

Abstract:

Security in travel is primary concern for every one. This Project describes a design of effective alarm system that can monitor an automotive / vehicle / car condition in traveling. This project is designed to inform about an accident that is occurred to a vehicle to the family members of the traveling persons. This project uses a piezo-electric sensor which can detect the abrupt vibration when an accident is occurred. This sends a signal to microcontroller.

This Project presents an automatic vehicle accident detection system using GPS and GSM modems. The system can be interconnected with the car alarm system and alert the owner on his mobile phone. This detection and messaging system is composed of a GPS receiver, Microcontroller and a GSM Modem. GPS Receiver gets the location information from satellites in the form of latitude and longitude.

The LPC2148 controller processes this information and this processed information is sent to the user/owner using GSM modem A GSM modem is interfaced to the MCU. The GSM modem sends an SMS to the predefined mobile number and informs about this accident. This enable it to monitor the accident situations and it can immediately alerts the police/ambulance service with the location of accident.

The project is built around the ARM7 LPC2148 controller. This micro controller provides all the functionality of the SMS alert system. It also takes care of filtering of the signals at the inputs. The uniqueness of this project is, not only alerting the neighbors by its siren, but also it sends a caution SMS to four mobile numbers. These numbers can be changed at any time by the user using a 3X4 key pad. These numbers are stored in EEPROM. When the accident occurs, the vibration sensor is activated then automatically vehicle door opens.

ARM7TDMI is an advanced version of microprocessors and forms the heart of the system. This autonomous robot is used to sense the obstacle and navigate the robot in forward, backward, left, right directions

The LPC2148 are based on a 16/32 bit ARM7TDMI-S™ CPU with real-time emulation and embedded trace support, together with 128/512 kilobytes of embedded high speed flash memory. A 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at maximum clock rate. For critical code size applications, the alternative 16-bit Thumb Mode reduces code by more than 30% with minimal performance penalty. With their compact 64 pin package, low power consumption, various 32-bit timers, 4- channel 10-bit ADC, USB PORT,PWM channels and 46 GPIO lines with up to 9 external interrupt pins these microcontrollers are particularly suitable for industrial control, medical systems, access control and point-of-sale. With a wide range of serial communications interfaces, they are also very well suited for communication gateways, protocol converters and embedded soft modems as well as many other general-purpose applications.

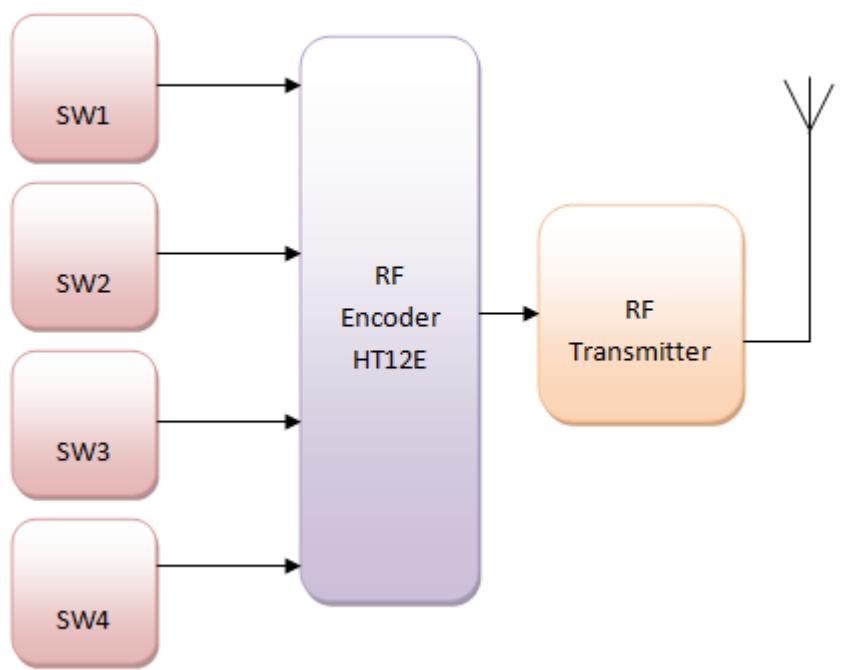
This project uses 12v (Lead Acid Battery). This project uses two power supplies, one is regulated 5V for modules and other one is 3.3V for LPC2148. 7805 three terminal voltage regulator is used for voltage regulation. Bridge type full wave rectifier is used to rectify the ac out put of secondary of 230/12V step down transformer.

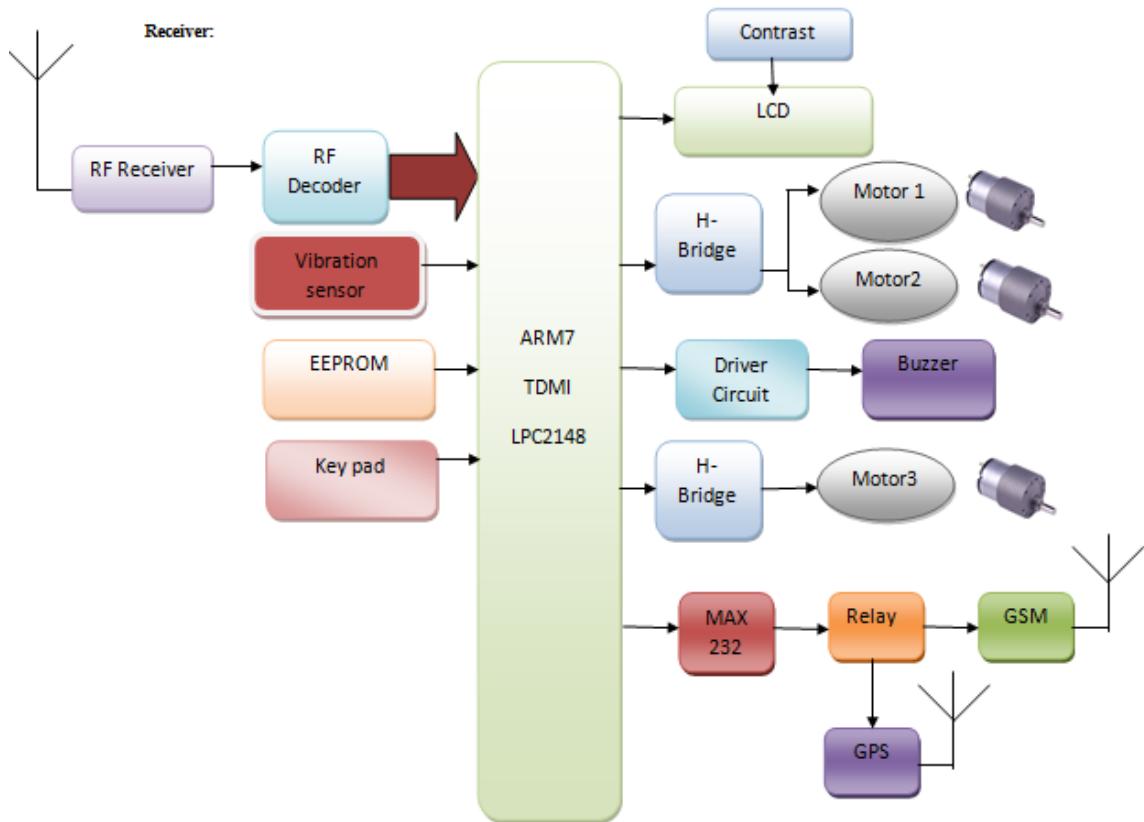
Advantages:

- Sophisticated security
- Monitors all hazards and threats
- Alert message to mobile phone for remote information
- Mobile number can be changed at any time

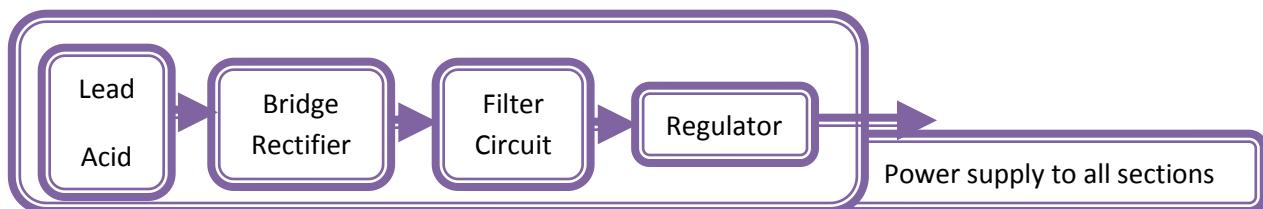
Applications:

- Auto motives and transport vehicles
- Security, Remote monitoring, Transportation and logistics
- This system is also can be interfaced with Vehicle airbag system.





Power Supply:



CHAPTER 1

INTRODUCTION

Introduction

An embedded system is a combination of software and hardware to perform a dedicated task. Some of the main devices used in embedded products are Microprocessors and Microcontrollers.

Microprocessors are commonly referred to as general purpose processors as they simply accept the inputs, process it and give the output.

In contrast, a microcontroller not only accepts the data as inputs but also manipulates it, interfaces the data with various devices, controls the data and thus finally gives the result.

All these tasks are possible with the microcontroller because the microcontroller has a CPU in addition to a fixed amount of RAM, ROM, I/O ports and timer all on a single chip. This fixed amount of RAM, ROM and number of I/O ports in microcontroller makes them ideal for many applications where cost and space are critical.

The system is designed to control devices at remote place. Relays or Triacs are connected to control the different equipments.

INTRODUCTION TO EMBEDDED SYSTEMS

An embedded system can be defined as a computing device that does a specific focused job. Appliances such as the air-conditioner, VCD player, DVD player, printer, fax machine, mobile phone etc. are examples of embedded systems. Each of these appliances will have a processor and special hardware to meet the specific requirement of the application along with the embedded software that is executed by the processor for meeting that specific requirement. The embedded software is also called “firm ware”. The desktop/laptop computer is a general purpose computer. You can use it for a variety of applications such as playing games, *word* processing, accounting, software development and so on. In contrast, the software in the embedded systems is always fixed listed below:

- Embedded systems do a very specific task, they cannot be programmed to do different things. . Embedded systems have very limited resources, particularly the memory. Generally, they do not have secondary storage devices such as the CDROM or the floppy disk. Embedded systems have to work against some deadlines. A specific job has to be completed within a specific time. In some embedded systems, called real-time systems, the deadlines are stringent. Missing a deadline may cause a catastrophe-loss of life or damage to property. Embedded systems are constrained for power. As many embedded systems operate through a battery, the power consumption has to be very low.
- Some embedded systems have to operate in extreme environmental conditions such as very high temperatures and humidity.

Application Areas

Nearly 99 per cent of the processors manufactured end up in embedded systems. The embedded system market is one of the highest growth areas as these systems are used in very market segment- consumer electronics, office automation, industrial automation, biomedical engineering, wireless communication,

data communication, telecommunications, transportation, military and so on.

Consumer appliances: At home we use a number of embedded systems which include digital camera, digital diary, DVD player, electronic toys, microwave oven, remote controls for TV and air-conditioner, VCO player, video game consoles, video recorders etc. Today's high-tech car has about 20 embedded systems for transmission control, engine spark control, air-conditioning, navigation etc. Even wristwatches are now

becoming embedded systems. The palmtops are powerful embedded systems using which we can carry out many general-purpose tasks such as playing games and word processing.

Office automation: The office automation products using em embedded systems are copying machine, fax machine, key telephone, modem, printer, scanner etc.

Industrial automation: Today a lot of industries use embedded systems for process control. These include pharmaceutical, cement, sugar, oil exploration, nuclear energy, electricity generation and transmission. The embedded systems for industrial use are designed to carry out specific tasks such as monitoring the temperature, pressure, humidity, voltage, current etc., and then take appropriate action based on the monitored levels to control other devices or to send information to a centralized monitoring station. In hazardous industrial environment, where human presence has to be avoided, robots are used, which are programmed to do specific jobs. The robots are now becoming very powerful and carry out many interesting and complicated tasks such as hardware assembly.

Medical electronics: Almost every medical equipment in the hospital is an embedded system. These equipment's include diagnostic aids such as ECG, EEG, blood pressure measuring devices, X-ray scanners; equipment used in blood analysis, radiation, colonoscopy, endoscopy etc. Developments in medical electronics have paved way for more accurate diagnosis of diseases.

Computer networking: Computer networking products such as bridges, routers, Integrated Services Digital Networks (ISDN), Asynchronous Transfer Mode (ATM), X.25 and frame relay switches are embedded systems which implement the necessary data communication protocols. For example, a router interconnects two networks. The two networks may be running different protocol stacks. The router's function is to obtain the data packets from incoming pores, analyze the packets and send them towards the destination after doing necessary protocol conversion. Most networking equipment's, other than the end systems (desktop computers) we use to access the networks, are embedded systems

. **Telecommunications:** In the field of telecommunications, the embedded systems can be categorized as subscriber terminals and network equipment. The subscriber terminals such as key telephones, ISDN phones, terminal adapters, web cameras are embedded systems. The network equipment includes multiplexers, multiple access systems, Packet Assemblers Dissemblers (PADs), satellite modems etc. IP phone, IP gateway, IP gatekeeper etc. are the latest embedded systems that provide very low-cost voice communication over the Internet.

Wireless technologies: Advances in mobile communications are paving way for many interesting applications using embedded systems. The mobile phone is one of the marvels of the last decade of the 20th century. It is a very powerful embedded system that provides voice communication while we are on the move. The Personal Digital Assistants and the palmtops can now be used to access multimedia services over the Internet. Mobile communication infrastructure such as base station controllers, mobile switching centers are also powerful embedded systems.

Insemination: Testing and measurement are the fundamental requirements in all scientific and engineering activities. The measuring equipment we use in laboratories to measure parameters such as weight, temperature, pressure, humidity, voltage, current etc. are all embedded systems. Test equipment such as oscilloscope, spectrum analyzer, logic analyzer, protocol analyzer, radio communication test set etc. are embedded systems built around powerful processors

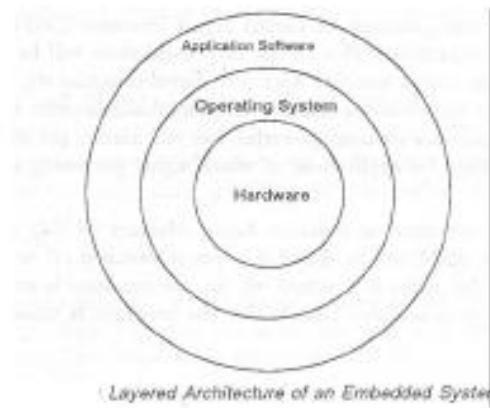
Security: Security of persons and information has always been a major issue. We need to protect our homes and offices; and also the information we transmit and store. Developing embedded systems for security applications is one of the most lucrative businesses nowadays. Security devices at homes, offices, airports etc. for authentication and verification are embedded systems. Encryption devices are nearly 99 per cent of

the processors that are manufactured end up in~ embedded systems. Embedded systems find applications in . every industrial segment- consumer electronics, transportation, avionics, biomedical engineering, manufacturing, process control and industrial automation, data communication, telecommunication, defense, security etc. Used to encrypt the data/voice being transmitted on communication links such as telephone lines. Biometric systems using fingerprint and face recognition are now being extensively used for user authentication in banking applications as well as for access control in high security buildings.

Finance: Financial dealing through cash and cheques are now slowly paving way for transactions using smart cards and ATM (Automatic Teller Machine, also expanded as Any Time Money) machines. Smart card, of the size of a credit card, has a small micro-controller and memory; and it interacts with the smart card reader! ATM machine and acts as an electronic wallet. Smart card technology has the capability of ushering in a cashless society. Well, the list goes on. It is no exaggeration to say that eyes wherever you go, you can see, or at least feel, the work of an embedded system!

Overview of Embedded System Architecture

Every embedded system consists of custom-built hardware built around a Central Processing Unit (CPU). This hardware also contains memory chips onto which the software is loaded. The software residing on the memory chip is also called the ‘firmware’. The embedded system architecture can be represented as a layered architecture as shown in Fig.

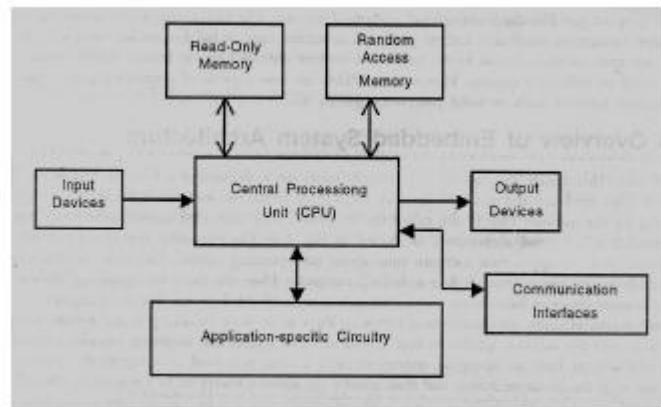


The operating system runs above the hardware, and the application software runs above the operating system. The same architecture is applicable to any computer including a desktop computer. However, there are significant differences. It is not compulsory to have an operating system in every embedded system. For small appliances such as remote control units, air conditioners, toys etc., there is no need *for* an operating system and you can write only the software specific to that application. For applications involving complex processing, it is advisable to have an operating system. In such a case, you need to integrate the application software with the operating system and then transfer the entire software on to the memory chip. Once the software is transferred to the memory chip, the software will continue to run *for* a long time you don’t need to reload new software.

Now, let us see the details of the various building blocks of the hardware of an embedded system. As shown in Fig. the building blocks are;

- Central Processing Unit (CPU)

- Memory (Read-only Memory and Random Access Memory)
- Input Devices
- Output devices
- Communication interfaces
- Application-specific circuitry



Central Processing Unit (CPU):

The Central Processing Unit (processor, in short) can be any of the following: microcontroller, microprocessor or Digital Signal Processor (DSP). A micro-controller is a low-cost processor. Its main attraction is that on the chip itself, there will be many other components such as memory, serial communication interface, analog-to digital converter etc. So, for small applications, a micro-controller is the best choice as the number of external components required will be very less. On the other hand, microprocessors are more powerful, but you need to use many external components with them. DSP is used mainly for applications in which signal processing is involved such as audio and video processing.

Memory:

The memory is categorized as Random Access Memory (RAM) and Read Only Memory (ROM). The contents of the RAM will be erased if power is switched off to the chip, whereas ROM retains the contents even if the power is switched off. So, the firmware is stored in the ROM. When power is switched on, the processor reads the ROM; the program is executed.

Input devices:

Unlike the desktops, the input devices to an embedded system have very limited capability. There will be no keyboard or a mouse, and hence interacting with the embedded system is no easy task. Many embedded systems will have a small keypad—you press one key to give a specific command. A keypad may be used to input only the digits. Many embedded systems used in process control do not have any input device *for* user interaction; they take inputs *from* sensors or transducers and produce electrical signals that are in turn fed to other systems.

Output devices:

The output devices of the embedded systems also have very limited capability. Some embedded systems will have a *few* Light Emitting Diodes (LEDs) *to* indicate the health status of the system modules, or *for* visual indication of alarms. A small Liquid Crystal Display (LCD) may also be used to display *some* important parameters.

Communication interfaces:

The embedded systems may need to, interact with other embedded systems as they may have to transmit data to a desktop. To facilitate this, the embedded systems are provided with one or a *few* communication interfaces such as RS232, RS422, RS485, Universal Serial Bus (USB), IEEE 1394, Ethernet etc.

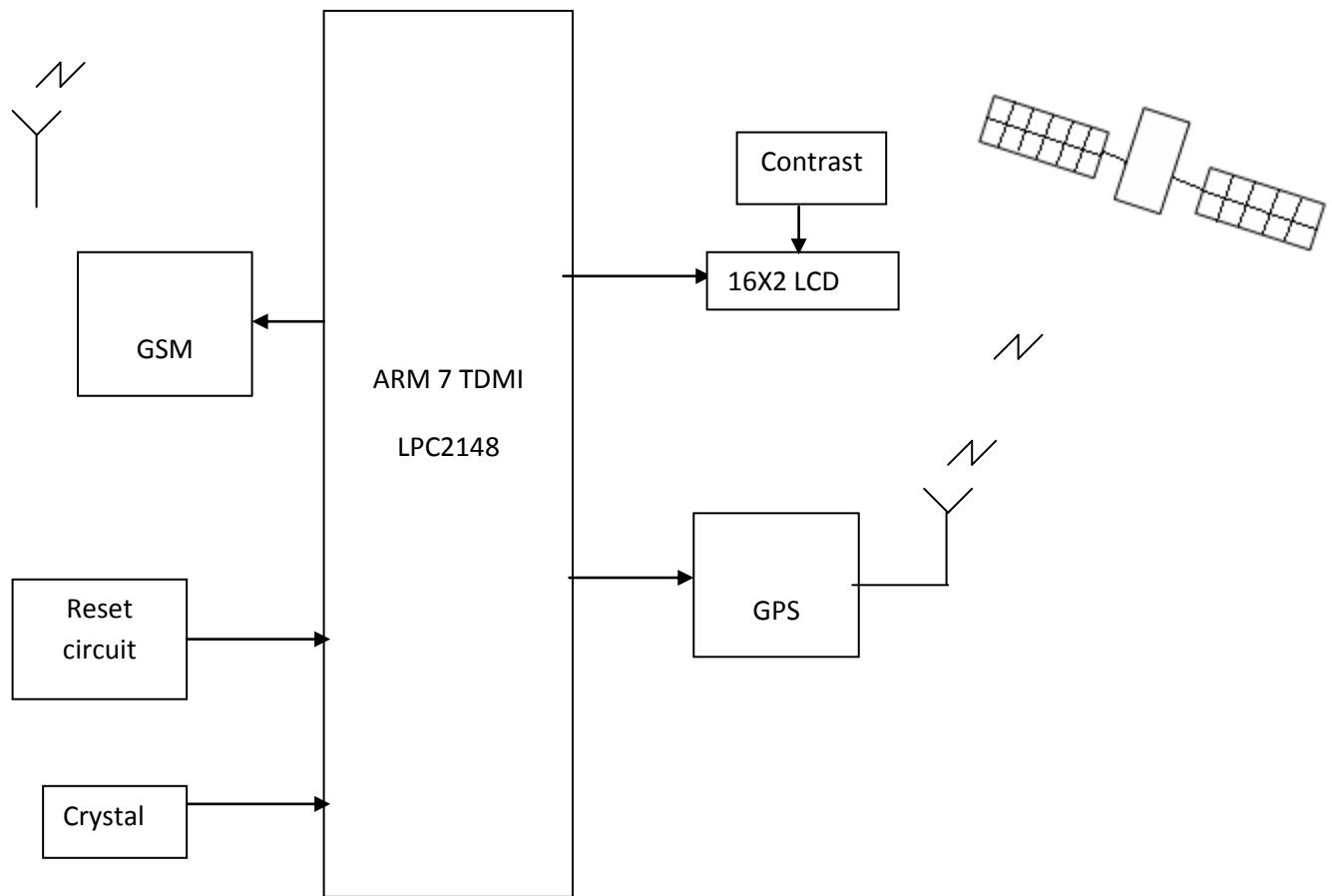
Application-specific circuitry:

Sensors, transducers, special processing and control circuitry may be required for an embedded system, depending on its application.

CHAPTER 2

BLOCK DIAGRAM

BLOCK DIAGRAM:



CHAPTER 3

LITERATURE SURVEY

3.1 Power supply

Power supply is a reference to a source of electrical power. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a **power supply unit** or PSU. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

The term "**power supply**" is sometimes restricted to those devices that *convert* some other form of energy into electricity (such as solar power and fuel cells and generators). A more accurate term for devices that convert one form of electric power into another form (such as transformers and linear regulators) is power converter. The most common conversion is from AC to DC.

AQ regulated power supply or stabilized power supply is one that includes circuitry to tightly control the output voltage and/or current to a specific value. The specific value is closely maintained despite variations in the load presented to the power supply's output, or any reasonable voltage variation at the power supply's input.

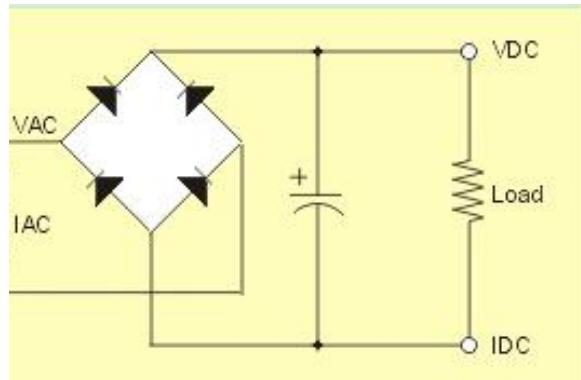


Fig 3.1 Power Supply Block diagram

The micro controller and other devices get power supply from AC to DC adapter through voltage regulator. The adapter output voltage will be 12V DC non-regulated. The 7805/7812 voltage regulators are used to convert 12V to 5VDC.

3.2 Rectifier:

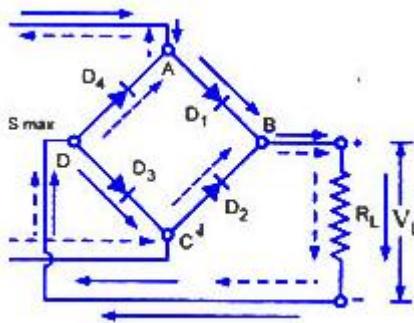
The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C. The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification.



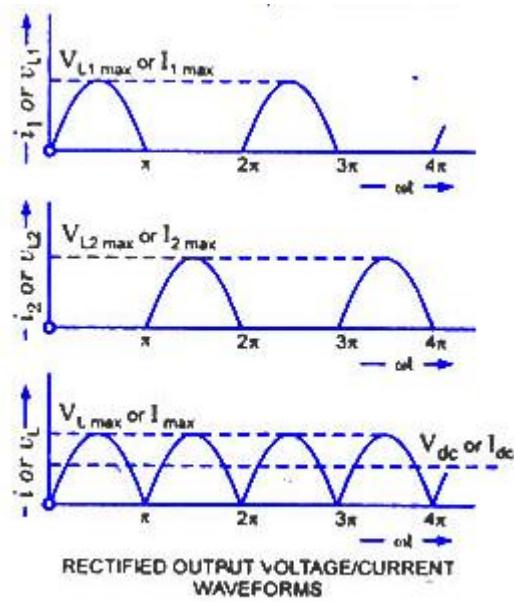
The Bridge rectifier is a circuit, which converts an ac voltage to dc voltage using both half cycles of the input ac voltage. The Bridge rectifier circuit is shown in the figure. The circuit has four diodes connected to form a bridge. The ac input voltage is applied to the diagonally opposite ends of the bridge. The load resistance is connected between the other two ends of the bridge.

For the positive half cycle of the input ac voltage, diodes D1 and D3 conduct, whereas diodes D2 and D4 remain in the OFF state. The conducting diodes will be in series with the load resistance R_L and hence the load current flows through R_L . For the negative half cycle of the input ac voltage, diodes D2 and D4 conduct whereas, D1 and D3 remain OFF. The conducting diodes D2 and D4 will be in series with the load resistance R_L and hence the current flows through R_L in the same direction as in the previous half cycle. Thus a bi-directional wave is converted into a unidirectional wave.

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothes the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.



Output wave forms:



Bridge Rectifier

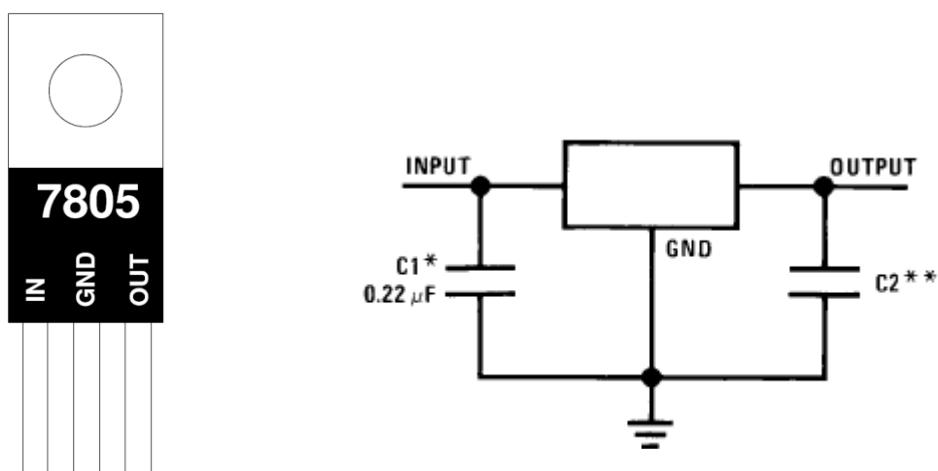
3.3 Filter:

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.

3.4 Regulator:

As the name itself implies, it regulates the input applied to it. A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. In this project, power supply of 5V and 12V are required. In order to obtain these voltage levels, 7805 and 7812 voltage regulators are to be used. The first number 78 represents positive supply and the numbers 05, 12 represent the required output voltage levels.

The L78xx series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-3, D2PAK and DPAK packages and several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1 A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltage and currents.



CHAPTER 4

ARM PROCESSOR

CHAPTER-4

ARM PROCESSOR

4.1Introduction to ARM:

Founded in November 1990, it is spun out of Acorn Computers, it Designs the ARM range of RISC processor cores. Licenses ARM core designs to semiconductor partners who fabricate and sell to their customers. ARM does not fabricate silicon itself, it also develop technologies to assist with the design-in of the ARM architecture. Software tools, boards, debug hardware, application software, bus architectures, peripherals etc.

The ARM processor core originates within a British computer company called Acorn. In the mid-1980s they were looking for replacement for the 6502 processor used in their BBC computer range, which were widely used in UK schools. None of the 16-bit architectures becoming available at that time met their requirements, so they designed their own 32-bit processor.

Other companies became interested in this processor, including Apple who was looking for a processor for their PDA project (which became the Newton). After much discussion this led to Acorn's processor design team splitting off from Acorn at the end of 1990 to become Advanced RISC Machines Ltd, now just ARM Ltd. Thus ARM Ltd now designs the ARM family of RISC processor cores, together with a range of other supporting technologies.

One important point about ARM is that it does not fabricate silicon itself, but instead just produces the design - we are an Intellectual Property (or IP) company. Instead silicon is produced by companies who license the ARM processor design.

Architectural overview:

The ARM7TDMI-S is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are much simpler than those of microprogrammed Complex Instruction Set Computers (CISC). This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective processor core. Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory. The ARM7TDMI-S processor also employs a unique architectural strategy known as Thumb, which makes it ideally suited to high-volume applications with memory restrictions, or applications where code density is an issue. The key idea behind Thumb is that of a super-reduced instruction set. Essentially, the ARM7TDMI-S processor has two instruction sets:

- The standard 32-bit ARM set.
- A 16-bit Thumb set.

The Thumb set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because Thumb code operates on the same 32-bit register set as ARM code. Thumb code is able to provide up to 65 % of the code size of ARM, and 160 % of the performance of an equivalent ARM processor connected to a 16-bit memory system. The particular flash implementation in the LPC2141/42/44/46/48 allows for full speed execution also in ARM mode. It is recommended to program performance critical and short code sections (such as interrupt service routines and DSP algorithms) in ARM mode.

ARM Powered Products in Industry



4.2 Data Sizes and Instruction Sets

- The ARM is a 32-bit architecture.
- When used in relation to the ARM:
 - Byte means 8 bits
 - Halfword means 16 bits (two bytes)
 - Word means 32 bits (four bytes)
 - Most ARM's implement two instruction sets
 - 32-bit ARM Instruction Set

- 16-bit Thumb Instruction Set
- Jazelle cores can also execute Java bytecode

The cause of confusion here is the term “word” which will mean 16-bits to people with a 16-bit background. In the ARM world 16-bits is a “half word” as the architecture is a 32-bit one, whereas “word” means 32-bits. Java byte codes are 8-bit instructions designed to be architecture independent. Jazelle transparently executes most byte codes in hardware and some in highly optimized ARM code. This is due to a trade off between hardware complexity (power consumption & silicon area) and speed.

4.3 Processor operating Modes

The ARM has seven basic operating modes:

- **User** : unprivileged mode under which most tasks run
- **FIQ** : entered when a high priority (fast) interrupt is raised
- **IRQ** : entered when a low priority (normal) interrupt is raised
- **Supervisor** : entered on reset and when a Software Interrupt instruction is executed
- **Abort** : used to handle memory access violations
- **Undef** : used to handle undefined instructions
- **System** : privileged mode using the same registers as user mode

The Programmers Model can be split into two elements - first of all, the processor modes and secondly, the processor registers. So let's start by looking at the modes. Now the typical application will run in an unprivileged mode known as “User” mode, whereas the various exception types will be dealt with in one of the privileged modes : Fast Interrupt, Supervisor, Abort, Normal Interrupt and Undefined (and we will look at what causes each of the exceptions later on).

NB - spell out the word FIQ, otherwise you are saying something rude in German!

One question here is what is the difference between the privileged and unprivileged modes? Well in reality very little really - the ARM core has an output signal (nTRANS on

ARM7TDMI, InTRANS, DnTRANS on 9, or encoded as part of HPROT or BPROT in AMBA) which indicates whether the current mode is privileged or unprivileged, and this can be used, for instance, by a memory controller to only allow IO access in a privileged mode. In addition some operations are only permitted in a privileged mode, such as directly changing the mode and enabling of interrupts.

