TRNG Testchip User Guide & Midterm Requirement

Balance You

2020/10/15

PUF security
an eMemory Company

IPR Notice

CONTENT

1 **Delivery**

2 **Schematic & Pin Assignment**

3 **Signal & Waveform**

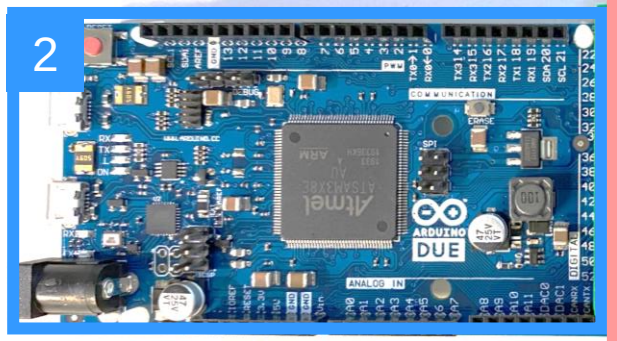
- PUF Mode
- TRNG Mode

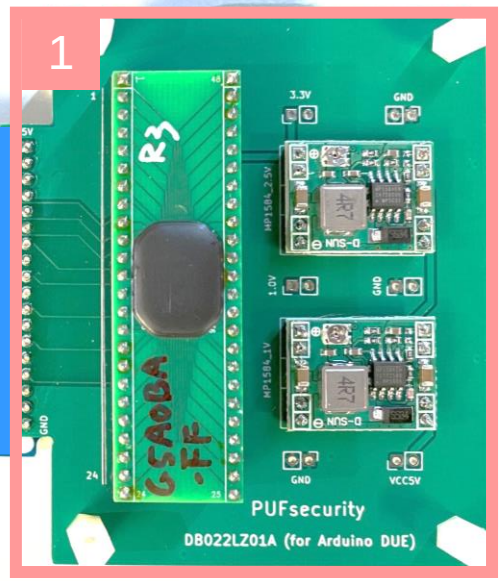
4 **Midterm Task**

PUFsecurity
an eMemory Company

Delivery

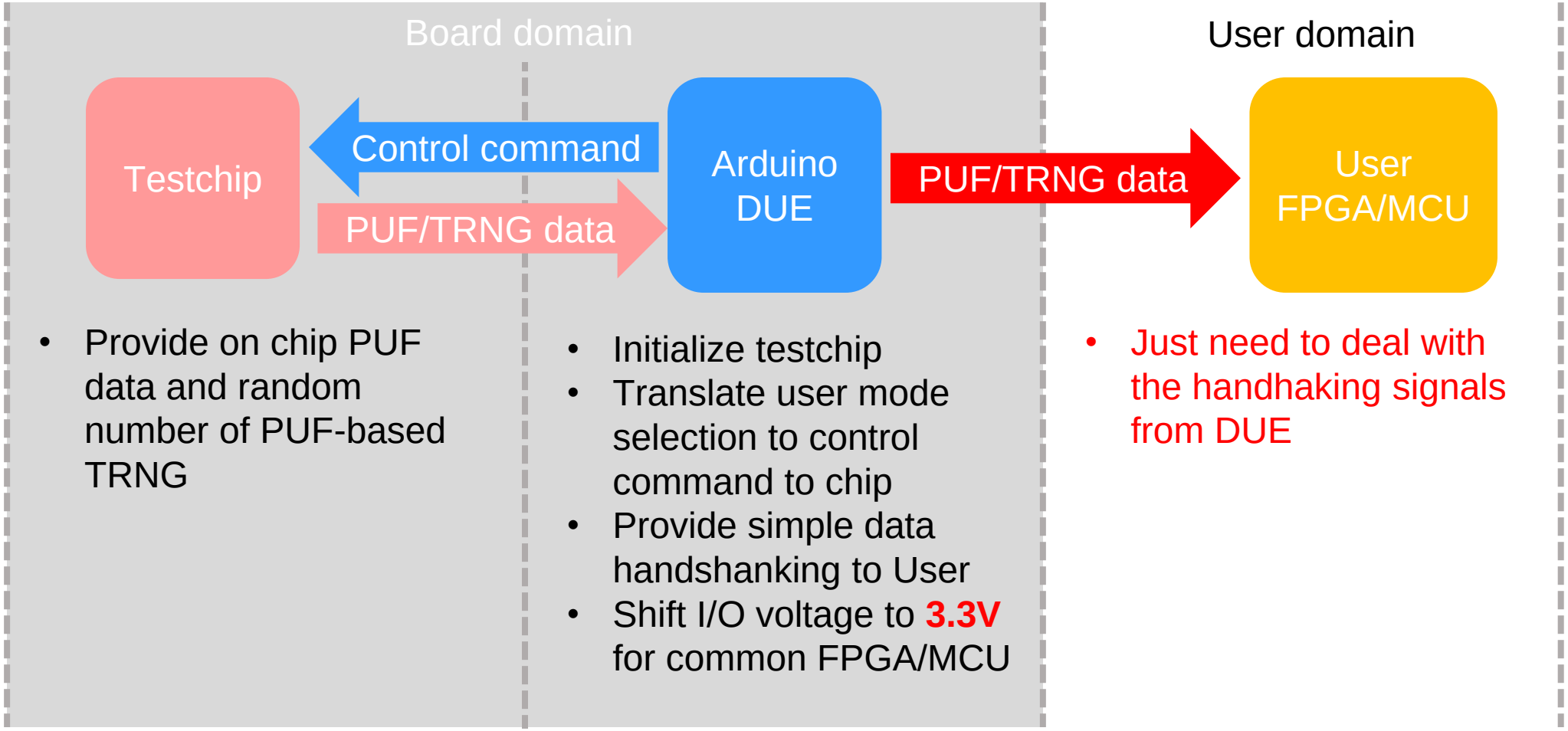
1. Testchip & PCB
2. Arduino DUE
3. USB cable
4. Testchip function testing code (compatible with Arduino DUE)





