

# **8/16-BIT MICROCONTROLLERS (MCUs)**

## **APPLICATION NOTES**

### **ABSTRACTS BY TOPIC**

**AUGUST 2000**

Ref. AN683

---

## Table of Contents

---

### ST6 FAMILY

<b>1.1</b>	<b>SYSTEM DESIGN</b>	<b>6</b>
1.1.1	<b>MOTOR CONTROL</b>	<b>6</b>
	AN392: MICROCONTROLLER AND TRIACS ON THE 110/240V MAINS	
	AN414: CONTROLLING A BRUSH DC MOTOR WITH AN ST6265 MCU	
	AN416: SENSORLESS MOTOR DRIVE WITH THE ST62 MCU + TRIAC	
	AN422: IMPROVES UNIVERSAL MOTOR DRIVE	
	AN863: IMPROVED SENSORLESS CONTROL WITH THE ST62 MCU FOR UNIVERSAL MOTOR	
1.1.2	<b>BATTERY MANAGEMENT</b>	<b>10</b>
	AN417: FROM NICKEL-CD TO NIMH FAST BATTERY CHARGING	
	AN433: ULTRA FAST BATTERY CHARGER USING ST6210 MICROCONTROLLER	
	AN859: AN INTELLIGENT ONE HOUR MULTICHARGER FOR Li-Ion, NiMH and NiCd BATTERIES	
1.1.3	<b>FUZZY LOGIC</b>	<b>12</b>
	AN419: AN APPROACH TO MOTOR CONTROL WITH FUZZY LOGIC	
	AN595: FUZZY VACUUM CLEANER USING ST6220 AND FUZZYTECH™ ST6 EXPLORER	
	AN597: TEMPERATURE CONTROL WITH FUZZY LOGIC	
	AN598: CASCADING FUZZY MODULES WITH ST6 FUZZYTECH	
	AN675: A RAPID CHARGER FOR BATTERIES WITH FUZZY LOGIC	
1.1.4	<b>HOME APPLIANCE</b>	<b>16</b>
	AN674: MICROCONTROLLERS IN HOME APPLIANCES: A SOFT REVOLUTION	
	AN885: ST62 MICROCONTROLLERS DRIVE HOME APPLIANCE MOTOR TECHNOLOGY	
1.1.5	<b>GRAPHICAL DESIGN</b>	<b>17</b>
	AN676: BATTERY CHARGER USING THE ST6-REALIZER®	
	AN677: PAINLESS "MCU" CODE BY GRAPHICAL APPLICATION DESCRIPTION	
	AN839: ANALOG MULTIPLE KEY DECODING USING THE ST6-REALIZER	
	AN840: CODED LOCK USING THE ST6-REALIZER	
	AN841: A CLOCK DESIGN USING THE ST6-REALIZER	
	AN842: 7 SEGMENT DISPLAY DRIVE USING THE ST6-REALIZER	
<b>1.2</b>	<b>SYSTEM OPTIMIZATION</b>	<b>19</b>
1.2.1	<b>COST REDUCTION</b>	<b>19</b>
	AN431: USING ST6 ANALOG INPUTS FOR MULTIPLE KEY DECODING	
	AN594: DIRECT SOFTWARE LCD DRIVE WITH ST621X AND ST626X	
	AN672: OPTIMIZING THE ST6 A/D CONVERTER ACCURACY	
	AN673: REDUCING CURRENT CONSUMPTION AT 32KHZ WITH ST62	

## Table of Contents

### 1.2.2 DESIGN IMPROVEMENTS ..... 22

AN420: EXPANDING A/D RESOLUTION OF THE ST6 A/D CONVERTER  
 AN432: USING ST62XX I/O PORTS SAFELY  
 AN434: MOVEMENT DETECTOR CONCEPTS FOR NOISY ENVIRONMENTS  
 AN435: DESIGNING WITH MICROCONTROLLERS IN NOISY ENVIRONMENTS  
 AN669: SIMPLE RESET CIRCUITS FOR THE ST62  
 AN670: OSCILLATOR SELECTION FOR ST62  
 AN671: PREVENTION OF DATA CORRUPTION IN ST6 ON-CHIP EEPROM  
 AN911: ST6 MICRO IS EMC CHAMPION  
 AN975: UPGRADING FROM ST625X/6XB TO ST625X/6XC  
 AN1015: ST6 SOFTWARE TECHNIQUES FOR IMPROVING EMC PERFORMANCE

### 1.2.3 PERIPHERAL OPERATIONS ..... 26

AN590: PWM GENERATION WITH ST62 AUTO-RELOAD TIMER  
 AN591: INPUT CAPTURE WITH ST62 AUTO-RELOAD TIMER  
 AN592: PLL GENERATION USING THE ST62 AUTO-RELOAD TIMER  
 AN593: ST62 IN-CIRCUIT PROGRAMMING  
 AN678: LCD DRIVING WITH ST6240  
 AN913: PWM GENERATION WITH ST62 16-BIT AUTO-RELOAD TIMER  
 AN914: USING ST626X SPI AS UART  
 AN1016: ST6 USING THE ST623XB/ST628XB UART  
 AN1050: ST6 INPUT CAPTURE WITH ST62 16-BIT AUTO-RELOAD TIMER  
 AN1127: USING THE ST62T6XC/5XC SPI IN MASTER MODE

## ST7 FAMILY

### 2.1 PROGRAMMING AND TOOLS ..... 29

AN978: KEY FEATURES OF THE STVD7 ST7 VISUAL DEBUG PACKAGE  
 AN983: KEY FEATURES OF THE COSMIC ST7 C-COMPILER PACKAGE  
 AN985: EXECUTING CODE IN ST7 RAM  
 AN986: USING THE ST7 INDIRECT ADDRESSING MODE  
 AN987: ST7 IN-CIRCUIT PROGRAMMING  
 AN988: STARTING WITH ST7 ASSEMBLY TOOL CHAIN  
 AN989: STARTING WITH ST7 HIWARE C  
 AN1039: ST7 MATH UTILITY ROUTINES  
 AN1064: WRITING OPTIMIZED HIWARE C LANGUAGE FOR ST7  
 AN1106: TRANSLATING ASSEMBLY CODE FROM HC05 TO ST7

### 2.2 EXAMPLE DRIVERS ..... 33

AN969: ST7 SCI COMMUNICATION BETWEEN THE ST7 AND A PC  
 AN970: ST7 SPI COMMUNICATION BETWEEN THE ST7 AND E<sup>2</sup>PROM  
 AN971: ST7 I<sup>2</sup>C COMMUNICATION BETWEEN THE ST7 AND E<sup>2</sup>PROM  
 AN972: ST7 SOFTWARE SPI MASTER COMMUNICATION  
 AN973: SCI SOFTWARE COMMUNICATION WITH A PC USING ST72251 16-BIT

---

## Table of Contents

---

### TIMER

- AN974: REAL TIME CLOCK WITH THE ST7 TIMER OUTPUT COMPARE
- AN976: DRIVING A BUZZER USING THE ST7 PWM FUNCTION
- AN979: DRIVING AN ANALOG KEYBOARD WITH THE ST7 ADC
- AN980: ST7 KEYPAD DECODING TECHNIQUES, IMPLEMENTING WAKE-UP ON KEYSTROKE
- AN1017: USING THE ST7 USB MICROCONTROLLER
- AN1041: USING ST7 PWM SIGNAL TO GENERATE ANALOG OUTPUT (SINUSOID)
- AN1042: ST7 ROUTINE FOR I<sup>2</sup>C SLAVE MODE MANAGEMENT
- AN1044: MULTIPLE INTERRUPT SOURCES MANAGEMENT FOR ST7 MCUS
- AN1045: ST7 SOFTWARE IMPLEMENTATION OF I<sup>2</sup>C BUS MASTER
- AN1046: ST7 UART EMULATION SOFTWARE
- AN1047: MANAGING RECEPTION ERRORS WITH THE ST7 SCI PERIPHERAL
- AN1048: ST7 SOFTWARE LCD DRIVER
- AN1078: ST7 TIMER PWM DUTY CYCLE SWITCH FOR TRUE 0% or 100% DUTY CYCLE
- AN1148: USING THE ST7263 FOR DESIGNING A USB MOUSE
- AN1149: HANDLING SUSPEND MODE ON A USB MOUSE
- AN1082: DESCRIPTION OF THE ST72141 MOTOR CONTROL PERIPHERAL REGISTERS
- AN1083: ST72141 BLDC MOTOR CONTROL SOFTWARE AND FLOWCHART EXAMPLE
- AN1129: PWM MANAGEMENT FOR BLDC MOTOR DRIVES USING THE ST72141
- AN1130: AN INTRODUCTION TO SENSORLESS BRUSHLESS DC MOTOR DRIVE APPLICATIONS WITH THE ST72141
- AN1180: USING THE ST7263 KIT TO IMPLEMENT A USB GAME PAD
- AN1182: USING THE ST7 USB LOW-SPEED FIRMWARE

## 2.3 PRODUCT OPTIMIZATION ..... 42

- AN982: USING CERAMIC RESONATORS WITH THE ST7
- AN1014: HOW TO MINIMIZE THE ST7 POWER CONSUMPTION
- AN1070: ST7 CHECKSUM SELF-CHECKING CAPABILITY
- AN1179: PROGRAMMING ST7 FLASH MICROCONTROLLERS IN REMOTE ISP

## 2.4 PRODUCT EVALUATION ..... 43

- AN910: ST7 AND ST9 PERFORMANCE BENCHMARKING
- AN990: ST7 BENEFITS VERSUS INDUSTRY STANDARD
- AN1086: ST7 / ST10U435 CAN-do SOLUTIONS FOR CAR MULTIPLEXING
- AN1150: BENCHMARK ST72 VS PC16
- AN1151: PERFORMANCE COMPARISON BETWEEN ST72254 & PC16F8

---

## Table of Contents

---

### ST9 FAMILY

#### 3.1 APPLICATION EXAMPLES ..... 46

AN413: INITIALIZATION OF THE ST9  
AN415: USING THE I2C-BUS PROTOCOL  
AN421: STACK OVERFLOW DETECTION USING THE ST9 TIMER WATCHDOG  
AN910: ST7 AND ST9 PERFORMANCE BENCHMARKING  
AN1069: ADDRESSING UP TO 4 MBYTES OF MEMORY FROM A ST9+ WITH A 16-BIT EXTERNAL BUS  
AN1075: USING THE ST9+ MEMORY MANAGEMENT UNIT (EXAMPLES FOR ST92195 & ST92R195)  
AN1076: ST9+ EXTERNAL MEMORY INTERFACE CONFIGURATION  
AN1087: ST9+ INTERRUPT RESPONSE TIME

### ST10 FAMILY

AN490: PROGRAMMING FLASH MEMORY OF THE ST10F166  
AN1086: ST7 / ST10U435 CAN-do SOLUTIONS FOR CAR MULTIPLEXING

### GENERAL

AN886: SELECTING BETWEEN ROM AND OTP FOR A MICROCONTROLLER  
AN887: MAKING IT EASY WITH MICROCONTROLLERS  
AN898: EMC GENERAL INFORMATION  
AN899: SOLDERING RECOMMENDATIONS and PACKAGING INFORMATION  
AN900: INTRODUCTION TO SEMICONDUCTOR TECHNOLOGY  
AN901: EMC GUIDE-LINES FOR MICROCONTROLLER - BASED APPLICATIONS  
AN902: QUALITY AND RELIABILITY INFORMATION  
AN912: A SIMPLE GUIDE TO DEVELOPMENT TOOLS  
AN1181: ELECTROSTATIC DISCHARGE SENSITIVITY MEASUREMENT

# 1 ST6 FAMILY

## 1.1 SYSTEM DESIGN

### 1.1.1 MOTOR CONTROL

Microcontrollers are now commonly used to drive motors. They enable control the speed, the torque or the power on the load. They include safety features and interface to a large variety of sensors. In addition, the flexibility provided by the software enables the adaptation the same device to different types of motors and equipment for a limited cost.