All current ARM cores implement system mode (added in architecture v4). This is simply a privileged version of user mode. Important for re-entrant exceptions because no exceptions can cause system mode to be entered.

4.4 The Registers

ARM has 37 registers all of which are 32-bits long.

- 1 dedicated program counter
- 1 dedicated current program status register
- 5 dedicated saved program status registers
- 30 general purpose registers
- The current processor mode governs which of several banks is accessible. Each mode can access
 - a particular set of r0-r12 registers
 - a particular r13 (the stack pointer, sp) and r14 (the link register, lr)
 - the program counter, r15 (pc)
 - the current program status register, cpsr
 - Privileged modes (except System) can also access
 - a particular spsr (saved program status register)

The ARM architecture provides a total of 37 registers, all of which are 32-bits long. However these are arranged into several banks, with the accessible bank being governed by the current processor mode. We will see this in more detail in a couple of slides. In summary though, in each mode, the core can access: A particular set of 13 general purpose registers (r0 - r12), A Particular r13 - which is typically used as a stack pointer. This will be a different r13 for each mode, so allowing each exception type to have its own stack, particular R14 - which is used as a

link (or return address) registers. Again this will be a different r14 for each mode. r15 - whose only use is as the Program counter. The CPSR (Current Program Status Register) - this stores additional information about the state of the processor and finally in privileged modes, a particular SPSR (Saved Program Status Register) this stores a copy of the previous CPSR value when an exception occurs. This combined with the link register allows exceptions to return without corrupting processor state.

Program Counter (r15)

- **When the processor is executing in ARM state:**
 - All instructions are 32 bits wide
 - All instructions must be word aligned
 - Therefore the **pc** value is stored in bits [31:2] with bits [1:0] undefined (as instruction cannot be half word or byte aligned).
- **When the processor is executing in Thumb state:**
 - All instructions are 16 bits wide
 - All instructions must be half word aligned
 - Therefore the **pc** value is stored in bits [31:1] with bit [0] undefined (as instruction cannot be byte aligned).
- **When the processor is executing in Jazelle state:**
 - All instructions are 8 bits wide
 - Processor performs a word access to read 4 instructions at once

ARM is designed to efficiently access memory using a single memory access cycle. So word accesses must be on a word address boundary, half word accesses must be on a half word address boundary. This includes instruction fetches.

Point out that strictly, the bottom bits of the PC simply do not exist within the ARM core - hence they are 'undefined'. Memory system must ignore these for instruction fetches.

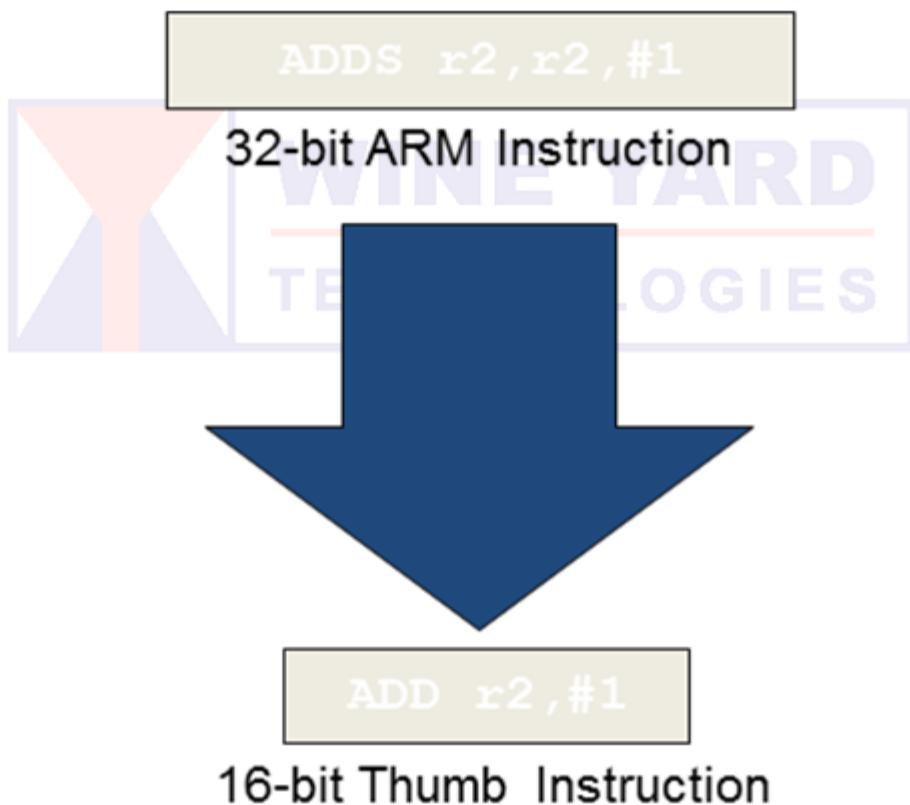
In Jazelle state, the processor doesn't perform 8-bit fetches from memory. Instead it does aligned 32-bit fetches (4-byte perfecting) which is more efficient. Note we don't mention the PC in Jazelle state because the 'Jazelle PC' is actually stored in r14 - this is technical detail that is not relevant as it is completely hidden by the Jazelle support code.

4.5 Thumb

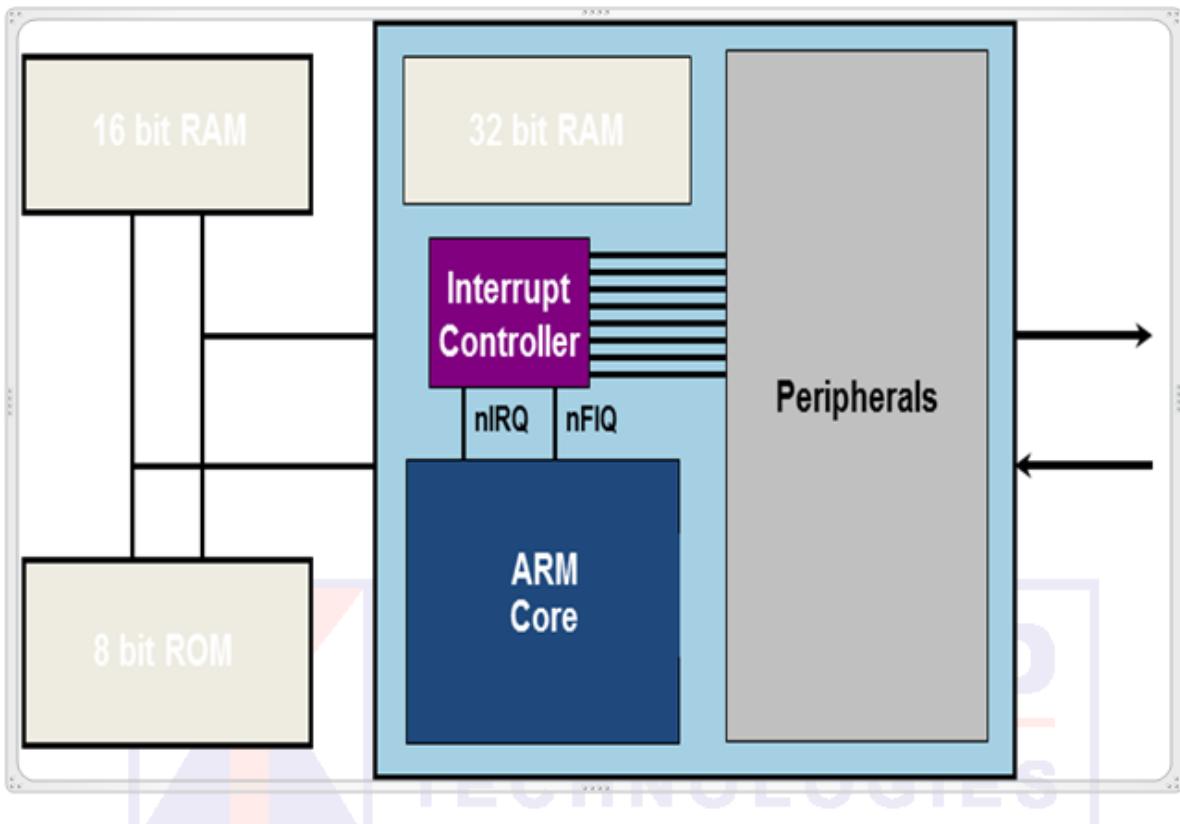
- Thumb is a 16-bit instruction set
 - Optimised for code density from C code (~65% of ARM code size)
 - Improved performance from narrow memory
 - Subset of the functionality of the ARM instruction set
- Core has additional execution state - Thumb
 - Switch between ARM and Thumb using BX instruction

For most instructions generated by compiler:

- Conditional execution is not used
- Source and destination registers identical
- Only Low registers used
- Constants are of limited size
- Inline barrel shifter not used



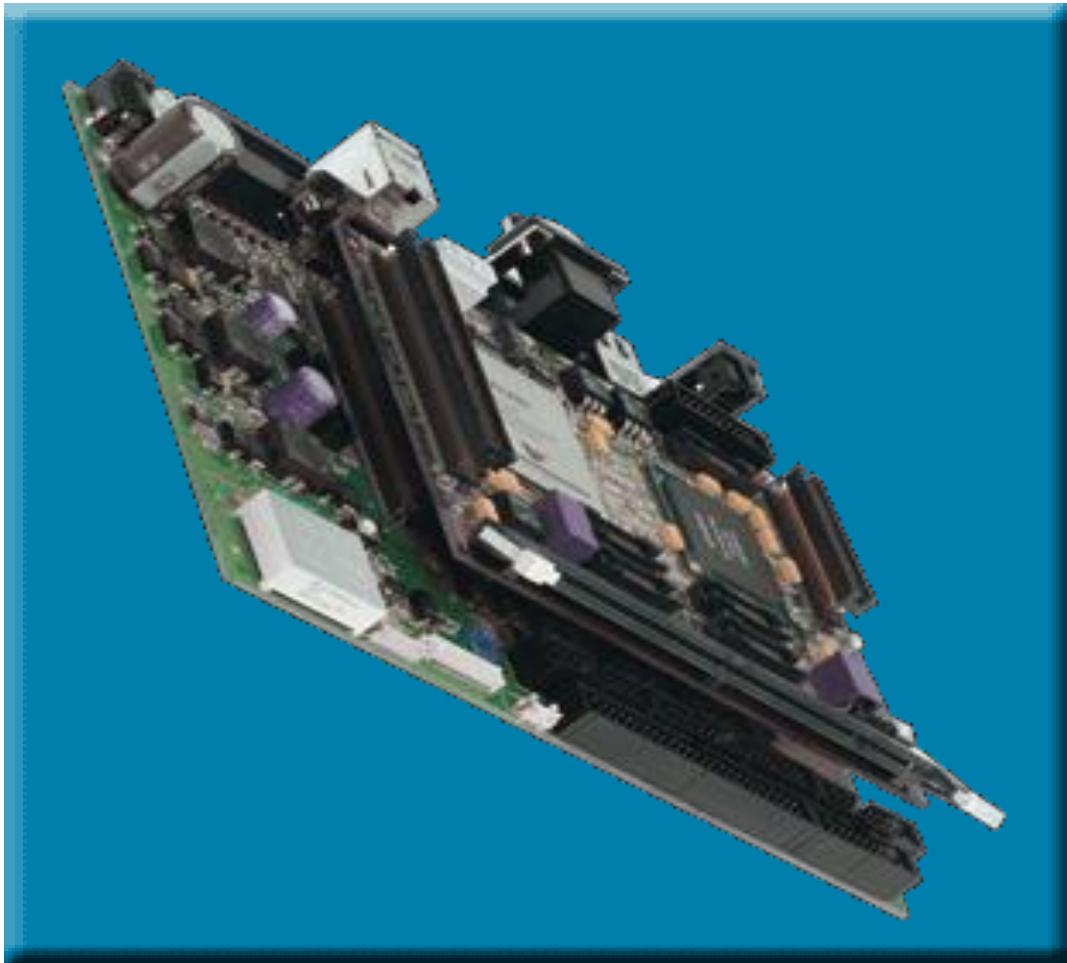
4.6.Example ARM-based System



This slide shows a very generic ARM based design that is actually fairly representative of the designs that we see being done.

On-chip there will be an ARM core (obviously) together with a number of system dependent peripherals. Also required will be some form of interrupt controller which receives interrupts from the peripherals and raises the IRQ or FIQ input to the ARM as appropriate. This interrupt controller may also provide hardware assistance for prioritizing interrupts.

As far as memory is concerned there is likely to be some (cheap) narrow off-chip ROM (or flash) used to boot the system from. There is also likely to be some 16-bit wide RAM used to store most of the runtime data and perhaps some code copied out of the flash. Then on-chip there may well be some 32-bit memory used to store the interrupt handlers and perhaps stacks.

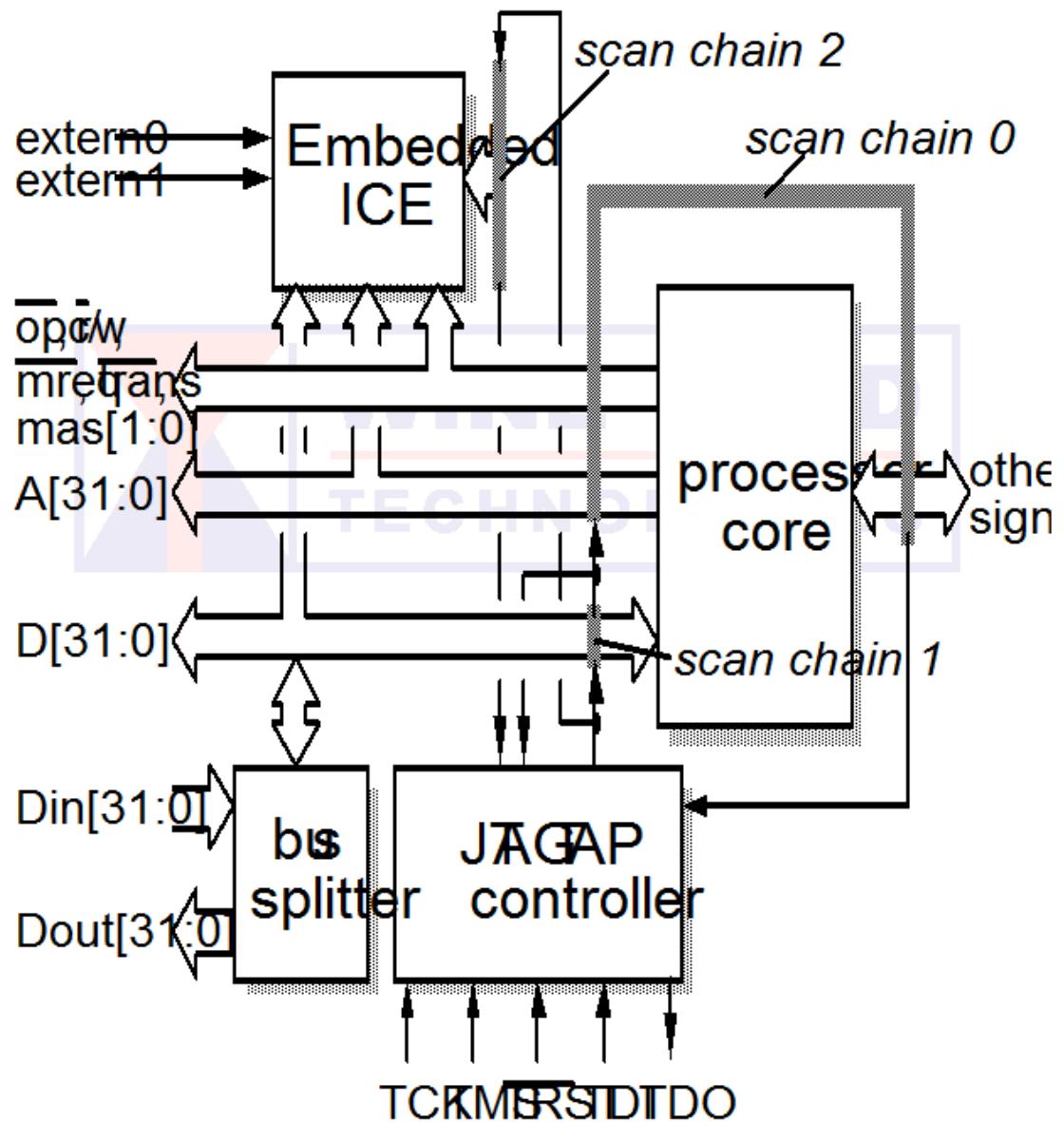


4.7 ARM7TDMI Processor Core

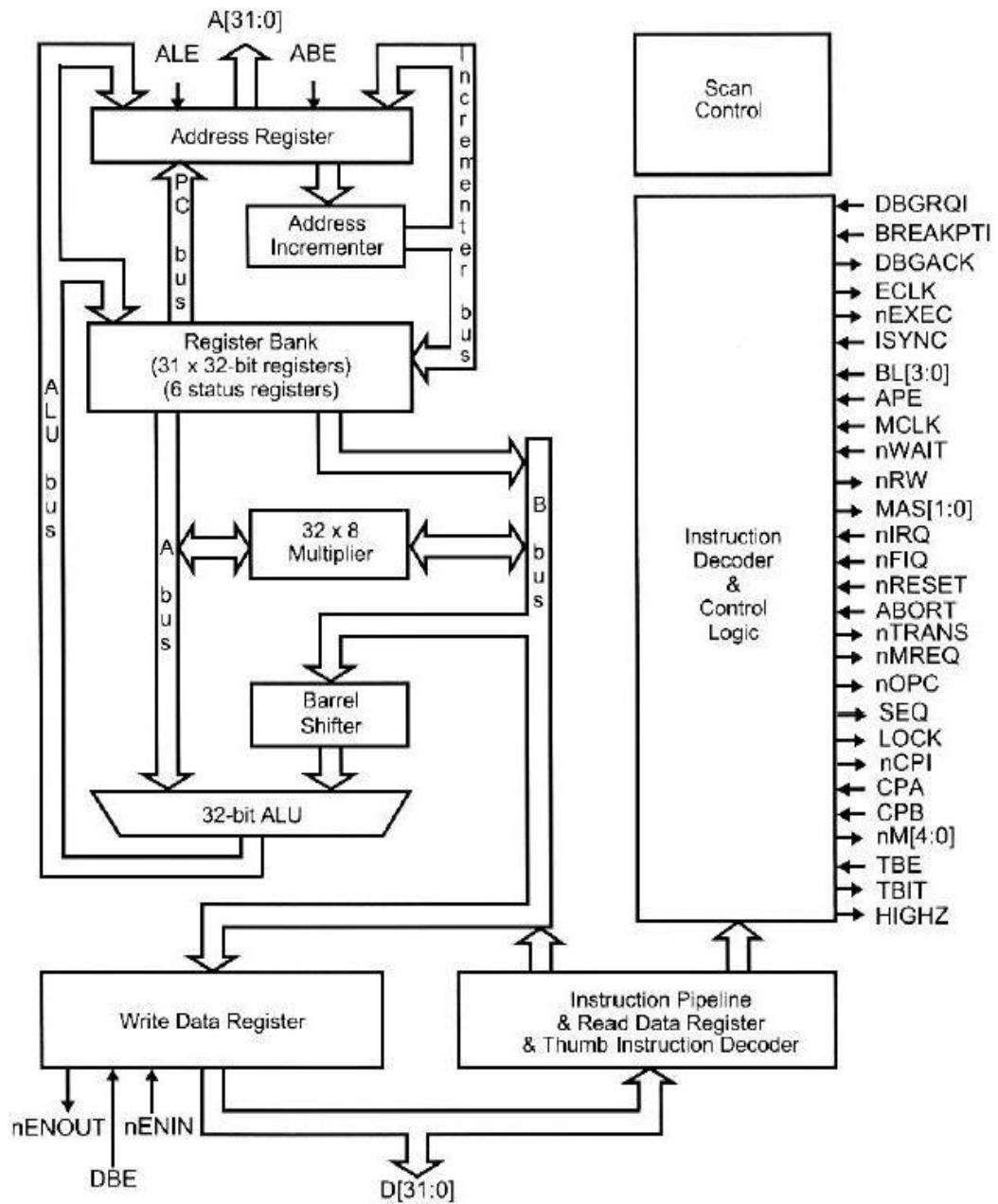
- Current low-end ARM core for applications like digital mobile phones
- TDMI
 - T: Thumb, 16-bit compressed instruction set

- D: on-chip Debug support, enabling the processor to halt in response to a debug request
- M: enhanced Multiplier, yield a full 64-bit result, high performance
- I: Embedded ICE hardware
- Von Neumann architecture

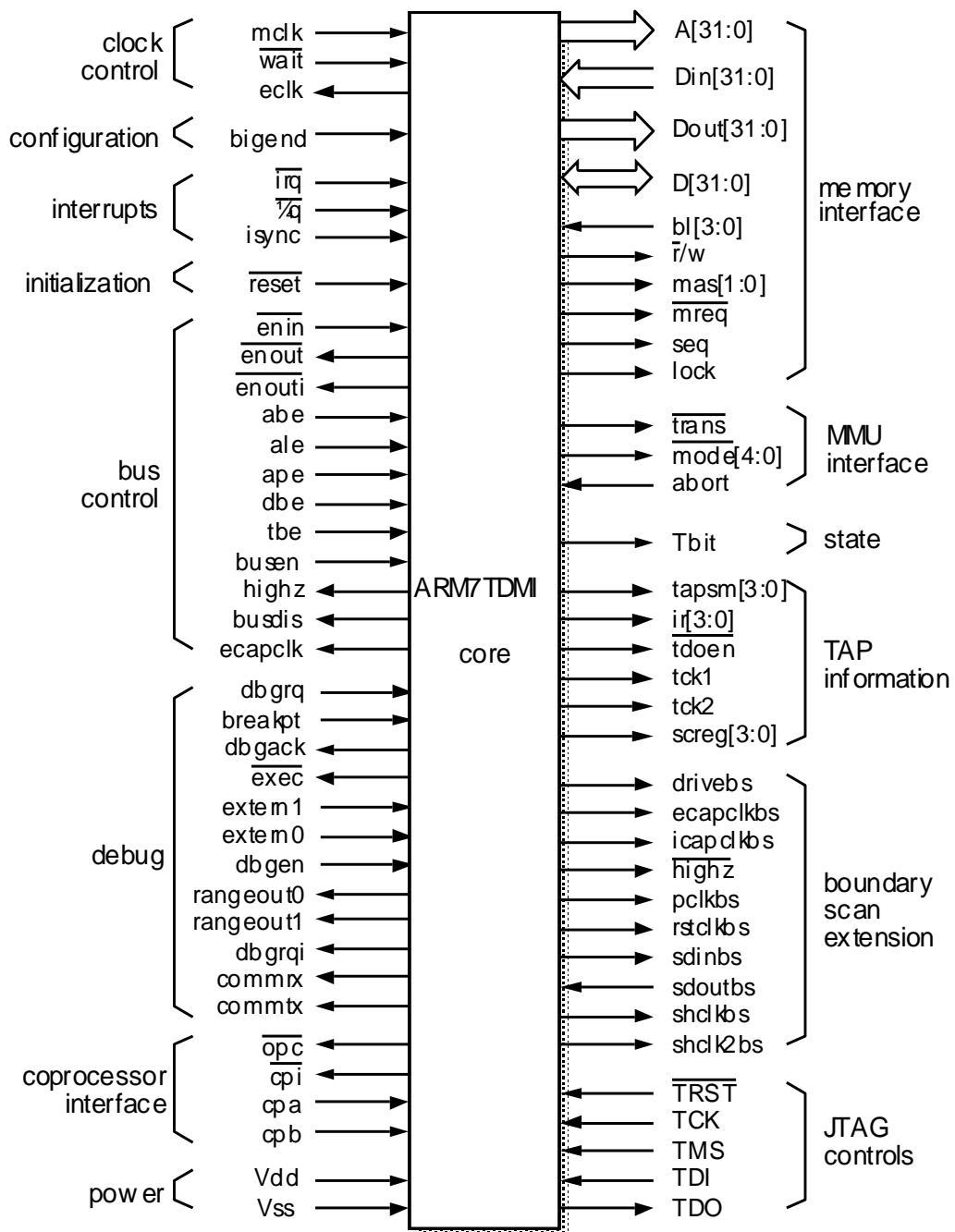
4.8 ARM7TDMI Block Diagram



4.9 ARM7TDMI Core Diagram



4.10 ARM7TDMI Interface Signals



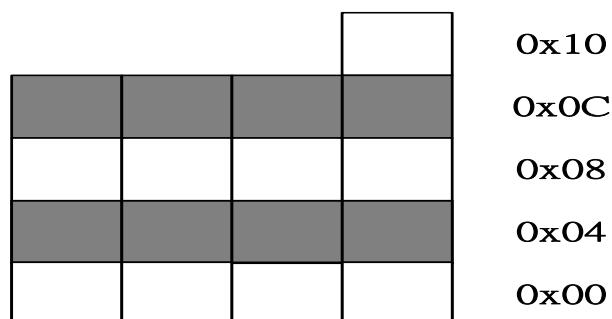
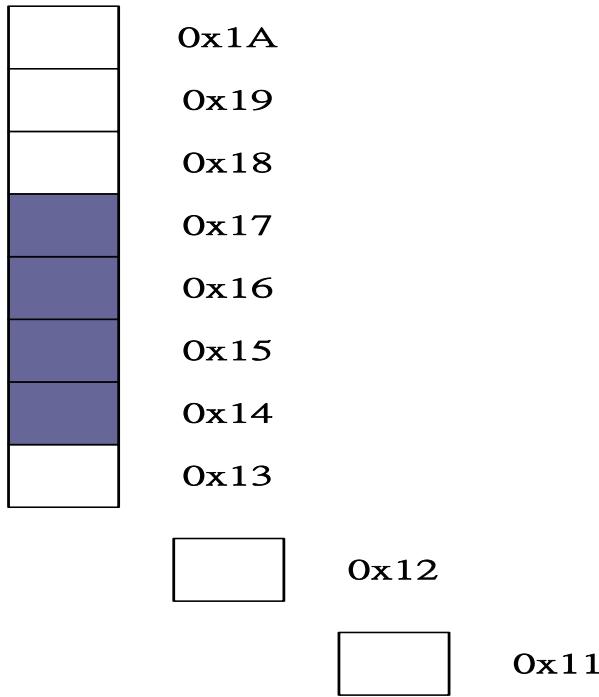
- Clock control
 - All state change within the processor are controlled by *mclk*, the memory clock
 - Internal clock = *mclk* AND *\wait*
 - *eclk* clock output reflects the clock used by the core
- Memory interface
 - 32-bit address *A[31:0]*, bidirectional data bus *D[31:0]*, separate data out *Dout[31:0]*, data in *Din[31:0]*
 - *seq* indicates that the memory address will be sequential to that used in the previous cycle
- Interrupt
 - *\fiq*, fast interrupt request, higher priority
 - *\irq*, normal interrupt request
 - *isync*, allow the interrupt synchronizer to be passed
- Initialization
 - *\reset*, starts the processor from a known state, executing from address 00000000_{16}

4.11 ARM7TDMI characteristics

Process	0.35 um	Transistors	74,209	MIPS	60
Metal layers	3	Core area	2.1 mm ²	Power	87 mW
Vdd	3.3 V	Clock	0 to 66 MHz	MIPS/W	690

Memory Access

- The ARM7 is a Von Neumann, load/store architecture, i.e.,
 - Only 32 bit data bus for both inst. And data.
 - Only the load/store inst. (and SWP) access memory.
- Memory is addressed as a 32 bit address space
- Data type can be 8 bit bytes, 16 bit half-words or 32 bit words, and may be seen as a byte line folded into 4-byte words



Memory as words

- Words must be aligned to 4 byte boundaries, and half-words to 2 byte boundaries.
- Always ensure that memory controller supports all three access sizes

Processor Core Vs CPU Core

- Processor Core
 - The engine that fetches instructions and execute them
 - E.g.: ARM7TDMI, ARM9TDMI, ARM9E-S
- CPU Core
 - Consists of the ARM processor core and some tightly coupled function blocks
 - Cache and memory management blocks
 - E.g.: ARM710T, ARM720T, ARM74T, ARM920T, ARM922T, ARM940T, ARM946E-S, and ARM966E-S

4.12 ARM Processor Family

<u>Processor family</u>	<u># of pipeline stages</u>	<u>Memory organization</u>	<u>Clock Rate</u>	<u>MIPS/MHz</u>
<u>ARM6</u>	<u>3</u>	<u>Von Neumann</u>	<u>25 MHz</u>	
<u>ARM7</u>	<u>3</u>	<u>Von Neumann</u>	<u>66 MHz</u>	<u>0.9</u>
<u>ARM8</u>	<u>5</u>	<u>Von Neumann</u>	<u>72 MHz</u>	<u>1.2</u>
<u>ARM9</u>	<u>5</u>	<u>Harvard</u>	<u>200 MHz</u>	<u>1.1</u>
<u>ARM10</u>	<u>6</u>	<u>Harvard</u>	<u>400 MHz</u>	<u>1.25</u>
<u>StrongARM</u>	<u>5</u>	<u>Harvard</u>	<u>233 MHz</u>	<u>1.15</u>
<u>ARM11</u>	<u>8</u>	<u>Von Neumann/</u>	<u>550 MHz</u>	<u>1.2</u>

CHAPTER 5

LPC2148 CONTROLLER

CHAPTER 5

LPC2148 CONTROLLER

5.1. General description:

The LPC2141/42/44/46/48 microcontrollers are based on a 16-bit/32-bit ARM7TDMI-CPU with real-time emulation and embedded trace support, that combine microcontroller with embedded high speed flash memory ranging from 32 kB to 512 kB. A 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at the maximum clock rate. For critical code size applications, the alternative 16-bit Thumb mode reduces code by more than 30 % with minimal performance penalty. Due to their tiny size and low power consumption, LPC2141/42/44/46/48 are ideal for applications where miniaturization is a key requirement, such as access control and point-of-sale. Serial communications interfaces ranging from a USB 2.0 Full-speed device, multiple UARTs, SPI, SSP to I2C-bus and on-chip SRAM of 8 kB up to 40 kB, make these devices very well suited for communication gateways and protocol converters, soft modems, voice recognition and low end imaging, providing both large buffer size and high processing power. Various 32-bit timers, single or dual 10-bit ADC(s), 10-bit DAC, PWM channels and 45 fast GPIO lines with up to nine edge or level sensitive external interrupt pins make these microcontrollers suitable for industrial control and medical systems.

5.2. Features

Key features

- 16-bit/32-bit ARM7TDMI-S microcontroller in a tiny LQFP64 package.
- 8 kB to 40 kB of on-chip static RAM and 32 kB to 512 kB of on-chip flash memory.
- 128-bit wide interface/accelerator enables high-speed 60 MHz operation.
- In-System Programming/In-Application Programming (ISP/IAP) via on-chip boot loader
- Software. Single flash sector or full chip erase in 400 ms and programming of 256 bytes in 1 ms.
- EmbeddedICE RT and Embedded Trace interfaces offer real-time debugging with the
- On-chip RealMonitor software and high-speed tracing of instruction execution.
- USB 2.0 Full-speed compliant device controller with 2 kB of endpoint RAM.
- In addition, the LPC2146/48 provides 8 kB of on-chip RAM accessible to USB by DMA.