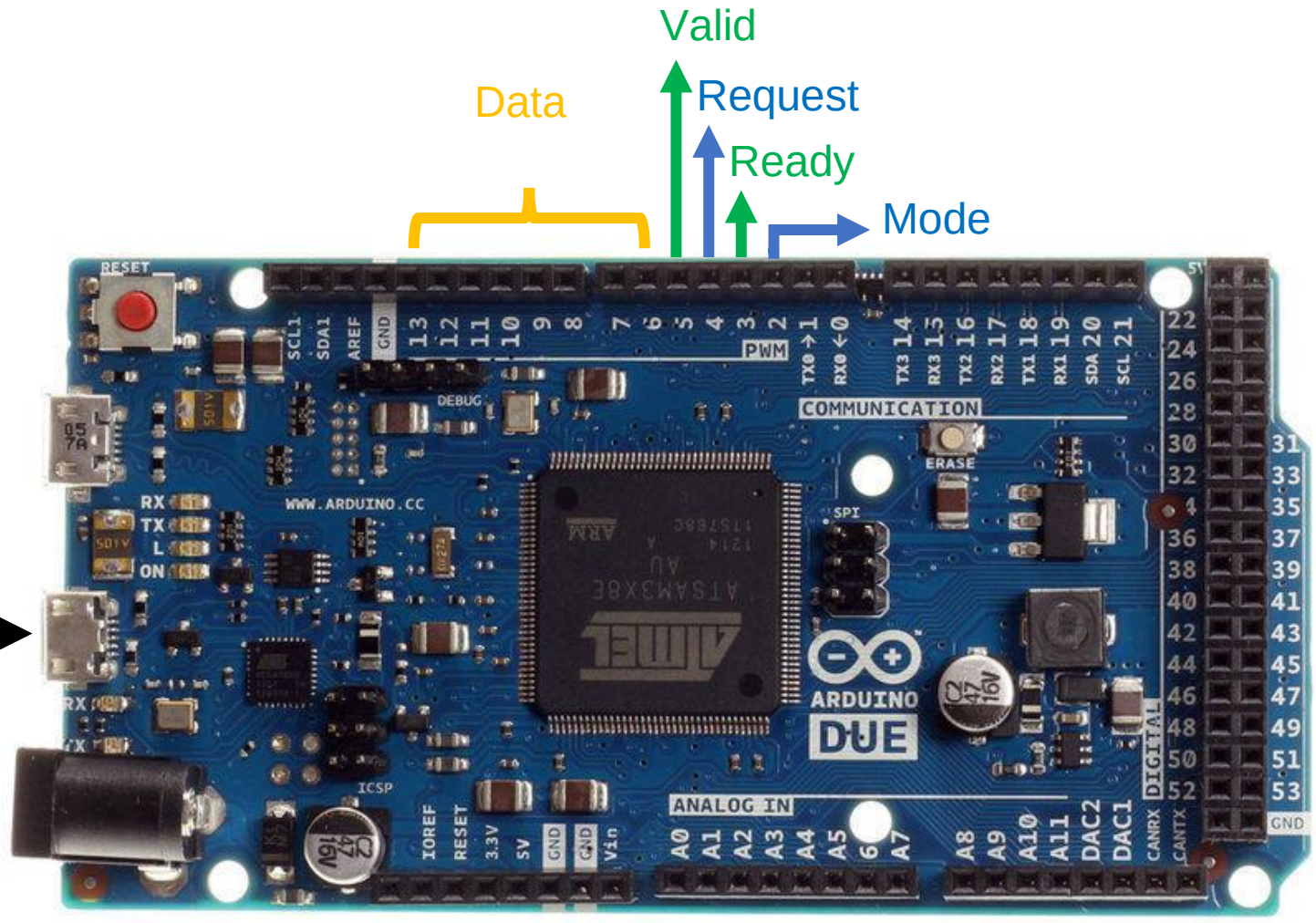
PUFsecurity an eMemory Company

Schematic Diagram



PUFsecurity an eMemory Company

Pin Assignment on DUE



PUF security
an eMemory Company

Pin Assignment on DUE

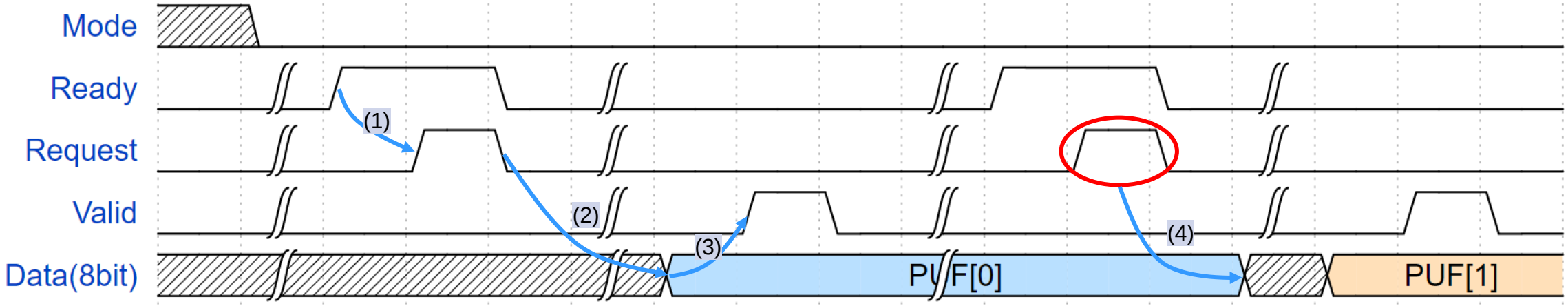
Signal	Direction	Pin @ Arduino DUE	Descriptions
Mode	Input	Digital Pin[2]	Set PUF/TRNG mode
Request		Digital Pin[4]	User request for a random number
Ready	Output	Digital Pin[3]	Whether data* is ready for user to request
Valid		Digital Pin[5]	Whether user can catch a random number on Data pins
Data		Digital Pin[6:13]	8-bit data*

* Data can be either the value of PUF or random number depending on the “Mode” state.

Feature

- Asynchronous data transfer
- Two mode for user to select
  - Set “Mode” pin to LOW: PUF output mode
  - Set “Mode” pin to HIGH: TRNG output mode
  - User can change mode without DUE reset
- 4096 bits PUF data
  - Output byte-by-byte with 512 times handshaking
- PUF-based true random number generator (TRNG)

PUF Mode Signal & Waveform



- Mode pin set LOW
- After 512 times handshaking, the next output PUF data will return to PUF[0] again.

