DEDICATED TO OUR
LOVABLE PARENTS

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE-641006

(Affiliated to Bharathiyar University)

CERTIFICATE

This is to certify that the Project Report entitled

USB COMMUNICATION

has been submitted by

....

In partial fulfillment for the award of the degree of Bachelor of Engineering in

ELECTRONICS AND COMMUNICATION ENGINEERING

Of the Bharathiar University, Coimbatore – 641006
during the academic year 2002-2003

.... 7.3.03
Faculty Guide

....
Head of the Department

Certified that the Candidate was examined by us in the project work viva-voce Examination held on................ and the University Register Number was ....

....
Internal Examiner

18/3/03
....
External Examiner

Statex

Statex Electronics

f'. SEL/2906

CERTIFICATE

TO WHOM SOEVER IT MAY CONCERN

his is to certify that the following Final Year B.E "Electronics nd communication " students of Kumaraguru College of Technology, oimbatore have successfully completed the project in our Organiation.

1. V.ARUN
2. D.MAHESH
3. V.MAHESHWARAN
4. N.D.PRABHU

Project Title: "USB COMMUNICATION"

Project Period: July 2002 to March 2003.

During this project, their attendance, conduct and behaviour were found to be good.

for STATEX ELECTRONICS,

15/3/2003

Authorised Signatory.

ACKNOWLEDGEMENT

"The secret of success is to put your best forward, nurtured at every step by the great men who give the sense of direction."

We thank our Principal Dr.K.K. Padmanabhan B.Sc(Engg), M.Tech, Ph.D and the Management for the facilities provided in the college to accomplish this project.

We place on record our deep sense of gratitude to our Head of the Department Prof. Muthuraman Ramasamy, M.E, MISTE, MIE, C.ENG(I), MBMESI, MIEEE(USA) for his invaluable advice and gentle reminders, that really motivated us through many a tough encounter, only to help us reach our goal.

We are greatly indebted to our guide Prof. S. Govindaraju, M.E, M.I.S.T.E. for his invaluable advice gentle reminders, that really motivated us through many a tough encounter, only to help us reach our goal.

We express our bountiful thanks to Mr. Doraiswamy, Vice President, Statex Engineering who opened the doors of opportunity and paved the way for our project, in spite of his busy schedule.

We thank our parents for their moral support and encouragement. We also take this opportunity to thank our friends and our non-teaching staff who have always lend their helping hands to us .

SYNOPSIS

SYNOPSIS

The Textile Industry with all the blooming ventures has many an advanced machines. All the Textile Testing Equipments that are being currently manufactured by the company has its own setup for measuring the required parameters, an in-built processor and a computer to display the parameter values. The customers felt that it is costly to have a computer for a single testing equipment. So, in order to satisfy the customers and with other computer and peripheral manufacturers supporting USB and with modern PCs being shipped with USB ports and all peripheral manufacturers using USB ports on their products, shifted its attention from serial communication towards USB communication.

In this approach, a project to design an interface board to interface multiple textile testing equipments with an RS232 output standard to a single USB port. The project has been implemented using an ATMEL 89C51 microcontroller chip and a USB function controller chip, USBN 9603. This approach works out to be economical than its alternatives.

CONTENTS

INDEX

INTRODUCTION

COMPANY PROFILE

STATEX ENGINEERING (P) LTD. was principally established in the year 1985 to supply high technical Textile Testing Instruments. STATEX provides technical services to all customers trough their service centres in Ahmedabad, Mumbai, Chandigarh and Coimbatore. Its group companies includes Statex Engineering, Dhansu Engineering (p) ltd., Statex Exporters, Statex Electronics and Statex Mills. More than 400 employees highly qualified in technical and commercial aspects are engaged to meet International standards. The company is managed by MR.K.Subramaniam as Managing Director.

STATEX covers a modest assesment of the total indian market in Research Organisations, Educational Institutions, Private/Public sector companies and Government agencies.STATEX ELECTRONICS, established in the year 1987 specially for Electronic Instruments in full fledged backup of R&D base with a team of qualified Engineers working throughout the year and strictly confirming to the quality and upgradation of technology at par with International standard at an affordable price. STATEX proudly exhibited their inventions and Instruments in various Trade Fairs.

In recognition of the excellent reputation in the exports of Textile Testing Instruments and Positive Clearers and its quality, the Organisation has been awarded the 'Vijaya Shree Award' by the Honourable Ministry of Industries.

INTRODUCTION TO USB

Universal Serial Bus is a likely solution any time a user wants to use a computer to communicate with devices outside the computer. The interface is suitable for one-of-kind and small-scale designs as well as mass-produced, standard peripherals. It has features that please both the users who run applications that access the peripherals and the developers who design the hardware and write the code that communicates with the interface.

From the user's perspective, the benefits to USB are ,

- Ease of Use:

  USB is a versatile enough to be usable with many kinds of peripherals. Instead of having a different connector and protocols for each peripheral, one interface serves many.

- Easy to connect:

  With USB, there's no need to open the computer's enclosure to add an expansion card for each peripheral. A typical PC has two ports.

- Simple cables:

  Cables as long as 5 meters can be used and with hubs, a link can extend as far as 30 meters.

- Speed:

  A full-speed USB interface communicates at 12 Megabits per second.

- Low Cost::

  Even though USB is more complex than other interfaces, its components and cables are inexpensive.

- Low power consumption:

  Power-saving circuits and code automatically power down USB peripherals when not in use, thus consuming less power.

Its other features are,

Interface	:	USB
Format	:	asynchronous serial
Number of devices	:	127
Length	:	16 feet (or up to 96 ft. with 5 hubs)
Speed	:	1.5M, 12M (480M in version 2.0)

It also have many advantages for developers. They are,

- Flexibility:

  The USB ‘s four transfer types and two speeds make it feasible for many types of peripherals. It simply supports the defined device requirements and protocols.

- Operating System Support:

  Windows operating system from version Windows 98 and all its successors support USB.

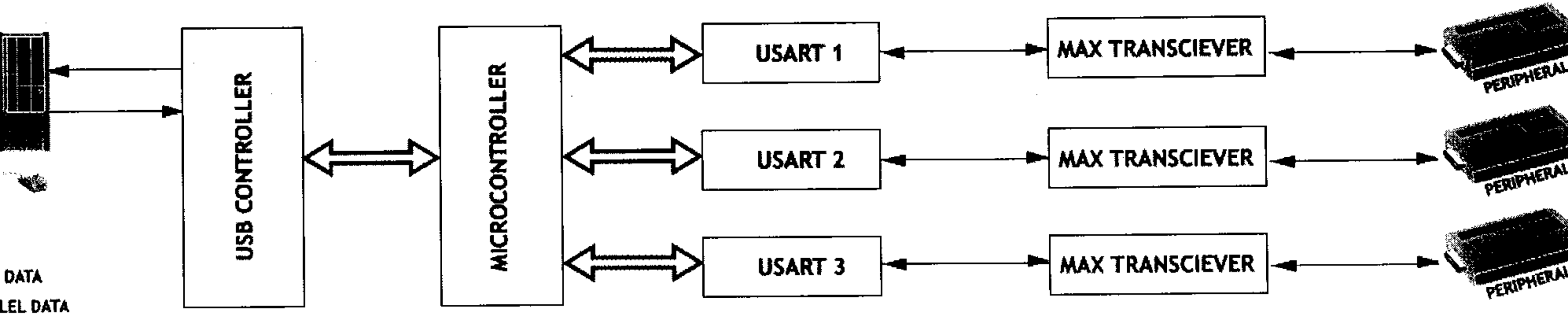
- Minimal Fees:

  Unlike other interfaces, the USB specification and related documents are available free online. Anyone can develop USB software without paying a licensing fee.

PROJECT DESCRIPTION:

In this project "USB COMMUNICATION" two different serial transmission standards are interfaced. On the peripheral side RS232 standard is used and on the PC side it is USB standard. A MAX232 IC is used to convert these signals into TTL voltage levels. The serial data is then converted into parallel data format using a USART 8251A IC. The data is then stored inside the RAM of a micro controller AT89C51.

This micro controller is used to control the entire operations of the interface board. The data is then routed into a USB Function Controller IC that converts the data into USB format. The data is then routed to the PC through a USB cable. In a similar manner control information in the form of data is moved from PC to peripheral.


DATA
LEL DATA
USB CONTROLLER
MICROCONTROLLER
USART 1
USART 2
USART 3
MAX TRANSCIEVER
MAX TRANSCIEVER
MAX TRANSCIEVER
PERIPHERAL
PERIPHERAL
PERIPHERAL

HARDWARE DIAGRAM

HARDWARE DESCRIPTION

HARDWARE DESCRIPTION

The objective of this project is to interface three peripherals to a single computer system through USB port. Here the peripherals are Textile testing equipments transferring data at the rate of 2400 bauds. The PC first sends a control information in the form of data to the required peripherals, requesting the peripherals to start the testing procedure. Now the peripherals begin to transfer the results i.e., data to the PC. The output from the peripherals is of RS232 standard and it is required to convert them into TTL outputs in order to be compatible with USB voltage levels. A MAX232 IC is used to perform this task. The RS232 output from the peripheral is given as input to the MAX232 IC and the output from MAX232 is the input to the next stage.

In a similar manner, the control data output by the PC, which is compatible with TTL voltage level is given as another input to this MAX232. The other output of MAX232 is given as input to the peripherals.

The data from the peripherals after being passed through a MAX232 is fed as input to a 8251A USART, through its RxD pin. The 8251A converts the serial data from MAX232 to a parallel data. Here the handshaking signals of USART are not used as required by the company. When a character is written to the buffer register, the RxRDY pin goes high and this is given as an input to a single port pin of 89c51. In a similar manner, the RxRDY pin of the other two 8251A ICs is given as separate input to two other port pins.

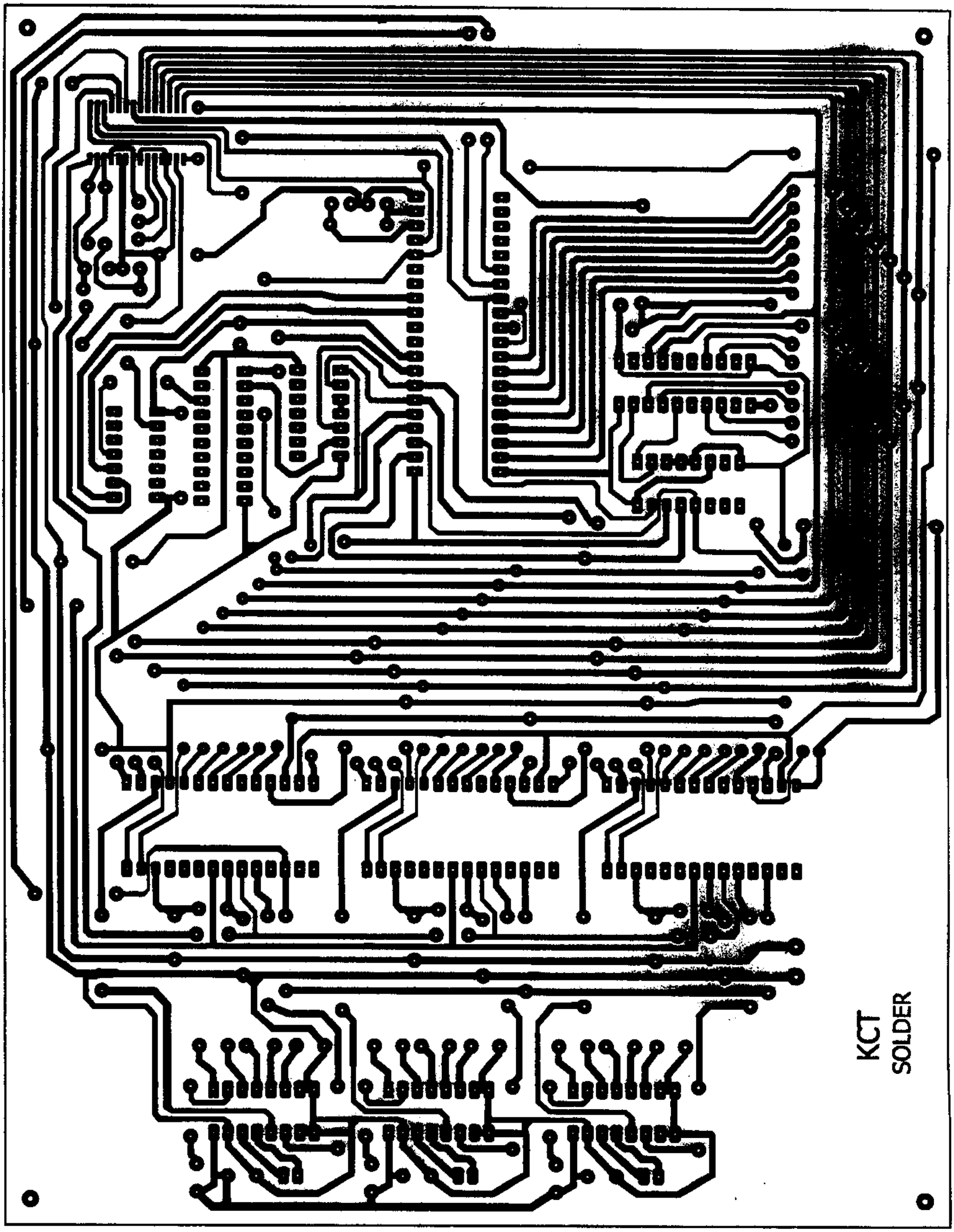
When the data from the PC is to be transmitted to the MAX232 IC it has to be transmitted through its TxD line. The RxRDY output of the three 8251A ICs is logically ORed using a 7432 OR gate IC. The output of this OR gate is given as input to a 20 pin octal buffer 74LS244. The output of the buffer is complemented using a 7404 NOT gate IC and it is used to interrupt the microcontroller by feeding the above output as an input to the second pin of port 3 INT0. The ENABLE pin of latch is connected to pin 8 of 89c51. This pin is addressed using software and is used to disable the latch whenever a 8251A has interrupted the microcontroller. The control or data word selection of 8251A is done through software by addressing the third pin of port 1.

The three 8251As are selected by assigning an 8-bit address to them initially. Whenever a particular 8251A is to be selected its corresponding address is output through the port 3 pins. The input of the latch is connected to the output of port 3 .The output of this latch is given as the input to 3x8 decoder, whose output is used to chip select the USART. The latch is enabled by connecting the ALE pin of 89c51 to ENABLE pin of the latch. The two most significant bits of the address field is used as decoder logic for chip select.

The data from 8251A is in parallel format and is stored inside the RAM location of 89C51. The RAM location of 89C51 is divided into four locations, three to store data from three 8251A separately and one to store the data from PC. The Reset pins of all IC's are configured to reset on power up. The oscillator circuit for 89C51 is connected between XTAL1 and XTAL2 and the capacitors for MAX232 are also connected.

