
PAINLESS MICROCONTROLLER CODE BY GRAPHICAL APPLICATION DESCRIPTION

By Olivier Rouy

INTRODUCTION

Some electromechanics and automatics engineers hesitate to use microcontroller (MCU) solutions despite their recognised advantages: High integration and flexibility for enhanced features. Their main worry is the unpleasant aspect of MCU application development: Learning, code writing, debugging through quite unconvivial tools.

Fortunately, it is possible today to use these tools (and to keep the genuine integrity of the code issued) through a graphic interface. This new complete toolbox, the ST6-Realizer[®], allows a graphical description of the system, automatic code generation, simulation and debugging.

1 A SMART SENSOR EXAMPLE

An application development is described here as an example which covers general items such as analog data acquisition and treatment with peripheral activation and with operation sequencing. The ST62 family of microcontroller is chosen as target device since it provides all the features needed for this kind of application: A/D convertors, high output current capability and a small package.

The application shown consists of an analog sensor input associated with data treatment issuing two types of information: The relative position, lower or higher, versus a meanpoint or reference value and the amplitude of the difference. This smart sensor can be used in a positioning controller, temperature regulator or any other system requiring as parameters a direction and amplitude.

A basic version reduced to a simple, straightforward data treatment is described. Its main purpose is to present the methodology used and main features of the tool.

- Application graphical description
- Analysis and code generation
- Simulation and validation

In a second step, the response of the system is modified to be compatible with non linear input signals or non linear interfaces.

Finally a self calibration feature is implemented to compensate for imprecision on the input sensor.

These last examples highlight the easy upgrading capability for an application.

2 A SIMPLE SMART SENSOR

In its basic form, the application features the minimum data treatment: a subtraction is performed between an entered analog value and a constant reference to control an output signal generator. This last circuit drives two external I/O pins according to the subtraction result: Whether the value is positive or negative, only one of the two outputs is activated issuing a squarewave signal with a frequency function of the amplitude of the difference.

2.1 Graphical description

The core of this development methodology is the graphical description of the application. This description uses a standard library allowing a system oriented description; Included are:

- Input / Output symbols
- Logic / Arithmetic symbols
- Oscillators / Pulse generators
- State machine for sequencing

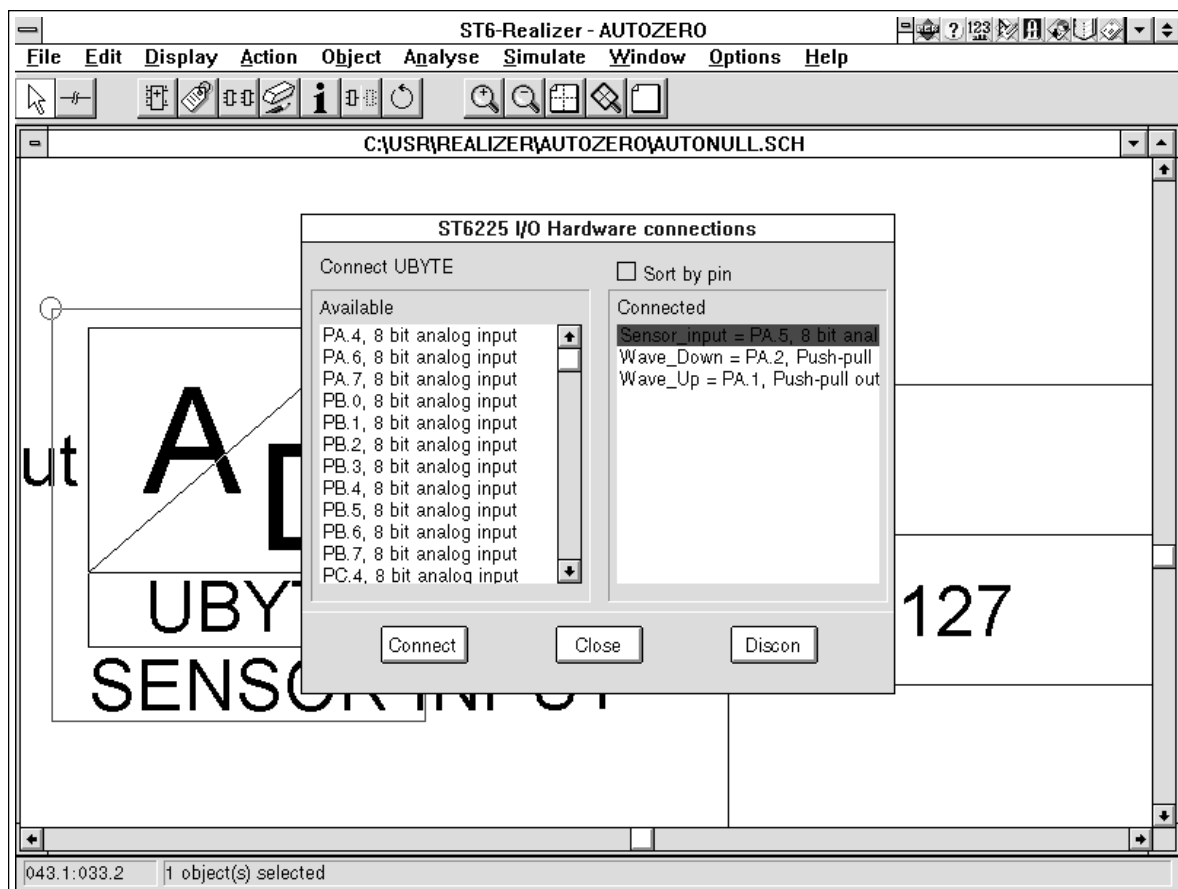
With such a library, it is possible to cover most application requirements as long as the time resolution require for the oscillators and pulse generators remains over 10ms, the value used as the time unit. This value defines only the minimum calibrated timebase available and not the speed of the application which is fixed by the MCU quartz crystal frequency and the complexity of the application.

