LA-UR-99-5892

General Description of Software for PPIA/Mayak Transparency Technology Demonstration

Rena Whiteson, Diana Langner, Robert P. Landry
Los Alamos National Laboratory

James Wolford, Tom Gosnell
Lawrence Livermore National Laboratory

This document describes the software in the five computers contained in the Attribute Measurement System with Information Barrier (AMS/IB) that will be demonstrated for PPIA/Mayak Transparency. Information-barrier features will not be addressed in this document. Details about the information barrier can be found in the following documents:

Mayak/PPIA Demonstration Attribute Measurement System with Information Barrier: Functional Requirements; Rena Whiteson and Duncan W. MacArthur; Los Alamos National Laboratory; LA-UR-99-5634.

Proposed Attribute Measurement System (AMS) with Information Barrier for the Mayak/PPIA Demonstration: System Overview; Duncan W. MacArthur; Los Alamos National Laboratory; LA-UR-99-5611.

Overview

The AMS/IB will measure six attributes. Each attribute will be compared to a stored threshold and the display will indicate for each attribute whether or not the threshold has been met. The attributes are:

1) presence of plutonium,
2) plutonium isotopic ratio,
3) plutonium mass,
4) plutonium age,
5) absence of oxide, and
6) symmetry of the plutonium.

The following three detectors will be used to measure the attributes:

1) Pu300/600—a medium-sized (50%), germanium-detector-based, high-resolution gamma-spectroscopy system;
2) Pu900—a large (100%), germanium-detector-based, high-resolution gamma-spectroscopy system; and
3) NMC—a neutron multiplicity well counter with the ability to read out each individual bank of tubes.

The following four analyzers will be used to determine the values of the attributes:

1) Pu300/600 Analyzer,
2) Pu900 Analyzer,
3) NMC Analyzer, and
4) Symmetry Analyzer.

Table 1 shows which detectors and analyzers determine the values for each attribute. Attribute values will be passed from the analyzers to the computational block, which will compare those values to stored thresholds and determine whether each attribute threshold has been met. The computational block will output the pass/fail results of the threshold comparisons to a display via the data barrier. The computational block will send two additional outputs to the display: a *measurement complete* signal and a *state of health* indicator.

Table 1. Detectors and Analyzers for Each Attribute

	Attribute	**Detector**	**Analyzer**
1	Presence of Plutonium	Pu300/600	Pu300/600 Analyzer
2	Plutonium Isotopic Ratio	Pu300/600	Pu300/600 Analyzer
3	Plutonium Mass*	Pu300/600 NMC	Pu300/600 Analyzer NMC Analyzer
4	Plutonium Age	Pu300/600	Pu300/600 Analyzer
5	Absence of Oxide†	Pu900 NMC	Pu900 Analyzer NMC Analyzer
6	Symmetry of the Plutonium	NMC	Symmetry Analyzer

* Plutonium Mass is computed in the computational block.
† Oxide must be found by both detectors to be considered present.

Figure 1 shows the data flow between detectors, analyzers, computational block, and data barrier.

