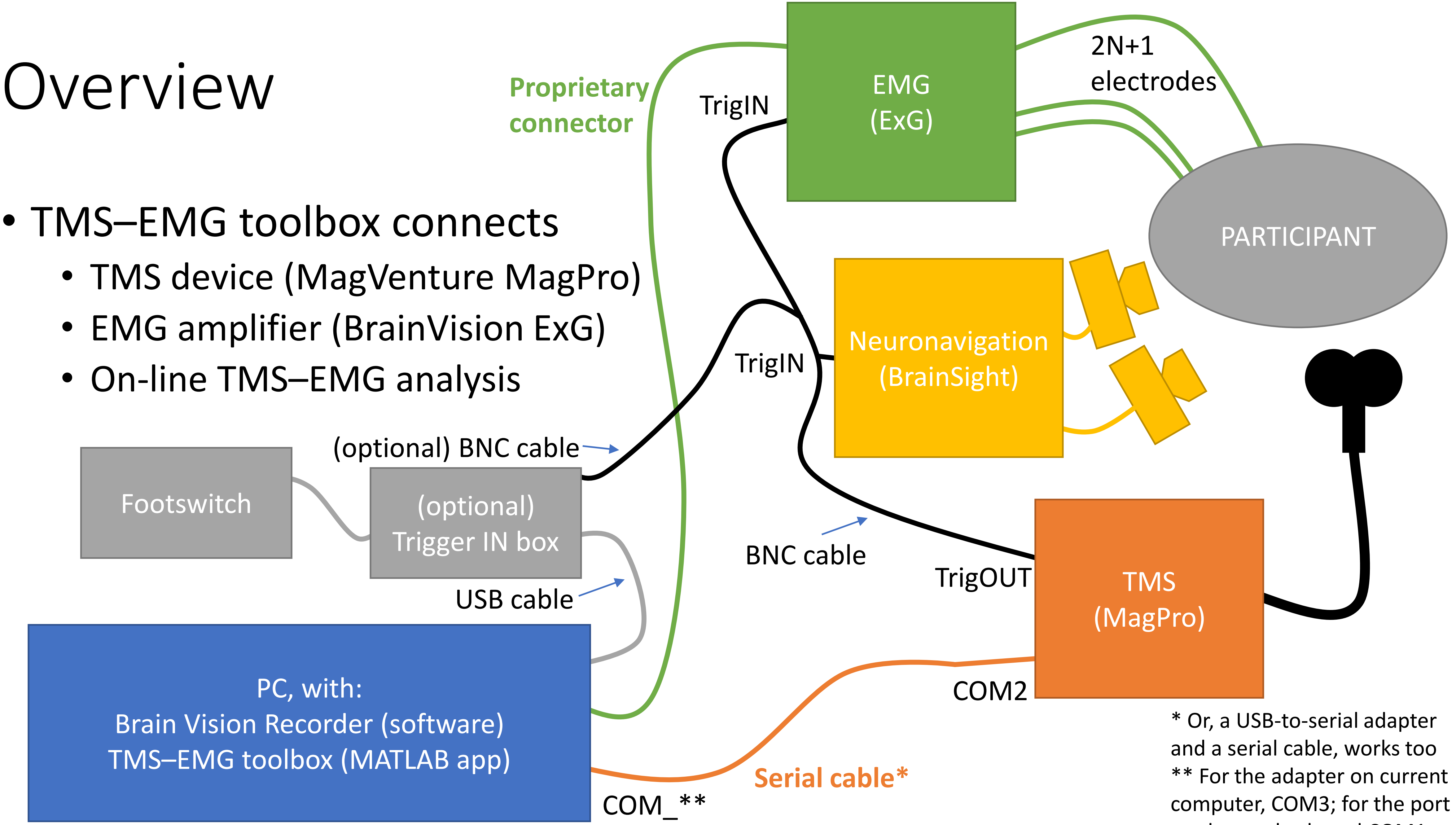
TMS–EMG toolbox, user manual

Lari Koponen

2021-04-21

Overview

- TMS–EMG toolbox connects
  - TMS device (MagVenture MagPro)
  - EMG amplifier (BrainVision ExG)
  - On-line TMS–EMG analysis