The graphical description follows a decomposition of the application into such basic functionalities; In the present case our needs are for:

- One analog input with an A/D convertor
- One constant reference
- One subtraction operator
- One wave generator
- Two output pins
- No sequencing is required.

The analog input and digital outputs are described by placing the appropriate symbols proposed in the main library of the object menu. The software proposes the MCU I/O pin available in each case: digital or analog inputs, current sink capabilities for output pins, insuring a full compatibility with the selected hardware (Figure 1).

Figure 1: Pin Assignment In St6-Realizer Environment

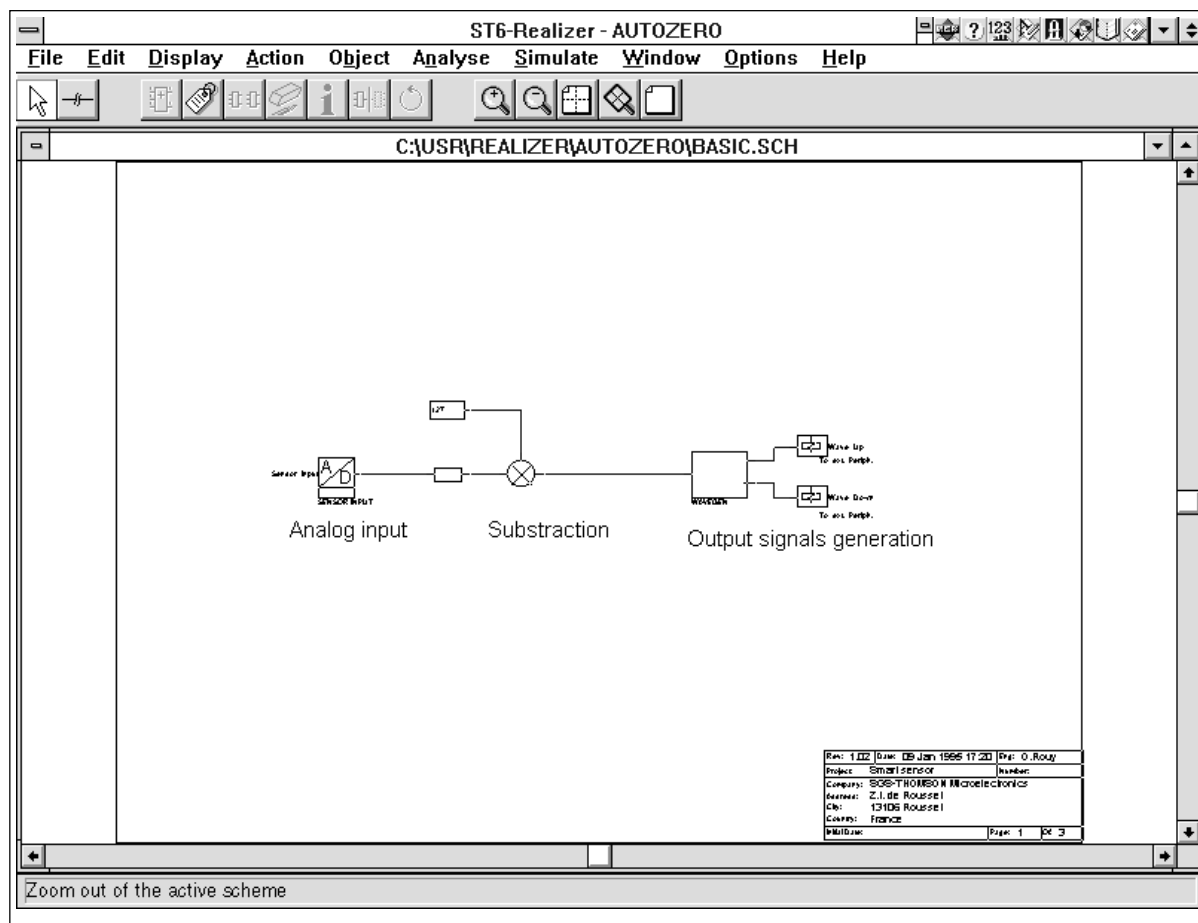


By choosing an 8-bit resolution for the A/D convertor, any analog input value is digitalized into a number between 0 and 255. The constant reference used as meanpoint is described by placing a specific symbol with the value 127.