- One or two (LPC2141/42 vs. LPC2144/46/48) 10-bit ADCs provide a total of 6/14 analog inputs, with conversion times as low as 2.44 μ s per channel.
- Single 10-bit DAC provides variable analog output (LPC2142/44/46/48 only).
- Two 32-bit timers/external event counters (with four capture and four compare channels each), PWM unit (six outputs) and watchdog.
- Low power Real-Time Clock (RTC) with independent power and 32 kHz clock input
- Multiple serial interfaces including two UARTs (16C550), two Fast I2C-bus (400 kbit/s), SPI and SSP with buffering and variable data length capabilities.
- Vectored Interrupt Controller (VIC) with configurable priorities and vector addresses.
- Up to 45 of 5 V tolerant fast general purpose I/O pins in a tiny LQFP64 package.
- Up to 21 external interrupt pins available.
- 60 MHz maximum CPU clock available from programmable on-chip PLL with settling
- Time of 100 μ s.
- On-chip integrated oscillator operates with an external crystal from 1 MHz to 25 MHz.
- Power saving modes include Idle and Power-down.
- Individual enable/disable of peripheral functions as well as peripheral clock scaling for
- Additional power optimization.
- Processor wake-up from Power-down mode via external interrupt or BOD.
- Single power supply chip with POR and BOD circuits:
- CPU operating voltage range of 3.0 V to 3.6 V ($3.3\text{ V} \pm 10\%$) with 5 V tolerant I/O

Ordering information:

Table 1: Ordering information

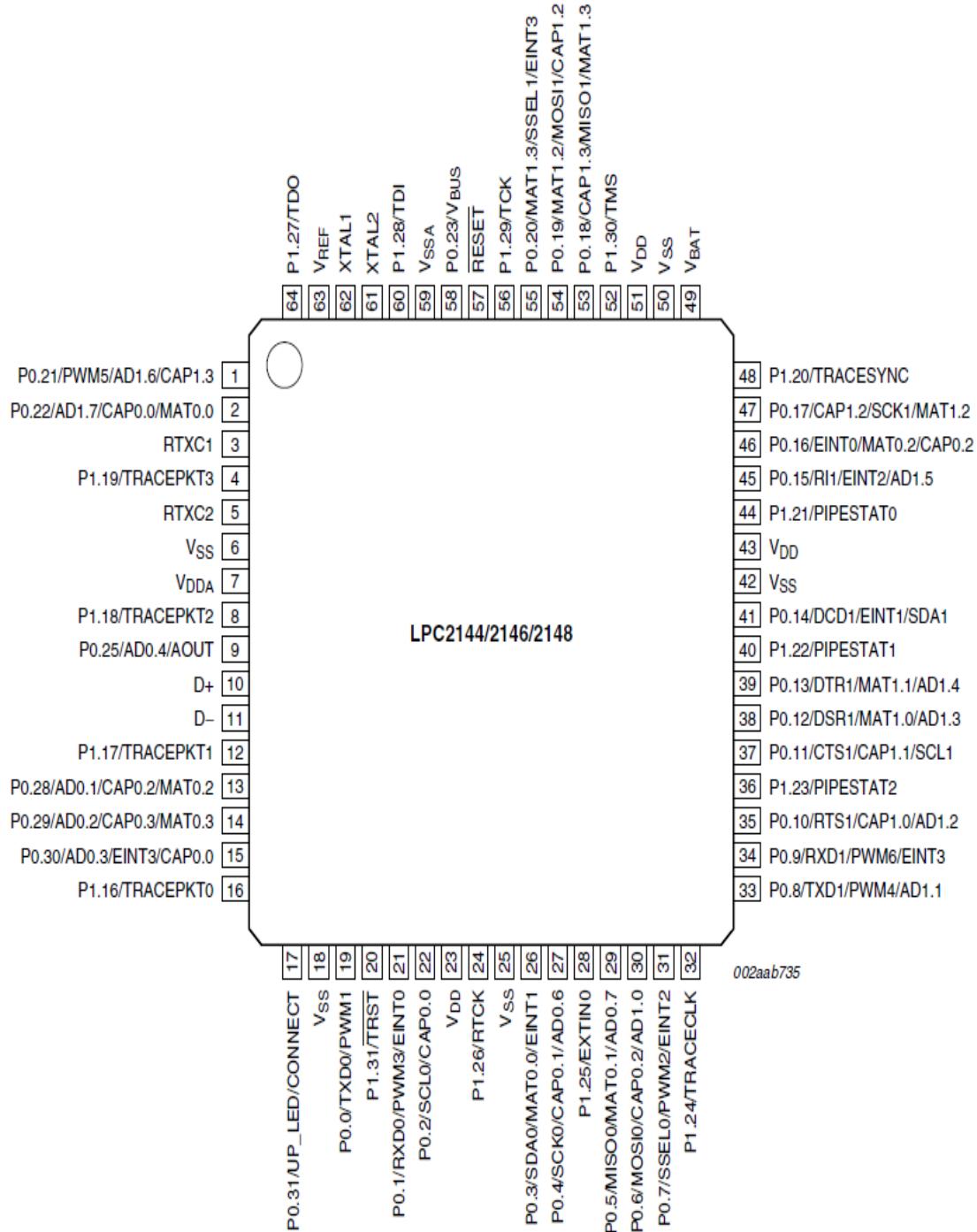
Type number	Package		
	Name	Description	Version
LPC2141FBD64	LQFP64	plastic low profile quad flat package; 64 leads; body 10 × 10 × 1.4 mm	SOT314-2
LPC2142FBD64			
LPC2144FBD64			
LPC2146FBD64			
LPC2148FBD64			

Ordering options

Table 2: Ordering options

Type number	Flash memory	RAM	Endpoint USB RAM	ADC (channels overall)	DAC	Temperature range (°C)
LPC2141FBD64	32 kB	8 kB	2 kB	1 (6 channels)	-	-40 to +85
LPC2142FBD64	64 kB	16 kB	2 kB	1 (6 channels)	1	-40 to +85
LPC2144FBD64	128 kB	16 kB	2 kB	2 (14 channels)	1	-40 to +85
LPC2146FBD64	256 kB	32 kB + 8 kB shared with USB DMA ^[1]	2 kB	2 (14 channels)	1	-40 to +85
LPC2148FBD64	512 kB	32 kB + 8 kB shared with USB DMA ^[1]	2 kB	2 (14 channels)	1	-40 to +85

5.3. Pin Diagram:



5.4. Port Pin Description:

Symbol	Pin	Type	Description
P0.0 to P0.31		I/O	Port 0: Port 0 is a 32-bit I/O port with individual direction controls for each bit. Total of 31 pins of the Port 0 can be used as a general purpose bidirectional digital I/Os while P0.31 is output only pin. The operation of port 0 pins depends upon the pin function selected via the pin connect block. Pins P0.24, P0.26 and P0.27 are not available.
P0.0/TXD0/ PWM1	19 ^[1]	I/O	P0.0 — General purpose input/output digital pin (GPIO).
		O	TXD0 — Transmitter output for UART0.
		O	PWM1 — Pulse Width Modulator output 1.
P0.1/RXD0/ PWM3/EINT0	21 ^[2]	I/O	P0.1 — General purpose input/output digital pin (GPIO).
		I	RXD0 — Receiver input for UART0.
		O	PWM3 — Pulse Width Modulator output 3.
		I	EINT0 — External interrupt 0 input
P0.2/SCL0/ CAP0.0	22 ^[3]	I/O	P0.2 — General purpose input/output digital pin (GPIO).
		I/O	SCL0 — I ² C0 clock input/output. Open-drain output (for I ² C-bus compliance).
		I	CAP0.0 — Capture input for Timer 0, channel 0.
P0.3/SDA0/ MAT0.0/EINT1	26 ^[3]	I/O	P0.3 — General purpose input/output digital pin (GPIO).
		I/O	SDA0 — I ² C0 data input/output. Open-drain output (for I ² C-bus compliance).
		O	MAT0.0 — Match output for Timer 0, channel 0.
		I	EINT1 — External interrupt 1 input.
P0.4/SCK0/ CAP0.1/AD0.6	27 ^[4]	I/O	P0.4 — General purpose input/output digital pin (GPIO).
		I/O	SCK0 — Serial clock for SPI0. SPI clock output from master or input to slave.
		I	CAP0.1 — Capture input for Timer 0, channel 0.
		I	AD0.6 — ADC 0, input 6.
P0.5/MISO0/ MAT0.1/AD0.7	29 ^[4]	I/O	P0.5 — General purpose input/output digital pin (GPIO).
		I/O	MISO0 — Master In Slave OUT for SPI0. Data input to SPI master or data output from SPI slave.
		O	MAT0.1 — Match output for Timer 0, channel 1.
		I	AD0.7 — ADC 0, input 7.
P0.6/MOSI0/ CAP0.2/AD1.0	30 ^[4]	I/O	P0.6 — General purpose input/output digital pin (GPIO).
		I/O	MOSI0 — Master Out Slave In for SPI0. Data output from SPI master or data input to SPI slave.
		I	CAP0.2 — Capture input for Timer 0, channel 2.
		I	AD1.0 — ADC 1, input 0. Available in LPC2144/46/48 only.
P0.7/SSEL0/ PWM2/EINT2	31 ^[2]	I/O	P0.7 — General purpose input/output digital pin (GPIO).
		I	SSEL0 — Slave Select for SPI0. Selects the SPI interface as a slave.
		O	PWM2 — Pulse Width Modulator output 2.
		I	EINT2 — External interrupt 2 input.
P0.8/TXD1/ PWM4/AD1.1	33 ^[4]	I/O	P0.8 — General purpose input/output digital pin (GPIO).
		O	TXD1 — Transmitter output for UART1.
		O	PWM4 — Pulse Width Modulator output 4.
		I	AD1.1 — ADC 1, input 1. Available in LPC2144/46/48 only.

Symbol	Pin	Type	Description
P0.9/RXD1/ PWM6/EINT3	34 [2]	I/O	P0.9 — General purpose input/output digital pin (GPIO).
		I	RXD1 — Receiver input for UART1.
		O	PWM6 — Pulse Width Modulator output 6.
		I	EINT3 — External interrupt 3 input.
P0.10/RTS1/ CAP1.0/AD1.2	35 [4]	I/O	P0.10 — General purpose input/output digital pin (GPIO).
		O	RTS1 — Request to Send output for UART1. LPC2144/46/48 only.
		I	CAP1.0 — Capture input for Timer 1, channel 0.
		I	AD1.2 — ADC 1, input 2. Available in LPC2144/46/48 only.
P0.11/CTS1/ CAP1.1/SCL1	37 [3]	I/O	P0.11 — General purpose input/output digital pin (GPIO).
		I	CTS1 — Clear to Send input for UART1. Available in LPC2144/46/48 only.
		I	CAP1.1 — Capture input for Timer 1, channel 1.
		I/O	SCL1 — I ² C1 clock input/output. Open-drain output (for I ² C-bus compliance)
P0.12/DSR1/ MAT1.0/AD1.3	38 [4]	I/O	P0.12 — General purpose input/output digital pin (GPIO).
		I	DSR1 — Data Set Ready input for UART1. Available in LPC2144/46/48 only.
		O	MAT1.0 — Match output for Timer 1, channel 0.
		I	AD1.3 — ADC input 3. Available in LPC2144/46/48 only.
P0.13/DTR1/ MAT1.1/AD1.4	39 [4]	I/O	P0.13 — General purpose input/output digital pin (GPIO).
		O	DTR1 — Data Terminal Ready output for UART1. LPC2144/46/48 only.
		O	MAT1.1 — Match output for Timer 1, channel 1.
		I	AD1.4 — ADC input 4. Available in LPC2144/46/48 only.
P0.14/DCD1/ EINT1/SDA1	41 [3]	I/O	P0.14 — General purpose input/output digital pin (GPIO).
		I	DCD1 — Data Carrier Detect input for UART1. LPC2144/46/48 only.
		I	EINT1 — External interrupt 1 input.
		I/O	SDA1 — I ² C1 data input/output. Open-drain output (for I ² C-bus compliance)
Note: LOW on this pin while $\overline{\text{RESET}}$ is LOW forces on-chip boot loader to take over control of the part after reset.			
P0.15/RI1/ EINT2/AD1.5	45 [4]	I/O	P0.15 — General purpose input/output digital pin (GPIO).
		I	RI1 — Ring Indicator input for UART1. Available in LPC2144/46/48 only.
		I	EINT2 — External interrupt 2 input.
		I	AD1.5 — ADC 1, input 5. Available in LPC2144/46/48 only.
P0.16/EINT0/ MAT0.2/CAP0.2	46 [2]	I/O	P0.16 — General purpose input/output digital pin (GPIO).
		I	EINT0 — External interrupt 0 input.
		O	MAT0.2 — Match output for Timer 0, channel 2.
		I	CAP0.2 — Capture input for Timer 0, channel 2.
P0.17/CAP1.2/ SCK1/MAT1.2	47 [1]	I/O	P0.17 — General purpose input/output digital pin (GPIO).
		I	CAP1.2 — Capture input for Timer 1, channel 2.
		I/O	SCK1 — Serial Clock for SSP. Clock output from master or input to slave.
		O	MAT1.2 — Match output for Timer 1, channel 2.

Symbol	Pin	Type	Description
P0.18/CAP1.3/ MISO1/MAT1.3	53 ^[1]	I/O	P0.18 — General purpose input/output digital pin (GPIO).
		I	CAP1.3 — Capture input for Timer 1, channel 3.
		I/O	MISO1 — Master In Slave Out for SSP. Data input to SPI master or data output from SSP slave.
		O	MAT1.3 — Match output for Timer 1, channel 3.
P0.19/MAT1.2/ MOSI1/CAP1.2	54 ^[1]	I/O	P0.19 — General purpose input/output digital pin (GPIO).
		O	MAT1.2 — Match output for Timer 1, channel 2.
		I/O	MOSI1 — Master Out Slave In for SSP. Data output from SSP master or input to SSP slave.
		I	CAP1.2 — Capture input for Timer 1, channel 2.
P0.20/MAT1.3/ SSEL1/EINT3	55 ^[2]	I/O	P0.20 — General purpose input/output digital pin (GPIO).
		O	MAT1.3 — Match output for Timer 1, channel 3.
		I	SSEL1 — Slave Select for SSP. Selects the SSP interface as a slave.
		I	EINT3 — External interrupt 3 input.
P0.21/PWM5/ AD1.6/CAP1.3	1 ^[4]	I/O	P0.21 — General purpose input/output digital pin (GPIO).
		O	PWM5 — Pulse Width Modulator output 5.
		I	AD1.6 — ADC 1, input 6. Available in LPC2144/46/48 only.
		I	CAP1.3 — Capture input for Timer 1, channel 3.
P0.22/AD1.7/ CAP0.0/MAT0.0	2 ^[4]	I/O	P0.22 — General purpose input/output digital pin (GPIO).
		I	AD1.7 — ADC 1, input 7. Available in LPC2144/46/48 only.
		I	CAP0.0 — Capture input for Timer 0, channel 0.
		O	MAT0.0 — Match output for Timer 0, channel 0.
P0.23/V _{BUS}	58 ^[1]	I/O	P0.23 — General purpose input/output digital pin (GPIO).
		I	V _{BUS} — Indicates the presence of USB bus power. Note: This signal must be HIGH for USB reset to occur.
P0.25/AD0.4/ AOUT	9 ^[5]	I/O	P0.25 — General purpose input/output digital pin (GPIO).
		I	AD0.4 — ADC 0, input 4.
P0.28/AD0.1/ CAP0.2/MAT0.2	13 ^[4]	I/O	P0.28 — General purpose input/output digital pin (GPIO).
		I	AD0.1 — ADC 0, input 1.
		I	CAP0.2 — Capture input for Timer 0, channel 2.
		O	MAT0.2 — Match output for Timer 0, channel 2.
P0.29/AD0.2/ CAP0.3/MAT0.3	14 ^[4]	I/O	P0.29 — General purpose input/output digital pin (GPIO).
		I	AD0.2 — ADC 0, input 2.
		I	CAP0.3 — Capture input for Timer 0, Channel 3.
		O	MAT0.3 — Match output for Timer 0, channel 3.
P0.30/AD0.3/ EINT3/CAP0.0	15 ^[4]	I/O	P0.30 — General purpose input/output digital pin (GPIO).
		I	AD0.3 — ADC 0, input 3.
		I	EINT3 — External interrupt 3 input.
		I	CAP0.0 — Capture input for Timer 0, channel 0.

Symbol	Pin	Type	Description
P0.31/UP_LED/ CONNECT	17 [6]	O	P0.31 — General purpose output only digital pin (GPO).
		O	UP_LED — USB Good Link LED indicator. It is LOW when device is configured (non-control endpoints enabled). It is HIGH when the device is not configured or during global suspend.
		O	CONNECT — Signal used to switch an external 1.5 kΩ resistor under the software control. Used with the SoftConnect USB feature. Important: This is an digital output only pin. This pin MUST NOT be externally pulled LOW when <u>RESET</u> pin is LOW or the JTAG port will be disabled.
P1.0 to P1.31		I/O	Port 1: Port 1 is a 32-bit bidirectional I/O port with individual direction controls for each bit. The operation of port 1 pins depends upon the pin function selected via the pin connect block. Pins 0 through 15 of port 1 are not available.
P1.16/ TRACEPKT0	16 [6]	I/O	P1.16 — General purpose input/output digital pin (GPIO).
		O	TRACEPKT0 — Trace Packet, bit 0. Standard I/O port with internal pull-up.
P1.17/ TRACEPKT1	12 [6]	I/O	P1.17 — General purpose input/output digital pin (GPIO).
		O	TRACEPKT1 — Trace Packet, bit 1. Standard I/O port with internal pull-up.
P1.18/ TRACEPKT2	8 [6]	I/O	P1.18 — General purpose input/output digital pin (GPIO).
		O	TRACEPKT2 — Trace Packet, bit 2. Standard I/O port with internal pull-up.
P1.19/ TRACEPKT3	4 [6]	I/O	P1.19 — General purpose input/output digital pin (GPIO).
		O	TRACEPKT3 — Trace Packet, bit 3. Standard I/O port with internal pull-up.
P1.20/ TRACESYNC	48 [6]	I/O	P1.20 — General purpose input/output digital pin (GPIO).
		O	TRACESYNC — Trace Synchronization. Standard I/O port with internal pull-up. Note: LOW on this pin while <u>RESET</u> is LOW enables pins P1.25:16 to operate as Trace port after reset.
P1.21/ PIPESTAT0	44 [6]	I/O	P1.21 — General purpose input/output digital pin (GPIO).
		O	PIPESTAT0 — Pipeline Status, bit 0. Standard I/O port with internal pull-up.
P1.22/ PIPESTAT1	40 [6]	I/O	P1.22 — General purpose input/output digital pin (GPIO).
		O	PIPESTAT1 — Pipeline Status, bit 1. Standard I/O port with internal pull-up.
P1.23/ PIPESTAT2	36 [6]	I/O	P1.23 — General purpose input/output digital pin (GPIO).
		O	PIPESTAT2 — Pipeline Status, bit 2. Standard I/O port with internal pull-up.
P1.24/ TRACECLK	32 [6]	I/O	P1.24 — General purpose input/output digital pin (GPIO).
		O	TRACECLK — Trace Clock. Standard I/O port with internal pull-up.
P1.25/EXTIN0	28 [6]	I/O	P1.25 — General purpose input/output digital pin (GPIO).
		I	EXTIN0 — External Trigger Input. Standard I/O with internal pull-up.
P1.26/RTCK	24 [6]	I/O	P1.26 — General purpose input/output digital pin (GPIO).
		I/O	RTCK — Returned Test Clock output. Extra signal added to the JTAG port. Assists debugger synchronization when processor frequency varies. Bidirectional pin with internal pull-up. Note: LOW on RTCK while <u>RESET</u> is LOW enables pins P1.31:26 to operate as Debug port after reset.
P1.27/TDO	64 [6]	I/O	P1.27 — General purpose input/output digital pin (GPIO).
		O	TDO — Test Data out for JTAG interface.

Symbol	Pin	Type	Description
P1.28/TDI	60 ^[6]	I/O	P1.28 — General purpose input/output digital pin (GPIO).
		I	TDI — Test Data in for JTAG interface.
P1.29/TCK	56 ^[6]	I/O	P1.29 — General purpose input/output digital pin (GPIO).
		I	TCK — Test Clock for JTAG interface.
P1.30/TMS	52 ^[6]	I/O	P1.30 — General purpose input/output digital pin (GPIO).
		I	TMS — Test Mode Select for JTAG interface.
P1.31/TRST	20 ^[6]	I/O	P1.31 — General purpose input/output digital pin (GPIO).
		I	TRST — Test Reset for JTAG interface.
D+	10 ^[7]	I/O	USB bidirectional D+ line.
D-	11 ^[7]	I/O	USB bidirectional D- line.
RESET	57 ^[8]	I	External reset input: A LOW on this pin resets the device, causing I/O ports and peripherals to take on their default states, and processor execution to begin at address 0. TTL with hysteresis, 5 V tolerant.
XTAL1	62 ^[9]	I	Input to the oscillator circuit and internal clock generator circuits.
XTAL2	61 ^[9]	O	Output from the oscillator amplifier.
RTXC1	3 ^[9]	I	Input to the RTC oscillator circuit.
RTXC2	5 ^[9]	O	Output from the RTC oscillator circuit.
V _{SS}	6, 18, 25, 42, 50	I	Ground: 0 V reference.
V _{SSA}	59	I	Analog ground: 0 V reference. This should nominally be the same voltage as V _{SS} , but should be isolated to minimize noise and error.
V _{DD}	23, 43, 51	I	3.3 V power supply: This is the power supply voltage for the core and I/O ports.
V _{DDA}	7	I	Analog 3.3 V power supply: This should be nominally the same voltage as V _{DD} but should be isolated to minimize noise and error. This voltage is only used to power the on-chip ADC(s) and DAC.
V _{REF}	63	I	ADC reference: This should be nominally less than or equal to the V _{DD} voltage but should be isolated to minimize noise and error. Level on this pin is used as a reference for ADC(s) and DAC.
V _{BAT}	49	I	RTC power supply: 3.3 V on this pin supplies the power to the RTC.

- [1] 5 V tolerant pad providing digital I/O functions with TTL levels and hysteresis and 10 ns slew rate control.
- [2] 5 V tolerant pad providing digital I/O functions with TTL levels and hysteresis and 10 ns slew rate control. If configured for an input function, this pad utilizes built-in glitch filter that blocks pulses shorter than 3 ns.
- [3] Open-drain 5 V tolerant digital I/O I²C-bus 400 kHz specification compatible pad. It requires external pull-up to provide an output functionality.
- [4] 5 V tolerant pad providing digital I/O (with TTL levels and hysteresis and 10 ns slew rate control) and analog input function. If configured for an input function, this pad utilizes built-in glitch filter that blocks pulses shorter than 3 ns. When configured as an ADC input, digital section of the pad is disabled.
- [5] 5 V tolerant pad providing digital I/O (with TTL levels and hysteresis and 10 ns slew rate control) and analog output function. When configured as the DAC output, digital section of the pad is disabled.
- [6] 5 V tolerant pad with built-in pull-up resistor providing digital I/O functions with TTL levels and hysteresis and 10 ns slew rate control. The pull-up resistor's value typically ranges from 60 kΩ to 300 kΩ.
- [7] Pad is designed in accordance with the Universal Serial Bus (USB) specification, revision 2.0 (Full-speed and Low-speed mode only).
- [8] 5 V tolerant pad providing digital input (with TTL levels and hysteresis) function only.
- [9] Pad provides special analog functionality.

5.5 Memory Organization

On-chip flash program memory:

The LPC2141/42/44/46/48 incorporate a 32 kB, 64 kB, 128 kB, 256 kB and 512 kB flash memory system respectively. This memory may be used for both code and data storage. Programming of the flash memory may be accomplished in several ways. It may be programmed In System via the serial port. The application program may also erase and/or program the flash while the application is running, allowing a great degree of flexibility for data storage field firmware upgrades, etc. Due to the architectural solution chosen for an on-chip boot loader, flash memory available for user's code on LPC2141/42/44/46/48 is 32 kB, 64 kB, 128 kB, 256 kB and 500 kB respectively. The LPC2141/42/44/46/48 flash memory provides a minimum of 100,000 erase/write cycles and 20 years of data-retention.

On-chip static RAM:

On-chip static RAM may be used for code and/or data storage. The SRAM may be accessed as 8-bit, 16-bit, and 32-bit. The LPC2141, LPC2142/44 and LPC2146/48 provide 8 kB, 16 kB and 32 kB of static RAM respectively. In case of LPC2146/48 only, an 8 kB SRAM block intended to be utilized mainly by the USB can also be used as a general purpose RAM for data storage and code storage and execution.

Memory map:

The LPC2141/42/44/46/48 memory map incorporates several distinct regions, as shown in Figure 5.

In addition, the CPU interrupt vectors may be remapped to allow them to reside in either flash memory (the default) or on-chip static RAM. This is described in Section 6.19

“System control”.

4.0 GB	AHB PERIPHERALS	0xFFFF FFFF
3.75 GB	VPB PERIPHERALS	0xF000 0000
3.5 GB		0xE000 0000
3.0 GB	RESERVED ADDRESS SPACE	0xC000 0000
2.0 GB	BOOT BLOCK (12 kB REMAPPED FROM ON-CHIP FLASH MEMORY)	0x8000 0000 0x7FFF FFFF
	RESERVED ADDRESS SPACE	0x7FFF D000 0x7FFF CFFF
	8 kB ON-CHIP USB DMA RAM (LPC2146/2148)	0x7FD0 2000 0x7FD0 1FFF
	RESERVED ADDRESS SPACE	0x7FD0 0000 0x7FCF FFFF
	32 kB ON-CHIP STATIC RAM (LPC2146)	0x4000 8000 0x4000 7FFF
	16 kB ON-CHIP STATIC RAM (LPC2142/2144)	0x4000 4000 0x4000 3FFF
	8 kB ON-CHIP STATIC RAM (LPC2141)	0x4000 2000 0x4000 1FFF
1.0 GB	RESERVED ADDRESS SPACE	0x4000 0000 0x3FFF FFFF
	TOTAL OF 512 kB ON-CHIP NON-VOLATILE MEMORY (LPC2148)	0x0008 0000 0x0007 FFFF
	TOTAL OF 256 kB ON-CHIP NON-VOLATILE MEMORY (LPC2146)	0x0004 0000 0x0003 FFFF
	TOTAL OF 128 kB ON-CHIP NON-VOLATILE MEMORY (LPC2144)	0x0002 0000 0x0001 FFFF
	TOTAL OF 64 kB ON-CHIP NON-VOLATILE MEMORY (LPC2142)	0x0001 0000 0x0000 FFFF
0.0 GB	TOTAL OF 32 kB ON-CHIP NON-VOLATILE MEMORY (LPC2141)	0x0000 8000 0x0000 7FFF 0x0000 0000

002aab558

5.6 Interrupt controller:

The Vectored Interrupt Controller (VIC) accepts all of the interrupt request inputs and categorizes them as Fast Interrupt Request (FIQ), vectored Interrupt Request (IRQ), and non-vectored IRQ as defined by programmable settings. The programmable assignment scheme means that priorities of interrupts from the various peripherals can be dynamically assigned and adjusted. Fast interrupt request (FIQ) has the highest priority. If more than one request is assigned to FIQ, the VIC combines the requests to produce the FIQ signal to the ARM processor. The fastest possible FIQ latency is achieved when only one request is classified as FIQ, because then the FIQ service routine does not need to branch into the interrupt service routine but can run from the interrupt vector location. If more than one request is assigned to the FIQ class, the FIQ service routine will read a word from the VIC that identifies which FIQ source(s) is (are) requesting an interrupt. Vectored IRQs have the middle priority. Sixteen of the interrupt requests can be assigned to this category. Any of the interrupt requests can be assigned to any of the 16 vectored IRQ slots, among which slot 0 has the highest priority and slot 15 has the lowest. Non-vectored IRQs have the lowest priority. The VIC combines the requests from all the vectored and non-vectored IRQs to produce the IRQ signal to the ARM processor. The IRQ service routine can start by reading a register from the VIC and jumping there. If any of the vectored IRQs are pending, the VIC provides the address of the highest-priority requesting IRQs service routine, otherwise it provides the address of a default routine that is shared by all the non-vectored IRQs. The default routine can read another VIC register to see what IRQs are active.

Interrupt sources:

Each peripheral device has one interrupt line connected to the Vectored Interrupt Controller, but may have several internal interrupt flags. Individual interrupt flags may also represent more than one interrupt source.

Pin connect block:

The pin connect block allows selected pins of the microcontroller to have more than one function. Configuration registers control the multiplexers to allow connection between the pin and the on chip peripherals. Peripherals should be connected to the appropriate pins prior to being activated,

and prior to any related interrupt(s) being enabled. Activity of any enabled peripheral function that is not mapped to a related pin should be considered undefined. The Pin Control Module with its pin select registers defines the functionality of the microcontroller in a given hardware environment. After reset all pins of Port 0 and 1 are configured as input with the following exceptions: If debug is enabled, the JTAG pins will assume their JTAG functionality; if trace is enabled, the Trace pins will assume their trace functionality. The pins associated with the I2C0 and I2C1 interface are open drain.

5.7 Fast general purpose parallel I/O (GPIO):

Device pins that are not connected to a specific peripheral function are controlled by the GPIO registers. Pins may be dynamically configured as inputs or outputs. Separate registers allow setting or clearing any number of outputs simultaneously. The value of the output register may be read back, as well as the current state of the port pins. LPC2141/42/44/46/48 introduce accelerated GPIO functions over prior LPC2000 devices:

- GPIO registers are relocated to the ARM local bus for the fastest possible I/O timing.
- Mask registers allow treating sets of port bits as a group, leaving other bits unchanged.
- All GPIO registers are byte addressable.
- Entire port value can be written in one instruction.

Features:

- Bit-level set and clear registers allow a single instruction set or clear of any number of bits in one port.
- Direction control of individual bits.
- Separate control of output set and clear.
- All I/O default to inputs after reset.

5.8 10-bit ADC:

The LPC2141/42 contain one and the LPC2144/46/48 contain two analog to digital converters. These converters are single 10-bit successive approximation analog to digital converters. While ADC0 has six channels, ADC1 has eight channels. Therefore, total number of available ADC inputs for LPC2141/42 is 6 and for LPC2144/46/48 is 14. The LPC2141/42 contain one and the LPC2144/46/48 contain two analog to digital converters. These converters are single 10-bit successive approximation analog to digital converters. While ADC0 has six channels, ADC1 has eight channels. Therefore, total number of available ADC inputs for LPC2141/42 is 6 and for LPC2144/46/48 is 14.

Features:

- 10 bit successive approximation analog to digital converter.
- Measurement range of 0 V to VREF ($2.0 \text{ V} \leq \text{VREF} \leq \text{VDDA}$).
- Each converter capable of performing more than 400,000 10-bit samples per second.
- Every analog input has a dedicated result register to reduce interrupt overhead.
- Burst conversion mode for single or multiple inputs.
- Optional conversion on transition on input pin or timer match signal.
- Global Start command for both converters (LPC2142/44/46/48 only).

5.9 10-bit DAC:

The DAC enables the LPC2141/42/44/46/48 to generate a variable analog output. The maximum DAC output voltage is the VREF voltage.

Features:

- 10-bit DAC.
- Buffered output.
- Power-down mode available.
- Selectable speed versus power.

5.10 USB 2.0 device controller:

The USB is a 4-wire serial bus that supports communication between a host and a number (127 max) of peripherals. The host controller allocates the USB bandwidth to attached devices through a token based protocol. The bus supports hot plugging, unplugging, and dynamic configuration of the devices. All transactions are initiated by the host controller. The LPC2141/42/44/46/48 is equipped with a USB device controller that enables 12 Mbit/s data exchange with a USB host controller. It consists of a register interface, serial interface engine, endpoint buffer memory and DMA controller. The serial interface engine decodes the USB data stream and writes data to the appropriate end point buffer memory. The status of a completed USB transfer or error condition is indicated via status registers. An interrupt is also generated if enabled. A DMA controller (available in LPC2146/48 only) can transfer data between an endpoint buffer and the USB RAM.

Features:

- Fully compliant with USB 2.0 Full-speed specification.
- Supports 32 physical (16 logical) endpoints.
- Supports control, bulk, interrupt and isochronous endpoints.
- Scalable realization of endpoints at run time.
- Endpoint maximum packet size selection (up to USB maximum specification) by software at run time.
- RAM message buffer size based on endpoint realization and maximum packet size.
- Supports SoftConnect and GoodLink LED indicator. These two functions are sharing one pin.

- Supports bus-powered capability with low suspend current.
- Supports DMA transfer on all non-control endpoints (LPC2146/48 only).
- One duplex DMA channel serves all endpoints (LPC2146/48 only).
- Allows dynamic switching between CPU controlled and DMA modes (only in LPC2146/48).
- Double buffer implementation for bulk and isochronous endpoints

5.11 UARTs:

The LPC2141/42/44/46/48 each contain two UARTs. In addition to standard transmit and receive data lines, the LPC2144/46/48 UART1 also provides a full modem control handshake interface. Compared to previous LPC2000 microcontrollers, UARTs in LPC2141/42/44/46/48 introduce a fractional baud rate generator for both UARTs, enabling these microcontrollers to achieve standard baudrates such as 115200 with any crystal frequency above 2 MHz. In addition, auto-CTS/RTS flow-control functions are fully implemented in hardware (UART1 in LPC2144/46/48 only).

Features:

- 16 byte Receive and Transmit FIFOs.
- Register locations conform to '550 industry standard.
- Receiver FIFO trigger points at 1, 4, 8, and 14 bytes
- Built-in fractional baud rate generator covering wide range of baud rates without a need for external crystals of particular values.
- Transmission FIFO control enables implementation of software (XON/XOFF) flow control on both UARTs.
- LPC2144/46/48 UART1 equipped with standard modem interface signals. This module also provides full support for hardware flow control (auto-CTS/RTS)

UART0 pin description

Pin	Type	Description
RXD0	Input Serial Input.	Serial receive data.
TXD0	Output Serial Output.	Serial transmit data

REGISTER DESCRIPTION UART0

Name	Description	Bit functions and addresses								Access	Reset value ^[1]	Address	
		MSB BIT7 BIT6 BIT5 BIT4 BIT3 BIT2 BIT1 BIT0 LSB											
U0RBR	Receiver Buffer Register	8-bit Read Data								RO	NA	0xE000 C000 (DLAB=0)	
U0THR	Transmit Holding Register	8-bit Write Data								WO	NA	0xE000 C000 (DLAB=0)	
U0DLL	Divisor Latch LSB	8-bit Data								R/W	0x01	0xE000 C000 (DLAB=1)	
U0DLM	Divisor Latch MSB	8-bit Data								R/W	0x00	0xE000 C004 (DLAB=1)	
U0IER	Interrupt Enable Register	-	-	-	-	-	-	En.ABTO	En.ABEO	R/W	0x00	0xE000 C004 (DLAB=0)	
		-	-	-	-	-	-	En.RX	Enable				
U0IIR	Interrupt ID Reg.	-	-	-	-	-	-	Lin.St.Int	THRE Int	R/W	0x01	0xE000 C008	
		FIFOs Enabled				-	-	IIR3	IIR2				
U0FCR	FIFO Control Register	RX Trigger				-	-	-	TX FIFO Reset	RX FIFO Reset	W/O	0x00	0xE000 C008
U0LCR	Line Control Register	DLAB	Set Break	Stick Parity	Even Par.Selct.	Parity Enable	No. of Stop Bits	Word Length Select			R/W	0x00	0xE000 C00C
U0LSR	Line Status Register	RX FIFO Error	TEM	THRE	BI	FE	PE	OE	DR	RO	0x60	0xE000 C014	
U0SCR	Scratch Pad Reg.	8-bit Data								R/W	0x00	0xE000 C01C	
U0ACR	Auto-baud Control Register	-	-	-	-	-	-	ABTO	ABEO	R/W	0x00	0xE000 C020	
		-	-	-	-	-	-	Int.Clr	Int.Clr				
U0FDR	Fractional Divider Register	Reserved[31:8]								R/W	0x10	0xE000 C028	
		MulVal											
U0TER	TX. Enable Reg.	TXEN	-	-	-	-	-	-	-	R/W	0x80	0xE000 C030	

UART 1

FEATURES

UART1 is identical to UART0, with the addition of a modem interface.

