

# Technical Note T1

## Batch Mode to Sequence Experiments

---



### 1. Introduction

Ivium Potentiostat/Galvanostats are controlled by IviumSoft Control and Data Analysis Software. There are three ways that IviumSoft can control your electrochemical experiment:

- 1) Direct control of the Potentiostat, i.e., setting a potential or current directly or using one of the special features such as the pulse generator or current interrupt.
- 2) Method control of the Potentiostat, i.e., using an electrochemical method (e.g., CV, EIS, etc.) with user programmable parameters.
- 3) Batch control of the Potentiostat, i.e., performing more complex experiments consisting of a sequence of electrochemical methods.

In IviumSoft, Batch control has the flexibility to configure sophisticated experimental sequences. The advanced design of Batch Mode makes this very easy. Because the electrochemical Methods are setup exactly the same way in both Method control and Batch control, using Batch control in IviumSoft is easy. You don't have to learn a new interface to use Batch Mode. The drop-down lists and built-in parameters take the guesswork out of setup.

Using Batch Mode, you can:

- Sequentially load and execute electrochemical Methods
- Create loops for repetitive experiments
- Edit a Method based on the previous experiments (e.g., increment the Scan Rate, adjust the Initial Potential, etc.)
- Automatically save the results
- Set time delays and program a schedule
- Switch the Multiplexer or MultiWE32 to fixed channels or change the channel for multiple samples
- Set the digital and analog outputs, for example send a trigger and/or control 3rd party equipment
- Wait for a HI or LO level on the digital input, i.e., receive a trigger

The Batch script is created and edited by the Batch Editor that is included in IviumSoft. Creating a new Batch script is fast and easy with the Batch Editor. The Batch script can be saved for later use.

### 2. Order of Communication

It is important to realize the order of communication between IviumSoft and the Ivium Potentiostat. When using Method control of the Ivium Potentiostat, the Method is first programmed in IviumSoft (PC) by choosing the desired parameters. Then the Method can be started by selecting the [Start] button. Following this, the Method is transferred to the microPC inside the Ivium Potentiostat and carried out from there, not from the PC<sup>1</sup>. The instrument will then send the data back to the PC.

In the same way, when including an electrochemical method in a Batch operation, the method is sent to the microPC and then executed. That is why an electrochemical method needs to be constructed (loaded) and subsequently executed in 2 different steps.

---

<sup>1</sup> For example: when a Method (or direct command) is carried out by an Ivium Potentiostat and during the experiment the IviumSoft program is closed (or the computer crashes), the Ivium Potentiostat will still continue the Method (or direct command).

### 3. The Batch Editor

The Batch Editor is activated by clicking the [BatchMode] button in the top bar of IviumSoft. This will open the Batch Tabsheet. Clicking the [BatchMode] button repeatedly will toggle between the Batch Tabsheet and the Method Tabsheet.

