

IoT Protocol Stack and IoT Application layer protocol - CoAP

Typical IoT stack for IEEE 802.15.4 Motes

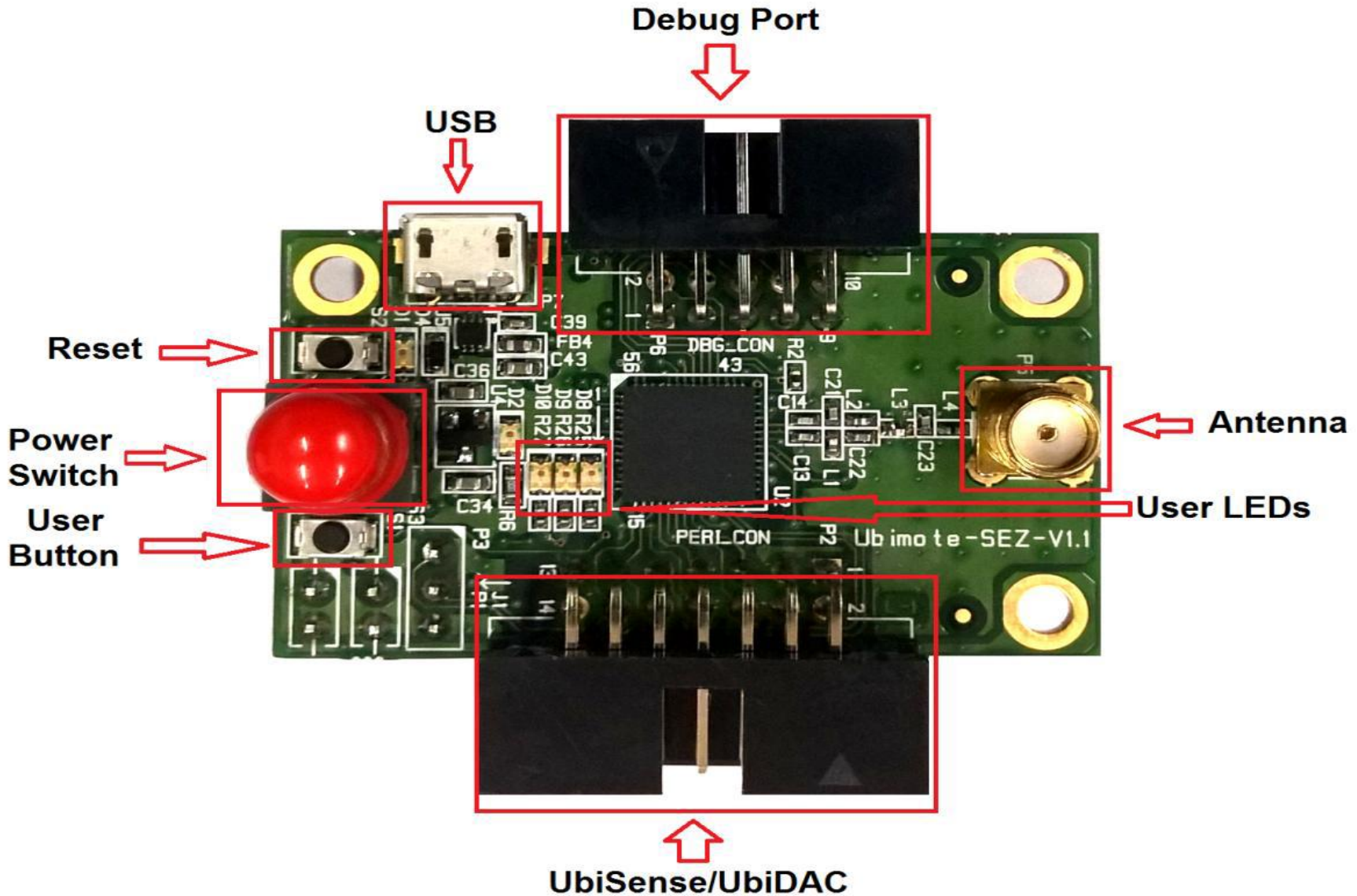
Application
Transport
Network, Routing
Adaptation
MAC
Duty Cycle
Radio

CoAP MQTT Websockets HTTP
TCP UDP
IPv4 IPv6 RPL
6LoWPAN
CSMA/CA
ContikiMAC
802.15.4