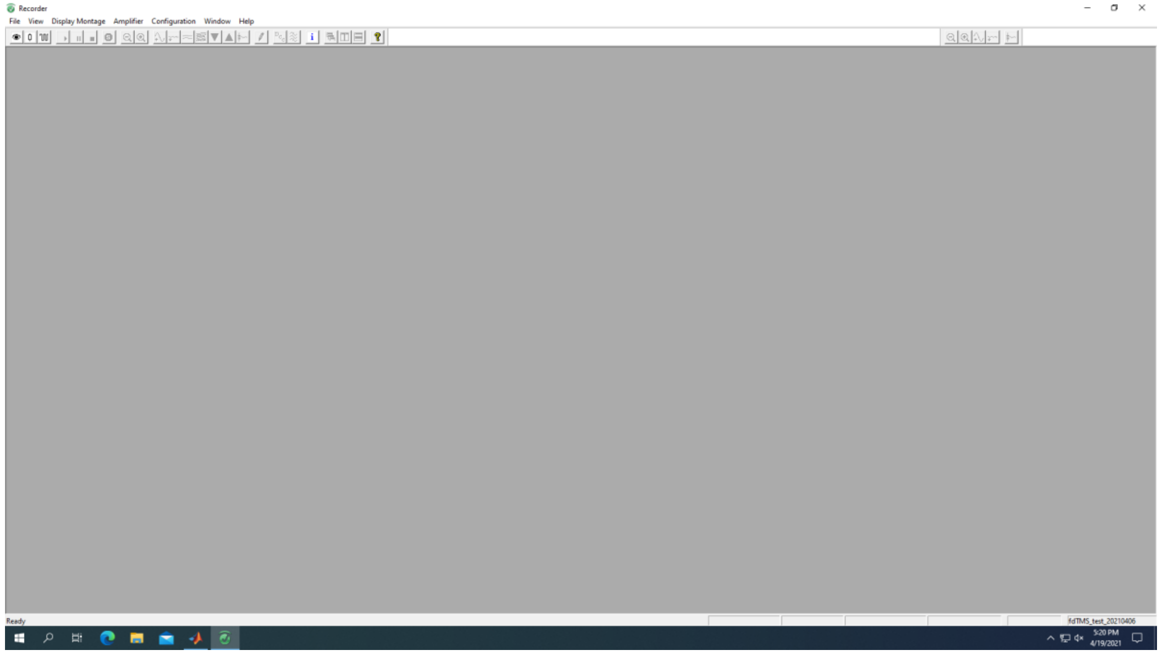
* Or, a USB-to-serial adapter and a serial cable, works too

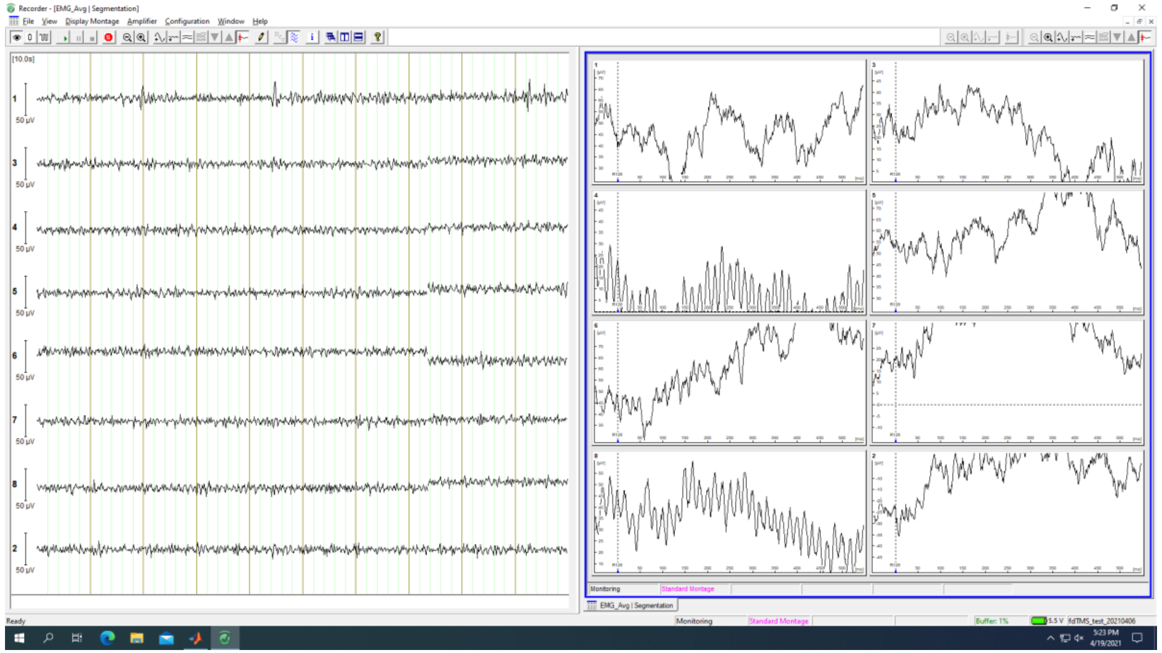
** For the adapter on current computer, COM3; for the port on the motherboard COM1

