



## PRECISION AND VACUUM TECHNOLOGY

Aleksander Stanik  
Chief Software Architect

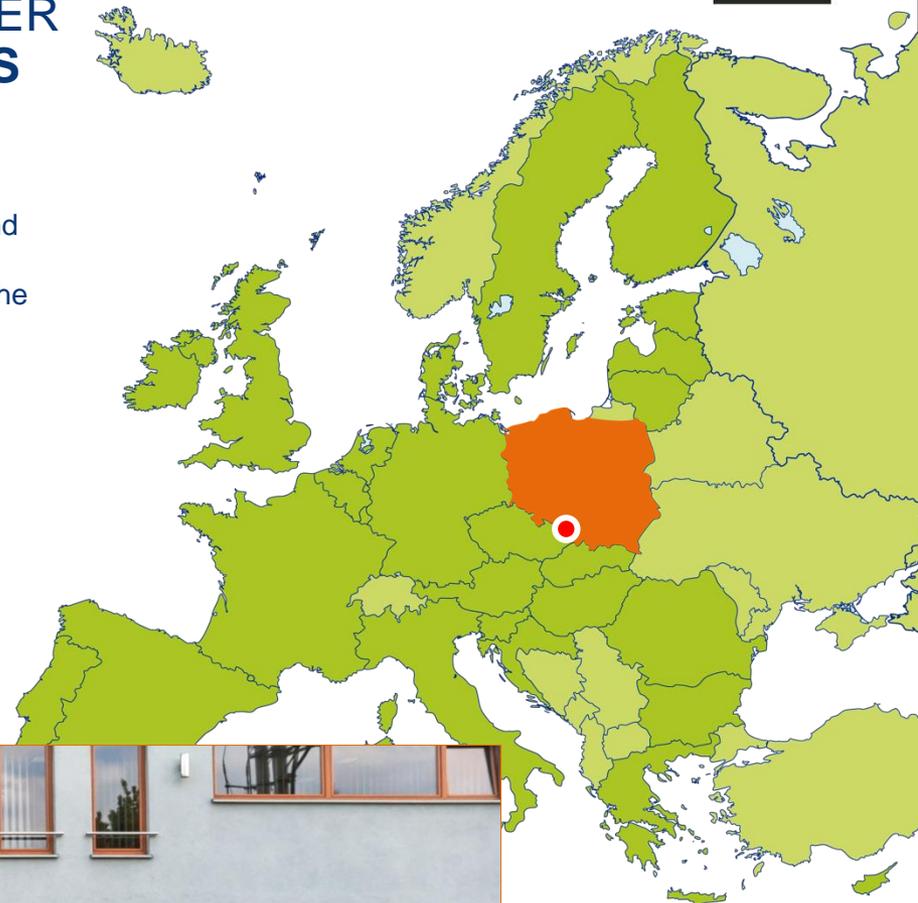
HV & UHV SYSTEMS ■ MANIPULATORS ■ CHAMBERS & MECHANISMS ◉ SAMPLE HOLDERS ■ INSTRUMENTS ■ ACCESSORIES ■ ELECTRONICS ■ SOFTWARE ■ SERVICES

# PREVAC THE MAIN PRODUCER OF UHV SYSTEMS

PREVAC was founded in 1996 in Rogów, Upper Silesia, Poland.

PREVAC can be distinguished by its highly skilled, young, dynamic and ambitious personnel consisting of the best specialists who along with the long time experience in the field of vacuum technology constitute the greatest potential of the company.

PREVAC employs a highly qualified staff of over 200 professionals – R&D constructors, manufacturing engineers, software developers etc., who works in various departments.



# SALES

Since its foundation PREVAC has been an international known leading manufacturer of scientific-research equipment used for studies under conditions of high and ultra-high vacuum.

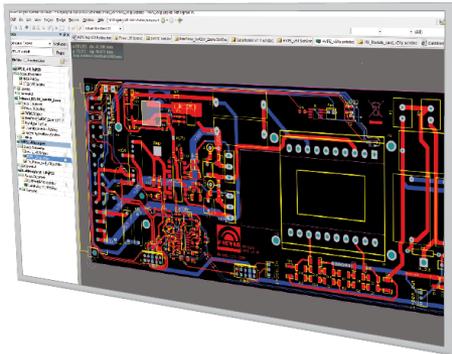
Products made by the company operate presently in such countries as Germany, France, Great Britain, Sweden, Norway, Italy, Spain, the Czech Republic, the USA, Canada, Japan, China, Russia, Australia, the Republic of South Africa, Poland etc.



