



Automotive Robotics (India) Pvt, Ltd.

4th Floor | Surya Towers | Secunderabad - 500003 | Phone: 040 - 30280431

web: www.autorobinc.com

Definition

CANalyzer is a pc based Graphical universal development Tool for CAN bus system, which assists in observing, analyzing, and supplementing data traffic on the Bus line.

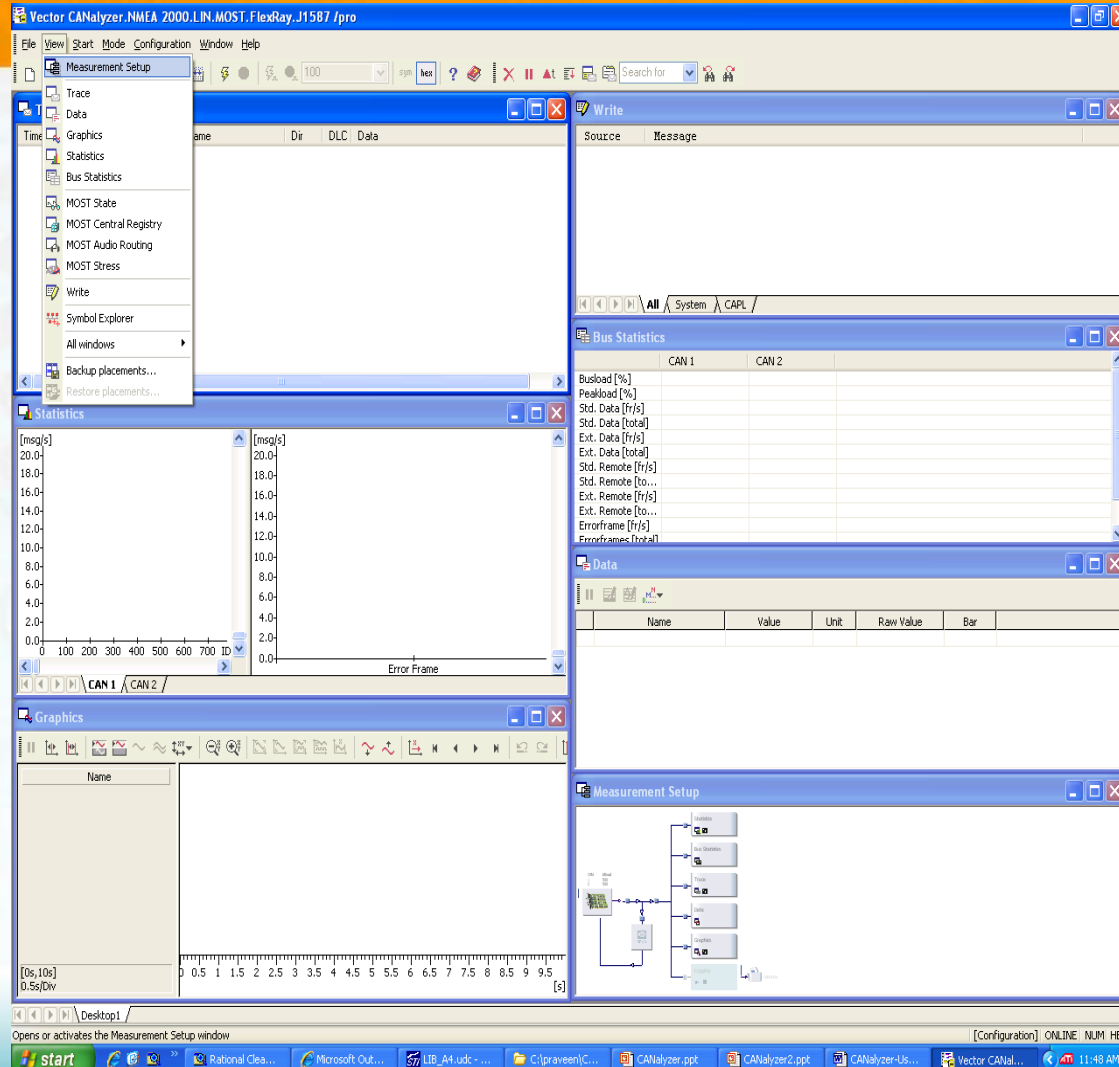
Features And Advantages

- CANalyzer is controlled from a Graphical Block Diagram.
- Depicts data flow from bus over various evaluation windows and to Log file.
- User can add or remove functional blocks and configure them, e.g. Generator, Filter, CAPL node.

Functionalities

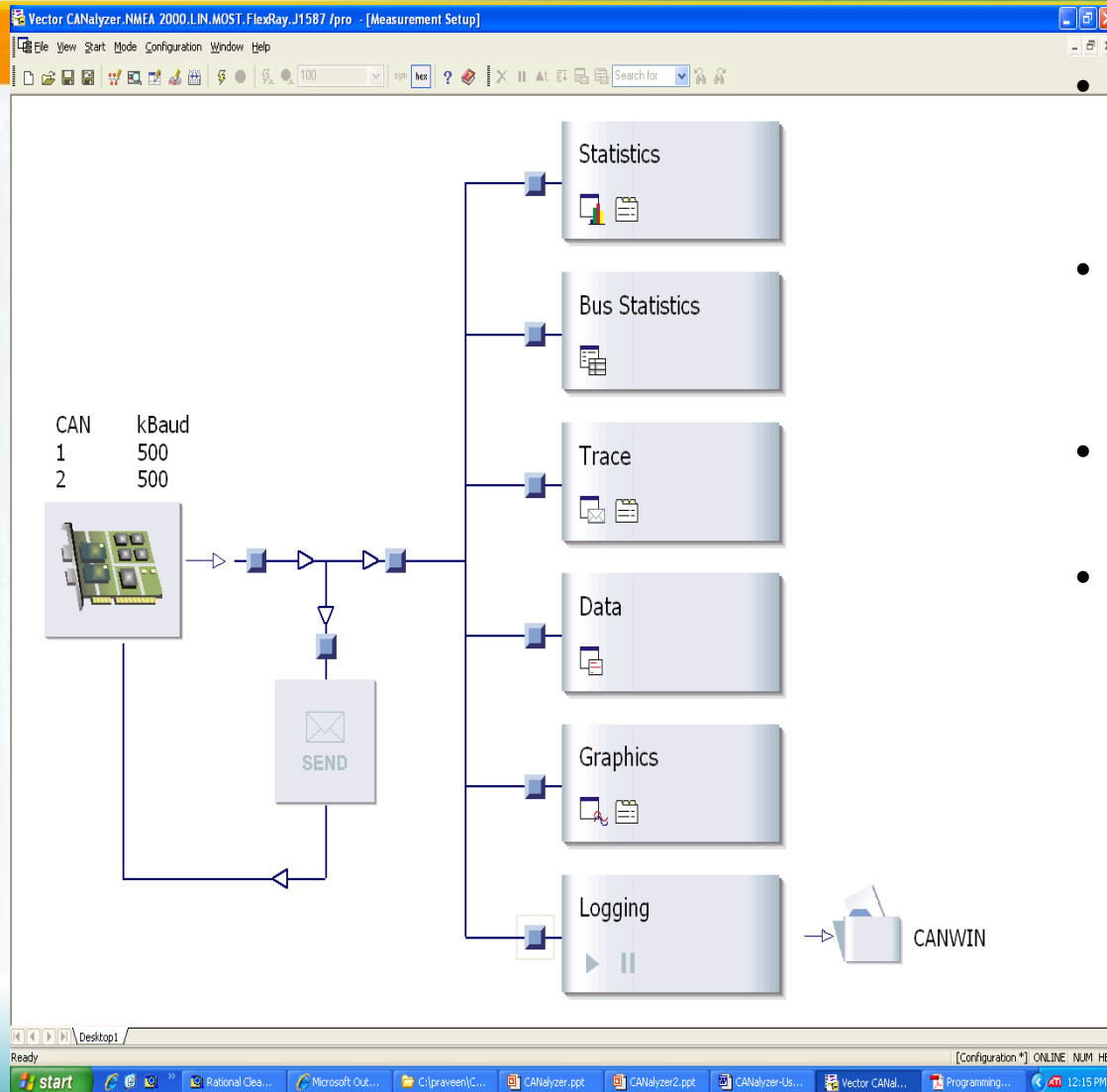
- Listing of Bus traffic (Tracing).
- Graphical and Textual display of signals.
- Interactive sending of predefined messages.
- Sending of logged messages.
- Statistics on messages.
- Statistics on bus loading and bus disturbances.
- Logging of messages for replay or offline evaluation.
- Generation of Bus disturbances.

Running CANalyzer



- Double click the CANalyzer icon in the program group and the window to the right appears.
- CANalyzer has various Evaluation windows (Data, Trace, Graphics, Statistics, Bus Statistics, Write and Measurement).
- User can access all programming windows from View menu on the Tool Bar.

Measurement Setup Window



- This Window shows the structural process and information flow of the CANalyzer configuration.
- PC Board represents the CAN hardware that handles the reception process.
- Send Block represents the transmission process.
- Additional functional block can be inserted or block can be restricted in the data flow at small black rectangles: -■- (Hotspots).

Setting the CAN bus

Vector CANalyzer.NMEA 2000.LIN.MOST.FlexRay.J1587 /pro - [Measurement Setup]

File View Start Mode Configuration Window Help

100

Search for

CAN kBaud

1	500
2	500

Channel Usage...
Hardware Configuration...
LIN Node...
Comment...
✓ Show comment
Import Measurement Setup Branch...