Usage 1/5

Recommended use pattern

1. Turn on TMS device
2. Turn on ExG
3. Start BrainVision recorder
4. **SEND TEST PULSES**
    1. **BrainVision Recorder sometimes does not capture the first trigger after turn-on, we do not want this to occur during recording**

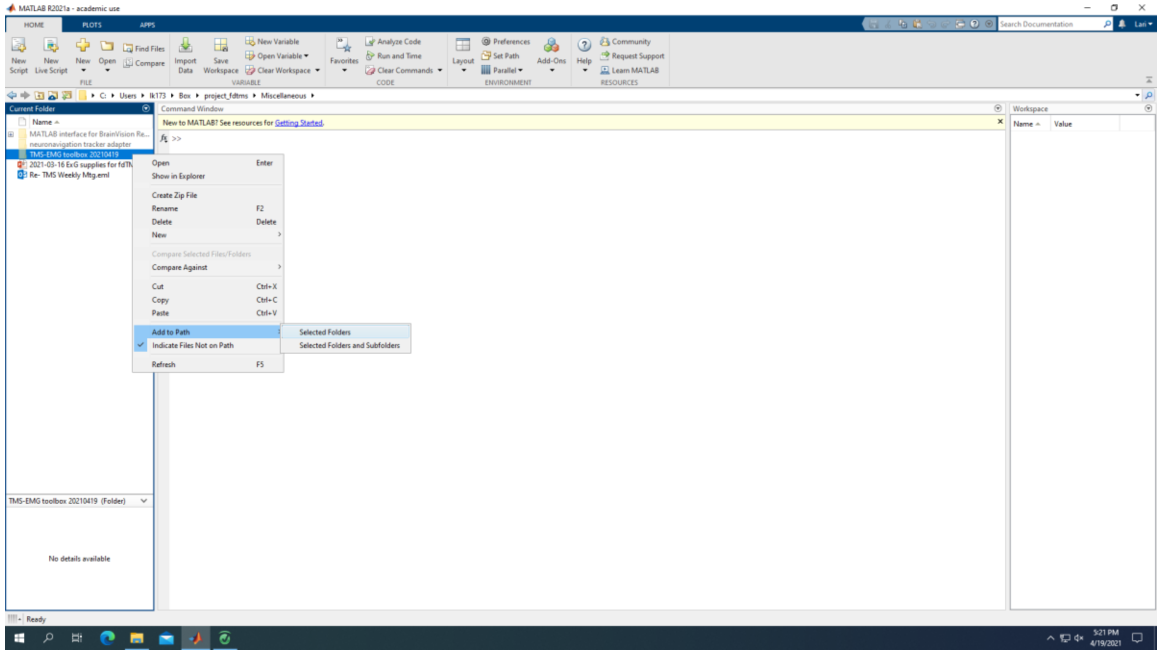

Recorder
File View Display Montage Amplifier Configuration Window Help
Ready


Recorder - [EMG_Avg | Segmentation]
File View Display Montage Amplifier Configuration Window Help
[10.0s]
50 µV
Monitoring
Standard Montage
EMG_Avg | Segmentation
Ready