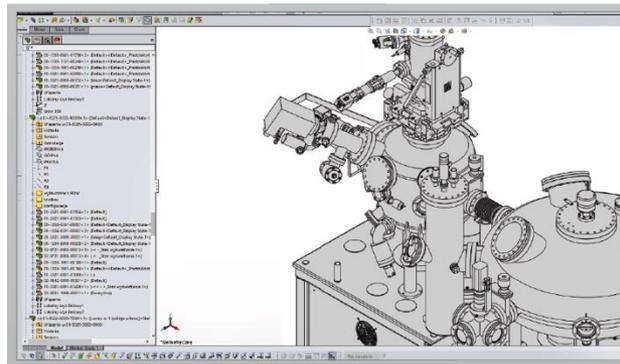
# YOUR PROJECT



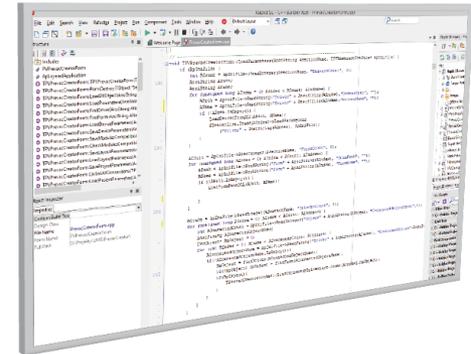
ELECTRONIC DEVICES



MECHANICAL DESIGNING



SOFTWARE APPLICATIONS



OUR PROJECT TEAM WORKS **FOR YOU**

# MULTITECHNIQUE UHV



**SISSY@EMIL**  
Helmholtz Zentrum Berlin

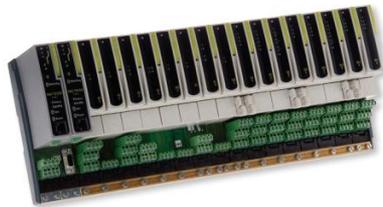
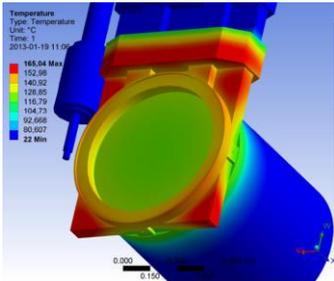
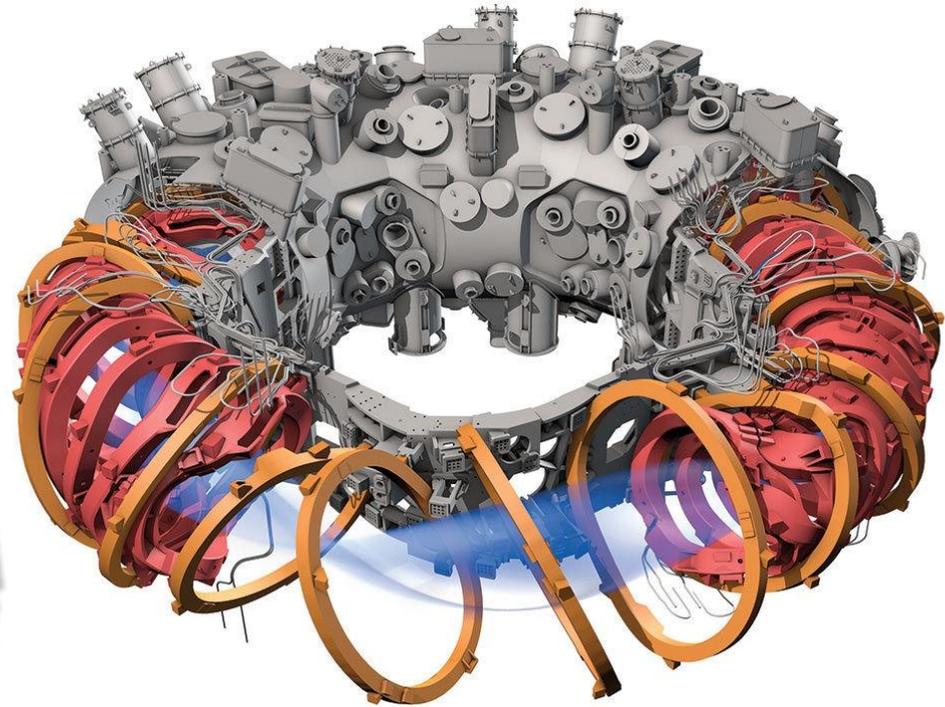
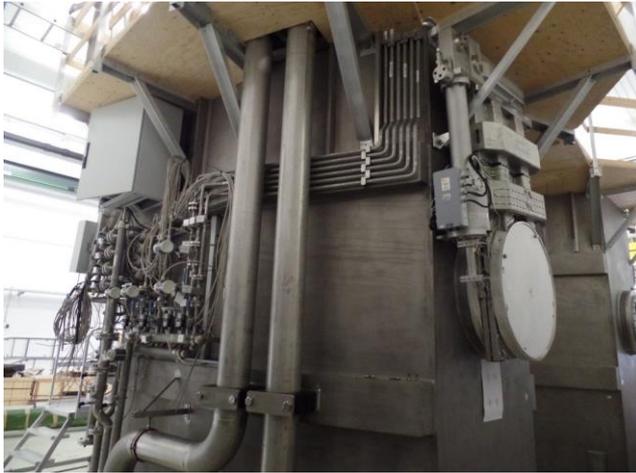
# END STATIONS