SEND

Statistics

Bus Statistics

Trace

Data

Graphics

Logging

CANWIN

Ready

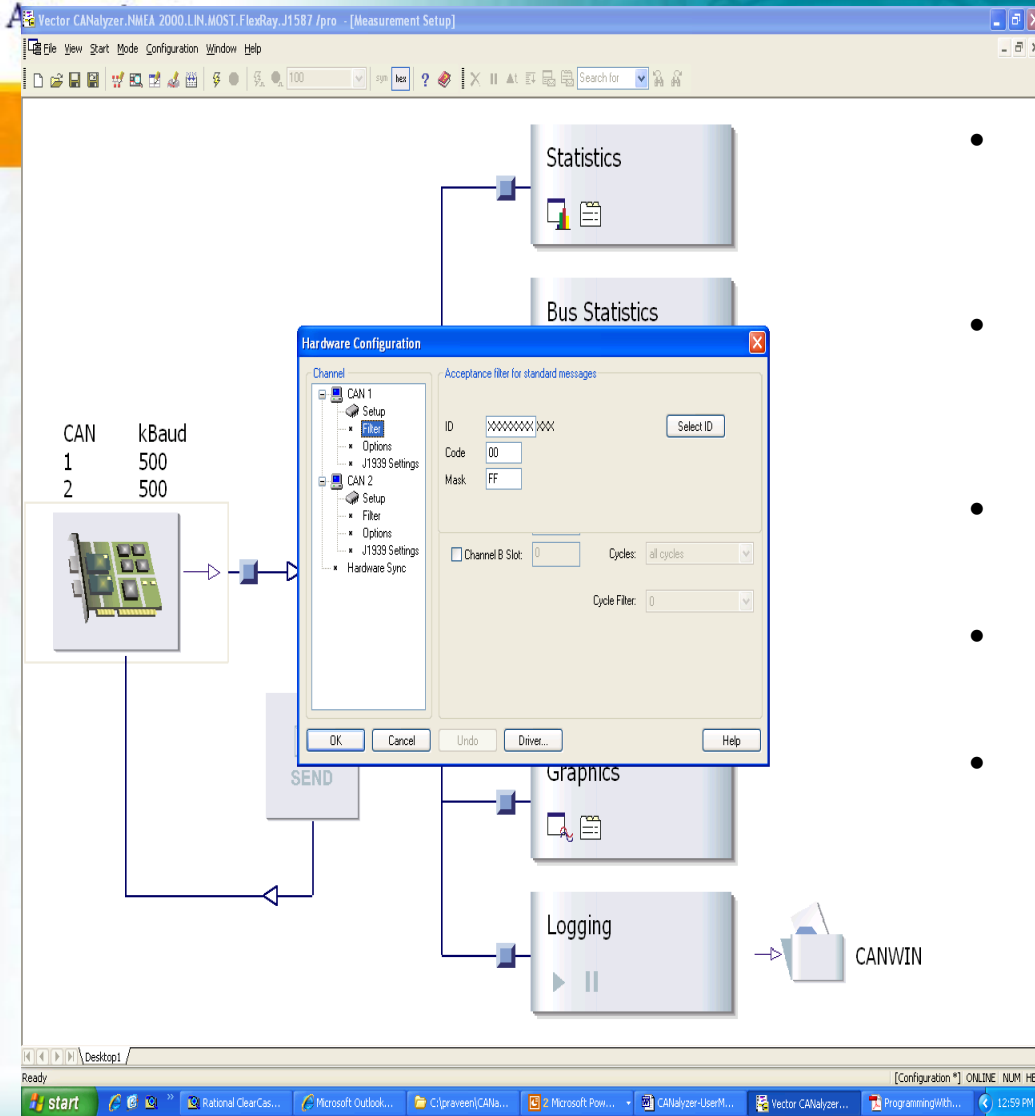
Desktop

[Configuration *] ONLINE NUM HEX

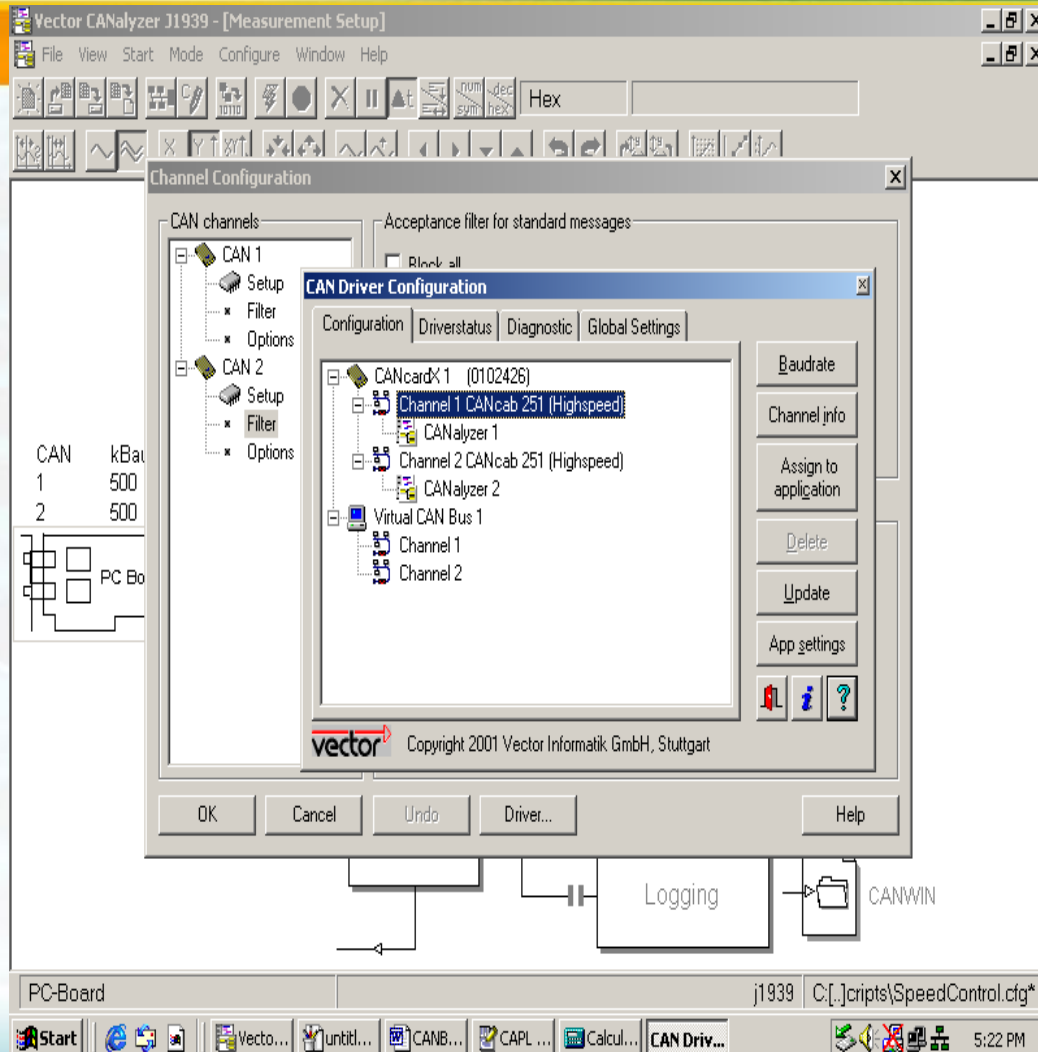
start

Rational ClearCa... Microsoft Outloo... C:\praveen\CAN... 2 Microsoft Po... CANalyzer-UserM... Vector CANalyzer... ProgrammingWith... 12:32 PM

- To Set the Bus parameters (Transmission Speed, Channel selection etc) click the right mouse button on the PC Board.
- Click on the Hardware configuration from the shortcut menu appeared to configure the Channel configuration.

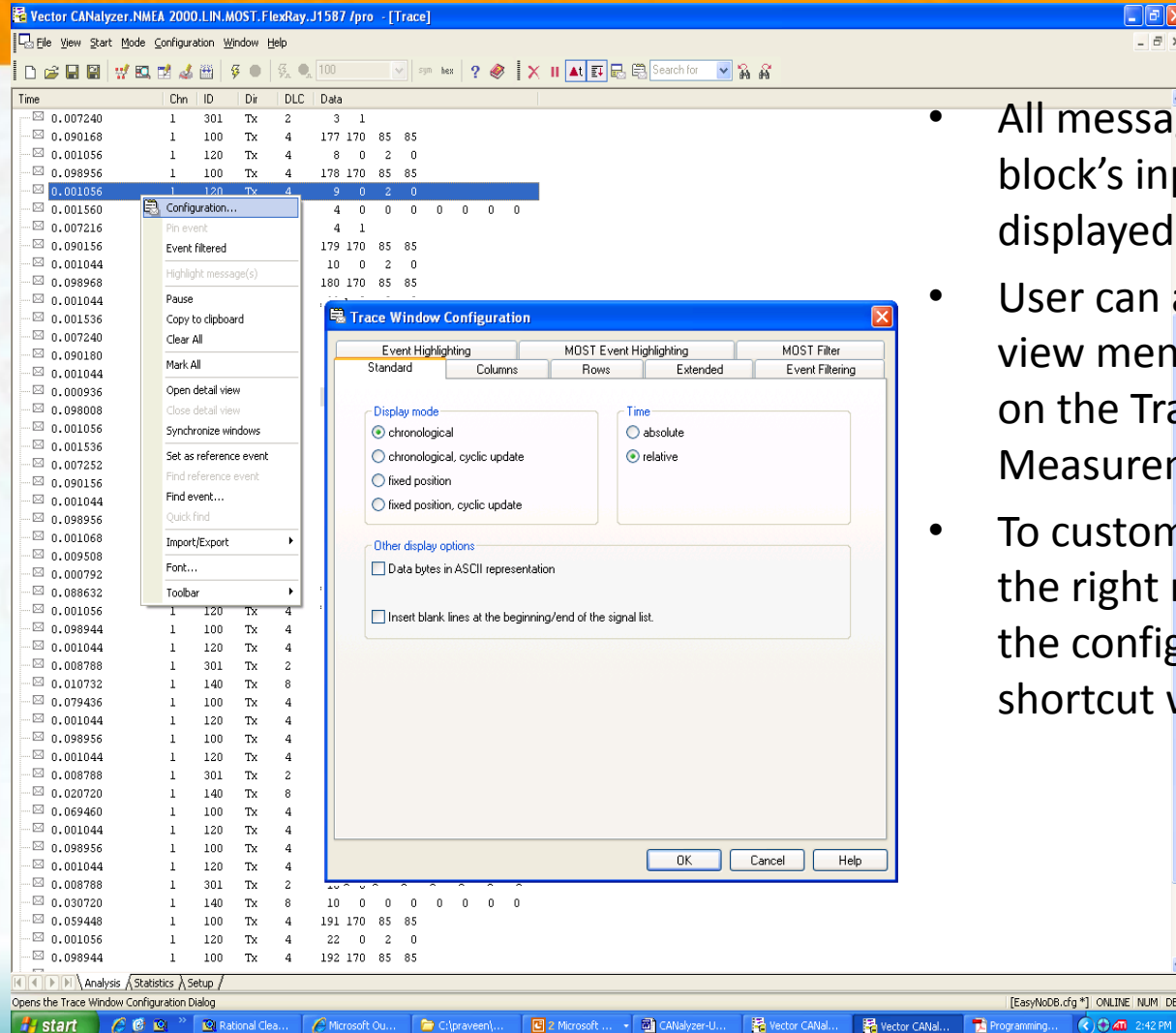


- Click on the Filter to configure the Acceptance filter for the Standard messages.
- The acceptance mask indicates which bit of the ID should be compared with the acceptance code.
- User can directly enter an Identifier by select ID.
- All the other identifiers are filtered out.
- To set the CAN driver configuration for transmission and reception of message, select the Driver button.



- Select Channel 1 CANcab 251[High Speed] and right click the mouse and select the option CANalyzer1 for sending messages.
- Select Channel 2 CANcab 251[High Speed] and CANalyzer2 for receiving messages.
- Then press update button to update the changes done.
- To configure the Baudrate click on the Baudrate button.

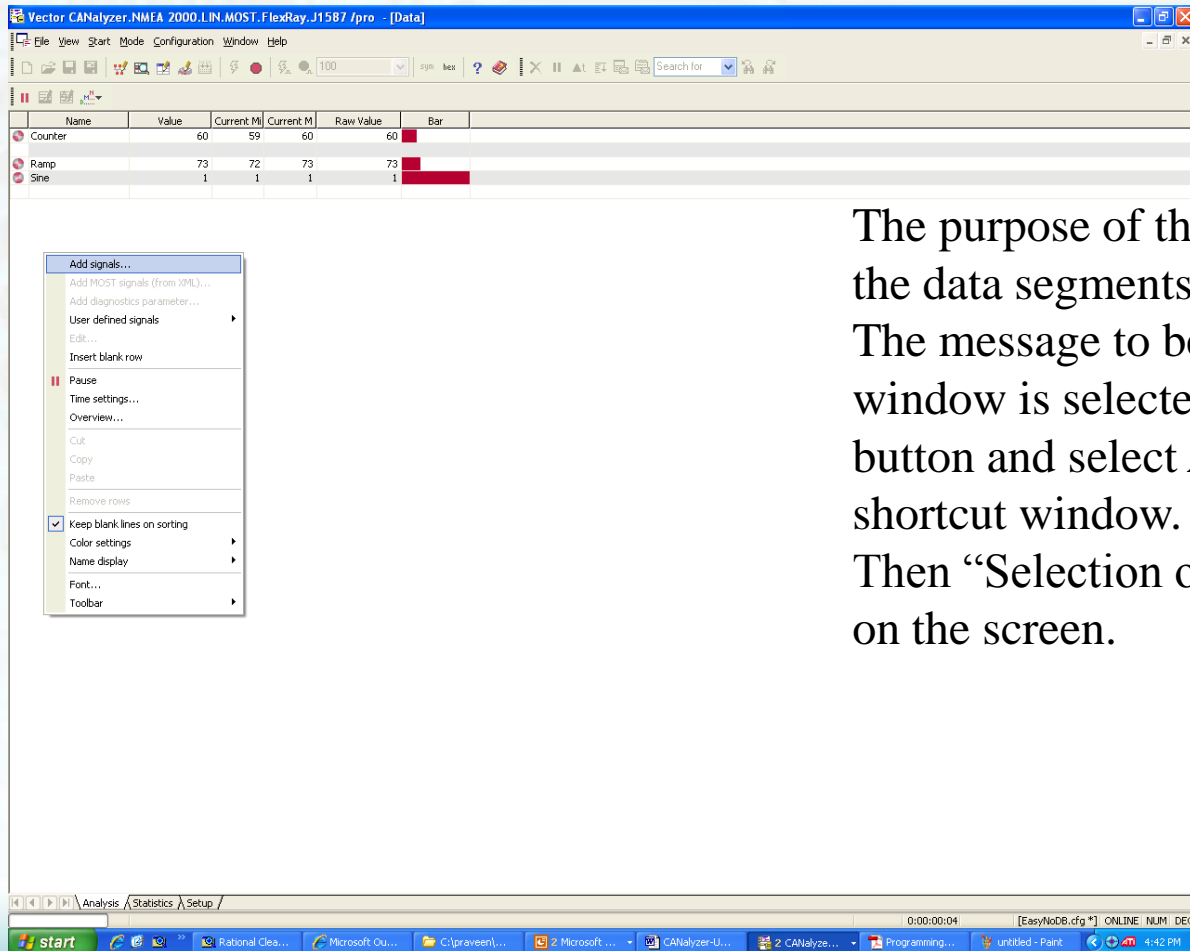
Trace Windows



The screenshot displays the Vector CANalyzer interface. The main window shows a list of CAN messages with columns for Time, Chn, ID, Dir, DLC, and Data. A context menu is open over the message list, showing options like Configuration..., Pin event, Event filtered, Highlight message(s), Pause, Copy to clipboard, Clear All, Mark All, Open detail view, Close detail view, Synchronize windows, Set as reference event, Find reference event, Find event..., Quick find, Import/Export, Font..., and Toolbar. The Trace Window Configuration dialog is also open, showing the Event Highlighting tab with options for Display mode (chronological, chronological cyclic update, fixed position, fixed position cyclic update) and Time (absolute, relative). Other display options include Data bytes in ASCII representation and Insert blank lines at the beginning/end of the signal list.