The subtraction operator and the wave generator are obtained through the same object menu and placed into the schematic.

Eventually, the graphical description is represented in Figure 2.

Figure 2: Simple Smart Sensor Graphical Description



2.2 Analysis / Code generation

After the schematics capture, an analysis of the graphical description must be launched with the analyse menu. This process first checks the integrity of the description, verifies I/O assignments and connectivity providing on-line information in the case of an error, guiding the user in the development work.

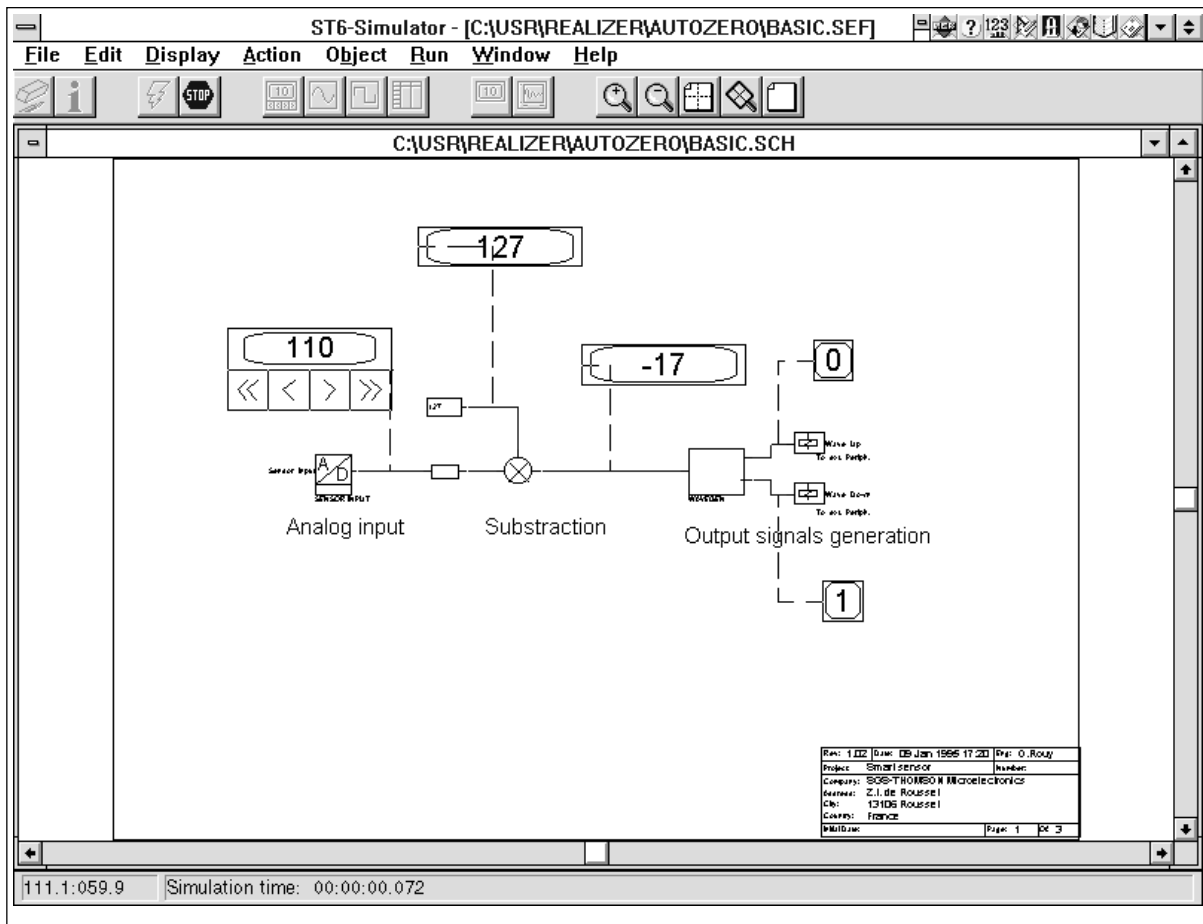
If this first analysis step is successful, all the files related to the ST62 target are automatically generated: The source file in macro-assembler language, the symbol table, the binary executable code. The code generation uses a library of predefined macro-instructions associated to the graphical symbols; thus the time needed from the graphical description to the binary executable code is considerably reduced.