## BEAMLINE ENDSTATION

?? Somewhere ☺

# LARGE SCALE PROJECTS



## WENDELSTEIN 7-X

Helmholtz Zentrum Berlin

# NEW COMPONENTS



**EA15**  
R&D PREVAC

# LAB CUSTOMERS NEEDS



- Focused on experiment
- Extend the experiment with smallest possible cost
- Access to experiments data

# LAB CUSTOMERS REALITY



- Equipment from one or many suppliers
- Every supplier/integrator has own custom solutions
- Manage experiment is not easy
- Extend the experiment cost a lot

# WHATS WRONG ?



THERE IS NO STANDARD !

# STANDARD LAB CS

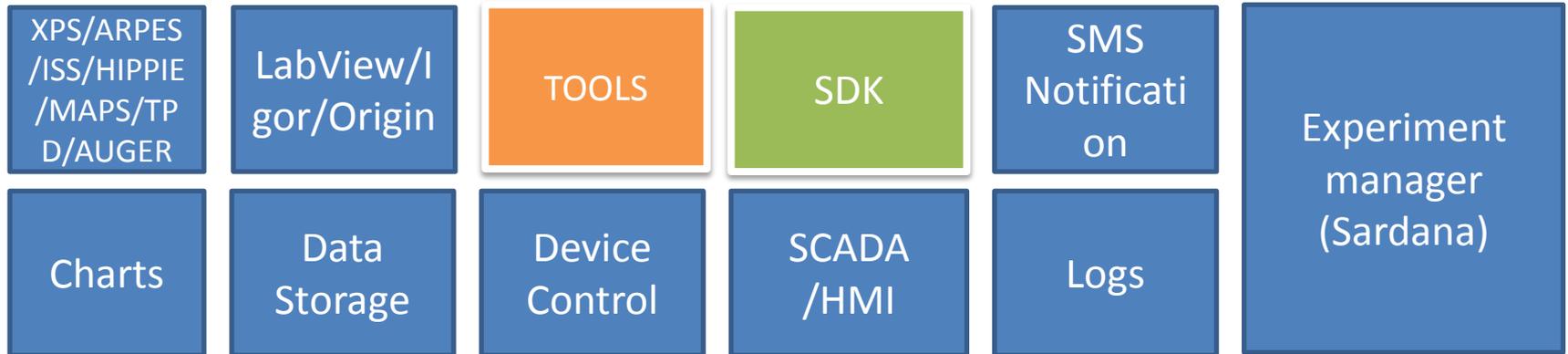


**BRING TANGO TO LAB !**

# TANGO IN LAB = RAPID FX

- 3 Tango base installations in 2016
- >20 installations in 2017
- and more...