Usage 2/5

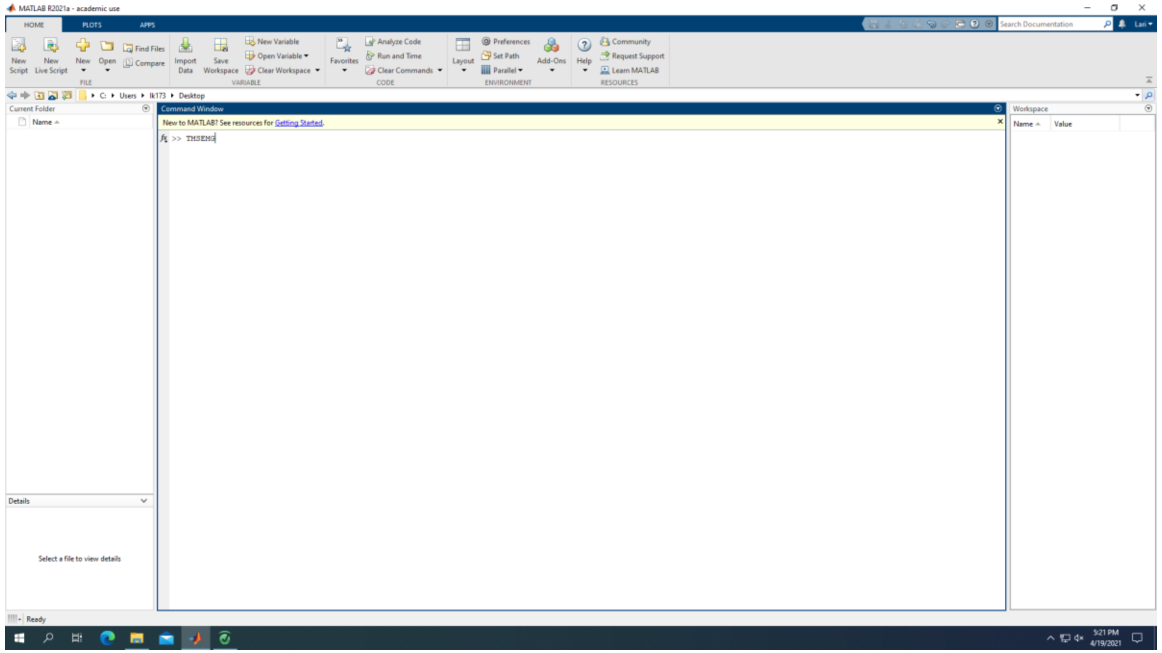
Recommended use pattern

1. Start MATLAB
   1. Add ‘TMS-EMG toolbox 20210419’ to path
   2. Navigate to a folder where you want to record the data
   3. Run command ‘TMSEMG’



New to MATLAB? See resources for Getting Started.

```
>>
```



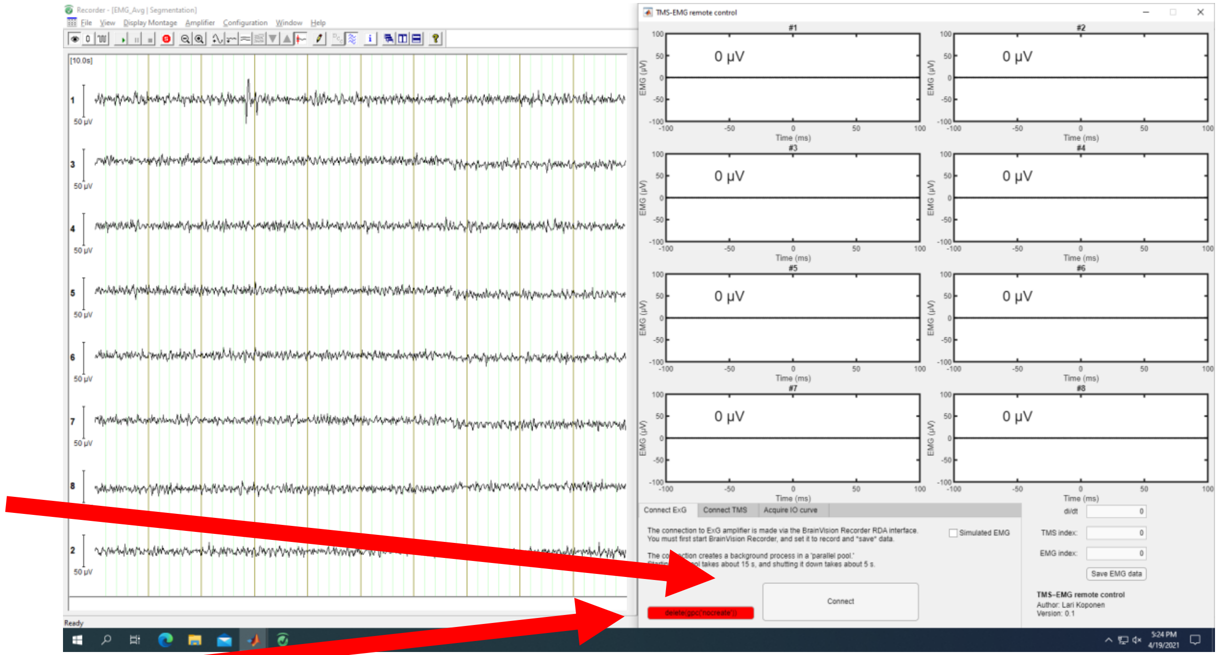
New to MATLAB? See resources for Getting Started.

```
>> TMSEMG
```