Finally a report file is provided with the Input/Output assignments and the used ROM and RAM. This can help to choose the most suitable MCU for the application. In the present case the report shows 27 RAM bytes and 434 ROM bytes are used, informing the developer about the memory requirement for the MCU and the memory space available for further developments.

2.3 Simulation and validation

When the analysis and code generation are successfully concluded, the simulation can be launched (Simulate menu). This allows the application to be simulated and for the signals to be monitored directly on the graphical description, without having to build hardware for debugging. The use of the numeric adjuster on the inputs, and numeric probes makes the validation of the application very convivial. By activating the input adjuster from 0 to 255 (the 8-bit A/D conversion result) any subtraction situation can be checked with its impact on the output. Figure 3 represents such a case where an input value of 110 is compared to the reference value of 127, resulting in a difference of -17 which produces the oscillation at the relevant output pin.

Figure 3: Simulation Results In The St6-Realizer Environment



It is possible to get information through the Run menu about the maximum time taken by the MCU to perform all the operations described in the application. This fixes the minimum time required between an input change and its impact on outputs, and thus the frequency response of the application. In the present case the maximum time is 630µs for a 8Mhz oscillator.

3 A SMART SENSOR WITH A NON-LINEAR RESPONSE

In the former version, the system has a linear response: If the entered value is increased by 3, the result sent to the wave generator is also increased by 3. However, many systems present non-linear behaviour either due to a non-linear input signal or to a non-linear output stage. In such cases, non-linear transformation capabilities are required.

In this new case a response after the subtractor, as shown in Figure 4, is obtained by only adding two symbols: A transformation table to pass from a linear evolution to a x2-like curve and a limiter symbol to limit the total excursion. The transformation table is a standard ASCII file with a very simple syntax : Input value, Output value, as shown in Figure 5.