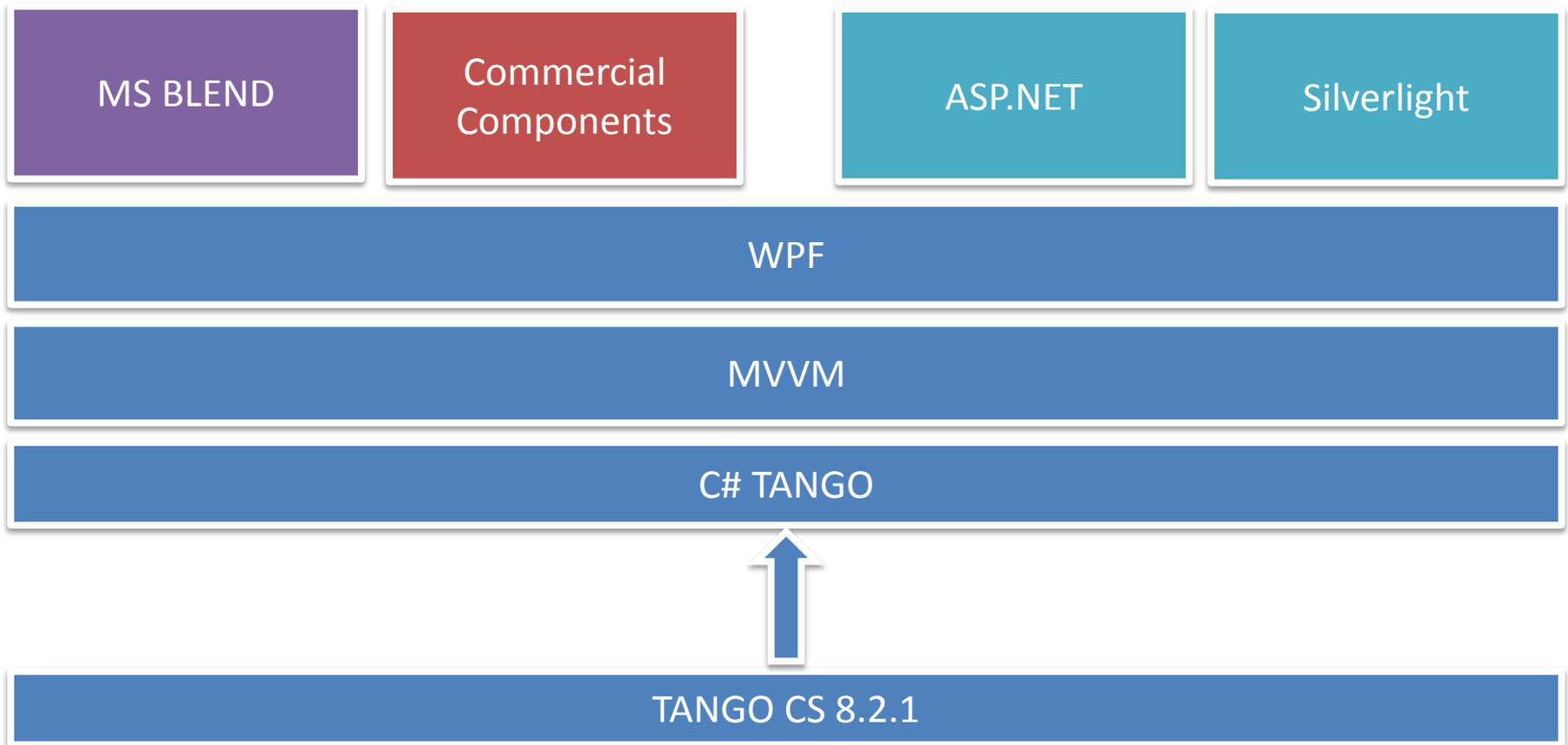
# RAPID FX PLATFORM



TANGO Framework – Python – C++/boost



