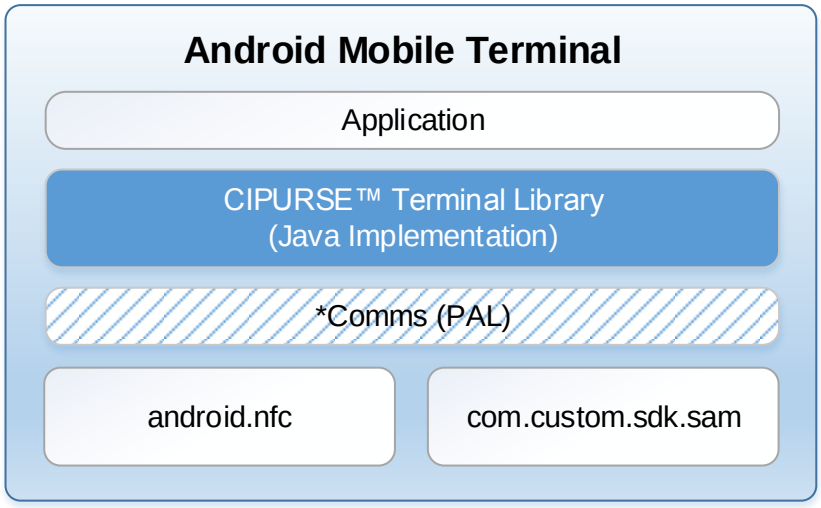

Figure 6 Using the CTL on an Android Mobile Terminal

4 Case Studies

We present 2 case studies to illustrate the use of the CTL in commercial readers. These readers are supported by software development kits from their manufacturers, thereby enabling new capabilities to be incorporated. The common theme is how quickly the **CTL infuses CIPURSE™ competency in manufacturers** and that **no reader hardware modification is required**.

4.1 Kenetics Reader

The **Kenetics Volaré SR14-ABC** is a Contactless Card Reader designed by **Kenetics Innovation Private Limited** based in Singapore. It is widely deployed on Singapore's public buses and at Mass Rapid Transit (MRT) stations as ticket validators and top-up machines. Over many years, the SR14 has proven itself to be a reliable workhorse operating under Singapore's hot and humid weather.

The SR14 supports several contactless technologies including ISO/IEC 14443 Type A/B, and has 4 ISO/IEC 7816 SAM slots. It employs an innovative dual ARM processor architecture: an RF processor providing "Contactless and SAM" access and an Application processor providing an RTOS environment for a reader application. This architecture offers deployment flexibility: a two processor configuration for demanding applications, or the single RF processor configuration for lower cost.

Both processors can be custom-programmed using the Kenetics Software Development Kit. Within two weeks, we integrated the **CTL (C Implementation)** into the Application processor firmware to produce a dual processor CIPURSE™ reader. Then with a little extra effort, Kenetics engineers integrated the CTL to the RF processor firmware to create an entry level CIPURSE™ reader.

At the contactless interface, the SR14 supports using the Protocol Parameter Selection (PPS) request to increase the baud rate up to 424kbps. At the SAM interface, the reader supports using the PPS to specify a division factor of 8 and to raise the ISO/IEC 7816 clock close to 10MHz, effectively reaching a baud rate of 1.2Mbps. When the reader executes the CTL examples, it demonstrates excellent transaction time performance.

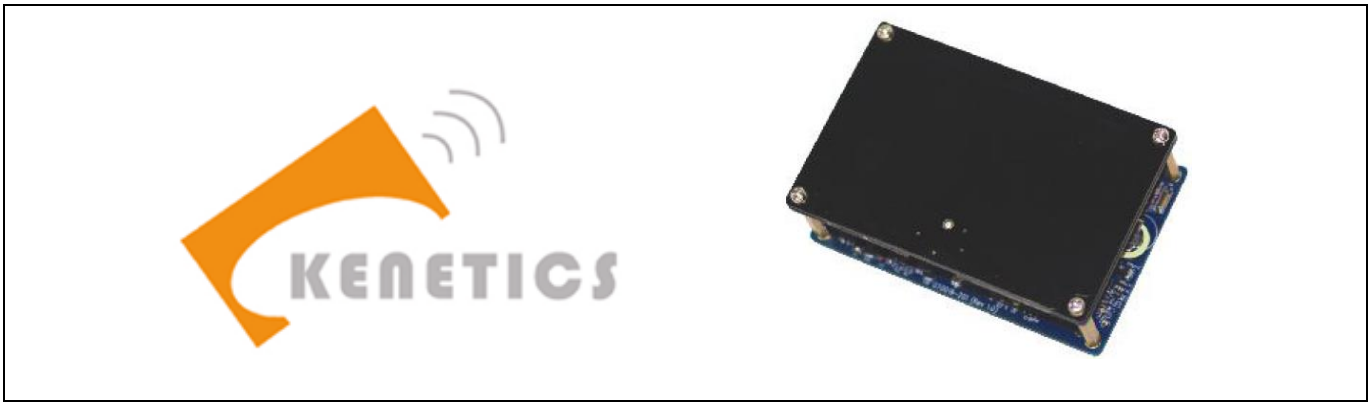

Figure 7 Kenetics Volaré SR14-ABC Reader

4.2 CipherLab Mobile Terminal

The **CipherLab RS30 series Touch Mobile Computer** is a rugged (IP54 industrial standard) Android mobile terminal produced by **CipherLab Company Limited**, which is headquartered in Taipei, Taiwan. It has a quad-core processor, 4.7in touch screen, 2500mAh Li-ion battery and a wide array of connectivity options: NFC, 1D or 2D barcode scanner, dual SIM slots, Bluetooth, Wi-Fi, microSDHC card slot and an ISO/IEC 7816 SAM slot.

The RS30 is used in logistics, healthcare, retail and ticketing applications. As a ticket inspector device, it can be used to scan tickets with QR codes, barcodes or NFC technology. CipherLab provides drivers in their Software Development Kit that enable developers to write Android applications incorporating barcode scanning, NFC and SAM access.

Using Android Studio and the **CTL (Java implementation)**, we employed the SAM driver and Android NFC stack to create an Android application for the CIPURSE™ Access Control use case. It took a week to integrate the CipherLab SAM driver and the CTL, and two weeks to fine-tune the performance. The CipherLab SAM interface automatically performs PPS to specify a division factor of 16 at a fixed ISO/IEC 7816 clock of 4.8MHz, reaching a baud rate of 300kbps. The Android NFC interface operates at 106kbps.

The CIPURSE™ transaction time is fast enough for the RS30 to serve as a ticket inspector device. **American Eagle**, a System Integrator for a CIPURSE™ implementation in Pierce County Ferry in Washington State, selected the CipherLab RS30 for this purpose. They have created a ferry ticketing Android application that seamlessly uses the barcode scanner, NFC, SAM and networking capabilities of the RS30.

Figure 8 CipherLab RS30 series Touch Mobile Computer

5 Conclusion and a Call to Action

Reader and terminal equipment manufacturers hoping to capitalize on the growing CIPURSE™ opportunity will need to make their products CIPURSE™ ready, and it isn't difficult to do. With the **CIPURSE™ Terminal Library**, any ISO/IEC 7816 and ISO/IEC 14443 compliant reader can fully support CIPURSE™.

Talk to an Infineon representative on how you can get it. Join us on this exciting CIPURSE™ journey!

Published by
Infineon Technologies AG
85579 Neubiberg, Germany

Order Number: B000-H0000-X-X-7600
Date: MM / 2017