ST62 Microcontrollers operate directly on the mains with a minimum of surrounding components thanks to their integrated noise immunity. In addition their embedded A/D converter and their ability to drive directly the power stage simplify the user and power interfacing.

Application notes AN392 and AN416 describe Universal motor controls where the power device is a triac. AN392 presents with a practical example of what an ST62 microcontroller can provide in a motor drive in term of control, protection and user interface. AN416 describes how the same control can be slightly adapted to achieve low cost speed compensation of the same motor without a speed sensor.

Application note AN414 presents an innovative motor control which enables high volumic power and a large speed variation range. The motor is a DC Permanent Magnet Motor; the Power stage is a Pulse Width Modulation chopper based on an IGBT and the control is achieved by an ST62 Microcontroller which directly interfaces to the analog sensors and the power stage.

These circuits can be used in a large variety of applications such as home appliances (washing machine, food processor, drill,...) and industrial systems (pump, light dimmer, alarm,...).

**AN392: MICROCONTROLLER AND TRIACS ON THE 110/240V MAINS****P. Rabier, L. Perier**

Microcontrollers are in common use in most areas of electronics. They now penetrate the very cost sensitive arena of home appliance applications. The demonstration board described in this Application Note shows that enhanced appliances can be designed with fast prototyping time using microcontrollers such as the ST6210.

The circuit presented is an enhanced light dimmer operating from the 110/240V mains. It drives incandescent and halogen lamps supplied either directly from the mains or through a low voltage transformer.

The same circuit can also drive a universal motor. It includes soft start and protection features. Different user interfaces can be chosen: touch sensor, push button or potentiometer.

All this is achieved with only few components: a microcontroller ST6210, a Logic Level or a snubberless triac and some passive components.

Additional features like presence detection, IR remote control, homebus interface and motor speed control can be implemented from the existing solution.

**AN414: CONTROLLING A BRUSH DC MOTOR WITH AN ST6265 MCU****J. Nicolai, T Castagnet**

Microcontrollers are used more and more in motor drives, most commonly to turn on and off the motor, or to control the triggering angle of a triac, associated with a universal motor. In this paper, the universal motor is replaced by a permanent magnet DC motor, which offers the advantage of yielding a high power in a small volume. The variable speed drive is supplied from the rectified mains voltage, and consists of a chopper driven by a PWM signal generated by the ST6265 microcontroller (MCU).

The ST6265 MCU measures the DC supply voltage and adjusts the PWM duty cycle accordingly. So the motor voltage is regulated in case of mains and load variations, the motor speed is adjusted and the motor current ripple is reduced, thus reducing acoustic noise and losses in the motor. This drive also implements a software power limitation, which avoids motor overheating in case of excessive loading. Software flexibility allows to easily modify drive parameters such as maximum power, time constants, etc... The drive principle and practical results are given.

### **AN416:SENSORLESS MOTOR DRIVE WITH THE ST62 MCU + TRIAC**

**Thierry Castagnet**

Home appliance applications require more and more electronic control in order to meet the new requests and constraints of the consumers.

Microcontrollers have been typically limited to high end applications because their performance appear to be overrated when related to the functions of the application. In reality, home appliances require microcontrollers which trade closely on the compromise between cost and performance.

This Application Note describes how a low cost speed drive is designed with a microcontroller, a triac and an a.c. universal motor. The control of the motor operates with sensorless speed control, based only on a motor current feedback. The example shown is adapted to food processor and to drill applications.

### **AN422:IMPROVES UNIVERSAL MOTOR DRIVE**

**JM. Bourgeois/JM. Charreton/P. Rault**

Universal motors are traditionally used in AC current mode of control with a Triac-based control circuit. This provides a low-cost circuit, but has potential drawbacks in high peak to peak current giving poor motor efficiency and high brush temperature leading to a limited motor life-time.

Operation in DC current mode provides a solution to these problems and also the increased efficiency allows a reduction in the size of the motor. Motor noise is also reduced.

This Application Note presents three solutions for control of a Universal Motor, in AC and DC current modes with a comparison of the efficiency of each. The modes covered are:

- AC drive with Triac,
- DC drive with Triac and rectifier bridge
- DC drive with an IGBT and rectifier bridge.

In all cases the control of the drive is made by a microcontroller, the ST6, with an illustration of the circuit and additional functionality possible.



**AN863:IMPROVED SENSORLESS CONTROL WITH THE ST62 MCU FOR UNIVERSAL MOTOR****J. Nicolai, A. Bailly, T. Castagnet**

The universal motor is today the most widely used motor in home appliances (vacuum cleaner, washer, hand tool, food processor...). This note describes a speed regulator without sensor: the speed sensing is per-formed indirectly by the ST6220, low-cost 8-bit microcontroller, measuring the motor current. Performance results are given, which are in line with the need of many home appliances.

### 1.1.2 BATTERY MANAGEMENT

Portable equipment is proliferating in the consumer, telecom and home appliance fields. This equipment requires batteries that are powerful, small, environmentally safe and fast to charge.

As answers to the diversity of the applications requests, a large variety of technologies of batteries are available on the market. Most of these batteries require a sophisticated control algorithm in order to achieve a fast and safe charge and to maximize the battery life time.

A standard ST62 microcontroller loaded with a dedicated program controls the temperature, the peak voltage (V), the  $dV/dt$  or the inflexion point of the battery voltage. It ensures that the charging stops when the electrochemical process is finished, once the battery is full and not yet heated-up.

Application notes AN417 and AN433 describe in detail the software algorithm and the hardware implementation for the fast charge of such batteries. The note AN433 details the charging in 1/4 hour of NiCd batteries using a (-DeltaV) method. AN417 presents another control technique based on the measurement of the inflexion point of the battery voltage, it is applied to a NiMH battery.

These programs can be used as a basis for customizing such as battery discharge, gas gauge, display or an adaptation to other battery types, for instance Alkaline Lithium or Lithium Ion.

### AN417:FROM NICD TO NIMH FAST BATTERY CHARGING

**J.Nicolai, L.Wuidart**

Rechargeable batteries are quickly becoming a major benefit to current lifestyles. They allow such utilities as portable telephones, camcorders, cordless power tools, portable appliances and audio equipment.

The charging of Rechargeable batteries often requires to take place in one hour and less for user convenience in applications which discharge rapidly, for example cordless power drills. The optimum fast charging techniques for Nickel Cadmium batteries are well known (please refer to AN433, Fast NiCd Battery Charging using ST6210 MCU), however these techniques are not suitable for the charging of batteries using the more environmentally friendly Nickel Hydride (NiMH) technology.

This Application Note shows the differences in the charging of the two technologies and how a Fast Battery Charger compatible with both NiCd and NiMH can be made with the ST6210 MCU. The MCU control shown is able to provide three level charge termination methods for safe charging.

**AN433:ULTRA FAST BATTERY CHARGER USING ST6210 MICROCONTROLLER****L. Wuidart, P. Richter**

Cordless and portable battery supplied equipment is proliferating thanks to the increasing capacity of rechargeable NiCd batteries.

A useful feature in applications where the battery is rapidly discharged, such as power tools, is ultra fast charging in less than half an hour.

The use of non-adapted monitoring charge methods may lead to a reduction of the battery life time and, in the worst case, to the explosion of the battery.

The solution described in this Application Note is an efficient 100 kHz converter charging an NiCd battery in less than 15 minutes. The battery charge is monitored by a low cost microcontroller ST6210 enabling battery voltage identification, temperature monitoring and charge control.

Overall performances and practical results are given. The program developed to control the battery charge is briefly described.

**AN859:AN INTELLIGENT ONE HOUR MULTICHARGER FOR Li-Ion, NiMH and NiCd BATTERIES****J-M. Ravon and L. Wuidart**

A new intelligent multicharger concept, fully compatible with Li-Ion, NiCd and NiMH battery technology, illustrating the power and flexibility offered by a low-cost industry standard Microcontroller and the ease with which existing designs may be adapted to cater for emerging technologies. A low-cost, high resolution, voltage measurement technique using capacitor charge time is also described.

### 1.1.3 FUZZY LOGIC

Fuzzy logic brings computer reasoning closer to the way people think. It suits well applications that are more easy to describe with a linguistic approach than with a mathematical model, especially non-linear systems. It provides also a bridge between the analog and digital world, enabling a designer with a good analog background to develop a microcontroller based control without digital expertise.

In this book, several application notes describe practical systems. They show how to design an optimized controller, even for relatively fast “real time” loops, without specific experience on digital control techniques. AN419 presents successive steps in the design of a fuzzy logic speed regulation of a motor. This closed loop control operates as a pseudo real time regulation with a bandwidth of around 100ms. The target is a food processor. The note AN595 presents an other example of closed loop fuzzy logic control in the case of a vacuum cleaner.

These applications have been designed with fuzzyTECH ST6 Explorer Edition. This fuzzy logic development tool is an easy to use, high level software tool optimized for the design of fuzzy logic controls with the ST62 microcontroller. It covers all the steps of a fuzzy logic design from the initial concept to the production of an optimized ST62 executable code. In addition, its MS-Windows™ based interface takes full advantage of the intuitive approach of fuzzy logic to define and optimize the control on a very friendly approach.

The solutions presented in these notes can be enlarged to home appliances (washing machine, vacuum cleaner and food processor,... ) , temperature control (air conditioning, refrigeration, cooker, oven, central heating, furnace,...), sensor interface (InfraRed detector, Alarm, ...), motor control (speed or position), or battery charger.