- 16 byte Receive and Transmit FIFOs.
- Register locations conform to '550 industry standard.
- Receiver FIFO trigger points at 1, 4, 8, and 14 bytes.
- Built-in fractional baud rate generator with autobauding capabilities.
- Mechanism that enables software and hardware flow control implementation.
- Standard modem interface signals included with flow control (auto-CTS/RTS) fully supported in hardware (LPC2144/6/8 only).

PIN DESCRIPTION

Pin	Type	Description
RXD1	Input	Serial Input. Serial receive data.
TXD1	Output	Serial Output. Serial transmit data.
CTS1 ^{U1}	Input	Clear To Send. Active low signal indicates if the external modem is ready to accept transmitted data via TXD1 from the UART1. In normal operation of the modem interface (U1MCR[4] = 0), the complement value of this signal is stored in U1MSR[4]. State change information is stored in U1MSR[0] and is a source for a priority level 4 interrupt, if enabled (U1IER[3] = 1).
DCD1 ^{U1}	Input	Data Carrier Detect. Active low signal indicates if the external modem has established a communication link with the UART1 and data may be exchanged. In normal operation of the modem interface (U1MCR[4]=0), the complement value of this signal is stored in U1MSR[7]. State change information is stored in U1MSR3 and is a source for a priority level 4 interrupt, if enabled (U1IER[3] = 1).
DSR1 ^{U1}	Input	Data Set Ready. Active low signal indicates if the external modem is ready to establish a communications link with the UART1. In normal operation of the modem interface (U1MCR[4] = 0), the complement value of this signal is stored in U1MSR[5]. State change information is stored in U1MSR[1] and is a source for a priority level 4 interrupt, if enabled (U1IER[3] = 1).
DTR1 ^{U1}	Output	Data Terminal Ready. Active low signal indicates that the UART1 is ready to establish connection with external modem. The complement value of this signal is stored in U1MCR[0].
RI1 ^{U1}	Input	Ring Indicator. Active low signal indicates that a telephone ringing signal has been detected by the modem. In normal operation of the modem interface (U1MCR[4] = 0), the complement value of this signal is stored in U1MSR[6]. State change information is stored in U1MSR[2] and is a source for a priority level 4 interrupt, if enabled (U1IER[3] = 1).
RTS1 ^{U1}	Output	Request To Send. Active low signal indicates that the UART1 would like to transmit data to the external modem. The complement value of this signal is stored in U1MCR[1].

REGISTER DESCRIPTION OF UART1

Name	Description	Bit functions and addresses								Access	Reset value ^[1]	Address
		MSB BIT7 BIT6 BIT5 BIT4 BIT3 BIT2 BIT1 BIT0 LSB										
U1RBR	Receiver Buffer Register			8-bit Read Data						RO	NA	0xE001 0000 (DLAB=0)
U1THR	Transmit Holding Register			8-bit Write Data						WO	NA	0xE001 0000 (DLAB=0)
U1DLL	Divisor Latch LSB			8-bit Data						R/W	0x01	0xE001 0000 (DLAB=1)
U1DLM	Divisor Latch MSB			8-bit Data						R/W	0x00	0xE001 0004 (DLAB=1)
U1IER	Interrupt Enable Register	-	-	-	-	-	-	En.ABTO	En.ABEO	R/W	0x00	0xE001 0004 (DLAB=0)
		En.CTS Int ^[2]	-	-	-	E.Modem St.Int ^[2]	En. RX Lin.St. Int	Enable THRE Int	En. RX Dat.Av.Int			
U1IIR	Interrupt ID Reg.	-	-	-	-	-	-	ABTO Int	ABEO Int	RO	0x01	0xE001 0008
		FIFOs Enabled		-	-	IIR3	IIR2	IIR1	IIR0			
U1FCR	FIFO Control Register	RX Trigger		-	-	-	TX FIFO Reset	RX FIFO Reset	FIFO Enable	WO	0x00	0xE001 0008
U1LCR	Line Control Register	DLAB	Set Break	Stick Parity	Even Par Selct.	Parity Enable	No. of Stop Bits	Word Length Select		R/W	0x00	0xE001 000C
U1MCR ^[2]	Modem Ctrl. Reg.	CTSen	RTSen	-	LoopBck.	-	-	RTS	DTR	R/W	0x00	0xE001 0010
U1LSR	Line Status Register	RX FIFO Error	TEM	THRE	BI	FE	PE	OE	DR	RO	0x60	0xE001 0014
U1MSR ^[2]	Modem Status Register	DCD	RI	DSR	CTS	Delta DCD	Trailing Edge RI	Delta DSR	Delta CTS	RO	0x00	0xE001 0018
U1SCR	Scratch Pad Reg.			8-bit Data						R/W	0x00	0xE001 001C
U1ACR	Auto-baud Control Register	-	-	-	-	-	-	ABTO IntClr	ABEO IntClr	R/W	0x00	0xE001 0020
		-	-	-	-	-	-	Aut.Rstrt.	Mode			
U1FDR	Fractional Divider Register			Reserved[31:8]						R/W	0x10	0xE001 0028
		MulVal		DivAddVal								
U1TER	TX. Enable Reg.	TXEN	-	-	-	-	-	-	-	R/W	0x80	0xE001 0030

5.12 I2C-bus serial I/O controller:

The LPC2141/42/44/46/48 each contain two I2C-bus controllers. The I2C-bus is bidirectional, for inter-IC control using only two wires: a serial clock line (SCL), and a serial data line (SDA). Each device is recognized by a unique address and can operate as either a receiver-only device (e.g., an LCD driver or a transmitter with the capability to both receive and send information (such as memory)). Transmitters and/or receivers can operate in either master or slave mode, depending on whether the chip has to initiate a data transfer or is only addressed. The I2C-bus is a multi-master bus, it can be controlled by more than one bus master connected to it. The I2C-bus implemented in LPC2141/42/44/46/48 supports bit rates up to 400 kbit/s (Fast I2C-bus).

Features:

- Compliant with standard I2C-bus interface.
- Easy to configure as master, slave, or master/slave.
- Programmable clocks allow versatile rate control.
- Bidirectional data transfer between masters and slaves.
- Multi-master bus (no central master).
- Arbitration between simultaneously transmitting masters without corruption of serial data on the bus.
- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus.
- Serial clock synchronization can be used as a handshake mechanism to suspend and resume serial transfer.
- The I2C-bus can be used for test and diagnostic purposes.

5.13 SPI serial I/O controller:

The LPC2141/42/44/46/48 each contain one SPI controller. The SPI is a full duplex serial interface, designed to handle multiple masters and slaves connected to a given bus. Only a single master and a single slave can communicate on the interface during a given data transfer. During a

data transfer the master always sends a byte of data to the slave, and the slave always sends a byte of data to the master.

Features

- Compliant with Serial Peripheral Interface (SPI) specification.
- Synchronous, Serial, Full Duplex, Communication.
- Combined SPI master and slave.
- Maximum data bit rate of one eighth of the input clock rate.

SSP serial I/O controller

The LPC2141/42/44/46/48 each contain one SSP. The SSP controller is capable of operation on a SPI, 4-wire SSI, or Microwire bus. It can interact with multiple masters and slaves on the bus. However, only a single master and a single slave can communicate on the bus during a given data transfer. The SSP supports full duplex transfers, with data frames of 4 bits to 16 bits of data flowing from the master to the slave and from the slave to the master. Often only one of these data flows carries meaningful data.

Features

- Compatible with Motorola's SPI, TI's 4-wire SSI and National Semiconductor's Microwire buses.
- Synchronous serial communication.
- Master or slave operation.
- 8-frame FIFOs for both transmit and receive.
- Four bits to 16 bits per frame.

General purpose timers/external event counters

The Timer/Counter is designed to count cycles of the peripheral clock (PCLK) or an externally supplied clock and optionally generate interrupts or perform other actions at specified timer

values, based on four match registers. It also includes four capture inputs to trap the timer value when an input signal transitions, optionally generating an interrupt. Multiple pins can be selected to perform a single capture or match function, providing an application with ‘or’ and ‘and’, as well as ‘broadcast’ functions among them.

The LPC2141/42/44/46/48 can count external events on one of the capture inputs if the minimum external pulse is equal or longer than a period of the PCLK. In this configuration, unused capture lines can be selected as regular timer capture inputs, or used as external interrupts.

Features

- A 32-bit timer/counter with a programmable 32-bit prescaler.
- External event counter or timer operation.
- Four 32-bit capture channels per timer/counter that can take a snapshot of the timer value when an input signal transitions. A capture event may also optionally generate an interrupt.
- Four 32-bit match registers that allow:
 - Continuous operation with optional interrupt generation on match.
 - Stop timer on match with optional interrupt generation.
 - Reset timer on match with optional interrupt generation.
- Four external outputs per timer/counter corresponding to match registers, with the following capabilities:
 - Set LOW on match.
 - Set HIGH on match.
 - Toggle on match.

- Do nothing on match.

Watchdog timer

The purpose of the watchdog is to reset the microcontroller within a reasonable amount of time if it enters an erroneous state. When enabled, the watchdog will generate a system reset if the user program fails to ‘feed’ (or reload) the watchdog within a predetermined amount of time.

Features

- Internally resets chip if not periodically reloaded.
- Debug mode.
- Enabled by software but requires a hardware reset or a watchdog reset/interrupt to be disabled.
- Incorrect/Incomplete feed sequence causes reset/interrupt if enabled.
- Flag to indicate watchdog reset.
- Programmable 32-bit timer with internal pre-scaler.
- Selectable time period from $(TPCLK \times 256 \times 4)$ to $(TPCLK \times 232 \times 4)$ in multiples of $TPCLK \times 4$.

5.14 Real-time clock

The RTC is designed to provide a set of counters to measure time when normal or idle operating mode is selected. The RTC has been designed to use little power, making it suitable for battery powered systems where the CPU is not running continuously (Idle mode).

Features

- Measures the passage of time to maintain a calendar and clock.
- Ultra-low power design to support battery powered systems.

- Provides Seconds, Minutes, Hours, Day of Month, Month, Year, Day of Week, and Day of Year.
- Can use either the RTC dedicated 32 kHz oscillator input or clock derived from the external crystal/oscillator input at XTAL1. Programmable reference clock divider allows fine adjustment of the RTC.
- Dedicated power supply pin can be connected to a battery or the main 3.3 V.

5.15 Pulse width modulator

The PWM is based on the standard timer block and inherits all of its features, although only the PWM function is pinned out on the LPC2141/42/44/46/48. The timer is designed to count cycles of the peripheral clock (PCLK) and optionally generate interrupts or perform other actions when specified timer values occur, based on seven match registers. The PWM function is also based on match register events.

The ability to separately control rising and falling edge locations allows the PWM to be used for more applications. For instance, multi-phase motor control typically requires three non-overlapping PWM outputs with individual control of all three pulse widths and positions. Two match registers can be used to provide a single edge controlled PWM output. One match register (MR0) controls the PWM cycle rate, by resetting the count upon match. The other match register controls the PWM edge position. Additional single edge controlled PWM outputs require only one match register each, since the repetition rate is the same for all PWM outputs. Multiple single edge controlled PWM outputs will all have a rising edge at the beginning of each PWM cycle, when an MR0 match occurs. Three match registers can be used to provide a PWM output with both edges controlled. Again, the MR0 match register controls the PWM cycle rate. The other match registers control the two PWM edge positions. Additional double edge controlled PWM outputs require only two match registers each, since the repetition rate is the same for all PWM outputs.

With double edge controlled PWM outputs, specific match registers control the rising and falling edge of the output. This allows both positive going PWM pulses (when the rising edge occurs prior to the falling edge), and negative going PWM pulses (when the falling edge occurs prior to the rising edge).

Features

- Seven match registers allow up to six single edge controlled or three double edge controlled PWM outputs, or a mix of both types.
- The match registers also allow:
 - Continuous operation with optional interrupt generation on match.
 - Stop timer on match with optional interrupt generation.
 - Reset timer on match with optional interrupt generation.
- Supports single edge controlled and/or double edge controlled PWM outputs. Single edge controlled PWM outputs all go HIGH at the beginning of each cycle unless the output is a constant LOW. Double edge controlled PWM outputs can have either edge occur at any position within a cycle. This allows for both positive going and negative going pulses.
- Pulse period and width can be any number of timer counts. This allows complete flexibility in the trade-off between resolution and repetition rate. All PWM outputs will occur at the same repetition rate.
- Double edge controlled PWM outputs can be programmed to be either positive going or negative going pulses.
- Match register updates are synchronized with pulse outputs to prevent generation of erroneous pulses. Software must ‘release’ new match values before they can become effective.
- May be used as a standard timer if the PWM mode is not enabled.
- A 32-bit Timer/Counter with a programmable 32-bit Prescaler.

5.16 System control

5.16.1 Crystal oscillator

On-chip integrated oscillator operates with external crystal in range of 1 MHz to 25 MHz. The oscillator output frequency is called *fosc* and the ARM processor clock frequency is referred to as *CCLK* for purposes of rate equations, etc. *fosc* and *CCLK* are the same value unless the PLL is running and connected. Refer to Section 6.19.2 “PLL” for additional information.

5.16.2 PLL

The PLL accepts an input clock frequency in the range of 10 MHz to 25 MHz. The input frequency is multiplied up into the range of 10 MHz to 60 MHz with a Current Controlled Oscillator (CCO). The multiplier can be an integer value from 1 to 32 (in practice, the multiplier value cannot be higher than 6 on this family of microcontrollers due to the upper frequency limit of the CPU). The CCO operates in the range of 156 MHz to 320 MHz, so there is an additional divider in the loop to keep the CCO within its frequency range while the PLL is providing the desired output frequency. The output divider may be set to divide by 2, 4, 8, or 16 to produce the output clock. Since the minimum output divider value is 2, it is insured that the PLL output has a 50 % duty cycle. The PLL is turned off and bypassed following a chip reset and may be enabled by software. The program must configure and activate the PLL, wait for the PLL to Lock, then connect to the PLL as a clock source. The PLL settling time is 100 μ s.

5.16.3 Reset and wake-up timer

Reset has two sources on the LPC2141/42/44/46/48: the RESET pin and watchdog reset. The RESET pin is a Schmitt trigger input pin with an additional glitch filter. Assertion of chip reset by any source starts the Wake-up Timer (see Wake-up Timer description below), causing the internal chip reset to remain asserted until the external reset is de-asserted, the oscillator is running, a fixed number of clocks have passed, and the on-chip flash controller has completed its initialization. When the internal reset is removed, the processor begins executing at address 0, which is the reset vector. At that point, all of the processor and peripheral registers have been initialized to predetermined values. The Wake-up Timer ensures that the oscillator and other analog functions required for chip operation are fully functional before the processor is allowed to execute instructions. This is important at power on, all types of reset, and whenever any of the aforementioned functions are turned off for any reason. Since the oscillator and other functions are turned off during Power-down mode, any wake-up of the processor from Power-down mode makes use of the Wake-up Timer. The Wake-up Timer monitors the crystal oscillator as the means of checking whether it is safe to begin code execution. When power is applied to the chip, or some event caused the chip to exit Power-down mode, some time is required for the oscillator to produce a signal of sufficient amplitude to drive the clock logic. The amount of time depends on many factors, including the rate of VDD ramp (in the case of power on), the type of crystal and its electrical characteristics (if a quartz crystal is used), as well as any other external circuitry (e.g. capacitors), and the characteristics of the oscillator itself under the existing ambient conditions.

5.16.4 Brownout detector

The LPC2141/42/44/46/48 include 2-stage monitoring of the voltage on the VDD pins. If this voltage falls below 2.9 V, the BOD asserts an interrupt signal to the VIC. This signal can be enabled for interrupt; if not, software can monitor the signal by reading dedicated register. The second stage of low voltage detection asserts reset to inactivate the LPC2141/42/44/46/48 when the voltage on the VDD pins falls below 2.6 V. This reset prevents alteration of the flash as operation of the various elements of the chip would otherwise become unreliable due to low voltage. The BOD circuit maintains this reset down below 1 V, at which point the POR circuitry maintains the overall reset. Both the 2.9 V and 2.6 V thresholds include some hysteresis. In normal operation, this hysteresis allows the 2.9 V detection to reliably interrupt, or a regularly-executed event loop to sense the condition.

5.16.5 Code security

This feature of the LPC2141/42/44/46/48 allow an application to control whether it can be debugged or protected from observation. If after reset on-chip boot loader detects a valid checksum in flash and reads 0x8765 4321 from address 0x1FC in flash, debugging will be disabled and thus the code in flash will be protected from observation. Once debugging is disabled, it can be enabled only by performing a full chip erase using the ISP

5.16.6 External interrupt inputs

The LPC2141/42/44/46/48 include up to nine edge or level sensitive External Interrupt Inputs as selectable pin functions. When the pins are combined, external events can be processed as four independent interrupt signals. The External Interrupt Inputs can optionally be used to wake-up the processor from Power-down mode. Additionally capture input pins can also be used as external interrupts without the option to wake the device up from Power-down mode.

5.16.7 Memory mapping control

The Memory Mapping Control alters the mapping of the interrupt vectors that appear beginning at address 0x0000 0000. Vectors may be mapped to the bottom of the on-chip flash memory, or to the on-chip static RAM. This allows code running in different memory spaces to have control of the interrupts

5.16.8 Power control

The LPC2141/42/44/46/48 supports two reduced power modes: Idle mode and Power-down mode. In Idle mode, execution of instructions is suspended until either a reset or interrupt occurs. Peripheral functions continue operation during idle mode and may generate interrupts to cause the processor to resume execution. Idle mode eliminates power used by the processor itself, memory systems and related controllers, and internal buses. In Power-down mode, the oscillator is shut down and the chip receives no internal clocks. The processor state and registers, peripheral

registers, and internal SRAM values are preserved throughout Power-down mode and the logic levels of chip output pins remain static. The Power-down mode can be terminated and normal operation resumed by either a reset or certain specific interrupts that are able to function without clocks. Since all dynamic operation of the chip is suspended, Power-down mode reduces chip power consumption to nearly zero. Selecting an external 32 kHz clock instead of the PCLK as a clock-source for the on-chip RTC will enable the microcontroller to have the RTC active during Power-down mode. Power-down current is increased with RTC active. However, it is significantly lower than in Idle mode. A Power Control for Peripherals feature allows individual peripherals to be turned off if they are not needed in the application, resulting in additional power savings during active and idle mode.

5.16.9 VPB bus

The VPB divider determines the relationship between the processor clock (CCLK) and the clock used by peripheral devices (PCLK). The VPB divider serves two purposes. The first is to provide peripherals with the desired PCLK via VPB bus so that they can operate at the speed chosen for the ARM processor. In order to achieve this, the VPB bus may be slowed down to 1/2 to 1/4 of the processor clock rate. Because the VPB bus must work properly at power-up (and its timing cannot be altered if it does not work since the VPB divider control registers reside on the VPB bus), the default condition at reset is for the VPB bus to run at 1/4 of the processor clock rate. The second purpose of the VPB divider is to allow power savings when an application does not require any peripherals to run at the full processor rate. Because the VPB divider is connected to the PLL output, the PLL remains active (if it was running) during Idle mode.

5.16.10. Emulation and debugging

The LPC2141/42/44/46/48 support emulation and debugging via a JTAG serial port. A trace port allows tracing program execution. Debugging and trace functions are multiplexed only with GPIOs on Port 1. This means that all communication, timer and interface peripherals residing o

5.16.11. Embedded ICE

Standard ARM Embedded ICE logic provides on-chip debug support. The debugging of the target system requires a host computer running the debugger software and an Embedded ICE protocol convertor. Embedded ICE protocol convertor converts the remote debug protocol commands to the JTAG data needed to access the ARM core. The ARM core has a Debug Communication Channel (DCC) function built-in. The DCC allows a program running on the target to communicate with the host debugger or another separate host without stopping the program flow or even entering the debug state. The DCC is accessed as a co-processor 14 by the program running on the ARM7TDMI-S core. The DCC allows the JTAG port to be used for sending and receiving data without affecting the normal program flow. The DCC data and control registers are mapped in to addresses in the Embedded ICE logic.

5.16.12. Embedded trace

Since the LPC2141/42/44/46/48 has significant amounts of on-chip memory, it is not possible to determine how the processor core is operating simply by observing the external pins. The Embedded Trace Macro cell (ETM) provides real-time trace capability for deeply embedded processor cores. It outputs information about processor execution to the trace port. The ETM is connected directly to the ARM core and not to the main AMBA system bus. It compresses the trace information and exports it through a narrow trace port. An external trace port analyzer must capture the trace information under software debugger control. Instruction trace (or PC trace) shows the flow of execution of the processor and provides a list of all the instructions that were executed. Instruction trace is significantly compressed

by only broadcasting branch addresses as well as a set of status signals that indicate the pipeline status on a cycle by cycle basis. Trace information generation can be controlled by selecting the trigger resource. Trigger resources include address comparators, counters and sequencers. Since trace information is compressed the software debugger requires a static image of the code being executed. Self-modifying code can not be traced because of this restriction.

Real Monitor:

Real Monitor is a configurable software module, developed by ARM Inc., which enables real-time debug. It is a lightweight debug monitor that runs in the background while users debug their foreground application. It communicates with the host using the DCC, which is present in the Embedded ICE logic. The LPC2141/42/44/46/48 contain a specific configuration of Real Monitor software programmed into the on-chip flash memory.

ARM7 LPC2148 is ARM7TDMI-S Core Board Microcontroller that uses 16/32-Bit 64 Pin (LQFP) Microcontroller No.LPC2148 from Philips (NXP). All resources inside LPC2148 is quite perfect, so it is the most suitable to learn and study because if user can learn and understand the applications of all resources inside MCU well, it makes user can modify, apply and develop many excellent applications in the future. Because Hardware system of LPC2148 includes the necessary devices within only one MCU such as USB, ADC, DAC, Timer/Counter, PWM, Capture, I2C, SPI, UART, and etc.

5.17 .Board Technical Specifications

Processor* : LPC2148

Clock speed : 11.0592 MHz / 22.1184 MHz

Clock Divisors : 6 (or) 12

Real time Clock : DS1307 on i2c Bus /w Battery

Data Memory : 24LCxx on i2c Bus

LCD : 16x2 Backlight

LED indicators : Power

RS-232 : +9V -9V levels

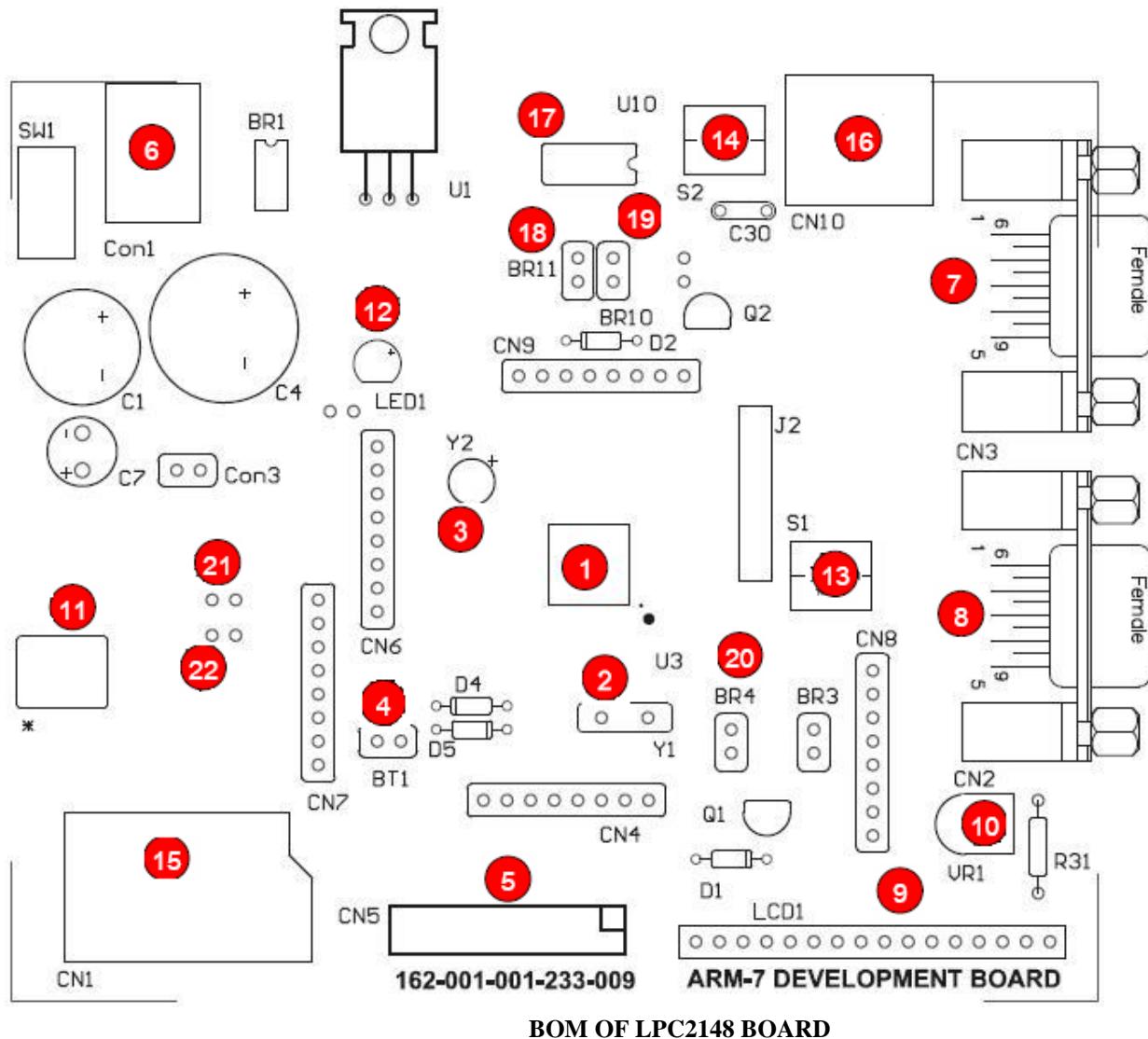
Power : 7-15V AC/DC @ 500 mA

Voltage Regulator : 5V Onboard LM7805

Specifications of Board:

- Use 16/32 Bit ARM7TDMI-S MCU No.LPC2148 from Philips (NXP)
- Has 512KB Flash Memory and 40KB Static RAM internal MCU
- Use 12.00MHz Crystal, so MCU can process data with the maximum high speed at 60MHz when using it with Phase-Locked Loop (PLL) internal MCU.
- Has RTC Circuit (Real Time Clock) with 32.768 KHz XTAL and Battery Backup.
- Support In-System Programming (ISP) and In-Application Programming (IAP) through On-Chip Boot-Loader Software via Port UART-0 (RS232)
- Has circuit to connect with standard 20 Pin JTAG ARM for Real Time Debugging
- 7-12V AC/DC Power Supply.
- Has standard 2.0 USB as Full Speed inside (USB Function has 32 End Point)
- Has Circuit to connect with Dot-Matrix LCD with circuit to adjust its contrast by using 16 PIN Connector.
- Has RS232 Communication Circuit by using 2 Channel.
- Has SD/MMC card connector circuit by using SSP.
- Has EEPROM interface using I2C.
- Has PS2 keyboard interface.
- All port pins are extracted externally for further interfaces.

5.18. BOM of LPC2148:

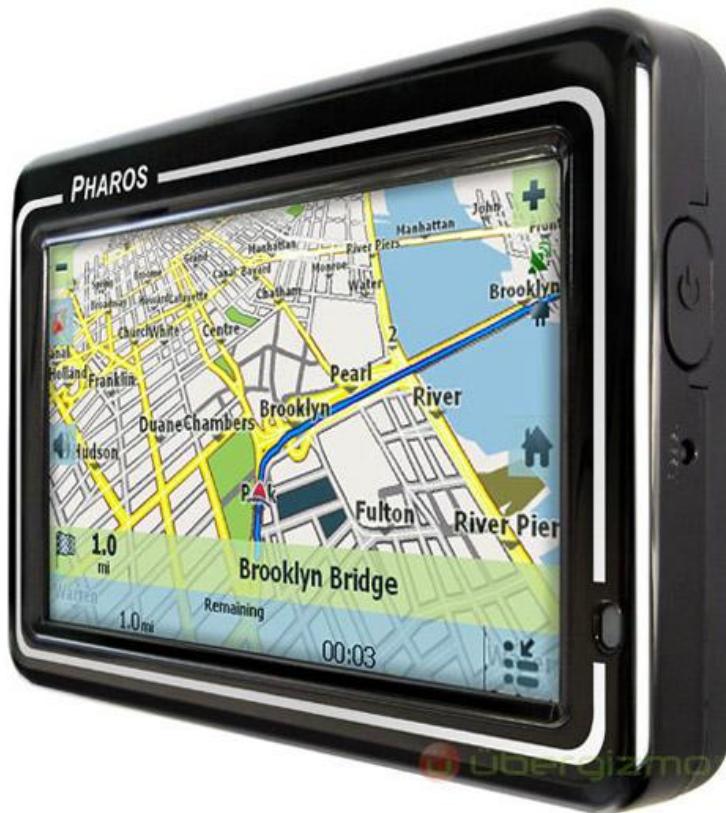


- **No 1** is MCU No.LPC2368 (100Pin LQFP).
 - **No.2** is 12MHz Crystal to be Time Base of MCU.
 - **No.3** is 32.768 KHz Crystal to be Time Base of RTC internal MCU.
 - **No.4** is 3V Battery for Backup of RTC.
 - **No.5** is JTAG ARM Connector for Real Time Debugging.
 - **No.6** is Power Supply Connector of board; it can be used with 7-12V AC/DC.
 - **No.7** is UART-0(RS232) Connector to use and Download Hex File into CPU.

- **No.8** is UART-2(RS232) Connector to use.
- **No.9** is Character LCD Connector; it can be used with +5V Supply LCD.
- **No.10** is VR to adjust the contrast or brightness of Character LCD.
- **No.11** is USB Connector to connect with USB Hub version 2.0.
- **No.12** is LED to display status of Power +VDD (+3V3).
- **No.13** is S1 that is ISP LOAD.
- **No.14** is S2 or RESET Switch.
- **No.15** is socket to insert Memory Card; it can be used with both SD Memory Card and MMC Memory Card.
- **No.16** is PS2 Connector to connect with PS2 keyboard.
- **No.17** is External Memory.
- **No.18** and **No.19** is jumper to connect External Memory to MCU.
- **No.20** is jumper to connect INT1.
- **No.21** and **No.22** is jumper to connect D- & D+ to the USB connector.

Jumper Settings for Interfaces:

<i>Jumper</i>	<i>State</i>	<i>Description</i>
BR10 – SCL	ON	Connects I2C SCL to EEPROM
BR11 – SDA	ON	Connects I2C SDA to EEPROM
BR5 – USB (D-)	ON	Connects USB Line D- to the USB connector
BR6 – USB (D+)	ON	Connects USB Line D+ to the USB connector
BR2 – Vbus	ON	Connects 5V USB supply voltage to the Vbus pin



GPS FUNDAMENTALS

What is GPS?

The Global Positioning System (GPS) is a worldwide radio-navigation system formed from a constellation of 24 satellites and their ground stations.

GPS uses these "man-made stars" as reference points to calculate positions accurate to a matter of meters. In fact, with advanced forms of GPS you can make measurements to better than a centimeter! In a sense it's like giving every square meter on the planet a unique address. GPS receivers have been miniaturized to just a few integrated circuits and so are becoming very economical. And that makes the technology accessible to virtually everyone. These days GPS is finding its way into cars, boats, planes, construction equipment, movie making gear, farm machinery, even laptop computers. Soon GPS will become almost as basic as the telephone. Indeed, at Trimble, we think it just may become a universal utility.

How GPS works?

Here's how GPS works in five logical steps:

- The basis of GPS is "triangulation" from satellites.
- To "triangulate," a GPS receiver measures distance using the travel time of radio signals.
- To measure travel time, GPS needs very accurate timing which it achieves with some tricks.
- Along with distance, you need to know exactly where the satellites are in space. High orbits and careful monitoring are the secret.
- Finally you must correct for any delays the signal experiences as it travels through the atmosphere.

Triangulating from Satellites

Improbable as it may seem, the whole idea behind GPS is to use satellites in space as reference points for locations here on earth.