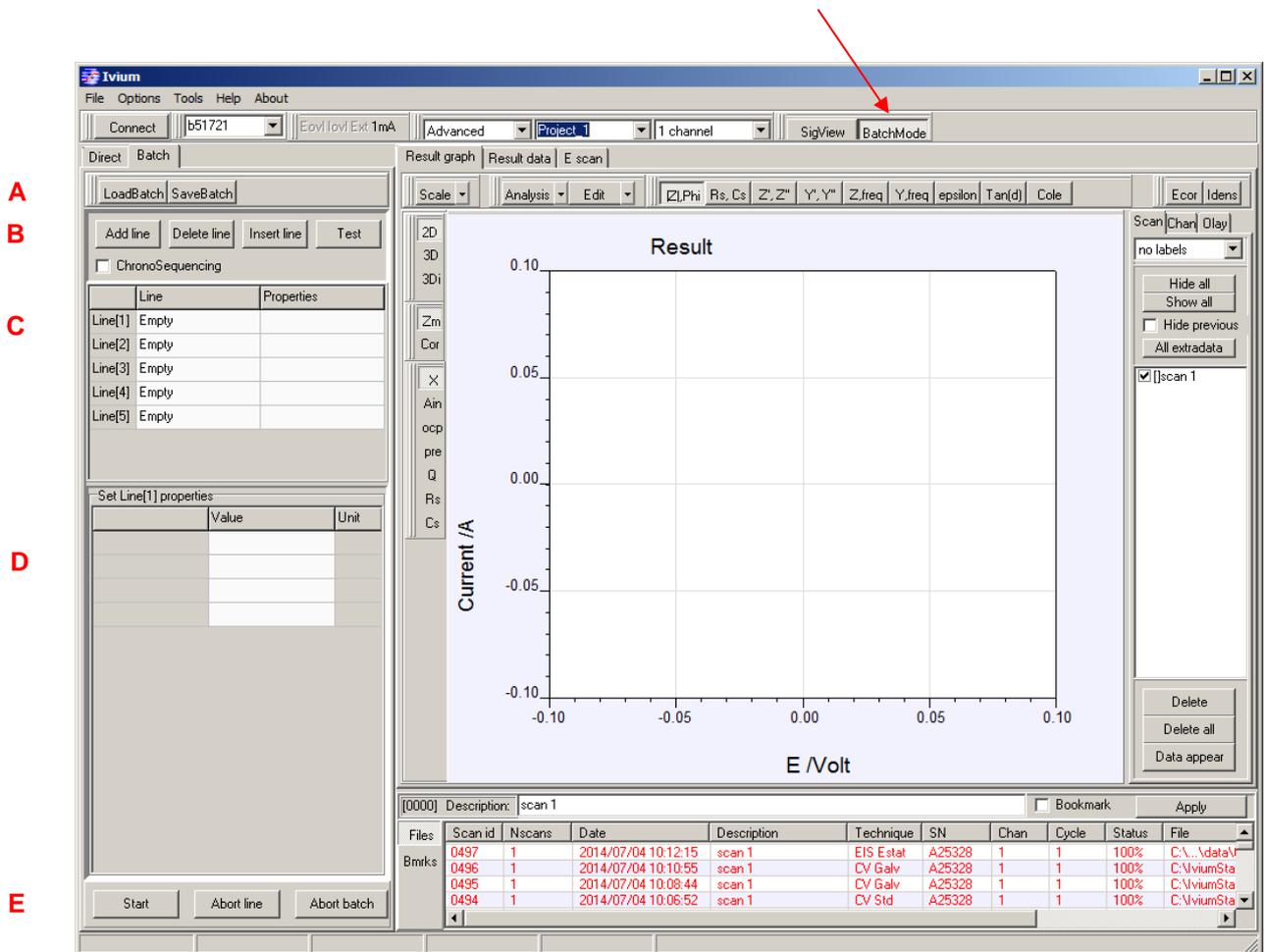


Figure 1. Opening the Batch Tabsheet in IviumSoft.

(A) At the top of the Batch Tabsheet are the Batch File controls to load and save Batch files (\*.ibf). Note that the active Project (Project\_1) is the default for the Batch files to be saved to and loaded from. Data stored during a Batch operation will also be stored in the active Project.

(B) Below the file-control buttons there are 4 buttons:

- [Add line]: Add a new line at the bottom of the script.
- [Delete line]: Delete currently selected line.
- [Insert line]: Insert a new line below the currently selected line.
- [Test]: Test the script for illegal Loop constructions and commands.

and a checkbox:

ChronoSequencing: When checked all Chrono (Transient) technique methods are displayed on a continuous time scale relative to the starting time of the batch. When multiple techniques are executed in sequence, their time variable accumulates.

*Note that if the "cleargraph" option is not used in batchmode, one can view all the Chrono results in one plot, in a chronological manner.*

(C) In the Batch Command Grid the Batch Script is created in a linear step-by-step fashion (in exactly the same way as you would do it manually).

There are seven commands in the Batch Command Grid:

|         | Line          | Properties |
|---------|---------------|------------|
| Line[1] | Empty         |            |
| Line[2] | Empty         |            |
| Line[3] | DefineMethod  |            |
| Line[4] | ExecuteMethod |            |
| Line[5] | Loop          |            |
|         | LoopEnd       |            |
|         | DirectCommand |            |
|         | EditMethod    |            |
|         | LoadMethod    |            |

