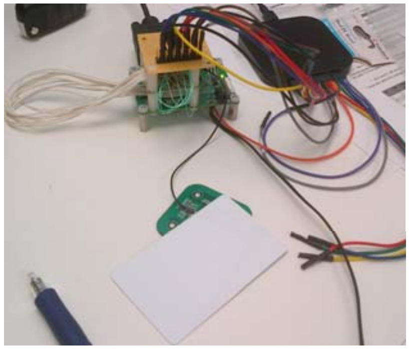


PicoPass Card Interoperability with the TRF796x/70A

Josh Wyatt

ERF Apps/Sytems

03/2012

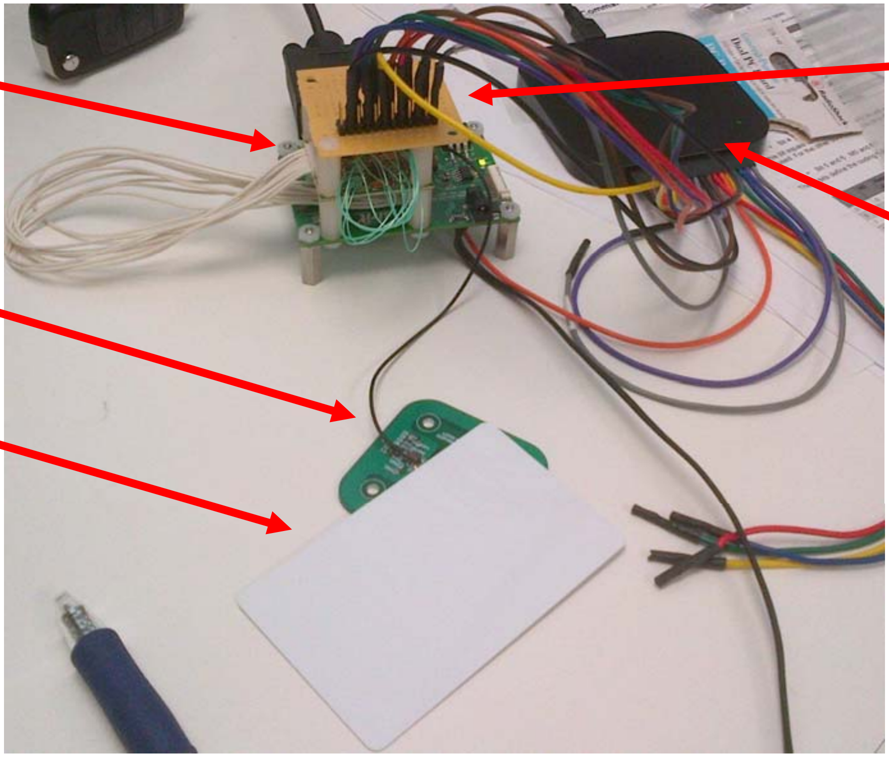
Background

- Customer is using TRF79xx for their reader/writer and is looking at using the PicoPass based card/tag devices from Inside Secure.
- These cards are have the following characteristics:
  - Carrier frequency: 13.56 MHz +/- 7 kHz
  - Data rate :
    - 26 kbit/s (ISO15693-2),
    - 106 or 424 kbit/s (ISO14443B-2 and ISO14443B-3)
  - Data coding : User can configure :
    - ISO15693-2 & ISO14443B-2 compliant with protocol auto-detection
  - Fast anti-collision :
    - Up to 50 chips/s using protocol ISO15693-2, and >100 chips using protocol ISO14443B-2
    - Several chips can operate independently in the field.
- While these cards are not fully ISO15693 or ISO14443B compliant, they do use the same air interface, so customer has requested TI to have a look at seeing how they might be used with the TRF79xx devices.

- **SIDE NOTE:** The Nexus S and Galaxy Nexus Cannot read the PicoPass cards at all whereas TRF79xx can get them to respond to standard REQB command.

Background (cont.)

- Sirit/I2R has implemented firmware to handle the PicoPass tags using ISO15693 air interface and PicoPass custom commands.
- We have this reader and their GUI, so a simple breadboard was made for robust logic analyzer connection between TRF79xx and the Freescale MCU that Sirit/I2R used.
- Customer provided some sample tags.