# RAPID FX PLATFORM



# RAPID FX SYSTEM VIEW

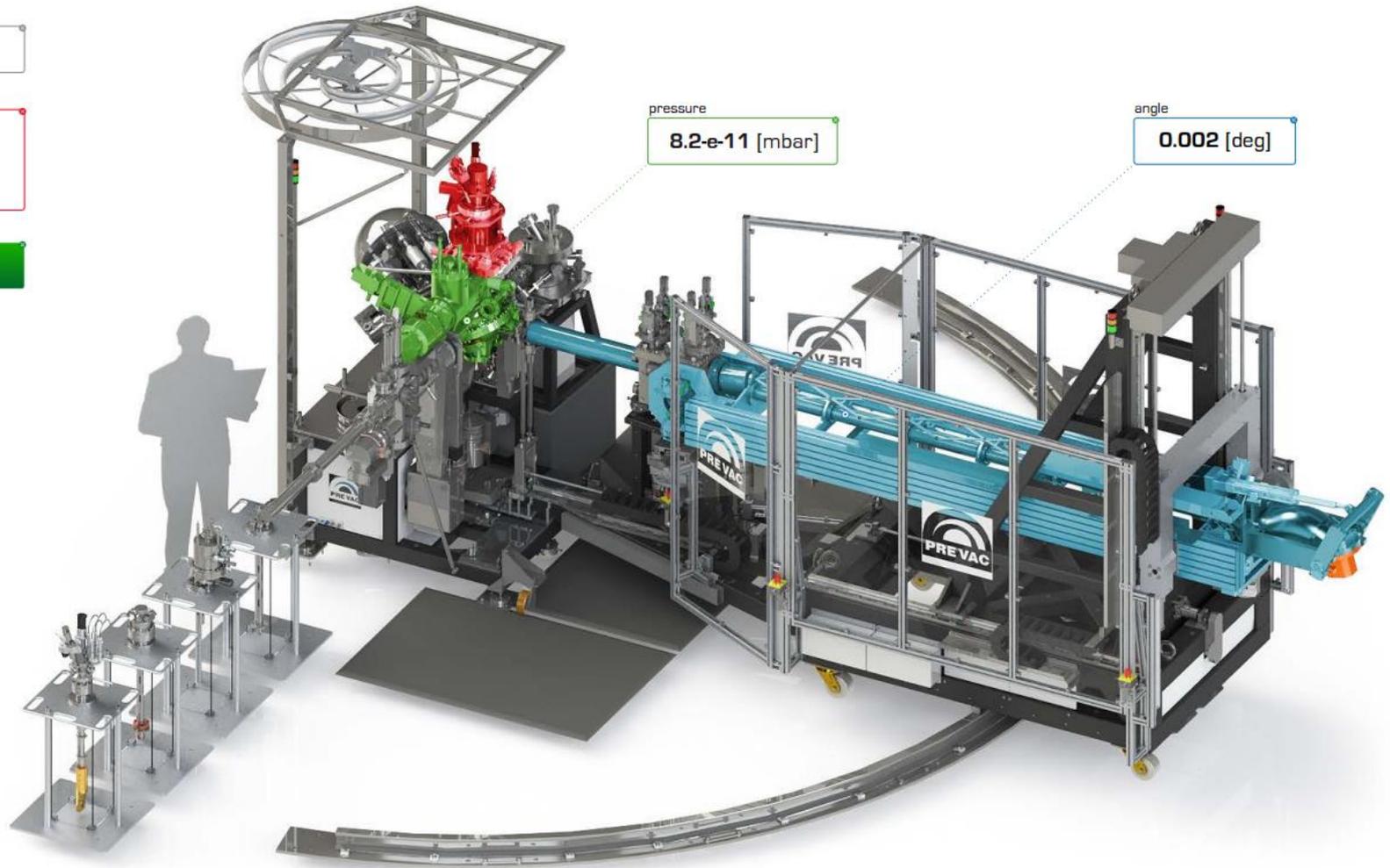
temperature

237.1 [°C]

