

## Subject-Oriented Software Development System for DSP

---

### Overview

Data acquisition for real-time applications on a DSP board is an engineering challenge. Currently DSP engineers can take up to one and a half years to develop a Visual C++ program. End users with limited programming skills cannot take advantage of the strength of DSP functionality, while DSP programmers cannot satisfy the time-to-market requirements of their customers. Data Translation has resolved this dilemma by creating its own DSP subject-oriented software development system. This system lets users access sophisticated DSP algorithms from the standard Windows® world. This means that users without in depth DSP programming knowledge can access the power of real-time DSP. This system is based on two major components – a subject-oriented DSP operating system and a patent-pending Windows-based rapid software development system. The subject-oriented DSP software development system uses techniques of aspect-oriented programming and gives maximum flexibility because of its purely task-based orientation. A user does not have to have programming skills. The system simply uses standard Windows drag and drop mechanisms. One graphical user interface can be used to develop Windows and DSP applications.

The purpose of this paper is to illustrate how an emerging technology such as the DSP subject-oriented development system can be used for real-time application development. This approach combines the minimum learning curve for Windows-based software development with the maximum performance of precompiled and optimized code for a DSP board. It also reduces time-to-market, because there is no need for code debugging. This method also uses Windows standard configurations which maximize its use in many different types of applications.

### Introduction

Subject-oriented programming in this paper is defined as purely problem-oriented. The solution itself stands in the center of interest. This system is based on the principles of subject-oriented programming (SOP).



A panel can be a data source, control source or event source that allows the software to interact with the DSP task (DSP function panels can be configured and connected together but carry no PC code functionality.) Data channels are the normal data transportation mechanisms used to transfer data from the hardware (data source) to a panel on the desktop. Data channels consist of real channels (physical channels), such as those on a data acquisition board, and mathematical channels (channels that have been created by combining two or more real channels). Control sources are a set of panel properties and actions that the user can control. Events allow the user to react to a change in the application. After putting all DSP functions on the container and configuring them, an aspect interaction file (AIL) file is created (see Figure 3.)

Developed by Data Translation and based on the concepts of AOP1, aspect-oriented language (AIL) is a patent-pending description language. It describes the behavior, appearance and connections of aspects and is used to store/save the descriptions of the aspects of the program. AIL is not a programming language – it does not become interpreted or compiled. It is only a description.

The AIL contains information that describes the task such as process ID, used pipes and variables, needed resources, used function objects and their properties, interconnection between function objects and memory consumption. The AIL file generated on the host interface is sent to the DSP system and the aspect interaction language parses that file and translates its contents to a DSP readable form.

The AIR (aspect interaction run-time) command processor receives the AIL file and compiles it to an aspect structure.

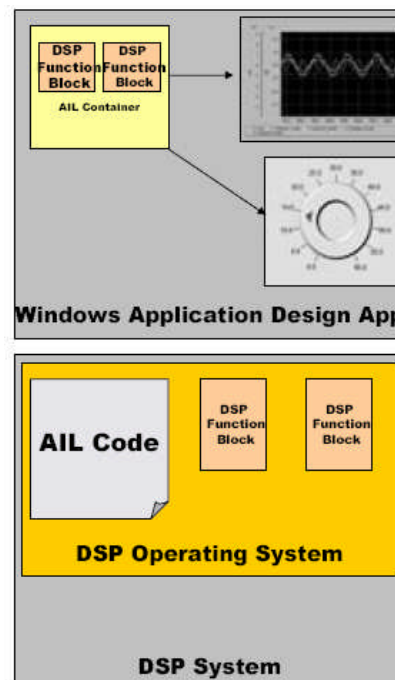


Figure 3. Underlying Concept of AIL

Once the AIR command processor receives an AIL file, the following are checked:

- if the process ID already exists, the old task is terminated and removed
- memory checks as needed
- resources needs are checked
- resources are allocated
- pipes and variables are allocated

Function objects are now created and their properties are set. After all the objects have been created, they have been interconnected. (If an error has occurred, the task is removed with all of its pipes and resources and an error message is sent). The task is ready to run now.

The DSP operating system is loaded via the USB 2.0 interface of the hardware platform and consists of the following parts:

- Resource Manager that manages allocation of board's hardware resources (A/D, D/A, counter/timer, digital I/O)
- Command processor translates incoming AIL
- Instrumentation capabilities such as monitors, CPU, data traffic, etc.
- Function libraries (e.g. digital filters, FFT, etc.)