Figure 4: Response Transformation Into A Non-Linear Curve

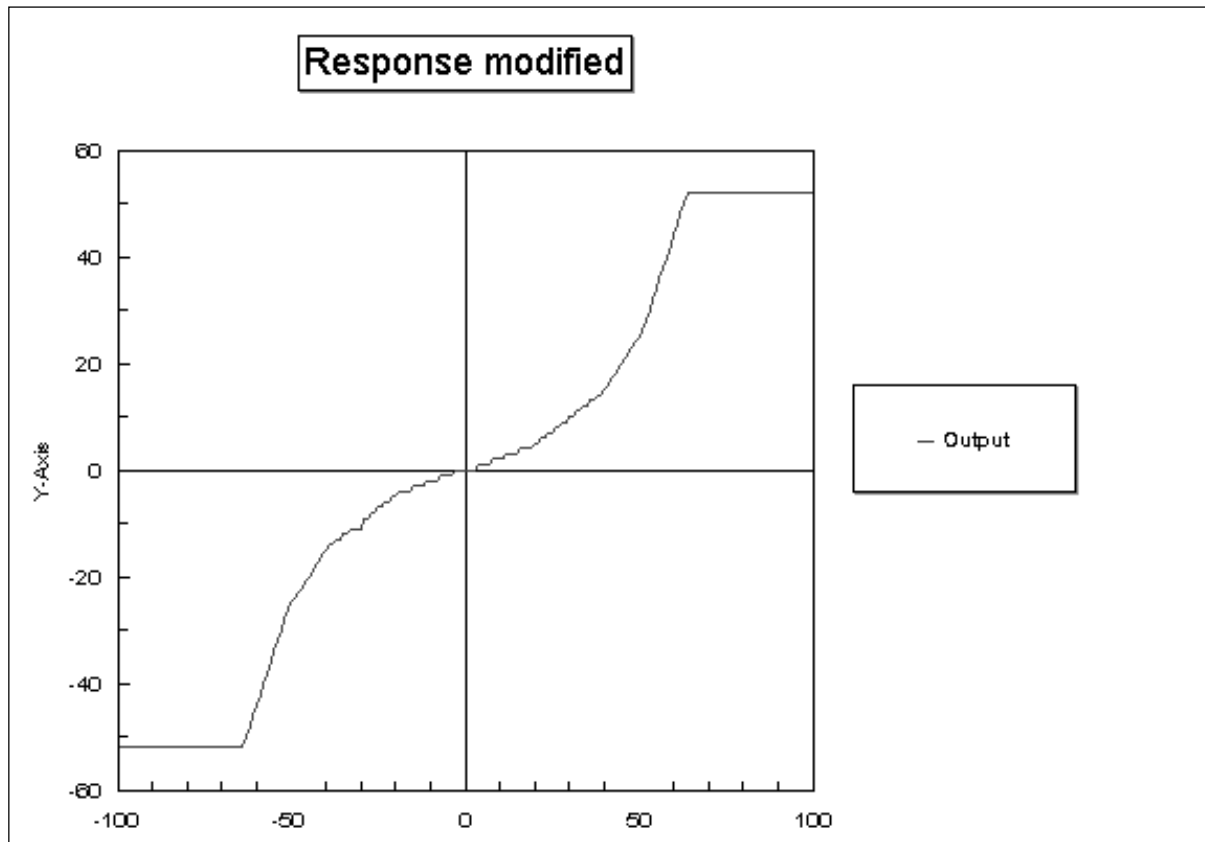
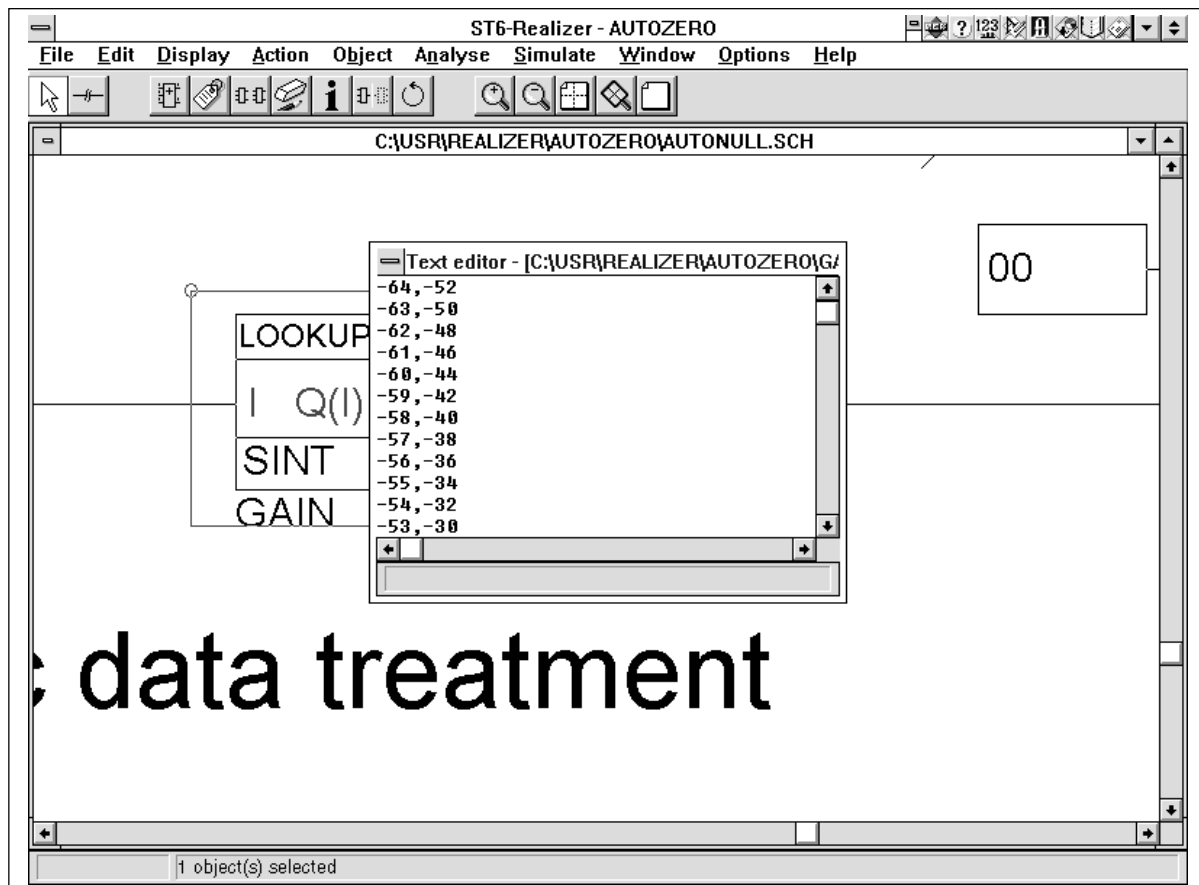