Sirit/I2R Reader w/TRF7960

PCB Antenna

PicoPass Card

Breadboard with 30AWG lines tapped into D0:D7, DATA_CLK and IRQ lines

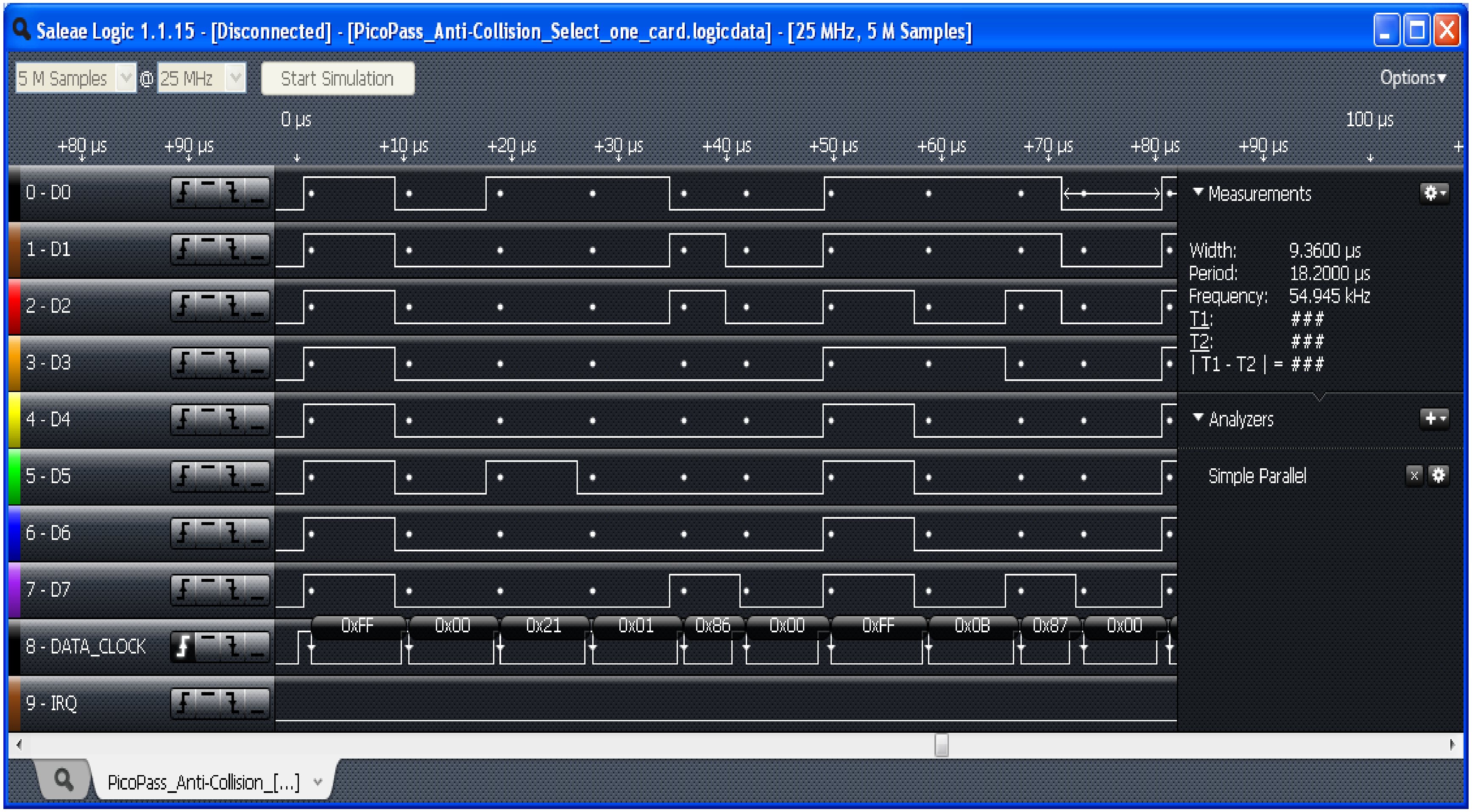
Saleae 16 channel

logic analyzer

TRF79xx Configuration for PicoPass ISO15693-like operation

- **Write Control Registers to set up TRF79xx for PicoPass**
  - **Chip Status Control Register (0x00)**
    - **Value ➔ 0x21**
      - **TX on, full power out, +5VDC operation**
  - **ISO Control Register (0x01)**
    - **Value ➔ 0x86**
      - **No RX CRC, ISO15693 high tag bit rate, FSK, 1 of 4 data coding**
  - **Regulator Control (0x0B)**
    - **Value ➔ 0x87**
      - **Automatic Settings**

Parallel Logic Analyzer capture of TRF79xx configuration for PicoPass operations



Saleae Logic 1.1.15 - [Disconnected] - [PicoPass_Anti-Collision_Select_one_card.logicdata] - [25 MHz, 5 M Samples]

5 M Samples @ 25 MHz

Measurements

Width: 9.3600 µs
Period: 18.2000 µs
Frequency: 54.945 kHz
T1: ###
T2: ###
| T1 - T2 | = ###

Analyzers

Simple Parallel

CRC Details

- These cards use same CRC polynomial equation that is seen in ISO15693-3, but use a different CRC start value than what is seen with compliant ISO15693-3 devices.
  - Polynomial is 0x8408 ➔ $x^{16} + x^{12} + x^{5} + 1$
  - Start (Preset) Value is ➔ 0xE012
  - Direction ➔ Backwards
- This means that ISO15693-like CRC engine needs to be in firmware now for these cards, but with the different preset value and it need to be sent out accordingly with some of the commands, as TRF79xx has ISO15693-3 compliant CRC generator built in (and will not be needed for these cards)
- CRC is not calculated on the command byte, just the data being sent.
  - For Example ➔
    - Read Operation on Block 0, send CRC with the packet of 0x7333 (example later)
    - Read Operation on Block 1, send CRC with the packet of 0xFA22 (example later)
- CRC for ISO14443B style operations follows exactly the ISO14443-3 (for Type B) standard CRC.