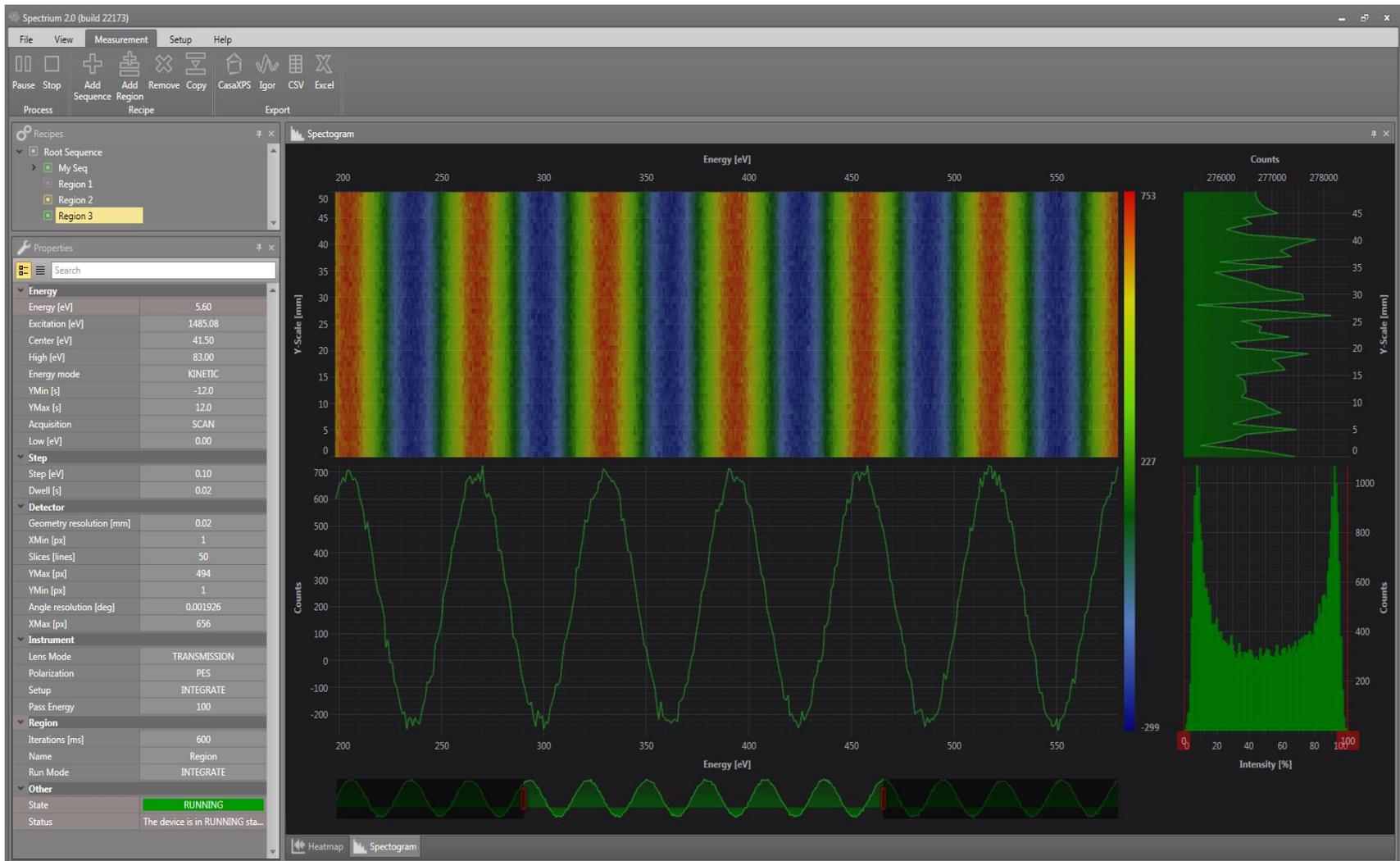
manipulator

X=0.0 [mm]  
Y=0.0 [mm]  
Z=10.0 [mm]