Each component of the target DSP has an entry in the XML-based description catalog of the function libraries. This standard flat file format describes the component's behavior, its relation to other components and its parameters.

In a future version, Data Translation's DT Measure Foundry™, test and measurement application software, will be the Windows-based interface to the DSP operating system. DT Measure Foundry will search for the library of DSP functions using XML and create the AIL container. This container will display graphical representations of functional blocks, such as an FFT or IIR filter. As the interface between Windows and the DSP, DT Measure

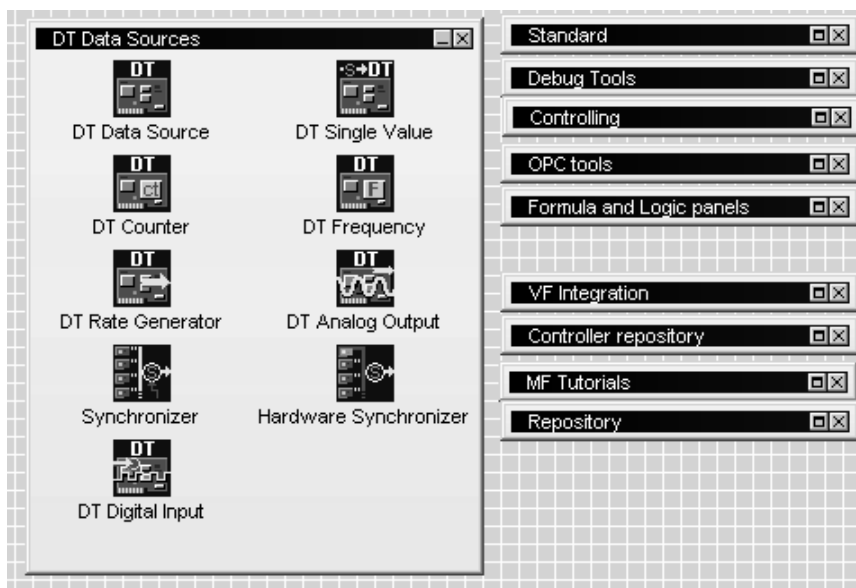
Foundry will create the AIL code and send it to the DSP operating system, which in turn, configures it into parameters of binary DSP modules. This is transparent to the user, so that programmers without in-depth knowledge of DSP programming can access the power of real-time DSP. The development of a measurement application follows the general procedure.

## Create a DSP Application Using DSP/OS

In the following example, a user creates a measurement application, connecting a DAQ board data source to a digital display panel, to display sampling data from selected data channels of the data source.

### Creating Connections

The user begins to build the application by setting up a connection to the data acquisition hardware that will let the measurement data be viewed and processed. The user places the data source on the design desktop, using conventional drag and drop techniques. Having clicked on the menu icon to open the menu list, the user then selects the “panels” entry, which opens a list of panel groups available for selection including data source panels, standard panels, controlling panels and other panel groups. The user selects the “data sources” panel list, which opens to present a list of all data sources (represented by icons) available on the system (Figure 4). From the list, the user clicks on the icon representing the data source of interest.



**Figure 4.**  
**List of Available Data Sources**

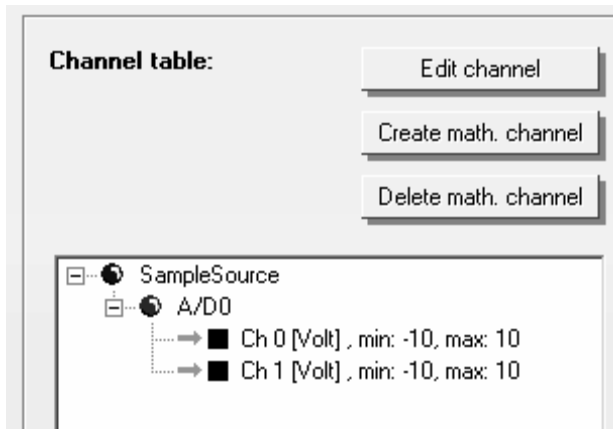
This action results in a cursor-controlled cross mark appearing on the desktop: moving this cross mark to a desired location on the desktop and clicking the left mouse button places a data source panel at the selected location. Alternatively the user may place a data source panel on the desktop by making use of the “flying” tool window. On the desktop, a set of “flying” tool bars that represent several groups of panels available to the user are displayed. These bars include a data source panels bar, a standard panels bar, a controlling panels bar, etc. Clicking on any of these bars opens a “flying” tool window containing icons representing each of the panels in that panel group. Clicking on this produces the same cursor-controlled cross mark referred to above, which is used to locate the selected panel at a desired location on the desktop.

