



MI0235 MIAC PIC

MI5809 MIAC dsPIC

MI8615 MIAC dsPIC with Wi-Fi

MI8759 MIAC dsPIC with Bluetooth

MI5466 MIAC ATmega

MI9335 MIAC ATmega with Wi-Fi

MI3449 MIAC ATmega with Bluetooth

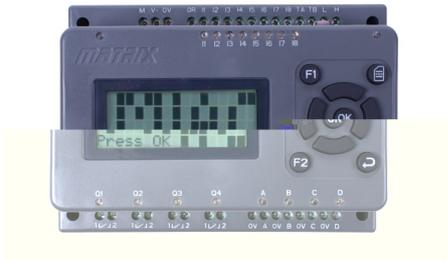
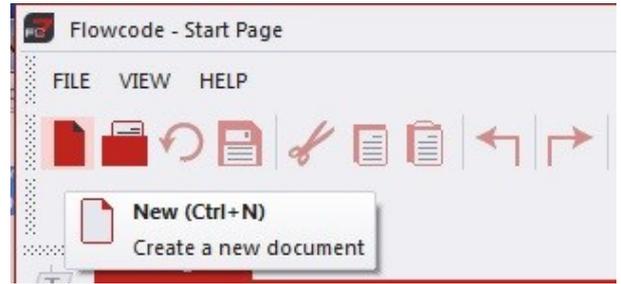
Introduction	3
Creating your project	4
Downloading to the MIAC	7
Flowcode components	8



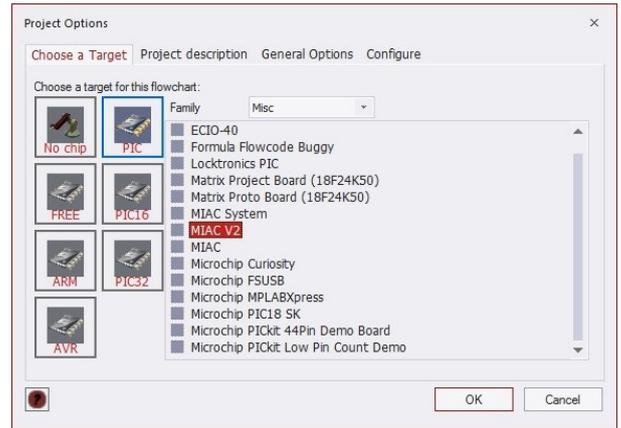
The PIC, dsPIC and ATmega (Arduino-compatible) versions of the Matrix TSL “MIAC” programmable controllers are supported by Flowcode V7.

Flowcode V7 enables the fast development of user applications. It also has the added advantage of being able to simulate your application on screen before deployment to the MIAC hardware.

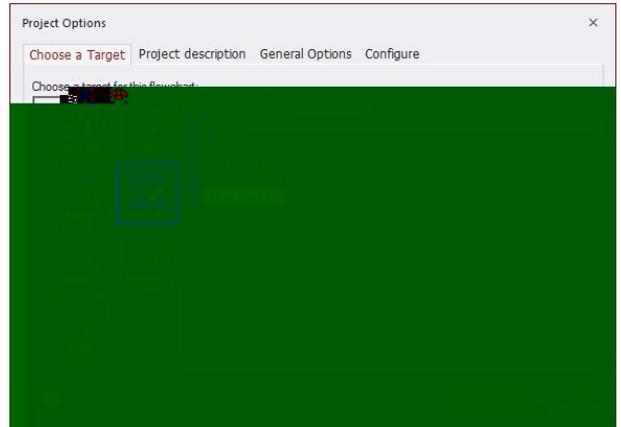
To create a new project, run Flowcode V7 and either click on the New icon or select from the File menu.



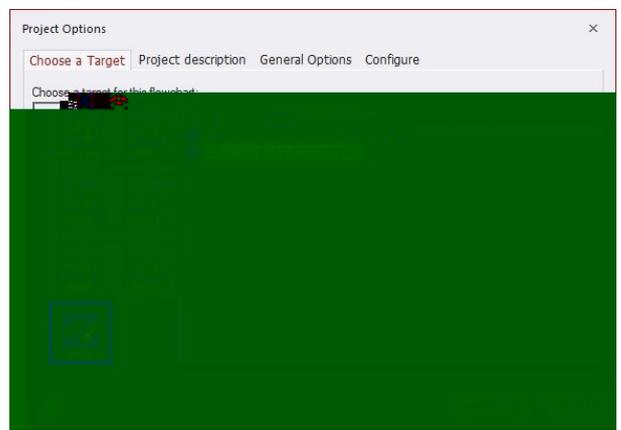
For the 8 bit PIC version of the MIAC , select the "PIC" target type, then "Misc" from the drop down box, select "MIAC V2" and click OK.