**AN419:AN APPROACH TO MOTOR CONTROL WITH FUZZY LOGIC****P. Guillemin**

Introduced in 1965, Fuzzy Logic now takes more and more importance in Industrial, Home Appliance and Consumer applications such as camcorders, washing machines, vacuum cleaners and microwave ovens. In a wide range of applications, from complex (chemical process regulation, automotive features like ABS) to very simple applications (temperature or voltage control), Fuzzy Logic concepts associated with Fuzzy Logic Development tools bring beneficial advantages to system design.

This Application Note describes the design of a Fuzzy Logic motor controller using the ST62 Microcontroller and a Fuzzy Logic design tool. The "*fuzzyTECH* ST6 Explorer Edition" covers all the steps of a Fuzzy Logic Design, from the initial concept definition, up to the generation of executable code for the ST62.

All the necessary characteristics of the motor, the way they are used with the development tool and the practical utilization of the tool are described in this Application Note.

*fuzzyTECH* ST6 Explorer Edition is a trademark of Inform Software Corporation.

**AN595:FUZZY VACUUM CLEANER USING ST6220 AND FUZZYTECH™ ST6 EXPLORER****Central Application Laboratory, Singapore**

For the past 20 years, the home environment has changed drastically and with the rise of living standards, the consumers' need for home cleaning has switched from a simple mop or scrub to a more sophisticated mode. A vacuum cleaner that is able to do the cleaning based on different characteristics of floor surfaces will be very desired for today's market.

This new requirement actually represents the first and most visible group of the next generation of consumer products based on a new control-fuzzy logic. Fuzzy logic is a relatively new technology that enables machines and products to operate more efficiently and independently by processing information similar to the way people do.

This note describes a universal motor power control implemented on a standard microcontroller running software using the fuzzy logic concept. The different stages of development of the motor power control for a vacuum cleaner are described with the ST6220 microcontroller and a fuzzy logic development tool, the "*fuzzyTECH*™ ST6 Explorer Edition".

### **AN597:TEMPERATURE CONTROL WITH FUZZY LOGIC**

**Lionel Picandet**

Fuzzy logic may be considered as an assortment of decision making techniques. In many applications like process control, the algorithm's outcome is ruled by a number of key decisions which are made in the algorithm. Defining the best decision requires extensive knowledge of the system. When experience or understanding of the problem is not available, optimizing the algorithm becomes very difficult. This is the reason why fuzzy logic is so useful.

We can split the problem into a discrete number of possible decisions by associating fuzzy logic membership functions with each input and output. The accuracy of the output depends on how many membership functions we define and how many rules we implement. The outcome is that a user without know-how or an extensive understanding can solve the problem.

This Application note describes the use of fuzzy logic to create a temperature controller suitable for home appliance needs. The example shown uses the ST6225 microcontroller with the ST6 fuzzyTECH Explorer Edition fuzzy logic development program. Practical steps are shown.

### **AN598:CASCADING FUZZY MODULES WITH ST6 FUZZYTECH**

**Lionel Picandet/Lim King Soon**

Sometimes an application using fuzzy logic may require two fuzzy modules to operate with the desired conditions.

The ST6 fuzzyTECH Explorer Edition is capable of generating only one fuzzy module at a time. However this note explains a technique for linking two fuzzy modules into one application at the same time, without extensive modifications to the ST6 source code files created by ST6 fuzzyTECH Explorer Edition.

It should be noted that these two modules cannot work at the same time, but they can operate sequentially or in a different time period.

Application of this note requires knowledge of the ST6 fuzzyTECH Explorer Edition fuzzy logic development tool and the ST6 software development tools.

**AN675: A RAPID CHARGER FOR BATTERIES WITH FUZZY LOGIC****BOSCH Corporation**

The advantages of Nickle-Cadmium (NiCd) batteries for portable appliances are well known (high peak current, low cost, and wide range of packaging). However NiCd batteries also have disadvantages for efficient and fast charging.

To correctly charge a NiCd battery a good estimation of the charge level of the battery needs to be made. This is dependent on a number of unknown variables such as age, charge and discharge state and temperature, which cannot be forecast.

While several techniques for charging are well known (refer to AN417 and AN433), this Application Note provides further extensions into the control domain by the application of fuzzy logic.

An overview of a practical and current application of fuzzy logic is shown with the Membership Functions and Rules chosen.

Application of this note requires prior knowledge of the ST6 fuzzyTECH Explorer Edition fuzzy logic development tool.

### 1.1.4 HOME APPLIANCE

#### **AN674: MICROCONTROLLERS IN HOME APPLIANCES: A SOFT REVOLUTION**

**L. Perier**

The industrial world is now an arena where many manufacturers produce low cost, high quality products. Cost cutting and outsourcing are no longer sufficient to ensure competitiveness. Creativity and time to market become a must to survive. As a result, traditional electro-mechanical solutions do not fit many new home appliance requirements.

Now microcontrollers (MCUs) offer a strategic advantage for the design of cheap, attractive and environmentally safe products. For example, new MCUs can operate directly from the mains and drive power loads with only few external components. They can reduce the energy consumption, motor size and the cost of the appliance.

In the first chapter, this article analyses the forces driving new appliance concepts. In a second part, it describes state of the art solutions. The third chapter presents MCUs well adapted to home appliances and tools for increasing a designer's productivity.

#### **AN885: ST62 MICROCONTROLLERS DRIVE HOME APPLIANCE MOTOR TECHNOLOGY**

**Bruno Maurice**

Most domestic appliances are driven by an electric motor; for the most part, these motors are controlled in a simple and rudimentary fashion, and electronics is only now beginning to be applied. This article describes the three main motor families – Universal, Induction and Electronically Commutated – as well as the relevant electronic control techniques, now possible thanks to the intrinsic characteristics of SGS-THOMSON's ST62 Family of microcontrollers.

ST62 MCUs, with their wide range of on-chip peripherals, their wide supply voltage range, their built-in ruggedness and their legendary noise immunity allow truly low total system cost, thus favouring the technological advancement of electrical motor design.

Basic electrical topologies are described, together with their associated power and signal electronics. The relative strengths and weaknesses are explored, using practical examples, in order to illustrate the advantages of electronic control using ST62 MCUs.



### **1.1.5 GRAPHICAL DESIGN**

#### **AN676: BATTERY CHARGER USING THE ST6-REALIZER<sup>®</sup>**

**Lionel Picandet**

Because competition becomes greater and greater it is important to reduce time to market. The ST6 Realizer helps to fulfill this duty. The time needed to realize a design is dramatically reduced. Design of an application takes a few days instead of a few weeks.

Users who develop ST6 applications are systems electronics engineers; Often they do not know the assembler well and there are reluctant to use it. The ST6 Realizer allows users to design their applications using symbols known by hardware designers such as comparators, counters, multiplexers. Once the design is over, the ST6 Realizer generates assembly code or executable code for the different ST6 target hardware.

#### **AN677: PAINLESS "MCU" CODE BY GRAPHICAL APPLICATION DESCRIPTION**

**Olivier Rouy**

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<sup>®</sup>, allows a graphical description of the system, automatic code generation, simulation and debugging.

#### **AN839: ANALOG MULTIPLE KEY DECODING USING THE ST6-REALIZER**

**Olivier Rouy**

Design of a multiple key decoder using the A/D converter present on the ST62 MCU. This note describes how the A/D convertor can be used to reduce the number of I/O lines required for key decoding. Software development is carried out using the ST6-REALIZER, and therefore does not involve writing code in assembly language.

### **AN840: CODED LOCK USING THE ST6-REALIZER**

**Olivier Rouy**

Design of a coded security lock. This application uses the EEPROM on the ST62 MCU to store the secret code. Code entry and recognition is performed under software control. Software development is carried out using the ST6-REALIZER, and therefore does not involve writing code in assembly language.

### **AN841: A CLOCK DESIGN USING THE ST6-REALIZER**

**Olivier Rouy**

Design of a clock system. This note provides an example of time management using the Timer embedded in the ST62 MCU. Current time setting and alarm time setting are carried out under software control. Software development is carried out using the ST6-REALIZER, and therefore does not involve writing code in assembly language.

### **AN842: 7 SEGMENT DISPLAY DRIVE USING THE ST6-REALIZER**

**Olivier Rouy**

Design of 7-segment driver functions. This note provides an example of the use of lookup tables for conversion or coding purposes. Single digit and multiple digit display applications are described. Software development is carried out using the ST6-REALIZER, and therefore does not involve writing code in assembly language.

## **1.2 SYSTEM OPTIMIZATION**

### **1.2.1 COST REDUCTION**

A microcontroller can save cost by itself but also by the external components it can save. The notes presented in this chapter show how to take full advantage of the ST62 flexible I/O pins and A/D converter to decrease the component count and provide enhanced features.

AN431 explains how to save pins on the ST62 microcontroller by using only one or two I/O pins which decode a keyboard with the A/D converter. AN594 proposes to use a standard ST62 in a small package to drive LCD displays by modifying the I/O configuration during the operation of the microcontroller. AN672 explains how to optimize the ST62 A/D converter accuracy using a software filter to eliminate the noise coming from the input signal or from the supply. AN673 suggests a simple circuit to reduce the ST62 power consumption when operating with a 32kHz oscillator.

### **AN431: USING ST6 ANALOG INPUTS FOR MULTIPLE KEY DECODING**

#### **J. Stockinger**

The ST6 on-chip Analog to Digital Converter (ADC) is a useful peripheral integrated into the silicon of the ST6 family members. The flexibility of the I/O port structure allows the multiplexing of up to 13/8 Analog Inputs into the converter in a 28/20 pin device for the ST6210/15 2k ROM and ST6220/25 4k ROM families, enabling full freedom in circuit layout. Many other members of the ST6 family also offer the Analog to Digital converter.

One of the more novel and practical applications of this converter, is to decode a number of keys. The technique is to connect the keys by resistive voltage dividers to the converter inputs. An example of key detection using 10 keys is illustrated in this note.

Using the Analog to Digital converter in this fashion does not require a static current and avoids false key detection.

### **AN594: DIRECT SOFTWARE LCD DRIVE WITH ST621X AND ST626X**

**T.Castagnet, J.Nicolai, N.Michel**

This note describes a technique for driving a Liquid Crystal Display (LCD) with a standard ST62 microcontroller, without any dedicated LCD driver. This technique offers a display capability for applications which require a small display at low cost together with the versatile capabilities of the standard ST62xx MCU. Higher display requirements are easily handled by dedicated members of the ST62xx MCU family, for example the ST6240.

The first part of this note describes the typical waveforms required to drive an LCD correctly with a multiplexing rate of 1 or 2 (duplex). The following parts present two solutions based on standard ST62 MCUs driving directly the LCD. The first is based on an ST6215 without using software interrupts and the second on an ST6265 where the LCD is controlled by timer interrupts.

In both examples the program size, the CPU time occupation due to the LCD drive and the number of surrounding components are minimized. Consequently many additional tasks can be added to the MCU program.