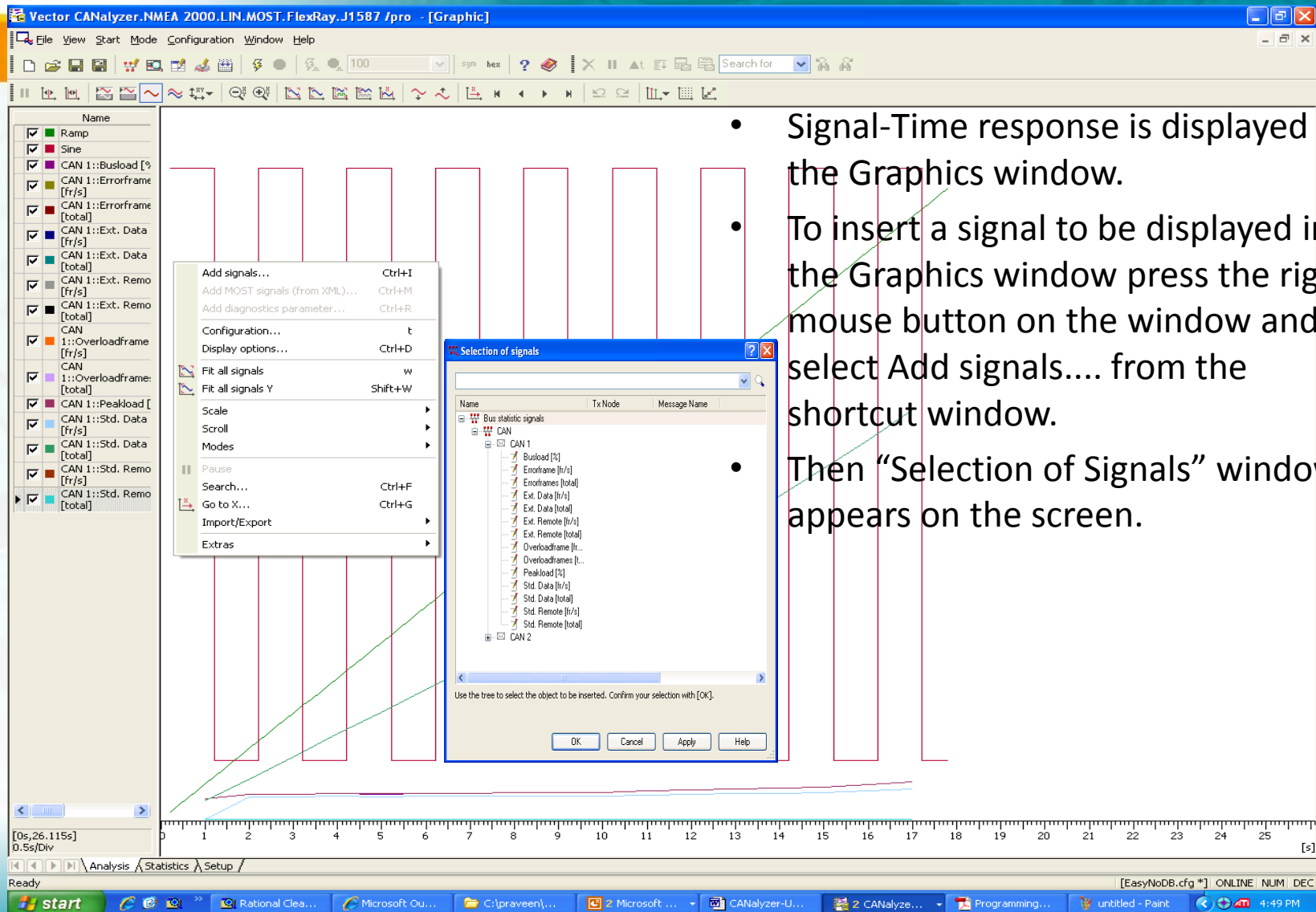
- All messages arriving at the Trace block's input are evaluated and displayed in the Trace Window.
- User can access this window from view menu or either by double click on the Trace window in the Measurement setup window.
- To customize the Trace window click the right mouse button and select the configuration from the menu shortcut window.

Data Window



The purpose of the Data window is to display the data segments (signals) of certain messages. The message to be displayed in the Data window is selected by pressing the right mouse button and select Add signals... from the menu shortcut window. Then “Selection of Signals” window appears on the screen.

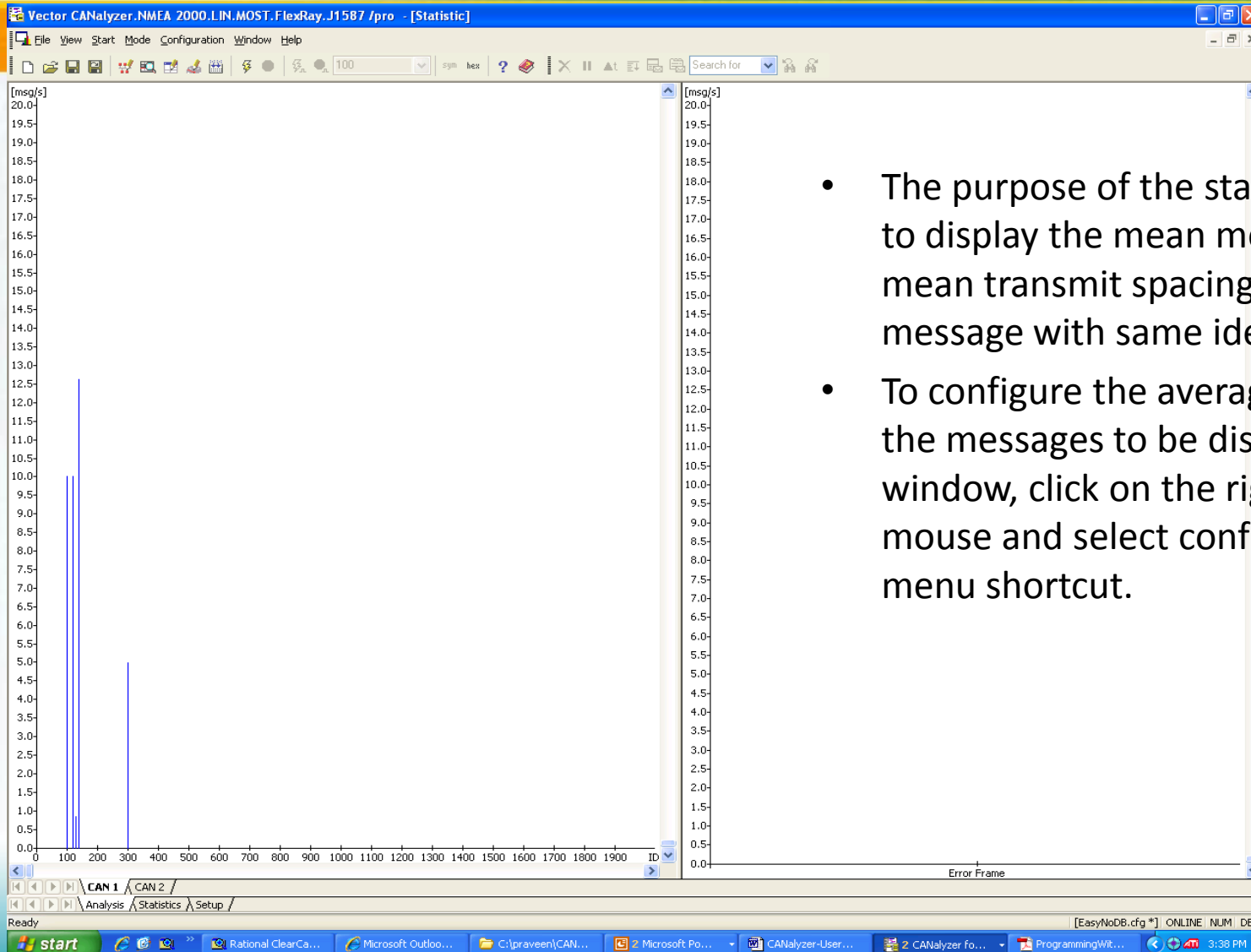
Graphics Window



The screenshot shows the Vector CANalyzer software interface. The main window is titled "Vector CANalyzer.NMEA 2000.LIN.MOST.FlexRay.J1587 /pro - [Graphic]". The left sidebar contains a list of signals, including "Ramp", "Sine", "CAN 1::Busload [%]", "CAN 1::Errorframe [fr/s]", "CAN 1::Errorframe [total]", "CAN 1::Ext. Data [fr/s]", "CAN 1::Ext. Data [total]", "CAN 1::Ext. Remo [fr/s]", "CAN 1::Ext. Remo [total]", "CAN 1::Overloadframe [fr/s]", "CAN 1::Overloadframe: [total]", "CAN 1::Peakload [%]", "CAN 1::Std. Data [fr/s]", "CAN 1::Std. Data [total]", "CAN 1::Std. Remo [fr/s]", and "CAN 1::Std. Remo [total]". The main plot area displays a signal-time response graph with a red signal line and a green trend line. A context menu is open over the plot, showing options like "Add signals...", "Add MOST signals (from XML)...", "Add diagnostics parameter...", "Configuration...", "Display options...", "Fit all signals", "Fit all signals Y", "Scale", "Scroll", "Modes", "Pause", "Search...", "Go to X...", "Import/Export", and "Extras". The "Selection of signals" dialog box is also open, showing a tree view of available signals for selection. The dialog box has a search bar and a list of signals, including "Bus statistic signals", "CAN", "CAN 1", and "CAN 2". The bottom status bar shows "Ready" and "Analysis / Statistics / Setup /".

- Signal-Time response is displayed in the Graphics window.
- To insert a signal to be displayed in the Graphics window press the right mouse button on the window and select Add signals.... from the shortcut window.
- Then "Selection of Signals" window appears on the screen.

Statistics Window



- The purpose of the statistics window is to display the mean message rate or the mean transmit spacing between message with same identifier.
- To configure the average timing between the messages to be displayed on the window, click on the right button of the mouse and select configure from the menu shortcut.

Bus Statistics

Vector CANalyzer.NMEA 2000.LIN.MOST.FlexRay.J1587 /pro - [BusStatistic]

File View Start Mode Configuration Window Help

100 sym hex ?

	CAN 1	CAN 2
Busload [%]	3.80	-
Peakload [%]	3.80	-
Std. Data [fr/s]	33	-
Std. Data [total]	281	-
Ext. Data [fr/s]	0	-
Ext. Data [total]	0	-
Std. Remote [fr/s]	0	-
Std. Remote [to...]	0	-
Ext. Remote [fr/s]	0	-
Ext. Remote [to...]	0	-
Errorframe [fr/s]	0	-
Errorframes [total]	0	-
Chip state	Active	Active

Ready [EasyNoDB.cfg *] ONLINE NUM DEC

start Rational ClearCa... Microsoft Outloo... C:\praveen\CAN... 2 Microsoft Po... CANalyzer-User... 2 CANalyzer fo... ProgrammingWit... 4:26 PM

- Bus Statistics for each CAN controller on the Network is displayed on the Bus Statistics window.
- This window displays the occurrence rate of various CAN frames such as Data, Remote, Error and Overload frames.
- The maximum Bus load since the measurement start is displayed in percent.
- The Error state information is also displayed on this window.

CANdb++

Automotive Robotics (India) Pvt, Ltd.

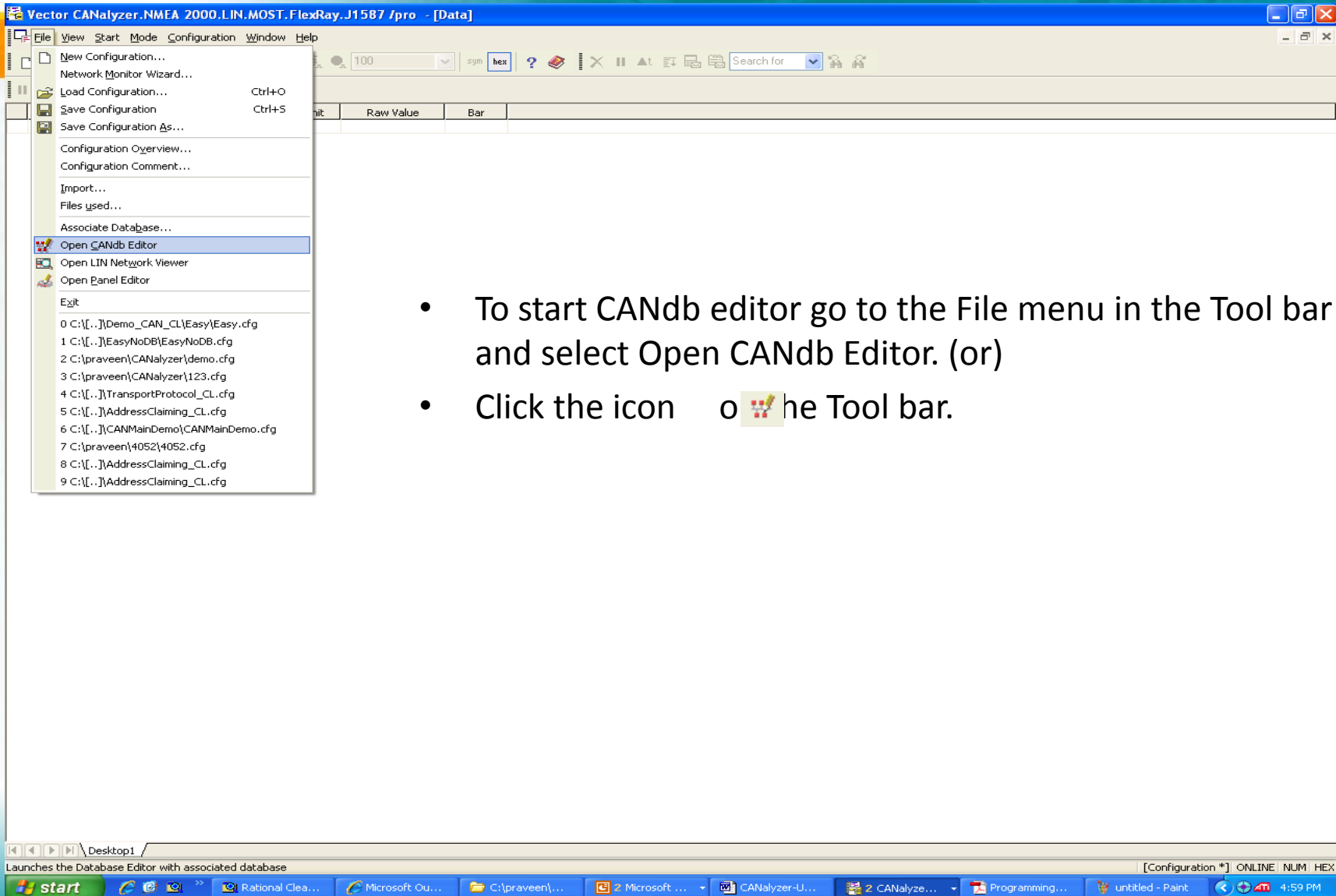
4th Floor | Surya Towers | Secunderabad - 500003 | Phone: 040 - 30280431

web: www.autorobinc.com

What is a Database?

- Database can be said as a blueprint that categorizes bus data in symbolic terms.
- The CANdb++ Editor allows the user to create and modify the bus data in symbolic terms.

