
Tausand AB1000 Matlab library example: Abacus Example

Table of Contents

Find devices and establish a connection	1
Read device settings	3
Write device settings	4
Read measurements from device	4
Multiple coincidence setting and reading	5
Close connection	6

A set of basic commands to test Tausand_AB1000_MatlabLibrary to be used in Matlab's command window.

Author: David Guzmán. Tausand Electronics, Colombia.

Created: 2021-03. Last revision: 2021-03-15. Version: 1.1.

Contact email: dguzman@tausand.com. Website: <http://www.tausand.com>

Find devices and establish a connection

```
disp('*****');
disp('MATLAB Abacus example');
disp('*****');
disp('1. Find devices and establish a connection');

*****
MATLAB Abacus example
*****

1. Find devices and establish a connection

Find Tausand Abacus devices with reference AB1504.

disp('1a. Find Tausand Abacus devices with reference AB1504');
[ports,n] = findDevices(1504); %1504: look for only 1504 devices
disp(['Found ',int2str(n),' device(s) with reference 1504']);

1a. Find Tausand Abacus devices with reference AB1504
Progress: 1/2
Progress: 2/2
Found 1 device(s) with reference 1504

Find Tausand Abacus devices with any reference.

disp('1b. Find Tausand Abacus devices with any reference');
[ports,n] = findDevices(); %(): any device
disp(['Found ',int2str(n),' device(s) with any reference']);

if n==0
```

```
    disp('No valid devices were found. Closing.');
```

```
    return
```

```
end
```



```
disp('Available valid devices:');
```

```
ports
```



```
1b. Find Tausand Abacus devices with any reference
```

```
Progress: 1/2
```

```
Progress: 2/2
```

```
Found 1 device(s) with any reference
```

```
Available valid devices:
```



```
ports =
```



```
    "COM23"
```



```
Connect to the first device found.
```



```
disp('Select the first found device to connect with.');
```

```
my_device_port = ports{1}
```

```
my_abacus = openAbacus(my_device_port)
```



```
disp('Connected to the following device:');
```

```
myidnstring = idnQuery(my_abacus);
```

```
mydevicetype = deviceTypeQuery(my_abacus);
```

```
disp(['idn (string): ',char(9),char(9),char(9),myidnstring]);
```

```
disp(['device type (integer): ',char(9),int2str(mydevicetype)]);
```



```
Select the first found device to connect with.
```



```
my_device_port =
```



```
    'COM23'
```



```
Serial Port Object : Serial-COM23 AB1504
```



```
Communication Settings
```

```
    Port:                COM23
```

```
    BaudRate:            115200
```

```
    Terminator:          'LF'
```



```
Communication State
```

```
    Status:               open
```

```
    RecordStatus:        off
```



```
Read/Write State
```

```
    TransferStatus:      idle
```

```
    BytesAvailable:      0
```

```
    ValuesReceived:      21
```

```
    ValuesSent:          7
```

```
Connected to the following device:
idn (string):    Tausand Abacus AB1504
device type (integer): 1504
```

Read device settings

Example of reading all device settings

```
disp("*****")
disp("2. Read device settings")
disp('Settings read from device, using queryAllSettings function:');
[my_sett_data,my_sett_labels]=queryAllSettings(my_abacus);
current_settings=[my_sett_labels,my_sett_data];
disp(current_settings);
```

```
*****
```

```
2. Read device settings
```

```
Settings read from device, using queryAllSettings function:
```

```
"sampling"           "2000"
"coincidence_window" "100"
"delay_A"            "0"
"delay_B"            "20"
"delay_C"            "36"
"delay_D"            "0"
"sleep_A"            "0"
"sleep_B"            "20"
"sleep_C"            "0"
"sleep_D"            "0"
"config_multiple_1"  "224"
```