### **AN672: OPTIMIZING THE ST6 A/D CONVERTER ACCURACY**

**J.Nicolai**

When using the internal Analog to Digital Converter of the ST62 family and maximum A/D converter accuracy is required, it is desirable to filter out any noise present on the analog input. This includes also noise present on the ground and Vcc supply lines of the MCU as Vcc is also the voltage reference of the A/D converter.

While good supply decoupling with capacitors is always recommended, and placing the ST6 into its WAIT state reduces potential noise induced by the digital switching within the MCU, digital filtering by averaging several successive A/D conversions can improve the accuracy of the conversion.

This is the most effective way to get the most accuracy out of the ST6 family A/D converter.

The code fragment included with this note demonstrates this digital filtering which gives the best results with a trade-off against the total time for conversion.

**AN673:REDUCING CURRENT CONSUMPTION AT 32KHZ WITH ST62****C.Pilon, L.Perier**

In many cases a 32kHz crystal is chosen for the oscillator of the ST62 microcontroller in order to achieve the minimum current consumption in the application.

This note provides a technique for minimising the current consumption when using a crystal oscillator at this frequency.

This short note should be read in conjunction with Application Note AN670, "Oscillator Selection for the ST62".

### 1.2.2 DESIGN IMPROVEMENTS

This section suggests ideas on how to expand the usage of ST62 peripherals and how to increase the design safety. AN420 describes a way to expand the A/D converter resolution of the ST62 up to 10 or 12 bits with few external components. AN435 proposes practical examples on how to maximize the ST62 noise immunity. It describes software and hardware solutions applicable in low cost, noisy applications such as power management or automotive.

The other notes show with examples how to avoid corruption of I/O pins or misprogramming (AN432), how to ensure a safe reset when the power supply changes (AN669), how to match a crystal or a ceramic resonator to an ST62 oscillator (AN670) and how to prevent EEPROM corruption in case of supply variation (AN671).

### AN420:EXPANDING A/D RESOLUTION OF THE ST6 A/D CONVERTER

#### P. Malusardi

Many members of the ST6 Microcontroller Family support an integrated Analog to Digital Converter. This converter allows the analog values produced by external sensors to be converted into digital form to take part in further digital control algorithms.

The standard resolution of the ST6 A/D Converter is 8-bit. Occasionally the analog signals provided require a higher resolution to extract the full dynamic range of the input.

The solution described in this Application Note provides this higher voltage resolution using only an additional Operational Amplifier and a few resistors. The tradeoff in the approach shown is the total conversion time to reach the required resolution.

The technique implemented is that of the Algebraic Adder. A full discussion of the principle of operation is given, with full ST6 source code.

### AN432:USING ST62XX I/O PORTS SAFELY

#### J. Stockinger

All members of the ST62 Series of Microcontrollers feature I/O ports with configurable bit functions. In addition many I/O bits may be set as inputs to the on-chip Analog to Digital Converter. This port bit function is in addition to the normal I/O functions of input (with or without internal pull-up resistor), output (open drain or push-pull) or edge/level selectable interrupt input (with pull-up). This flexibility makes the ST62 series suitable for many industrial control applications (and for many other uses).

This Application Note explains the architecture of the I/O bit associated with these port functions and provides some indications on the correct use of these features for functions such as keyboard scanning and analog inputs. The correct manner to switch between these function is also demonstrated in order to prevent potential malfunctions in operation.

## **AN434:MOVEMENT DETECTOR CONCEPTS FOR NOISY ENVIRONMENTS**

### **H. Sax**

The sales of movement detectors, which react to human-body temperature, are increasing at a fantastic rate.

No Do-it-Yourself shop proposes less than 4 models for sale if it is serious about its image, however the majority of clients are novices who wish to install the system themselves. This installation often causes frustration, partly caused by a lack of knowledge of the operation of the system, but also by the weakness of the products. This weakness can be improved by the use of microcontrollers.

Most movement detectors available, whether using discrete components or integrated circuits, have a similar circuit concept. This Application Note shows concepts on how a microcontroller with analog inputs (the ST6210) can replace discrete components and add additional functionality.

Cost is not an essential factor, but carries a high prejudice against this concept. As shown, the decision to use a Microcontroller with analog inputs carries a series of advantages, together with its logical functionality.

## **AN435:DESIGNING WITH MICROCONTROLLERS IN NOISY ENVIRONMENTS**

Microcontrollers (MCU) make possible the design of integrated and flexible controls for a constantly decreasing cost. As a result, they are spreading rapidly among most electronic applications and especially noise sensitive equipments such as for power control or automotive use.

An MCU operates with sequential logic, so the control of an application can be lost during a disturbance, as with analog control, but also after a power glitch in the system. In addition, a modern MCU includes several tens of thousands of transistors switching in the MHz range, potentially radiating interference of high magnitude in a large frequency spectrum. Consequently, noise sensitivity and generation have to be considered as early as possible in MCU based designs.

This Application note presents numerous methods to effectively reduce noise problems. The first part presents a short overview on noise and proposes hardware solutions to increase the equipment immunity to noise. The second part concerns the writing of software more immune to disturbances. The behaviour versus disturbances of MCUs designed for noisy environments, the ST62 family, is presented. Practical examples and results are shown.

### **AN669:SIMPLE RESET CIRCUITS FOR THE ST62**

**T.Castagnet, J.Nicolai, L Perier**

The circuit schematics shown in this Application Note provide examples of reset circuits for the ST62xx microcontrollers. These circuits range from a very simple solution, which is only efficient at power up, to a circuit providing power up and power down monitoring with a delay at power on.

When used with the watchdog and a software implementation, an efficient and reliable reset of the ST62 can be made.

### **AN670:OSCILLATOR SELECTION FOR ST62**

**C.Pilon, L.Perier**

The purpose of this note is to give indications on how to choose a resonator or a quartz crystal in order to achieve reliable oscillation with the ST62 Microcontroller. This document provides first the major resonator parameters useful for a design. It then proposes measurement methods to ensure a safe oscillation.

### **AN671:PREVENTION OF DATA CORRUPTION IN ST6 ON-CHIP EEPROM**

**C Pilon**

The ST6 Microcontroller has been designed to avoid any potential corruption of data programmed into its on-chip EEPROM (when available). Data integrity can be ensured as long as the application designer follows the guidelines provided in this note.

In general, EEPROM data corruption occurs whenever the reset signal is not controlled when the power supply goes up or down. This is particularly true with a slow ramp-up and/or slow fall time of the power supply, since the device may be in a supply voltage area when the device functionality is not guaranteed for a long time.

If no special care is taken during the power up sequence regarding the reset signal then the microcontroller may start writing into the EEPROM. The same behaviour can be present upon a power down.

This note proposes two complementary solutions to prevent these unwanted actions, a software solution and a hardware solution.



**AN911:ST6 MICRO IS EMC CHAMPION****David Jacquinod & Edouard Presson**

Since January 1996, Electro-Magnetic Compliance is required by international law for any electrical equipment that is manufactured including a printed circuit board. As early as 1991, SGS-THOMSON took this change in the law into account when planning the design and manufacture of the ST62 microcontroller family. An EMC environment was installed in the Design, Quality and Engineering center and ST quality standards were enlarged to include EMC performance criteria, with the result that the ST62 microcontrollers meet the EMC standards five years ahead of most 8-bit microcontrollers.

This short article describes the characteristics of the ST6 microcontroller in the EMC context and how this benefits the customer.

**AN975:UPGRADING FROM ST625X/6XB TO ST625X/6XC****Microcontroller Application Team**

As part of a process of continuous improvement, STMicroelectronics has replaced all ST62T5XB and ST62T6XB devices by ST62T5XC and ST62T6XC.

This opportunity was taken to include new features such as the Low Voltage Detector (LVD) for safe reset, the Oscillator Safeguard (OSG) and a new RC oscillator.

This application note details these new features and draws attention to some precautions that it is mandatory to take when upgrading an application developed with B revision to C revision devices. The first part of the document is related to the silicon itself and the second one to the development tools.

**AN1015:ST6 SOFTWARE TECHNIQUES FOR IMPROVING EMC PERFORMANCE****A. Niaussat**

A major contributor to improved EMC performance in microcontroller-based electronics systems is the design of hardened software. To achieve this goal, EMC considerations must be included as early as possible in the design phase of the project. A quality approach to software increases the security and the reliability of the application. EMC-hardened software is inexpensive to implement, it improves the MCU's immunity performance and saves hardware costs. This application note describes how to implement preventive tricks and active detection in the application software.

### 1.2.3 PERIPHERAL OPERATIONS

This chapter explains with practical examples how to use the ST62 Autoreload timer for PWM generation, input capture and PLL generation.

#### AN590: PWM GENERATION WITH ST62 AUTO-RELOAD TIMER

**J.Nicolai**

This note presents how to use the ST62 8-bit Auto-reload Timer (ARTimer) for the generation of a Pulse Width Modulated (PWM) signal tunable in frequency and duty cycle.

Two examples of this are shown, the first with a specific frequency and duty cycle, and the second with the generation of a 30kHz PWM signal with the duty cycle proportional to an analog voltage converted through the on-chip Analog to Digital Converter.

An introduction to the generation of PWM using the timer, and the software for the examples are provided.

#### AN591: INPUT CAPTURE WITH ST62 AUTO-RELOAD TIMER

**J.Nicolai**

This note presents how to use the 8-bit Auto-reload Timer (ARTimer) of the ST62 to measure time duration or frequency of an input signal.

The Capture Mode with reset is used to measure the time elapsed between two edges of an input signal: two rising edges, two falling edges, or one rising edge and one falling edge if the configuration of the ARTimer is modified after the first edge is detected.

The minimum duration of one signal to measure depends on the microcontroller clock and on the required precision. With an 8MHz quartz crystal, a signal of 8µs duration can be measured with a resolution of 1/64.

A software example is provided.

#### AN592: PLL GENERATION USING THE ST62 AUTO-RELOAD TIMER

**J.Nicolai**

This note describes how to generate a digital signal locked in phase and frequency (PLL) with a calibrated delay starting from an active edge on the 8-bit Auto-reload timer (AR Timer) input pin.

An example is given for a digital input signal of 15kHz presented to the ARTimer input pin. A phase-locked signal at 15kHz with a falling edge delayed 19µs from the input rising edge, and a duty cycle of 75%, is generated.

An explanation of the function and software for the function are provided.

### **AN593:ST62 IN-CIRCUIT PROGRAMMING**

This note provides information on the steps required in order to perform in-circuit programming of ST62Exx EPROM or OTP devices for both on-chip EPROM and EEPROM (where available).

In-circuit programming is possible if the relevant pins of the programming socket located on the ST62 EPROM Programming tool (either an ST6 Starter Kit, Remote Programming board, or Gang Programmer) are connected to a 16-pin connector (8x2 header), which must be provided on the application by the user.

Note: In-circuit programming embedded in program test is not possible. If the EPROM programmer cable is connected to the application, the RESET signal for example is tied to Ground before and after programming.

