

# Secure click

PID: MIKROE-2522

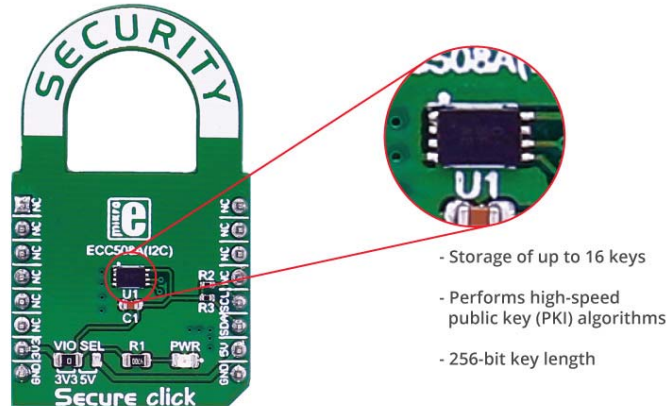
Weight: 25 g



**Secure click** carries the ATECC508A cryptographic coprocessor with secure hardware-based key storage. The click is designed to run on either 3.3V or 5V power supply. It communicates with the target microcontroller over I2C interface.

## ATECC508A features

Microchip's ATECC508A integrates ECDH (Elliptic Curve Diffie Hellman) security protocol, an ultra-secure method to provide key agreement for encryption/decryption. It also integrates the ECDSA (Elliptic Curve Digital Signature Algorithm) sign-verify authentication for the Internet of Things (IoT) market, including home automation, industrial networking, accessory and consumable authentication, medical, mobile and more.



The ATECC508A includes an EEPROM array which **can be used for storage of up to 16 keys**, certificates, miscellaneous read/write, read-only or secret data, consumption logging, and security configurations. Access to the various sections of memory can be restricted in a variety of ways and then the configuration can be locked to prevent changes.


It features a wide array of defense mechanisms specifically designed to prevent physical attacks on the device itself, or logical attacks on the data transmitted between the device and the system. Hardware restrictions on the ways in which keys are used or generated provide further defense against certain styles of attack.

### Specifications

Type	EEPROM
Applications	IoT node security and ID, secure download and boot, ecosystem control, message security, anti-cloning, etc.
On-board modules	ATECC508A cryptographic co-processor
Key Features	Performs high-speed public key (PKI) algorithms, NIST Standard P256 elliptic curve support, SHA-256 hash algorithm with HMAC option, 256-bit key length, storage for up to 16 Keys
Interface	I2C
Input Voltage	3.3V or 5V
Click board size	M (42.9 x 25.4 mm)

## Pinout diagram

This table shows how the pinout on **Secure click** corresponds to the pinout on the mikroBUS™ socket (the latter shown in the two middle columns).

Notes	Pin			Pin	Notes		
	NC	1	AN	PWM	16	NC	
	NC	2	RST	INT	15	NC	
	NC	3	CS	TX	14	NC	
	NC	4	SCK	RX	13	NC	
	NC	5	MISO	SCL	12	<b>SCL</b>	I2C clock
	NC	6	MOSI	SDA	11	<b>SDA</b>	I2C data
Power supply	<b>+3.3V</b>	7	3.3V	5V	10	<b>+5V</b>	Power supply
Ground	<b>GND</b>	8	GND	GND	9	<b>GND</b>	Ground

## Jumpers and settings

Designator	Name	Default Position	Default Option	Description
JP1	PWR.SEL.	Left	3.3V	Power Supply Voltage Selection 3.3V/5V, left position 3.3V, right position 5V

## Programming

Code examples for Secure click, written for MikroElektronika hardware and compilers are available on Libstock.

### *Code snippet*

The following code snippet gets the public key from slot 0, generates the signature for a message using that key, and then verifies that signature.

```

01 if (atcab_get_pubkey(0, publicKeyBuffer) == ATCA_SUCCESS)
02     {
03         LOG( "rnrn Public key: " );
04         outputHex (publicKeyBuffer, 64);
05     }
06 else LOG( "rn Getting public key failed..." );
07 //Generates the signature from input message, using key in slot 0.
08 if (atcab_sign(0, messageBuffer, signatureBuffer) == ATCA_SUCCESS)
09     {
10         LOG( "rnrn Generating signature: " );
11         outputHex (signatureBuffer, 64);
12     }
13 else LOG( "rn Generating signature failed..." );
14 //Verification of signature
15 LOG( "rnrn Signature verification..." );
16 if (atcab_verify_extern(messageBuffer, signatureBuffer,
publicKeyBuffer, responseBuffer) == ATCA_SUCCESS)
17     {
18         if (responseBuffer [0])
19             LOG( "rn Signature is valid " );
20         else LOG( "rn Signature is invalid " );
21     }
22 else LOG( "rn Verification operation failed..." );

```