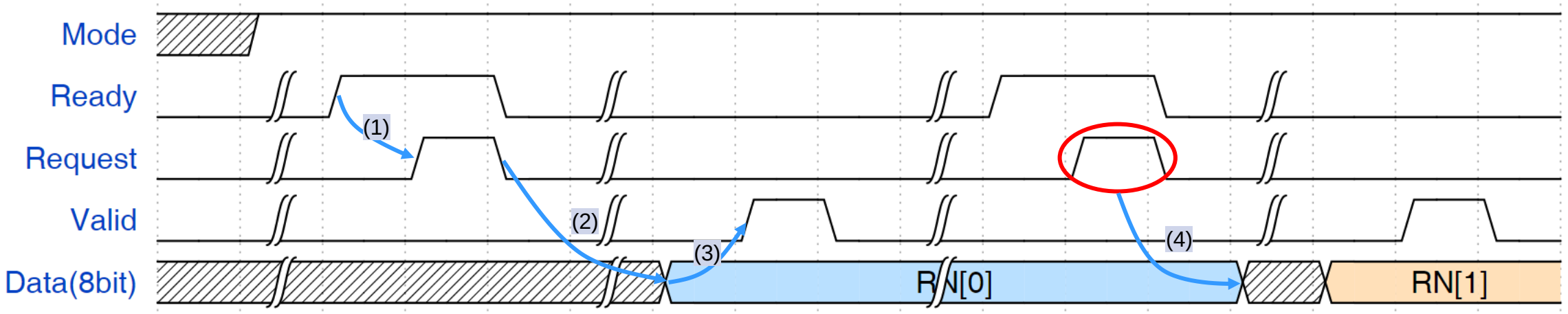
(1) When “Ready” goes HIGH, user can send a “Request” pulse with minimum pulse width 12ns.

(2) When DUE gets “Request”, it starts to set 8 bits PUF value at “Data” pins.

(3) When “Data” pins are ready, DUE generates a pulse with 1 us at “Valid” pin for user detection.

(4) “Data” pins keeps last PUF value until DUE gets next “Request”.

TRNG Mode Signal & Waveform



- Mode pin set HIGH

(1) When “Ready” goes HIGH, user can send a “Request” pulse with minimum pulse width 12ns.