Running CANdb Editor




Vector CANalyzer.NMEA 2000.LIN.MOST.FlexRay.J1587 /pro - [Data]

File View Start Mode Configuration Window Help

- New Configuration...
- Network Monitor Wizard...
- Load Configuration... Ctrl+O
- Save Configuration Ctrl+S
- Save Configuration As...
- Configuration Overview...
- Configuration Comment...
- Import...
- Files used...
- Associate Database...
- Open CANdb Editor
- Open LIN Network Viewer
- Open Panel Editor
- Exit
- 0 C:\[...]\Demo_CAN_CL\Easy\Easy.cfg
- 1 C:\[...]\EasyNoDB\EasyNoDB.cfg
- 2 C:\praveen\CANalyzer\demo.cfg
- 3 C:\praveen\CANalyzer\123.cfg
- 4 C:\[...]\TransportProtocol_CL.cfg
- 5 C:\[...]\AddressClaiming_CL.cfg
- 6 C:\[...]\CANMainDemo\CANMainDemo.cfg
- 7 C:\praveen\4052\4052.cfg
- 8 C:\[...]\AddressClaiming_CL.cfg
- 9 C:\[...]\AddressClaiming_CL.cfg

100 sym hex ? X || ▲ ▼ ↺ ↻ Search for

Bit	Raw Value	Bar

- To start CANdb editor go to the File menu in the Tool bar and select Open CANdb Editor. (or)
- Click the icon  the Tool bar.

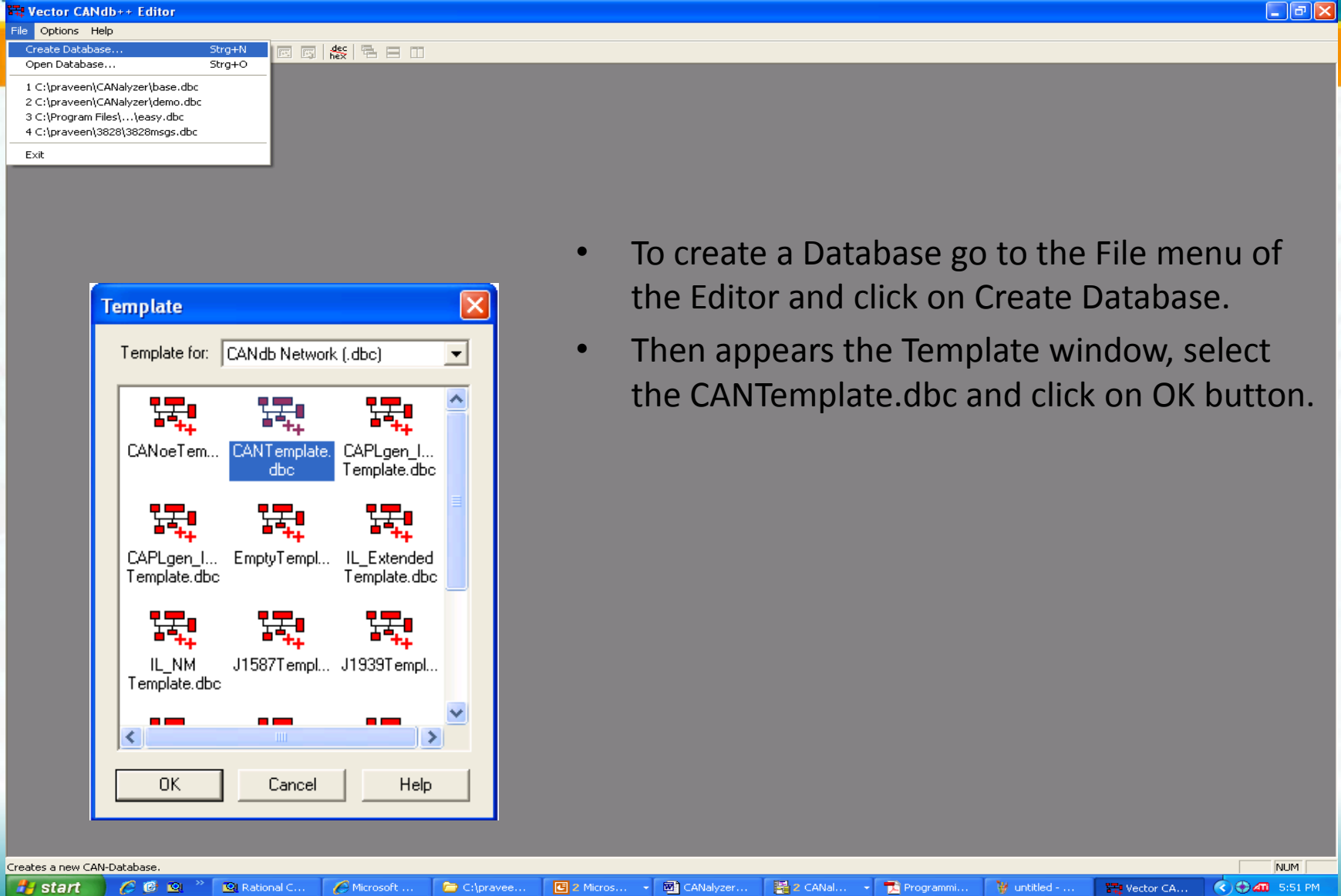
Desktop1

Launches the Database Editor with associated database

[Configuration *] ONLINE NUM HEX

start Rational Clea... Microsoft Ou... C:\praveen\... 2 Microsoft ... CANalyzer-U... 2 CANalyze... Programming... untitled - Paint 4:59 PM

Creating the Database



The screenshot shows the Vector CANdb++ Editor application. The 'File' menu is open, displaying options: 'Create Database...' (Strg+N), 'Open Database...' (Strg+O), a list of recent databases, and 'Exit'. The 'Template' dialog box is also open, showing a list of templates for 'CANdb Network (.dbc)'. The 'CANTemplate.dbc' is selected. The dialog box includes buttons for 'OK', 'Cancel', and 'Help'.

- To create a Database go to the File menu of the Editor and click on Create Database.
- Then appears the Template window, select the CANTemplate.dbc and click on OK button.

Database Organization

Vector CANdb++ Editor - base.dbc - [Overall View]

File Edit View Options Window Help

dec hex

Networks
base
Network nodes
Messages
Signals

Name	Address	Comment
------	---------	---------

CANdb Database

```

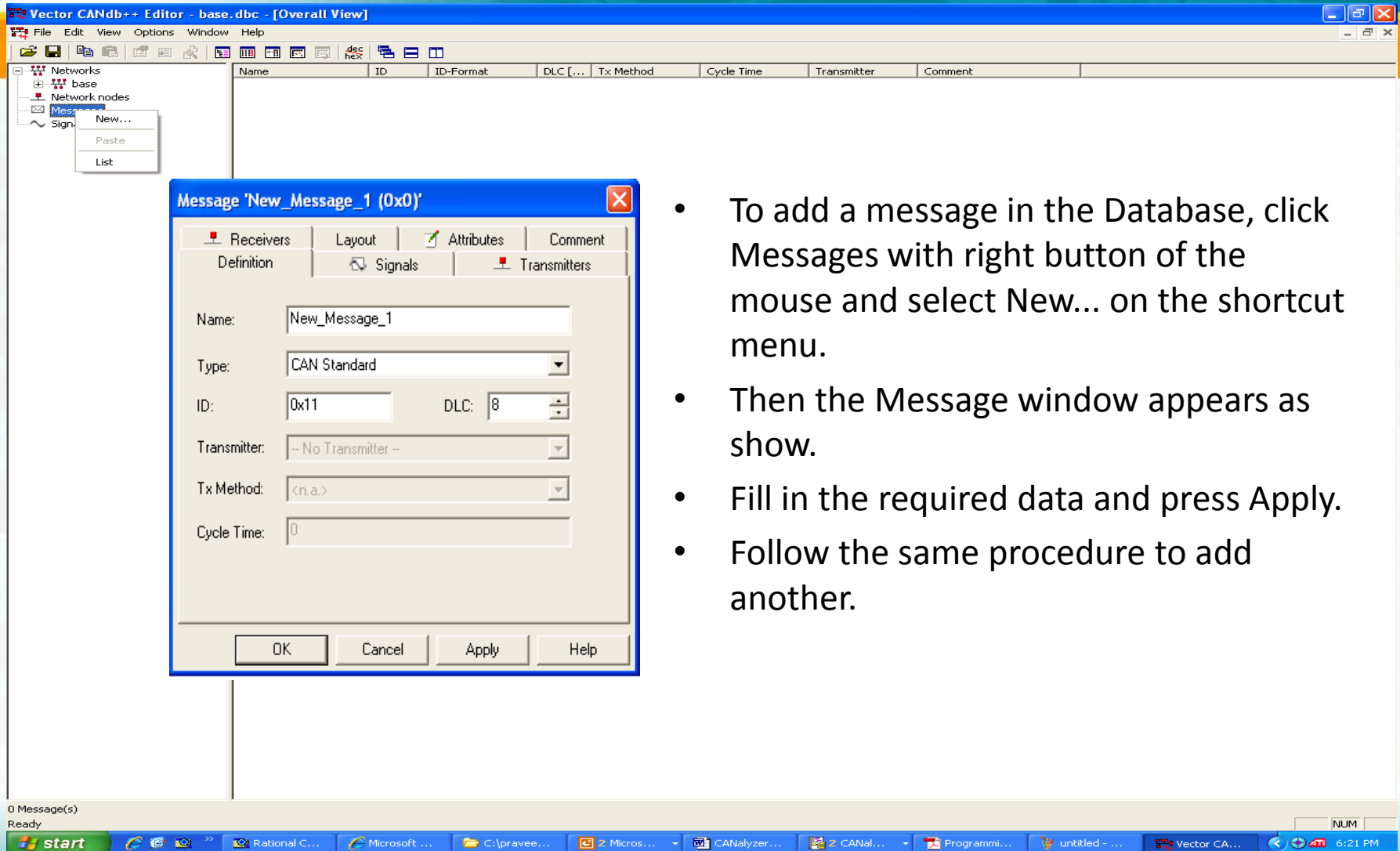
graph TD
    Root[CANdb Database] --- Attributes[Attributes]
    Root --- Network[Network]
    Root --- ValueTables[Value Tables]
    Root --- EnvVars[Environment Variables (CANoe Only)]
    Root --- Node1[Node]
    Root --- Node2[Node]
    Node1 --- M1[Message]
    Node1 --- M2[Message]
    Node2 --- M3[Message]
    Node2 --- M4[Message]
    M1 --- S1[Signal]
    M1 --- S2[Signal]
    M2 --- S3[Signal]
    M2 --- S4[Signal]
    M3 --- S5[Signal]
    M3 --- S6[Signal]
    M4 --- S7[Signal]
    M4 --- S8[Signal]
  
```

0 Node(s)
Ready

start Rational C... Microsoft ... C:\pravee... Z Micros... CANalyzer... Z CANAl... Programmi... untitled - ... Vector CA... NUM 6:08 PM

- Network: A CAN bus network that contains multiple ECUs to exchange data.
- Nodes: Network interfaces of an ECU used to transmit and receive data from the CAN bus.
- Messages: Containers of information that are transmitted on the CAN bus.
- Signals: Independent data regions in the data field of a message.

Add a Message to Database

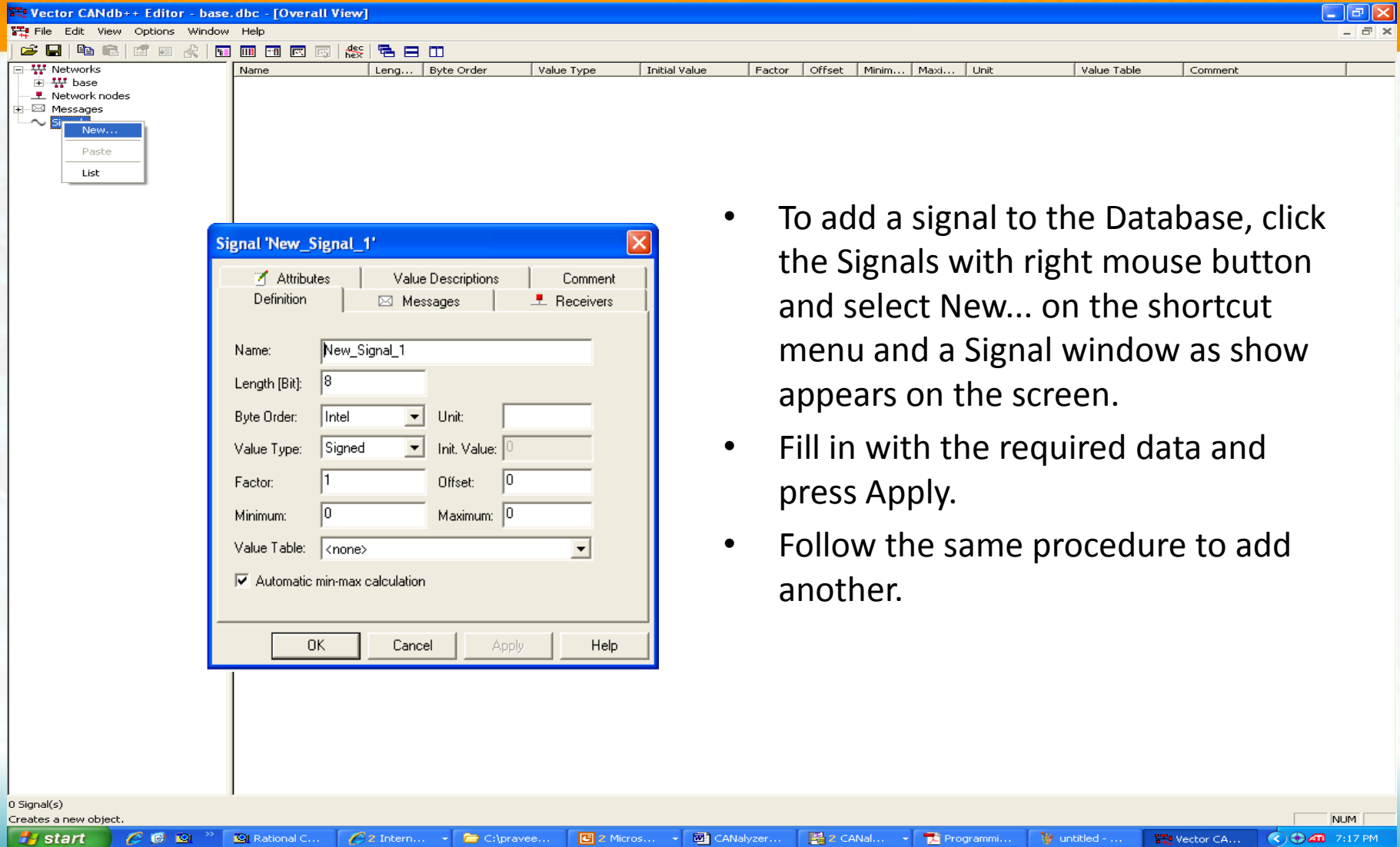


The screenshot shows the Vector CANdb++ Editor interface. A context menu is open over the 'Messages' folder in the left sidebar, with 'New...' selected. A dialog box titled 'Message 'New_Message_1 (0x0)'' is displayed in the foreground. The dialog has tabs for 'Receivers', 'Layout', 'Attributes', and 'Comment'. The 'Definition' sub-tab is active, showing fields for Name, Type, ID, DLC, Transmitter, Tx Method, and Cycle Time. The values are: Name: New_Message_1, Type: CAN Standard, ID: 0x11, DLC: 8, Transmitter: -- No Transmitter --, Tx Method: <n.a.>, and Cycle Time: 0. At the bottom are buttons for OK, Cancel, Apply, and Help.