1. *DefineMethod*: Will allow you to define a Method procedure inside the Batch Script. You can select the Technique and edit measurement parameters directly. After creation, the Method will be embedded in the Batch File, i.e., when you save the Batch file and e-mail it, the receiver will be able to run it with exactly the same Method parameters.
2. *Execute Method*: Runs the electrochemical method with the parameters as defined in the Method file.
3. *Loop*: Creates a loop for a repetitive action. The number of loops is unlimited and 32 levels of nested loops are allowed.
4. *LoopEnd*: Sets the end of a loop. Each Loop must be matched with a LoopEnd command.
5. *DirectCommand*: 7 commands will allow measurement scheduling, setting the channel of the (Hi)Mux or MultiWE32, setting DAC and digital I/O values, and/or setting a period of waiting.
6. *EditMethod*: Used to modify one parameter of the electrochemical Method in active use. Examples of the application of EditMethod include incrementing the scan rate in a cyclic voltammetry experiment or the DC voltage in an EIS measurement.
7. *Load Method*: Loads an electrochemical method that has previously been set up and saved. When this command is selected, an extra field appears below the Batch file controls (A). The "Method folder" allows locating and selection of the folder where the stored method is located.

(D) When a command in the Batch Command Grid is selected, its Properties can be specified in the bottom Batch Parameter Grid (Line Properties).

(E) Below the Batch Parameter Grid there are 3 buttons that operate the batch:

- [Start]: Execute the Batch Script. If the Script contains errors, it will not be executed.
- [AbortLine]: Abort the action on the line that is currently being executed and proceed to the next line.
- [AbortBatch]: Abort the action on the line that is currently being executed and stop the Batch script.

**A summary of all the available commands for Batch Mode is shown in Annex 1.**

## 4. Data and Batch file location

Batch files are *NOT* saved automatically, but must be saved manually by the user. The active Project is suggested by default for Batch files to be saved to, and loaded from. Data files stored during a Batch operation will be automatically stored in the active Project.

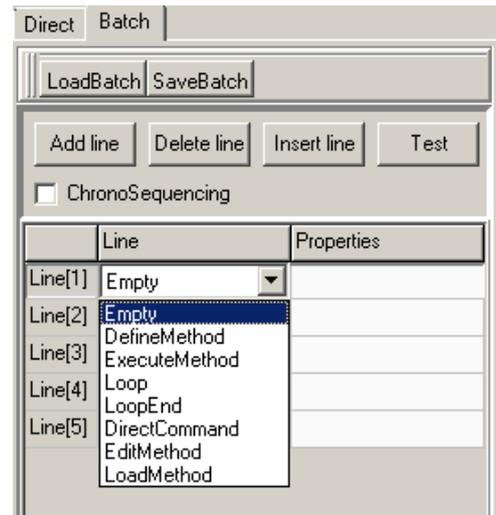
## 5. Examples

### 5.1. Batch Command Grid

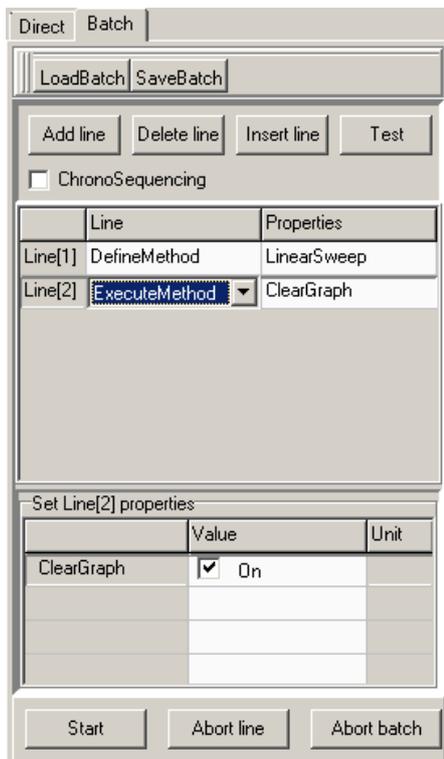
To create a Batch script, click on the first "Empty" Line field to open a drop-down list with the available commands (C).

There are only 7 commands in the Batch Command Grid:

1. DefineMethod: Allows you to define a method
2. Execute Method: Runs the electrochemical method with the instruments parameters as defined in the Method file.
3. Loop: Creates a loop for a repetitive action.
4. LoopEnd: Sets the end of a loop
5. DirectCommand: 7 direct controls are available used for scheduling, setting the channel of the HiMux, controlling I/O, etc.
6. EditMethod: Used to modify one parameter of the electrochemical Method in use.
7. Load Method: Loads an electrochemical method previously setup and saved.



When a command in the Batch Command Grid is selected, its Properties can be specified in (D) the bottom Batch Parameter Grid (Line Properties).



Example: Setting the "Execute Method" parameters (Properties) in the Batch script.