Connections are shown for the ST62E1x/2x, ST62E4x and ST62E6x/E9x and the corresponding OTP devices.

### **AN678:LCD DRIVING WITH ST6240**

#### **Olivier Rouy**

This application note describes the basic guidelines to achieve a fast and efficient LCD drive application development.

The alphanumeric LCD panel of the ST624x Starter-Kit is used as example and more general concerns are highlighted.

Hardware and software issues are described to demonstrate the benefits brought about while using a ST62 LCD driver.

### **AN913:PWM GENERATION WITH ST62 16-BIT AUTO-RELOAD TIMER**

#### **Microcontroller Division Application Team**

The 16-bit Autoreload timer (ARTimer) is a 16-bit downcounter timer with prescaler. It includes auto-reload PWM, capture and compare capability with two input and two output pins. This note presents how to use the ST62 16-bit Auto-Reload Timer (ARTimer) for generating a DTMF signal (Dual-Tone Multiple Frequency) with the PWM. In the example shown, the PWM output pin generates a DTMF to dial a telephone number.

### **AN914:USING ST626X SPI AS UART**

#### **Microcontroller Division Application Team**

This note shows how to use the ST626x SPI to perform UART serial communication. The operating principles and limitations are described. An example is developed for reception and transmission at 9600 baud, however, baud rates up to 19200 can be obtained. The assembly source code of the example is provided.

### **AN1016:ST6 USING THE ST623XB/ST628XB UART**

#### **Microcontroller Division Application Team**

This brief note describes the problem/solution for managing potential spurious UART interrupts during reset. A short descriptive paragraph outlines the problem. This is followed by a table of values for inserting in the application program covering a range of baud rates. A short assembly code example is

provided.

### **AN1050:ST6 INPUT CAPTURE WITH ST62 16-BIT AUTO-RELOAD TIMER**

#### **Microcontroller Division Application Team**

This note presents how to use the ST62 16-bit Auto-Reload Timer (ARTimer) to measure durations or frequencies of an input signal. An example shows how to capture an input signal to make an output signal with the same frequency as input signal but with a duty cycle equal to 50%. The ARTimer has a 16-bit downcounter timer with prescaler. It includes auto-reload PWM, capture and compare capability with two input and two output pins.

### **AN1127:USING THE ST62T6XC/5XC SPI IN MASTER MODE**

#### **Microcontroller Division Application Team**

To avoid problems when switching from Rev B to Rev C devices, special attention must be paid to the programming of the I/O port Data Direction Register when using the SPI in master mode.

In Rev B devices, the SPI functions in master mode without depending on whether the DDR bit for the I/O port pin (PC4) used for the SCK is set or reset (programmed as input or output).

In Rev C devices, the DDR bit has to be programmed as INPUT (left at the reset value).

## **2 ST7 FAMILY**

### **2.1 PROGRAMMING AND TOOLS**

#### **AN978:KEY FEATURES OF THE STVD7 ST7 VISUAL DEBUG PACKAGE**

##### **Microcontroller Division Application Team**

The STVD7 is the brand new debugger developed by STMicroelectronics, which replaces the WGDB7 and which is still free of charge.

This is an IDE (Integrated Development Environment) which means that the same graphical Windows interface can be used for both editing and debugging.

The purpose of this application note is to explain how to get started with this ST Visual Debugger and the ST Assembly tool chain and to describe the main features of the STVD7. Similar application notes have been written for the STVD7 and ST7 C Compilers (developed by COSMIC and HIWARE).

The following operating systems are supported: Windows 95, 98, NT, 2000 (soon).

#### **AN983:KEY FEATURES OF THE COSMIC ST7 C-COMPILER PACKAGE**

##### **Microcontroller Division Application Team**

COSMIC Software is a privately-owned company founded in 1983 in Paris, France by Dr. Maurice Fathi and Jean-Pierre Lavandier, two engineers experienced in UNIX systems and embedded development tools. COSMIC develops among other things C compilers and debuggers for ST7 8-bit microcontrollers.

The purpose of this application note is to explain how to get started with the C COSMIC tool chain and the ST Visual Debugger developed by ST (STVD7) or the IDE (Integrated Development Environment) developed by COSMIC (IDEA: Integrated Development Environment for Embedded Applications) using ST7 microcontrollers.

The following operating systems are supported: Windows 95, 98, NT.

#### **AN985:EXECUTING CODE IN ST7 RAM**

##### **Microcontroller Division Application Team**

Using the ST72251 as an example, this application note describes how to execute programs in the on-chip RAM area of the ST7. The code to be executed can be copied from the ROM area or loaded from an external device such as a host system or serial device such as E<sup>2</sup>PROM. With the aid of flowcharts and an example source in assembly language, this document explains the essential steps required: linking, copying the code and calling the program.

### **AN986:USING THE ST7 INDIRECT ADDRESSING MODE**

#### **Microcontroller Division Application Team**

The ST7 assembly language instruction set includes the indirect addressing mode (indexed or not indexed) for short and long variables. This document shows using examples how using the indirect addressing mode allows the programmer to write more compact code in both Assembly and C language programs.

### **AN987:ST7 IN-CIRCUIT PROGRAMMING**

#### **Microcontroller Division Application Team**

This application note describes the advantages of In-Circuit Programming vs. programming on an EPROM programming board. It also documents how to implement In-Circuit Programming targeting most of the ST7 general purpose microcontrollers. These devices are all the ST7 MCUs supported by the ST7 starter kits and ST7 EPBs in the MDT1, MDT3 and MDT4 tool families.

### **AN988:STARTING WITH ST7 ASSEMBLY TOOL CHAIN**

#### **Microcontroller Division Application Team**

This document gives guidelines on how to start an ST7 application design based on the ST7 Assembly tool chain. The ST7 tool chain is a DOS or UNIX hosted cross development system for ST7 microcontroller based applications. The application note describes the use of the tool chain in the DOS environment. An overview of the tool chain is given and the Assembler options are described.

### **AN989:STARTING WITH ST7 HIWARE C**

#### **Microcontroller Division Application Team**

This document gives guidelines on how to start an ST7 application design based on the HIWARE C Compiler chain. A description is given of how to set up a project using the Hi-Cross C Compiler combined with the WINEDIT editor/project management tool integrated in the Hi-Cross package. The ST7 Hi-Cross/Hi-Light tool chain is a cross development system for ST7 microcontroller applications from HIWARE A.G.

**AN1039:ST7 MATH UTILITY ROUTINES****Microcontroller Division Applications**

The goal of this application note is to present the following mathematical routines:

- division of two 8-bit numbers
- multiplication of two 16-bit numbers
- division of one 32 bit number by a 16-bit one (result stored into a word)
- addition of two 16-bit numbers
- subtraction of two 16-bit numbers
- test if a 16 bit number value is within a predefined range
- binary to decimal conversion

In this application, the MCU used is a ST72251.

**AN1064:WRITING OPTIMIZED HIWARE C LANGUAGE FOR ST7****Microcontroller Division Applications**

The purpose of this note is to present how to write an optimized C software application for an ST7-based embedded system. The main topics focus on how to write C source code that generates the smallest code and data size. To reach this goal some specific C language extensions have to be used like compiler options and pragmas.

### **AN1106:TRANSLATING ASSEMBLY CODE FROM HC05 TO ST7**

#### **Microcontroller Division Application Team**

This application note has been written to help users translate their HC05 assembly source code into ST7 source code.

Even if both assembly languages are quite similar, the philosophy and program structure are quite different.

A software translator ("migr2st7") has been developed by STMicroelectronics and is available on the MCU ON CD.

For more information on the ST7 Assembly Tool Chain, please refer to the application note "Starting with ST7 Assembly Tool Chain" (AN988) and the software library available on Internet ([www.st.com/product/support](http://www.st.com/product/support)) or the ST7 CD ROM.



## 2.2 EXAMPLE DRIVERS

### **AN969:ST7 SCI COMMUNICATION BETWEEN THE ST7 AND A PC**

#### **Microcontroller Division Application Team**

This application note shows how to implement standard RS232 protocol serial communications between an ST7 microcontroller and a PC. The various sections of the document describe the RS232 protocol, the ST7 SCI interface, how to configure the SCI, setting up hardware and writing the driver routines for initializing the communication parameters and performing data exchange. A source assembly listing is provided at the end of the document.

### **AN970:ST7 SPI COMMUNICATION BETWEEN THE ST7 AND E<sup>2</sup>PROM**

#### **Microcontroller Division Application Team**

This application note gives a useful example of communication using the ST7 SPI peripheral. Simple communication between an ST7 microcontroller and a SPI E<sup>2</sup>PROM is implemented by performing, through SPI, a write in the memory, followed by a read of the written data. The hardware interfacing, software initialization and communication protocols are described and illustrated with schematics and flowcharts. A section describes how to address several E<sup>2</sup>PROM devices from the ST7. An assembly listing is provided at the end of the document.

### **AN971:ST7 I<sup>2</sup>C COMMUNICATION BETWEEN THE ST7 AND E<sup>2</sup>PROM**

#### **Microcontroller Division Application Team**

This application note presents a practical example of a communication using the I<sup>2</sup>C peripheral of the ST7. It describes a basic single master communication between an ST7 microcontroller and an I<sup>2</sup>C bus E<sup>2</sup>PROM. The purpose is to execute, from the ST7 through the on-chip I<sup>2</sup>C interface, a write and a read in the external E<sup>2</sup>PROM without error management. The ST7 I<sup>2</sup>C peripheral allows multi master and slave communication with bus error management. In this application, only single master mode is used without error management. As polling mode is more difficult to implement, the application is based on this mode, but it can be easily adapted to interrupt mode. An assembly source listing is provided.

### **AN972:ST7 SOFTWARE SPI MASTER COMMUNICATION**

#### **Microcontroller Division Application Team**

This application note presents a basic software driver for emulating SPI full duplex communication in master mode using the ST7 standard I/O ports. The principles of the SPI (Serial Peripheral Interface) are briefly introduced and an algorithm for 8-bit full duplex communication is described. A source assembly listing is provided at the end of the document.

### **AN973:SCI SOFTWARE COMMUNICATION WITH A PC USING ST72251 16-BIT TIMER**

#### **Microcontroller Division Application Team**

The Serial Communication Interface (SCI) offers a flexible means of full-duplex data exchange with external equipment requiring an industry standard NRZ asynchronous serial data format. This document shows how to emulate SCI communication by software, using the ST7 timer. The application presented is for RS232 communication between an ST7 microcontroller and a PC. Initialization, interrupts and receive and transmit routines are described with the aid of diagrams and flowcharts. A source assembly listing is provided at the end of the document.

### **AN974:REAL TIME CLOCK WITH THE ST7 TIMER OUTPUT COMPARE**

#### **Microcontroller Division Application Team**

This note explains how to use the ST7 Timer output compare function. The application example presents a real time clock with second, minute and hour counters based on a fixed time base. Flowcharts describe hardware configuration, initialization and register updating procedures. A source assembly code listing is given at the end of the document.