Name	ID	ID-Format	DLC	Tx Method	Cycle Time	Transmitter	Comment
New_Message_1	0x11		8	<n.a.>	0	-- No Transmitter --	

- To add a message in the Database, click Messages with right button of the mouse and select New... on the shortcut menu.
- Then the Message window appears as show.
- Fill in the required data and press Apply.
- Follow the same procedure to add another.

Add a Signal to the Database



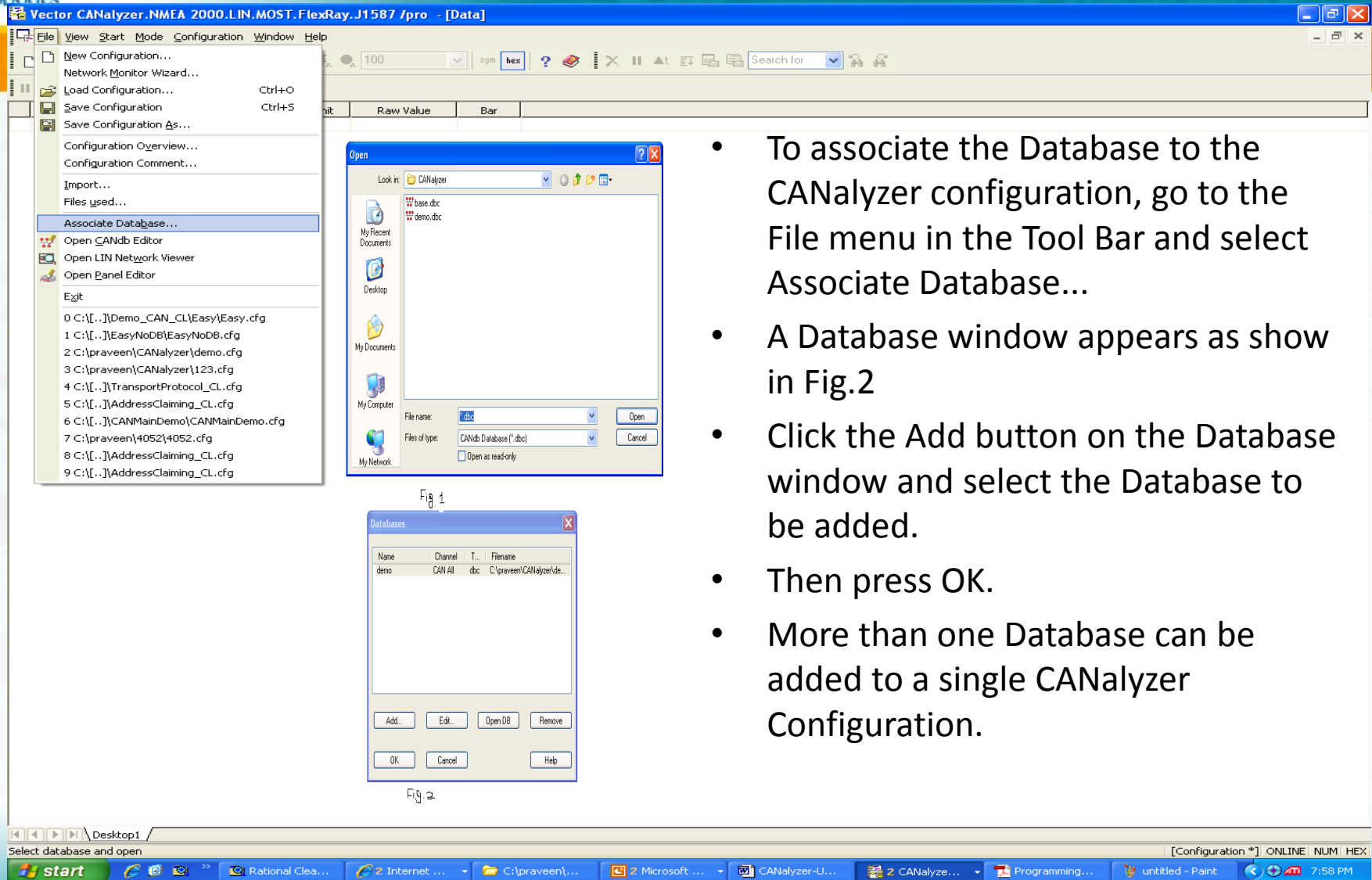
The screenshot shows the Vector CANdb++ Editor interface. The 'Signals' folder in the left sidebar is right-clicked, and the 'New...' option is selected from the context menu. This opens the 'Signal 'New_Signal_1'' dialog box. The dialog has tabs for 'Attributes', 'Value Descriptions', and 'Comment'. The 'Attributes' tab is active, showing fields for Name, Length [Bit], Byte Order, Value Type, Factor, Minimum, Maximum, and Value Table. The 'Automatic min-max calculation' checkbox is checked. The 'Messages' and 'Receivers' tabs are also visible.

0 Signal(s)
Creates a new object.

start | Rational C... | 2 Intern... | C:\pravee... | 2 Micros... | CANalyzer... | 2 CANal... | Programmi... | untitled - ... | Vector CA... | 7:17 PM

- To add a signal to the Database, click the Signals with right mouse button and select New... on the shortcut menu and a Signal window as show appears on the screen.
- Fill in with the required data and press Apply.
- Follow the same procedure to add another.

Associate Database



The screenshot shows the Vector CANalyzer software interface. The 'File' menu is open, and 'Associate Database...' is selected. This opens the 'Open' dialog box, which shows the file list with 'demo.dbc' selected. The 'File name' field is empty, and the 'Files of type' dropdown is set to 'CANdb Database (*.dbc)'. The 'Open' button is highlighted. Below the 'Open' dialog is the 'Databases' dialog box, which contains a table with the following data:

Name	Channel	T...	Filename
demo	CAN All	dbc	C:\praveen\CANalyzer\de...

Below the table are buttons for 'Add...', 'Edit...', 'Open DB', 'Remove', 'OK', 'Cancel', and 'Help'.

- To associate the Database to the CANalyzer configuration, go to the File menu in the Tool Bar and select Associate Database...
- A Database window appears as show in Fig.2
- Click the Add button on the Database window and select the Database to be added.
- Then press OK.
- More than one Database can be added to a single CANalyzer Configuration.

CAPL

Automotive Robotics (India) Pvt, Ltd.

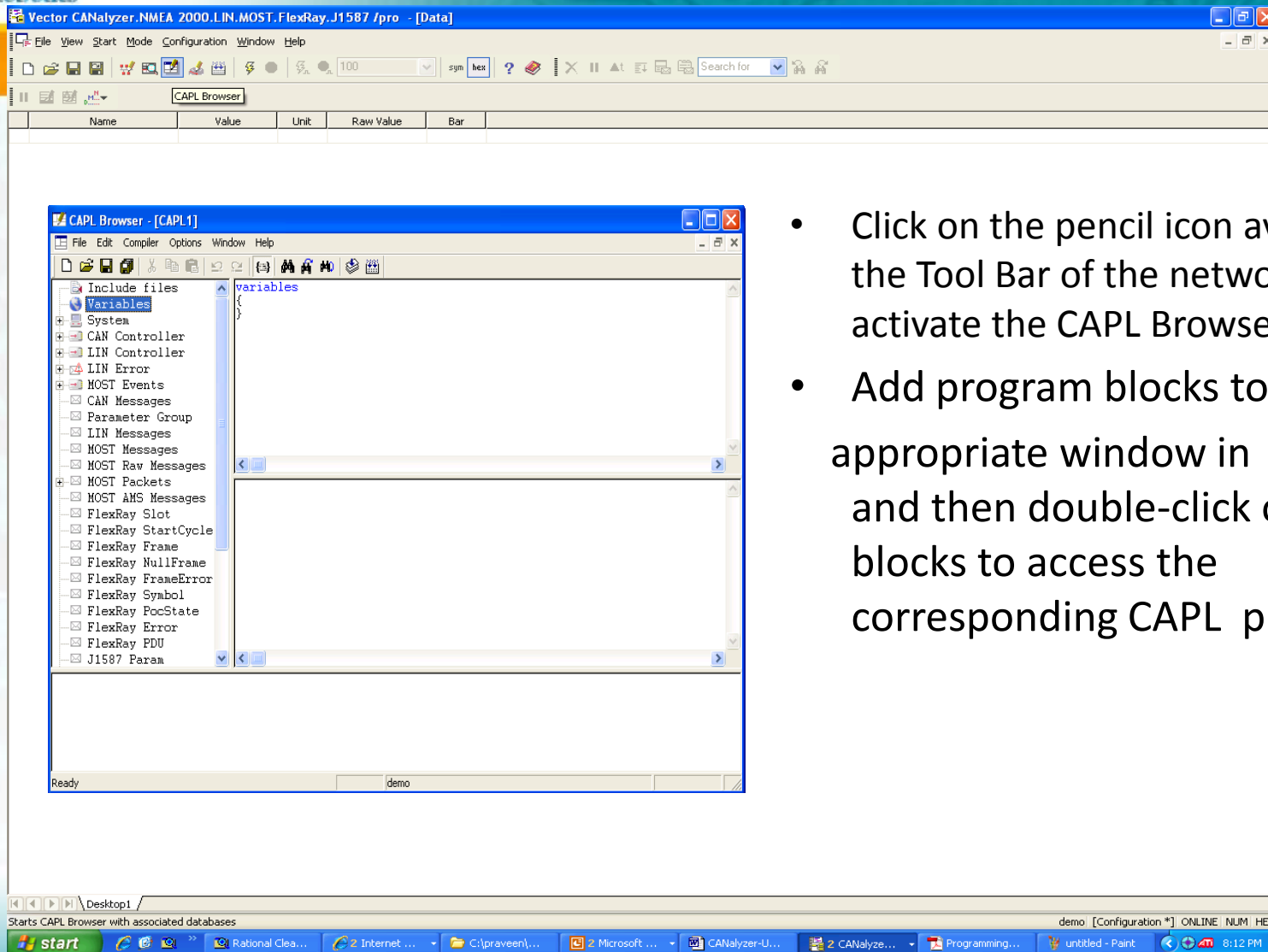
4th Floor | Surya Towers | Secunderabad - 500003 | Phone: 040 - 30280431

web: www.autorobinc.com

Introduction

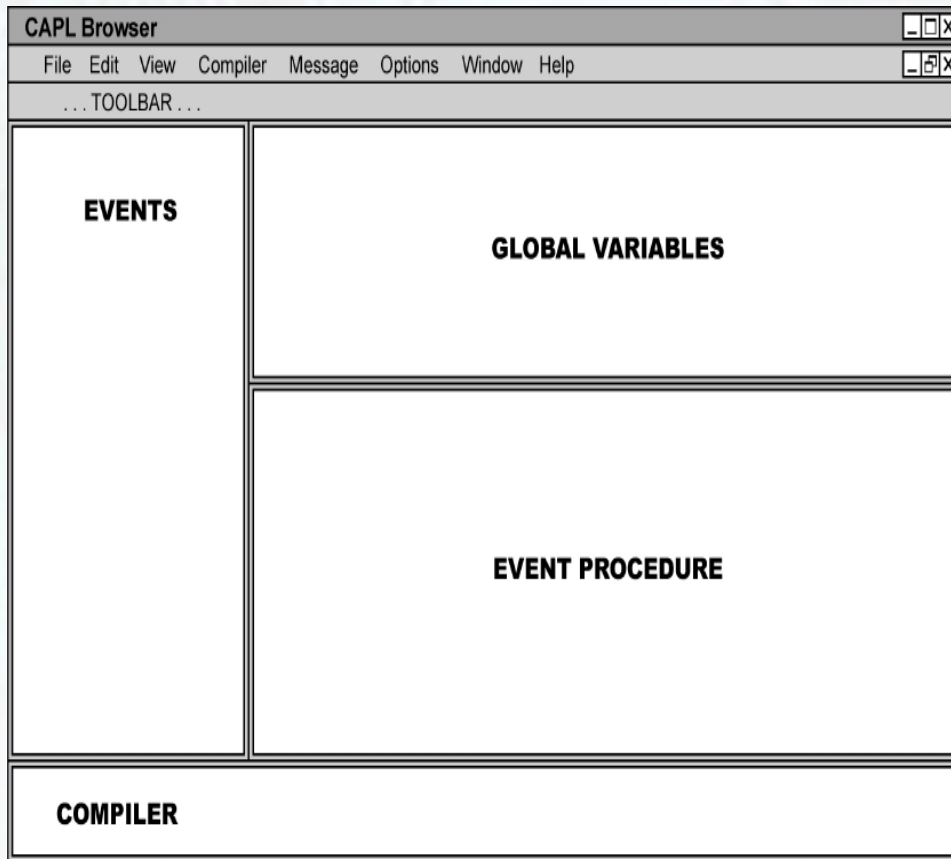
- CAN Access Programming Language is intended to meet the CAN-based Distributed Embedded system developers requirements such as:
 - Maximum control of all test and measurement operations.
 - Maximum support for one or more communication channels.
 - Maximum event and message recording and playback control.
- CAPL can be used in the application to:
 - Analyze specific messages or specific data.
 - Analyze data traffic.
 - Create and modify the tool's measurement environment.
- The CAPL Browser is the fundamental programming environment used to develop all CAPL programs.