(2) When DUE gets “Request”, it starts to set 8 bits random number at “Data” pins.

(3) When “Data” pins are ready, DUE generates a pulse with 1 us at “Valid” pin for user detection.

(4) “Data” pins keeps last random number value until DUE gets next “Request”.

Midterm task

At least

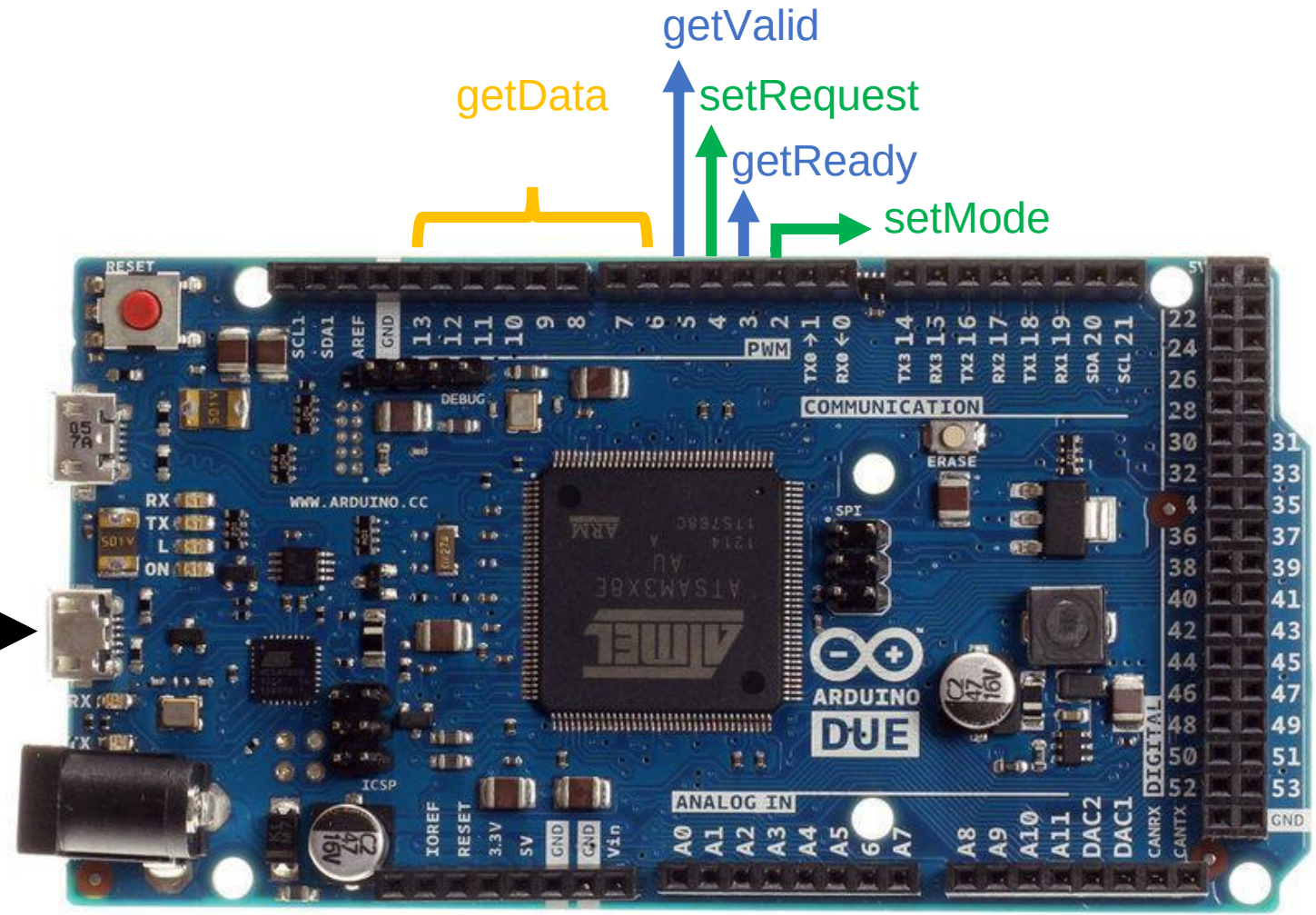
- Write an Arduino program for reading PUF data from the testchip
- Write an Arduino program for reading TRNG data from the testchip
- Write an Arduino program for using 128 bits PUF as the key to encrypt a 128-bit plaintext using AES algorithm