After the electrochemical method has been selected and its parameters defined, the method can be executed. This requires the separate ExecuteMethod command.

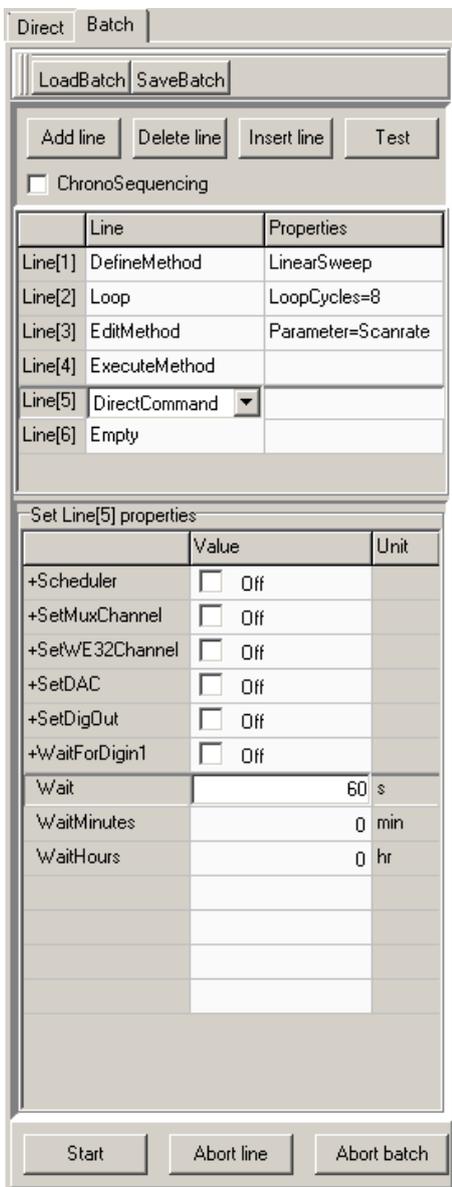
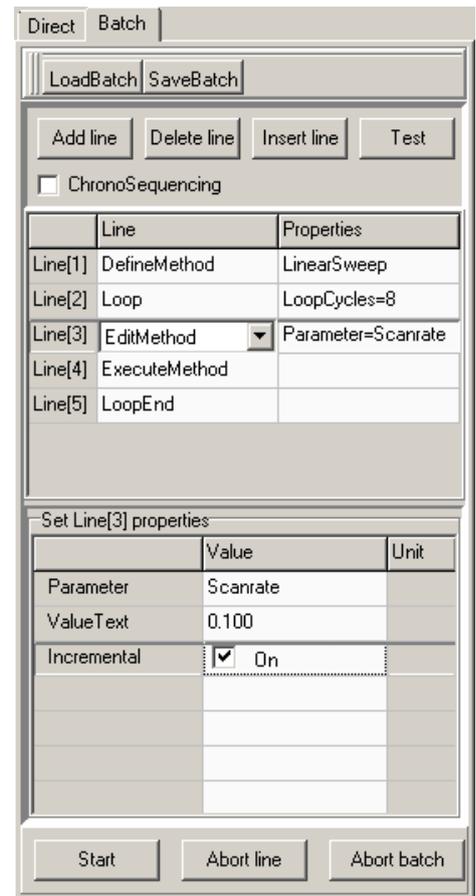
Data generated during the execution of an electrochemical method is Checking "ClearGraph" will clear the view window (it will not delete data). If left unchecked, all data/consecutive scans will appear in the viewing window. However, only data of the same format and appearance as the active scan will be visible.

Example: Setting the "Edit Method" parameters (Properties) in the Batch script.

Line 1: Define method  
 Line 2: start Loop (8 cycles)  
 Line 3: Edit method:

In this case, the parameter to be edited is the Scan Rate. In each loop, the Scan Rate is incremented by 100 mV/sec (0.1 V/sec).

Line 4: Execute method with incremented scan rate  
 Line 5: End Loop



Example: Setting the "Direct Command" parameters (Properties) in the Batch script. All of the "Direct Command" parameters are visible in the Batch Parameter Grid, so there's no need to remember a custom technique or language to quickly setup your Batch Script.

In Line 5 of the Batch, inside the Loop, a Wait time of 60 seconds is specified in this case.

## 5.2. Creating a Sequence Using Batch Mode

Create a sequence that will carry out a CV experiment, then an impedance measurement, followed by a 1 hour waiting period, on all 8 channels of a multiplexer consecutively.