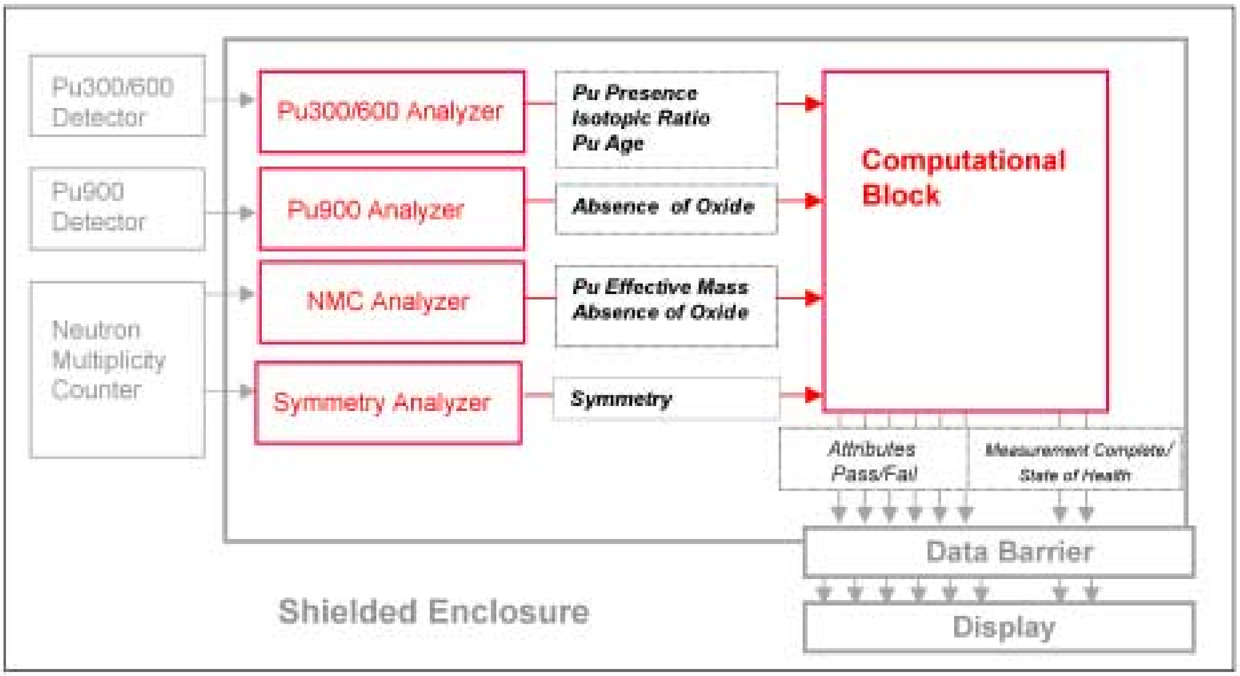

Fig. 1. Data flow between analyzers, computational block, data barrier, and display. The five computers in the AMS/IB are shown in red.

Computers

There will be five computers in the AMS/IB. Each of the four analyzers and the computational block will be a simple computer, based on PC104 technology, running the DOS operating system. The analysis programs that run on the analyzers were written especially for the AMS/IB either at Los Alamos National Laboratory (LANL) or at Lawrence Livermore National Laboratory (LLNL). The data barrier and the display modules contain no software. The following are brief descriptions of the software programs on each computer.

1. Neutron Multiplicity Counter (NMC) Analyzer

Mass Attribute (Multiplicity Counter)

Author: William C. Harker
Language: Microsoft Visual C 1.52
Platform: MS-DOS compatible
Description:
The NMC used in this demonstration consists of thirty-six ^{3}He tubes mounted within the four walls of a rectangular polyethylene enclosure. During normal operation, the count rates for total neutrons, doubly coincident, and triply coincident events (singles, doubles, and triples) are measured in the NMC detector. If the efficiency and die-away characteristics of this detector have been measured, then the singles, doubles, and triples can be used to calculate the ^{240}Pu effective mass, the fraction of (α,n) reactions (or α), and the multiplication in the sample. With the isotopic ratio, the ^{240}Pu effective mass is used to calculate the Pu mass and α is used as an indicator of oxide presence. The determination of oxide absence will be performed within the computational block, which processes the measured parameter alpha and the determination (obtained from the Pu900 Analyzer) of whether an oxygen line is present in the high-resolution gamma-ray spectrum.

The software used for this process is a modified version of the standard international neutron-coincidence-counting code INCC. Shift-register communication and analysis routines are unmodified from the commercial INCC. Executive functions, switch communication, and video-output routines are new and are extremely streamlined compared to those of INCC. Except for switch communication, all user input functions have been removed.

Symmetry Attribute

Author: William C. Harker
Language: Microsoft Visual C 1.52
Platform: MS-DOS compatible
Description:
As well as being combined for the multiplicity measurement, signals from the eight octants of the neutron detector will be counted individually to provide an indication of the symmetry (in 2 dimensions) of the object under test.

2. Pu300/600 Analyzer

Acquisition Software (common to the Pu300/600 and the Pu900)

Author: Gregory K. White
Language: ANSI C
Platform: MS-DOS compatible
Description:
The acquisition software is a collection of five modules that perform the necessary tasks for acquiring data from the Canberra InSpector and submitting those data to any of three Pu*x*00 applications (Pu300, Pu600, Pu900) for attribute determination. Functionally, the modules are separate, executable processes that run in a prescribed sequence and are controlled by a batch file. Table 2 summarizes their identities and purposes.

Table 2. Acquisition Software Modules and Purposes

Name	Purpose
ACQUIRE.EXE	*MCA Acquisition Software*
BCKGEN1.EXE	*Background Peak Finder*
DONE.EXE	*Status Communication to Computational Block*
SENDOUT.EXE	*Results Communication to Computational Block*
STARTCD.BAT	*Process-Control Batch File*

As their names suggest, the process modules correspond to files on the analyzers. STARTCD.BAT executes ACQUIRE.EXE, and data acquisition proceeds according to the state of the control switchblock, which indicates whether acquisition for calibration, background, or source will take place. These acquisitions differ subtly and lead to different outputs and return values from the “acquire” process.