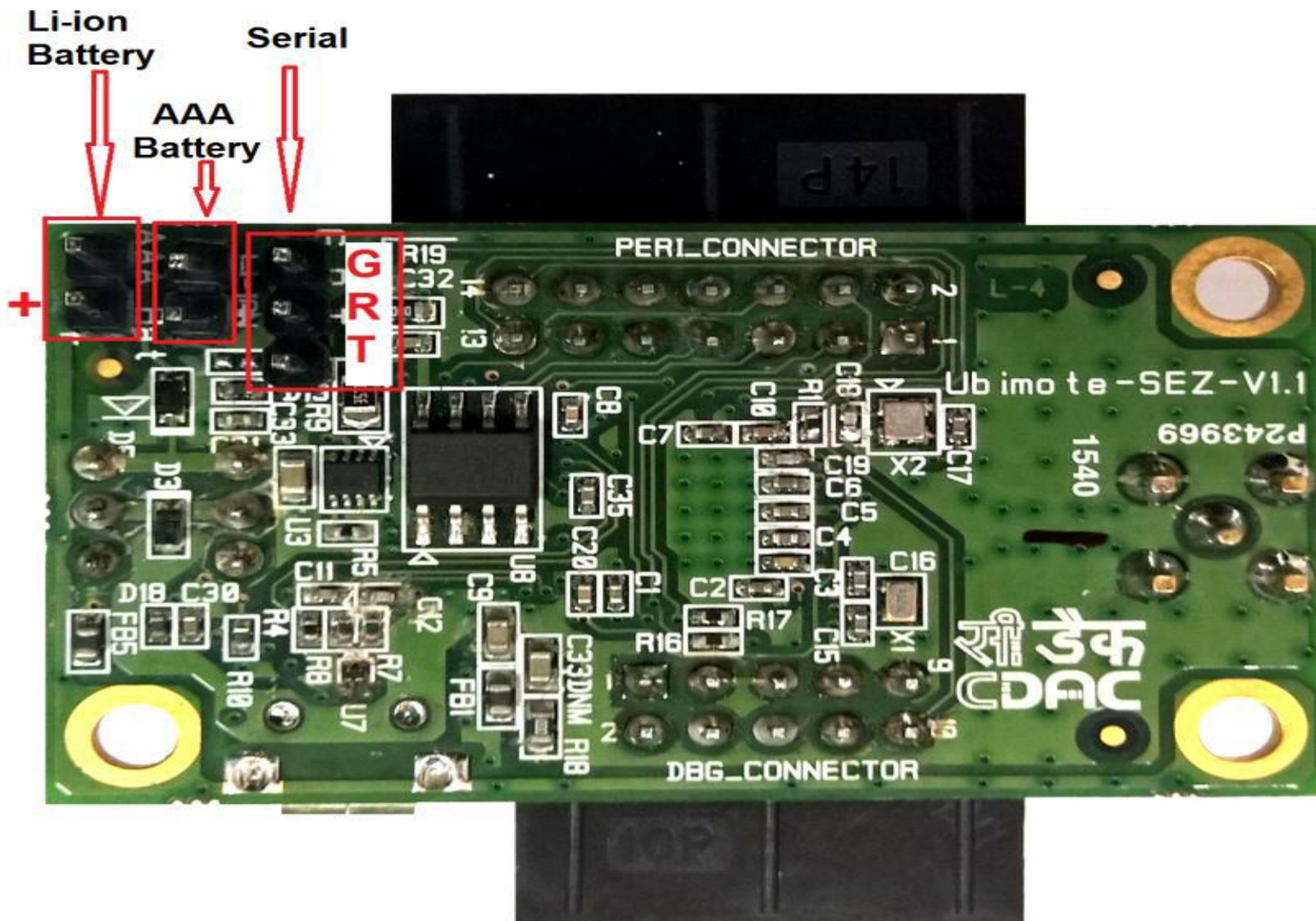
Ubimote

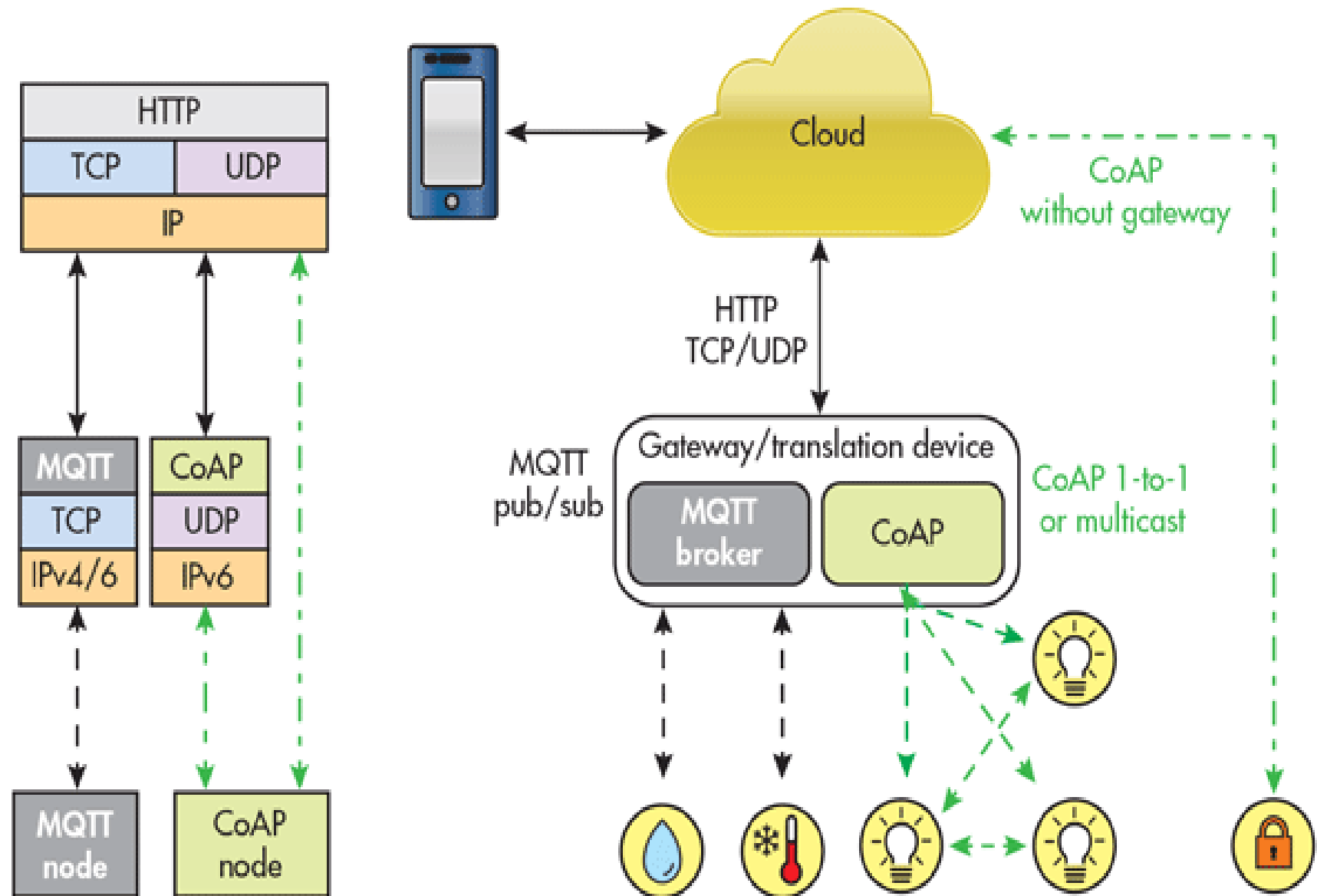
- TI CC2538, an ARM CortexM3 with 32 KB on-chip RAM and 512 KB on-chip flash with a robust IEEE 802.15.4 radio.
- SMA connector for interfacing 2.4 GHz SMA antenna
- 8Mbit Serial Flash memory for additional storage.
- Sensor Connector for interfacing UbiSense/UbiDAC
- JTAG Connector for programming and debugging
- USB interface for peripheral functionality.
- Power through USB Connector, Li-Po Battery
- Li-Po Battery charger