Examples reading single settings

```
disp('Query each setting by separate:');
value=queryDelay(my_abacus, 'A');
disp([' Current delay in A: ',int2str(value), 'ns']);
value=queryDelay(my_abacus, 'B');
disp([' Current delay in B: ',int2str(value), 'ns']);
value=querySleep(my_abacus, 'A');
disp([' Current sleep in A: ',int2str(value), 'ns']);
value=querySleep(my_abacus, 'B');
disp([' Current sleep in B: ',int2str(value), 'ns']);
value=queryCoincidenceWindow(my_abacus);
disp([' Current coincidence window: ',int2str(value), 'ns']);
value=querySamplingTime(my_abacus);
disp([' Current sampling: ',int2str(value), 'ms']);
text=queryMultipleCoincidence(my_abacus);
if text~=""
    disp([' Current multiple coincidence setting: ',text]);
end
```

```
Query each setting by separate:
```

```
Current delay in A: 0ns
Current delay in B: 20ns
```

```
Current sleep in A: 0ns  
Current sleep in B: 20ns  
Current coincidence window: 100ns  
Current sampling: 2000ms  
Current multiple coincidence setting: ABC
```

Write device settings

Examples of writing a new setting value

```
disp('*****')  
disp('3. Write device settings')  
  
*****  
3. Write device settings  
  
Set sampling time = 2000ms = 2s  
  
configureSamplingTime(my_abacus,2000);    %set sampling=2000ms  
value = querySamplingTime(my_abacus);  
disp([' Current sampling:',int2str(value),'ms']);  
  
Current sampling:2000ms
```

Set delay in channel B = 20ns

```
configureDelay(my_abacus,'B',20);    %set delay_B=20ns  
value = queryDelay(my_abacus,'B');  
disp([' Current delay B:',int2str(value),'ns']);  
  
Current delay B:20ns
```

Set coincidence window = 100ns

```
configureCoincidenceWindow(my_abacus,100);    %set coinc_wind=100ns  
value = queryCoincidenceWindow(my_abacus);  
disp([' Current coincidence window:',int2str(value),'ns']);  
  
Current coincidence window:100ns
```

Read measurements from device

```
disp('*****')  
disp('4. Read measurements from device')  
disp('Waiting to complete a measurement:');  
waitForAcquisitionComplete(my_abacus,true);    %using default  
max_wait=10 seconds, printing messages  
disp('Waiting to complete a measurement with a maxtime (1s) shorter  
than sampling time (2s):');  
waitForAcquisitionComplete(my_abacus,false,1); %using  
max_wait=1second, no printing messages  
  
*****  
4. Read measurements from device  
Waiting to complete a measurement:
```

```
Current ID is 29
Next data is available in 0.0s
Now, current ID is 30
Waiting to complete a measurement with a maxtime (1s) shorter than
sampling time (2s):
Warning: Maxwait expired. Consider extending
your maxwait.
```

Read data from device

```
disp('Measurements read from device, using readMeasurement
function:');
[my_meas_data,my_meas_labels]=readMeasurement(my_abacus);
current_measurements=[my_meas_labels,my_meas_data];
disp(current_measurements);
```

Measurements read from device, using readMeasurement function:

```
"counters_ID"          "30"
"counter_A"            "0"
"counter_B"            "0"
"counter_C"            "0"
"counter_D"            "0"
"counter_AB"           "0"
"counter_AC"           "0"
"counter_AD"           "0"
"counter_BC"           "0"
"counter_BD"           "0"
"counter_CD"           "0"
"counter_multiple_1"   "0"
"time_left"            "939"
```

Multiple coincidence setting and reading

```
text=queryMultipleCoincidence(my_abacus);
if text~=""
    disp('*****')
    disp('5. Multiple coincidence setting and reading')
    %if using a device able to measure multiple coincidences
    disp([' Current multiple coincidence setting: ',text]);
    configureMultipleCoincidence(my_abacus,'ABC');
    text=queryMultipleCoincidence(my_abacus);
    disp([' New multiple coincidence setting: ',text]);
    waitForAcquisitionComplete(my_abacus); %using default
max_wait=10 seconds
    [my_meas_data,my_meas_labels]=readMeasurement(my_abacus);
    my_index = find(my_meas_labels=='counter_multiple_1',1);
    if ~isempty(my_index)
        value = my_meas_data(my_index);
        disp([' Current coincidences in ABC: ',int2str(value)]);
    end
end
*****
```

5. Multiple coincidence setting and reading
Current multiple coincidence setting: ABC
New multiple coincidence setting: ABC
Current coincidences in ABC: 0

Close connection

```
closeAbacus(my_abacus)
```

Published with MATLAB® R2017a