### **AN976:DRIVING A BUZZER USING THE ST7 PWM FUNCTION**

#### **Microcontroller Division Application Team**

This "musical" application describes how to use the ST7 PWM to generate synthesized music using a buzzer. The document covers musical score, note duration and tone generation and volume control. A source assembly listing is provided at the end of the document.

### **AN979:DRIVING AN ANALOG KEYBOARD WITH THE ST7 ADC**

#### **Microcontroller Division Application Team**

This application note presents a standard example of the use of the Analog to Digital Converter (ADC) of the ST7. The ST7 on-chip ADC is used to emulate a 16-key analog keyboard. The hardware interfacing techniques are outlined in the first part of the document and the software is described by means of flowcharts. An assembler source listing is given at the end of the document.

### **AN980:ST7 KEYPAD DECODING TECHNIQUES, IMPLEMENTING WAKE-UP ON KEYSTROKE**

#### **Microcontroller Division Application Team**

The goal of this application note is to present an example of the use of ST7 HALT mode. In this application, the MCU (here a ST72251) is woken up by an external interrupt caused by pressing a key on the 4x4 matrixed keypad.

**AN1017:USING THE ST7 USB MICROCONTROLLER****Microcontroller Division Application Team**

The ST7 USB interface is a Universal Serial Bus peripheral that provides a means of connecting a PC peripheral serving as a function to a PC host. It supports low speed data transfers. This application note describes an example firmware for interaction with the USB interface hardware and support interactions between a USB device and a host system. The associated source code of the firmware is available.

**AN1041:USING ST7 PWM SIGNAL TO GENERATE ANALOG OUTPUT (SINUSOID)****Microcontroller Division Application Team**

This note shows how to use the ST7 PWM/BRM to generate a 50Hz sinusoid that can be tuned both in average and amplitude. This application has been done using an ST72511R4.

**AN1042:ST7 ROUTINE FOR I<sup>2</sup>C SLAVE MODE MANAGEMENT****Microcontroller Division Application Team**

This application note presents a useful example of communication using the I<sup>2</sup>C peripheral of the ST7. The ST7 microcontroller is used as a slave and can communicate with any master. This slave, through the I<sup>2</sup>C interface, receives words from the master implementing error management and returns them. This application has been implemented with a ST72E251 and using 7-bit addressing mode.

**AN1044:MULTIPLE INTERRUPT SOURCES MANAGEMENT FOR ST7 MCUS****Microcontroller Division Application Team**

The goal of this application note is to present a technique for managing several external I/O interrupts with a member of the ST7 series of MCUs (here a ST72251).

**AN1045:ST7 SOFTWARE IMPLEMENTATION OF I<sup>2</sup>C BUS MASTER****Microcontroller Division Application Team**

This application note implements an I<sup>2</sup>C communications software interface that can be used in any general-purpose ST7 device without specific I<sup>2</sup>C on-chip peripheral hardware. The program is written in C language. It implements the I<sup>2</sup>C master transmitter and master receiver functions. The ST7 acts as the bus master and communicates via the I<sup>2</sup>C bus to a slave EEPROM device.

### **AN1046:ST7 UART EMULATION SOFTWARE**

#### **Microcontroller Division Application Team**

All members of the STMicroelectronics ST7 Series of Microcontrollers feature a 16 bit timer with several possibilities such as output compares and input captures.

This note describes a technique for emulating an RS232 UART with the ST7 timer without any additional hardware. Only two pins are required for the serial communication.

The first part of this note will explain the protocol used for serial communication and how to adapt it for the ST7 timer. The other sections of this note describe more precisely how the program deals with transmitter mode and receiver mode.

Timings are used to illustrate the important points.

The user can easily adapt the example to his own application as only a small amount of code is required by the UART program.

The software was tested by connecting a ST72251 to the serial port of a PC and communicating in all possible modes.

### **AN1047:MANAGING RECEPTION ERRORS WITH THE ST7 SCI PERIPHERAL**

#### **Microcontroller Division Application Team**

This application note provides guidelines for managing communication errors with the ST7 Serial Communications Peripheral (SCI) in reception mode. It describes how the ST7 SCI peripheral works when errors occur. A explanation is given of how to interpret the various error flags and to determine if the received byte is corrupted or not. An example interrupt service routine written in assembly language is provided at the end of the document.

### **AN1048:ST7 SOFTWARE LCD DRIVER**

#### **Microcontroller Division Application Team**

This note describes a technique for driving a Liquid Crystal Display (LCD) with any standard ST72 Microcontroller i.e without any specific on-chip LCD driver hardware. This technique offers a solution for applications which require a small display at low cost together with the versatile capabilities of the ST72 MCU. The first solution uses the ST7 timer output compare feature to generate the LCD timing. The second solution targets low power applications, switching the ST7 into Halt mode between two I/O refreshes. An external RC circuit is used to wake up the ST7 using an external interrupt.

**AN1078:ST7 TIMER PWM DUTY CYCLE SWITCH FOR TRUE 0% or 100% DUTY CYCLE****Microcontroller Division Application Team**

This application note presents a program that uses the 16-bit timer of the ST7 in PWM output mode. The program can be used to perform a hot switch from one duty cycle to another and obtain a true fixed period and true duty cycle percentage values between 0% and 100%. The example program has been developed for the ST7GP family (ST72251G1 and G2).

**AN1148:USING THE ST7263 FOR DESIGNING A USB MOUSE****Microcontroller Division Application Team**

This application note describes the implementation of a cost-effective USB Mouse using the ST7263 microcontroller. A detailed description of low-consumption power management mode (resume mode) is given in section 5.

ST provides a complete architecture as well as firmware drivers to help you develop your application. A list of reference documents is provided at the end of the application note. It is assumed that the reader is familiar with the ST7263 microcontroller and USB.

**AN1149:HANDLING SUSPEND MODE ON A USB MOUSE****Microcontroller Division Application Team**

All USB devices must support Suspend mode. Suspend mode enables the devices to enter low-power mode if no bus activity is detected for more than 3.0 ms.

Like USB keyboards and pointing devices, USB mice must be able to exit Suspend mode if a button has been pressed or if a movement has been detected. This feature is called Remote wake-up mode. A Remote wake-up involves a Resume sequence on the USB lines and recovery of communication between the mouse and the host.

The following application note describes the implementation of Suspend and Remote wake-up modes on a USB mouse using the ST7263 microcontroller. The first chapter focuses on the recommendations before entering Suspend mode. Then a description of the RC external circuit for handling Remote wake-up mode is detailed. It contains power management recommendations and RC value proposals. The third chapter describes Resume mode. Then chapter 4 and chapter 5 describe software implementation and program flow.

It is assumed that the reader is familiar with the ST7263 microcontroller and USB.

### **AN1082:DESCRIPTION OF THE ST72141 MOTOR CONTROL PERIPHERAL REGISTERS**

#### **Microcontroller Division Application Team**

The ST72141 is designed for controlling Brushless Permanent Magnet DC motors with or without sensors.

The motor control is performed by the hardware of the on-chip Motor Control peripheral (MTC). The MTC is functionally divided into four parts.

- Zero-crossing and End of Demagnetisation detector
- Delay manager
- PWM manager
- Channel manager

### **AN1083:ST72141 BLDC MOTOR CONTROL SOFTWARE AND FLOWCHART EXAMPLE**

#### **Microcontroller Division Application Team**

The software examples described in this application note are those generated by the ST7MTC1 Kanda kit. 80% of the code is generic, the remaining 20% is specific to the implementation of the ST72141 in the ST7MTC1 kit (user interface and communication).

The purpose of this application note is to give the flowcharts of the software examples showing how to drive the motor in both current and voltage modes and give examples of open loop or closed loop speed regulation.

The software examples given in the file attached with this Application note illustrate Current mode and closed loop driving mode for a 4-pole BLDC motor.

**AN1129: PWM MANAGEMENT FOR BLDC MOTOR DRIVES USING THE ST72141****Microcontroller Division Application Team**

Brushless DC motors are efficient, quiet and can provide a very high starting torque, this is partly due to the built-in permanent magnet. For these reasons they are used more and more in a large range of applications like hard disk drives, fans, pumps, compressors, etc... However brushless DC motors run properly only as synchronous motors: they need electronic circuitry to run, including rotor position sensors, switching devices and a control unit.

In most cases the switching devices are MOSFET transistors or IGBTs and are organized in a three-phase bridge with free-wheeling diodes. Traditionally Hall sensors are used by the control unit to detect rotor position before changing motor coil to which the power applied.

STMicroelectronics has introduced the ST72141 microcontroller which is based on an industry standard architecture and is especially designed for driving BLDC motors. One of the major advantages of the ST72141 is that, without using sensors, it is capable of controlling motors precisely by reading the Back Electromotive Force (BEMF). Instead of using expensive Hall sensors, three resistors connecting the three windings directly to the ST72141 input ports, provide the microcontroller with the rotor position information. The microcontroller replaces the standard control unit and the rotor position sensors. However, the ST72141 can be used with sensors as well.

Although using sensorless mode has big advantages in terms of cost and size, it makes the motor drive a little bit more complicated. The purpose of this application note is to explain in which cases the ST72141 motor control unit can directly read the BEMF voltage and how to quickly set up its control registers in order to use all the advanced features of this product.

### **AN1130:AN INTRODUCTION TO SENSORLESS BRUSHLESS DC MOTOR DRIVE APPLICATIONS WITH THE ST72141**

#### **Microcontroller Division Application Team**

Electric motors are an essential component of our industrialised society with no less than 5 billion motors built world wide every year.

Brushless DC motors are already used in hard disk drives and many industrial applications, and their market share is growing significantly in automotive, appliance and industrial applications.

The ST72141 has been developed by STMicroelectronics to control synchronous motors or, more specifically, 3-phase brushless DC motors. The most common applications of this type of motor are industrial control, automotive equipment, refrigerators, air conditioners, compressors and fans, where brushless DC motors are already used due to their high efficiency, silent operation, compact form, reliability and longevity.

The ST72141 devices are members of the ST7 microcontroller family designed specifically for motor control applications and including A/D converter and SPI interface capabilities. They include an on-chip peripheral for control of electric brushless DC motor either in sensor or sensorless mode.



## **AN1180:USING THE ST7263 KIT TO IMPLEMENT A USB GAME PAD**

### **Microcontroller Division Application Team**

The game pad described in this application note is a low speed, self powered device. It has digital and analog capabilities for the X and Y axes, and includes 10 buttons and two motors for vibration.

In order to demonstrate the use of the ST7263 microcontroller, we adapted this game pad to the ST7263 demo kit to make a USB game pad, with USB mouse and hotkey functions.

