

# TCP Communication with NG - BioPro Software

Written by J.S.McCaskill,  
Ruhr University Bochum

using the Mathlink Interface Software connected to a customized C interface communicating by TCP with Uwe Tangen's NG-BioPro software written in C.

After making any changes, the saved TCPComm.m file needs to be placed in the userdir/Library/Mathematica/Applications folder, to be found in application use.

Users should install in *Mathematica* the palettes created by the command NGPalette[] and MAPalette[] in order to have assisted input of key interface and custom *Mathematica* commands.

An example application of this package is found in the notebook TCPCommEG.nb

---

## Command definitions

### ■ Version History

Ver 1.0	10.08.07 Basic version of interface commands
Ver 1.1	10.04.08 Implement timed loops
Ver 1.2	16.04.08 Implement verbose mode and heirarchical labelling in nested loop
Ver 1.2.1	25.04.08 Correct problem with raster image display, address time discrepancy. Subimage display.
Ver 1.2.2	17.06.08 Table raster
Ver 1.2.3	30.06.08 Palette for NG commands. NG with autospace.
Ver 1.2.4	4.07.08 Correct NG command tooltip
Ver 1.2.5	21.07.08 Correct MA command for tooltip extraction and the imwrite function for lists of image numbers
Ver 1.2.6	4.8.08 Accelerated image conversion and display

## ■ Command Palettes

To install these palettes, which are also found in the example notebook TCPCommEG.nb, select them and using the Palettes Menu

Generate Selection from Palette, and then install palette, with the names as shown on the tags below.

The palettes are equipped with argument tool tips, to advise on usage.

NB: Only edit the arguments, one at a time, to avoid corrupting the templates.

NG Commands

	Sess open	Sess close	Quit	Status	BackOff	BackOn
	Synch	Window	Cam par	Snap	AOTF	Filter
Out[80]=	Pump par	Pump	List pins	Electrode	Cycle	List sensors
	Sensor data	Intensity	List names	Temp	XY move	XY def
	ZStage	Verbose				

MA Commands

Timed Loop	Set ROI	I Snap	I Convert	I Display	I Dispsub
I Color	I Write	I Read	I Size	I RowCol	I Line
I Bin	I ROI	I Comb	NG Start	NG Exit	

## ■ Package initialization and exported functions

Help on the package commands below can be obtained using the mathematica statements in the form ?command (e.g. ?NG)

```
In[1]:= BeginPackage["TCPComm`"];
```

```

In[2]:= NG::usage = "converts a list of arguments to a
      string and sends it to the communication channel via Mathlink";
NGPalette::usage = "creates a palette for user friendly input
      of NG communication commands (no arguments)";
NGCmd::usage = "Low level for NG interface: arguments [s_String]";
NGCmds::usage = "List of commands for NG interface with syntax";
NGCommune::usage =
  "Call mathlink to communicate with biopro via TCP: arguments [8086,machinename]";
NGlink::usage = "Name of Mathlink link for biopro";
NGInstall::usage = "install TCP-mathlink communication
      to biopro machine dyck and report status : no arguments";
NGClose::usage = "Call mathlink to closedown TCP link: no arguments []";
NGExit::usage = "shutdown TCP-mathlink communication: no arguments";
□::usage = "placeholder for palette entry with tooltips";
untool::usage = "strip Tooltips from parameter list : arguments [x_]";
MA::usage = "strip tooltip mathematica command generated from MA palette";
MAPalette::usage = "creates a palette for user
      friendly input of some custom Mathematica commands (no arguments)";
MACmds::usage = "List of commands for MA custom palette use with syntax";
status::usage = "interpret machine status code string";
imconvert::usage = "convert machine read string image to MA format {array,comments}";
TimedLoop::usage =
  "timed loop macro with arguments [name_String,body_,intvlsec_,niter_Integer]";
imwrite::usage =
  "write an image to disk : arguments [im_List,name_String,num_,comment_String]";
imread::usage = "read an image from disk : arguments [filename_String]";
imsize::usage =
  "size of image: arguments [imagearray_?ArrayQ] or [{imagearray_?ArrayQ,comments}]";
imrowcol::usage =
  "extract given row and column from image: arguments [im_?ArrayQ, {x_, y_}]";
imrowdiag::usage = "diagonal intensity row: arguments [im_?ArrayQ,{{x1_,y1_},{x2_,y2_}}]
      or [im_?ArrayQ, {{x1_, y1_}, {x2_, y2_}}, w_]";
imbin::usage = "bin image either square or rectangular: arguments
      [im_?ArrayQ, xb_Integer] or [im_?ArrayQ, {xb_Integer, yb_Integer}]";
imsetROI::usage = "set default image coords for ROI: arguments
      [x_Integer, y_Integer, xm1_Integer, ym1_Integer]";
imextract::usage = "extract ROI subimage: arguments [im_?ArrayQ, {x0_, y0_, xm_, ym_}]";
imcombine::usage =
  "combine two images to multi-color image: arguments [im1_?ArrayQ,im2_?ArrayQ]";
imshow::usage = "show image: 1-4 arguments
      [im_?(ArrayQ[#,2]&),namedcolor_RGBColor,maxim_Integer:0,imROI_List:{0,0,0,0}];
imROI::usage = "";
imcsuROI::usage = "";
verbose::usage = "toggles value of verbose output: either True or False";

In[32]:= Begin["`Private`"];

```

- NG Interface and Templates
- Interface interpretation
- Macro support
- Image processing
- Templates for Mathematica command input, via "MA Command" palette
- Finish Package

```
In[76]:= End[];
```

```
In[77]:= EndPackage[];
```