Step 1: Activate the Batch Editor.

Step 2: Create Batch routine in the logical order that you would expect

| Line     | Line          | Properties   |
|----------|---------------|--|
| Line[1]: | Loop          | Loopcycles = 8; SetMuxToIndex = check  |
| Line[2]: | DefineMethod  | Method=CyclicVoltammetry; set all appropriate method parameters.<br><i>Note that "Basic/Advanced" in the top menu bar will also allow more or less Method parameters, just like in the Method sheet.</i> |
| Line[3]: | ExecuteMethod | ClearGraph = check; to clear graph prior to executing each Method {optional, since only data with same format as from the active Method is shown automatically}  |
| Line[4]: | DefineMethod  | Method=Impedance; set all appropriate method parameters.   |
| Line[5]: | ExecuteMethod | ClearGraph = check   |
| Line[6]: | DirectCommand | Wait = 1 hr  |
| Line[7]: | LoopEnd       |  |

Step 3: Check the script for errors by clicking "Test". If no errors are detected, then nothing happens. If an error is detected, a pop-up will appear showing the error

Step 4: Start the Batch script by selecting the Start button.

The final Batch script looks like this;

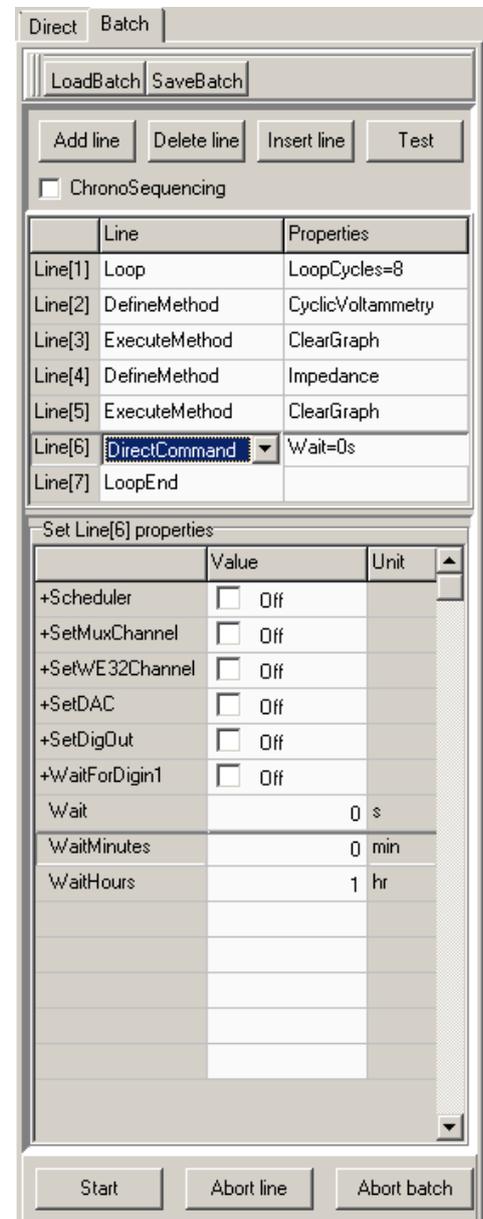
### NOTE:

*In the IviumSoft result graph window only data is displayed that is compliant with the active plot. So, if like in this example an EIS is carried out after a CV scan, the CV data will be hidden from view when the EIS scan starts. But the CV data is not deleted, it is just hidden from view.*

*When the next CV in the following loop is carried out, the result graph window will show all CV data again, and it will hide the EIS data, etc.*

*The result graph window always follows the active scan.*

*The same is valid for data loaded from memory: only data compliant with the active scan in memory is shown. When multiple scans are loaded, containing data recorded with different techniques, the hidden ones can be viewed by mouseclick, while keeping CTRL pressed. This will make the clicked scan active, thus redefining the plot "mode".*



### 5.3. Creating a Scheduled Sequenced Impedance Measurement

To run an impedance test exactly once every x minutes automatically, you need to create a Batch file that uses the Scheduler function.

The tricky part, however, is that you will not know how long the impedance test will take exactly, depending on the times of oversampling necessary, etc. To be sure and be able to calculate the length of time needed for the impedance measurement exactly, you would have to in the instrument options allow only 1 time oversampling/1 gain change, etc.

Alternatively, you will need to allow enough time to elapse to make sure the impedance measurement is completed.