This application uses the key features of the ST7263 USB microcontroller, the analog converter, the 3 USB Endpoints (control Endpoint, interrupt IN, interrupt OUT) and PWM capabilities with the two output compare waveforms on Port A.

The source code of the software described on this application note is available from ST. As with other USB devices, Endpoint 0 is the control Endpoint used for device enumeration. Endpoint 1 is an interrupt IN Endpoint, this allows the device to send data to the PC. This Endpoint is shared using the reportID capabilities specific to the HID class. In this case, the first byte sent to the PC is the reportID number. In this demo application we use three reportIDs:

- reportID number 1 is used for the game pad data
- reportID number 2 is used for the mouse data
- reportID number 3 for the hotkey function

Endpoint 2 is an interrupt OUT Endpoint to send data from the PC to the device. This Endpoint is used to control the motors for the vibration feature.

## **AN1182:USING THE ST7 USB LOW-SPEED FIRMWARE**

### **Microcontroller Division Application Team**

This application note describes how to use the ST7\_USB firmware. This firmware, written in C, using the Hiware C compiler, provides a complete USB protocol layer for low-speed USB microcontrollers (such as the ST7263, ST7262 and ST72774). The source code is available free to STMicroelectronics customers.

### 2.3 PRODUCT OPTIMIZATION

#### **AN982:USING CERAMIC RESONATORS WITH THE ST7**

##### **Microcontroller Division Application Team**

The goal of this application note is to show, using results obtained by Murata, that ceramic resonators can be used instead of quartz crystals. Ceramic resonators are cheaper than quartz crystals and as some resonators have built-in capacitors, so they allow you to use less components. The results described here have been obtained for the ST72251 but are also available for ST72101, ST72121, ST72212, ST72213, ST72221, ST72331 and ST72311.

#### **AN1014:HOW TO MINIMIZE THE ST7 POWER CONSUMPTION**

##### **Microcontroller Division Application Team**

This document presents a way of minimising the ST7 power consumption for low power applications. This note is based on the ST72311, but is applicable to all ST7 general purpose devices. Use of ST7 Slow mode, Wait Mode and Halt mode is discussed and tables with examples of power consumption measurements are given.

#### **AN1070:ST7 CHECKSUM SELF-CHECKING CAPABILITY**

##### **Microcontroller Division Application Team**

The goal of this application note is to present a software technique for determining if data and program in EPROM have been corrupted and if so not to run the user's program. The program described in this application note has been written for the ST7GP family (ST72101G1 and G2, ST72121J2, ST72212G2, ST72213G1, ST72251G1 and G2, ST72311N2 and ST72331N2). In this application, we chose to use a ST72251G2.

#### **AN1179:PROGRAMMING ST7 FLASH MICROCONTROLLERS IN REMOTE ISP**

##### **Microcontroller Division Application Team**

This application note is divided into two parts. The first part describes the ISP and FLASH programming specifications for the following ST7 devices: ST72C104, ST72C124, ST72C171, ST72C215, ST72C216, ST72C254, ST72C314, ST72C334, ST72C411 (supports two ISP protocols, refer to the datasheet).

The second part of this application note gives an example of how to use the ISP protocol to program the FLASH memory and the option bytes of a ST72C254, using another ST7 as a programming tool.

## 2.4 PRODUCT EVALUATION

### AN910:ST7 AND ST9 PERFORMANCE BENCHMARKING

#### A. Albella, G. Bouvier and J. Pauvert

STMicroelectronics has developed a set of test routines related to 8-bit and low-end 16-bit microcontroller applications to evaluate computing performance and interrupt processing performance of microcontroller cores. These routines have been implemented on ST7 and ST9 Microcontroller Units (MCUs) as well as several MCUs available on the market.

The routines have been written in assembler language to optimize their implementation and focus on core performance, without being dependent upon compiler code transformation.

For each test, the two parameters of interest are execution time and code size. Timings have been either measured whenever possible, or theoretically calculated when there was no other alternative. In most cases, programs have really run and execution times have actually been measured, so that assembly sources should not contain implementation errors and results can be considered as correct and reliable.

The results of this study point out the capability of the ST9+ to compete with 16-bit MCUs on 8-bit and low-end 16-bit applications and confirms its position of high-end 8/16-bit MCU. It also confirms the ST7 as an outstanding 8-bit MCU.

### AN990:ST7 BENEFITS VERSUS INDUSTRY STANDARD

#### Microcontroller Division Application Team

This note presents, from the application developer's point of view, the main advantages of the ST7 core over the corresponding industry standard architecture in terms of application cost, speed and flexibility. The ST7 enhancements discussed include the Y Index register, Indirect memory access mode, Stack pointer access, PUSH/POP instructions, SWAP instruction and interrupt vectors.

### **AN1086:ST7 / ST10U435 CAN-do SOLUTIONS FOR CAR MULTIPLEXING**

**L. Perier / A. Coen**

Replacing a classical harness with a multiplexing (mux) network makes cars more competitive as it increases their flexibility and simplifies the wiring. CAN is the leading protocol for car mux systems thanks to its large speed spectrum and noise immunity. But each application has specific constraints in terms of protocol, cost and performance. So a single node architecture does not fit all the needs.

This article compares first of all, the major car mux protocols: CAN, J1850 and SCI/UART. The second part describes optimized nodes including microcontrollers (ST725x, ST92F120, ST10F167) with embedded FLASH/ROM and physical line interfaces (U435). Then it presents roadmaps for MCU cores, embedded FLASH memories and super-integrations.

### **AN1150:BENCHMARK ST72 VS PC16**

#### **Microcontroller Division Application Team**

This document presents the results of a competitive analysis between the STMicroelectronics ST72254 and the Microchip PIC16F876. These two microcontrollers (MCUs) have been chosen for comparison because they are in a similar performance category and were introduced on the market at the same time.

The comparison of the two MCUs is divided into two major parts. First the cores, with a comparison of their architecture including performance benchmarks. These benchmarks are based on assembler and C routines that are representative of typical microcontroller applications. The second part examines the peripherals in terms of their functionality and to what extent they off-load the core and the driver software.

Finally, you will find a table summarizing the weak and the strong points of each MCU.

Two files are appended to this document, you can find them in our Web server ([mcu.st.com](http://mcu.st.com)) in the application note section. The first one entitled "Performance comparison between ST72254 and PIC16F876" includes the results given in this document plus the description of the source and the compilation options used. This file was created in order to allow you to easily reproduce the benchmark. The second file regroups all the source files used.

The information on the PIC16F876 is based on the Microchip datasheet: DS30292A.PDF

**AN1151:PERFORMANCE COMPARISON BETWEEN ST72254 & PC16F8****Microcontroller Division Application Team**

STMicroelectronics has developed two sets of test routines related to 8-bit and low-end 16-bit microcontroller applications to evaluate the computing performance of microcontroller cores. These routines have been implemented on ST72254 and PIC16F876 Microcontroller Units.

The first set of routines has been written in assembler language to optimize their implementation and focus on core performance, without being dependent upon compiler code transformation.

The second set tries to evaluate the performance of the two MCUs and their respective C compilers. This benchmark uses a C language program, representative of an automotive application. The C compilers used were from Hiware on the ST72 and from Hi-Tech on the PIC16.

The speed of the two MCUs has been compared in two ways:

- Firstly, at the maximum frequency commercially available on each MCU. this means at an external frequency of 16MHz on the ST72 and of 20MHz on the PIC16.
- Secondly, at the same current consumption level (10mA).

\* this value is determined by interpolation

As we can see, to reach the same consumption level on the two MCUs, the PIC's running frequency must be lowered to 10Mhz (ext.) and the ST72 can keep its maximum frequency of 16MHz (ext.).

### 3 ST9 FAMILY

#### 3.1 APPLICATION EXAMPLES

##### AN413:INITIALIZATION OF THE ST9

###### **P. Guillemin**

The ST9 family offers the microprocessor designer a wide variety of architectural features and peripheral units fully configurable to the user's specific application requirements.

Configuration is typically implemented by simple software routines included in the power-on- or system- reset routines. The sole difficulty which the user may initially encounter stems, in fact, from the power and versatility of this approach to system design. The large number of available options means that the user must specify a large number of system parameters by initializing control register contents for the specific peripheral units.

The objective of this Application Note is to suggest to the user a programming structure and philosophy to aid in the initial configuration of the system. The approach is illustrated by a number of specific examples selected from the wide range available.

##### AN415:USING THE I2C-BUS PROTOCOL

###### **M. Chabaud, A. Dunworth**

The Serial Peripheral Interface (SPI) included in all ST9 family members has been designed to handle a wide variety of serial bus protocols, including SBUS, IMBUS, and I2C-bus.

Certain features of the popular I2C-bus serial communication standard have not been directly implemented in hardware, but may be realized with simple software routines, based on the SPI contained in the standard ST9 core.

This Application note gives an example of such routines, suitable for interfacing the ST9 with a serial memory device.

**AN421: STACK OVERFLOW DETECTION USING THE ST9 TIMER WATCHDOG****P. Guillemin**

In real time applications, the implementation of software protection is not always easy, but allows reaching a high security level for the software against malfunction. This is particularly true for in-board applications in disturbed environments, such as automotive, power meter or industrial applications.

To help avoid non-controlled functionality and damage to real time system due to possible perturbations on the ST9  $\mu$ C core and I/O port, a special peripheral able to act as a watchdog is available on all the ST9 family members: the Timer Watchdog.

A periodic restarting of the Timer Watchdog by program, associated with the automatic detection of possible stack overflow, add to the protection of real time application software.

This application note shows how to detect stack overflow by using the Timer Watchdog in watchdog mode.

**AN910: ST7 AND ST9 PERFORMANCE BENCHMARKING****A. Albella, G. Bouvier and J. Pauvert**

STMicroelectronics has developed a set of test routines related to 8-bit and low-end 16-bit microcontroller applications to evaluate computing performance and interrupt processing performance of microcontroller cores. These routines have been implemented on ST7 and ST9 Microcontroller Units (MCUs) as well as several MCUs available on the market.

The routines have been written in assembler language to optimize their implementation and focus on core performance, without being dependent upon compiler code transformation.

For each test, the two parameters of interest are execution time and code size. Timings have been either measured whenever possible, or theoretically calculated when there was no other alternative. In most cases, programs have really run and execution times have actually been measured, so that assembly sources should not contain implementation errors and results can be considered as correct and reliable.

The results of this study point out the capability of the ST9+ to compete with 16-bit MCUs on 8-bit and low-end 16-bit applications and confirms its position of high-end 8/16-bit MCU. It also confirms the ST7 as an outstanding 8-bit MCU.

**AN1069: ADDRESSING UP TO 4 MBYTES OF MEMORY FROM A ST9+ WITH A 16-BIT EXTERNAL BUS****Microcontroller Division Application Team**

This application note is to help developers of ST9+ applications that need to address external memory. It refers to ST9+ microcontrollers which have only a 16-bit external address bus (such as the ST90158). The information in this application note does not apply to ST9+ microcontrollers that have a 22-bit external address bus.