The USBN9603 is selected to operate in Multiplexed mode by connecting pin 25 to Vcc and pin 24 to GnD. The USB chip is selected using the decoder select logic. The ALE pins of USBN9603 and 89C51 are connected. The pins V3.3 and AGND are connected to ground. The oscillator circuit is connected between the pins XIN and XOUT. The differential pins D+ and D- are connected to the USB cable from PC. The Vcc pin of USB cable is connected to the Vcc pin of USB function controller. The data pins of all IC's are connected together. The power supply for all IC's are given using an external power supply of 5V.

PCB DESIGN


KCT
SOLDER

USBN9603 Universal Serial Bus Full Speed Node Controller

General Description

The USBN9603is an integrated, USB Node controllers The device provides enhanced DMA support with many automatic data handling features. It is compatible with USB specification versions 1.0 and 1.1, and is an advanced version of the USBN9602. The device integrates the required USB transceiver with a 3.3V regulator, a Serial Interface Engine (SIE), USB endpoint (EP) FIFOs, a versatile 8-bit parallel interface, a clock generator and a MICROWIRE/PLUS™ interface.

Seven endpoint pipes are supported: one for the mandatory control endpoint and six to support interrupt, bulk and isochronous endpoints. Each endpoint pipe has a dedicated FIFO, 8 bytes for the control endpoint and 64 bytes for the other endpoints. The 8-bit parallel interface supports multiplexed and non-multiplexed style CPU address/data buses. A programmable interrupt output scheme allows device configuration for different interrupt signaling requirements.

Features

- Full-speed USB node device.
- Integrated USB transceiver.
- Supports 24 MHz oscillator circuit with internal 48 MHz clock generation circuit.
- Programmable clock generator.

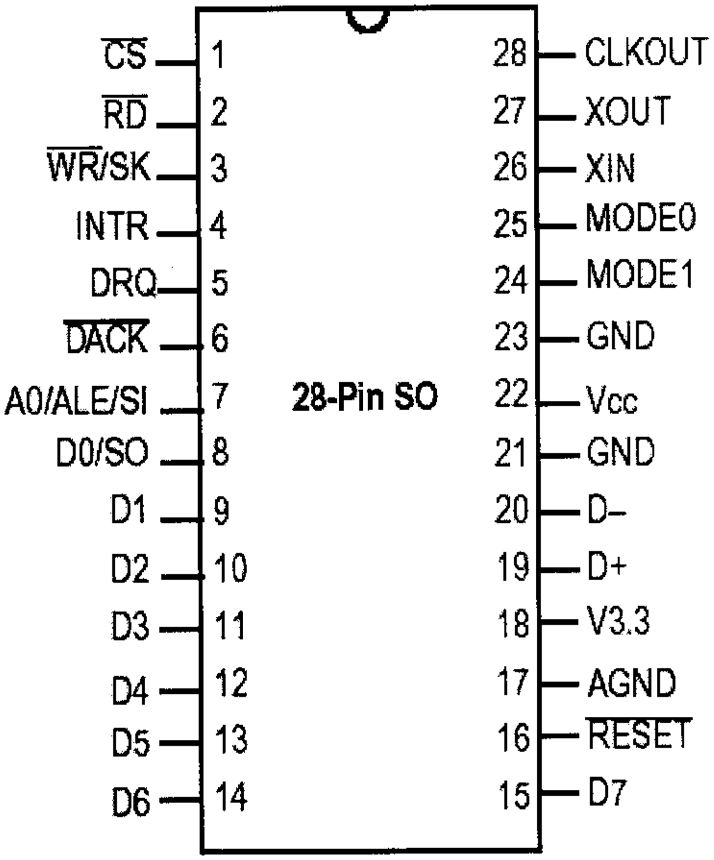
- Serial Interface Engine (SIE) consisting of Physical Layer Interface (PHY) and Media Access Controller (MAC), USB Specification 1.0 and 1.1 compliant.
- Control/Status register file.
- USB Function Controller with seven FIFO-based Endpoints:
  - — One bidirectional Control Endpoint 0 (8 bytes).
  - — Three Transmit Endpoints (64 bytes each).
  - — Three Receive Endpoints (64 bytes each).
- 8-bit parallel interface with two selectable modes:
  - — Non-multiplexed.
  - — Multiplexed (Intel compatible).

Enhanced DMA support:

- — Automatic DMA (ADMA) mode for fully CPU-independent transfer of large bulk or ISO packets.
- — DMA controller, together with the ADMA logic, can transfer a large block of data in 64-byte packets via the USB.
- — Automatic Data PID toggling/checking and NAK packet recovery (maximum 256x64 bytes of data =16K bytes)

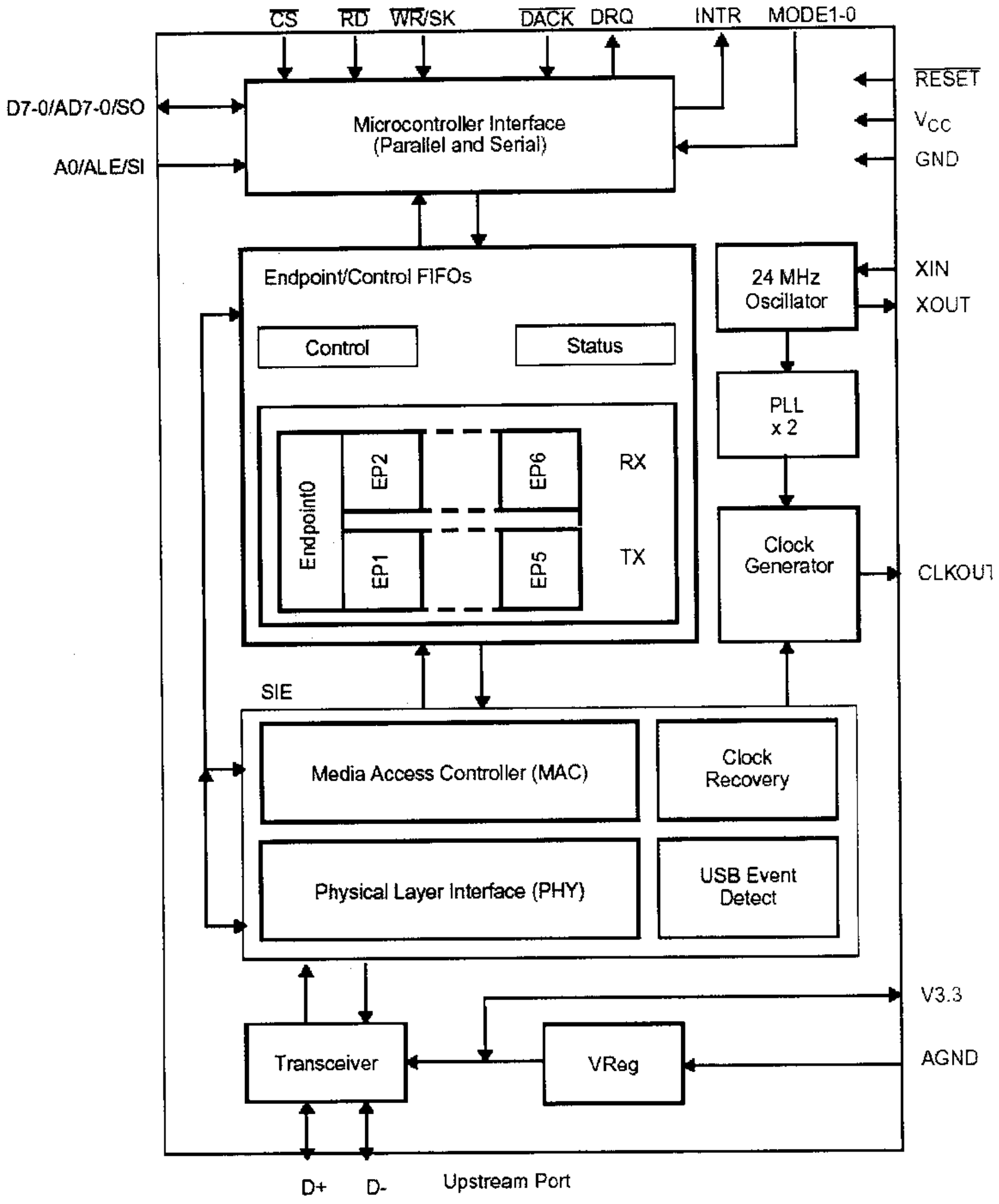
- MICROWIRE/PLUS interface
- Low EMI, low standby current.
- Fully static HALT mode with asynchronous wake-up for bus powered operation
- 5V or 3.3V operation
- Improved input range 3.3V signal voltage regulator
- Power-up reset and startup delay counter simplify system design

PIN DIAGRAM:



USBN9603

BLOCK DIAGRAM



PIN DESCRIPTION

Power Supply:

Vcc-Digital Power Supply:

Power-on reset is detected when the input voltage is at the same level as GND and then raised to the required Vcc level. The power-on reset causes all registers to be set to their reset values, the clock generator to be reset and stalls the CLKOUT output for 214 XIN clock cycles. During this time, no internal register is accessible.

GND-Digital Power Supply:

Ground pin.

AGND-Analog Power Supply:

Ground pin.

V3.3-Transceiver 3.3V Voltage Supply:

This pin can be used as the internal 3.3V voltage regulator output. The regulator is intended to power only the internal transceiver and one external pull-up. An external 1 mF de-coupling capacitor is required on this pin. The voltage regulator output is disabled upon reset. When the internal voltage regulator is left disabled, this pin must be used as a 3.3V supply input for the internal transceiver. This is the case during 3.3V operation.

Oscillator, Clock and Reset:

XIN-Crystal Oscillator Input:

Input for internal 24 MHz crystal oscillator circuit. A 24 MHz fundamental crystal may be used.

XOUT-Crystal Oscillator Output

CLKOUT-Clock Output:

This programmable clock output may be disabled and configured for different speeds via the Clock Configuration register. After a power-on reset and hardware reset (assertion of RESET), a 4 MHz clock signal is output (there may be an initial phase discontinuity). In the USBN9604, a hardware reset causes CLKOUT to stall for 214 XIN clock cycles while the internal DLL is synchronized to the external reference clock.

RESET-Reset:

Active low, assertion of RESET indicates a hardware reset, which causes all registers in the device to revert to their reset values.In the USBN9604, the hardware reset action is identical to a power-on reset. Signal conditioning is provided on this input to allow use of a simple, RC power-on reset circuit.

USB Port:

D+-USB D+ Upstream Port:

This pin requires an external 1.5k pull-up to 3.3V to signal full speed operation.

D– -USB D– Upstream Port

Microprocessor Interface:

MODE1-0- Interface Mode:

Each of these pins should be hard-wired to VCC or GND to select the interface mode:

MODE1-0 = 00. Mode 0: Non-multiplexed parallel interface mode

MODE1-0 = 01. Mode 1: Multiplexed parallel interface mode

MODE1-0 = 10. Mode 2: MICROWIRE interface mode

MODE1-0 = 11. Mode 3: Reserved

DACK-DMA Acknowledge:

This active low signal is only used if DMA is enabled. If DMA is not used, this pin must be tied to VCC.

DRQ-DMA Request:

This pin is used for DMA request only if DMA is enabled.

INTR-Interrupt:

The interrupt signal modes (active high, active low or open drain) can be configured via the Main Control register. During reset, this signal is TRI-STATE.

CS-Chip Select:

Active low chip select.

RD-Read:

Active low read strobe, parallel interface.

WR-Write:

Active low write strobe, parallel interface.

SK-MICROWIRE Shift Clock: Mode 2

A0-A0 Address Bus Line: Mode 0, parallel interface

ALE-Address Latch Enable: Mode 1, parallel interface

SI-MICROWIRE Serial Input: Mode 2

D0-Data Bus Line D0: Mode 0

AD0-Address/Data Bus LIne AD0: Mode 1

SO-MICROWIRE Serial Output: Mode 2

D1-Data Bus Line D1: Mode 0

AD1-Address/Data Bus Line AD1: Mode 1

D2-Data Bus Line D2: Mode 0

AD2-Address/Data Bus Line AD2: Mode 1

D3-Data Bus Line D3: Mode 0

AD3-Address/Data Bus Line AD3: Mode 1

D4-Data Bus Line D4: Mode 0

AD4-Address/Data Bus Line AD4: Mode 1

D5-Data Bus Line D5: Mode 0

AD5-Address/Data Bus Line AD5: Mode 1

D6-Data Bus Line D6: Mode 0

AD6-Address/Data Bus Line AD6: Mode 1

D7-Data Bus Line D7: Mode 0

AD7-Address/Data Bus Line AD7: Mode 1

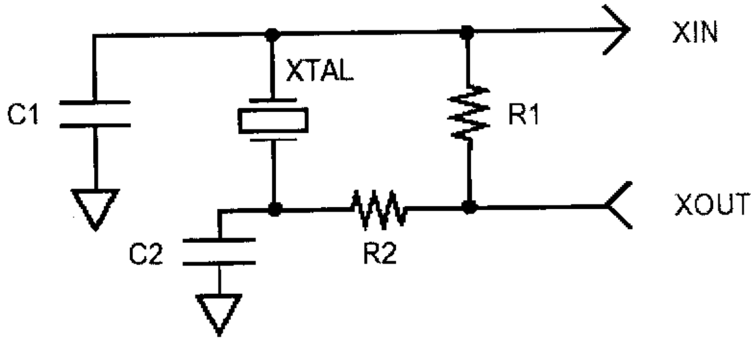
Oscillator Circuit:

The XIN and XOUT pins may be connected to make a 24 MHz closed-loop, crystal-controlled oscillator. Alternately, an external 24 MHz clock source may be used as the input clock for the device. The internal crystal oscillator uses a 24 MHz fundamental crystal. If an external clock source is used, it is connected to XIN. XOUT should remain unconnected. Stray capacitance and inductance should be kept as low as possible in the oscillator circuit. Trace lengths should be minimized by positioning the crystal and external components as close as possible to the XIN and XOUT pins.

External Elements:

Choose C1 and C2 capacitors to match the crystal's load capacitance. The load capacitance CL "seen" by the crystal is comprised of C1 in series with C2, and in parallel with the parasitic capacitance of the circuit. The parasitic capacitance is caused by the chip package, board layout and socket (if any), and can vary from 0 to 8 pF. The rule of thumb in choosing these capacitors is:

CL = (C1*C2)/(C1+C2)+C Parasitic



OSCILLATOR CIRCUIT

Functional Overview

The device is a Universal Serial Bus (USB) Node controller compatible with USB Specification, 1.0 and 1.1. It integrates onto a single IC the required USB transceiver with a 3.3V regulator, the Serial Interface Engine (SIE), USB endpoint FIFOs, a versatile (8-bit parallel or serial) interface and a clock generator. A total of seven endpoint pipes are supported: one unidirectional for the mandatory control EP0 and an additional six for unidirectional endpoints to support USB interrupt, bulk and isochronous data transfers. The 8-bit parallel interface supports multiplexed and non-multiplexed style CPU address/data buses. The synchronous serial MICROWIRE interface allows adapting to CPUs without external address/data buses. A programmable interrupt output scheme allows adapting to different interrupt signaling requirements.