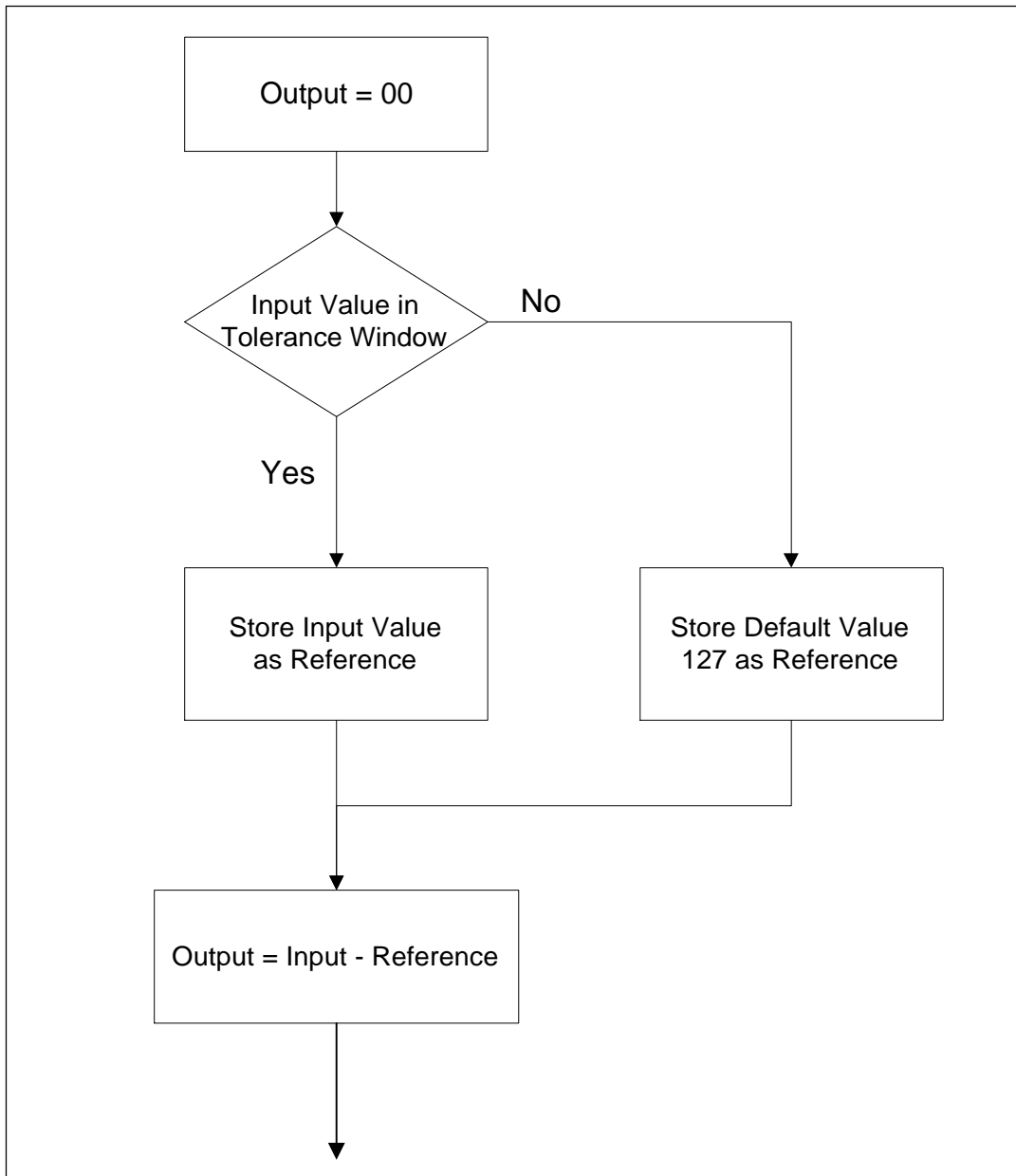
Figure 5: Constant Table Editing In The St6-Realizer Environment