Usage 3/5

Recommended use pattern

1. On the ‘Connect ExG’ tab, press connect
    1. This takes 15–20 s
    2. In case an error popup shows, about ‘Parallel Processing Toolbox’, and you are not running a second copy of the software, push the red button
2. On the ‘Connect TMS’ tab, name the (raw) log file for TMS pulses and press connect
    1. In case an error popup shows, serial port is already in use (run ‘instrreset’, check device manager)



The connection to ExG amplifier is made via the BrainVision Recorder RDA interface.
You must first start BrainVision Recorder, and set it to record and "save" data.

The connection creates a background process in a 'parallel pool.'
Starting the pool takes about 15 s, and shutting it down takes about 5 s.

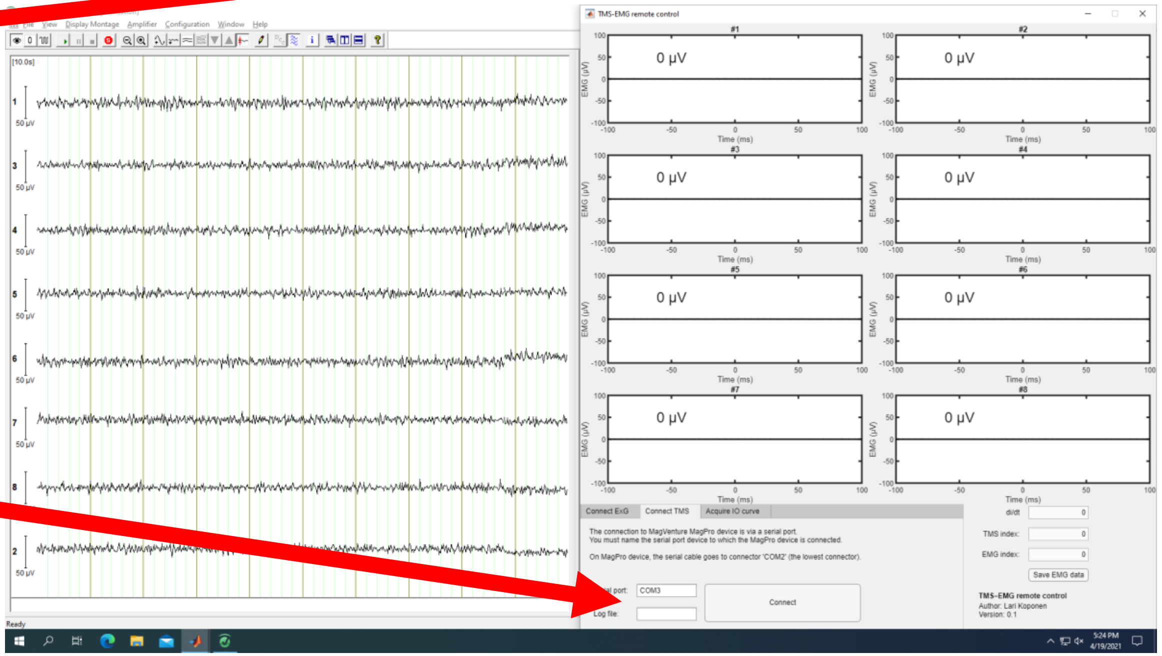
□ Simulated EMG

di/dt: 0

TMS index: 0

EMG index: 0

TMS–EMG remote control
Author: Lari Koponen
Version: 0.1



The connection to MagVenture MagPro device is via a serial port.
You must name the serial port device to which the MagPro device is connected.

On MagPro device, the serial cable goes to connector 'COM2' (the lowest connector).