Bonus

- Design two types of S-box, 1) look-up-table based, 2) arithmetic on finite field based
- Compare with the processing time on different types

Proposed Pin Set on Your DUE

Your DUE



USB cable (Program with testing code)

PUF security an eMemory Company

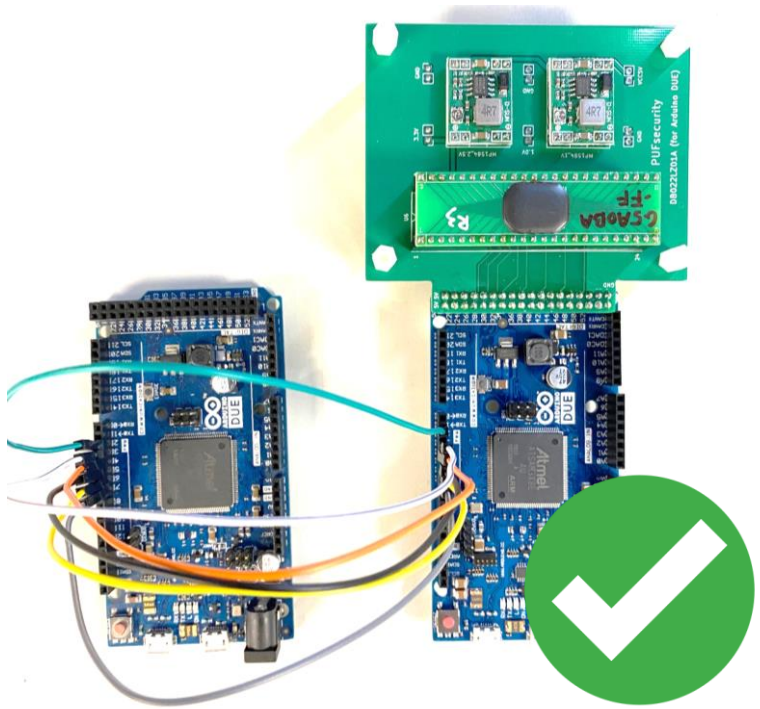
Corresponding Pin Definition in Arduino Code