4 A SMART SENSOR WITH NON-LINEAR RESPONSE AND AUTOCALIBRATION

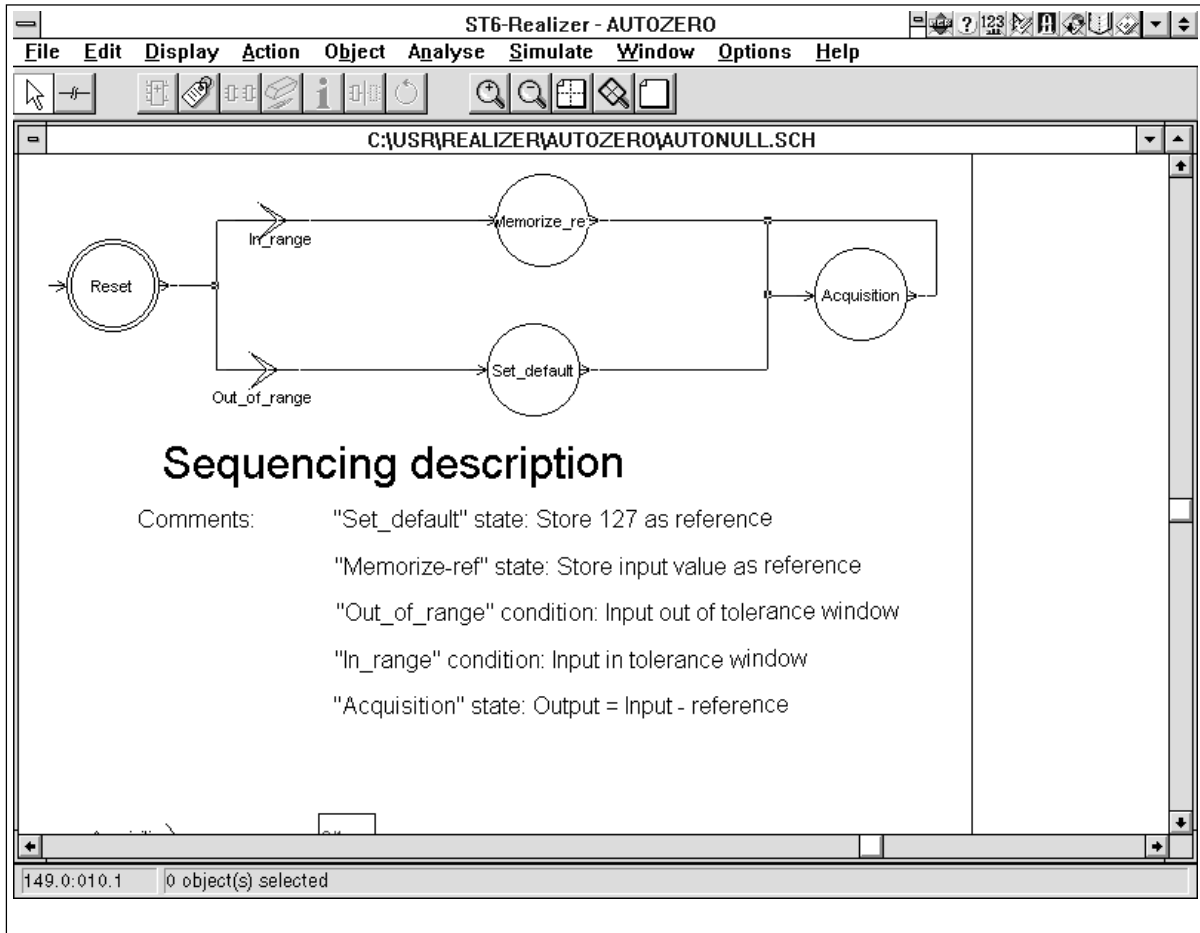
In the case of a non-calibrated sensor, or of mechanical drift of the input system, the meanpoint value may be slightly different from the theoretical reference value and subsequently induce some unwanted actions on the outputs. One solution is to perform an autozero process allowing a tolerance window for the meanpoint, as represented on the flowchart in Figure 6: As long as the initial value at reset is within a tolerance window around the theoretical value of 127, the shift is considered as irrelevant and that initial value is used as the new reference; On the contrary, any input value out of this tolerance window is understood as an instruction request to use 127 as the reference value.

Figure 6: Flowchart Of The Autozero Process At Reset



The description of the flow-chart is very intuitive thanks to the symbols provided in the library for state machine description. As seen in Figure 7 there is a direct correspondence between the flow-chart and the described state machine.