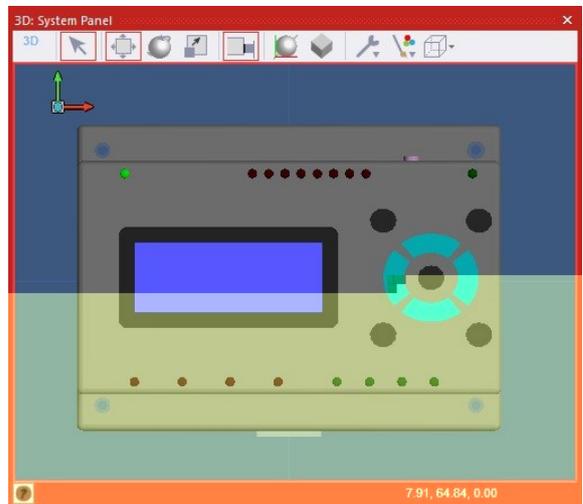
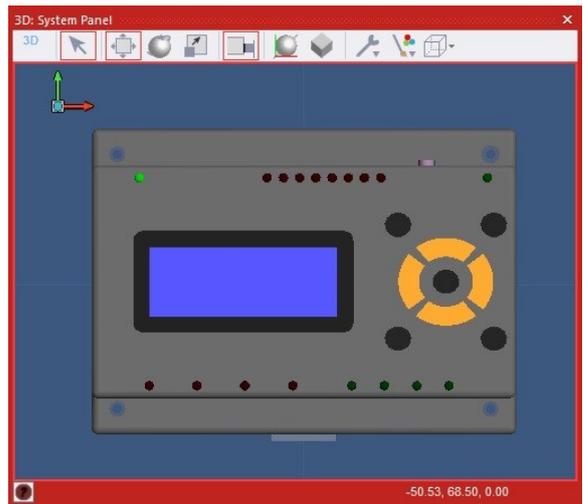
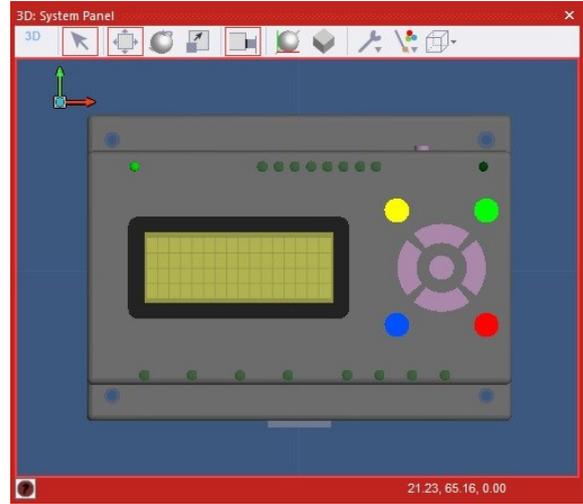
For the dsPIC version of the MIAC , select the "PIC16" target type, then "Misc" from the drop down box, select "MIAC (dsPIC)" and click OK.



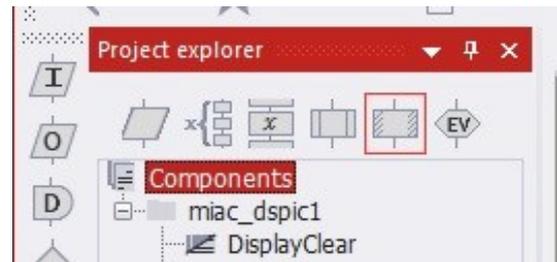
For the ATmega (Arduino-compatible) version of the MIAC , select the "AVR" target type, then "Misc" from the drop down box, select "MIAC (Arduino-compatible)" and click OK.



Select the matching MIAC component from the “MIAC Module” toolbar icon.
 Drag it to either the Dashboard or 3D System panel.