Serial port: COM3

Log file:

di/dt: 0

TMS index: 0

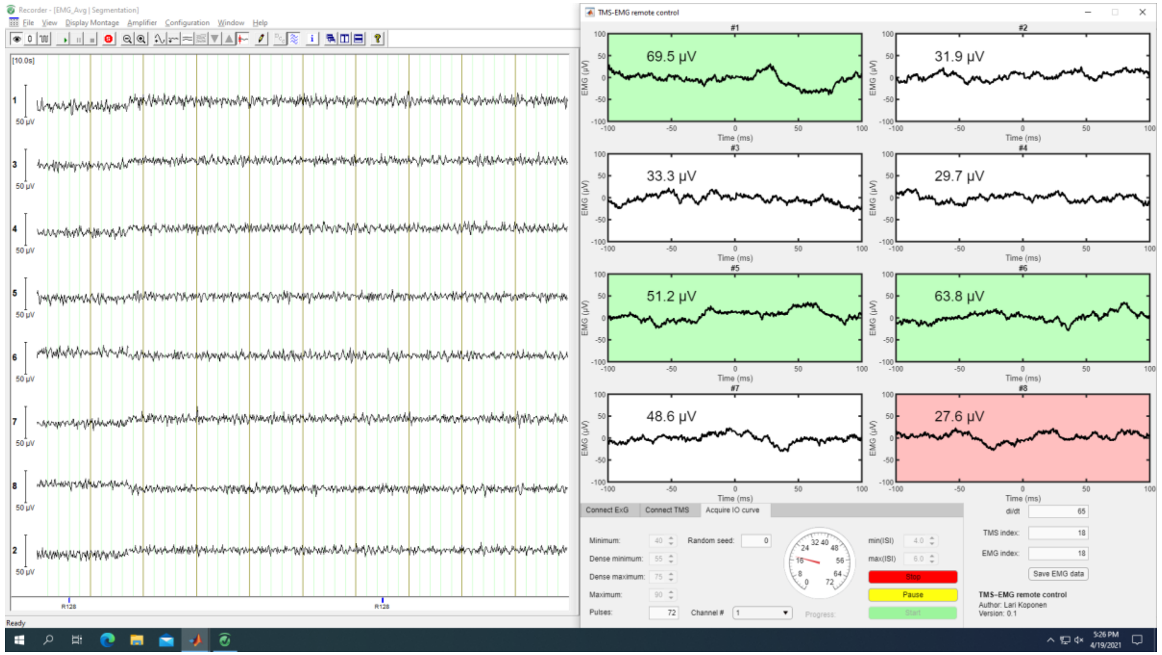
EMG index: 0

TMS–EMG remote control
Author: Lari Koponen
Version: 0.1

Usage 4/5

Recommended use pattern

1. Use ‘Acquire IO curve’ tab
   1. One pulse per intensity between min & max
   2. Two pulses between dense min & dense max
   3. Random seed for pseudorandom sampling sequence (pulse are saved into the ‘EMG’ data output)
   4. Channel number (just for note, saved to ‘EMG’ data output)
   5. ISI, uniformly random between limits
   6. **Start: start sampling**
   7. **Pause sampling (resume with start)**
   8. **Stop: cancel sampling (permanently, start starts from beginning)**