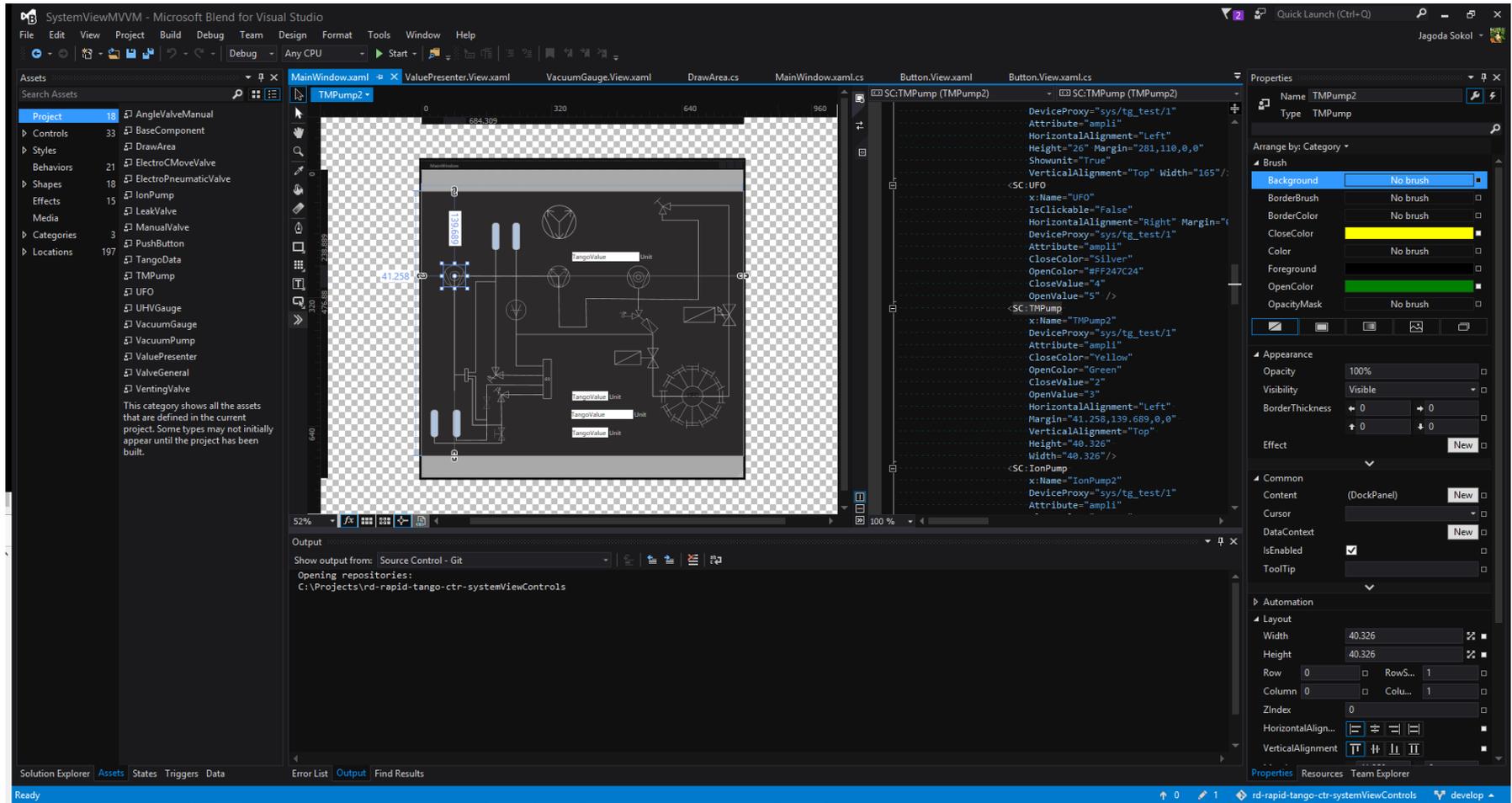
System ready



# RAPID FX SPECTRIUM

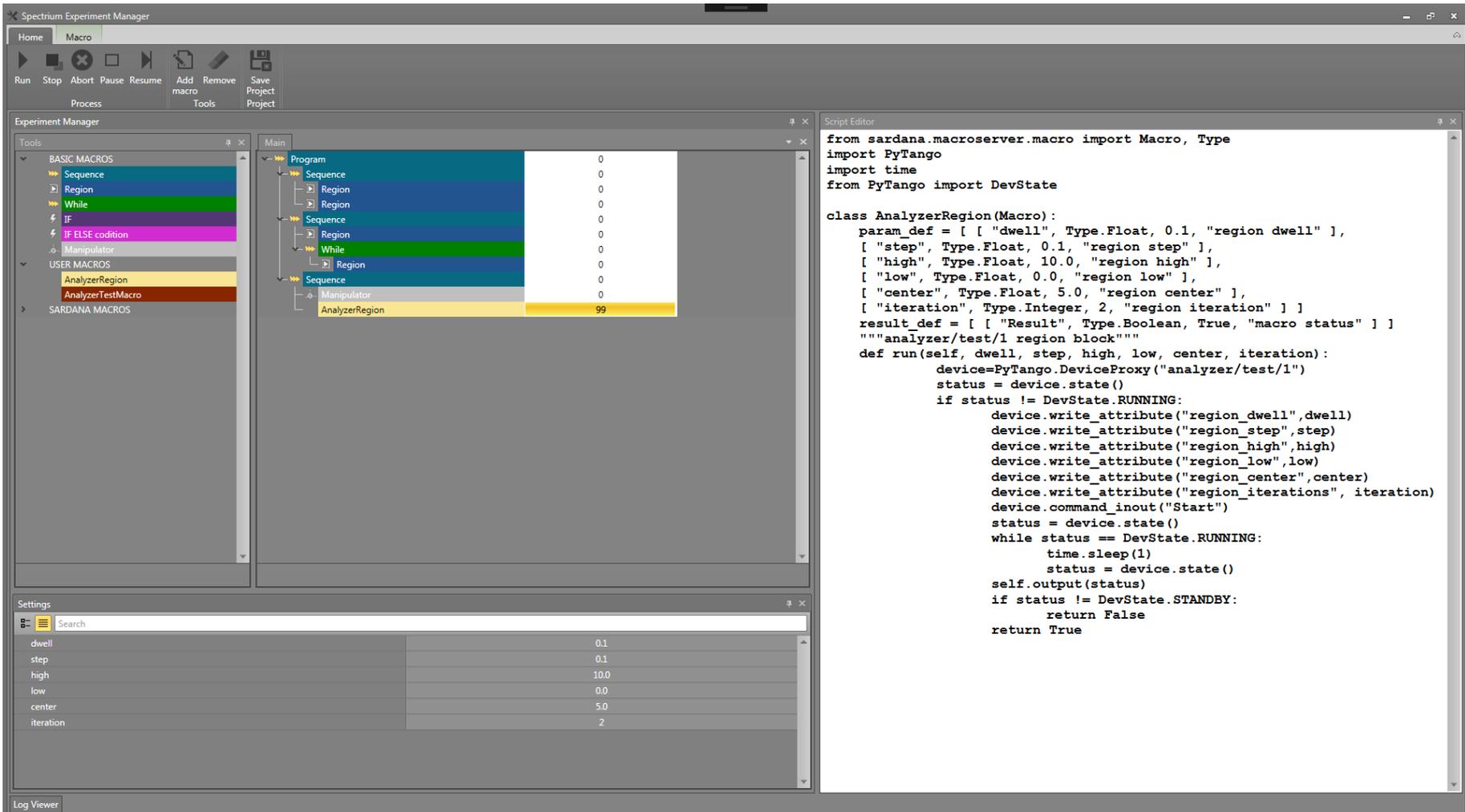


# RAPID FX SCADA IN BLEND



The screenshot displays the Microsoft Blend for Visual Studio interface. The central workspace shows a SCADA control design for a 'TMPump2' component, featuring a dark background with white and yellow graphical elements representing pumps, valves, and piping. The left-hand 'Assets' pane lists various UI controls and behaviors available for reuse. The right-hand 'Properties' pane is configured for the selected 'TMPump2' control, showing its XAML-based styling and layout properties. The bottom status bar indicates the current project is 'rd-rapid-tango-ctr-systemViewControls' in 'develop' mode.

# RAPID FX XMANAGER



The screenshot displays the Spectrum Experiment Manager interface. The top menu bar includes 'Home' and 'Macro'. Below it is a toolbar with icons for Run, Stop, Abort, Pause, Resume, Add macro, Remove macro, Tools, and Save Project. The main workspace is divided into three panes:

- Experiment Manager (Left):** A tree view showing macro components. Under 'BASIC MACROS', there are 'Sequence', 'Region', and 'While'. Under 'USER MACROS', there is 'AnalyzerRegion' and 'AnalyzerTestMacro'. Under 'SARDANA MACROS', there is 'Manipulator'.
- Main (Center):** A detailed view of the selected 'AnalyzerRegion' macro, showing its internal structure: 'Program' (0), 'Sequence' (0), 'Region' (0), 'While' (0), 'Region' (0), 'Sequence' (0), 'Region' (0), 'While' (0), 'Region' (0), 'Manipulator' (0), and 'AnalyzerRegion' (99).
- Script Editor (Right):** A Python script defining the 'AnalyzerRegion' macro class. The script includes imports for 'Macro', 'PyTango', and 'DevState', and defines a 'run' method that configures device parameters and performs a loop of operations.

At the bottom, a 'Settings' pane contains a search bar and a table of parameters:

Parameter	Value
dwell	0.1
step	0.1
high	10.0
low	0.0
center	5.0
iteration	2



THANK YOU FOR YOUR ATTENTION