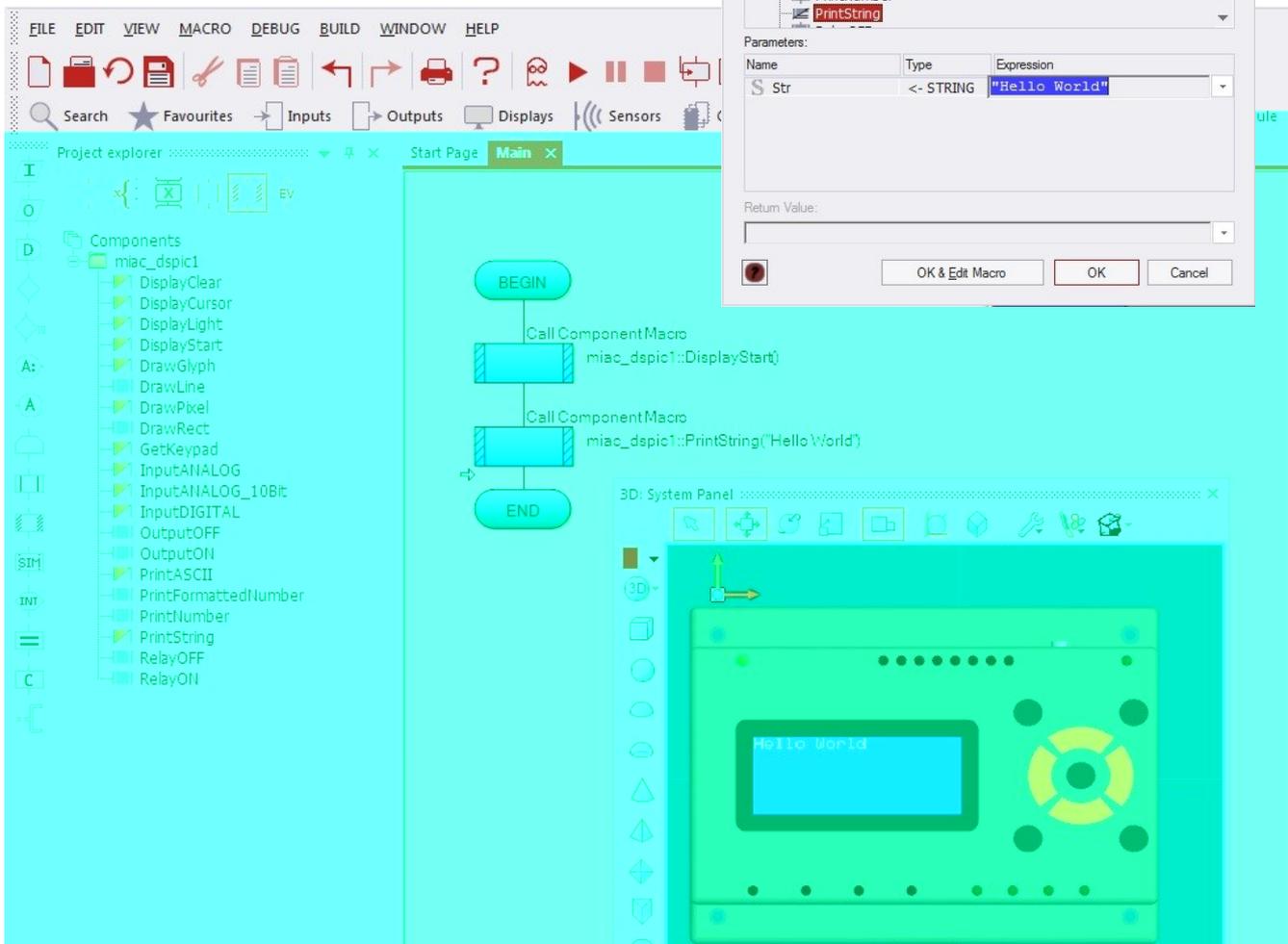
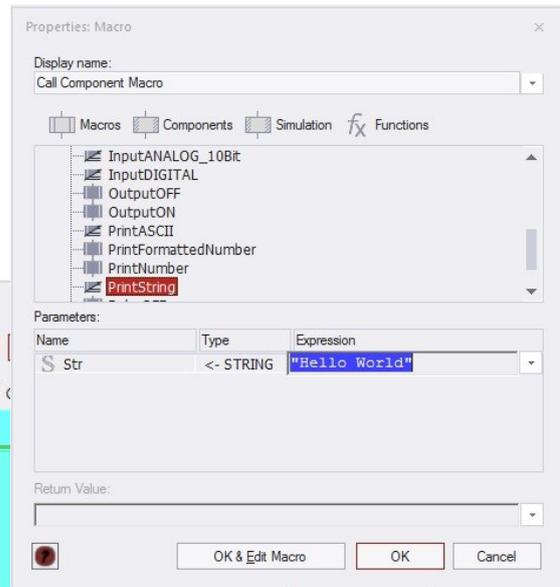


In the “Project explorer” window select the display of the available components for use with the MIAC by clicking on the “Components” icon.