Minimum: 40

Dense minimum: 55

Dense maximum: 75

Maximum: 90

Pulses: 72

Random seed: 0

Channel # 1

min(ISI) 4.0

max(ISI) 6.0

di/dt 65

TMS index: 18

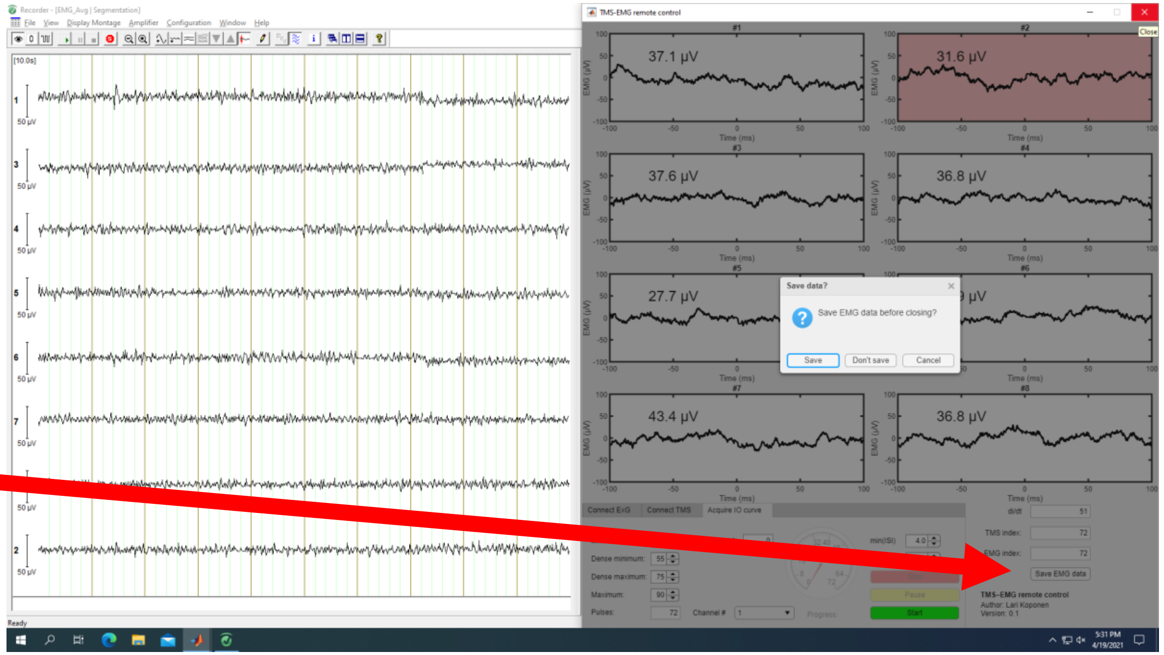
EMG index: 18

TMS–EMG remote control
Author: Lari Koponen
Version: 0.1

- IO curve acquisition can be started or paused with a press of the footswitch.
  - The footswitch connects to the OPTIONAL trigger input box, which further (optionally) takes trigger output signal from the TMS device.
  - If this signal is present, the output data file will contain field ‘triggerIN’ with a ‘datetime’ object for each trigger signal by the TMS device, this data can be used to resolve the ambiguity with paired pulses
    - The on-line trigger counts follow the serial data, the correction is left to the user off-line, POST session
  - These signals are generated even if the TMS device is not connected in software, as long as the TMSEMG window is active!

Usage 5/5

1. Saving data
   1. ‘Save EMG data’ button
      1. Saves all data acquired during current session to ‘.mat’ file
   2. If not saved, asks on close
2. Example of (synthetic) error
   1. COM3 is already in use
      1. By running the TMS device backend manually
      2. Solvable by ‘delete()’ of that object