TRANSCEIVER:

The device contains a high-speed transceiver which consists of three main functional blocks:

- Differential receiver
- Single-ended receiver with on-chip voltage reference
- Transmitter with on-chip current source.

To minimize signal skew, the differential output swings of the transmitter are well balanced. Slew-rate control is used on the driver to minimize radiated noise and crosstalk. The drivers support TRI-STATE operation to allow bidirectional, half-duplex operation of the transceiver.

The differential receiver operates over the complete common mode range, and has a delay guaranteed to be larger than that of the single-ended receivers. This avoids potential glitches in the Serial Interface Engine (SIE) after single-ended zeros.

Single-ended receivers are present on each of the two data lines. These are required, in addition to the differential receiver, to detect an absolute voltage with a switching threshold between 0.8V and 2.0V (TTL inputs). To increase Vcc rejection, without glitching, a voltage reference sets the single-ended switching reference. An external 1.5 +5% K ohm or -5%K ohm resistor is required on D+ to indicate that this is a high-speed node. This resistor should be tied to a voltage source between 3.0V and 3.6V, and referenced to the local ground, such as the output provided on pin V3.3.

VOLTAGE REGULATOR (VREG):

The voltage regulator provides 3.3V for the integrated transceiver from 5.0V device power or USB bus power. This output can be used to supply power to the 1.5 K ohm pull-up resistor. This output must be decoupled with a 1 micro Farad tantalum capacitor to ground. It can be disabled under software control to allow using the device in a 3.3V system.

SERIAL INTERFACE ENGINE (SIE):

The SIE is comprised of physical (PHY) and Media Access Controller (MAC) modules. The PHY module includes the digital clock recovery circuit, a digital glitch filter, End Of Packet (EOP) detection circuitry, and bit stuffing and unstuffing logic. The MAC module includes packet formatting, CRC generation and checking, and endpoint address detection. It provides the necessary control to give the NAK, ACK and STALL responses as determined by the Endpoint Pipe Controller (EPC) for the specified endpoint pipe. The SIE is also responsible for detecting and reporting USB-specific events, such as

NodeReset, NodeSuspend and NodeResume. The module output signals to the transceiver are well matched (under 1 nS) to minimize skew on the USB signals.

The USB specifications assign bit stuffing and unstuffing as the method to ensure adequate electrical transitions on the line to enable clock recovery at the receiving end. The bit stuffing block ensures that whenever a string of consecutive 1's is encountered, a 0 is inserted after every sixth 1 in the data stream. The bit unstuffing logic reverses this process.

The clock recovery block uses the incoming NRZI data to extract a data clock (12 MHz) from a 48 MHz input clock. This input clock is derived from a 24 MHz oscillator in conjunction with PLL circuitry (clock doubler). This clock is used in the data recovery circuit. The output of this block is binary data (decoded from the NRZI stream) which can be appropriately sampled using the extracted 12 MHz clock.

ENDPOINT PIPE CONTROLLER (EPC):

The EPC provides the interface for USB function endpoints. An endpoint is the ultimate source or sink of data. An endpoint pipe facilitates the movement of data between USB and memory, and completes the path between the USB host and the function endpoint. According to the USB specification, up to 31 such endpoints are supported at any given time. USB allows a total of 16 unidirectional endpoints for receive and 16 for transmit. As the control endpoint 0 is always bidirectional, the total number is 31. Seven endpoint pipes with the same function address are supported.

A USB function is a USB device that is able to transmit and receive information on the bus. A function may have one or more configurations, each of which defines the interfaces that make up the device. Each interface, in turn, is composed of one or more endpoints.

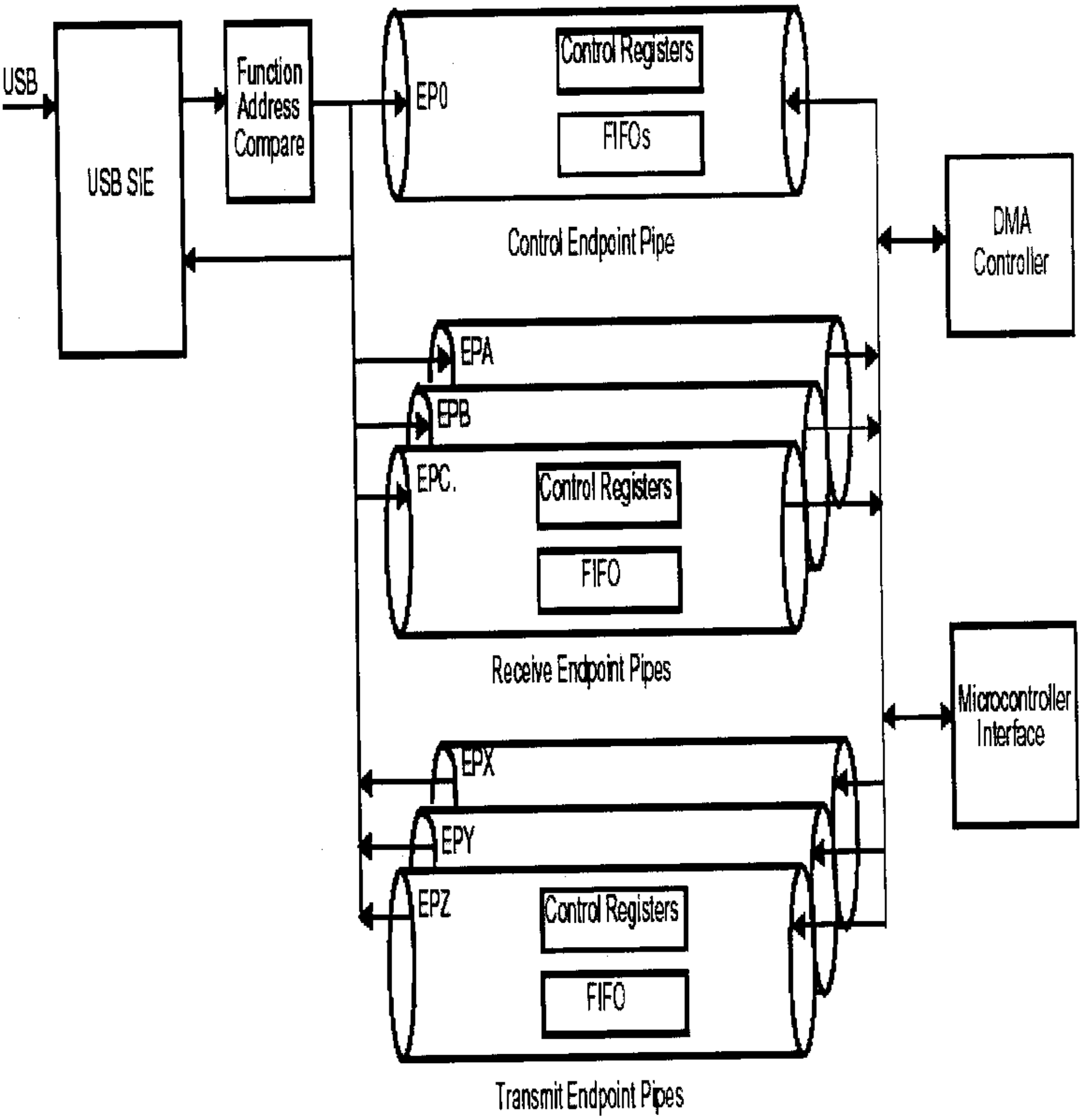
Each endpoint is an addressable entity on USB and is required to respond to IN and OUT tokens from the USB host (typically a PC). IN tokens indicate that the host has requested to receive information from an endpoint, and OUT tokens indicate that it is about to send information to an endpoint.

On detection of an IN token addressed to an endpoint, the endpoint pipe should respond with a data packet. If the endpoint pipe is currently stalled, a STALL handshake packet is sent under software control. If the endpoint pipe is enabled but no data is present, a NAK (Negative Acknowledgment) handshake packet is sent automatically. If the endpoint pipe is isochronous

and enabled but no data is present, a bit stuff error followed by an end of packet is sent on the bus.

Similarly, on detection of an OUT token addressed to an endpoint, the endpoint pipe should receive a data packet sent by the host and load it into the appropriate FIFO. If the endpoint pipe is stalled, a STALL handshake packet is sent. If the endpoint pipe is enabled but no buffer is present for data storage, a NAK handshake packet is sent. If the endpoint is isochronous and enabled but cannot handle the data, no handshake packet is sent.

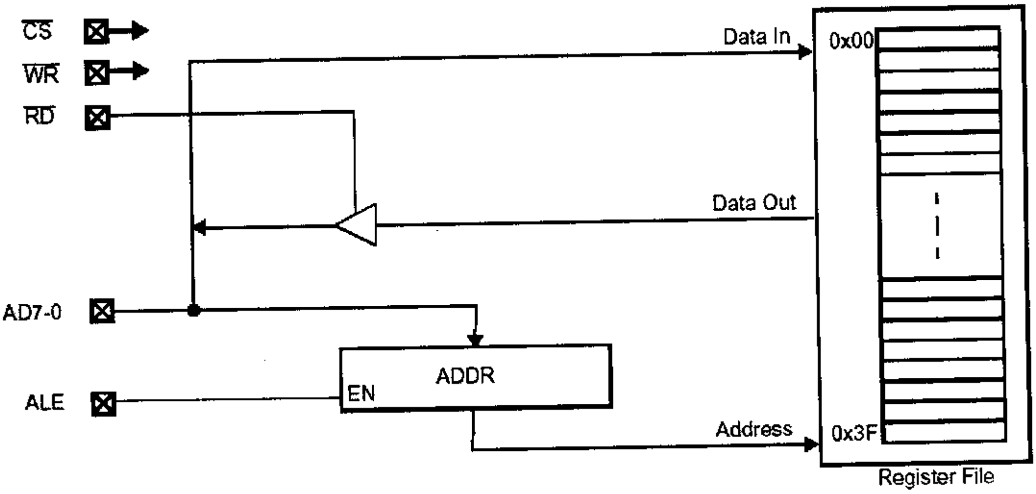
A disabled endpoint does not respond to IN, OUT, or SETUP tokens. The EPC maintains separate status and control information for each endpoint pipe. For IN tokens, the EPC transfers data from the associated FIFO to the host. For OUT tokens, the EPC transfers data in the opposite direction.



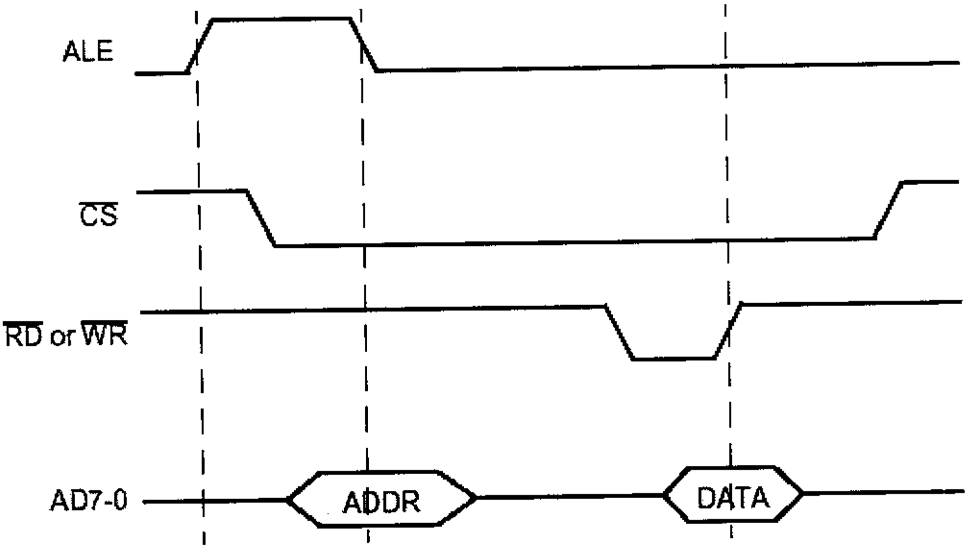
ENDPOINT PIPE CONTROLLER (EPC) OPERATION

MULTIPLEXED MODE

The device can be connected to a CPU or microcontroller via the 8-bit parallel or MICROWIRE interface. The interface type is selected by the input mode pins MODE0 and MODE1. In addition, a configurable interrupt output is provided. The interrupt type can be configured to be either open-drain active-low or push-pull active high or low. Multiplexed mode uses the control pins CS, RD, WR, the address latch enable signal ALE and the bidirectional address data bus AD7-0. This mode is selected by tying MODE1 to GND and MODE0 to VCC. The address is latched into the ADDR register when ALE is high. Data is output/input with the next active RD or WR signal. All registers are directly accessible in this interface mode.



MULTIPLEXED MODE BLOCK DIAGRAM

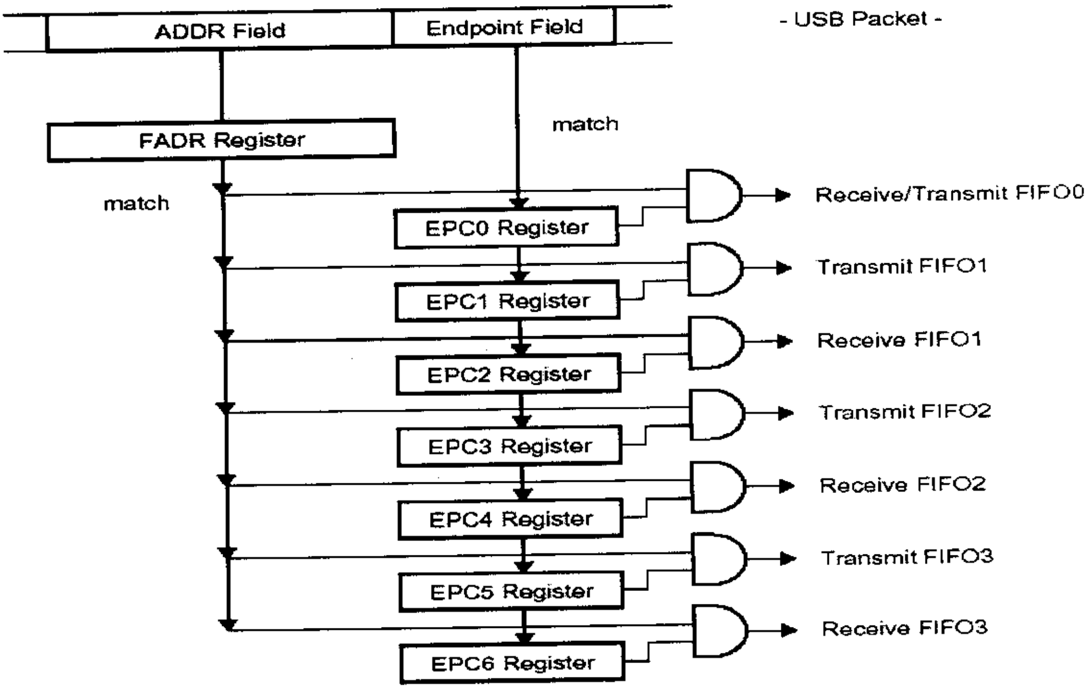


MULTIPLEXED MODE TIMING DIAGRAM

ENDPOINT OPERATION

Address Detection:

Packets are broadcast from the host controller to all the nodes on the USB network. Address detection is implemented in hardware to allow selective reception of packets and to permit optimal use of microcontroller bandwidth. One function address with seven different endpoint combinations is decoded in parallel. If a match is found, then that particular packet is received into the FIFO; otherwise it is ignored. The incoming USB Packet Address field and Endpoint field are extracted from the incoming bit stream. Then the address field is compared to the Function Address register (FADR). If a match is detected, the Endpoint field is compared to all of the Endpoint Control registers (EPCx) in parallel. A match then causes the payload data to be received or transmitted using the respective endpoint FIFO.



