

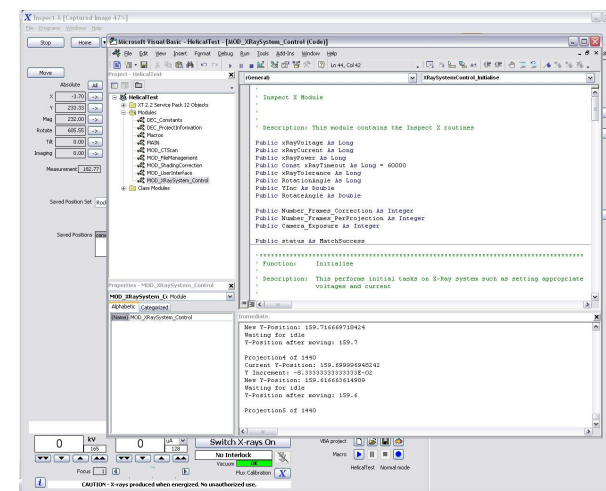
Parmesh Gajjar^{1,2}, Sam Mcdonald¹, Jakob S. Jørgensen³, Julia Behnsen¹, Chris Johnson², Nico Gray², William Lionheart², Martin Turner¹ and Philip J. Withers¹

¹ Henry Moseley X-Ray Imaging Facility, School of Materials, The University of Manchester, ² School of Mathematics and Manchester Centre for Nonlinear Dynamics, The University of Manchester, ³ Department of Applied Mathematics and Computer Science, Technical University of Denmark
Contact email: parmesh.gajjar@alumni.manchester.ac.uk; Funding provided by Collaborative Computational Project in Tomographic Imaging, www.ccp.ac.uk

X-Ray CT machines have been powerful tools for complete 3D imaging and analysis of medical specimens and industrial artefacts, ranging from the scanning of steam location and aeroplane jet engines to parasitic bacteria. Technological advances have meant that the cost of these machines is now affordable to many institutions. But, despite the wide scope and increased availability, most machines are restricted to being used as “black-boxes”, with the user limited to only those routines provided by the manufacturers.

One unique feature of the 320/225 kV Nikon XTEK bay is its ability to be “hacked”, namely programmed in a custom manner by the user. The Nikon Inspect-X software comes with a graphical user interface that allows the user to perform 2D and 3D scans, inspections, along with many options such regions of interest, shading corrections, automatic reconstruction and batch processing. The Inspect-X software also contains a module enabling the entire system to be controlled through Visual Basic for Applications (VBA). The VBA editor is housed within the Inspect-X software, and is the same VBA editor found in Microsoft Word, meaning that generic code can be developed away from the CT machine. Each aspect of the X-Ray source, manipulator and image processing pipeline can be accessed using individual functions, allowing the user to build up a series of simple macros for performing certain tasks, e.g. switch x-rays on and off, capture and save an image or rotate the manipulator by a set angle. These simple macros can then be combined together in complex applications. The VBA programming capability of the 320/225 kV Nikon XTEK bay demonstrates the nearly unlimited potential of lab-based X-Ray CT systems to resolve a whole range of new complex scientific and industrial problems.

The panel to the right contains some sample code for switching x-rays on and off, whilst the panels to below display three more complex applications. The code is available on CCPForge.



Sample VBA Code

```
' Small routine for switching X-Rays on and off
Public Sub main()

' Place X-ray controller in external mode
XrayController.ControlMode = OperationalMode_ExternalMode

' Set voltage to 100 kV
Dim DemandVoltage As Integer
DemandVoltage = 100
XrayController.kVDemand = DemandVoltage

' Set current to 80 uA
Dim DemandCurrent As Integer
DemandCurrent = 80
XrayController.uADemand = DemandCurrent

' Specify tolerance percentage of 6%
Dim Tolerance As Integer
Tolerance = 6

' Specify timeout of 1 min = 60 seconds
Dim TimeoutSec As Integer
TimeoutSec = 60

' Switch x-rays on
XrayController.XrayDemand = XrayControlMode_xraysOn

' Intialise status variable
Dim status As MatchSuccess

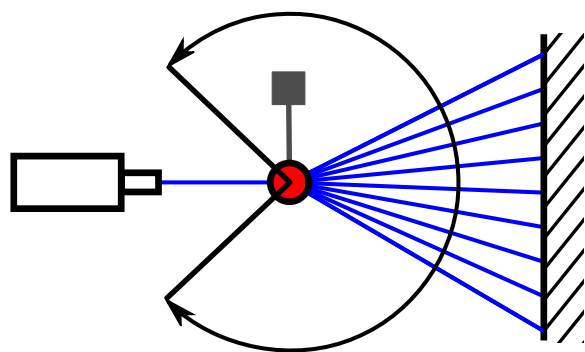
' Wait until X-rays have stabilised
status = XrayController.WaitForXrays _
    (WaitMatchProfile_matchBoth, TimeoutSec, _
    Tolerance * DemandVoltage / 100, _
    Tolerance * DemandCurrent / 100)

' If status is ok, wait 10 seconds before switching off
If status = MatchSuccess_success Then
    InspectX.Wait (10000)
    ' Print status message
    Debug.Print "Success"
End If

' Switch X-Rays off
XrayController.XrayDemand = XrayControlMode_xraysOff

' Place X-ray controller back to user mode
XrayController.ControlMode = OperationalMode_ManualMode

End Sub
```

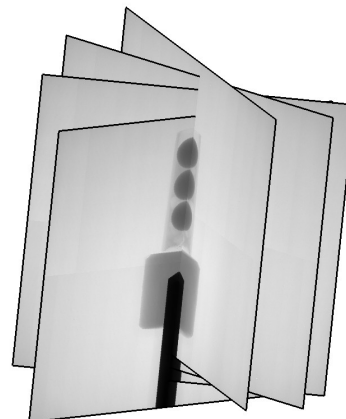


Limited Angle Scan

Some experimental rigs have support structures made from dense metals. These support structures may cause excessive absorption and/or scattering when they pass in front of the x-ray source in a full circular scan, rendering some projections unusable. By defining a set starting angle (say 30°) and a set finishing angle (say 300°), the specimen can be rotated through a limited angular range (in this case 270°). This also allows the specimen to be placed closer to the source, giving better resolution, without the support structures touching and damaging the source.

Helical Scan

Reconstructions of circular scans are known to cause artefacts away from the centre slice. An alternative is to acquire projections in a helical fashion, with the specimen simultaneously rotated and translated along the rotation axis. The VBA code allows each axis of the manipulator to be individually controlled, meaning that the specimen can be rotated and vertically translated to a new position before acquiring each new projection image. However, the reconstructed image is sensitive to the helical radius and detector-screen misalignment. For full details of this project, see poster 001.



In-situ Granular Shear Box

Code was written to allow two-way communication between the X-Ray scanner and an in-situ granular shear box. After a scan had been performed, a signal was sent to start the shearing of the box. Once the shearing had finished, a further scan could be initiated. The two communication code was developed away from the X-Ray machine using Microsoft Word, with the circular scan macro command replaced by a simple wait command. The final VBA code thus allowed more than 100 scan-shear cycles to be performed in a completely automated manner.