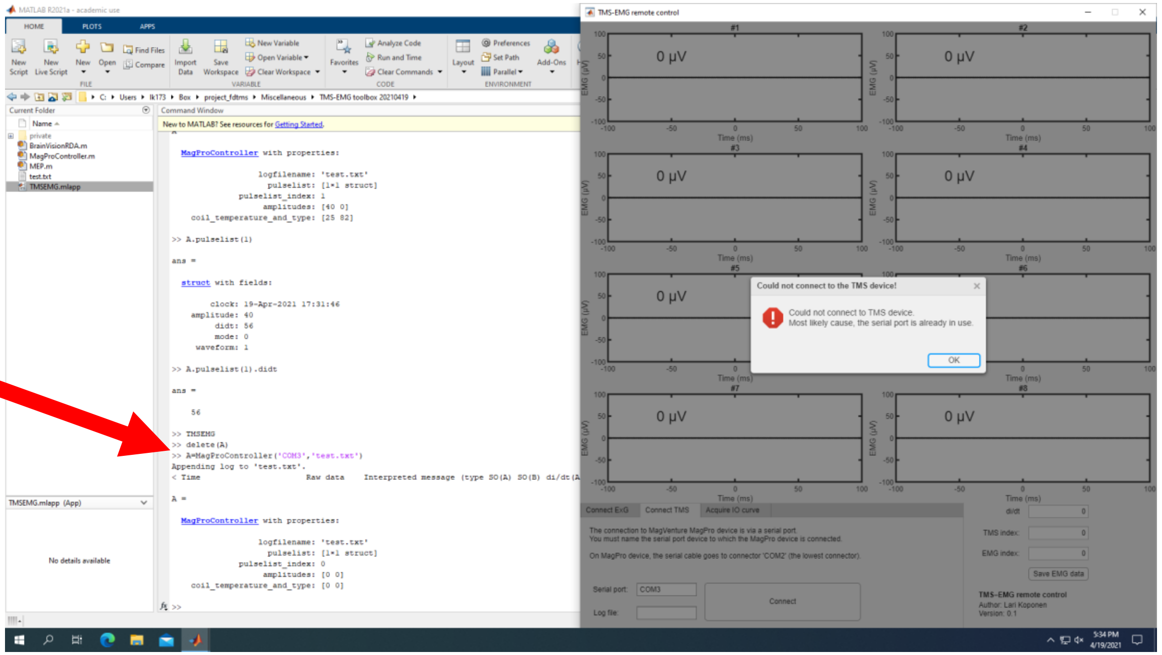


Save data?

Save EMG data before closing?

Save | Don't save | Cancel

TMS–EMG remote control
Author: Lari Koponen
Version: 0.1



```
  MagProController with properties:

                 logfilename: 'test.txt'
                   pulselist: [1×1 struct]
           pulselist_index: 1
                  amplitudes: [40 0]
    coil_temperature_and_type: [25 82]

>> A.pulselist(1)

ans =

  struct with fields:

        clock: 19-Apr-2021 17:31:46
    amplitude: 40
         didt: 56
         mode: 0
     waveform: 1

>> A.pulselist(1).didt

ans =

    56

>> TMSEMG
>> delete(A)
>> A=MagProController('COM3','test.txt')
Appending log to 'test.txt'.
< Time                     Raw data    Interpreted message (type SO(A) SO(B) di/dt(A

A =

  MagProController with properties:

                 logfilename: 'test.txt'
                   pulselist: [1×1 struct]
           pulselist_index: 0
                  amplitudes: [0 0]
    coil_temperature_and_type: [0 0]

fx >>
```

Could not connect to the TMS device!

Could not connect to TMS device.
Most likely cause, the serial port is already in use.

OK

The connection to MagVenture MagPro device is via a serial port.
You must name the serial port device to which the MagPro device is connected.

On MagPro device, the serial cable goes to connector 'COM2' (the lowest connector).

TMS–EMG remote control
Author: Lari Koponen
Version: 0.1

Output data formats

- EMG data file (.mat)
  - ‘description’ the date the data was saved
  - ‘data’, struct with cell arrays of
    - ‘TMS’ for each TMS pulse
      - ‘clock’ (datetime object)
      - ‘amplitude’ (% MSO)
      - ‘didt’ (MA/s)
      - ‘mode’ (0=standard, 1=power, 2=twin, 3=dual)
      - ‘waveform’ (0=monophasic, 1=biphasic, 2=halfsine, 3=biphasic burst)
    - ‘EMG’ for each acquired epoch
      - {‘clock’ ‘raw data [8×2751]’ ‘filtered data [8 8×2751]’ ‘classification [8×2]’}
        - Data, rows of FILTERED EMG data from –0.2 to 0.2 s
        - Classification rows of MEP amplitude and MEP classification (1=MEP, 0=no MEP, –1=rejected trial)
    - ‘IO’ for each started IO curve
      - {‘clock’ ‘pulse sequence [1×N]’ ‘TMS index at start’ ‘EMG index at start’}

- TMS device logfile (.txt)
  - Plain text of all data outputted by the TMS device
  - Event types 1-3, described in the MagPro user manual, have been decoded, the rest are just saved
  - Mostly for FYI, and for records
  - Contains every amplitude adjustment made on the TMS device front panel etc.
  - Data key provided in the class help of ‘MagProController’

Known limitations

- Neuronavigation is not coupled to the on-line loop
    - Cannot do “Nexstim-like” features, like ‘trigger-only-if-coil-is-within-tolerance-of-the-correct-target’
- Serial data from MagPro has undefined behavior
    - Dual/twin mode sends “randomly” one or two pulse messages
        - Case point: PP with 2.9 ms IPI sends RANDOMLY one or two messages
    - Solutions:
        - Extend BNC to PC, the triggers are deterministic
            - Via a parallel port (or via Arduino combining the two data streams into one…)
        - Make experiment specific assumptions of pulse parameters, and try to deconstruct what pulses have actually happened