Calibration begins with a pole-zero acquire, followed by a brief spectrum acquire to provide the peaks needed for the energy calibration. The process then creates/overwrites output file CALIB.IN, which provides corrections to the prescribed multichannel analyzer (MCA) settings of pole-zero and fine gain. When calibration concludes, STARTCD executes DONE.EXE to signal the computational block that this operation is complete.

To prepare for background acquisition as directed by the control switchblock, ACQUIRE.EXE reads the results of the calibration process from CALIB.IN and resets the appropriate MCA parameters. It then acquires background for sufficient time to yield an adequate baseline. It preserves the background spectrum in BCKGEN.IN before exiting. STARTCD then executes BCKGEN1.EXE to extract relevant peaks from the background spectrum and format them as required for subsequent physics analysis (Pu*x*00).

When the control switchblock is in source mode, ACQUIRE reads and applies the calibration information in CALIB.IN. It then acquires a spectrum for a period prescribed by the intended physics application. This is used to produce a PU*x*00.IN file for subsequent analysis.

Once the analysis has yielded its respective attribute, STARTCD uses SENDOUT to communicate the result to the computational block as a text string, via serial line.

Age Attribute (Pu300)

Author: Dan Archer
Language: ANSI FORTRAN 90
Platform: MS-DOS compatible
Description:
Pu300 analyzes plutonium spectra over a narrow range of photon energy, 330 keV $\leq E \leq$ 350 keV, collected by a high-purity germanium detector for the purpose of estimating the "age" of the source material. Age in this context refers to the time interval since ^{241}Am was extracted from the source material in preparing the latter for service as weapons-grade plutonium. The age estimate rests on the assumption that, by design, no ^{241}Am is present in such sources after processing. From knowledge of the decay paths and half-lives of ^{241}Pu and its decay products, one can calculate the age and an error from the relative quantity of ^{241}Am which has "grown into" the source since its processing to weapon-grade material. STARTCD executes Pu300 at the conclusion of a source acquire, having previously collected a background spectrum. Pu300 begins by searching the source spectrum for telltale lines from ^{239}Pu to verify its presence. The software accepts information about the radiation background present during the measurement and corrects the source spectrum as needed. It deconvolves the constituent intensities in well-known lines generated by the various nuclides present and derives the contributions due to ^{241}Am, ^{241}Pu, and ^{239}Pu. It then determines the atomic ratio ^{241}Am/^{241}Pu and uses this to calculate the age of the material based upon exponential radioactive decay. Finally, it uses estimates of the measurement uncertainties in source and background to derive an age error. These results are written to a RAM-disk file for later transmission to the computational block by SENDOUT.

Material Grade Attribute (Pu600)

Author: John Luke
Language: ANSI FORTRAN 90
Platform: MS-DOS compatible
Description:
Pu600 analyzes plutonium spectra over a narrow range of photon energy, 630 keV $\leq E \leq$ 670 keV, collected by a high-purity germanium detector. This portion of the spectrum contains intense lines from ^{240}Pu and ^{239}Pu that can be used to calculate the ratio of these two nuclides in the source material. STARTCD executes Pu600 at the conclusion of a source acquire, having previously collected a background spectrum. Pu600 begins by searching the source spectrum for telltale lines from ^{239}Pu to verify its presence. The software accepts information about the radiation background present during the measurement and corrects the source spectrum as needed. It then calculates the mass ratio ^{240}Pu/^{239}Pu and an error based on known detector livetime and background correction. These results are written to a RAM-disk file for later transmission to the computational block by SENDOUT.

3. Pu900 Analyzer

Absence of Oxide Attribute (Pu900)

Author: John Luke, Greg White
Language: ANSI C
Platform: MS-DOS compatible
Description:
Pu900 analyzes plutonium spectra over a narrow range of photon energy, 840 keV ≤ E ≤ 880 keV, collected by a high-purity germanium detector for the purpose of detecting the presence of significant amounts of oxygen in the source sample. Absence of oxygen in the source spectrum would imply that the source was not composed of plutonium oxide and could be assumed therefore to be plutonium metal. The method extracts the intensities of oxide indicator peaks and compares them to background to infer the presence or absence of oxygen in the source. STARTCD executes the Pu900 routine at the conclusion of a source acquire after a background spectrum has been acquired. The software uses information about the radiation background that is present during the measurement and corrects the source spectrum as needed. It locates the prescribed peaks and determines, from their intensities, the prevalence of oxygen in the source. Finally, it uses estimates of the measurement uncertainties in the source and background measurements to derive an error. These results are written to a RAM-disk file for later transmission to the computational block by SENDOUT.