USB FUNCTION ADDRESS/ENDPOINT DECODING

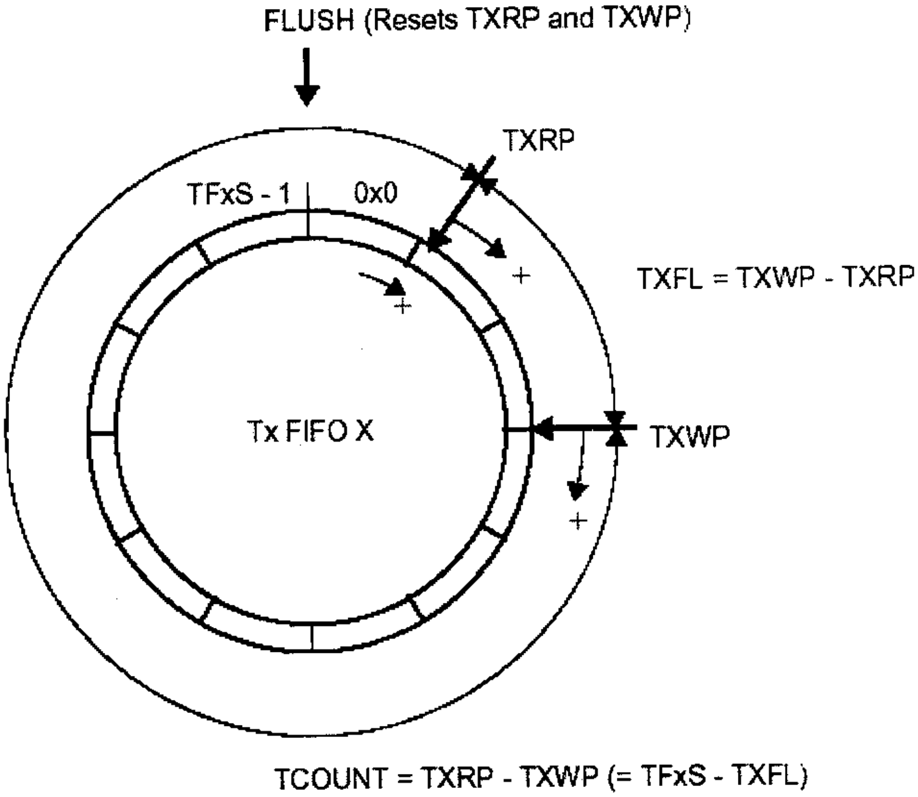
Transmit and Receive Endpoint FIFOs:

The device uses a total of seven transmit and receive FIFOs: one bidirectional transmit and receive FIFO for the mandatory control endpoint, three transmit FIFOs and three receive FIFOs. As shown in Table 4, the bidirectional FIFO for the control endpoint is 8 bytes deep. The additional unidirectional FIFOs are 64 bytes each for both transmit and receive. Each FIFO can be programmed for one exclusive USB endpoint, used together with one globally decoded USB function address. The firmware must not enable both transmit and receive FIFOs for endpoint zero at any given time.

Endpoint No.	TX FIFO		RX FIFO	
	Size (Bytes)	Name	Size (Bytes)	Name
0	8 FIFO0			
1	64	TXFIFO1		
2			64	RXFIFO1
3	64	TXFIFO2		
4			64	RXFIFO2
5	64	TXFIFO3		
6			64	RXFIFO3

Transmit Endpoint FIFO Operation (TXFIFO1, TXFIFO2, TXFIFO3):

The Transmit FIFOs for Endpoints 1, 3 and 5 support bulk, interrupt and isochronous USB packet transfers larger than the actual FIFO size. Therefore. the firmware must update the FIFO contents while the USB packet is transmitted on the bus.


FLUSH (Resets TXRP and TXWP)
TXRP
TFxS - 1
0x0
+
+
TXFL = TXWP - TXRP
Tx FIFO X
TXWP
+
TCOUNT = TXRP - TXWP (= TFxS - TXFL)

TFxS:

Transmit FIFO x Size. This is the total number of bytes available within the FIFO.

TXRP:

Transmit Read Pointer. This pointer is incremented every time the Endpoint Controller reads from the transmit FIFO. This pointer wraps around to zero if TFxS is reached. TXRP is never incremented beyond the value of the write pointer TXWP. An underrun condition occurs if TXRP equals TXWP and an attempt is made to transmit more bytes when the LAST bit in the TXCMDx register is not set.

TXWP:

Transmit Write Pointer. This pointer is incremented every time the firmware writes to the transmit FIFO. This pointer wraps around to zero if TFxS is reached. If an attempt is made to write more bytes to the FIFO than actual space available (FIFO overrun), the write to the FIFO is ignored. If so, TCOUNT is checked for an indication of the number of empty bytes remaining.

TXFL:

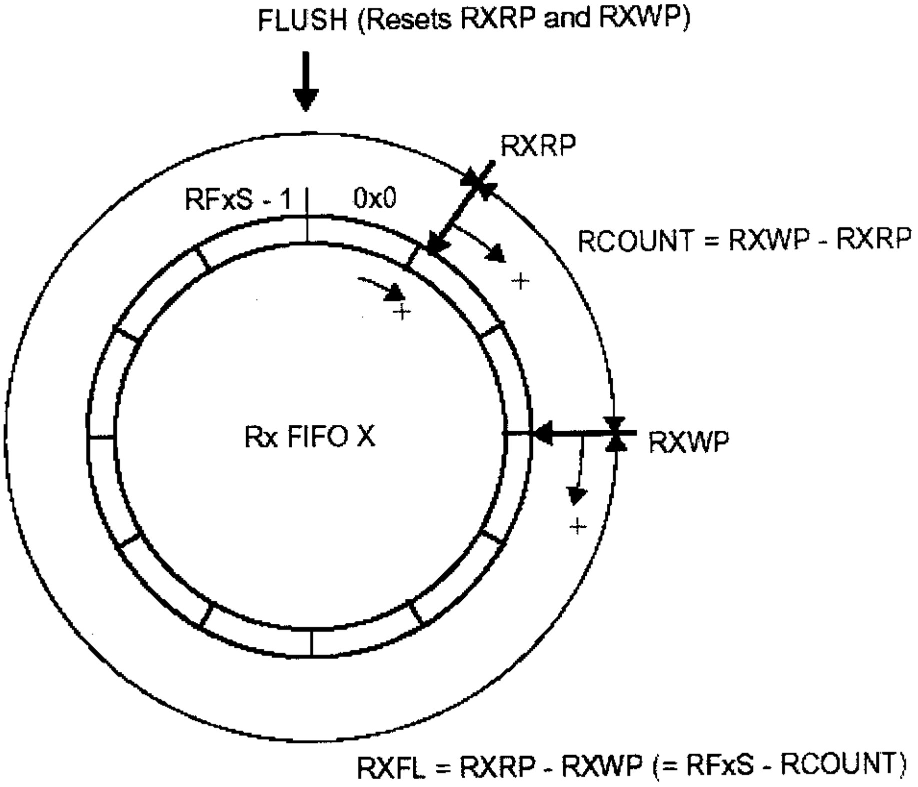
Transmit FIFO Level. This value indicates how many bytes are currently in the FIFO.A FIFO warning is issued if TXFL decreases to a specific value. The respective WARNx bit in the FWR register is set if TXFL is equal to or less than the number specified by the TFWL bit in the TXCx register.

TCOUNT:

Transmit FIFO Count. This value indicates how many empty bytes can be filled within the transmit FIFO. This value is accessible by firmware via the TxSx register.

Receive Endpoint FIFO Operation (RXFIFO1, RXFIFO2, RXFIFO3):

The Receive FIFOs for the Endpoints 2, 4 and 6 support bulk, interrupt and isochronous USB packet transfers larger than the actual FIFO size. If the packet length exceeds the FIFO size, the firmware must read the FIFO contents while the USB packet is being received on the bus. Figure 22 shows the detailed behavior of receive FIFOs.


FLUSH (Resets RXRP and RXWP)
RXRP
RFxS - 1
0x0
RCOUNT = RXWP - RXRP
+
+
Rx FIFO X
RXWP
+
RXFL = RXRP - RXWP (= RFxS - RCOUNT)

RFxS:

Receive FIFO x Size. This is the total number of bytes available within the FIFO.

RXRP:

Receive Read Pointer. This pointer is incremented with every read of the firmware from the receive FIFO. This pointer wraps around to zero if RFxS is reached. RXRP is never incremented beyond the value of RXWP. If an attempt is made to read more bytes than are actually available (FIFO underrun), the last byte is read repeatedly.

RXWP:

Receive Write Pointer. This pointer is incremented every time the Endpoint Controller writes to the receive FIFO. This pointer wraps around to zero if RFxS is reached. An overrun condition occurs if RXRP equals RXWP and an attempt is made to write an additional byte.

RXFL:

Receive FIFO Level. This value indicates how many more bytes can be received until an overrun condition occurs with the next write to the FIFO. A FIFO warning is issued if RXFL decreases to a specific value. The respective WARNx bit in the FWR register is set if RXFL is equal to or less than the number specified by the RFWL bit in the RXCx register.

RCOUNT:

Receive FIFO Count. This value indicates how many bytes can be read from the receive FIFO. This value is accessible by firmware via the RXSx register.

CLOCK GENERATION

The Clock Generator provides the CLKOUT output signal based on the programming of the Clock Configuration register (CCONF). This allows disabling of the output clock and selection of a clock divisor. The clock divisor supports a programmable output in the range of 48 MHz to 2.82 MHz. On a power-on reset, the output clock defaults to 4 MHz. A software reset has no effect on the programming of the CCONF, and thus no effect on the CLKOUT signal. In the USBN9603, the clock generation circuit is not reset. In the USBN9603, however, assertion of the RESET input does cause all registers to revert to their reset values, including CCONF, which then forces the CLKOUT signal to its default of 4 MHz.. As part of the clock generation reset, a delay of 214 XIN clock cycles is incurred before the CLKOUT signal is output

REGISTER MAP

Address	Register Mnemonic	Register Name
0x00	MCNTRL	Main Control
0x01	CCONF	Clock Configuration
0x02	Reserved	
0x03	RID	Revision Identifier
0x04	FAR	Function Address
0x05	NFSR	Node Functional State
0x06	MAEV	Main Event
0x07	MAMSK	Main Mask
0x08	ALTEV	Alternate Event
0x09	ALTMSK	Alternate Mask
0x0A	TXEV	Transmit Event
0x0B	TXMSK	Transmit Mask
0x0C	RXEV	Receive Event
0x0D	RXMSK	Receive Mask
0x0E	NAKEV	NAK Event
0x0F	NAKMSK	NAK Mask
0x10	FWEV	FIFO Warning Event
0x11	FWMSK	FIFO Warning Mask
0x12	FNH	Frame Number High Byte
0x13	FNL	Frame Number Low Byte
0x14	DMACNTRL	DMA Control
0x15	DMAEV	DMA Event
0x16	DMAMSK	DMA Mask
0x17	MIR	Mirror

Address	Register Mnemonic	Register Name
0x18	DMACNT	DMA Count
0x19	DMAERR	DMA Error Count
0x1A	Reserved	
0x1B	WKUP	Wake-Up
0x1C - 0x1F	Reserved	
0x20	EPC0	Endpoint Control 0
0x21	TXD0	Transmit Data 0
0x22	TXS0	Transmit Status 0
0x23	TXC0	Transmit Command 0
0x24	Reserved	
0x25	RXD0	Receive Data 0
0x26	RXS0	Receive Status 0
0x27	RXC0	Receive Command 0
0x28	EPC1	Endpoint Control 1
0x29	TXD1	Transmit Data 1
0x2A	TXS1	Transmit Status 1
0x2B	TXC1	Transmit Command 1
0x2C	EPC2	Endpoint Control 2
0x2D	RXD1	Receive Data 1
0x2E	RXS1	Receive Status 1
0x2F	RXC1	Receive Command 1
0x30	EPC3	Endpoint Control 3
0x31	TXD2	Transmit Data 2
0x32	TXS2	Transmit Status 2
0x33	TXC2	Transmit Command 2
0x34	EPC4	Endpoint Control 4
0x35	RXD2	Receive Data 2
0x36	RXS2	Receive Status 2
0x37	RXC2	Receive Command 2
0x38	EPC5	Endpoint Control 5
0x39	TXD3	Transmit Data 3
0x3A	TXS3	Transmit Status 3
0x3B	TXC3	Transmit Command 3
0x3C	EPC6	Endpoint Control 6
0x3D	RXD3	Receive Data 3
0x3E	RXS3	Receive Status 3
0x3F	RXC3	Receive Command 3

ATMEL 89C51

GENERAL DESCRIPTION

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4K bytes of Flash Programmable and Erasable Read Only Memory . The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard MCS-51™ instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer.