With this length of time you need to create a Batch file that includes the scheduler. Assuming your impedance test will take ca. 10 minutes maximum:

Step 1: Activate the Batch Editor.

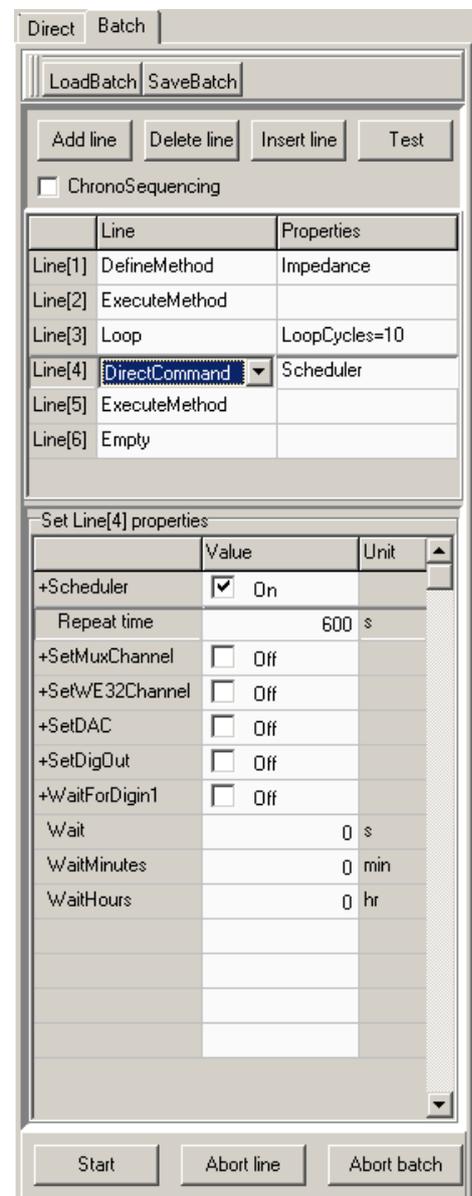
Step 2: Create Batch routine in the logical order that you would expect

| Line            | Line          | Properties  |
|-----------------|---------------|---|
| <b>Line[1]:</b> | DefineMethod  | Method=Impedance;<br>set all appropriate method parameters.   |
| <b>Line[2]:</b> | ExecuteMethod | ClearGraph = check<br>{for the first run of the impedance measurement}  |
| <b>Line[3]:</b> | Loop          | Loopcycles = 10;<br>{assuming you want 10 scheduled impedance measurements}   |
| <b>Line[4]:</b> | DirectCommand | Scheduler = check;<br>Repeat time = 600s<br>{make sure the 600s interval is enough to finish the previous measurement, if not and the instrument is still busy, the routine will crash} |
| <b>Line[5]:</b> | ExecuteMethod | ClearGraph = uncheck<br>{this will show all measurements in the view window simultaneously}   |
| <b>Line[6]:</b> | LoopEnd       |   |

Step 3: Check the script for errors by clicking "Test". If no errors are detected, then nothing happens. If an error is detected, a pop-up will appear showing the error

Step 4: Start the Batch script by selecting the Start button.

The final Batch script looks like this;