Starting CAPL Browser



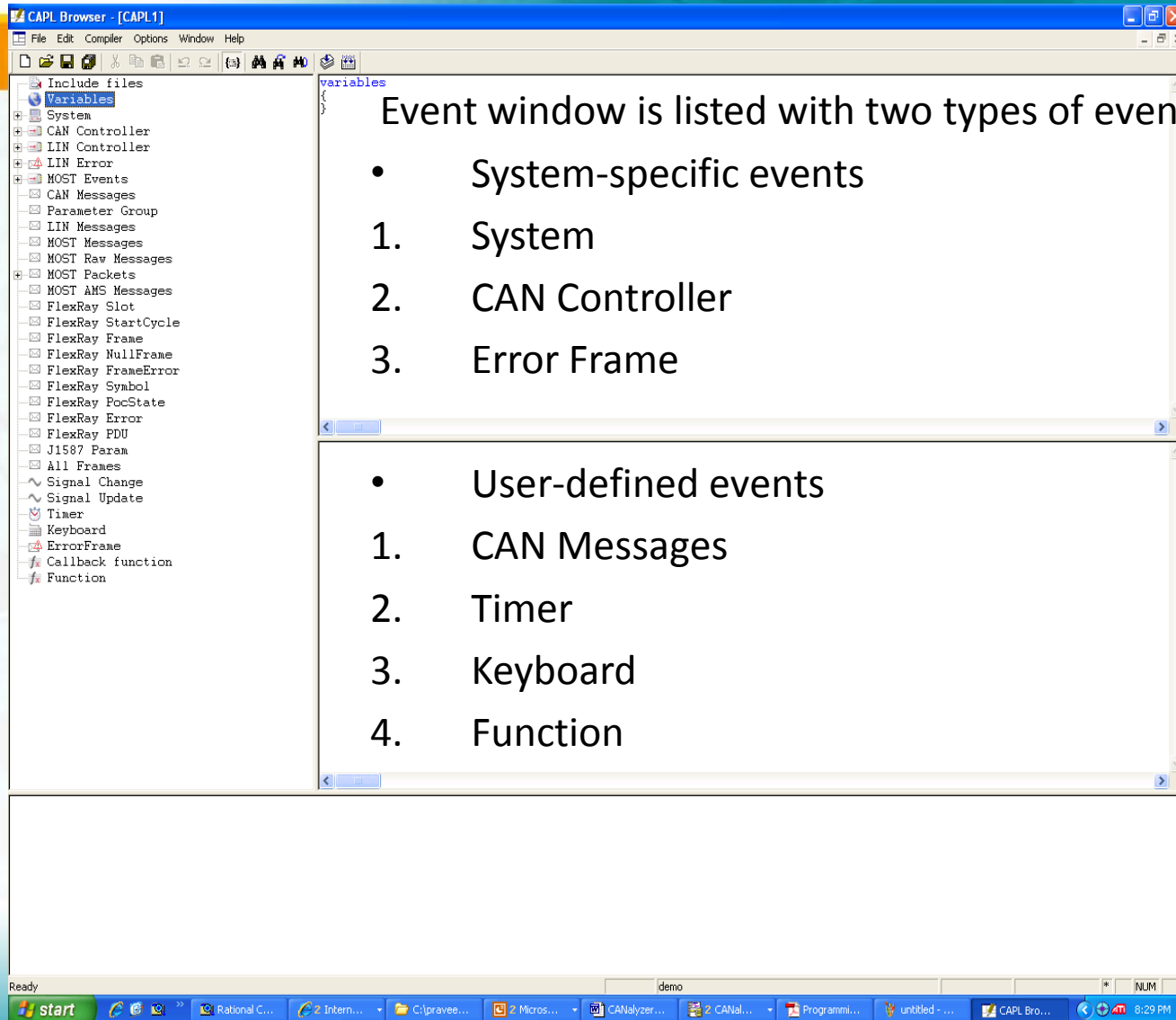
- Click on the pencil icon available on the Tool Bar of the network node will activate the CAPL Browser. (or)
- Add program blocks to the appropriate window in CANalyzer and then double-click on the blocks to access the corresponding CAPL program.

Browser Organization



- **Events** window, lists both the different types of CAPL events and the names of the procedures associated with each event category in a tree view.
- **Global Variables** window, is used to declare all necessary global variables for the CAPL program.
- **Event Procedure** window, displays the source code entered for the selected procedure highlighted in the **Events** window.
- **Compiler** window, shows compiler activities and results.

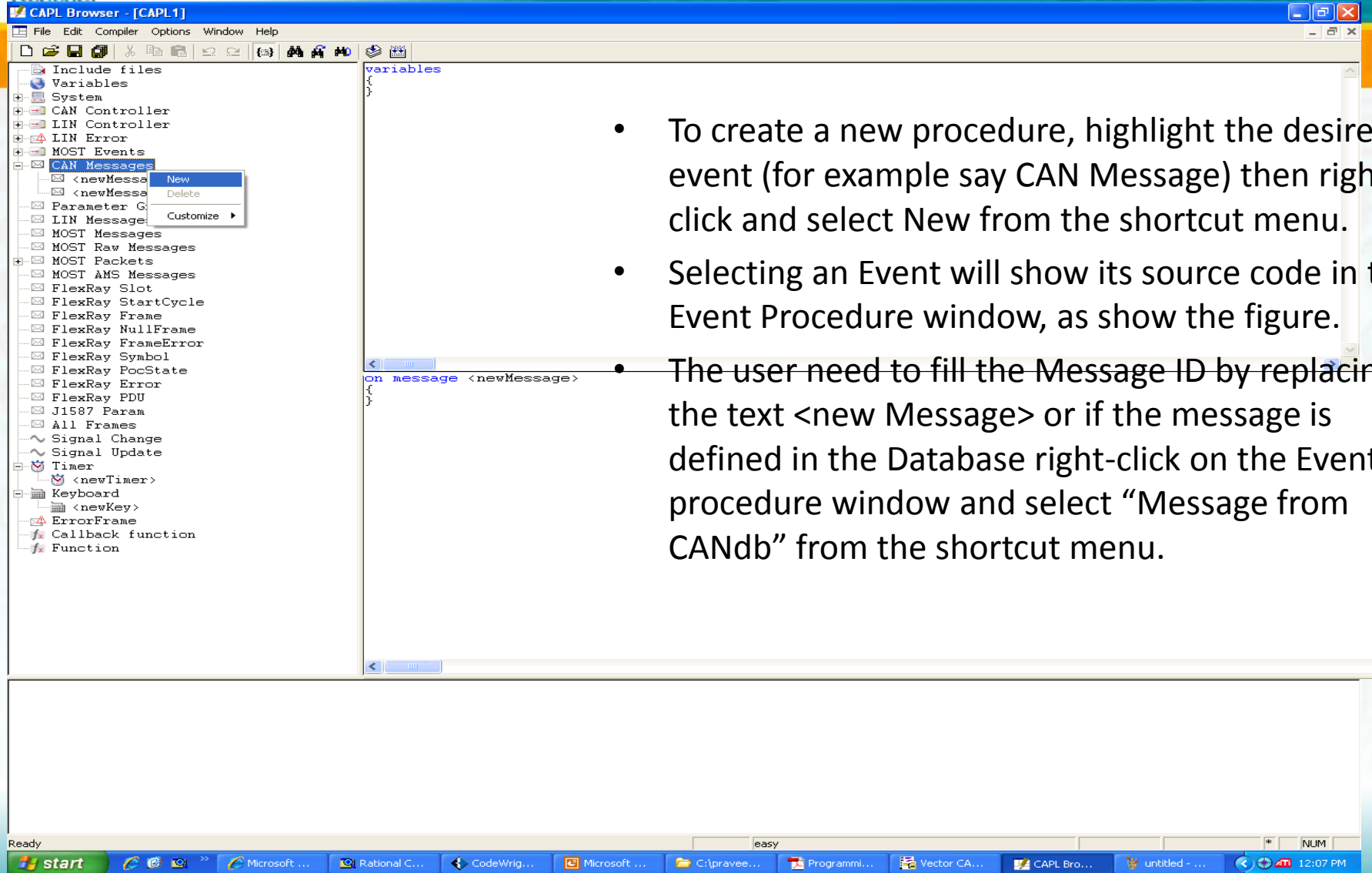
Event Window



The screenshot shows the CAPL Browser window with the following structure:

- Include files**
 - Variables**
 - System
 - System
 - CAN Controller
 - Error Frame
 - User-defined events
 - CAN Messages
 - Timer
 - Keyboard
 - Function

Creating an Event Procedure



The screenshot shows the CAPL Browser interface. On the left, a tree view lists various event categories. The 'CAN Messages' category is expanded, and a right-click context menu is open, showing options: 'New', 'Delete', and 'Customize'. The 'New' option is highlighted. The main window displays the source code for the selected event procedure, which is an 'on message' event. The code is as follows:

```
variables
{
}

on message <newMessage>
{
}
```

The Windows taskbar at the bottom shows several open applications, including Microsoft Word, Rational C++, CodeWrig..., Microsoft..., C:\pravee..., Programmi..., Vector CA..., CAPL Bro..., and untitled - ... The system clock shows 12:07 PM.

- To create a new procedure, highlight the desired event (for example say CAN Message) then right click and select New from the shortcut menu.
- Selecting an Event will show its source code in the Event Procedure window, as show the figure.
- The user need to fill the Message ID by replacing the text <new Message> or if the message is defined in the Database right-click on the Event procedure window and select “Message from CANdb” from the shortcut menu.

CAPL Keywords

Supported:

break
case
char
continue
default
do
double
else
float
for
if
int
long
return
switch
while

Not supported:

auto
const
enum
extern
goto
register
short
signed
sizeof
static
struct
typedef
union
unsigned
volatile

CAPL Data Types

Data Types supported by CAPL programming are as show below:

Data Type	Description	Size	Unsigned/Signed
char	character	8 bit	unsigned
byte	byte	8 bit	unsigned
int	integer	16 bit	signed
word	word	16 bit	unsigned
long	long integer	32 bit	signed
dword	double word	32 bit	unsigned
float	single precision floating point	64 bit ¹	signed
double	single precision floating point	64 bit	signed
message	a communication message	--	--
timer	a timer with second resolution	--	--
msTimer	a timer with millisecond resolution	--	--

Note: If a variable is assigned during declaration, then they are considered as constant variables.

Operators

As with C, CAPL also supports a wealth of operators to process information, such as:

- Arithmetic Operators
- Assignment Operators.
- Relational Operators.
- Boolean Operators.
- Bitwise Operators.
- Miscellaneous Operators.

Unsupported Operators are as below:

Symbol	Operation	NOT ALLOWED IN CAPL	Example
&	Address	&variable Address of variable	x = &y
#	Preprocessing	#keyword Preprocessing directives	#define ...
*	Address	*pointer_expression Address of variable	x = *y
->	Member	structure_pointer_expression -> structure_member Member structure_member of structure pointed to by structure_pointer_expression	x = y -> z

Automotive Robotics (India) Pvt, Ltd.

4th Floor | Surya Towers | Secunderabad - 500003 | Phone: 040 - 30280431

web: www.autorobinc.com

Messages in CAPL

- Messages in CAPL are objects, hence they should be declared similar to variables in C.

Syntax:

message <message identifier or name > <variable name>

Example:

```
message 0x100 Temp;
```

- Fields of the Message can be accessed using a "." operator.

Example: Individual byte of the Data field in the message can be accessed as:

```
message.BYTE(0) = 0x11;
```

```
message.BYTE(1) = 0x12;
```

```
message.LONG(2) = 0x51FF016E;
```


Functions in CAPL

Pre-defined API's in CAPL are organized in categories:

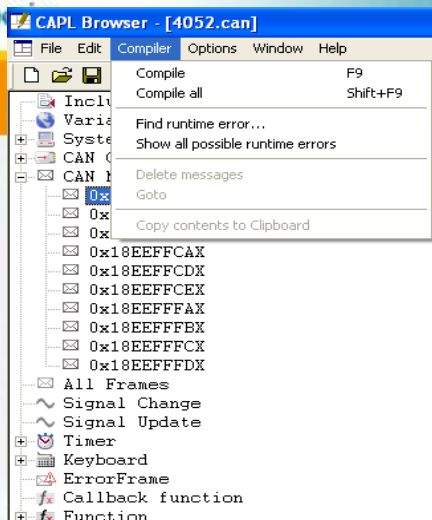
- Mathematical functions (`sin()`, `sqrt()`).
- User-interface function (`beep()`, `Write()`).
- Time and Timer functions (`setTimer()`, `cancelTimer()`).
- Message handling functions (`output()`).
- Database functions (`getMessageName()`, `getNextCANdbName()`).
- Byte ordering functions (`swapInt()`, `swapLong()`).
- Logging functions (`setLogFileName()`, `startLogging()`).
- String Handling functions (`strlen()`, `strncmp()`).
- CAN Protocol functions (`resetCAN()`, `setBtr()`).
- Port functions (`inport()`, `outport()`).