Ubimote - Components



Ubimote - Components





<http://www.electronicdesign.com/iot/mqtt-and-coap-underlying-protocols-iot>

CoAP

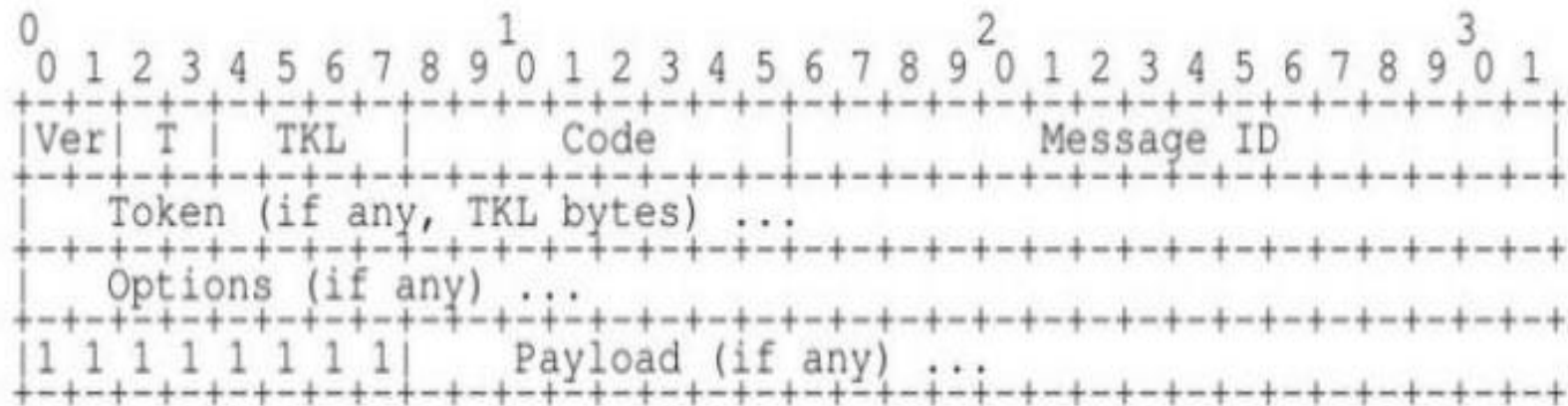
CoAP

RFC 7252 Constrained Application Protocol

“The Constrained Application Protocol (CoAP) is a specialized web transfer protocol for use with constrained nodes and constrained networks in the **Internet of Things**.

The protocol is designed for machine-to-machine (M2M) applications such as smart energy and building automation.”

- UDP-*reliable* (confirmable), SMS supported
- CoRE *Link-format* (GET /.well known/core)
- Client/Server
- IANA Registered (error codes, content format)
- Resource Discovery and asynchronous subscription
- Four-bytes compact header
- Multicast and one-to-one supported
- HTTP verbs GET, PUT, POST, DELETE
- HTTP-like header (*Options*)
- URI (Uniform Resource Identifier)



Ver - Version (1)

T - Message Type (Confirmable, Non-Confirmable, Acknowledgement, Reset)

TKL- Token Length, if any, the number of Token bytes after this header

Code - Request Method (1-10) or Response Code (40-255)

Message ID - 16-bit identifier for matching responses

Token - Optional response matching token

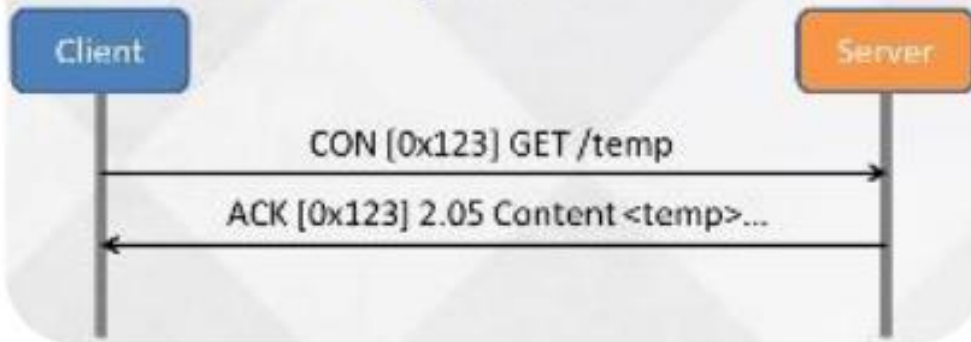
<http://www.slideshare.net/zdshelby/coap-tutorial>