That's right, by very, very accurately measuring our distance from three satellites we can "triangulate" our position anywhere on earth. Forget for a moment how our receiver measures this distance. We'll get to that later. First consider how distance measurements from three satellites can pinpoint you in space.

The Big Idea Geometrically:

Step One:

Suppose we measure our distance from a satellite and find it to be 11,000 miles. Knowing that we're 11,000 miles from a particular satellite narrows down all the possible locations we could be in the whole universe to the surface of a sphere that is centered on this satellite and has a radius of 11,000 miles.

Step Two:

Next, say we measure our distance to a second satellite and find out that it's 12,000 miles away.

That tells us that we're not only on the first sphere but we're also on a sphere that's 12,000 miles from the second satellite. Or in other words, we're somewhere on the circle where these two spheres intersected.

Step Three:

If we then make a measurement from a third satellite and find that we're 13,000 miles from that one, that narrows our position down even further, to the two points where the

13,000 mile sphere cuts through the circle that's the intersection of the first two spheres.

So by ranging from three satellites we can narrow our position to just two points in space. To decide which one is our true location we could make a fourth measurement. But usually one of the two points is a ridiculous answer (either too far from Earth or moving at an impossible velocity) and can be rejected without a measurement.

A fourth measurement does come in very handy for another reason however, but we'll tell you about that later.

Location

"Where am I?"

The first and most obvious application of GPS is the simple determination of a "position" or location. GPS is the first positioning system to offer highly precise location data for any point on the planet, in any weather. That alone would be enough to qualify it as a major utility, but the accuracy of GPS and the creativity of its users is pushing it into some surprising realms.

Knowing the precise location of something, or someone, is especially critical when the consequences of inaccurate data are measured in human terms. For example, when a stranded motorist was lost in a South Dakota blizzard for 2 days, GPS helped rescuers find her. GPS is also being applied in Italy to create exact location points for their nationwide geodetic network which will be used for surveying projects. Once in place it will support the first implementation of a nationally created location survey linked to the WGS-84 global grid.

Sometimes an exact reference locator is needed for extremely precise scientific work. Just getting to the world's tallest mountain was tricky, but GPS made measuring the growth of Mt. Everest easy. The data collected strengthened past work, but also revealed that as the Khumbu glacier moves toward Everest's Base Camp, the mountain itself is getting taller.

Tracking

If navigation is the process of getting something from one location to another, then tracking is the process of monitoring it as it moves along. Commerce relies on fleets of vehicles to deliver goods and services either across a crowded city or through nationwide corridors. So, effective fleet management has direct bottom-line implications, such as telling a customer when a package will arrive, spacing buses for the best scheduled service, directing the nearest ambulance to an accident, or helping tankers avoid hazards. GPS used in conjunction with communication links and computers can provide the backbone for systems tailored to applications in agriculture, mass transit, urban delivery, public safety, and vessel and vehicle tracking. So it's no surprise that police, ambulance, and fire departments are adopting systems like Trimble's GPS-based AVL (Automatic Vehicle Location) Manager to pinpoint both the location of the emergency and the location of the nearest response vehicle on a computer map. With this kind of clear visual picture of the situation, dispatchers can react immediately and confidently. Chicago developed a GPS tracking system to monitor emergency vehicles through their streets, saving precious time responding to 911 calls. And on the commercial front, two taxi companies in Australia track their cabs for better profit and improved safety.

GPS RECEIVER TECHNICAL SPECIFICATIONS



GPS RECEIVER TECHNICAL SPECIFICATIONS

The ultra-sensitive GPS receiver can acquire GPS signals from 32 channels of satellites and generate fast position fixes with high accuracy in extremely challenging environments and under poor signal conditions due to its active antenna and high sensitivity. The bi-directional NMEA 0183 v3.0 protocol offers industry standard data messages and a command set for easy interface to mapping software and embedded devices.

Features

- □High sensitivity -159dBm
- Searching up to 32 Channel of satellites
- Fast Position Fix with LED indication of status
- Low power consumption
- RTCM- in ready
- Built-inWAAS/EGNOS/MSAS Demodulator
- Supports NMEA0183 V 3.01 data protocol
- Real time navigation for location based services
- For Car Navigation, Marine Navigation, Fleet Management, AVL and Location-Based Services, Auto Pilot, Personal Navigation or touring devices, Tracking devices/systems
- Mapping devices application

Applications

- □Automotive and Marine Navigation
- □Automotive Navigator Tracking
- □Emergency Locator
- □Geographic Surveying
- □Personal Positioning
- □Sporting and Recreation
- □Embedded applications

NMEA Protocol

This section provides a brief overview of the NMEA 0183 protocol, and describes both the standard and optional messages offered by the GPS Receiver. NMEA 0183 is a simple, yet comprehensive ASCII protocol which defines both the communication interface and the data format. The NMEA 0183 protocol was originally established to allow marine navigation equipment to share information. Since it is a well established industry standard, NMEA 0183 has also gained popularity for use in applications other than marine electronics. The GPS receiver supports the latest release of NMEA 0183, Version 3.0 (July 1, 2000). The primary change in release 3.0 is the addition of the mode indicators in the GLL, RMC, and VTG messages. For those applications requiring output only from the GPS receiver, the standard NMEA 0183 sentences are a popular choice. Many standard application packages support the standard NMEA output messages. The standard NMEA output only messages are: GGA, GLL, GSA, GSV, RMC, VTC, and ZDA.

EEPROM

In the design of all microprocessors-based systems, semiconductor memories are used as primary storage for code and data. Semiconductor memories are connected directly to the CPU and they are the memory that the CPU first asks for information (code and data). For this reason, semiconductor memories are sometimes referred to as primary memory.

Important Terminology common to all Semiconductor Memories:

Memory capacity:

The number of bits that a semiconductor memory chip can store is called chip capacity. It can be in units of Kilobits, Megabits and so on. This must be distinguished from the storage capacity of computer system. While the memory capacity of a memory IC chip is always given in bits, the memory capacity of a computer system is given in bytes.

Memory organization:

Memory chips are organized into a number of locations within the IC. Each location can hold 1 bit, 4 bits, 8 bits or even 16 bits, depending on how it is designed internally. The number of bits that each location within the memory chip can hold is always equal to the number of data pins on the chip. i.e., the total number of bits that a memory chip can store is equal to the number of locations times the number of data bits per location.

Speed:

One of the most important characteristics of a memory chip is the speed at which its data can be accessed. The speed of the memory chip is commonly referred to as its access time. The access time of memory chip varies from a few nanoseconds to hundreds of nanoseconds, depending on the IC technology used in the design and fabrication process.

The different types of memories are RAM, ROM, EPROM and EEPROM. RAM and ROM are inbuilt in the microprocessor. This project requires the data such as the total number of available units and the pulse count to be stored permanently and this data modifies upon the power

consumption. Thus this data has to be stored in such a location where it cannot be erased when power fails and also the data should be allowed to make changes in it without the system interface i.e., there should be a provision in such a way that the data should be accessed (or modified) while it is in system board but not external erasure and programming. The flash memory inbuilt in the microcontroller can erase the entire contents in less than a second and the erasure method is electrical. But the major drawback of Flash memory is that when flash memory's contents are erased, the entire device will be erased but not a desired section or byte.

For this purpose, we prefer EEPROM in our project.

EEPROM (Electrically Erasable Programmable Read only memory)

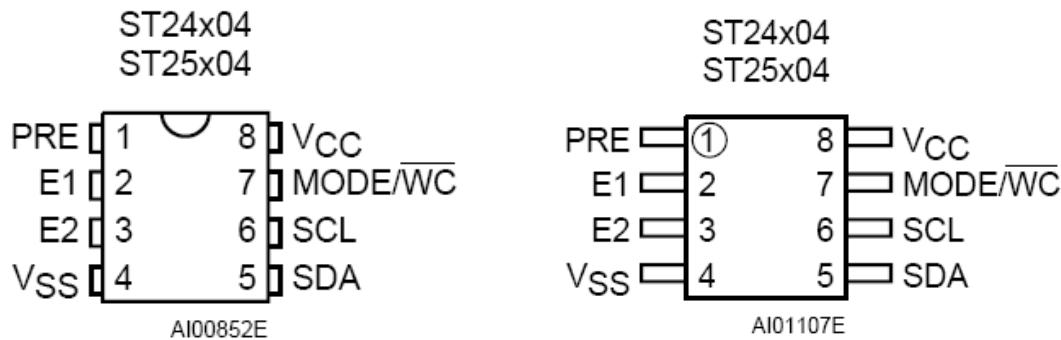
EEPROM has several advantages over other memory devices, such as the fact that its method of erasure is electrical and therefore instant. In addition, in EEPROM one can select which byte to be erased, in contrast to flash, in which the entire contents of ROM are erased. The main advantage of EEPROM is that one can program and erase its contents while it is in system board. It does not require physical removal of the memory chip from its socket. In general, the cost per bit for EEPROM is much higher when compared to other devices.

The EEPROM used in this project is 24C04 type.

Features of 24C04 EEPROM:

- 1 million erase/write cycles with 40 years data retention.
- Single supply voltage:
3v to 5.5v for st24x04 versions.
2.5v to 5.5v for st25x04 versions.
- Hardware write control versions:
st24w04 and st25w04.
- Programmable write protection.
- Two wire serial interface, fully i2c bus compatible.
- Byte and multibyte write (up to 4 bytes).

- Page write (up to 8 bytes).
- Byte, random and sequential read modes
- Self timed programming cycle
- Automatic address incrementing
- Enhanced ESD/Latch up performances



DIP Pin Connections

SO Pin Connection

PRE	Write Protect Enable
E1-E2	Chip Enable Inputs
SDA	Serial Data Address Input/Output
SCL	Serial Clock
MODE	Multibyte/Page Write Mode (C version)
WC	Write Control (W version)
V _{CC}	Supply Voltage
V _{SS}	Ground

Fig: Signal Names

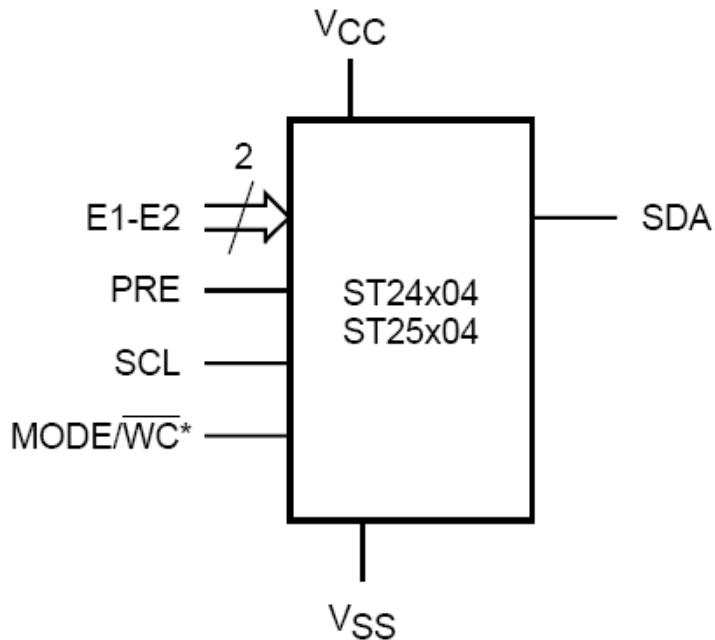


Fig: Logic Diagram

DESCRIPTION

The 24C04 is a 4 Kbit electrically erasable programmable memory (EEPROM), organized as 2 blocks of 256 x 8 bits. They are manufactured in ST Microelectronics' Hi-Endurance Advanced CMOS technology which guarantees an endurance of one million erase/write cycles with a data retention of 40 years. Both Plastic Dual-in-Line and Plastic Small Outline packages are available. The memories are compatible with the I²C standard, two wire serial interface which uses a bi-directional data bus and serial clock. The memories carry a built-in 4 bit, unique device identification code (1010) corresponding to the I²C bus definition. This is used together with 2 chip enable inputs (E2, E1) so that up to 4 x 4K devices may be attached to the I²C bus and selected individually. The memories behave as a slave device in the I²C protocol with all memory

operations synchronized by the serial clock. Read and write operations are initiated by a START condition generated by the bus master. The START condition is followed by a stream of 7 bits (identification code 1010), plus one read/write bit and terminated by an acknowledge bit.

	Device Code				Chip Enable		Block Select	\overline{RW}
Bit	b7	b6	b5	b4	b3	b2	b1	b0
Device Select	1	0	1	0	E2	E1	A8	\overline{RW}

Note: The MSB b7 is sent first.

Table: Device Select Mode

Mode	\overline{RW} bit	MODE	Bytes	Initial Sequence
Current Address Read	'1'	X	1	START, Device Select, $\overline{RW} = '1'$
Random Address Read	'0'	X	1	START, Device Select, $\overline{RW} = '0'$, Address,
	'1'			reSTART, Device Select, $\overline{RW} = '1'$
Sequential Read	'1'	X	1 to 512	Similar to Current or Random Mode
Byte Write	'0'	X	1	START, Device Select, $\overline{RW} = '0'$
Multibyte Write ⁽²⁾	'0'	V_{IH}	4	START, Device Select, $\overline{RW} = '0'$
Page Write	'0'	V_{IL}	8	START, Device Select, $\overline{RW} = '0'$

Notes: 1. X = V_{IH} or V_{IL}

2. Multibyte Write not available in ST24/25W04 versions.

Table: Operating Modes

When writing data to the memory it responds to the 8 bits received by asserting an acknowledge bit during the 9th bit time. When data is read by the bus master, it acknowledges the receipt of the data bytes in the same way. Data transfers are terminated with a STOP condition.

Power On Reset: VCC lock out write protect.

In order to prevent data corruption and inadvertent write operations during power up, a Power On Reset (POR) circuit is implemented. Until the VCC voltage has reached the POR threshold value, the internal reset is active, all operations are disabled and the device will not respond to any command. In the same way, when VCC drops down from the operating voltage to below the POR threshold value, all operations are disabled and the device will not respond to any command. A stable VCC must be applied before applying any logic signal.

SIGNAL DESCRIPTIONS

Serial Clock (SCL).

The SCL input pin is used to synchronize all data in and out of the memory. A resistor can be connected from the SCL line to VCC to act as a pull up.

Serial Data (SDA).

The SDA pin is bi-directional and is used to transfer data in or out of the memory. It is an open drain output that may be wire-OR'ed with other open drain or open collector signals on the bus. A resistor must be connected from the SDA bus line to VCC to act as pull up.

Chip Enable (E1 - E2).

These chip enable inputs are used to set the 2 least significant bits (b2, b3) of the 7 bit device select code. These inputs may be driven dynamically or tied to VCC or VSS to establish the device select code.

Protect Enable (PRE).

The PRE input pin, in addition to the status of the Block Address Pointer bit (b2, location 1FFh as in below figure), sets the PRE write protection active.

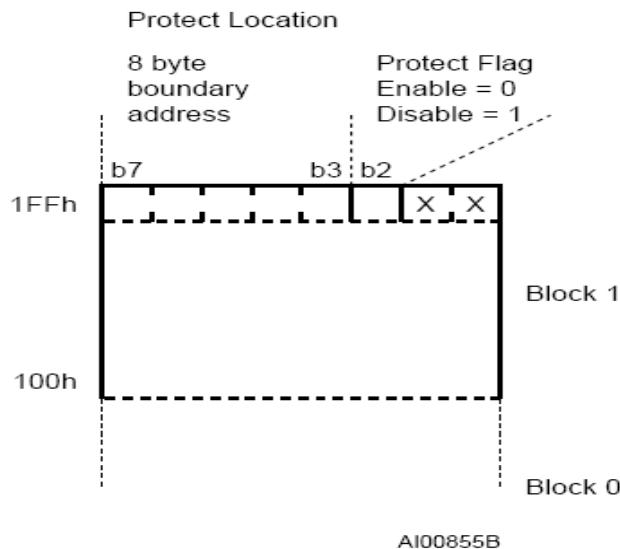


Fig: Memory Protection

Mode (MODE).

The MODE input is available on pin 7 and may be driven dynamically. It must be at VIL or VIH for the Byte Write mode, VIH for Multi byte Write mode or VIL for Page Write mode. When unconnected, the MODE input is internally read as VIH (Multi byte Write mode).

Write Control (WC).

An hardware Write Control feature (WC) is offered only for ST24W04 and ST25W04 versions on pin 7. This feature is useful to protect the contents of the memory from any erroneous

erase/write cycle. The Write Control signal is used to enable (WC = VIH) or disable (WC =VIL) the internal write protection. When unconnected, the WC input is internally read as VIL and the memory area is not write protected.

DEVICE OPERATION

I2C Bus Background

The ST24/25x04 supports the I2C protocol. This protocol defines any device that sends data onto the bus as a transmitter and any device that reads the data as a receiver. The device that controls the data transfer is known as the master and the other as the slave. The master will always initiate a data transfer and will provide the serial clock for synchronization.

The ST24/25x04 is always slave devices in all communications.

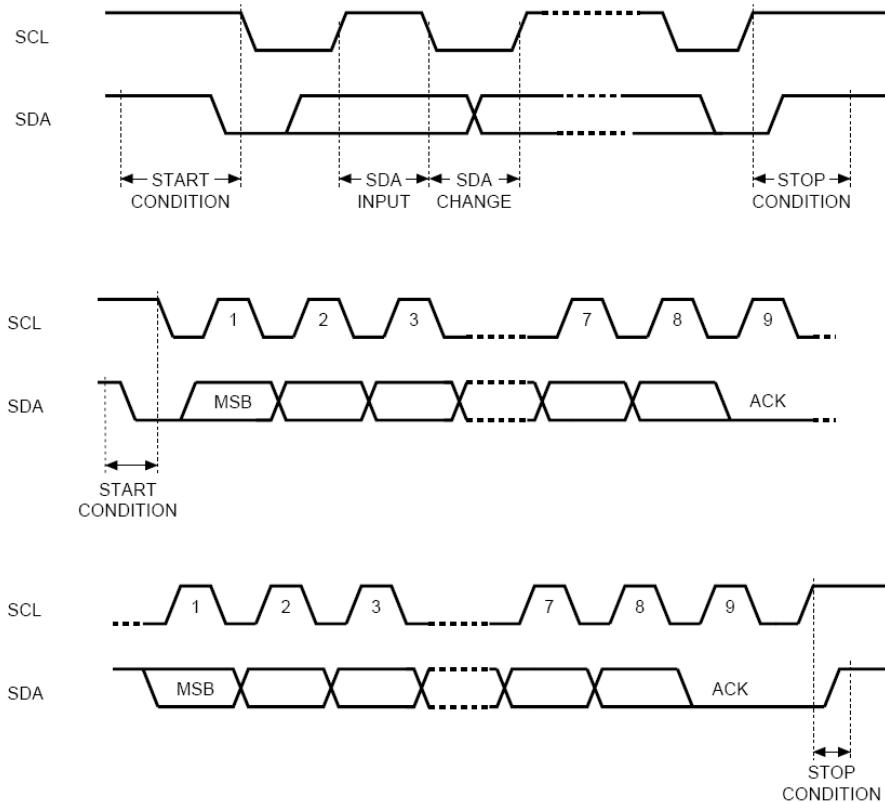


Fig: I2C Protocol

Start Condition.

START is identified by a high to low transition of the SDA line while the clock SCL is stable in the high state. A START condition must precede any command for data transfer. Except during a programming cycle, the ST24/25x04 continuously monitor the SDA and SCL signals for a START condition and will not respond unless one is given.

Stop Condition.

STOP is identified by a low to high transition of the SDA line while the clock SCL is stable in the high state. A STOP condition terminates communication between the ST24/25x04 and the bus master. A STOP condition at the end of a Read command, after and only after a No Acknowledge, forces the standby state. A STOP condition at the end of a Write command triggers the internal EEPROM write cycle.

Acknowledge Bit (ACK).

An acknowledge signal is used to indicate a successful data transfer. The bus transmitter, either master or slave, will release the SDA bus after sending 8 bits of data. During the 9th clock pulse period the receiver pulls the SDA bus low to acknowledge the receipt of the 8 bits of data.

Data Input.

During data input the ST24/25x04 sample the SDA bus signal on the rising edge of the clock SCL. Note that for correct device operation the SDA signal must be stable during the clock low to high transition and the data must change ONLY when the SCL line is low.

Memory Addressing.

To start communication between the bus master and the slave ST24/25x04, the master must initiate a START condition. Following this, the master sends onto the SDA bus line 8 bits (MSB first) corresponding to the device select code (7 bits) and a READ or WRITE bit. The 4 most significant bits of the device select code are the device type identifier, corresponding to the I2C bus definition. For these memories the 4 bits are fixed as 1010b. The following 2 bits identify the specific memory on the bus. They are matched to the chip enable signals E2, E1. Thus up to 4 x 4K memories can be connected on the same bus giving a memory capacity total of 16 Kilobits. After a START condition any memory on the bus will identify the device code and compare the following 2 bits to its chip enable inputs E2, E1. The 7th bit sent is the block number (one block = 256 bytes). The 8th bit sent is the read or write bit (RW), this bit is set to '1' for read and '0' for write operations. If a match is found, the corresponding memory will acknowledge the identification on the SDA bus during the 9th bit time.

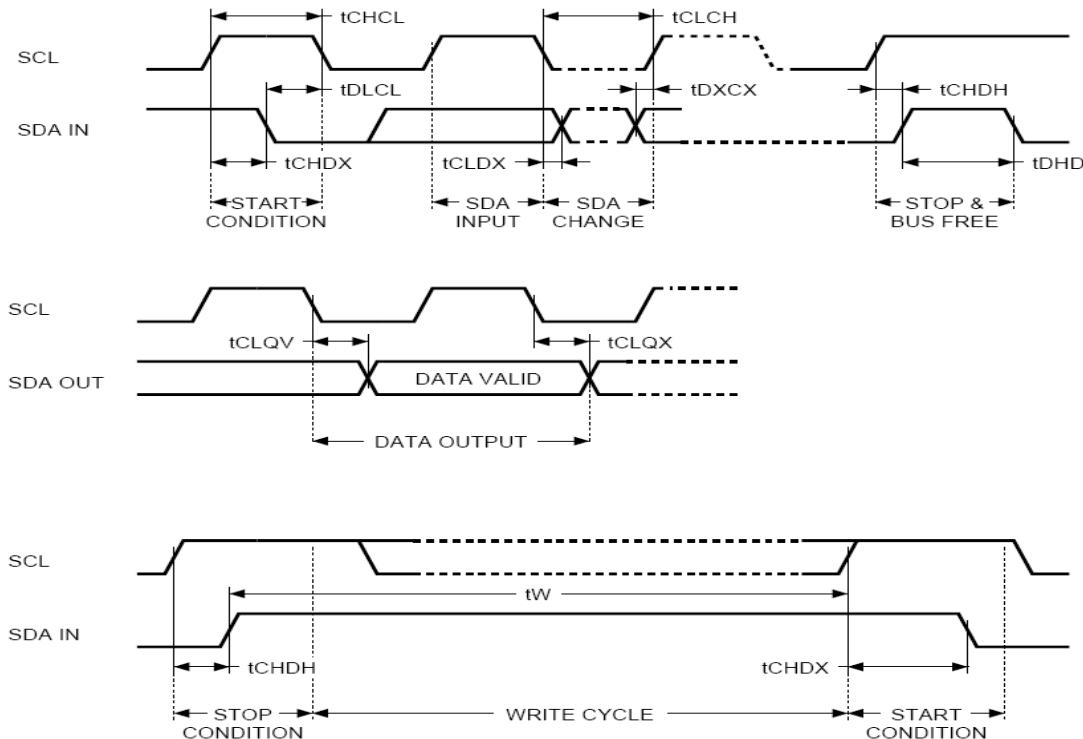


Fig: AC Waveforms

Write Operations

The Multi byte Write mode (only available on the ST24/25C04 versions) is selected when the MODE pin is at VIH and the Page Write mode when MODE pin is at VIL. The MODE pin may be driven dynamically with CMOS input levels. Following a START condition the master sends a device select code with the RW bit reset to '0'. The memory acknowledges this and waits for a byte address. The byte address of 8 bits provides access to one block of 256 bytes of the memory. After receipt of the byte address the device again responds with an acknowledge. For the ST24/25W04 versions, any write command with WC = 1 will not modify the memory content.

Byte Write.

In the Byte Write mode the master sends one data byte, which is acknowledged by the memory. The master then terminates the transfer by generating a STOP condition. The Write mode is independent of the state of the MODE pin which could be left floating if only this mode was to be used. However it is not a recommended operating mode, as this pin has to be connected to either VIH or VIL, to minimize the stand-by current.

Multi byte Write.

For the Multi byte Write mode, the MODE pin must be at VIH. The Multi byte Write mode can be started from any address in the memory. The master sends from one up to 4 bytes of data, which are each acknowledged by the memory. The transfer is terminated by the master generating a STOP condition. The duration of the write cycle is $T_w = 10\text{ms}$ maximum except when bytes are accessed on 2 rows (that is have different values for the 6 most significant address bits A7-A2), the programming time is then doubled to a maximum of 20ms. Writing more than 4 bytes in the Multi byte Write mode may modify data bytes in an adjacent row (one row is 8 bytes long). However, the Multi byte Write can properly write up to 8 consecutive bytes

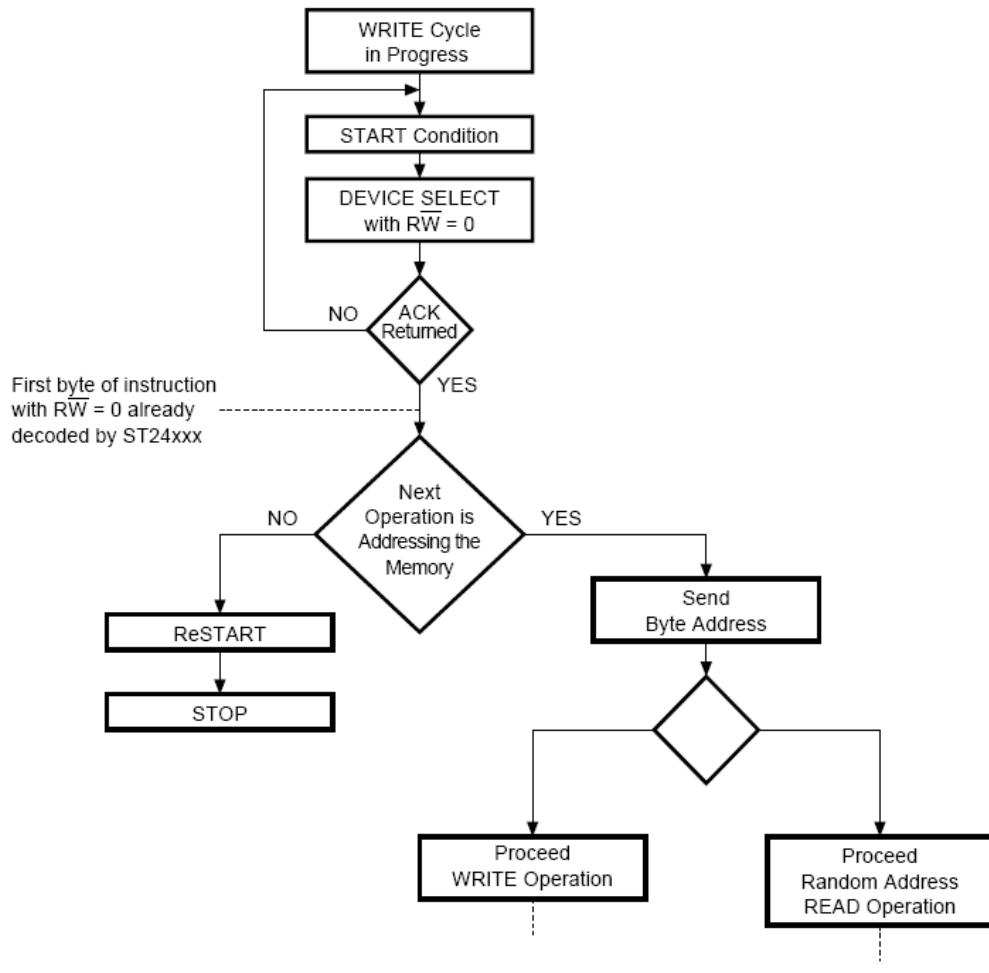
as soon as the first address of these 8 bytes is the first address of the row, the 7 following bytes being written in the 7 following bytes of this same row.

Page Write.

For the Page Write mode, the MODE pin must be at VIL. The Page Write mode allows up to 8 bytes to be written in a single write cycle, provided that they are all located in the same 'row' in the memory: that is the 5 most significant memory address bits (A7-A3) are the same inside one block. The master sends from one up to 8 bytes of data, which are each acknowledged by the memory. After each byte is transferred, the internal byte address counter (3 least significant bits only) is incremented. The transfer is terminated by the master generating a STOP condition. Care must be taken to avoid address counter 'roll-over' which could result in data being overwritten. Note that, for any write mode, the generation by the master of the STOP condition starts the internal memory program cycle. All inputs are disabled until the completion of this cycle and the memory will not respond to any request.

Minimizing System Delays by Polling on ACK.

During the internal write cycle, the memory disconnects itself from the bus in order to copy the data from the internal latches to the memory cells. The maximum value of the write time (T_w) is given from the AC Characteristics, since the typical time is shorter, the time seen by the system may be reduced by an ACK polling sequence issued by the master.



AI01099B

Fig: Write Cycle Polling using ACK

Data in the upper block of 256 bytes of the memory may be write protected. The memory is write protected between a boundary address and the top of memory (address 1FFh) when the PRE input pin is taken high and when the Protect Flag (bit b2 in location 1FFh) is set to '0'. The boundary address is user defined by writing it in the Block Address Pointer. The Block Address Pointer is an 8 bit EEPROM register located at the address 1FFh. It is composed by 5 MSBs Address Pointer, which defines the bottom boundary address and 3 LSBs which must be programmed at '0'. This Address Pointer can therefore address a boundary in steps of 8 bytes.

The sequence to use the Write Protected feature is:

- write the data to be protected into the top of the memory, up to, but not including, location 1FFh;
- set the protection by writing the correct bottom boundary address in the Address Pointer (5 MSBs of location 1FFh) with bit b2 (Protect flag) set to '0'. Note that for a correct functionality of the memory, all the 3 LSBs of the Block Address Pointer must also be programmed at '0'. The area will now be protected when the PRE input pin is taken High. While the PRE input pin is read at '0' by the memory, the location 1FFh can be used as a normal EEPROM byte.

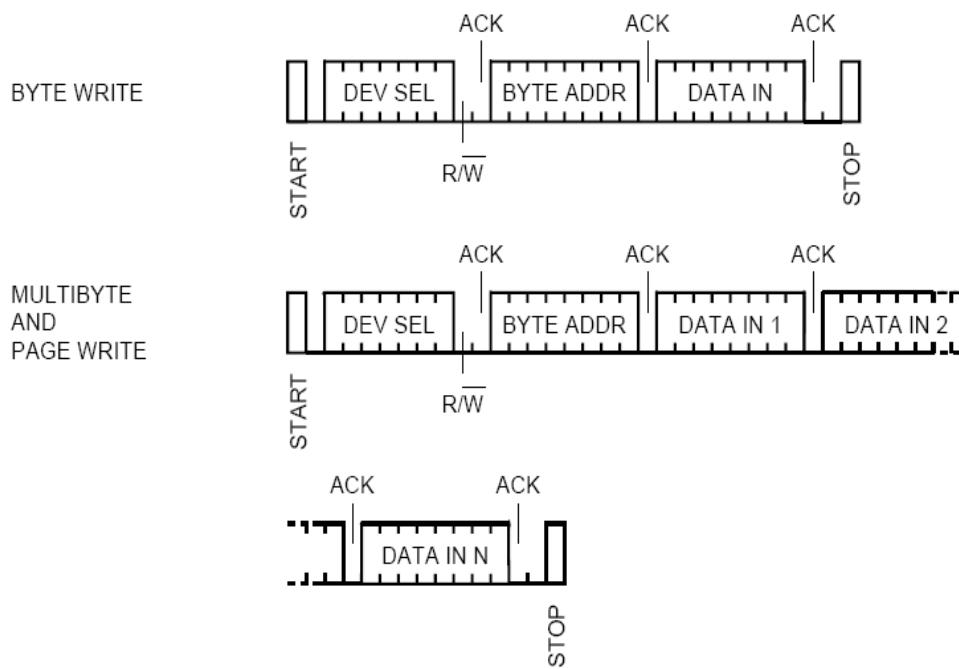


Fig: Write Modes Sequence

Read Operations

Read operations are independent of the state of the MODE pin. On delivery, the memory content is set at all "1's" (or FFh).

Current Address Read.

The memory has an internal byte address counter. Each time a byte is read, this counter is incremented. For the Current Address Read mode, following a START condition, the master sends a memory address with the RW bit set to '1'. The memory acknowledges this and outputs the byte addressed by the internal byte address counter. This counter is then incremented. The master does NOT acknowledge the byte output, but terminates the transfer with a STOP condition.

Random Address Read.

A dummy write is performed to load the address into the address counter. This is followed by another START condition from the master and the byte address is repeated with the RW bit set to '1'. The memory acknowledges this and outputs the byte addressed. The master has to NOT acknowledge the byte output, but terminates the transfer with a STOP condition.

Sequential Read.