Timer



Timer is used to trigger periodic events. In order to use a Timer in programming it should be

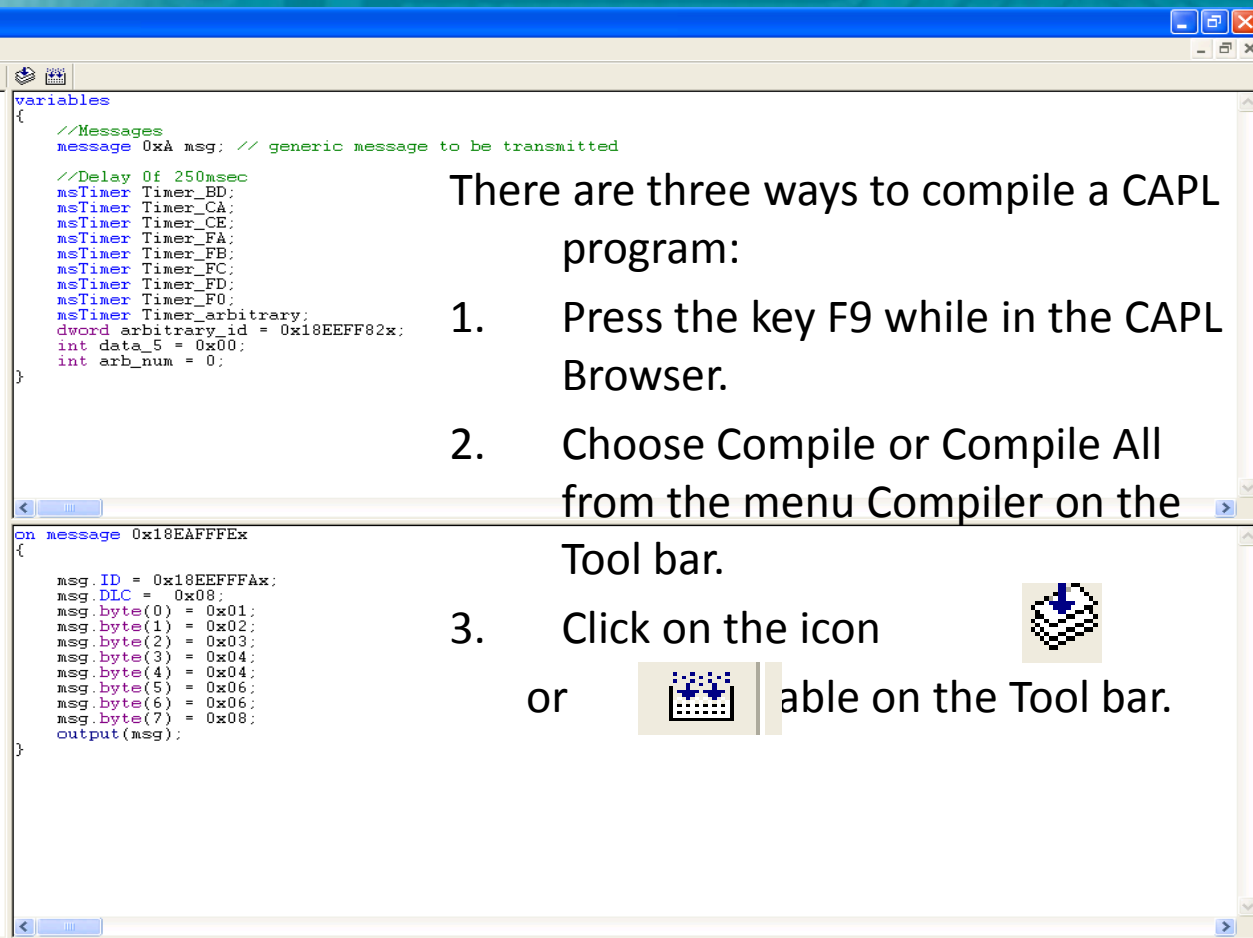
1. Declared:
 - If the unit of timer is in seconds then use **Timer** to declare a variable in the Global variable window.
 - If the unit of timer is in mseconds then use **msTimer** to declare a variable in the Global variable window.
2. Set the Timer:
 - To start a Timer user should invoke the function **setTimer()**.
3. Define an on timer event:
 - when a timer or clock is expired, the corresponding timer will be executed.

Compiling CAPL Code



There are three ways to compile a CAPL program:

1. Press the key F9 while in the CAPL Browser.
2. Choose Compile or Compile All from the menu Compiler on the Tool bar.
3. Click on the icon  or  able on the Tool bar.



```

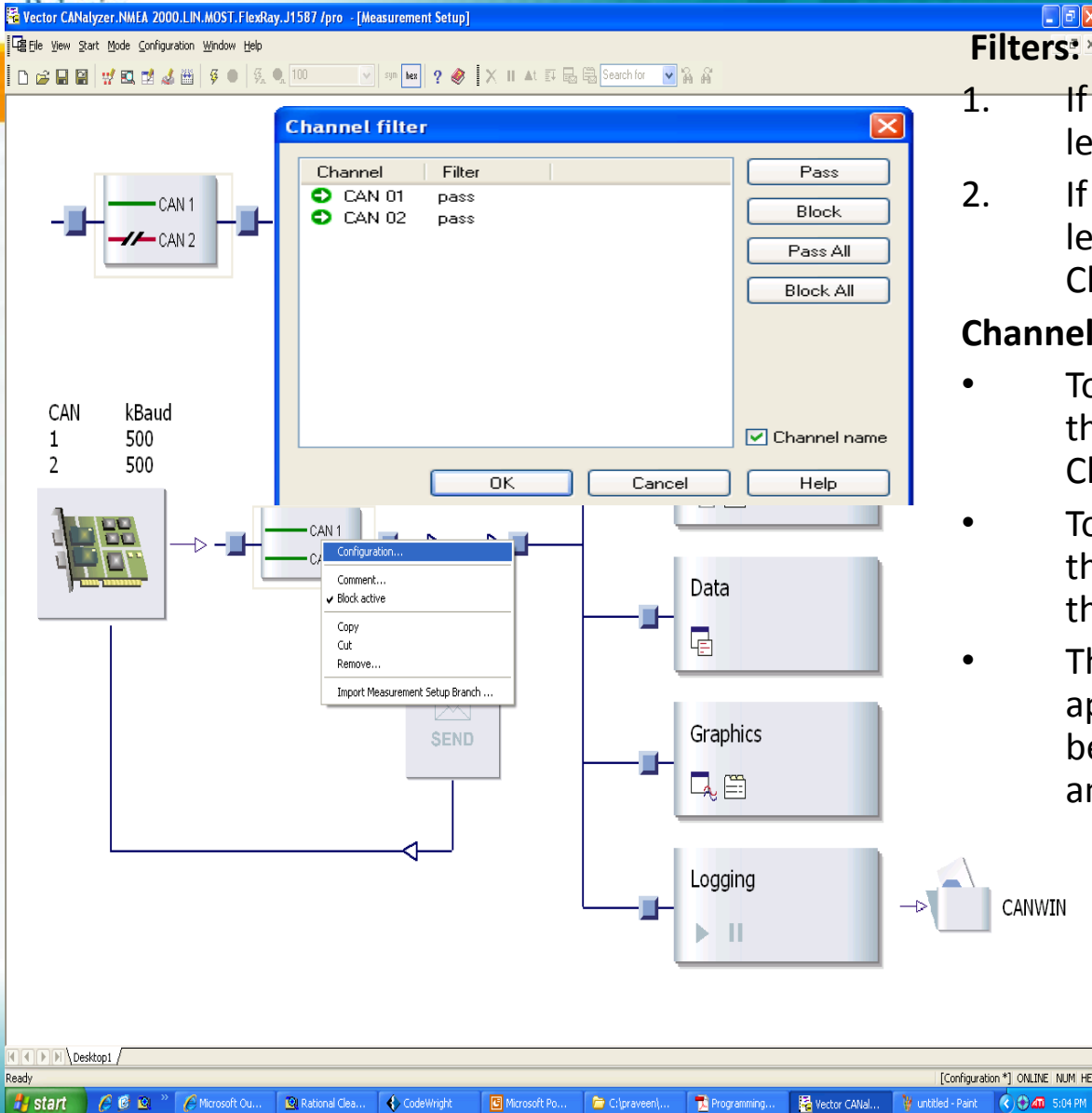
variables
{
    //Messages
    message 0xA msg; // generic message to be transmitted

    //Delay Of 250msec
    nsTimer Timer_BD;
    nsTimer Timer_CA;
    nsTimer Timer_CE;
    nsTimer Timer_FA;
    nsTimer Timer_FB;
    nsTimer Timer_FC;
    nsTimer Timer_FD;
    nsTimer Timer_F0;
    nsTimer Timer_arbitrary;
    dword arbitrary_id = 0x18EEFF82x;
    int data_5 = 0x00;
    int arb_num = 0;
}

on message 0x18EAFFFEx
{
    msg.ID = 0x18EEFFFAx;
    msg.DLC = 0x08;
    msg.byte(0) = 0x01;
    msg.byte(1) = 0x02;
    msg.byte(2) = 0x03;
    msg.byte(3) = 0x04;
    msg.byte(4) = 0x04;
    msg.byte(5) = 0x06;
    msg.byte(6) = 0x06;
    msg.byte(7) = 0x08;
    output(msg);
}
        
```

web: www.autorobinc.com

Function Blocks



The screenshot shows the Vector CANalyzer interface. A measurement setup is visible with two CAN channels (CAN 1 and CAN 2) connected to a hardware device. A context menu is open over the CAN 1 block, showing options like 'Configuration...', 'Block active', 'Copy', 'Cut', 'Remove...', and 'Import Measurement Setup Branch...'. The 'Channel filter' dialog box is also open, displaying a table with the following data:

Channel	Filter
CAN 01	pass
CAN 02	pass

The dialog box also includes buttons for 'Pass', 'Block', 'Pass All', and 'Block All', and a checkbox for 'Channel name' which is checked. The main workspace shows a signal flow from the hardware device through various function blocks (Data, Graphics, Logging) to a CANWIN output.

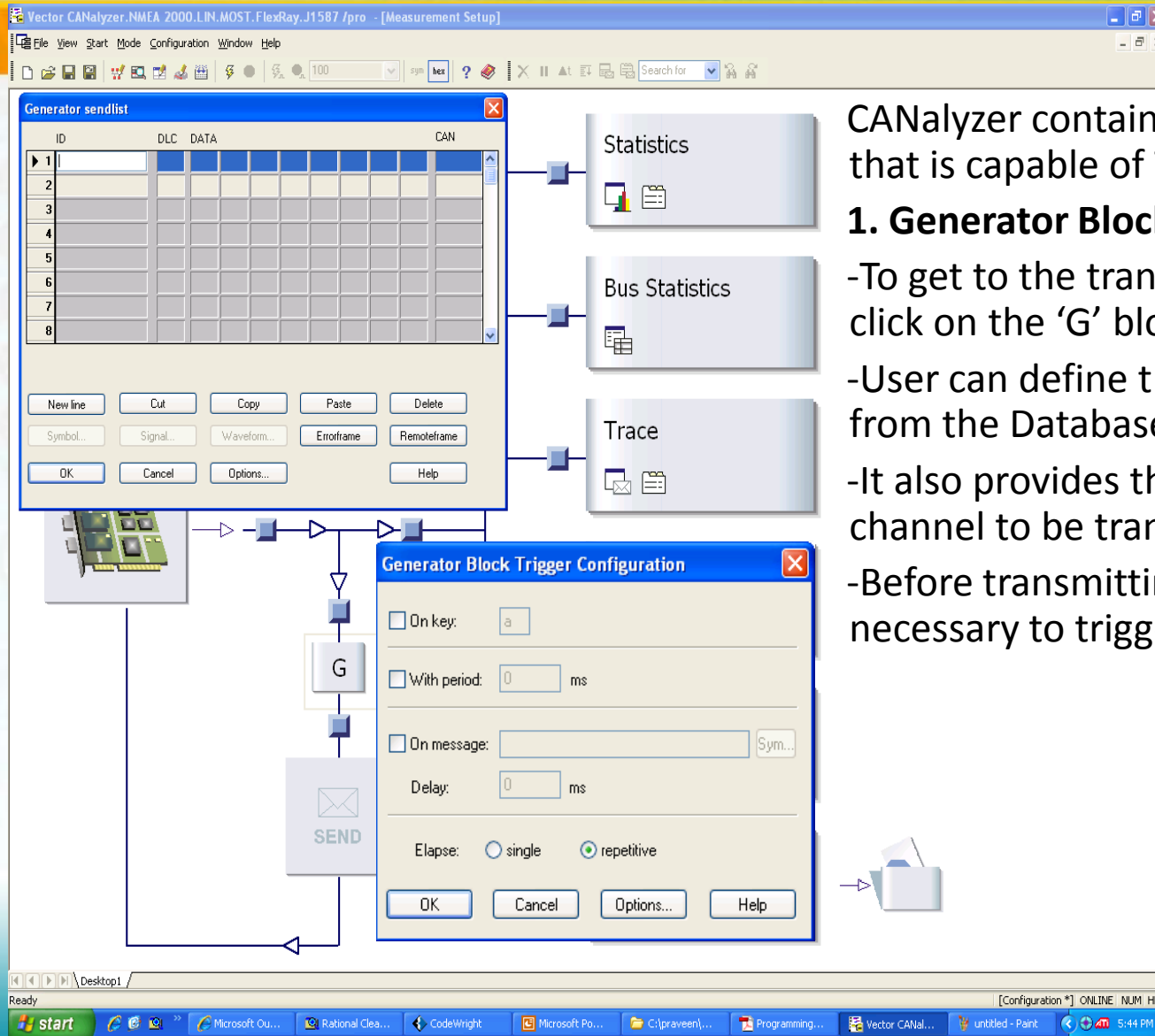
Filters:

1. If filtering is to be done at message level then go for Filter Block.
2. If filtering is to be done at Channel level then we have to choose Channel Filter Block.

Channel Filter:

- To insert a filter block, right-click on the Hotspot and select Insert Channel Filter from the menu.
- To configure the filter, right-click on the block and select configure... from the menu.
- The window named Channel Filter appears, then select the Channel to be blocked and click Block button and then click OK.