4. Symmetry Analyzer

Spherical Symmetry Attribute (Neutron Measurement)

Author: Tom Gosnell
Language: ANSI FORTRAN 77
Platform: MS-DOS compatible
Description:

Program nSymmetry analyzes singles neutron counts provided by eight banks of four ^{3}He tubes in the neutron multiplicity counter. The eight detector banks surround the fissile-material storage container, and the effective centers of all of the detector banks are equidistant from the center of the storage container. For the FMTT demonstration, the measurement is made about the axis of the storage container.
In principle, a cylindrically symmetrical object placed in the center of the storage container will produce the same count rate in each of the eight detector banks. Because small differences in detection efficiency occur in each of the detector banks, the net counts from each of the banks are adjusted by nSymmetry for small efficiency differences based on an initial calibration with a cylindrically symmetrical source. The efficiency calibration constants are written into the compiled nSymmetry code.
The symmetry attribute is determined by a simple analysis of the adjusted net (background subtracted) detector counts to find the one detector that deviates the most from the average value, $\bar{y}$, of the adjusted net counts from all of the detectors. The absolute fractional deviation about the average, s is computed.

$$s = \max\left(\frac{|y_i - \bar{y}|}{\bar{y}}\right),\ \sigma_s^2 = \frac{\sqrt{y_i}}{\bar{y}}$$

Where y_i is the number of net counts recorded in the *i*th detector bank,
$\bar{y}$ is the mean number of counts from all 8 banks, and
σ_s^2 is the variance of s .

Program nSymmetry passes the value of s and s/σ_s on to the computational block. The computational block compares these values to arbitrary threshold values chosen for the FMTT demonstration. To be declared asymmetric, the value of s must exceed 0.15 and this value must be statistically significant as determined by a value of s/σ_s in excess of 3.0.

5. Computational Block

Author: Robert P. Landry, Connie Buenafe
Language: Microsoft Visual C 1.52
Platform: MS-DOS compatible
Description:
Classified data from the detectors, the Pu300/600 system, the Pu900 system, and the NMC are input to a computational block where the DATA_ATT.C program performs the final calculations to determine whether the measured item exceeds or falls short of agreed-upon threshold values. The unclassified attribute thresholds are stored in the computational block in read-only memory.
DATA_ATT.C performs the following functions:

- stores the isotopic ratio of the analyzed material;
- computes the Pu mass from the ^{240}Pu effective mass and the isotopic ratio;
- compares the Pu mass to the mass threshold;
- compares the Pu ratio to the ratio threshold;
- compares the Pu age to the age threshold;
- compares the oxide value from the NMC to the oxide threshold;
- compares the oxide value from the Pu900 to the oxide threshold; and
- compares the symmetry value to the symmetry threshold.

The six pass/fail results are then output to the data barrier. These unclassified outputs indicate only whether the measured values meet or fail to meet the stored thresholds.

DATA_ATT.C executes the following steps.

1. Both the video card and the parallel I/O card are initialized.
2. The serial ports internal to the processor card are initialized.
3. The lights on the display panel are cleared.
4. A loop is entered, waiting for one of four switches to be pressed. The four switches are: background, measurement control, calibrate, and measurement.
5. After a switch is pressed, another loop is entered, waiting for data to be received on the serial port.

6. Data received from the serial port are analyzed, compared to set limits, and appropriate lights are illuminated on the display panel.
7. Step 4 is executed again.

As a switch is pressed, the computational block digital I/O card receives the “switch down” signal and changes a data word that is readable by the processor card. The program running on the processor card leaves the perpetual loop when the data word is non-zero, indicating one switch has been pressed. The program then determines which switch has been pressed and starts looking for data to be placed in a null data string. An interrupt process is continuously running in the background, accepting all data that come in through the serial ports and placing the incoming data into a data string. As data enter the data string, the string is no longer null, and the program will recognize this fact. Data in the string are then analyzed. Based on that analysis, different lights are illuminated on the display panel. To accomplish this, a bit string is constructed with various “ones” placed in specific locations. A “one” indicates the light associated with the bit is to be turned on, and a “zero” indicates the light is to be turned off. This bit string is output to the digital I/O card and is held present on the inputs to the flip-flops located on the data barrier. This is followed by another bit string containing a single “one” bit to clock the data through the flip-flops on the data barrier and illuminate the lights on the display panel. The flip-flops are extremely efficient low-pass filters that restrict data to low-frequency data pulses only, thus preventing information from being transmitted on a high-frequency data string.

Additional Software Components

The following commercially available components are used in the AMS/IB. These proprietary processors are controlled by software.

- Two Canberra InSpectors.
- One Aquila PSRB Shift Register.