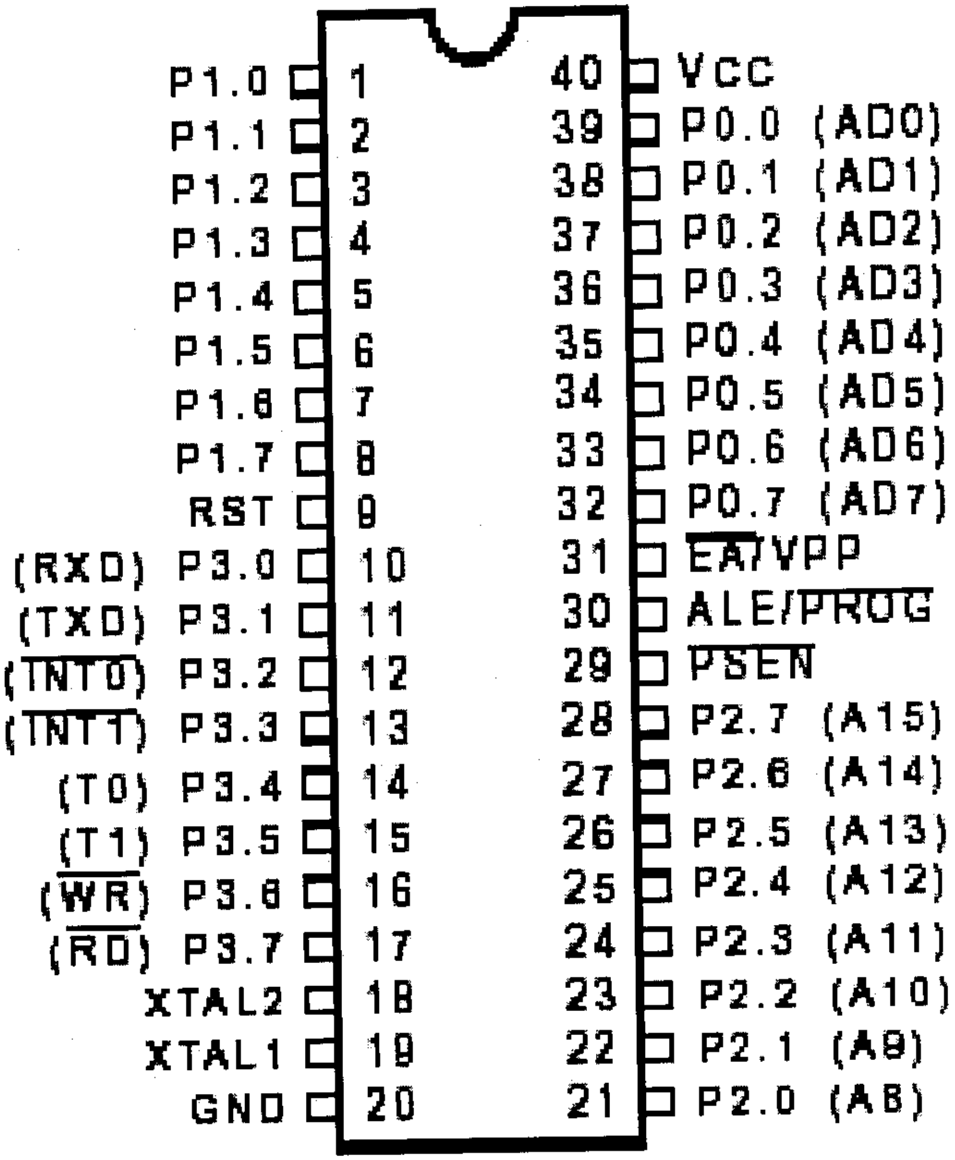
Features

- Compatible with MCS-51™ Products
- 4K Bytes of In-System Reprogrammable Flash Memory

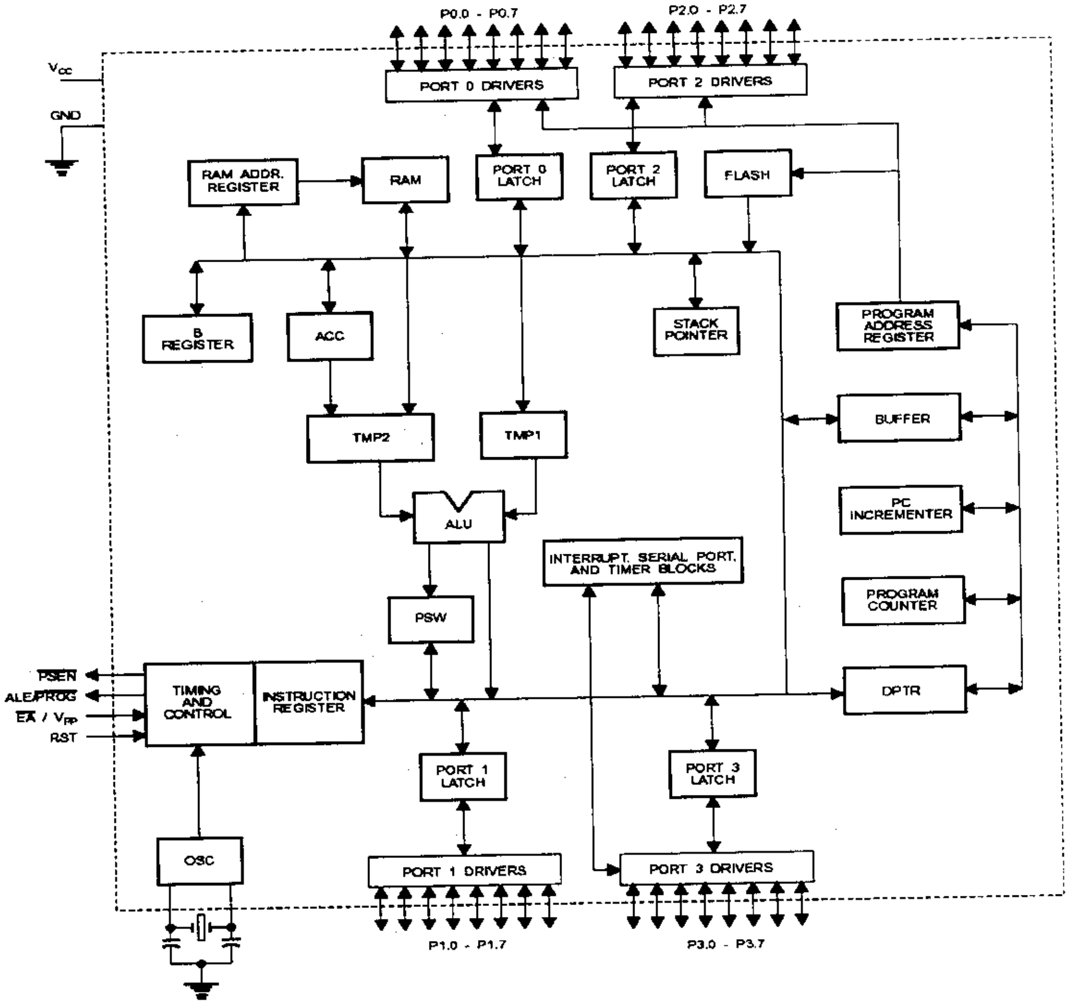
– Endurance: 1,000 Write/Erase Cycles

- Fully Static Operation: 0 Hz to 24 MHz
- Three-Level Program Memory Lock
- 128 x 8-Bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial Channel
- Low Power Idle and Power Down Modes

PIN DIAGRAM


P1.0 1
P1.1 2
P1.2 3
P1.3 4
P1.4 5
P1.5 6
P1.6 7
P1.7 8
RST 9
(RXD) P3.0 10
(TXD) P3.1 11
($\overline{\text{INT0}}$) P3.2 12
($\overline{\text{INT1}}$) P3.3 13
(T0) P3.4 14
(T1) P3.5 15
($\overline{\text{WR}}$) P3.6 16
($\overline{\text{RD}}$) P3.7 17
XTAL2 18
XTAL1 19
GND 20
40 VCC
39 P0.0 (AD0)
38 P0.1 (AD1)
37 P0.2 (AD2)
36 P0.3 (AD3)
35 P0.4 (AD4)
34 P0.5 (AD5)
33 P0.6 (AD6)
32 P0.7 (AD7)
31 $\overline{\text{EA}}$/VPP
30 ALE/$\overline{\text{PROG}}$
29 $\overline{\text{PSEN}}$
28 P2.7 (A15)
27 P2.6 (A14)
26 P2.5 (A13)
25 P2.4 (A12)
24 P2.3 (A11)
23 P2.2 (A10)
22 P2.1 (A9)
21 P2.0 (A8)

BLOCK DIAGRAM



Pin Description

VCC:

Supply voltage.

GND:

Ground.

Port 0:

Port 0 is an 8-bit open drain bidirectional I/O port. As an output port each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high impedance inputs. Port 0 may also be configured to be the multiplexed low order address/data bus during accesses to external program and data memory. In this mode P0 has internal pull ups. Port 0 also receives the code bytes during Flash programming, and outputs the code bytes during program verification. External pull ups are required during program verification.

Port 1:

Port 1 is an 8-bit bidirectional I/O port with internal pull ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins they are pulled high by the internal pull ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (IIL) because of the internal pull ups. Port 1 also receives the low-order address bytes during Flash programming and verification.

Port 2:

Port 2 is an 8-bit bidirectional I/O port with internal pull ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins they are pulled high by the internal pull ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (IIL) because of the internal pull ups. Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that uses 16-bit addresses (MOVX @ DPTR). In this application it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that uses 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3:

Port 3 is an 8-bit bidirectional I/O port with internal pull ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins they are pulled high by the internal pull ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (IIL) because of the pull ups. Port 3 also serves the functions of various special features of the AT89C51 as listed below: Port 3 also receives some control signals for Flash programming and verification.

Port Pin Alternate Functions:

P3.0 RXD (serial input port)

P3.1 TXD (serial output port)

P3.2 INT0 (external interrupt 0)

P3.3 INT1 (external interrupt 1)

P3.4 T0 (timer 0 external input)

P3.5 T1 (timer 1 external input)

P3.6 WR (external data memory write strobe)

P3.7 RD (external data memory read strobe)

RST:

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/PROG:

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming. In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

PSEN :

Program Store Enable is the read strobe to external program memory. When the AT89C51 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

EA/VPP:

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 12-volt programming enable voltage (VPP) during Flash programming, for parts that require 12-volt VPP.

XTAL1:

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

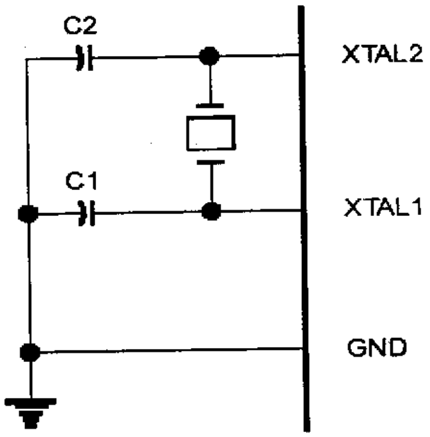
XTAL2:

Output from the inverting oscillator amplifier.

Oscillator Characteristics:

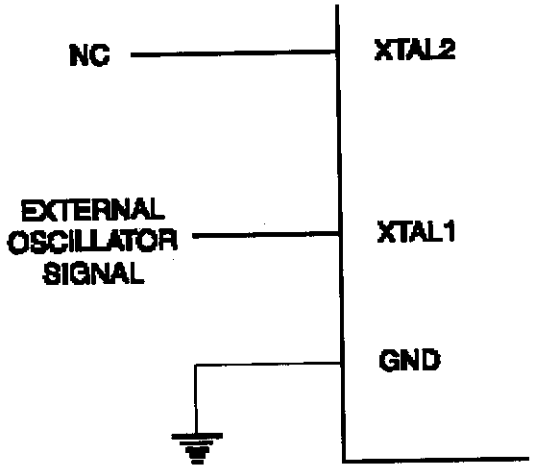
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum Voltage, high and low time specifications must be observed.

OSCILLATOR CONNECTIONS:



C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

External Clock Drive Configuration:



PROGRAMMING THE 89C51

Idle Mode:

In idle mode, the CPU puts itself to sleep while all the on chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset. It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Power Down Mode:

In the power down mode the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power down mode is terminated. The only exit from power down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before VCC is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Program Memory Lock Bits:

On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below: When lock bit 1 is programmed, the logic level at the EA pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value, and holds that value until reset is activated. It is necessary that the latched value of EA be in agreement with the current logic level at that pin in order for the device to function properly.

Programming the Flash:

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (VCC) program enable signal. The low voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third party Flash or EPROM programmers. The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table. The AT89C51 code memory array is programmed byte-by-byte in either programming mode. To program any nonblank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.

Programming Algorithm:

Before programming the AT89C51, the address, data and control signals should be set up according to the Flash programming mode table. To program the AT89C51, take the following steps.

1. Input the desired memory location on the address lines.

2. Input the appropriate data byte on the data lines.

3. Activate the correct combination of control signals.

4. Raise EA/VPP to 12V for the high-voltage programming mode.

5. Pulse ALE/PROG once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

Data Polling:

The AT89C51 features Data Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

Ready/Busy:

The progress of byte programming can also be monitored by the RDY/BSY output signal. P3.4 is pulled low after ALE goes high during programming to indicate BUSY. P3.4 is pulled high again when programming is done to indicate READY.

Program Verify:

If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase:

The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/PROG low for 10 ms. The code array is written with all “1”s. The chip erase operation must be executed before the code memory can be re-programmed.

Programming Interface:

Every code byte in the Flash array can be written and the entire array can be erased by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

USART 8251A

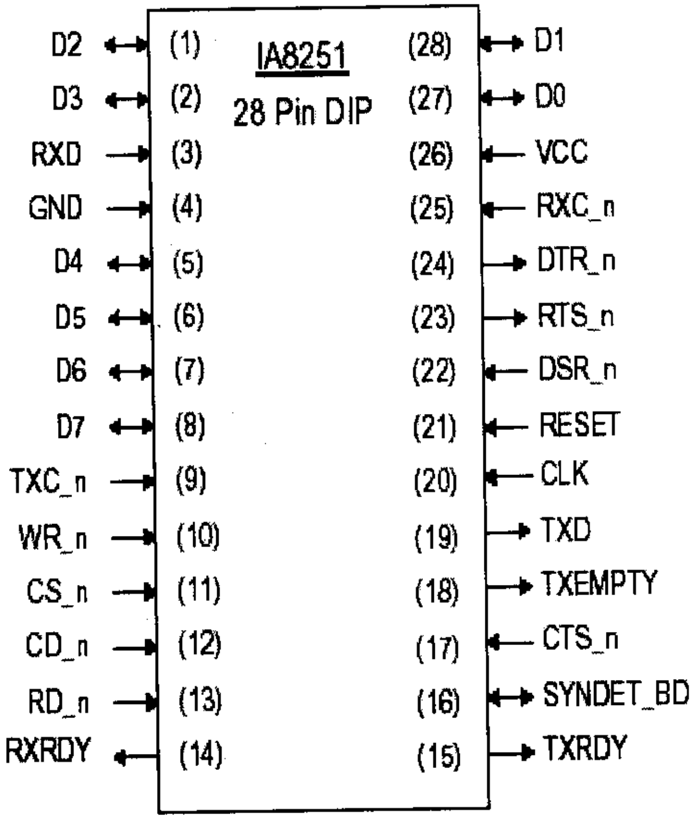
GENERAL DESCRIPTION:

8251A is the industry standard Universal Synchronous/Asynchronous Receiver/Transmitter (USART) designed for data communication. It is used as a peripheral device and is programmed by the CPU. USART accepts data characters from the CPU in parallel format and then converts them into a continuous serial data stream for transmission. It can also receive serial data streams and convert them into parallel data characters for the CPU. The USART will signal the CPU whenever it can accept a new character for transmission or whenever it has received a character for the CPU. The CPU can read the complete status of the USART at any time.