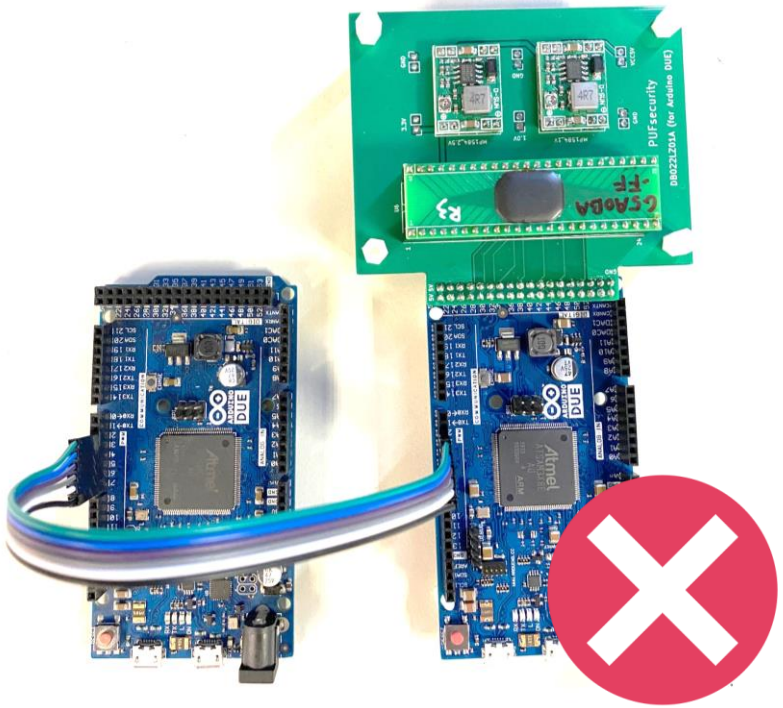
**Warning: Never program any file to the testchip DUE

```
/*** data handshake pins ***/
const byte modePin = 2; //attach to mode pin
const byte requestPin = 4;  //attach to request pin
const byte getReadyPin = 3; //attach to Ready pin
const byte getInterruptPin  = 5; // attach to valid pin
const byte dataPin[8] = {6, 7, 8, 9, 10, 11, 12, 13}; //attach to data pin
```

Note of Pin connection between Two DUEs

- Valid, Ready, Request, Mode signals are **sensitive**, please have those jump wires separated when connecting

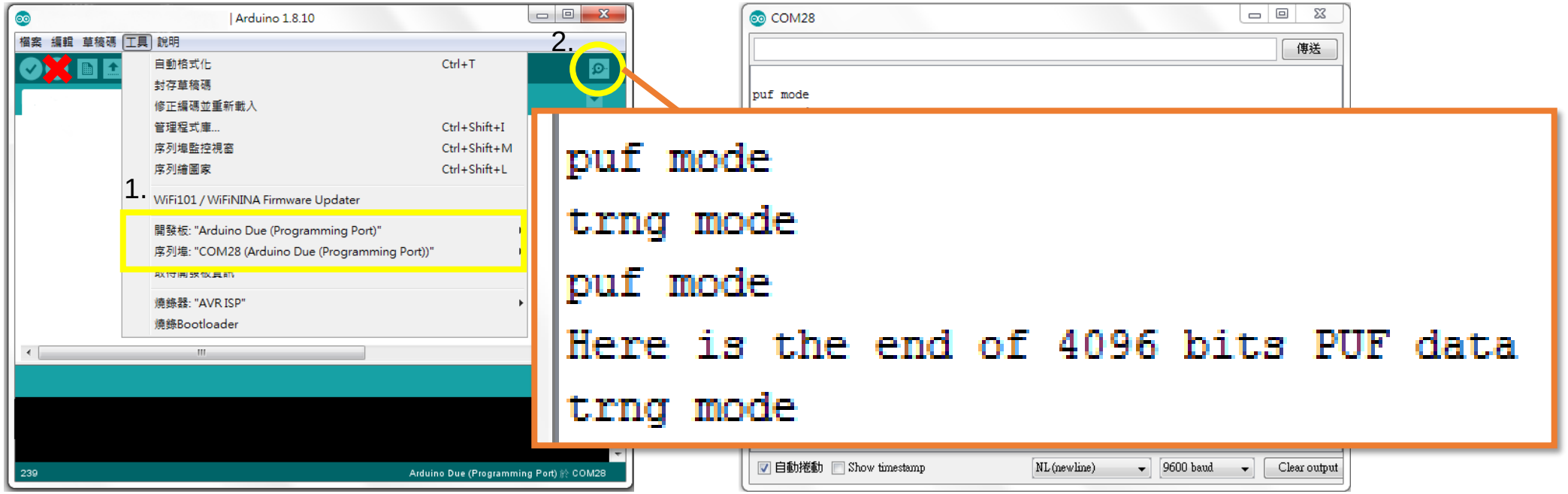




PUFsecurity an eMemory Company

Status of Testchip DUE

- It shows the following information by opening a serial monitor which is connected to the port of testchip DUE
  - Current mode you set
  - Whether you finish the PUF data reading or not(PUF mode only)



PUFsecurity an eMemory Company

THANK YOU

PUF security

Secure the connected world

PUF security
an eMemory Company