Message Blocks



The screenshot displays the Vector CANalyzer software interface. The main window shows a project setup for 'Vector CANalyzer.NMEA 2000.LIN.MOST.FlexRay.J1587 /pro - [Measurement Setup]'. The 'Generator sendlist' window is open, showing a table with columns for ID, DLC, DATA, and CAN. Below the table are buttons for 'New line', 'Cut', 'Copy', 'Paste', 'Delete', 'Symbol...', 'Signal...', 'Waveform...', 'Errorframe', 'Remoteframe', 'OK', 'Cancel', 'Options...', and 'Help'. To the right of the table are three stacked windows: 'Statistics', 'Bus Statistics', and 'Trace'. Below the 'Generator sendlist' window is a 'Generator Block Trigger Configuration' window. This window has checkboxes for 'On key:', 'With period:', and 'On message:', each with associated input fields. It also has a 'Delay:' field and an 'Elapse:' section with 'single' and 'repetitive' radio buttons. At the bottom are 'OK', 'Cancel', 'Options...', and 'Help' buttons. The main workspace shows a block diagram with a 'G' block (Generator) and a 'SEND' block (Transmitter) connected by a line. The status bar at the bottom shows 'Ready' and various system icons.

CANalyzer contains four functional blocks that is capable of Transmitting messages.

1. Generator Block(G):

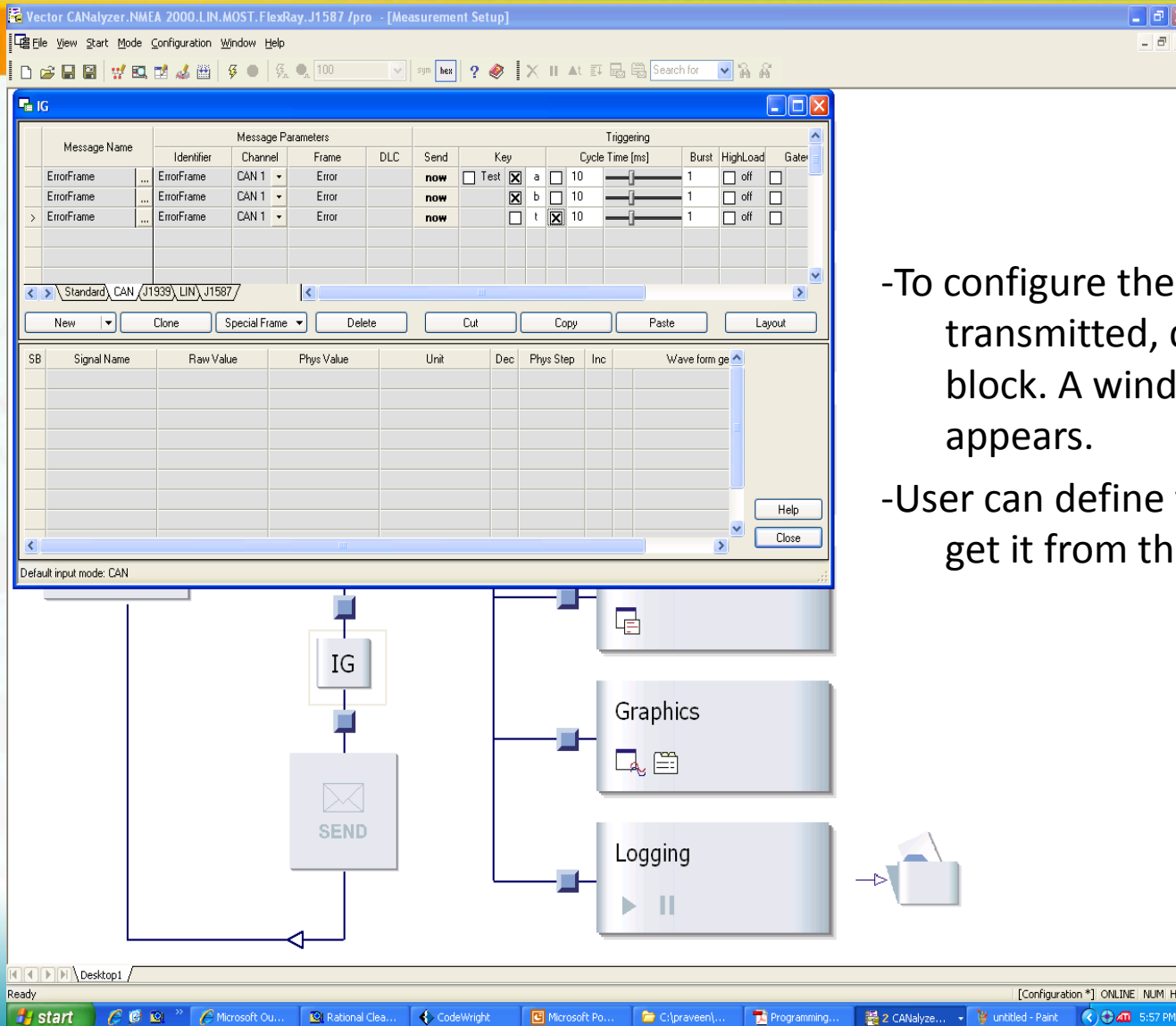
- To get to the transmission block, double-click on the 'G' block.

- User can define the message or select from the Database.

- It also provides the option to select the channel to be transmitted.

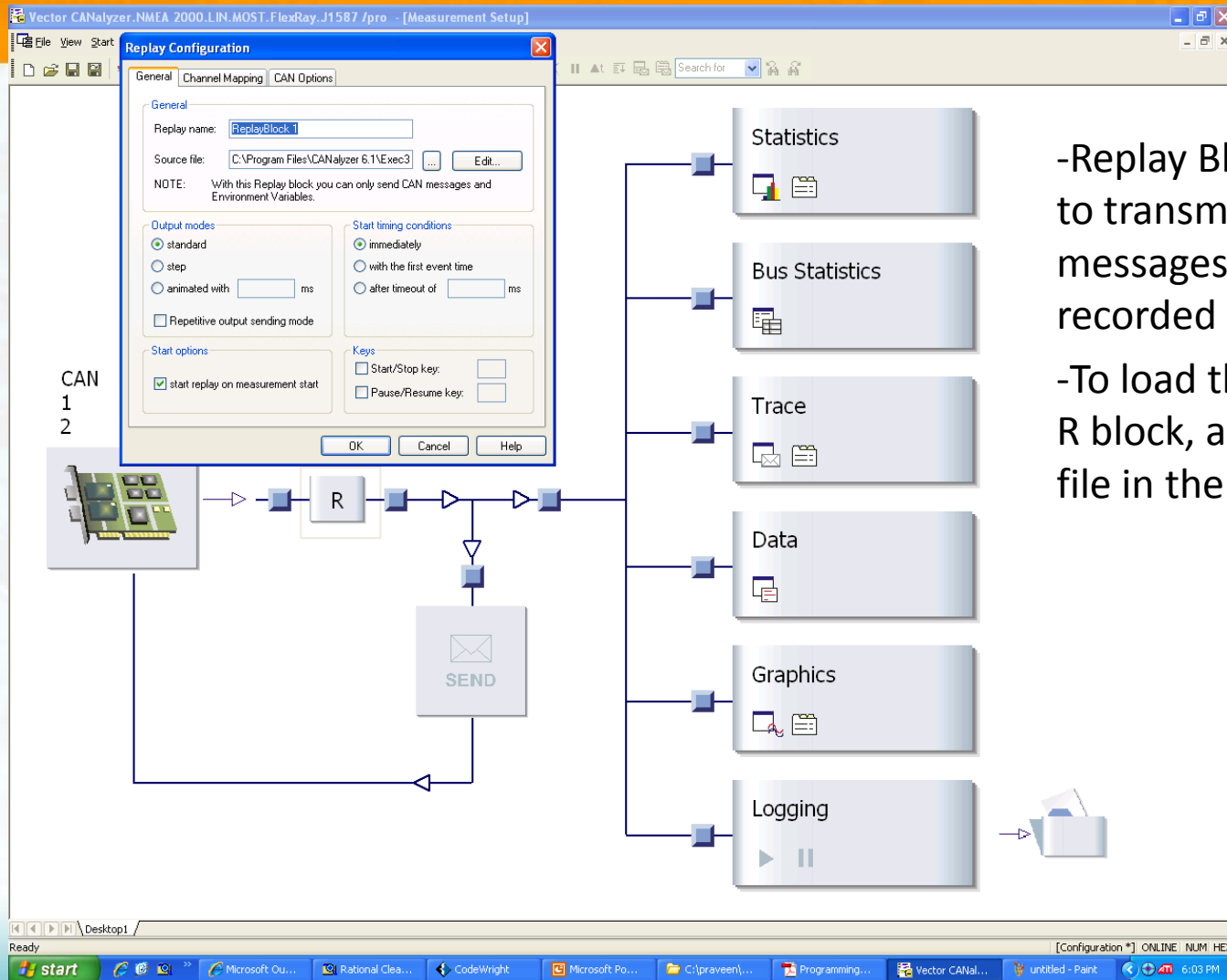
- Before transmitting the message it is necessary to trigger the block.

Interactive Generator Block(IG)



- To configure the message to be transmitted, double-click the IG block. A window show to the left appears.
- User can define the messages or can get it from the Database.

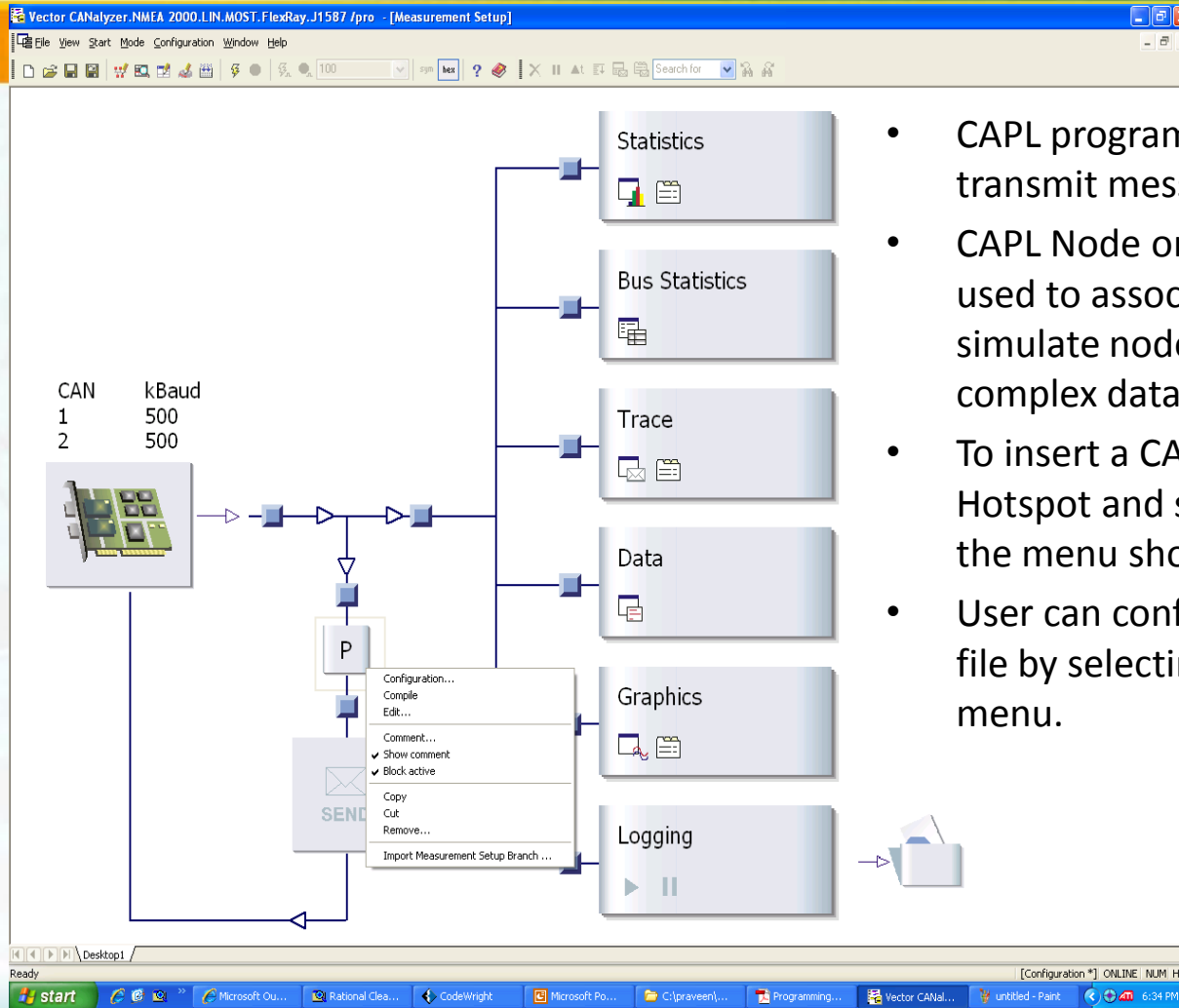
Replay Block(R)



- Replay Block has the capability to transmit a sequence of messages that have been recorded by the Logging file.

- To load the file double-click the R block, and enter the path of the file in the in field Source file:

CAPL Block




- CAPL programming has the ability to transmit messages.
- CAPL Node or Program block (**P**) that is used to associate a CAPL program to simulate node behavior or perform complex data evaluation.
- To insert a CAPL block, right-click on the Hotspot and select Insert CAPL Node from the menu shortcut.
- User can configure the node i.e. can load a file by selecting configure.. from the menu.

Measurement

Start Measurement:

Click on the icon  on the Tool bar to start the Measurement.

Stop Measurement:

Click on the icon  on the Tool bar to stop the Measurement.

References

Documents:

- Programming with CAPL.pdf
- CAPL Function Reference Manual.pdf
- CANalyzer_Datasheet.pdf
- CANalyzer Reference Manual.doc



Links:

- Will Let you know soon.



THANK YOU

Automotive Robotics (India) Pvt, Ltd.

4th Floor | Surya Towers | Secunderabad - 500003 | Phone: 040 - 30280431

web: www.autorobinc.com