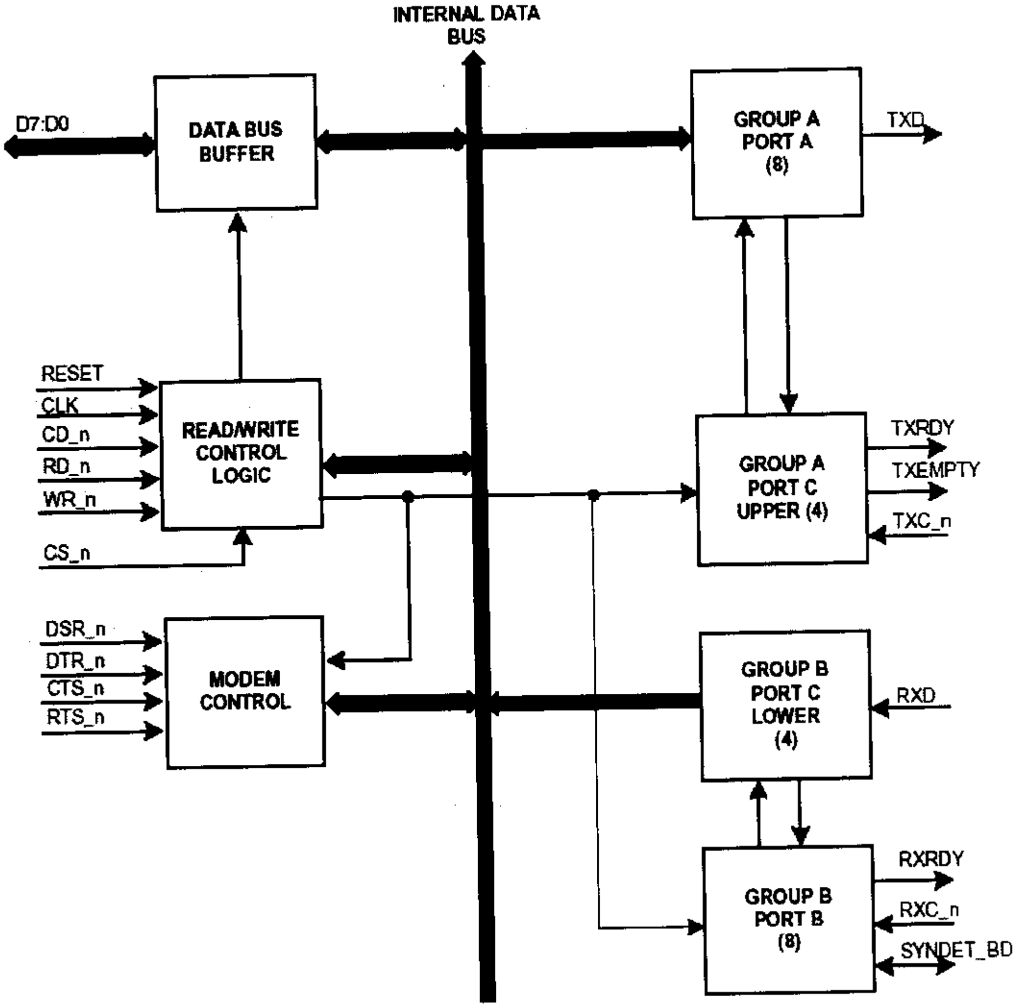
FEATURES

- Synchronous and Asynchronous Operation.
- Asynchronous 5-8 Bit Characters; Clock Rate-1, 16 or 64 times Baud Rate; Break Character Generation; 1, 1.5 or 2 Stop Bits; False Start Bit Detection; Automatic Break Detect and Handling.
- Asynchronous Baud Rate-DC to 19.2K Baud.
- All inputs and outputs are TTL Compatible.
- Full Duplex, Double-Buffered Transmitter and Receiver.
- Programmable Word Length, Stop Bits, and Parity.
- Offers Divide-by-1, -16, or –64 Mode.
- 28-Pin DIP Package.

PIN DIAGRAM


IA8251
28 Pin DIP
D2 (1)
D3 (2)
RXD (3)
GND (4)
D4 (5)
D5 (6)
D6 (7)
D7 (8)
TXC_n (9)
WR_n (10)
CS_n (11)
CD_n (12)
RD_n (13)
RXRDY (14)
(28) D1
(27) D0
(26) VCC
(25) RXC_n
(24) DTR_n
(23) RTS_n
(22) DSR_n
(21) RESET
(20) CLK
(19) TXD
(18) TXEMPTY
(17) CTS_n
(16) SYNDET_BD
(15) TXRDY

BLOCK DIAGRAM



PIN DESCRIPTION:

CS-Chip Select:

When this signal is low the 8251A is selected for communication.

C/D-Control/Data:

When this signal is high, the control register or the status register is addressed; when it is low, the data buffer is addressed. They are differentiated by write and read signals.

WR-Write:

When this signal goes low, the control register is written or output is sent to data buffer.

RESET-Reset:

A high on this input resets the 8251A and forces it into the idle mode.

CLK-Clock:

This is the clock input, usually connected to the system clock.

TxD-Transmit Data:

Serial bits are transmitted on this line.

TxC-Transmitter Clock:

This input signal controls the rate at which bits are transmitted by the USART. The clock frequency can be 1, 16 or 64 times the baud.

TxRDY-Transmitter Ready:

This is an output signal. When it is high, it indicates that the buffer register is empty and the USART is ready to accept a byte.

TxE-Transmitter Empty:

This is an output signal. Logic 1 on this line indicates that the output register is empty. This signal when a byte is transferred from the buffer to the output registers.

RxD-Receive Data:

Bits are received serially on this line and converted into a parallel byte in the receiver input register.

RxC-Receiver Clock:

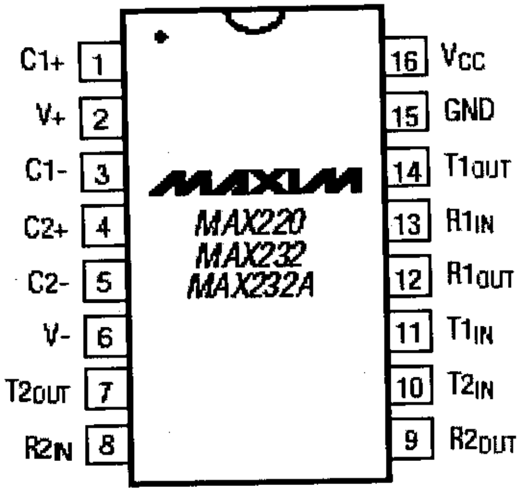
This input signal controls the rate at which bits are received by the USART. The clock frequency can be 1, 16 or 64 times the baud.

RxRDY-Receiver Ready:

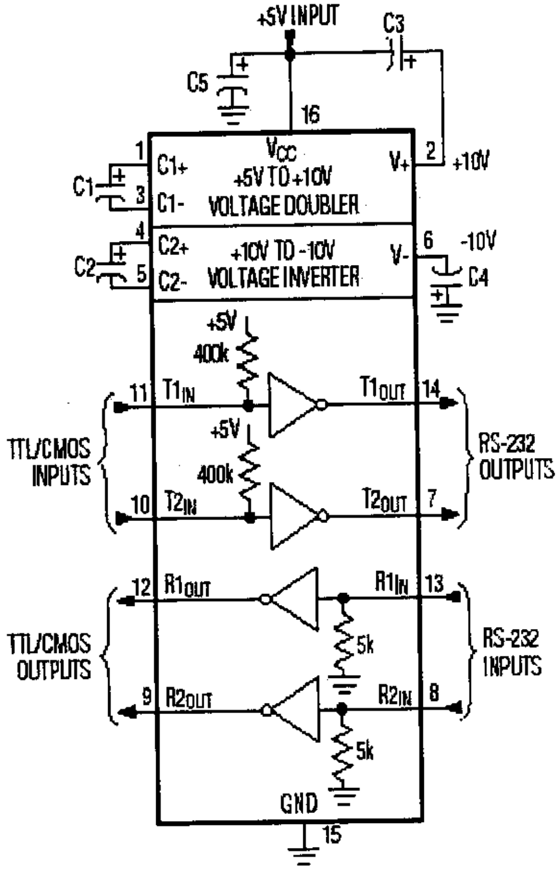
This is an output signal. It goes high when the USART has a character in the buffer register and is ready to transfer it. This line is used to either indicate the status or as interrupt.

MAX232

PIN DIAGRAM:



DIP/SO



C1=C2=C3=C4=C5=0.1µf

PIN DESCRIPTION

The MAX232 contains three sections:

- dual charge-pump DC-DC voltage converters
- RS-232 drivers
- RS-232 receivers

Dual Charge-Pump Voltage Converter

The MAX232 has two internal charge-pumps that convert +5V to ±10V (unloaded) for RS-232 driver operation. The first converter uses capacitor C1 to double the +5V input to +10V on C3 at the V+ output. The second converter uses capacitor C2 to invert +10V to -10V on C4 at the V- output. A small amount of power may be drawn from the +10V (V+) and -10V (V-) outputs to power external circuitry. V+ and V- are not regulated, so the output voltage drops with increasing load current. Do not load V+ and V- to a point that violates the minimum ±5V EIA/TIA-232E driver output voltage when sourcing current from V+ and V- to external circuitry. For applications where a +10V external supply is applied to the V+ pin (instead of using the internal charge pump to generate +10V), the C1 capacitor must not be installed and the SHDN pin must be tied to VCC. This is because V+ is internally connected to VCC in shutdown mode.

RS-232 Drivers

The typical driver output voltage swing is ±8V when loaded with a nominal 5kohm RS-232 receiver and VCC = +5V. Output swing is guaranteed to meet the EIA/TIA- 232E and V.28 specification, which calls for ±5V minimum driver output levels under worst-case conditions. These include a minimum 3kohm load, VCC = +4.5V, and maximum operating temperature. Unloaded driver output voltage ranges from (V+ -1.3V) to (V- +0.5V). Input thresholds are both TTL and CMOS compatible. The inputs of unused drivers can be left unconnected since 400kohm input pull-up resistors to VCC are built in. The pull-up resistors force the outputs of unused drivers low because all drivers invert. The internal input pull-up resistors typically source 12μA, except in shutdown mode where the pull-ups are disabled.

Driver outputs turn off and enter a high-impedance state where leakage current is typically microamperes (maximum 25μA) when in shutdown mode, in three state modes, or when device power is removed. Outputs can be driven to ±15V. The power supply current typically drops to 8μA in shutdown mode. When in low-power shutdown mode, the driver outputs are turned off and their leakage current is less than 1μA with the driver output pulled to ground. The driver output leakage remains less than 1μA, even if the transmitter output is back driven between 0V and (VCC + 6V). Below -0.5V, the transmitter is diode clamped to ground with 1kohm series impedance. The transmitter is also zener clamped to approximately VCC + 6V, with a series impedance of 1kohm.The driver output slew rate is limited to less than 30V/μs as required by the EIA/TIA-232E and V.28 specifications. Typical slew rates are 24V/μs unloaded and 10V/μs loaded with 3ohm and 2500pF.

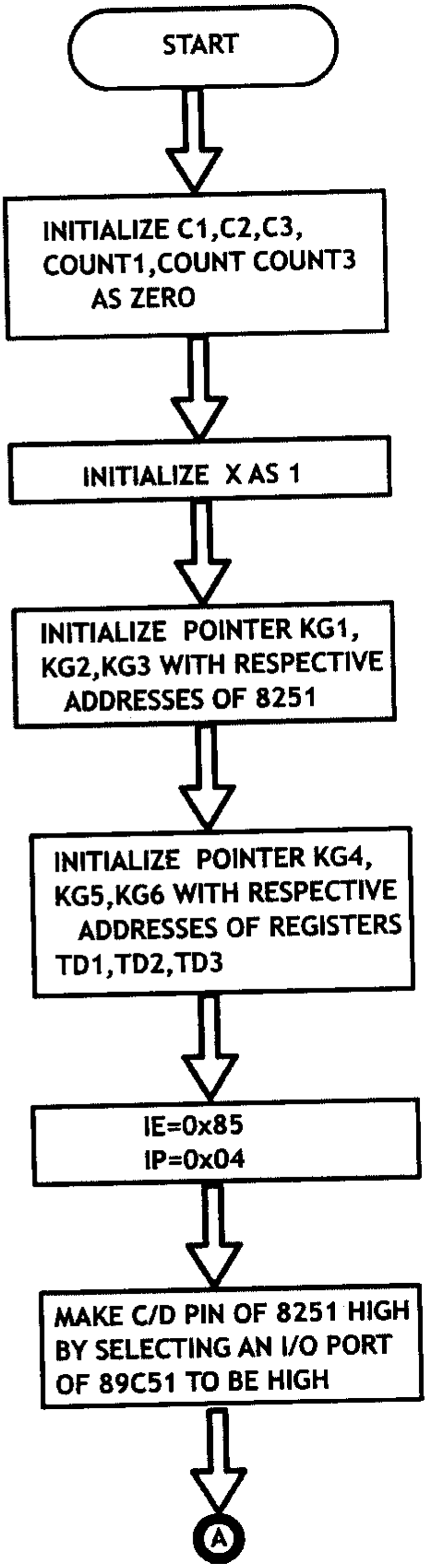
RS-232 Receivers

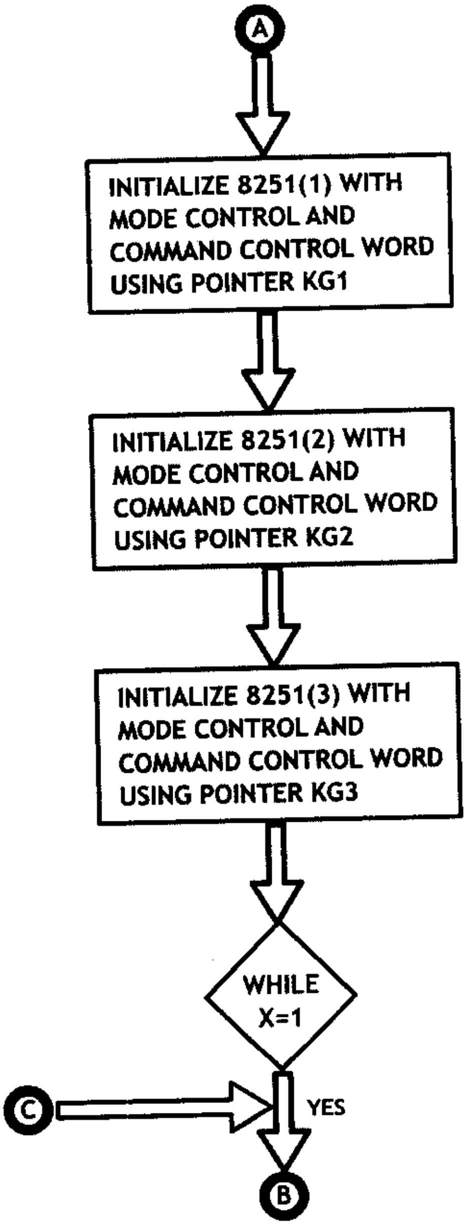
EIA/TIA-232E and V.28 specifications define a voltage level greater than 3V as a logic 0, so all receivers invert. Input thresholds are set at 0.8V and 2.4V, so receivers respond to TTL level inputs as well as EIA/TIA-232E and V.28 levels. The receiver inputs withstand an input over voltage up to ±25V and provide input terminating resistors with nominal 5kohm values. The receiver input hysteresis is typically 0.5V with a guaranteed minimum of 0.2V. This produces clear output transitions with slow-moving input signals, even with moderate amounts of noise and ringing. The receiver propagation delay is typically 600ns and is independent of input swing direction.