This mode can be initiated with either a Current Address Read or a Random Address Read. However, in this case the master DOES acknowledge the data byte output and the memory continues to output the next byte in sequence. To terminate the stream of bytes, the master must NOT acknowledge the last byte output, but MUST generate a STOP condition. The output data is from consecutive byte addresses, with the internal byte address counter automatically incremented after each byte output. After a count of the last memory address, the address counter will 'roll- over' and the memory will continue to output data.

Acknowledge in Read Mode.

In all read modes the ST24/25x04 wait for an acknowledge during the 9th bit time. If the master does not pull the SDA line low during this time, the ST24/25x04 terminate the data transfer and switches to a standby state.

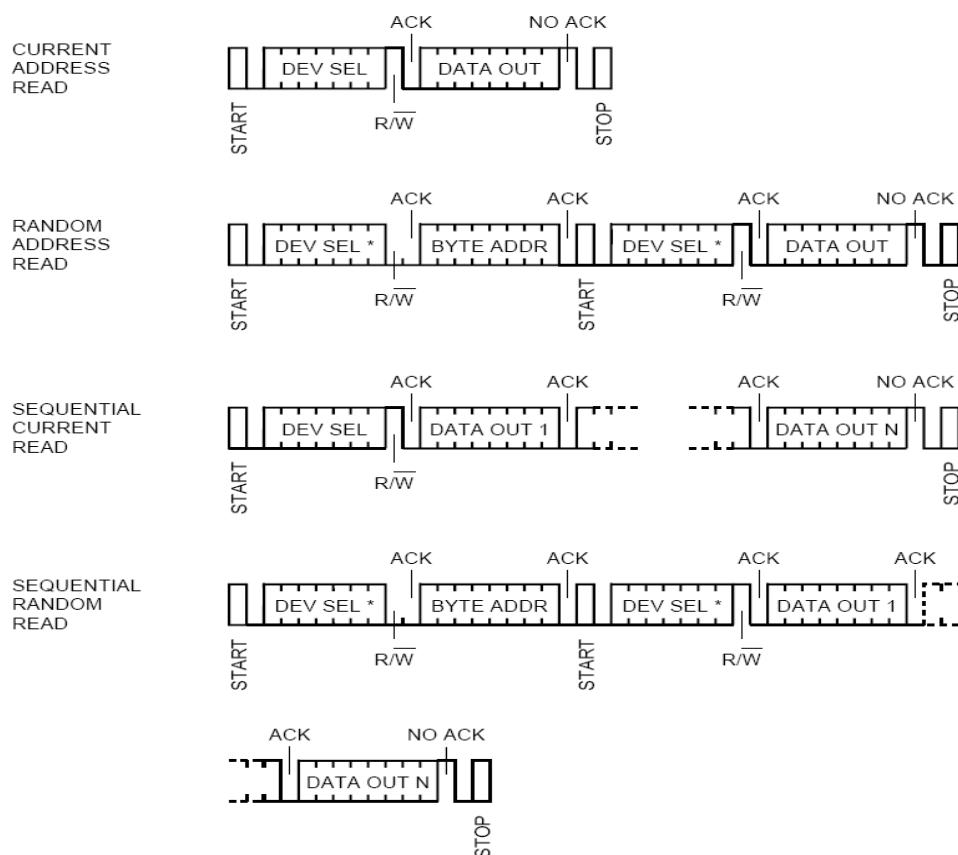
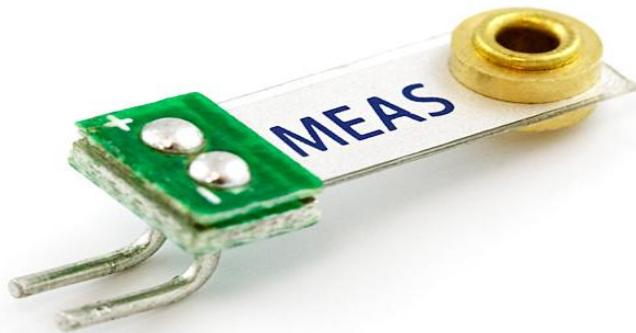


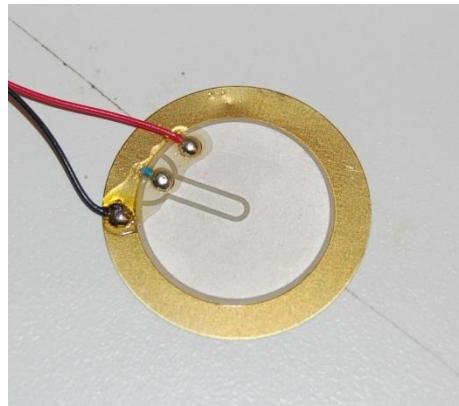
Fig: Read Modes Sequence

Piezoelectric sensor/vibration sensor

- A piezoelectric sensor is a device that uses the [piezoelectric effect](#), to measure changes in [pressure](#), [acceleration](#), [strain](#) or [force](#) by converting them to an [electrical](#) charge. The prefix piezo- is Greek for 'press' or 'squeeze'.



Piezo Electric Sensor



A piezoelectric sensor is a device that uses the piezoelectric effect to measure pressure, acceleration, strain or force by converting them to an electrical signal. Piezoelectric sensors have proven to be versatile tools for the measurement of various processes. They are used for quality assurance, process control and for research and development in many different industries. From the Curie's initial discovery in 1880, it took until the 1950s before the piezoelectric effect was used for industrial sensing applications. Since then, the utilization of this measuring principle has experienced a constant growth and can be regarded as a mature technology with an outstanding inherent reliability. It has been successfully used in various applications as for example in medical, aerospace, nuclear instrumentation and in mobile's touch key pad as pressure sensor.

In the automotive industry piezoelectric elements are used as the standard devices for engine indicating in developing internal combustion engines. The combustion processes are measured with piezoelectric sensors. The sensors are either directly mounted into additional holes into the cylinder head or the spark/glow plug is equipped with a built in miniature piezoelectric sensor.

The rise of piezoelectric technology is directly related to a set of inherent advantages. The high modulus of elasticity of many piezoelectric materials is comparable to that of many metals and goes up to 105 N/m^2 . Even though piezoelectric sensors are electromechanical systems that react on compression, the sensing elements show almost zero deflection. This is the reason why piezoelectric sensors are so rugged, have an extremely high natural frequency and an excellent linearity over a wide amplitude range. Additionally, piezoelectric technology is insensitive to electromagnetic fields and radiation, enabling measurements under harsh conditions. Some materials used (especially gallium phosphate or tourmaline) have an extreme stability over temperature enabling sensors to have a working range of up to 1000°C . Tourmaline shows pyroelectricity in addition to the piezoelectric effect; this is the ability to generate an electrical signal when the temperature of the crystal changes. This effect is also common to piezoceramic materials.

One disadvantage of piezoelectric sensors is that they cannot be used for true static measurements. A static force will result in a fixed amount of charges on the piezoelectric

material. While working with conventional readout electronics, imperfect insulating materials, and reduction in internal sensor resistance will result in a constant loss of electrons, and yield a decreasing signal. Elevated temperatures cause an additional drop in internal resistance and sensitivity. The main effect on the piezoelectric effect is that with increasing pressure loads and temperature the sensitivity is reduced due to twin-formation. While quartz sensors need to be cooled during measurements at temperatures above 300°C special types of crystals like GaPo₄ gallium phosphate do not show any twin formation up to the melting point of the material itself.

Anyhow, it would be a misconception that piezoelectric sensors can only be used for very fast processes or at ambient conditions. In fact, there are numerous applications that show quasi-static measurements while there are other applications that go to temperatures far beyond 500°C.

Piezoelectric sensors are also seen in nature. Dry bone is piezoelectric, and is thought by some to act as a biological force sensor.

RF COMMUNICATION

WHAT IS RF?

Radio frequency (RF) is a frequency or rate of oscillation within the range of about 3 Hz to 300 GHz. This range corresponds to frequency of alternating current electrical signals used to produce and detect radio waves. Since most of this range is beyond the vibration rate that most mechanical systems can respond to, RF usually refers to oscillations in electrical circuits or electromagnetic radiation

.

PROPERTIES OF RF:

Electrical currents that oscillate at RF have special properties not shared by direct current signals. One such property is the ease with which it can ionize air to create a conductive path through air. This property is exploited by 'high frequency' units used in electric arc welding. Another special property is an electromagnetic force that drives the RF current to the surface of conductors, known as the skin effect. Another property is the ability to appear to flow through paths that contain insulating material, like the dielectric insulator of a capacitor. The degree of effect of these properties depends on the frequency of the signals.

DIFFERENT RANGES PRESENT IN RF AND APPLICATIONS IN THEIR RANGES?

Extremely low frequency

ELF 3 to 30 Hz

10,000 km to 100,000 km directly audible when converted to sound, communication with submarines

Super low frequency

SLF 30 to 300 Hz

1,000 km to 10,000 km

directly audible when converted to sound, AC power grids (50 hertz and 60 hertz)

Ultra low frequency

ULF 300 to 3000 Hz

100 km to 1,000 km

directly audible when converted to sound, communication with mines

Very low frequency

VLF 3 to 30 kHz

10 km to 100 km

directly audible when converted to sound (below ca. 18-20 kHz; or "ultrasound" 20-30+ kHz)

Low frequency

LF 30 to 300 kHz

1 km to 10 km

AM broadcasting, navigational beacons, lowFER

Medium frequency

MF 300 to 3000 kHz

100 m to 1 km

navigational beacons, AM broadcasting, maritime and aviation communication

High frequency

HF 3 to 30 MHz

10 m to 100 m

Shortwave, amateur radio, citizens' band radio

Very high frequency

VHF 30 to 300 MHz

1 m to 10 m

FM broadcasting broadcast television, aviation, GPR

Ultra high frequency

UHF 300 to 3000 MHz

10 cm to 100 cm

Broadcast television, mobile telephones, cordless telephones, wireless networking, remote keyless entry for automobiles, microwave ovens, GPR

Super high frequency

SHF 3 to 30 GHz

1 cm to 10 cm

Wireless networking, satellite links, microwave links, Satellite television, door openers.

Extremely high frequency

EHF 30 to 300 GHz

1 mm to 10 mm

Microwave data links, radio astronomy, remote sensing, advanced weapons systems, advanced security scanning

BRIEF DESCRIPTION OF RF:

Radio frequency (abbreviated RF) is a term that refers to alternating current (AC) having characteristics such that, if the current is input to an antenna, an electromagnetic (EM) field is generated suitable for wireless broadcasting and/or communications. These frequencies cover a significant portion of the electromagnetic radiation spectrum, extending from nine kilohertz (9 kHz), the lowest allocated wireless communications frequency (it's within the range of human hearing), to thousands of gigahertz(GHz).

When an RF current is supplied to an antenna, it gives rise to an electromagnetic field that propagates through space. This field is sometimes called an RF field; in less technical jargon it is a "radio wave." Any RF field has a wavelength that is inversely proportional to the frequency. In the atmosphere or in outer space, if f is the frequency in megahertz and λ is the wavelength in meters, then

$$\lambda = 300/f$$

The frequency of an RF signal is inversely proportional to the wavelength

of the EM field to which it corresponds. At 9 kHz, the free-space wavelength is approximately 33 kilometers (km) or 21 miles (mi). At the highest radio frequencies, the EM wavelengths measure approximately one millimeter (1 mm). As the frequency is increased beyond that of the RF spectrum, EM energy takes the form of infrared (IR), visible, ultraviolet (UV), X rays, and gamma rays.

Many types of wireless devices make use of RF fields. Cordless and cellular telephone, radio and television broadcast stations, satellite communications systems, and two-way radio services all operate in the RF spectrum. Some wireless devices operate at IR or visible-light frequencies, whose electromagnetic wavelengths are shorter than those of RF fields. Examples include most television-set remote-control boxes, Some cordless computer keyboards and mice, and a few wireless hi-fi stereo headsets.

The RF spectrum is divided into several ranges, or bands. With the exception of the lowest-frequency segment, each band represents an increase of frequency corresponding

to an order of magnitude (power of 10). The table depicts the eight bands in the RF spectrum, showing frequency and bandwidth ranges. The SHF and EHF bands are often referred to as the **microwave spectrum**.

WHY DO WE GO FOR RF COMMUNICATION?

RF Advantages:

1. No line of sight is needed.
2. Not blocked by common materials: It can penetrate most solids and pass through walls.
3. Longer range.
4. It is not sensitive to the light;
5. It is not much sensitive to the environmental changes and weather conditions.

WHAT CARE SHOULD BE TAKEN IN RF COMMUNICATION?

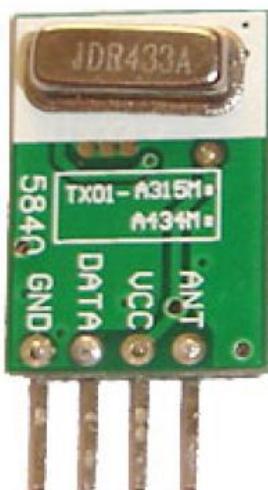
RF Disadvantages:

1. Interference: communication devices using similar frequencies - wireless phones, scanners, wrist radios and personal locators can interfere with transmission
2. Lack of security: easier to "eavesdrop" on transmissions since signals are spread out in space rather than confined to a wire
3. Higher cost than infrared
4. Federal Communications Commission(FCC) licenses required for some products
5. Lower speed: data rate transmission is lower than wired and infrared transmission

WHAT ARE THE MAIN REQUIREMENTS FOR THE COMMUNICATION USING RF?

- RF Transmitter
- RF Receiver
- Encoder and Decoder

RF TRANSMITTER STT-433MHz:



STT-433 MHz TRANSMITTER

FACTORS INFLUENCED TO CHOOSE STT-433MHz

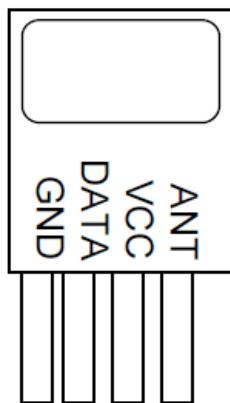
ABOUT THE TRANSMITTER:

- The STT-433 is ideal for remote control applications where low cost and longer range is required.
- The transmitter operates from a 1.5-12V supply, making it ideal for battery-powered applications.
- The transmitter employs a SAW-stabilized oscillator, ensuring accurate frequency control for best range performance.
- The manufacturing-friendly SIP style package and low-cost make the STT-433 suitable for high volume applications.

Features

- 433.92 MHz Frequency
- Low Cost
- 1.5-12V operation
- Small size

PIN DESCRIPTION:



GND

Transmitter ground. Connect to ground plane

DATA

Digital data input. This input is CMOS compatible and should be driven with CMOS level inputs.

VCC

Operating voltage for the transmitter. VCC should be bypassed with a .01uF ceramic capacitor and filtered with a 4.7uF tantalum capacitor. Noise on the power supply will degrade transmitter noise performance.

ANT

50 ohm antenna output. The antenna port impedance affects output power and harmonic emissions. Antenna can be single core wire of approximately 17cm length or PCB trace antenna.

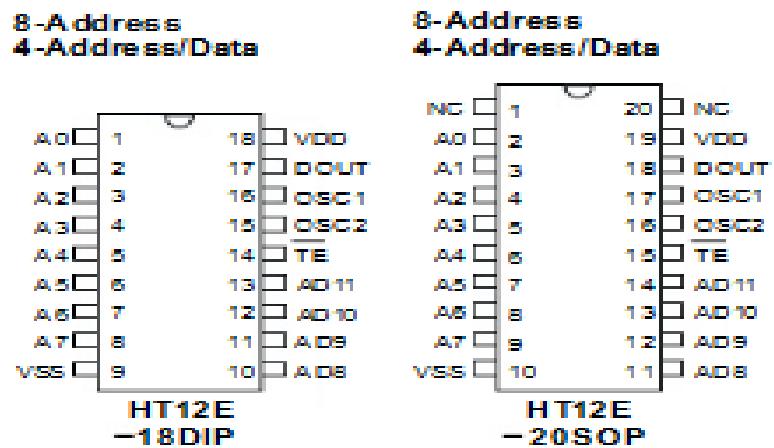
APPLICATION:

The typical connection shown in the above figure cannot work exactly at all times because there will be no proper synchronization between the transmitter and the microcontroller unit. i.e., whatever the microcontroller sends the data to the transmitter, the transmitter is not able to accept this data as this will be not in the radio frequency range. Thus, we need an intermediate device which can accept the input from the microcontroller, process it in the range of radio frequency range and then send it to the transmitter. Thus, an encoder is used.

The encoder used here is **HT12E** from **HOLTEK SEMICONDUCTORS INC.**

The HT 12E Encoder ICs are series of CMOS LSIs for Remote Control system applications. They are capable of Encoding 12 bit of information which consists of N address bits and 12-N data bits. Each address/data input is externally trinary programmable if bonded out.

ENCODER HT12E:



PIN DESCRIPTION:

Pin Description

Pin Name	I/O	Internal Connection	Description
A0-A7	I	CMOS IN Pull-high (HT12A) NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	Input pins for address A0-A7 setting These pins can be externally set to VSS or left open
AD8-AD11	I	NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	Input pins for address/data AD8-AD11 setting These pins can be externally set to VSS or left open
D8-D11	I	CMOS IN Pull-high	Input pins for data D8-D11 setting and transmission enable, active low These pins should be externally set to VSS or left open (see Note)
DOUT	O	CMOS OUT	Encoder data serial transmission output
L/MB	I	CMOS IN Pull-high	Latch/Momentary transmission format selection pin: Latch: Floating or VDD Momentary: VSS

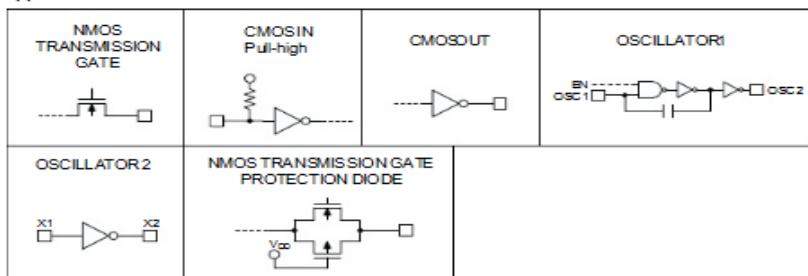
PIN DESCRIPTION:

Pin Name	I/O	Internal Connection	Description
TE	I	CMOS IN Pull-high	Transmission enable, active low (see Note)
OSC1	I	OSCILLATOR 1	Oscillator input pin
OSC2	O	OSCILLATOR 1	Oscillator output pin
X1	I	OSCILLATOR 2	455kHz resonator oscillator input
X2	O	OSCILLATOR 2	455kHz resonator oscillator output
VSS	I	—	Negative power supply, grounds
VDD	I	—	Positive power supply

Note: D8-D11 are all data input and transmission enable pins of the HT12A.

TE is a transmission enable pin of the HT12E.

Approximate internal connections



Absolute Maximum Ratings

Supply Voltage (HT12A)	-0.3V to 5.5V	Supply Voltage (HT12E)	-0.3V to 13V
Input Voltage.....	V_{SS} =-0.3 to V_{DD} +0.3V	Storage Temperature.....	-50°C to 125°C
Operating Temperature.....	-20°C to 75°C		

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

HT12E

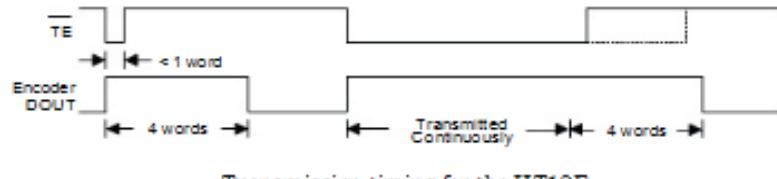
Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{DD}	Operating Voltage	—	—	2.4	5	12	V
I_{STB}	Standby Current	3V	Oscillator stops	—	0.1	1	μ A
		12V		—	2	4	μ A
I_{DD}	Operating Current	3V	No load $f_{osc}=3\text{kHz}$	—	40	80	μ A
		12V		—	150	300	μ A
I_{DOUT}	Output Drive Current	5V	$V_{OH}=0.9V_{DD}$ (Source)	-1	-1.6	—	mA
			$V_{OL}=0.1V_{DD}$ (Sink)	1	1.6	—	mA
V_{IH}	"H" Input Voltage	—	—	0.8 V_{DD}	—	V_{DD}	V
V_{IL}	"L" Input Voltage	—	—	0	—	0.2 V_{DD}	V
f_{osc}	Oscillator Frequency	5V	$R_{osc}=1.1\text{M}\Omega$	—	3	—	kHz
R_{TE}	TE Pull-high Resistance	5V	$V_{TE}=0V$	—	1.5	3	$\text{M}\Omega$

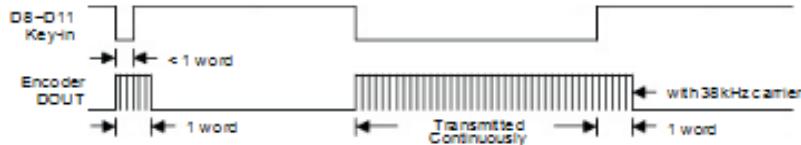
Functional Description

Operation

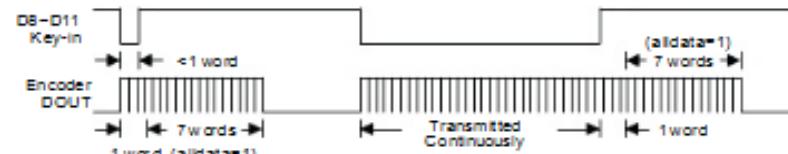
The 2¹² series of encoders begin a 4-word transmission cycle upon receipt of a transmission enable ($\overline{\text{TE}}$ for the HT12E or D8-D11 for the HT12A, active low). This cycle will repeat itself as long as the transmission enable ($\overline{\text{TE}}$ or D8-D11) is held low. Once the transmission enable returns high the encoder output completes its final cycle and then stops as shown below.



Transmission timing for the HT12E



Transmission timing for the HT12A (L/MB=Floating or VDD)

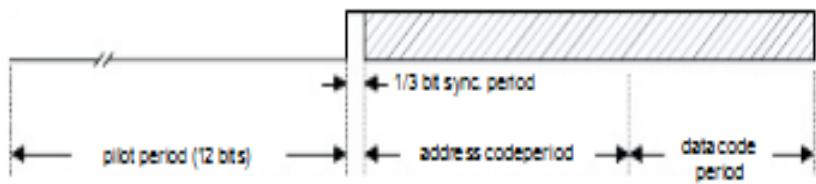


Transmission timing for the HT12A (L/MB=VSS)

Information word

If L/MB=1 the device is in the latch mode (for use with the latch type of data decoders). When the transmission enable is removed during a transmission, the DOUT pin outputs a complete word and then stops. On the other hand, if L/MB=0 the device is in the momentary mode (for use with the momentary type of data decoders). When the transmission enable is removed during a transmission, the DOUT outputs a complete word and then adds 7 words all with the "1" data code.

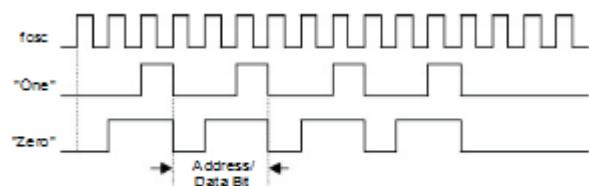
An information word consists of 4 periods as illustrated below.



Composition of information

Address/data waveform

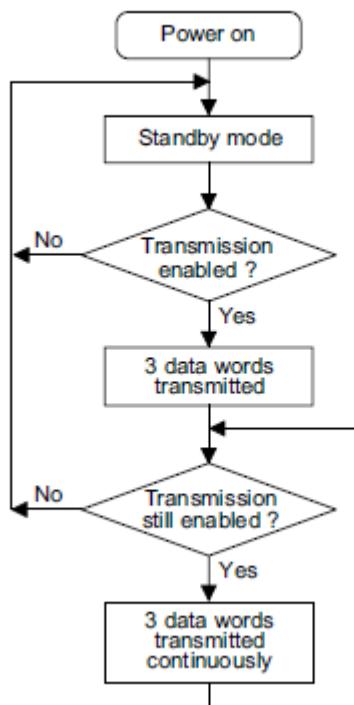
Each programmable address/ data can be externally set to one of the following two logic states as shown below



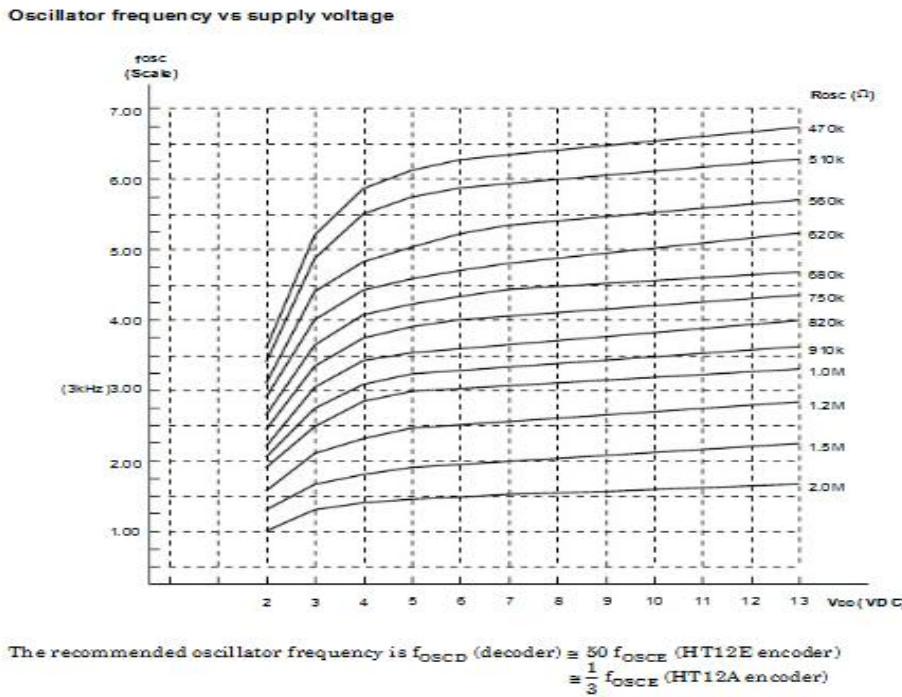
Address/Data bit waveform for the HT12E

Transmission enable

For the TE trigger type of encoders, transmission is enabled by applying a high signal to the TE pin. But for the Data trigger type of encoders, it is enabled by applying a high signal to one of the data pins D12~D17.



FLOWCHART:

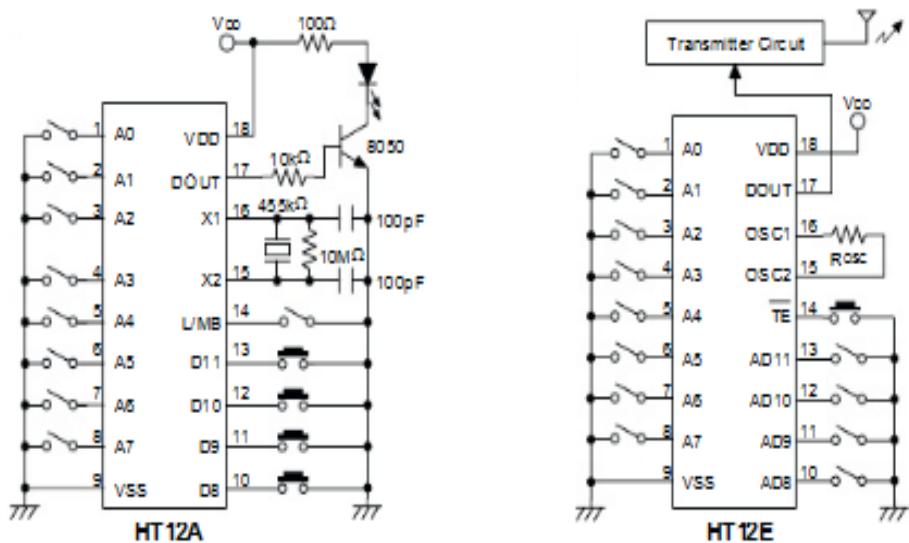


Graph showing Frequency versus Voltage

The graph shown above decides the resistance value to be connected to the oscillator pins of the encoder. The oscillator resistance will have an effect on startup time and steady state amplitude. For the data communication at a particular frequency in the RF range, both the transmitter and receiver should be set to a particular frequency. The exact setting of the frequency can be obtained in the encoder and decoder circuits. The frequency value can be set using the graph. The operating voltage of encoder and decoder is 5V. Thus looking at the graph at 5V VDD, if we select the frequency in the range of 1.25 and 1.50 we are selecting 220k resistance.

BASIC APPLICATION CIRCUIT OF HT12 ENCODER:

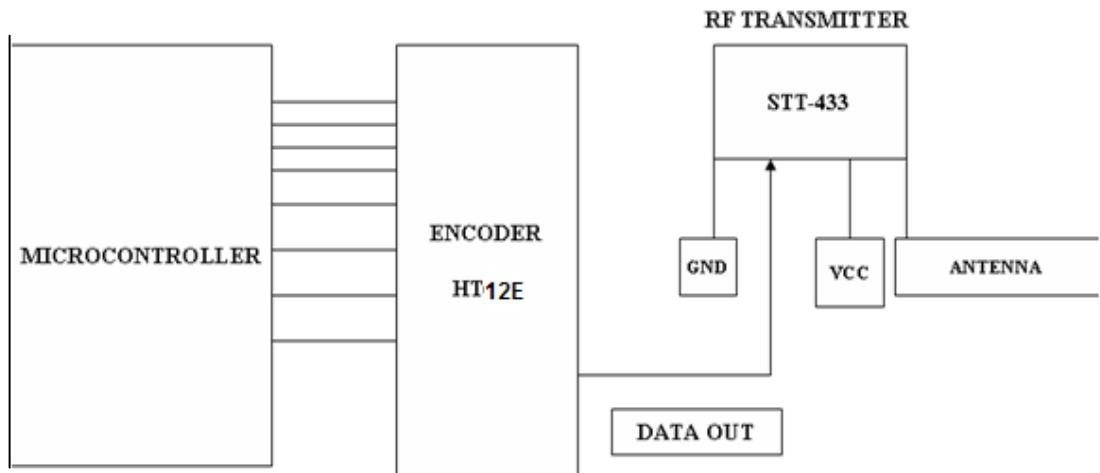
Application Circuits



Note: Typical infrared diode: EL-1L2 (KODENSHI CORP.)

Typical RF transmitter: JR-220 (JUWA CORP.)

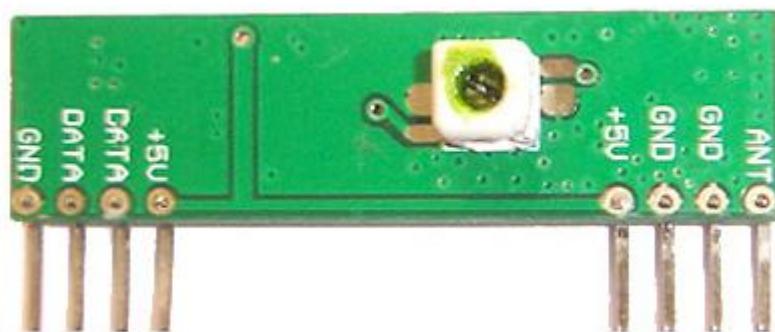
DEMO CIRCUIT: Transmission Circuit



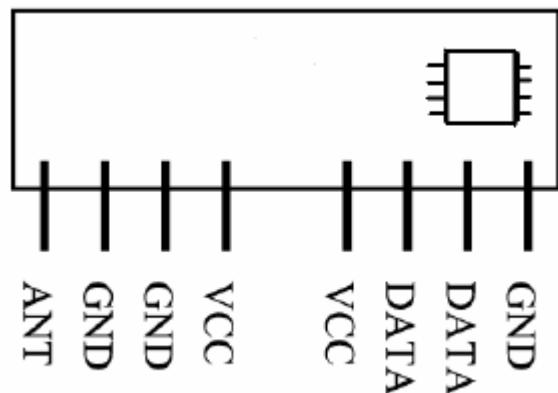
The data sent from the microcontroller is encoded and sent to RF transmitter. The data is transmitted on the antenna pin. Thus, this data should be received on the destination i.e, on RF receiver.

FACTOR INFLUENCED TO CHOOSE STR-433MHz

RF RECEIVER STR-433 MHz:



The data is received by the RF receiver from the antenna pin and this data is available on the data pins. Two Data pins are provided in the receiver module. Thus, this data can be used for further applications



PINOUT:

ANT

Antenna input.

GND

Receiver Ground. Connect to ground plane.

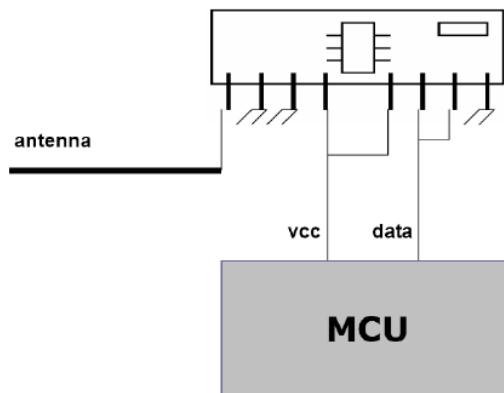
VCC (5V)

VCC pins are electrically connected and provide operating voltage for the receiver. VCC can be applied to either or both. VCC should be bypassed with a $.1\mu\text{F}$ ceramic capacitor. Noise on the power supply will degrade receiver sensitivity.