Parity Bit Handling for PicoPass Command Opcodes

- For the command opcodes, the PicoPass cards use 1 byte format.
- B7 is the parity bit of the byte and is calculated bitwise over B0:B6 using XOR

OPCODE (1 byte)							
B7	B6	B5	B4	B3	B2	B1	B0
P	M1	M0	K	Instruction			

- For example:
  - Command is ACTALL (0xA = 1010_2)
  - K = 0 (should be 0 for all commands other than READCHECK)
  - M0 = 1 (ISO15693-2 FSK coding)
  - M1 = 0
  - Parity calculated by XOR'ing B0:B6 to find B7 value
  - Resulting Byte is: 0xAA = 10101010_2

OPCODE (for ACTALL)							
B7	B6	B5	B4	B3	B2	B1	B0
1	0	1	0	1	0	1	0

Command Opcodes Used

- The following PicoPass commands were used with the sample cards in ISO15693-2 mode:
  - ACS (combo command)
    - ACTALL ➔ 0xAA, response is SOF
      - TRF Command String: 0x8F, 0x90, 0x3D, 0x00, 0x10, 0xAA
    - Identify ➔ 0xAC, response is ASNB + CRC
      - TRF Command String: **0x8F, 0x90, 0x3D, 0x00, 0x10, 0xAC**
      - ASNB = **0x58, 0xB7, 0x6B, 0xE0, 0x00, 0x40, 0x02, 0x1C**
      - CRC = **0xAC, 0x98**
        - » **For this card**
    - Select ➔ 0x21 + ASNB, response is SN + CRC
      - TRF Command String: **0x8F, 0x90, 0x3D, 0x00, 0x90, 0x21, ASNB** (for this card used)
      - SN (Contents of Block 0, Chip UID) = **E0120007035DBAC0**
      - CRC = **0xE01B**
- ACS **must** be performed before these other commands
  - Read Block
  - Update (Write Block)
  - Halt
  - Select

Command Opcodes Used (cont.)

- ACS **must** be performed before these other commands - otherwise no response IRQ value will be returned.
  - **Read Block** (0xAC + Block Address + CRC)
    - 0xAC
    - Block Address (for example 0x01, 0x02, etc.)
    - CRC (according to prior instructions)
      - For example: TRF79xx command string for reading block 0 on tag that has been through ACS command set.
        - » 0x8F, 0x90, 0x3D, 0x00, 0x40, 0xAC, 0x00, 0x73, 0x33
      - Response will be contents of Block 0 (8 bytes of tag UID)
        - » 0xC0, 0xBA, 0x5D, 0x03, 0x07, 0x00, 0x12, 0xE0
        - » Needs to be rotated as this is LSByte first format (expected)
        - » Results in 0xE0120007035DBAC0

Command Opcodes Used (cont.)

- **again** ACS **must** be performed before these other commands - otherwise no response IRQ value will be returned.
  - **Select** (0x21 + ASNB)
    - Two ways, using:
      - 0x21
      - ASNB (from Identify Command, during the ACS)
        - or
      - SN (from the first Identify command response)
      - For example: TRF79xx command string for doing Select on tag that has been through ACS command set.
        - 0x8F, 0x90, 0x3D, 0x00, 0x90, 0x21, ASNB or SN, CRC
      - Response will be:
        - SN of tag (Block 0) i.e. 0xC0, 0xBA, 0x5D, 0x03, 0x07, 0x00, 0x12, 0xE0
        - Plus CRC
        - Needs to be rotated as this is LSByte first format (expected)
        - Results in 0xE0120007035DBAC0

Command Opcodes Used (cont.)

- Select be performed before these other commands
  - **Update** (Write Block, to Block X, according to the data sheet)
    - 0x27 + Block Address + Block Data (LSByte first) + CRC
      - 0x27
      - Block Address
        - » for example 0x01, 0x02, etc.
      - Block Data (8 bytes) **make sure you know about what bits do in the given blocks here!!!**
      - CRC (according to prior instructions)
      - For example: TRF79xx command string for updating a block on tag that has been through ACS/Select command set.
        - » 0x8F, 0x90, 0x3D, 0x00, 0xC0, 0x27, Block #, Block Data,
      - Response will be contents written to the block (8 bytes)
  - **Halt**
    - 0x10
      - For example: TRF79xx command string for halting a tag.
        - » 0x8F, 0x90, 0x3D, 0x00, 0x10
      - Once tag is halted, it must be removed from the field or Select command used (with the tag SN).

Other Opcodes which can be implemented

- ACT (0x2E)
- READ4 (0x26)
- DETECT (0xF)
- PAGESEL (0x24)
- READCHECK (0x18, auth command)
- CHECK (0x25, finishes auth command)
- UPDATE(Secured) (0x27)