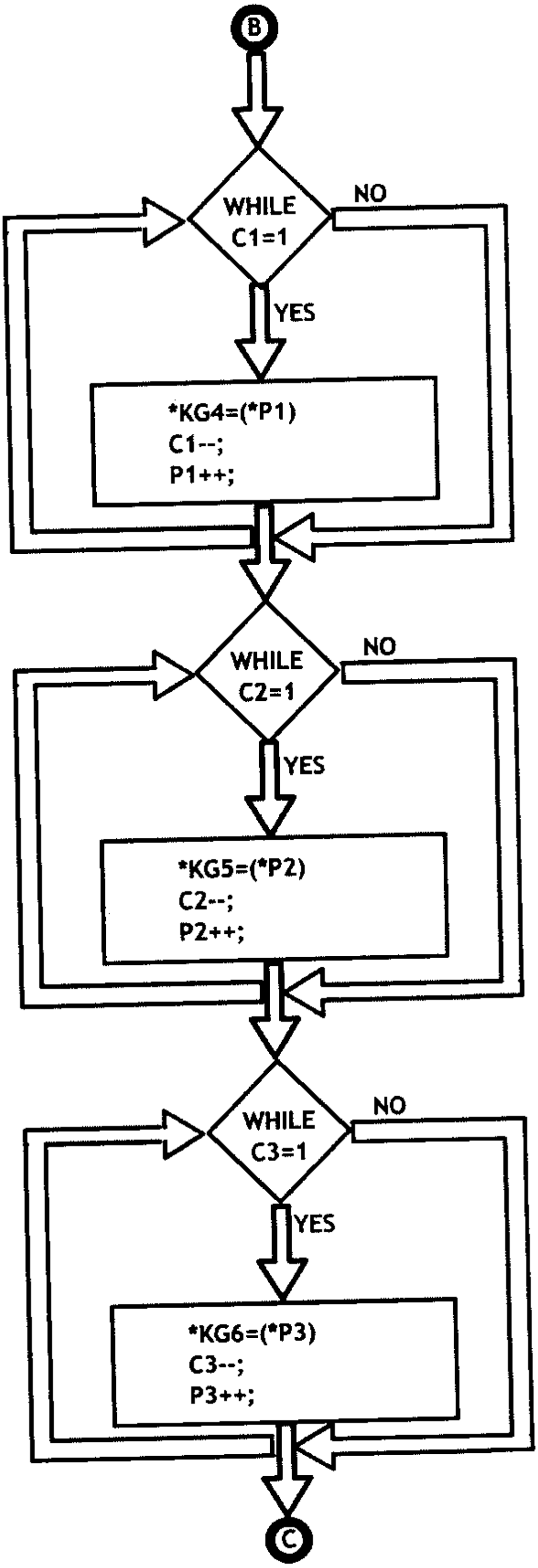
NOTE:

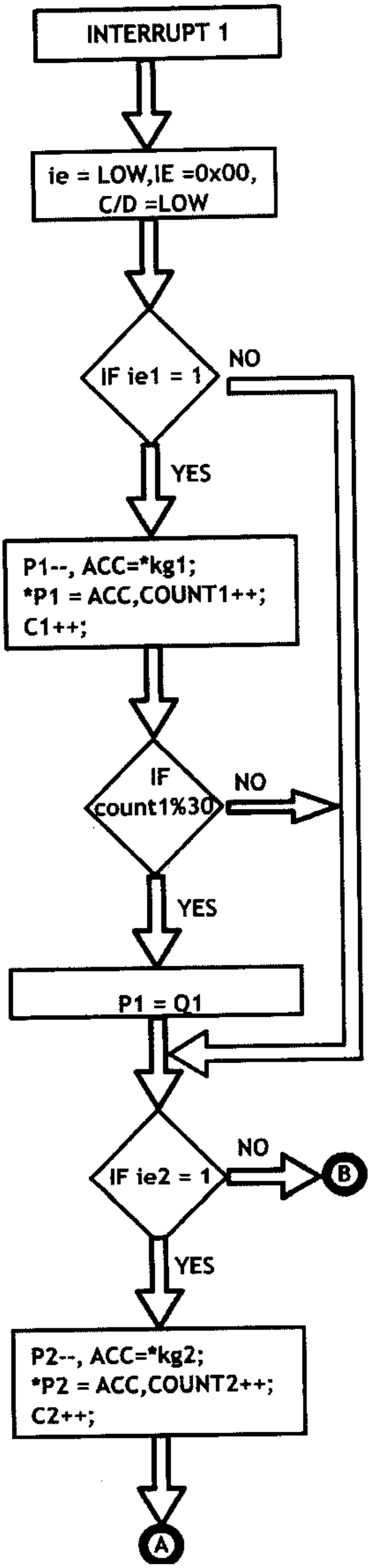
In applications that are sensitive to power-supply noise, Vcc should be decoupled to ground with a capacitor of the same value as C1 and C2 connected as close as possible to the device.

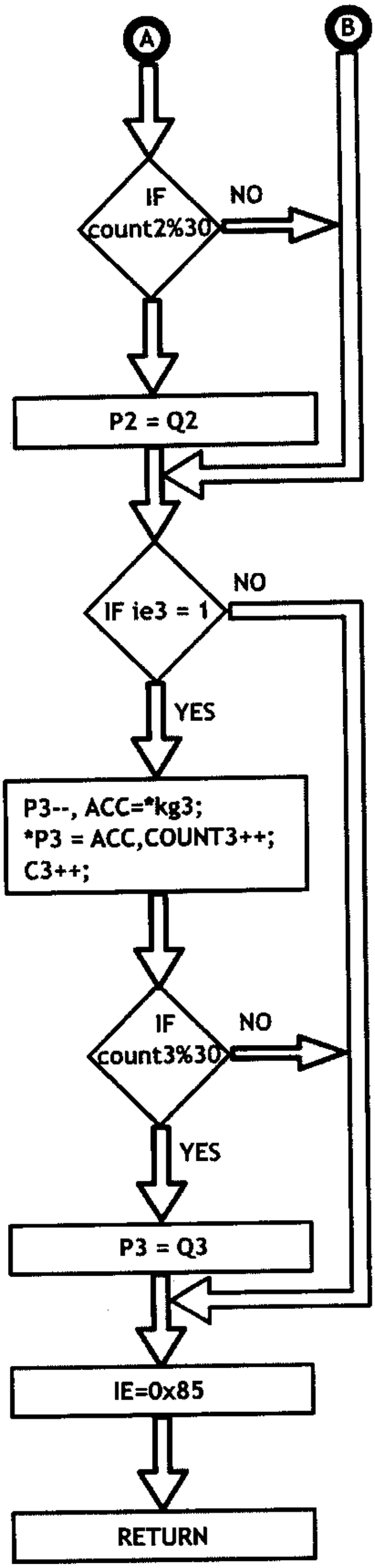
FLOWCHART


START
INITIALIZE C1,C2,C3, COUNT1,COUNT COUNT3 AS ZERO
INITIALIZE X AS 1
INITIALIZE POINTER KG1, KG2,KG3 WITH RESPECTIVE ADDRESSES OF 8251
INITIALIZE POINTER KG4, KG5,KG6 WITH RESPECTIVE ADDRESSES OF REGISTERS TD1,TD2,TD3
IE=0x85
IP=0x04
MAKE C/D PIN OF 8251 HIGH BY SELECTING AN I/O PORT OF 89C51 TO BE HIGH
A


A
INITIALIZE 8251(1) WITH MODE CONTROL AND COMMAND CONTROL WORD USING POINTER KG1
INITIALIZE 8251(2) WITH MODE CONTROL AND COMMAND CONTROL WORD USING POINTER KG2
INITIALIZE 8251(3) WITH MODE CONTROL AND COMMAND CONTROL WORD USING POINTER KG3
WHILE X=1
YES
C
B

B
WHILE C1=1
NO
YES
*KG4=(*P1)
C1--;
P1++;
WHILE C2=1
NO
YES
*KG5=(*P2)
C2--;
P2++;
WHILE C3=1
NO
YES
*KG6=(*P3)
C3--;
P3++;
C


INTERRUPT 1
ie = LOW,IE =0x00,
C/D =LOW
IF ie1 = 1
NO
YES
P1--, ACC=*kg1;
*P1 = ACC,COUNT1++;
C1++;
IF
count1%30
NO
YES
P1 = Q1
IF ie2 = 1
NO
B
YES
P2--, ACC=*kg2;
*P2 = ACC,COUNT2++;
C2++;
A


A
B
IF
count2%30
NO
P2 = Q2
IF ie3 = 1
NO
YES
P3--, ACC=*kg3;
*P3 = ACC,COUNT3++;
C3++;
IF
count3%30
NO
YES
P3 = Q3
IE=0x85
RETURN

C CODING

PROGRAM

```
#include<reg51.h>
#define high 1;
#define low 0;

unsigned char xdata *kg1;
unsigned char xdata *kg2;
unsigned char xdata *kg3;
unsigned char xdata *kg4;
unsigned char xdata *kg5;
unsigned char xdata *kg6;

unsigned  char *p1 = 0x1e;
unsigned  char *p2 = 0x3c;
unsigned  char *p3 = 0x5a;
unsigned  char *p4 = 0x78;
unsigned  char *q1 = 0x1e;
unsigned  char *q2 = 0x3c;
unsigned  char *q3 = 0x5a;
unsigned  char *q4 = 0x78;

sbit cd = P3^4;
sbit i1 = P1^4;
sbit i2 = P1^5;
sbit i3 = P1^6;
sbit ie = P1^7;

int count1=0;
int count2=0;
int count3=0;
int c1=0;
int c2=0;
int c3=0;
int x = 1;
void initialize_8251();
```

```
nain()

P1=0x00;
TCON=0x0a;
e = high;
IE=0x85;
IP=0x04;
kg1 =0x40;
kg2 =0x80;
kg3 =0xc0;
kg4 =0x29;
kg5 = 0x31;
kg6 = 0x39;
initialize_8251();
 while(x)
 {
 while(c1)
 {
 *kg4=(*p1)
 c1--;
 p1++;
 }
  while(c2)
  {
  *kg5=(*p2)
  c2--;
  p2++;
  }
   while(c3)
   {
   *kg6=(*p3)
   c3--;
   p3++;
   }
   }
   }
   void initialize_8251()
   {
   cd = high;
   *kg1=0x4e;
   *kg1=0x04;
```

```
kg2=0x4e;
kg3=0x04;
*kg4=0x4e;
*kg4=0x04;

void interrupt_8051() interrupt 1
{
e = low;
IE = 0x00;
cd =low;
if (i1==1)
 {
 p1--;
 ACC=*kg1;
 *p1 = ACC;
 count1++;
 c1++;
 if((c1==30))
  {
  }
 if((count1%30)==0)
  {
  p1 = q1;
  }
 }
 if (i2 == 1)
  {
  p2--;
  ACC=*kg2;
  *p2 = ACC;
  count2++;
  c2++;
  if((c2==30))
   {
   }
  if((count2%30)==0)
   {
   p2 = q2;
   }
  }
```

10 Transmit Mask Register (TXMSK)

en set and the corresponding bit in the TXEV register is set, TX_EV in the MAEV register is set. When cleared, the corponding bit in the TXEV register does not cause TX_EV to be set.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Same Bit Definition as TXEV Register							
0	0	0	0	0	0	0	0
r/w							

.11 Receive Event Register (RXEV)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
RXFIFO3	RXFIFO2	RXFIFO1	FIFO0	RXFIFO3	RXFIFO2	RXFIFO1	FIFO0
RXOVRRN3-0				RXFIFO3-0			
0	0	0	0	0	0	0	0
CoR				r			

XFIFO

eceive FIFO. These bits are set whenever either RX_ERR or RX_LAST in the respective Receive Status (RXSx) register set. Reading the corresponding RXSx register automatically clears these bits.

e device discards all packets for Endpoint 0 received with errors. This is necessary in case of retransmission due to media rors, ensuring that a good copy of a SETUP packet is captured. Otherwise, the FIFO may potentially be tied up, holding rrupted data and unable to receive a retransmission of the same packet (the RXFIFO0 bit does only reflect the value of X_LAST for Endpoint 0).

data streaming is used for the receive endpoints (EP2, EP4 and EP6) the firmware must check with the respective X_ERR bits to ensure the packets received are not corrupted by errors.

XOVRRN

eceive Overrun. These bits are set in the event of a FIFO overrun condition. They are cleared when the register is read.

he firmware must check with the respective RX_ERR bits that packets received for the other receive endpoints (EP2, EP4 nd EP6) are not corrupted by errors, as these endpoints support data streaming (packets which are longer than the actual IFO depth).

.1.12 Receive Mask Register (RXMSK)

Vhen set and the corresponding bit in the RXEV register is set, RX_EV in the MAEV register is set. When cleared, the coresponding bit in the RXEV register does not cause RX_EV to be set.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Same Bit Definition as RXEV Register							
0	0	0	0	0	0	0	0
r/w							

Register Set (Continued)

.13 NAK Event Register (NAKEV)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
RXFIFO3	RXFIFO2	RXFIFO1	FIFO0	TXFIFO3	TXFIFO2	TXFIFO1	FIFO0
OUT3-0				IN3-0			
0	0	0	0	0	0	0	0
CoR				CoR			

t to 1 when a NAK handshake is generated for an enabled address/endpoint combination (AD_EN in the Function Ad-
ess, FAR, register is set to 1 and EP_EN in the Endpoint Control, EPCx, register is set to 1) in response to an IN token.
is bit is cleared when the register is read.