The data source panel placed by a user on the desktop, like all other panels available to the user, represents a software object, here termed an “aspect,” comprising the attributes, actions and events appropriate to a virtual data source instrument, as described in further detail below.

Having placed the data source panel at a desired location on the desktop, the user then proceeds to configure the data acquisition hardware corresponding to the data source panel. The user configures the data source panel by opening the property page of the data source panel. For example, the property page of the data source may be opened by clicking the right mouse button with the cursor placed over the top frame of the panel.

Opening the property page of the data source aspect results in opening and presenting the user with a dropdown list of data acquisition devices then available on the system that are of the type “DT data source.”

Upon selection by the user of a specific data source from the data source listing, a box displays a listing of subsystems associated with said data source. For example, there may be an “A/D subsystem 1” check box and a “digital out 1” subsystem check box. The user makes a selection of a subsystem to be used, say the “A/D subsystem”, and then a new A/D subsystem button is created. On the right side of the “properties of data source” page, there appears a channel table (see Figure 5), identifying (with the sample source) a default channel Ch 1 [volt] min – 10 max 10).



**Figure 5. Channel Table Showing Defaults**

Clicking the A/D subsystem button opens a configuration page for the A/D subsystem in a channel table forming the right side of the data source property page. The analog input channels provided by the A/D subsystem of the selected data source are displayed on the channel table.

The user is thus enabled to configure the specific hardware device to be used as a data source, for example by setting the clock speed (say, from 50hz to 500hz), channel range (say, from a - 10 min to a + 10 max), and the buffer size that will be employed as the operating parameters of the selected input channel of the selected data acquisition hardware device. Upon selecting an input channel, by checking the box adjoining the selected input channel in the presented listing of input channels provided by the selected data source device, the selected channel becomes listed in the right-hand “channel table.”

The channel table view includes buttons respectively marked “edit channel”, “create math channel” and “delete math channel.” The “edit channel” button, when clicked, opens an “edit channel properties” frame comprising drop down menus for modifying a pre-set selection of channel properties (which have been pre-selected taking into account the inherent capabilities and characteristics of the various channels provided by specific types of data source devices). Thus certain channels are dedicated to voltage outputs, and the “unit” value of the data may not be altered; in other cases the user is offered a choice between, say, voltage or current units. In all cases the user may select a name for the channel. Typically, the user is provided with the opportunity to select a range including a minimum value and a maximum value (say, - 10 to + 10). The user, having made selections from the proffered choices of properties, activates those selections by pressing the “apply” button (which activates a selection but leaves the “edit channel properties open” page open) or the “OK” button.

The user next places a visualization panel on the design desktop, on which to display measurement data received from the data source. In the present example, the user has chosen to place a digital display panel on the desktop. The digital display panel is capable of displaying up to six digits. The digital display panel is found within the standard panels group. Therefore, the user can place a digital display panel on the desktop by clicking open the Main Menu and selecting Panels/Standard/Digital display. Alternatively, the user can drag and drop a digital display icon from the standard panels “flying” tool window on the design desktop.

Preferably, the user activates a box location and sizing tool. The user holds down the left mouse button and drags the mouse in a diagonal to size a digital display panel of desired dimensions. This procedure is in all respects similar to that used for creating boxes in conventional drawing programs. Upon release of the left mouse button by the user, the panel is placed and sized as defined by the rectangle drawn by the user.

### **Creating Properties**

The user then proceeds to select desired properties for the newly-created digital display panel by opening its property page. The property page of the digital display panel offers the user three separate groups of properties for making selections, including a “general” group, a “data reduction” group, and a “channels” group, each identified and selectable from a button bearing the property group name.

The set of “general” properties available for user selection includes choices of “component name,” “digit’s color”, “background color”, “number of digits”, and “decimal point” number (from 1 through 6).

To select the input channel that is to be displayed on the digital display panel, the user presses the “channels” button on the “property of digital display”, thereby opening the data sources view page. The data sources view page includes a “data sources” drop down list, which presents a list of available data sources. User selection of one of the listed data sources results in the presentation, on a “channels” view box (located below the “data sources” drop-down list box) of a tree list of the channels available from the



selected data source, with selection check boxes adjoining each listed channel. As an example, a DT data source 1 has been selected, and its two available A/D0 channels are available for selection with the “Input channel 2” check box checked for illustrative purposes. The user may then close the “Digital display” property page by clicking the “Apply” button, then the “OK” button in the lower part of the property of digital display page. It should be understood that the user may select as many “digital display” panels as are needed or useful for the user’s measurement project, and the user may accordingly open a second digital display panel, using the procedures described above, in order to create a digital display on which to display the output of a second input channel associated with “data source 1.”