### **AN1075: USING THE ST9+ MEMORY MANAGEMENT UNIT (EXAMPLES FOR ST92195 & ST92R195)**

#### **Microcontroller Division Application Team**

This application note describes techniques for creating software applications using the Memory Management Unit (MMU) of the ST9+. In addition, it provides useful hints on using the ST9+ C Compiler. A description of the main characteristics of the ST9+ MMU is given. Then, the C compiler is briefly described, emphasizing the Memory Management Unit aspects. Finally, the subject matter is developed using examples for a ROMless and a ROM microcontroller, the ST92R195 and the ST92195 respectively.

### **AN1076: ST9+ EXTERNAL MEMORY INTERFACE CONFIGURATION**

#### **Microcontroller Division Application Team**

This application note presents the different ST9+ resources for configuring and initializing its external memory interface.

The ST9+ has a single 4 Mbyte memory space segmented in 64 segments of 64 Kbytes, plus an independent register file space. The memory space contains internal memories (internal ROM and RAM with predefined addresses) and you can map your external memories (at the addresses in any segments not used for internal memories). Please refer to the MMU chapter of the ST9+ datasheet for more information on the way this memory space is addressed.

The ST9+ external memory access cycle is composed of 2 clock phases (cf. Figure 1):

- Phase T1: the memory address is output through the ST9+ EMI (External Memory Interface).
- Phase T2: if the memory access is a Read cycle, the data signals are sampled by the ST9+.

If the memory access is a Write cycle, the ST9+ outputs data to be written in external memory.

### **AN1087: ST9+ INTERRUPT RESPONSE TIME**

#### **Microcontroller Division Application Team**

This application note presents the ST9+ interrupt response time calculation for each kind of interrupt in the best and the worst cases. The interrupt response time is the time between the interrupt event occurrence and the start of the corresponding interrupt service routine. The different phases of interrupt processing are described in detail.



## 4 ST10 FAMILY

### AN490: PROGRAMMING FLASH MEMORY OF THE ST10F166

The ST10F166 high end microcontroller with on-chip Flash Memory fulfills the requirements of applications requiring an update to a part or all the program code. The block erase capability is also of use during the application development stage or for program updating. For data acquisition, the ST10F166 allows the programming of 16- or 32-bit data independently.

Operations on the Flash Memory are under software control. Erasure or programming is a simple procedure, however precautions must be taken to prevent damage to the ST10F166.

This Application Note describes the basic characteristics of the Flash Memory cell, and the different algorithms used for erasure and programming.

### AN1086: ST7 / ST10U435 CAN-do SOLUTIONS FOR CAR MULTIPLEXING

**L. Perier / A. Coen**

Replacing a classical harness with a multiplexing (mux) network makes cars more competitive as it increases their flexibility and simplifies the wiring. CAN is the leading protocol for car mux systems thanks to its large speed spectrum and noise immunity. But each application has specific constraints in terms of protocol, cost and performance. So a single node architecture does not fit all the needs.

This article compares first of all, the major car mux protocols: CAN, J1850 and SCI/UART. The second part describes optimized nodes including microcontrollers (ST725x, ST92F120, ST10F167) with embedded FLASH/ROM and physical line interfaces (U435). Then it presents roadmaps for MCU cores, embedded FLASH memories and super-integrations.

### 5 GENERAL

#### AN886: SELECTING BETWEEN ROM AND OTP FOR A MICROCONTROLLER

##### Micro Division

A customer who *develops* an MCU based application needs different levels of flexibility in the ability to perform code modifications (these levels are explained on the next page). To satisfy these requirements, SGS-THOMSON supports several device types: Windowed EPROM, OTP and ROM.

Costs are highly depending on the flexibility given to the device (ability to be easily erased or programmed). ROM is the cheapest technology but provides little flexibility whereas OTP and EPROM are more flexible but their manufacturing cost is higher. The high cost of EPROM MCU devices is due to the price of ceramic packages.

This application note gives some guidelines on how to select between ROM and OTP.

#### AN887: MAKING IT EASY WITH MICROCONTROLLERS

##### Micro Division

A few years ago, system control functions were implemented using logic components and were usually large, heavy boxes. Later on, microprocessors were used and the entire controller could fit onto a small circuit board. As the process of miniaturization continued, all of the components needed for a controller were built right onto one chip. By only including the features specific to the task, cost is relatively low.

This note makes a good description of the general features of a microcontroller (CPU, memory and peripherals) and shows its typical applications. It also tackles some power supplies issues.

#### AN898: EMC GENERAL INFORMATION

##### Micro Division

Because many electronic circuits are in proximity to each other, it is essential that their design is not affected by external noise sources and that the circuit itself is not a noise source affecting other circuits. This relationship is known as electromagnetic compatibility or EMC. Sources of electromagnetic noise are numerous and have both natural and man-made origins. This note describes some EMC general information such as Electromagnetic Interference (EMI) & Susceptibility (EMS) and give some precision about EMC regulations.

For detailed information regarding EMC guidelines for microcontroller - based applications, please refer to AN901.

**AN899: SOLDERING RECOMMENDATIONS and PACKAGING INFORMATION****Micro Division**

SGS-THOMSON supports various package types to adapt MCUs to customer requirements. Beside the available mounting technology (SMD or Throughhole), the choice is often driven by technical and economical concerns. This application note describes the various package types used for MCUs, introduces the various mounting technologies and gives soldering recommendations.

**AN900: INTRODUCTION TO SEMICONDUCTOR TECHNOLOGY****Micro Division**

An integrated circuit is a small but sophisticated device implementing several electronic functions. It is made up of two major parts: a tiny and very fragile silicon chip (die) and a package which is intended to protect the internal silicon chip and to provide users with a practical way of handling the component. This note describes the various “front-end” and “back-end” manufacturing processes and takes the Transistor as an example, because it uses the MOS technology. Actually, this technology is used for the majority of the ICs manufactured at SGS-THOMSON.

**AN901: EMC GUIDE-LINES FOR MICROCONTROLLER - BASED APPLICATIONS****Edouard PRESSON and David JACQUINOD**

EMC must be taken into account at the very beginning of a project; the cost of correcting an EMC problem of an application encountered at the start of the production can be far greater than the cost of detailed EMC study during the development phase.

This note aims to provide guide-lines to the designer of microcontroller-based applications in such a way that the optimum level of EMC performance can be achieved.

For more general information about EMC, please refer to AN898.

**AN902: QUALITY AND RELIABILITY INFORMATION****Micro Division**

We think that maintaining an optimal quality level is very important but we also believe that our customers contribute to the quality chain when they handle or program our MCU devices. Quality is involved at each step but it is important to notice that the customer also has a major role in quality assurance. This application note describes all the stages an SGS-THOMSON's product needs to get over to be qualified, passing the various reliability tests.

### **AN912:A SIMPLE GUIDE TO DEVELOPMENT TOOLS**

#### **K. Bigué**

MCU Development Tools can be used to program and evaluate one or several microcontrollers. This application note describes the types of tools that exist and the tasks for which they are used. With the aid of diagrams and illustrations, this application note provides easily-understandable answers to questions like “What are Development Tools?”, “What are the characteristics of High Level and Low level languages?” or “What is a Debugger?”. A general description of hardware tools allows the purpose of various tool packages such as emulators and starter kits to be compared.

### **AN1181:ELECTROSTATIC DISCHARGE SENSITIVITY MEASUREMENT**

#### **Micro Division**

This application note describes a procedure for determining the susceptibility of microcontroller devices to ESD damage.

---

## Index

---

AN 392 .....	7	AN 841 .....	18	AN1016 .....	28
AN 413 .....	45	AN 842 .....	18	AN1017 .....	34
AN 414 .....	7	AN 859 .....	11	AN1039 .....	30
AN 415 .....	45	AN 863 .....	9	AN1041 .....	34
AN 416 .....	8	AN 885 .....	16	AN1042 .....	34
AN 417 .....	10	AN 886 .....	49	AN1044 .....	34
AN 419 .....	13	AN 887 .....	49	AN1045 .....	34
AN 420 .....	22	AN 898 .....	49	AN1046 .....	35
AN 421 .....	46	AN 899 .....	50	AN1047 .....	35
AN 422 .....	8	AN 900 .....	50	AN1048 .....	35
AN 431 .....	19	AN 901 .....	50	AN1050 .....	28
AN 432 .....	22	AN 902 .....	50	AN1064 .....	30
AN 433 .....	11	AN 910 .....	42, 46	AN1069 .....	46
AN 434 .....	23	AN 911 .....	25	AN1070 .....	41
AN 435 .....	23	AN 912 .....	51	AN1075 .....	47
AN 490 .....	48	AN 913 .....	27	AN1076 .....	47
AN 590 .....	26	AN 914 .....	27	AN1078 .....	36
AN 591 .....	26	AN 969 .....	32	AN1082 .....	37
AN 592 .....	26	AN 970 .....	32	AN1083 .....	37
AN 593 .....	27	AN 971 .....	32	AN1086 .....	43, 48
AN 594 .....	20	AN 972 .....	32	AN1087 .....	47
AN 595 .....	13	AN 973 .....	33	AN1106 .....	31
AN 597 .....	14	AN 974 .....	33	AN1127 .....	28
AN 598 .....	14	AN 975 .....	25	AN1129 .....	38
AN 669 .....	24	AN 976 .....	33	AN1130 .....	39
AN 670 .....	24	AN 979 .....	33	AN1148 .....	36
AN 671 .....	24	AN 980 .....	33	AN1149 .....	36
AN 672 .....	20	AN 982 .....	41	AN1150 .....	43
AN 673 .....	21	AN 985 .....	29	AN1151 .....	44
AN 674 .....	16	AN 986 .....	29	AN1179 .....	41
AN 675 .....	15	AN 987 .....	29	AN1180 .....	40
AN 676 .....	17	AN 988 .....	29	AN1181 .....	51
AN 677 .....	17	AN 989 .....	30	AN1182 .....	40
AN 678 .....	27	AN 990 .....	42		
AN 839 .....	17	AN1014 .....	41		
AN 840 .....	18	AN1015 .....	25		

## GENERAL

---

### Notes:

Information furnished is believed to be accurate and reliable. However, STMicroelectronics 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 STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

©2000 STMicroelectronics - All Rights Reserved.

Purchase of I<sup>2</sup>C Components by STMicroelectronics conveys a license under the Philips I<sup>2</sup>C Patent. Rights to use these components in an I<sup>2</sup>C system is granted provided that the system conforms to the I<sup>2</sup>C Standard Specification as defined by Philips.

STMicroelectronics Group of Companies

Australia - Brazil - China - Finland - France - Germany - Hong Kong - India - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain  
Sweden - Switzerland - United Kingdom - U.S.A.

<http://www.st.com>