Then start coding your flowchart by dragging, for example, “DisplayStart” and “PrintString” to the flowchart as shown.

In the component details dialog box enter the text to be displayed and click the OK button.



Click the Play icon to simulate this simple application and see the text message appear on screen.



To deploy your project to the PIC (MI0235) MIAC, first download and install the MIAC USB driver from the resource section of the Matrix website.

For dsPIC versions simply connect via a USB cable and apply power to the MIAC. The device will be detected and the drivers installed by Windows.



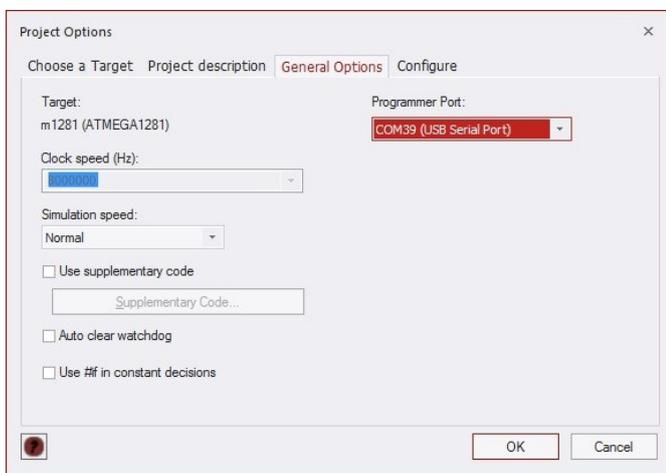
Click the “Compile to Chip” icon.

Press the Reset button on the MIAC when requested to do so.



To deploy your project to the AVR (Arduino-Compatible) versions of MIAC, connect via a USB cable and apply power to the MIAC.

The USB communication port will be detected and, if it is the first time connected, Windows will install the required drivers. If it does not, please download and install the FT230X VCP drivers from www.ftdichip.com



From the Flowcode “Build” menu select the “Project Options” dialog and click on the “General Options” tab.

In the “Programmer Port” drop list find the COM port associated with this MIAC. Click the OK button to save the setting.

Click the “Compile to Chip” icon and the program will be compiled and download to the MIAC.

The MIAC Flowcode component contains the API/macros for basic functionality of the MIAC, such as display, keypad, inputs and outputs. Some MIAC versions however have many more internal features and these are easily implemented in a user application by adding additional Flowcode components into the project.

These components can be found under their relevant toolbar groups, or via the search facility.

Most of the components listed are MIAC-aware in that their properties will be automatically set to values relevant to the MIAC target version being used. The list below gives an overview of these additional components and their use. Please see additional online and in-component Flowcode help.



The dsPIC and ATmega MIAC versions contain a Real Time Clock that has backup power to keep the clock active for up to a few days without power.

The RTC (Real Time Clock) component can be added to a project to give read and write access this internal RTC device.

The clock is set and read as time and date strings. For example, to set the clock to 15:27 call the SetTime macro with a Time parameter of "152700".



The dsPIC and ATmega MIAC versions have an internal Serial EEPROM (non-volatile memory) device for which the component can be added to a project to give read and write access.

This component has Read and Write macros that retrieve and set a data byte at a given address in the non-volatile memory.



The CAN component is used to access the CAN functionality of MIAC. Note: This is not required for MIAC System projects, as these automatically include all the CAN code that is required.



The UART (RS232) component is used to access the UART based serial features of the dsPIC and ATmega versions of the MIAC. It is a generic UART control component so can also be used to access and control the RS232 and RS485 interfaces.

In addition, the optional wireless modules can be accessed with this component. Note that the wireless modules are factory set at 57600 baud and connected to UART 1. Please see the MIAC Datasheet for details on serial interface configurations.

The Bluetooth module can be used as a communication link by simply connecting using the default PIN code 1234.

The UART component properties will need to be configured manually, depending upon the application, as the table below.

		dsPIC Port Pin	ATmega Port Pin
UART 0 (RS232)	TX	E0 (RP80)	E1
	RX	E1 (RPI81)	E0
	CTS	C13	F0
	RTS	C14	F1
UART 1 (RS485)	TX	E4 (RP84)	D3
	RX	E3 (RPI83)	D2
	OE	G9	G4



The SD card component can be used to enhance storage capacity of the dsPIC and ATmega versions of the MIAC when used in conjunction with an optional micro SD card inserted into the internal micro SD card slot.

MATRIX