In this example, the digital display panel is provided with a “data reduction” set of properties. Clicking on the “data reduction” button of the digital display property page thus allows the user to select a desired form of presentation from the following list of pre-set options for presenting data received as buffer data from a data source:

- First value in buffer
- Last value in buffer
- Average of buffer
- Minimum of buffer
- Maximum of buffer
- Root-mean-square of buffer
- Standard deviation of buffer

In order to receive and display data from “data source 1” on the newly created digital display panel, the user needs only to create a suitable controller panel and connect it to the data source panel of “data source 1.”

This controller panel will allow the user, in run time, to start and stop the presentation of data from the selected data channel of the data source to the digital display panel. Where the selected source of measurement data is a data acquisition board, commonly known as a “DAQ” board, this will require a DAQ controller panel. The user proceeds to place such a panel on

the design desktop by selecting, from the Main Menu, Panels/Controlling/DAQ controller (or, alternatively, by drag and drop of a DAQ controller panel from the appropriate “flying” tool window). As earlier described for the data source panel and the digital display panel, the user locates and sizes the DAQ controller panel by holding down the left mouse button while dragging the mouse along a diagonal path; releasing the left mouse button locates the DAQ controller panel in the space thus defined by the user.

The DAQ controller panel has the shape of a vertically oriented rectangular box and contains two control buttons respectively marked “Start sampling” and “Stop sampling.” The DAQ controller panel further comprises a “buffer size” drop-down box and a “clock frequency” drop down box, for use in selecting a desired sampling buffer size and a desired sampling frequency.

To connect the DAQ controller to the data source, and to the digital display panel previously created by the user to display data from the data source, the user accesses the property page of the DAQ controller panel.

On the property page the user clicks the down arrow on a “data source” box to open a drop down list that lists all data sources available on the system, with adjoining check boxes for making a selection. The user checks the box adjoining the desired data source, and then clicks the “Apply” button located at the lower part on said property page to activate the selection. Upon the user closing the property page, the user is returned to the DAQ controller panel, which is now empowered to control the sampling of data from the selected data source, for immediate display on the digital display page. By pressing the “Start” sampling button on DAQ controller panel, the user initiates the transport of measurement data from data source to the digital display panel, and such measurement data will be presented in the form selected on the property page of the digital display panel. Preferably, the “buffer size” and the “clock speed” drop down buttons of the DAQ controller panel are adapted to allow modification of the buffer size setting and of the clock speed setting at any time during run time.

In this example, a “Preview mode” may be used to display the run-time version of the measurement application as created by the user, for review and

possible further modification by the user. In Preview mode, the application can sample and display either simulated data or real data (where the computer system is connected to an external data acquisition device). Finally a “run-time mode” is also provided, in which a measurement application that was designed in the design mode can be loaded and run; however, the means to change the application in run-time mode are disabled and not accessible. Thus, panels cannot be added or removed in run-time mode. In design mode, however, the user may have selected control panels that permit the user to access and modify particular panel properties in the course of run-time. Thus, properties such as sampling buffer size may generally be modified in run-time. In addition, a distribution wizard is provided, whereby the measurement application may be packaged in a form suitable for distribution to other computers, including the provision of conventional application installation means for the automatic installation of the measurement application on other Windows-based personal computers.

## **Conclusions**

Significant advances in programming have been made over the last thirty years, with technologies such as structured programming, object-oriented programming, Booch object-oriented design, etc. Even with these advances, customers want faster, cheaper and easier-to-use code because as systems become larger and more complex, the scale and complexity problem continues to grow. Subject-oriented software development is new technology that addresses these concerns. Using SOP, programs are modularized using “aspects.” These aspects can be methods, properties, relationships and other attributes that characterize, collectively, each particular object. In OOP, these objects can contain redundant inheritance information. In SOP, an aspect is described only once. The reduced description is expressed in an aspect description language.

SOP is an abstract concept so the biggest challenge facing it is to change human nature! Programmers have a level of comfort and intellectual investment in their training, which is understandable. However, being open minded to a new way of thinking about development overcomes one major obstacle facing all programmers: the language drives the process. With SOP,

programmers have the freedom to make changes without factoring in restrictions imposed by the language.

### **Reference**

[1] <http://aspectj.org>