## Annex 1: List of Batch Commands

| Command              | Parameters  | Description   |
|----------------------|---|---|
| <b>DefineMethod</b>  | <b>Define a Method procedure inside the BatchScript.</b><br>In the "Set Line[]" properties window, the Method group can be selected from the "Method" drop down list, the Technique can be selected from the "Technique" drop down list.  |   |
|                      | Load  | Will load (import) the parameters from an existing method file on disk. These are loaded when you click this button, and embedded in the active BatchFile. If the original source file would be removed/changed afterwards, it would not affect the embedded Method definitions.  |
|                      | Save  | Will save (export) the method parameters to a method file "*.imf".  |
|                      | Get Active  | Will copy the parameters from the active Method, as it is displayed on the main Method tabsheet.  |
|                      | Set Active  | Will export the DefineMethod parameters to the active Method, and will be displayed on the main Method tabsheet, replacing the previous active Method.  |
|                      | It is important to note, that while a BatchScript is being edited, the DefineMethod parameters can be different from the active Method settings on the MethodSheet. Also note that a BatchScript may contain several different Method definitions. However, when a Batch is run and a DefineMethod line is executed, the active Method parameters will be set according to the embedded Method definition on the active line. |   |
| <b>ExecuteMethod</b> | <b>Execute defined/loaded Method.</b>   |   |
|                      | ClearGraph  | Clear plot before the next scan starts. This can be useful when different data are plotted, i.e., CV and impedance (Note that only data with the same format can be displayed at the same time).  |
|                      |   | Data is automatically saved to file, using an automatically generated file name.  |
| <b>Loop</b>          | <b>Create a repetitive loop.</b><br>The number of loops is unlimited. Nesting is allowed up to 32 levels deep. Each loop must be matched with a LoopEnd command.  |   |
|                      | LoopCycles  | Number of repetitions   |
|                      | SetMuxToIndex   | Switch the multiplexer channel to the loop-index.   |
|                      | SetWE32ToIndex  | Switch the MultiWE32 channel to the loop-index.   |
| <b>LoopEnd</b>       | <b>Set the end of the loop.</b>   |   |
| <b>DirectCommand</b> | <b>Execute a direct command.</b><br>Used for scheduling, setting the channel of the HiMux, setting an analog I/O, etc.  |   |
|                      | Scheduler   | Events can be scheduled: the Repeat time-timer keeps track of total time lapsed since the Batch was started.<br>Events can be scheduled at a fixed time, independent of the durations of previous executed scans. Thus measurements can be repeated exactly every minute, hour, etc.<br>Note the difference with the "DirectCommand.Wait" function, that adds a delay. To activate, check Scheduler, and set Repeat time variable.<br>Scheduler will take precedence over other DirectCommands in the same Batchline: they will be set only after the Repeat timer period has passed. |

|                   |  |  |
|-------------------|--|--|
|                   | Scheduler.Repeat time  | Repeat period can be set:<br>1 sec to 1E7 s (> 3 months)   |
|                   | SetMuxChannel  | If checked, the channel is switched to the selected channel no#  |
|                   | SetMuxChannel.channel  | Channel no# to be set  |
|                   | SetWE32Channel   | If checked, the channel is switched to the selected channel no#  |
|                   | SetWE32Channel.channel   | Channel no# to be set  |
|                   | SetDAC   | If checked, the analog outputs are set   |
|                   | SetDAC.DAC 1   | New value for An_out 1   |
|                   | SetDAC.DAC 2   | New value for An_out 2   |
|                   | SetDigOut  | If checked, the DigOuts are set  |
|                   | SetDigOut.DigOut 1   | On if checked (HI default, 0 if checked)   |
|                   | SetDigOut.DigOut 2   | On if checked (HI default, 0 if checked)   |
|                   | SetDigOut.DigOut 3   | On if checked (HI default, 0 if checked)   |
|                   | WaitForDigIn1  | If checked, the execution is halted until the digital input 1 reaches the desired level, or timeout is exceeded.   |
|                   | WaitForDigIn1.WaitForHi  | If checked, the system waits until a HI level is applied. If unchecked, it waits for a LO level.   |
|                   | WaitForDigIn1.TimeOut  | Maximum period that is waited for the selected level on diginput 1.  |
|                   | Wait   | Program a delay [seconds]  |
|                   | WaitMinutes  | Program a delay [minutes]  |
|                   | WaitHours  | Program a delay [hours]  |
| <b>EditMethod</b> | <b>Edit a single parameter of defined/loaded Method.</b>   |  |
|                   | Parameter  | Textual expression of the parameter-name; exact and case sensitive. For a primary parameters type the exact name; for a secondary parameter type the primary parameter name followed by a "." and then the secondary parameter.  |
|                   | ValueText  | Textual expression of the parameter value. The format of the supplied value must correspond with the type of the selected parameter name. If the selected parameter is a checkbox, a value of 'true' will correspond to the checked condition, anything else will uncheck the box. Numerical text strings must be of the correct format. |
|                   | Incremental  | If unchecked (default), the new value will replace the existing one. If checked, the new value will be added to the existing parameter value. This option is only valid for parameters that accept numerical data.   |
| <b>LoadMethod</b> | <b>Load Method file from disk</b><br>This command is not recommended. Instead, create a new Method using the "DefineMethod" command to avoid compatibility issues. |  |
|                   | Methodfile   | Use the ".." prompt box to open a pop-up window to navigate to the method file to be loaded.   |
|                   | Method folder  | In the field that appears below the Batch file commands when LoadMethod is selected, select the appropriate folder where the method file to be loaded is located.  |