DATA

Digital data output.

This output is capable of driving one TTL or CMOS load. It is a CMOS compatible output.



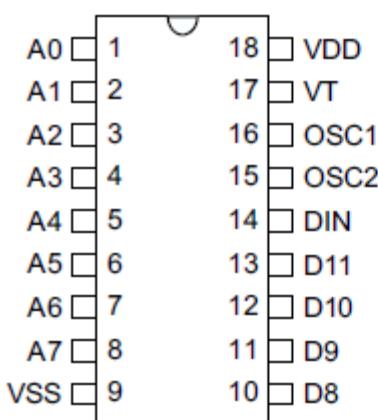
Remark: Antenna length about: 17cm for 433MHz

Similarly, as the transmitter requires an encoder, the receiver module requires a decoder.

The decoder used is **HT12D** from **HOLTEK SEMICONDUCTOR INC.**

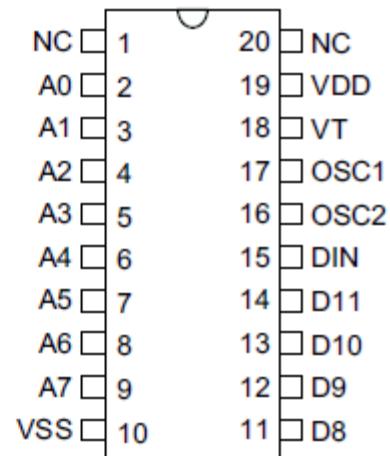
Pin Assignment

8-Address
4-Data



HT12D
- 18 DIP

8-Address
4-Data



HT12D
- 20 SOP

Pin Description

Pin Name	I/O	Internal Connection	Description			
A0~A11	I	NMOS TRANSMISSION GATE	Input pins for address A0~A11 setting They can be externally set to VDD or VSS.			
D8~D11	O	CMOS OUT	Output data pins			
DIN	I	CMOS IN	Serial data input pin			
VT	O	CMOS OUT	Valid transmission, active high			
OSC1	I	OSCILLATOR	Oscillator input pin			
OSC2	O	OSCILLATOR	Oscillator output pin			
VSS	I	—	Negative power supply (GND)			
VDD	I	—	Positive power supply			

Electrical Characteristics

Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	5	12	V
I _{STB}	Standby Current	5V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I _{DD}	Operating Current	5V	No load f _{OSC} =150kHz	—	200	400	μA
I _O	Data Output Source Current (D8~D11)	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	Data Output Sink Current (D8~D11)	5V	V _{OL} =0.5V	1	1.6	—	mA
I _{VT}	VT Output Source Current	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	VT Output Sink Current		V _{OL} =0.5V	1	1.6	—	mA
V _{IH}	"H" Input Voltage	5V	—	3.5	—	5	V
V _{IL}	"L" Input Voltage	5V	—	0	—	1	V
f _{OSC}	Oscillator Frequency	5V	R _{OSC} =51kΩ	—	150	—	kHz

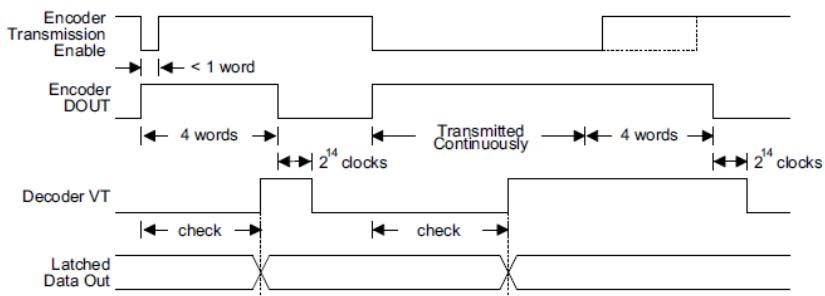
Features

- Operating voltage: 2.4V~12V.
- Low power and high noise immunity CMOS technology.
- Low standby current.
- Capable of decoding 18 bits of information.
- Pairs with HOLTEK's 318 series of encoders.
- 8~18 address pins.
- 0~8 data pins.

HOW DOES THE DECODER WORK?

- The 3^18 decoders are a series of CMOS LSIs for remote control system applications. They are paired with the 3^18 series of encoders.
- For proper operation, a pair of encoder/decoder pair with the same number of address and data format should be selected.
- The 3^18 series of decoders receives serial address and data from that series of encoders that are transmitted by a carrier using an RF medium.
- A signal on the DIN pin then activates the oscillator which in turns decodes the incoming address and data.
- It then compares the serial input data twice continuously with its local address.
- If no errors or unmatched codes are encountered, the input data codes are decoded and then transferred to the output pins.
- The VT pin also goes high to indicate a valid transmission. That will last until the address code is incorrect or no signal has been received.
- The 3^18 decoders are capable of decoding 18 bits of information that consists of N bits of address and 18-N bits of data.

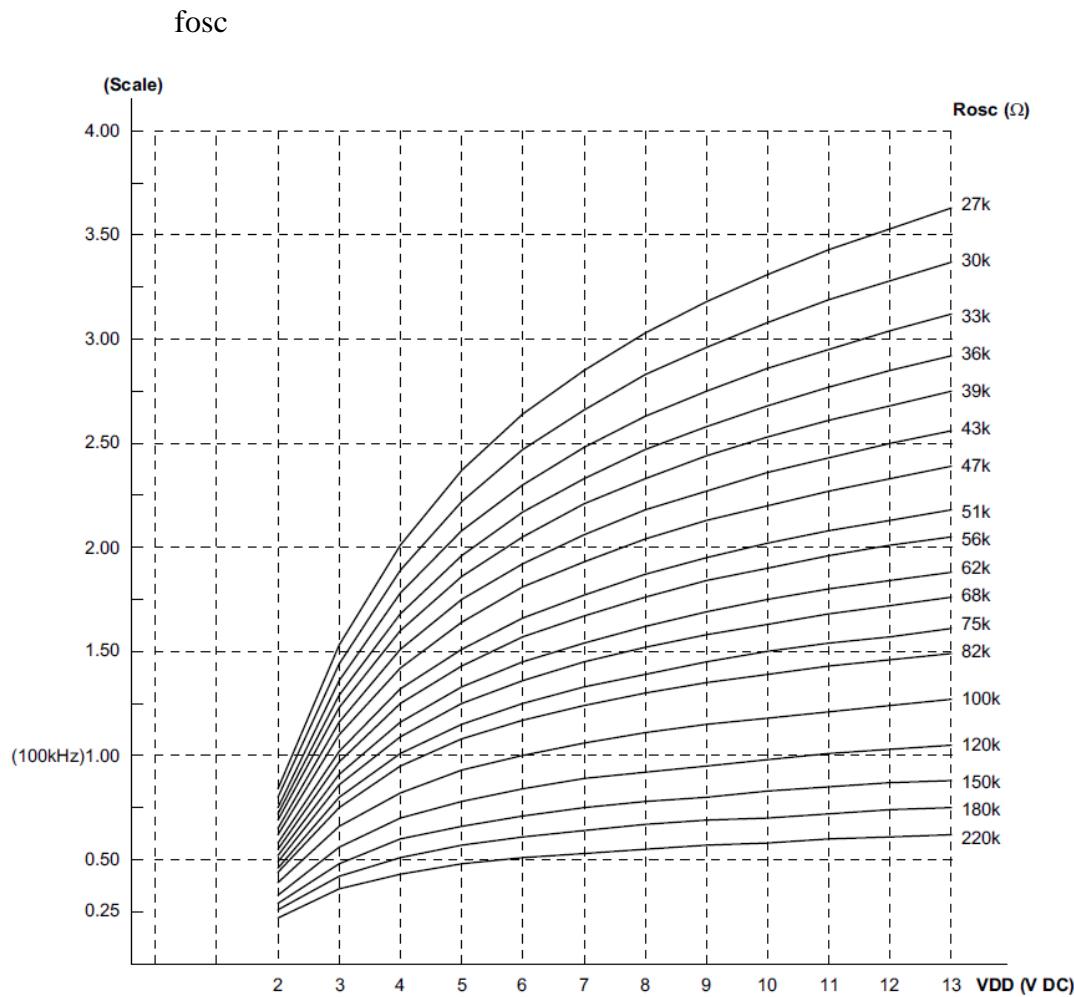
Decoder timing



Encoder/Decoder cross reference table

Decoders Part No.	Data Pins	Address Pins	VT	Pair Encoder	Package			
					Encoder		Decoder	
					DIP	SOP	DIP	SOP
HT12D	4	8	✓	HT12A	18	20	18	20
				HT12E	18	20		
HT12F	0	12	✓	HT12A	18	20	18	20
				HT12E	18	20		

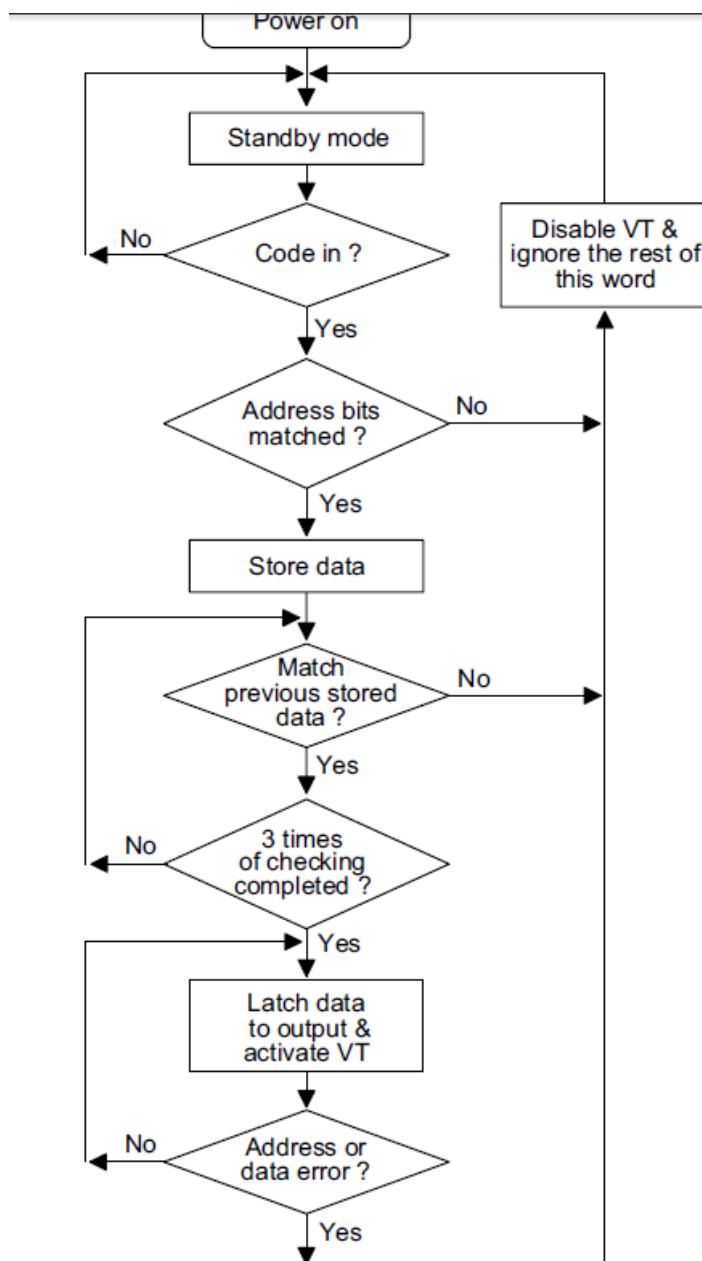
Oscillator frequency versus supply voltage



The recommended oscillator frequency is f_{OSCD} (decoder) $\cong 50 f_{OSCE}$ (HT12E encoder)

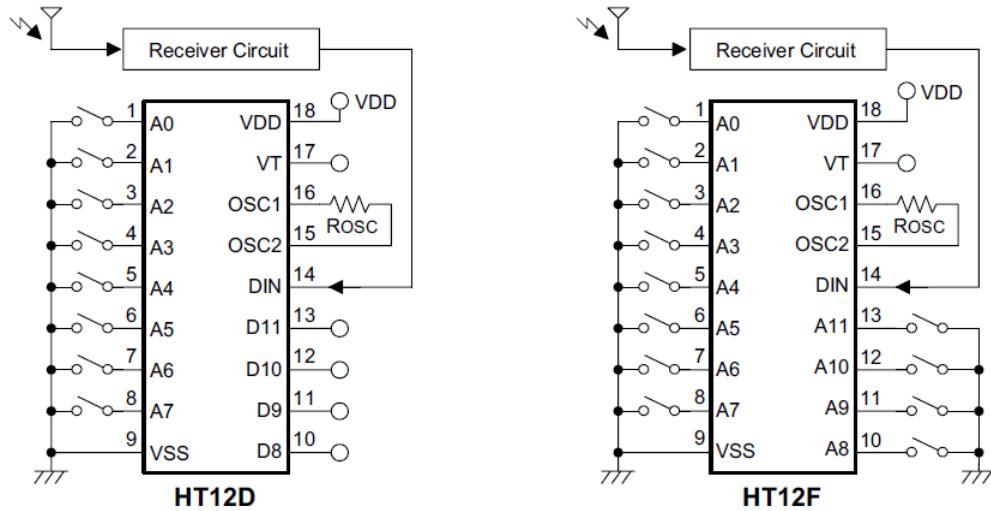
$$\cong \frac{1}{3} f_{OSCE} \text{ (HT12A encoder).}$$

FLOW CHART:



BASIC APPLICATION CIRCUIT OF HT12D DECODER:

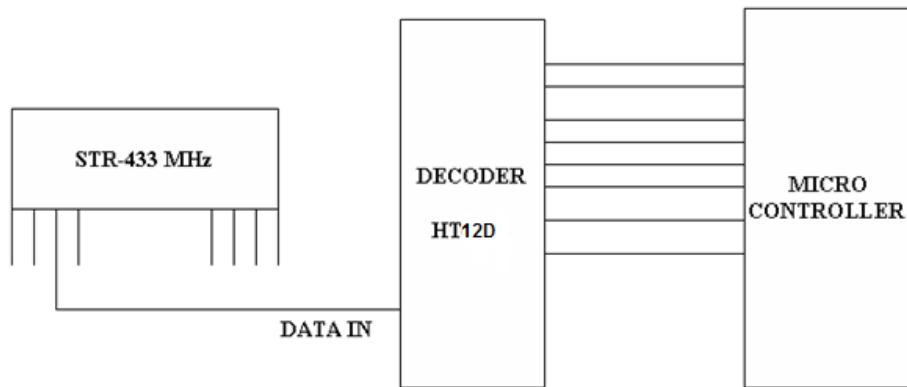
Application Circuits



Notes: Typical infrared receiver: PIC-12043T/PIC-12043S (KODESHI CORP.)
or LTM9052 (LITEON CORP.)

Typical RF receiver: JR-200 (JUWA CORP.)
RE-99 (MING MICROSYSTEM, U.S.A.)

DEMO CIRCUIT: Reception circuit



The data transmitted into the air is received by the receiver. The received data is taken from the data line of the receiver and is fed to the decoder .The output of decoder is given to microcontroller and then data is processed according to the applications.

BC 557 TRANSISTOR:

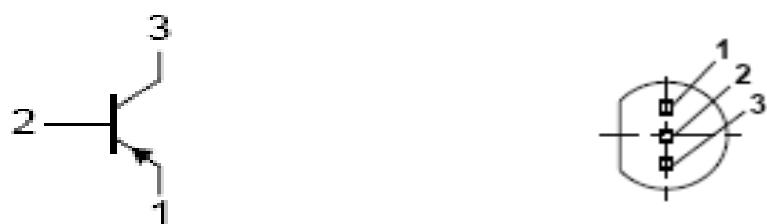


Fig: Simplified outline and symbol

PIN	DESCRIPTION
1	emitter
2	base
3	collector

Fig: Pin diagram

FEATURES

- Low current (max. 100 mA)
- Low voltage (max. 65 V).

APPLICATIONS

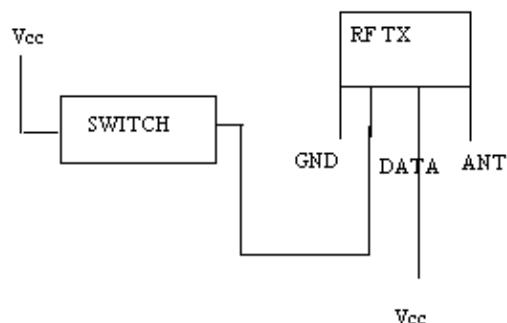
- General purpose switching and amplification

LIMITING VALUES:

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CBO}	collector-base voltage BC556	open emitter	-	-80	V
	BC557			-50	V
V_{CEO}	collector-emitter voltage BC556	open base	-	-65	V
	BC557			-45	V
V_{EBO}	emitter-base voltage	open collector	-	-5	V
I_c	collector current (DC)		-	-100	mA
I_{CM}	peak collector current		-	-200	mA
I_{BM}	peak base current		-	-200	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25^\circ\text{C}$	-	500	mW
T_{stg}	storage temperature		-65	+150	°C
T_j	junction temperature		-	150	°C
T_{amb}	ambient temperature		-65	+150	°C

BC 557 PNP Transistor acts as a switch is used in this project.

TRANSMITTER CIRCUIT:



DC MOTOR AND H-BRIDGE

THEORY OF DC MOTOR

The speed of a DC motor is directly proportional to the supply voltage, so if we reduce the supply voltage from 12 Volts to 6 Volts, the motor will run at half the speed. How can this be achieved when the battery is fixed at 12 Volts? The speed controller works by varying the average voltage sent to the motor. It could do this by simply adjusting the voltage sent to the motor, but this is quite inefficient to do. A better way is to switch the motor's supply on and off very quickly. If the switching is fast enough, the motor doesn't notice it, it only notices the average effect.

When you watch a film in the cinema, or the television, what you are actually seeing is a series of fixed pictures, which change rapidly enough that your eyes just see the average effect - movement. Your brain fills in the gaps to give an average effect.

Now imagine a light bulb with a switch. When you close the switch, the bulb goes on and is at full brightness, say 100 Watts. When you open the switch it goes off (0 Watts). Now if you close the switch for a fraction of a second, then open it for the same amount of time, the filament won't have time to cool down and heat up, and you will just get an average glow of 50 Watts. This is how lamp dimmers work, and the same principle is used by speed controllers to drive a motor. When the switch is closed, the motor sees 12 Volts, and when it is open it sees 0 Volts. If the switch is open for the same amount of time as it is closed, the motor will see an average of 6 Volts, and will run more slowly accordingly. The graph below shows the speed of a motor that is being turned on and off

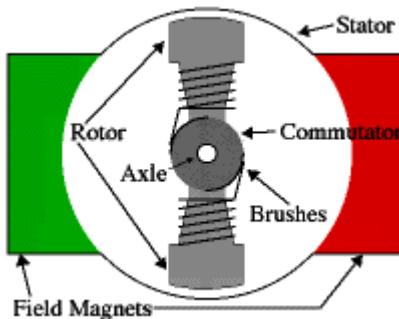
DC MOTOR

An electric motor is a machine which converts electrical energy into mechanical energy.

Principles of operation

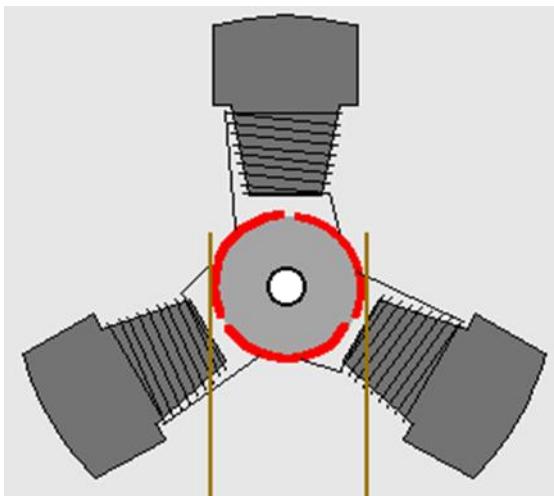
In any electric motor, operation is based on simple electromagnetism. A current-carrying conductor generates a magnetic field; when this is then placed in an external magnetic field, it will experience a force proportional to the current in the conductor, and to the strength of the external magnetic field. As you are well aware of from playing with magnets as a kid, opposite (North and South) polarities attract, while like polarities (North and North, South and South) repel. The internal configuration of a DC motor is designed to harness the magnetic interaction between a current-carrying conductor and an external magnetic field to generate rotational motion.

Let's start by looking at a simple 2-pole DC electric motor (here red represents a magnet or winding with a "North" polarization, while green represents a magnet or winding with a "South" polarization).



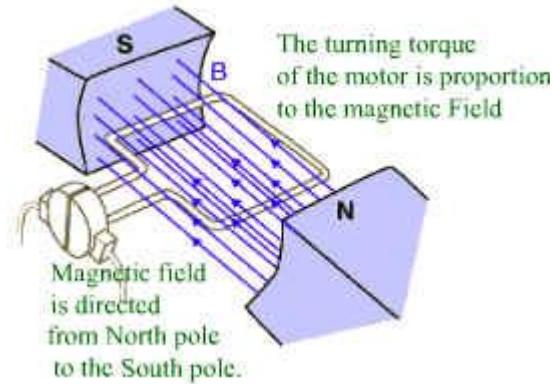
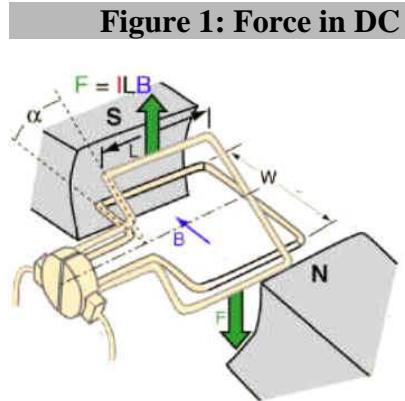
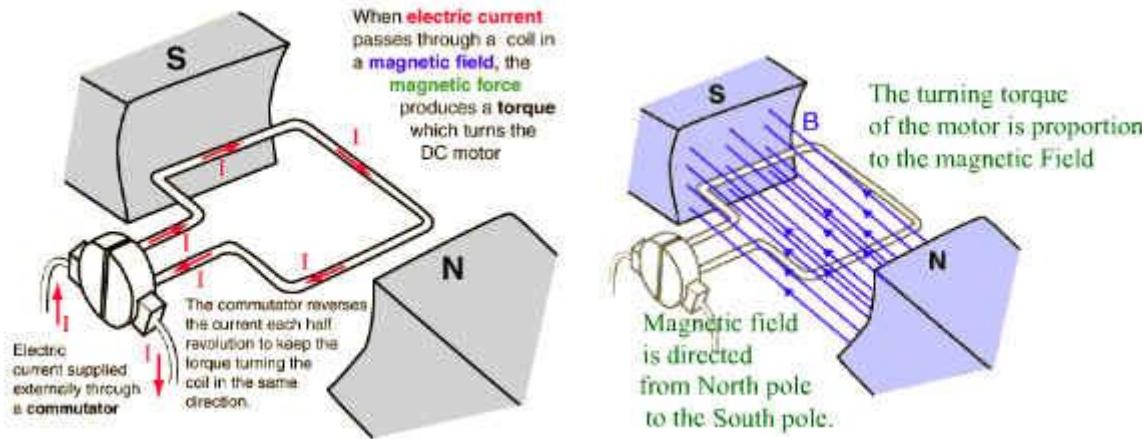
Every DC motor has six basic parts -- axle, rotor (a.k.a., armature), stator, commutator, field magnet(s), and brushes. In most common DC motors (and all that BEAMers will see), the external magnetic field is produced by high-strength permanent magnets¹. The stator is the stationary part of the motor -- this includes the motor casing, as well as two or more permanent magnet pole pieces. The rotor (together with the axle and attached commutator) rotate with respect to the stator. The rotor consists of windings (generally on a core), the windings being electrically connected to the commutator. The above diagram shows a common motor layout -- with the rotor inside the stator (field) magnets.

The geometry of the brushes, commutator contacts, and rotor windings are such that when power is applied, the polarities of the energized winding and the stator magnet(s) are misaligned, and the rotor will rotate until it is almost aligned with the stator's field magnets. As the rotor reaches alignment, the brushes move to the next commutator contacts, and energize the next winding. Given our example two-pole motor, the rotation reverses the direction of current through the rotor winding, leading to a "flip" of the rotor's magnetic field, driving it to continue rotating.



In real life, though, DC motors will always have more than two poles (three is a very common number). In particular, this avoids "dead spots" in the commutator. You can imagine how with our example two-pole motor, if the rotor is exactly at the middle of its rotation (perfectly aligned with the field magnets), it will get "stuck" there. Meanwhile, with a two-pole motor,

there is a moment where the commutator shorts out the power supply (i.e., both brushes touch both commutator contacts simultaneously). This would be bad for the power supply, waste energy, and damage motor components as well. Yet another disadvantage of such a simple motor is that it would exhibit a high amount of torque "ripple" (the amount of torque it could produce is cyclic with the position of the rotor).

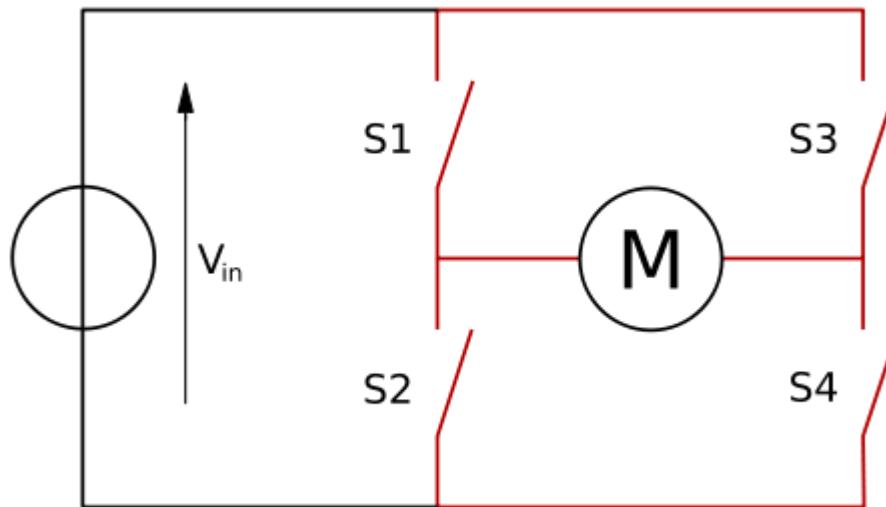


If an Electric current flows through two copper wires that are between the poles of a magnet, an upward force will move one wire up and a downward force will move the other wire down. The loop can be made to spin by fixing a half circle of copper which is known as COMMUTATOR to each end of the loop. Current is passed into and out of the loop by brushes that press onto the strips. The BRUSHES do not go round so the wire do not get twisted. This arrangement also

makes sure that the current always passes down on the right and back on the left so that the rotation continues. This is how a simple Electric motor is made.

H-BRIDGE

An H-bridge is an electronic circuit which enables DC electric motors to be run forwards or backwards. These circuits are often used in robotics. H-bridges are available as integrated circuits, or can be built from discrete components.



The two basic states of a H-bridge. The term "H-bridge" is derived from the typical graphical representation of such a circuit. An H-bridge is built with four switches (solid-state or mechanical). When the switches S1 and S4 (according to the first figure) are closed (and S2 and S3 are open) a positive voltage will be applied across the motor. By opening S1 and S4 switches and closing S2 and S3 switches, this voltage is reversed, allowing reverse operation of the motor.

Using the nomenclature above, the switches S1 and S2 should never be closed at the same time, as this would cause a short circuit on the input voltage source. The same applies to the switches S3 and S4. This condition is known as shoot-through.

Operation

The H-Bridge arrangement is generally used to reverse the polarity of the motor, but can also be used to 'brake' the motor, where the motor comes to a sudden stop, as the motors terminals are shorted, or to let the motor 'free run' to a stop, as the motor is effectively disconnected from the circuit. The following table summarizes operation.

S1	S2	S3	S4	Result
1	0	0	1	Motor moves right
0	1	1	0	Motor moves left
0	0	0	0	Motor free runs
0	1	0	1	Motor brakes

H-Bridge Driver:

The switching property of this H-Bridge can be replace by a Transistor or a Relay or a Mosfet or even by an IC. Here we are replacing this with an IC named L293D as the driver whose description is as given below.

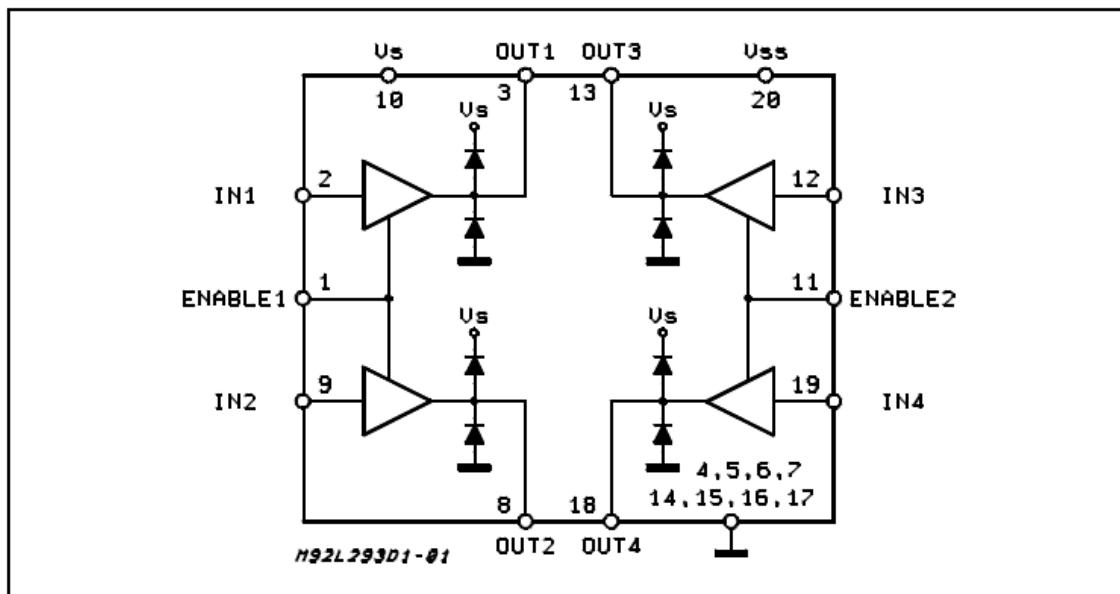
Features:

- 600mA OUTPUT CURRENT CAPABILITY
- PER CHANNEL
- 1.2A PEAK OUTPUT CURRENT (non repetitive)
- PER CHANNEL
- ENABLE FACILITY
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V
- (HIGH NOISE IMMUNITY)
- INTERNAL CLAMP DIODES

DESCRIPTION

The Device is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and drive inductive loads (such as relays solenoides, DC and stepping motors) and switching power transistors. To simplify use as two bridges each pair of channels is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a lower voltage and internal clamp diodes are included. This device is suitable for use in switching applications at frequencies up to 5 kHz. The L293D is assembled in a 16 lead plastic packaage which has 4 center pins connected together and used for heatsinking. The L293DD is assembled in a 20 lead surface mount which has 8 center pins connected together and used for heatsinking.

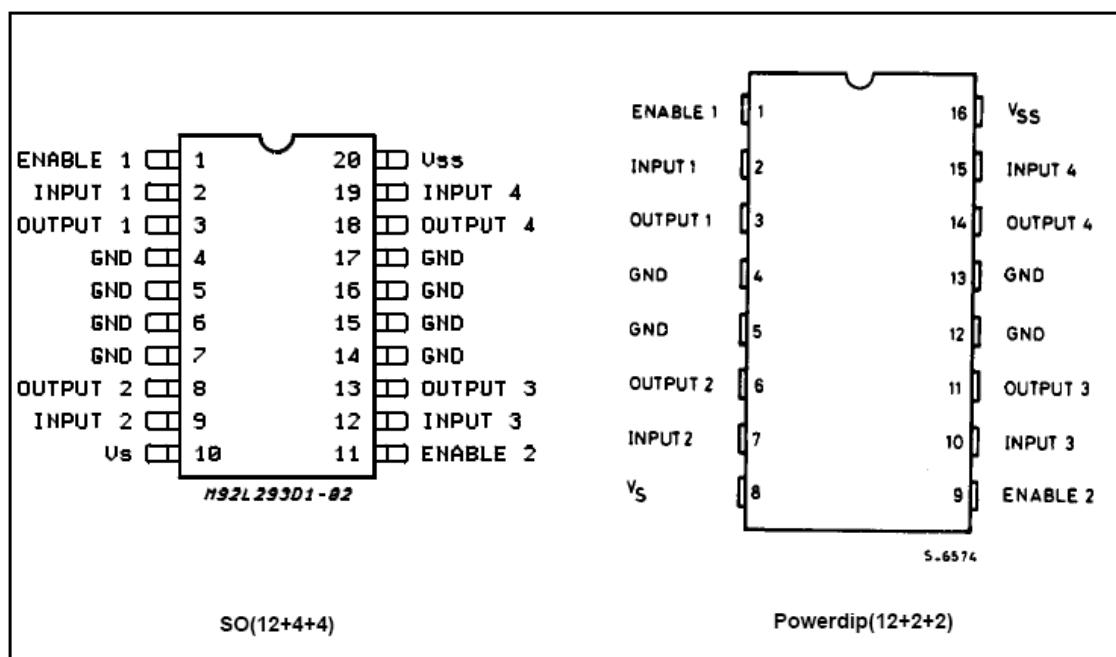
BLOCK DIAGRAM:



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_s	Supply Voltage	36	V
V_{ss}	Logic Supply Voltage	36	V
V_i	Input Voltage	7	V
V_{en}	Enable Voltage	7	V
I_o	Peak Output Current (100 μ s non repetitive)	1.2	A
P_{tot}	Total Power Dissipation at $T_{pins} = 90^\circ\text{C}$	4	W
T_{stg}, T_j	Storage and Junction Temperature	- 40 to 150	$^\circ\text{C}$

PIN CONNECTIONS



RELAYS:

A relay is an electrically controllable switch widely used in industrial controls, automobiles and appliances.