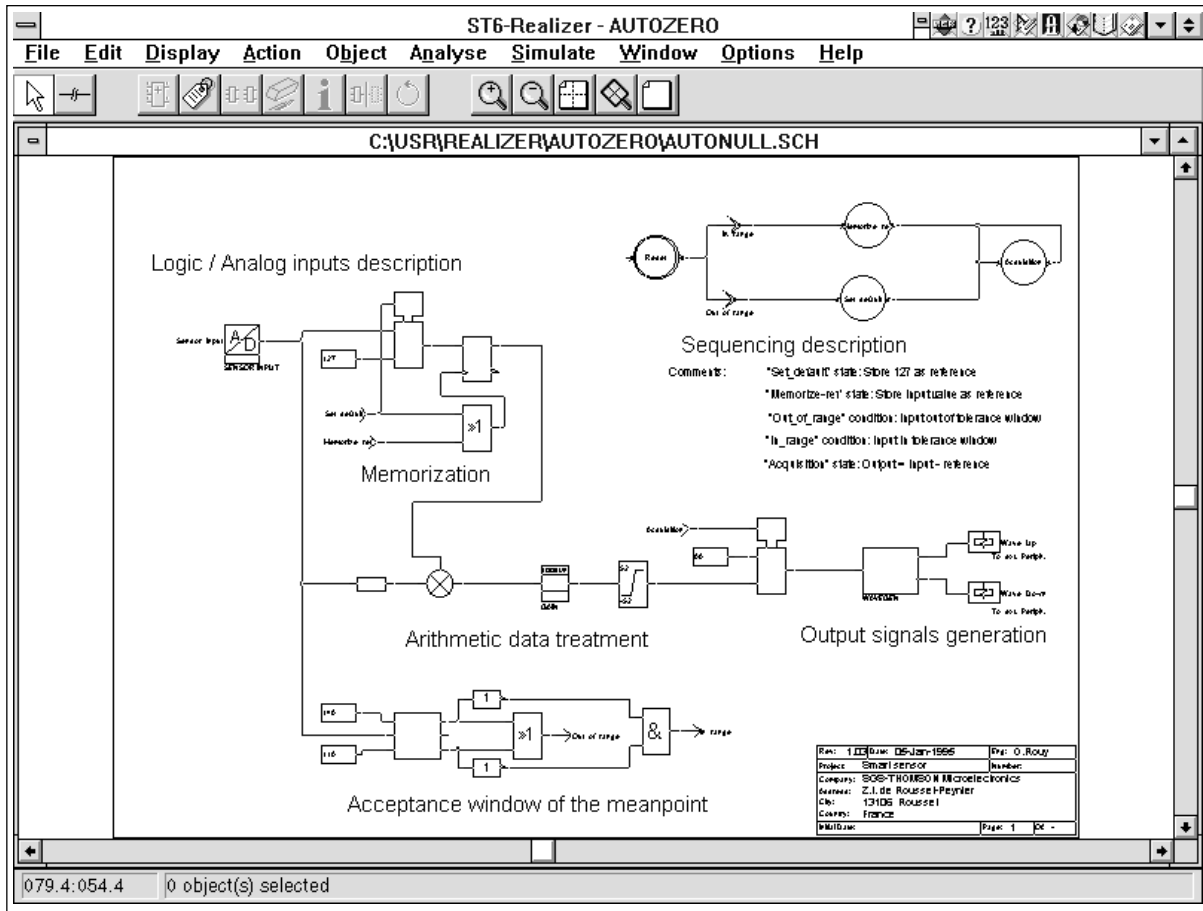
Figure 7: Sequencing Description By State Machine



The description of comparison features for the tolerance window, and the reference data memorisation is done following the methodology of decomposition into basic functions already used in the start-up version.

Eventually, the complete schematic is represented in Figure 8, where the addition of the autozero process and the non linearization feature is easily distinguishable from the start-up version. The modular description and development of the application are very easy to manage.

Figure 8: Smart Sensor With Autocalibration And Non-Linear Capabilities



5 HARDWARE IMPLEMENTATION

The binary executable code generated during the analyse process is simply programmed into an EPROM or OTP MCU device to validate the application in the real world. By using a ST62E25 mounted on an ST62 Starter-kit, it has been checked that:

By turning the potentiometer to the right or to the left, the appropriate output to one of the two connected LEDs oscillates with a frequency increasing with the potentiometer rotation.

The frequency evolution is not linear with the potentiometer rotation according to the desired response transformation.

When a reset is performed with the potentiometer slightly uncentered, any oscillation is turned-off due to the autozero process.

6 CONCLUSION

The presented application demonstrates that a powerful and intuitive tool is available for data treatment and sequencing control in microcontroller based designs. The actual limitation to 10ms of its timing features is compatible with a very wide range of applications in automatic or electromechanics systems, for example: Smart sensors, replacement of relay-based systems, car equipment (window lifter, body mirror) and coded keys identifiers.

Finally, the files generated by the Analyse tool (The source file in macro-assembler language, symbolic table and binary executable code) are obtained by a hidden call to genuine tools which combine the power and integrity of the development tools with a convivial user interface. The generated files are available to MCU designers who wish to mix or link an ST6-Realizer based application with other application code.

The system shown is an open system that may interest experienced designers who want to reduce their development time as well as the engineers with system know-how who want to easily shift to MCU solutions.

Actum Realizer is a trademark of Actum Solutions, The Netherlands.

PAINLESS MICROCONTROLLER CODE BY GRAPHICAL APPLICATION DESCRIPTION

NOTES:

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of SGS-THOMSON Microelectronics.

© 1995 SGS-THOMSON Microelectronics - All Rights Reserved

Purchase of I2C Components by SGS-THOMSON Microelectronics, conveys a license under the Philips I2C Patent. Rights to use these components in an I2C system, is granted provided that the system conforms to the I2C Standard Specifications as defined by Philips.

SGS-THOMSON Microelectronics Group of Companies
Australia - Brazil - China - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco
The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.