CoAP URI

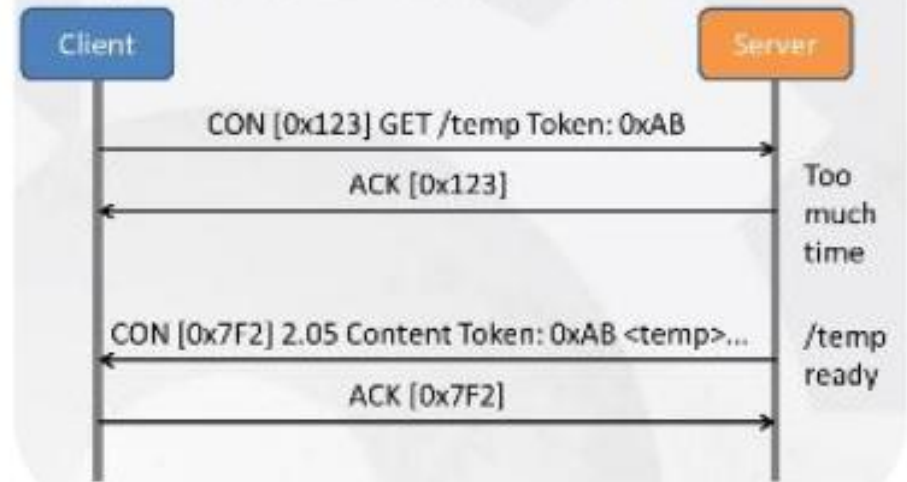
coap://[aaaa::c30c:0:0:1234]:5683/actuators/leds?color=b

Host	Port	Path	Query
------	------	------	-------

Confirmable request



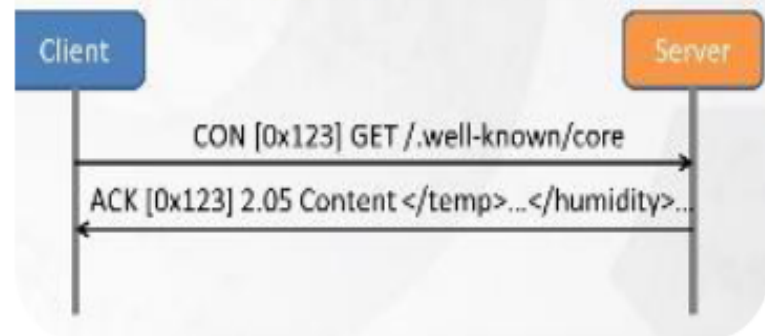
Response back after a while



Observer



Resource discovery



Only on Firefox: install the following plug-in:



Copper (Cu) 0.18.4.1-signed

by Matthias Kovatsch

The Copper (Cu) CoAP user-agent for Firefox installs a handler for the 'coap' URI scheme and allows users to browse and interact with Internet of Things devices.

Only with Firefox — Get Firefox Now!

This add-on has been preliminarily reviewed by Mozilla. [Learn more](#)

<https://addons.mozilla.org/en-US/firefox/addon/copper-270430/>
<http://people.inf.ethz.ch/mkovatsc/copper.php>

CoAP server for resource discovery – an Implementation Snapshot

ContikiRPL x [aaaa::c30c:0:0:13c8] x

coap://[aaaa::c30c:0:0:13c8]:5683/sensors/adxl345

Discover Ping GET POST PUT DELETE Observe Payload Text Behavior CoAP 18

[aaaa::c30c:0:0:13c8]:5683

2.05 Content (Blockwise) (Download finished)

[aaaa::c30c:0:0:13c8]:5683

- .well-known
 - core
- actuators
 - leds
 - toggle
- sensors
 - adxl345
 - button
 - test
 - hello
 - separate

Hea...	Value	Option	Value	Info
Type	Acknowledgment	Content-Format	application/json	50
Code	2.05 Content	Block2	1 (32 B/block)	1 byte
Message...	36998			
Token	empty			

Combined Payload (37)

Incoming Rendered Outgoing

```

{
  adxl345:
  {
    X: 24
    Y: 135
    Z: 18
  }
}

```

☒ Debug Control Reset

Token

use hex (0x..) or string x

Request Options

Accept

application/json

Content-Format

application/json

Block1 (Req.) Block2 (Res.) Auto

block no. x block no. x ☒

Size1 Size2

total size x total size x

Observe

use integer x

ETag

use hex (0x..) or string x