The relay allows the isolation of two separate sections of a system with two different voltage sources i.e., a small amount of voltage/current on one side can handle a large amount of voltage/current on the other side but there is no chance that these two voltages mix up.

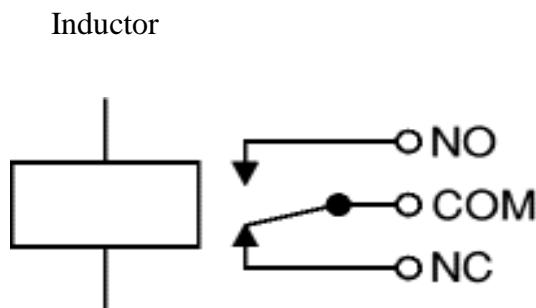


Fig: Circuit symbol of a relay

Operation:

When a current flows through the coil, a magnetic field is created around the coil i.e., the coil is energized. This causes the armature to be attracted to the coil. The armature's contact acts like a switch and closes or opens the circuit. When the coil is not energized, a spring pulls the armature to its normal state of open or closed. There are all types of relays for all kinds of applications.

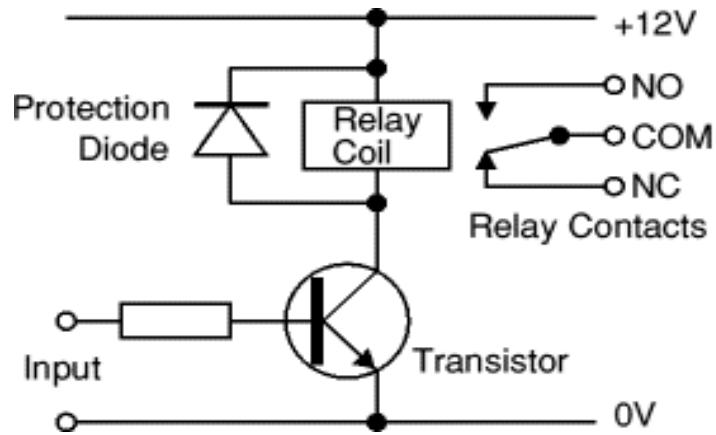


Fig: Relay Operation and use of protection diodes

Transistors and ICs must be protected from the brief high voltage 'spike' produced when the relay coil is switched off. The above diagram shows how a signal diode (eg 1N4148) is connected across the relay coil to provide this protection. The diode is connected 'backwards' so that it will normally not conduct. Conduction occurs only when the relay coil is switched off, at this moment the current tries to flow continuously through the coil and it is safely diverted through the diode. Without the diode no current could flow and the coil would produce a damaging high voltage 'spike' in its attempt to keep the current flowing.

In choosing a relay, the following characteristics need to be considered:

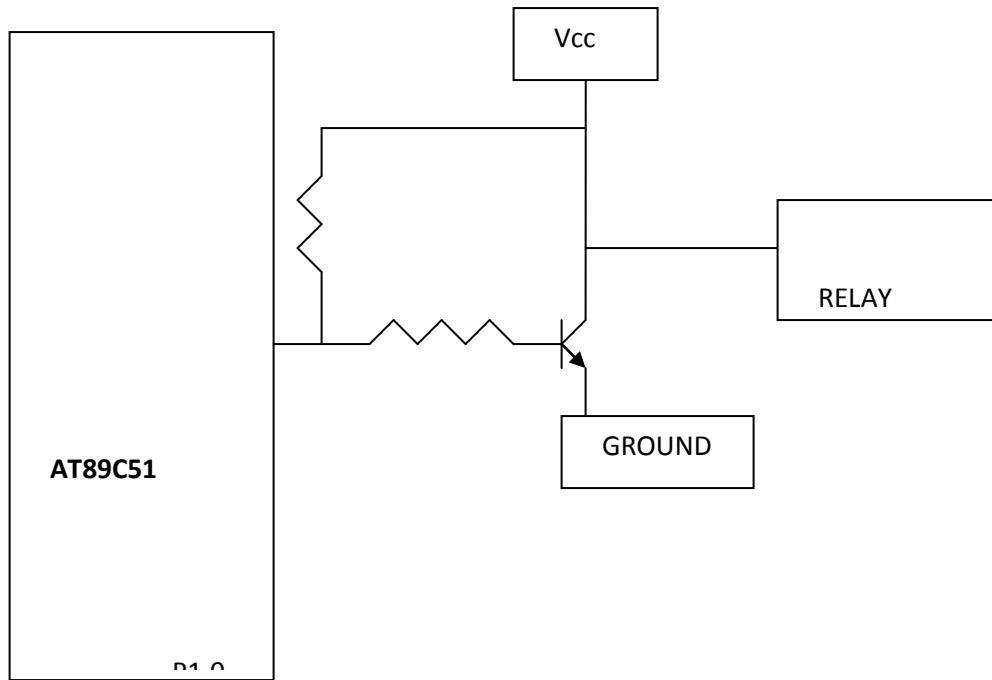
1. The contacts can be normally open (NO) or normally closed (NC). In the NC type, the contacts are closed when the coil is not energized. In the NO type, the contacts are closed when the coil is energized.
2. There can be one or more contacts. i.e., different types like SPST (single pole single throw), SPDT (single pole double throw) and DPDT (double pole double throw) relays.

3. The voltage and current required to energize the coil. The voltage can vary from a few volts to 50 volts, while the current can be from a few milliamps to 20 milliamps. The relay has a minimum voltage, below which the coil will not be energized. This minimum voltage is called the “pull-in” voltage.
4. The minimum DC/AC voltage and current that can be handled by the contacts. This is in the range of a few volts to hundreds of volts, while the current can be from a few amps to 40A or more, depending on the relay.

TRANSISTOR DRIVER CIRCUIT:

An SPDT relay consists of five pins, two for the magnetic coil, one as the common terminal and the last pins as normally connected pin and normally closed pin. When the current flows through this coil, the coil gets energized. Initially when the coil is not energized, there will be a connection between the common terminal and normally closed pin. But when the coil is energized, this connection breaks and a new connection between the common terminal and normally open pin will be established. Thus when there is an input from the microcontroller to the relay, the relay will be switched on. Thus when the relay is on, it can drive the loads connected between the common terminal and normally open pin. Therefore, the relay takes 5V from the microcontroller and drives the loads which consume high currents. Thus the relay acts as an isolation device.

Digital systems and microcontroller pins lack sufficient current to drive the relay. While the relay's coil needs around 10 milliamps to be energized, the microcontroller's pin can provide a maximum of 1-2 milliamps current. For this reason, a driver such as a power transistor is placed in between the microcontroller and the relay.



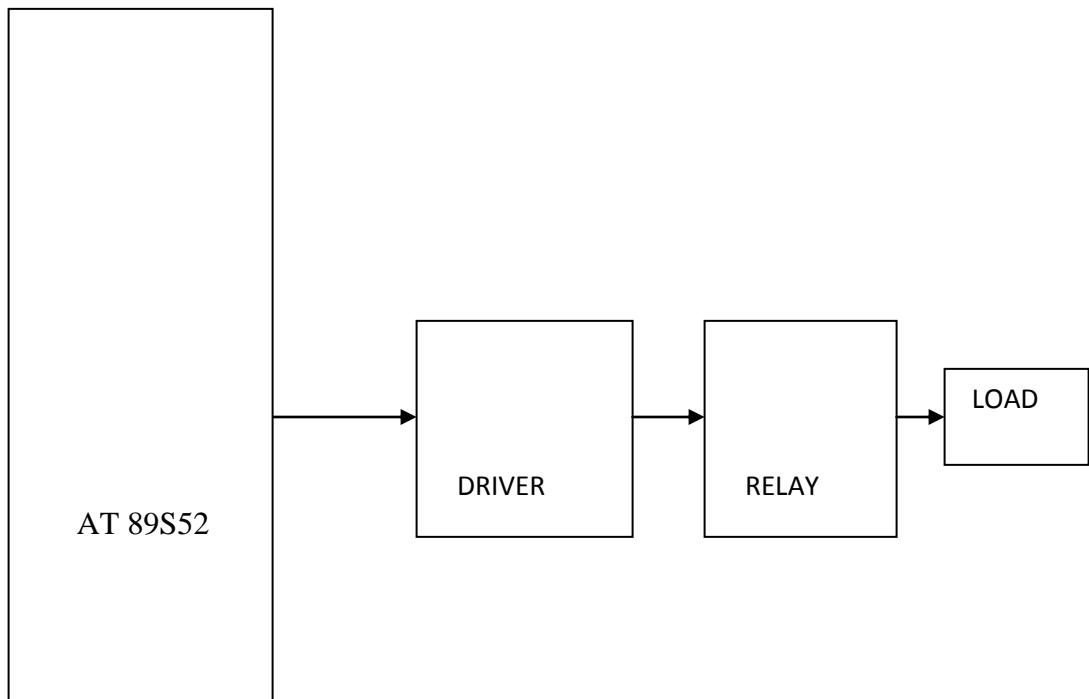
The operation of this circuit is as follows:

The input to the base of the transistor is applied from the microcontroller port pin P1.0.

The transistor will be switched on when the base to emitter voltage is greater than 0.7V (cut-in voltage). Thus when the voltage applied to the pin P1.0 is high i.e., P1.0=1 ($>0.7V$), the transistor will be switched on and thus the relay will be ON and the load will be operated.

When the voltage at the pin P1.0 is low i.e., P1.0=0 ($<0.7V$) the transistor will be in off state and the relay will be OFF. Thus the transistor acts like a current driver to operate the relay accordingly.

RELAY INTERFACING WITH THE MICROCONTROLLER:



MAX 232:

Max232 IC is a specialized circuit which makes standard voltages as required by RS232 standards. This IC provides best noise rejection and very reliable against discharges and short circuits. MAX232 IC chips are commonly referred to as line drivers.

To ensure data transfer between PC and microcontroller, the baud rate and voltage levels of Microcontroller and PC should be the same. The voltage levels of microcontroller are logic1 and logic 0 i.e., logic 1 is +5V and logic 0 is 0V. But for PC, RS232 voltage levels are considered and they are: logic 1 is taken as -3V to -25V and logic 0 as +3V to +25V. So, in order to equal these voltage levels, MAX232 IC is used. Thus this IC converts RS232 voltage levels to microcontroller voltage levels and vice versa.

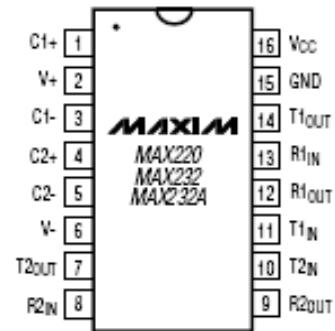
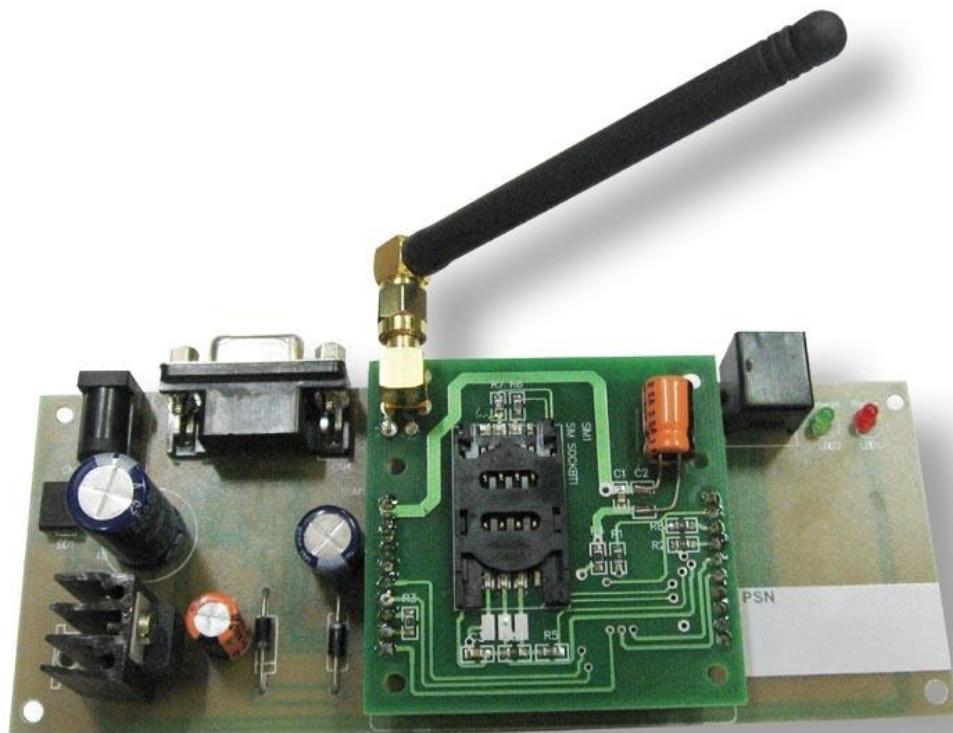


Fig: Pin diagram of MAX 232 IC

SPECIFICATIONS OF GSM MODEM



GSM Modem Specification

This GSM modem is a highly flexible plug and play quad band GSM modem for direct and easy integration to RS232. Supports features like Voice, Data/Fax, SMS,GPRS and integrated TCP/IP stack.

Features

- Quad Band GSM/GPRS 850/900/1800/1900 Mhz
- GPRS multi-slot class 10/8
- GPRS Mobile station class B
- Compliant to GSM Phase 2/2+
- Class 4 (2W@850/900Mhz)
- Class 1(1W@1800/1900Mhz)
- Control via AT commands(GSM 07.07,07.05 and enhanced AT commands)
- Operation Temperature(-20 deg C to +55 deg C)

Specifications for Voice

- Tricodec
- Half rate(HR)
- Full rate(FR)
- Enhanced Full rate(EFR)
- Hands-free operation(Echo suppression)

Specifications for Fax

Group 3, class 1

Specifications for data

- GPRS class 10: max 85.6 kbps(downlink)
- PBCCH support
- Coding schemes CS 1,2,3,4
- CSD upto 14.4 kbps
- USSD
- Non transparent mode
- PPP-Stack

Specifications for SMS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Power Supply

- Use AC – DC Power Adaptor with following ratings
- DC Voltage : 12V
- DC Current : 1A
- Polarity : Centre +ve & Outside –ve
- Current Consumption in normal operation 250mA, can rise up to 1Amp while transmission.

Interfaces

- RS-232 through D-TYPE 9 pin connector, Serial port baud rate adjustable 1200 to 115200 bps (9600 default)
- Stereo connector for MIC & SPK
- Power supply through DC socket
- SMA antenna connector
- Push switch type SIM holder
- LED status of GSM / GPRS module

Getting Started

- **Insert SIM card:** Press the yellow pin to remove the tray from the SIM cardholder. After properly fixing the SIM card in the tray, insert the tray in the slot provided.
- **Connect Antenna:** Screw the RF antenna on the RF cable output provided.
- If voice call is needed, connect the mic and speaker to stereo sockets.
- **Connect RS232 Cable:** (Cable provided for RS232 communication) Default baud rate is 9600 with 8-N-1, no hardware handshaking. Cable provided has pins 7 and 8 shorted that will set to no hardware handshaking. If you need hardware handshaking the pins 7-8 can be taken for signaling.
- Pin 2 is RS232 level TX out
- Pin 3 is RS232 level RX in
- Pin 5 is Ground
- Pin 7 RTS in (shorted to pin 8 in cable for no hardware handshaking)

- Pin 8 CTS out (shorted to pin 7 in cable for no hardware handshaking)
- **Connect the power Supply** (9-12V) to the power jack. Polarity should be Center +ve and outer -ve DC jack.
- **Network Led** indicating various status of GSM module eg. Power on, network registration & GPRS connectivity.
- After the Modem registers the network, led will blink in step of 3 seconds. At this stage you can start using Modem for your application.

LIQUID CRYSTAL DISPLAY:

LCD stands for **Liquid Crystal Display**. LCD is finding wide spread use replacing LEDs (seven segment LEDs or other multi segment LEDs) because of the following reasons:

1. The declining prices of LCDs.
2. The ability to display numbers, characters and graphics. This is in contrast to LEDs, which are limited to numbers and a few characters.
3. Incorporation of a refreshing controller into the LCD, thereby relieving the CPU of the task of refreshing the LCD. In contrast, the LED must be refreshed by the CPU to keep displaying the data.
4. Ease of programming for characters and graphics.

These components are “specialized” for being used with the microcontrollers, which means that they cannot be activated by standard IC circuits. They are used for writing different messages on a miniature LCD.



Function	Pin Number	Name	Logic State	Description
Ground	1	Vss	-	0V
Power supply	2	Vdd	-	+5V
Contrast	3	Vee	-	0 - Vdd
	4	RS	0 1	D0 – D7 are interpreted as commands D0 – D7 are interpreted as data
Control of operating	5	R/W	0 1	Write data (from controller to LCD) Read data (from LCD to controller)
	6	E	0 1 From 1 to 0	Access to LCD disabled Normal operating Data/commands are transferred to LCD
Data / commands	7	D0	0/1	Bit 0 LSB
	8	D1	0/1	Bit 1
	9	D2	0/1	Bit 2
	10	D3	0/1	Bit 3
	11	D4	0/1	Bit 4
	12	D5	0/1	Bit 5
	13	D6	0/1	Bit 6
	14	D7	0/1	Bit 7 MSB

A model described here is for its low price and great possibilities most frequently used in practice. It is based on the HD44780 microcontroller (*Hitachi*) and can display messages in two lines with 16 characters each. It displays all the alphabets, Greek letters, punctuation marks, mathematical symbols etc. In addition, it is possible to display symbols that user makes up on its own. Automatic shifting message on display (shift left and right), appearance of the pointer, backlight etc. are considered as useful characteristics.

Pins Functions

There are pins along one side of the small printed board used for connection to the microcontroller. There are total of 14 pins marked with numbers (16 in case the background light is built in). Their function is described in the table below:

LCD screen:

LCD screen consists of two lines with 16 characters each. Each character consists of 5x7 dot matrix. Contrast on display depends on the power supply voltage and whether messages are displayed in one or two lines. For that reason, variable voltage 0-Vdd is applied on pin marked as Vee. Trimmer potentiometer is usually used for that purpose.

Some versions of displays have built in backlight (blue or green diodes). When used during operating, a resistor for current limitation should be used (like with any LE diode).

LCD Basic Commands:

All data transferred to LCD through outputs D0-D7 will be interpreted as commands or as data, which depends on logic state on pin RS:

RS = 1 - Bits D0 - D7 are addresses of characters that should be displayed. Built in processor addresses built in “map of characters” and displays corresponding symbols. Displaying position is determined by DDRAM address. This address is either previously defined or the address of previously transferred character is automatically incremented.

RS = 0 - Bits D0 - D7 are commands which determine display mode. List of commands which LCD recognizes are given in the table below:

Command	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0	Execution Time
Clear display	0	0	0	0	0	0	0	0	0	1	1.64mS
Cursor home	0	0	0	0	0	0	0	0	1	x	1.64mS
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	40uS
Display on/off control	0	0	0	0	0	0	1	D	U	B	40uS
Cursor/Display Shift	0	0	0	0	0	1	D/C	R/L	x	x	40uS
Function set	0	0	0	0	1	DL	N	F	x	x	40uS
Set CGRAM address	0	0	0	1							40uS
Set DDRAM address	0	0	1								40uS
Read “BUSY” flag (BF)	0	1	BF								-

Write to CGRAM or
DDRAM 1 0 D7 D6 D5 D4 D3 D2 D1 D0 40uS

Read from CGRAM or
DDRAM 1 1 D7 D6 D5 D4 D3 D2 D1 D0 40uS

List of commands which LCD recognizes

I/D 1 = Increment (by 1)	R/L 1 = Shift right
0 = Decrement (by 1)	0 = Shift left
S 1 = Display shift on	DL 1 = 8-bit interface
0 = Display shift off	0 = 4-bit interface
D 1 = Display on	N 1 = Display in two lines
0 = Display off	0 = Display in one line
U 1 = Cursor on	F 1 = Character format 5x10 dots
0 = Cursor off	0 = Character format 5x7 dots
B 1 = Cursor blink on	D/C 1 = Display shift
0 = Cursor blink off	0 = Cursor shift

LCD Connection:

Depending on how many lines are used for connection to the microcontroller, there are 8-bit and 4-bit LCD modes. The appropriate mode is determined at the beginning of the process in a phase called “initialization”. In the first case, the data are

transferred through outputs D0-D7 as it has been already explained. In case of 4-bit LED mode, for the sake of saving valuable I/O pins of the microcontroller, there are only 4 higher bits (D4-D7) used for communication, while other may be left unconnected.

Consequently, each data is sent to LCD in two steps: four higher bits are sent first (that normally would be sent through lines D4-D7), four lower bits are sent afterwards. With the help of initialization, LCD will correctly connect and interpret each data received. Besides, with regards to the fact that data are rarely read from LCD (data mainly are transferred from microcontroller to LCD) one more I/O pin may be saved by simple connecting R/W pin to the Ground. Such saving has its price. Even though message displaying will be normally performed, it will not be possible to read from busy flag since it is not possible to read from display.

LCD Initialization:

Once the power supply is turned on, LCD is automatically cleared. This process lasts for approximately 15mS. After that, display is ready to operate. The mode of operating is set by default. This means that:

1. Display is cleared

2. Mode

$DL = 1$ Communication through 8-bit interface

$N = 0$ Messages are displayed in one line

F = 0 Character font 5 x 8 dots

3. Display/Cursor on/off

D = 0 Display off

U = 0 Cursor off

B = 0 Cursor blink off

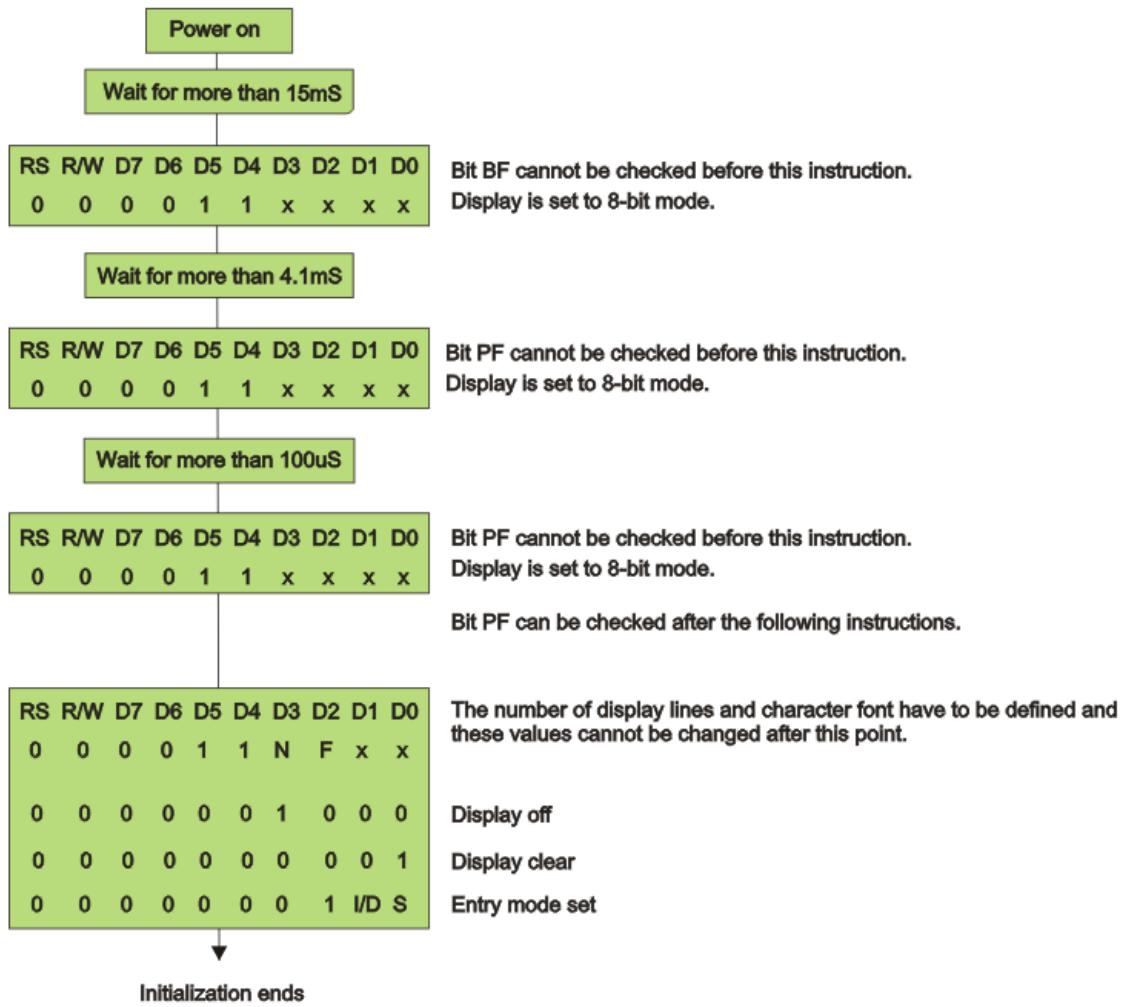
4. Character entry

ID = 1 Addresses on display are automatically incremented by 1

S = 0 Display shift off

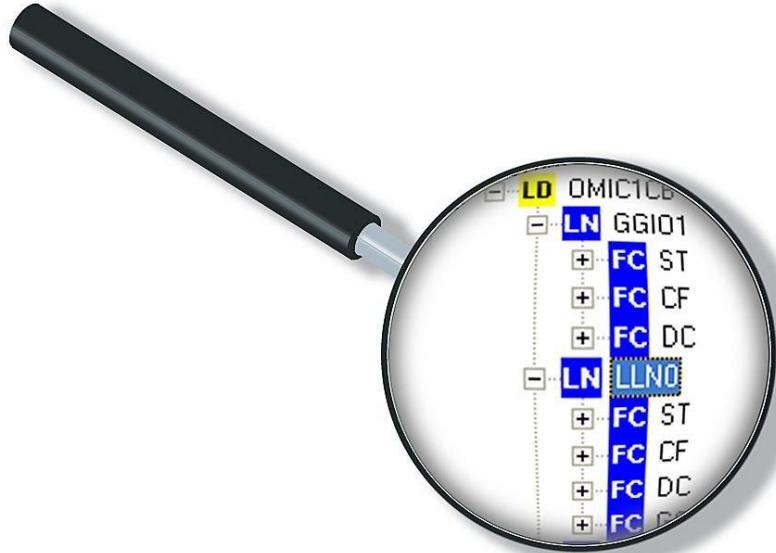
Automatic reset is mainly performed without any problems. Mainly but not always! If for any reason power supply voltage does not reach full value in the course of 10mS, display will start perform completely unpredictably. If voltage supply unit can not meet this condition or if it is needed to provide completely safe operating, the process of initialization by which a new reset enabling display to operate normally must be applied.

Algorithm according to the initialization is being performed depends on whether connection to the microcontroller is through 4- or 8-bit interface. All left over to be done after that is to give basic commands and of course- to display messages.



Procedure on 8-bit initialization.

Software Tools



KEIL SOFTWARE:

Keil compiler is a software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. Keil compiler also supports C language code.

STEPS TO WRITE AN ASSEMBLY LANGUAGE PROGRAM IN KEIL AND HOW TO COMPILE IT:

1. Install the Keil Software in the PC in any of the drives.
2. After installation, an icon will be created with the name “Keil uVision3”. Just drag this icon onto the desktop so that it becomes easy whenever you try to write programs in keil.
3. Double click on this icon to start the keil compiler.
4. A page opens with different options in it showing the project workspace at the leftmost corner side, output window in the bottom and an ash coloured space for the program to be written.
5. Now to start using the keil, click on the option “project”.
6. A small window opens showing the options like new project, import project, open project etc. Click on “New project”.
7. A small window with the title bar “Create new project” opens. The window asks the user to give the project name with which it should be created and the destination location. The project can be created in any of the drives available. You can create a new folder and then a new file or can create directly a new file.
8. After the file is saved in the given destination location, a window opens where a list of vendors will be displayed and you have to select the device for the target you have created.

9. The most widely used vendor is Atmel. So click on Atmel and now the family of microcontrollers manufactured by Atmel opens. You can select any one of the microcontrollers according to the requirement.

10. When you click on any one of the microcontrollers, the features of that particular microcontroller will be displayed on the right side of the page. The most appropriate microcontroller with which most of the projects can be implemented is the AT89C51. Click on this microcontroller and have a look at its features. Now click on “OK” to select this microcontroller.
11. A small window opens asking whether to copy the startup code into the file you have created just now. Just click on “No” to proceed further.
12. Now you can see the TARGET and SOURCE GROUP created in the project workspace.
13. Now click on “File” and in that “New”. A new page opens and you can start writing program in it.
14. After the program is completed, save it with any name but with the .asm extension. Save the program in the file you have created earlier.
15. You can notice that after you save the program, the predefined keywords will be highlighted in bold letters.
16. Now add this file to the target by giving a right click on the source group. A list of options open and in that select “Add files to the source group”. Check for this file where you have saved and add it.
17. Right click on the target and select the first option “Options for target”. A window opens with different options like device, target, output etc. First click on “target”.
18. Since the set frequency of the microcontroller is 11.0592 MHz to interface with the PC, just enter this frequency value in the Xtal (MHz) text area and put a tick on the Use on-chip ROM. This is because the program what we write here in the keil will later be dumped into the microcontroller and will be stored in the inbuilt ROM in the microcontroller.

19. Now click the option “Output” and give any name to the hex file to be created in the “Name of executable” text area and put a tick to the “Create HEX file” option present in the same window. The hex file can be created in any of the drives. You can change the folder by clicking on “Select folder for Objects”.
20. Now to check whether the program you have written is errorless or not, click on the icon exactly below the “Open file” icon which is nothing but Build Target icon. You can even use the shortcut key F7 to compile the program written.
21. To check for the output, there are several windows like serial window, memory window, project window etc. Depending on the program you have written, select the appropriate window to see the output by entering into debug mode.
22. The icon with the letter “d” indicates the debug mode.
23. Click on this icon and now click on the option “View” and select the appropriate window to check for the output.
24. After this is done, click the icon “debug” again to come out of the debug mode.
25. The hex file created as shown earlier will be dumped into the microcontroller with the help of another software called Proload.

PROLOAD:

Proload is a software which accepts only hex files. Once the machine code is converted into hex code, that hex code has to be dumped into the microcontroller placed in the programmer kit and this is done by the Proload. Programmer kit contains a microcontroller on it other than the one which is to be programmed. This microcontroller has a program in it written in such a way that it accepts the hex file from the keil compiler and dumps this hex file into the microcontroller which is to be programmed. As this programmer kit requires power supply to be operated, this power supply is given from the power supply circuit designed above. It should be noted that this programmer kit contains a power supply section in the board itself but in order to switch on that power supply, a source is required. Thus this is accomplished from the power supply board with an output of 12volts or from an adapter connected to 230 V AC.

1. Install the Proload Software in the PC.
2. Now connect the Programmer kit to the PC (CPU) through serial cable.
3. Power up the programmer kit from the ac supply through adapter.
4. Now place the microcontroller in the GIF socket provided in the programmer kit.
5. Click on the Proload icon in the PC. A window appears providing the information like Hardware model, com port, device type, Flash size etc. Click on browse option to select the hex file to be dumped into the microcontroller and then click on “Auto program” to program the microcontroller with that particular hex file.
6. The status of the microcontroller can be seen in the small status window in the bottom of the page.
7. After this process is completed, remove the microcontroller from the programmer kit and place it in your system board. Now the system board behaves according to the program written in the microcontroller.

Advantages:

- Easy to operate
- Sophisticated security
- Simple and Reliable Design
- Isolates both GSM and GPS signal

Applications:

- Automotives
- Logistics
- Personal Cars
- Public Transportation Vehicles
- Trains and Locomotives

CONCLUSION

This project presents vehicle accident detection and alert system with SMS to the user defined mobile numbers. The GPS tracking and GSM alert based algorithm is designed and implemented with LPC2148 Microcontroller in embedded system domain. The proposed Vehicle accident detection system can track geographical information automatically and sends an alert SMS on accident condition. Experimental work has been carried out carefully. The result shows that higher sensitivity and accuracy is indeed achieved using the project. A keypad is also provided to enter the user defined mobile numbers of his choice. EEPROM is interfaced to store the mobile numbers permanently. This made the project more user-friendly and reliable. The proposed method is verified to be highly beneficial for the automotive industry.