JT

t to 1 when a NAK handshake is generated for an enabled address/endpoint combination (AD_EN in the FAR register is
t to 1 and EP_EN in the EPCx register is set to 1) in response to an OUT token. This bit is not set if NAK is generated as
sult of an overrun condition. It is cleared when the register is read.

1.14 NAK Mask Register (NAKMSK)

hen set and the corresponding bit in the NAKEV register is set, the NAK bit in the MAEV register is set. When cleared, the
orresponding bit in the NAKEV register does not cause NAK to be set.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Same Bit Definition as NAKEV Register							
0	0	0	0	0	0	0	0
r/w							

.2 TRANSFER REGISTERS

.2.1 FIFO Warning Event Register (FWEV)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
RXFIFO3	RXFIFO2	RXFIFO1	-	TXFIFO3	TXFIFO2	TXFIFO1	-
RXWARN3-1			Reserved	TXWARN3-1			Reserved
0	0	0	-	0	0	0	-
r				r			-

TXWARN

Transmit Warning. Set to 1 when the respective transmit endpoint FIFO reaches the warning limit, as specified by the TFWL
bits of the respective TXCx register, and transmission from the respective endpoint is enabled. This bit is cleared when the
warning condition is cleared by either writing new data to the FIFO when the FIFO is flushed, or when transmission is done,
as indicated by the TX_DONE bit in the TXSx register.

RXWARN

Receive Warning. Set to 1 when the respective receive endpoint FIFO reaches the warning limit, as specified by the RFWL
bits of the respective EPCx register. This bit is cleared when the warning condition is cleared by either reading data from the
FIFO or when the FIFO is flushed.

Register Set (Continued)

2 FIFO Warning Mask Register (FWMSK)

en set and the corresponding bit in the FWEV register is set, WARN in the MAEV register is set. When cleared, the cor- ponding bit in the FWEV register does not cause WARN to be set.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Same Bit Definition as FWEV Register							
0	0	0	0	0	0	0	0
r/w							

2.3 Frame Number High Byte Register (FNH)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
MF	UL	RFC	Reserved		FN10-8		
1	1	0	-		0	0	0
r	r	w/r0	-		r		

N

rame Number. This is the current frame number received in the last SOF packet. If a valid frame number is not received ithin 12060 bit times (Frame Length Maximum, FLMAX, with tolerance) of the previous change, the frame number is incre- ented artificially. If two successive frames are missed or are incorrect, the current FN is frozen and loaded with the next ame number from a valid SOF packet.

the frame number low byte was read by firmware before reading the FNH register, the user actually reads the contents of buffer register which holds the value of the three frame number bits of this register when the low byte was read. Therefore, e correct sequence to read the frame number is: FNL, FNH. Read operations to the FNH register, without first reading the rame Number Low Byte (FNL) register directly, read the actual value of the three MSBs of the frame number. On reset, FN set to 0.

FC

Reset Frame Count. Setting this bit resets the frame number to 0x000, after which this bit clears itself. This bit always reads .

L

Unlock Flag. This bit indicates that at least two frames were received without an expected frame number, or that no valid SOF was received within 12060 bit times. If this bit is set, the frame number from the next valid SOF packet is loaded in FN. On reset, this flag is set to 1.

MF

Missed SOF Flag. This flag is set when the frame number in a valid received SOF does not match the expected next value, or when an SOF is not received within 12060 bit times. On reset, this flag is set to 1.

7.2.4 Frame Number Low Byte Register (FNL)

This register holds the low byte of the frame number, as described above. To ensure consistency, reading this low byte causes the three frame number bits in the FNH register to be locked until this register is read. The correct sequence to read the frame number is: FNL, FNH. On reset, FN is set to 0.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
FN7-0							
0	0	0	0	0	0	0	0
r							

Register Set (Continued)

5 Function Address Register (FAR)

s register sets the device function address. The different endpoint numbers are set for each endpoint individually via the dpoint Control registers.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
AD_EN	AD6-0						
0	0	0	0	0	0	0	0
r/w	r/w						

)

dress. This field holds the 7-bit function address used to transmit and receive all tokens addressed to the device.

)_EN

dress Enable. When set to 1, bits AD6-0 are used in address comparison (see Section 6.2 for a description). When eared, the device does not respond to any token on the USB bus.

ote: If the DEF bit in the Endpoint Control 0 register is set, Endpoint 0 responds to the default address.

2.6 DMA Control Register (DMACNTRL)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
DEN	IGNRXTGL	DTGL	ADMA	DMOD	DSRC2-0		
0	0	0	0	0	0		0
r/w	r/w	r/w	r/w	r/w	r/w		

SRC

MA Source. The DMA source bit field holds the binary-encoded value that specifies which of the endpoints, 1 to 6, is enabled for DMA support. The DSRC bits are cleared on reset. Table 7 summarizes the DSRC bit settings.

Table 7. DSRC Bit Description

DSRC			Endpoint No.
2	1	0	
0	0	0	1
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	x	Reserved

)MOD

DMA Mode. This bit specifies when a DMA request is issued. If reset, a DMA request is issued on transfer completion. For transmit endpoints EP1, EP3 and EP5, the data is completely transferred as indicated by the TX_DONE bit (to fill the FIFO with new transmit data). For receive endpoints EP2, EP4 and EP6, this is indicated by the RX_LAST bit. When the DMOD bit is set, a DMA request is issued when the respective FIFO warning bit is set. The DMOD bit is cleared on reset.

MA request from a transmit endpoint is activated until the request condition clears. If DMOD is set to 0, DMA requests issued either until the firmware reads the respective Transmit Status (TXSx) register, thus resetting the TX_DONE bit, the TX_LAST bit in the Transmit Command (TXCx) register is set by firmware. If DMOD is set to 1, DMA requests are ued until the FIFO warning condition clears, either due to sufficient bytes being transferred to the endpoint, or if the _DONE bit is set due to a transmission.

A requests from a receive endpoint are activated until the request condition clears. If DMOD is set to 0, DMA requests issued either until the firmware reads the respective Receive Status (RXSx) register, thus resetting the RX_LAST bit, or e endpoint FIFO becomes empty due to sufficient reads. If DMOD is set to 1, DMA requests are issued until the FIFO rning condition clears, or if the endpoint FIFO becomes empty due to sufficient reads.

MOD is set to 0 and the endpoint and DMA are enabled, DMA requests are issued until the firmware reads the respective Sx or RXSx register, thus resetting the TX_DONE/RX_LAST bit. If DMOD is set to 1 and the endpoint and DMA are ened, DMA requests are issued until the FIFO warning condition clears.

MA

tomatic DMA. Setting this bit automatically enables the selected receive or transmit endpoint. Before ADMA mode can be abled, the DEN bit in the DMA Control (DMACNTRL) register must be cleared. ADMA mode functions until any bit in the IA Event (DMAEV) register is set, except for NTGL. To initiate ADMA mode, all bits in the DMAEV register must be ared, except for NTGL.

r receive operations, the receiver is automatically enabled; when the packet is received, it is transferred via DMA to mem-

/.

r transmit operations, the packet data is transferred via DMA from memory; then the transmitter is automatically enabled.

r ADMA operations, the DMOD bit is ignored. All operations proceed as if DMOD is set to 0.

hen the device enters ADMA mode, any existing endpoint state may be lost. If there is already data in the FIFO, it is shed. The existing state of the RX_EN or TX_EN state may also change.

earing ADMA exits ADMA mode. DEN may either be cleared at the same time or later. If at the same time, all DMA operions cease immediately and firmware must transfer any remaining data. If later, the device completes any current DMA eration before exiting ADMA mode (see the description of the DSHLT bit in the DMAEV register for more information).

TGL

MA Toggle. This bit is used to determine the initial *state* of ADMA operations. Firmware initially sets this bit to 1 if starting ith a DATA1 operation, and to a 0 if starting with a DATA0 operation.

/rites to this bit also update the NTGL bit in the DMAEV register.

GNRXTGL

nore RX Toggle. If this bit is set, the compare between the NTGL bit in the DMAEV register and the TOGGLE bit in the espective RXSx register is ignored during receive operations. In this case, a mismatch of both bits during a receive operaon does not stop ADMA operation. If this bit is not set, the ADMA stops in case of a mismatch of the two toggle bits. After eset, this bit is set to 0.

EN

MA Enable. This bit enables DMA mode when set. If this bit is reset and the current DMA cycle is completed (or was not et issued) the DMA transfer is terminated. When the device operates in serial interface mode (MODE1 pin is tied high) DMA node cannot be enabled, thus setting this bit has no effect. This bit is cleared on reset.

.2.7 DMA Event Register (DMAEV)

he bits in this register are used with ADMA mode. Bits 0 to 3 may cause an interrupt if not cleared, even if the device is not set to ADMA mode. Until all of these bits are cleared, ADMA mode cannot be initiated. Conversely, ADMA mode is automatcally terminated when any of these bits are set..

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved		NTGL	Reserved	DSIZ	DCNT	DERR	DSHLT
-		0	-	0	0		0
-		r	-	CoW	CoW	CoW	CoW

DSHLT

DMA Software Halt. This bit is set when ADMA operations have been halted by firmware. This bit is set only after the DMA

If the ADMA bit is cleared (but DEN remains set). In this case, the current operation (if any) is completed. This means that any data in the FIFO is either transmitted or transferred to memory by DMA (if receiving). The DSHLT bit is set only after this has occurred. Note that since DEN remains set, it may need to be cleared later. This commonly is done inside the DSHLT interrupt handler.

If the DEN bit is cleared (ADMA may either remain set, or may be cleared at the same time). This ceases all DMA operations and immediately sets the DSHLT bit. If there is data in the FIFOs, it is retained but not transmitted.

If the firmware attempts to read the FIFO (if receiving) or write to the FIFO (if transmitting). This ceases all DMA operations and immediately sets the DSHLT bit. The read or write operation may not succeed since this operation is likely to corrupt the FIFO and lose some data.

If the firmware attempts to read to/write from the corresponding EPCx, TXCx, RXCx, TXSx, or RXSx registers (when DEN and ADMA in the DMACNTRL register are both set). This halts all DMA operations and immediately sets the DSHLT bit. The read or write operation is not effected.

ERR

MA Error. This bit is set to indicate that a packet has not been received or transmitted correctly. It is also set if the TOGGLE in the RXSx/TXSx register does not equal the NTGL bit in the DMAEV register after packet reception/transmission. (Note at this comparison is made before the NTGL bit changes state due to packet transfer).

r receiving, DERR is equivalent to RX_ERR. For transmitting, it is equivalent to TX_DONE (set) and ACK_STAT (not set).

he AEH bit in the DMA Error Count (DMAERR) register is set, DERR is not set until DMAERRCNT in the DMAERR register cleared, and another error is detected. Errors are handled as specified in the DMAERR register.

CNT

MA Count. This bit is set when the DMA Count (DMACNT) register is 0 (see the DMACNT register for more information).

SIZ

MA Size. This bit is only significant for DMA receive operations. It indicates that a packet has been received which is less an the full length of the FIFO. This normally indicates the end of a multi-packet transfer.

TGL

ext Toggle. This bit determines the toggle state of the next data packet sent (if transmitting), or the expected toggle state f the next data packet (if receiving). This bit is initialized by writing to the DTGL bit of the DMACNTRL register. It then changs state with every packet sent or received on the endpoint presently selected by DSRC2-0. If DTGL write operation occurs imultaneously with the bit update operation, the write takes precedence.

transmitting, whenever ADMA operations are in progress the DTGL bit overrides the corresponding TOGGLE bit in the XCx register. In this way, the alternating data toggle occurs correctly on the USB.

lote that there is no corresponding mask bit for this event because it is not used to generate interrupts.

.2.8 DMA Mask Register (DMAMSK)

ny bit set to 1 in this register enables automatic setting of the DMA bit in the ALTEV register when the respective event in he DMAEV register occurs. Otherwise, setting the DMA bit is disabled. For a description of bits 0 to 3, see the DMAEV register.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
				DSIZ	DCNT	DERR	DSHLT
-				0	0		0
-				r/w	r/w	r/w	r/w

9 Mirror Register (MIR)

s is a read only register. Since reading it does not alter the state of the TXSx or RXSx register to which it points, the ware can freely check the status of the channel.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
STAT							
-							
r							

AT

atus. This field mirrors the status bits of the transmitter or receiver selected by the DSRC2-0 field in the DMACNTRL reg- er (DMA need not be active or enabled). It corresponds to TXSx or RXSx, respectively.

2.10 DMA Count Register (DMACNT)

is register allows a maximum count to be specified for ADMA operations

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
DCOUNT7-0							
-							
r/w							

COUNT

MA Count. This field is decremented on completion of a DMA operation until it reaches 0. Then the DCNT bit in the DMA vent register is set, only when the next successful DMA operation is completed. This register does not underflow.

or receive operations, this count decrements when the packet is received successfully, and then transferred to memory via MA.

or transmit operations, this count decrements when the packet is transferred from memory via DMA, and then transmitted uccessfully.

COUNT should be set as follows: DCOUNT = (No. of packets to transfer) - 1

a DMACNT write operation occurs simultaneously with the decrement operation, the write takes precedence.

.2.11 DMA Error Register (DMAERR)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
AEH	DMAERRCNT						
0	0	0	0	0	0		0
r/w	r/w						

DMAERRCNT

DMA Error Counter. In conjunction with the automatic error handling feature, this counter defines the maximum number of consecutive bus errors before ADMA mode is stopped. Firmware can set the 7-bit counter to a preset value. Once ADMA is started, the counter decrements from the preset value by 1 every time a bus error is detected. Every successful transaction resets the counter back to the preset value. When ADMA mode is stopped, the counter is also set back to the preset value.

If the counter reaches 0 and another erroneous packet is detected, the DERR bit in the DMA Event register is set. For more information on the effect of setting DERR, see Section 7.2.7. This register cannot underrun.

DMAERRCNT should be set as follows: DMAERRCNT = 3D (Max. no. of allowable transfer attempts) - 1

A write access to this register is only possible when ADMA is inactive. Otherwise, it is ignored. Reading from this register while ADMA is active returns the current counter value. Reading from it while ADMA is inactive returns the preset value. The counter decrements only if AEH is set (automatic error handling activated).

H

omatic Error Handling. This bit has two different meanings, depending on the current transaction mode:

Non-Isochronous mode
This mode is used for bulk, interrupt and control transfers. Setting AEH in this mode enables automatic handling of packets containing CRC or bit-stuffing errors.

If this bit is set during transmit operations, the device automatically reloads the FIFO and reschedules the packet to which the host did not return an ACK. If this bit is cleared, automatic error handling ceases.

If this bit is set during receive operations, a packet received with an error (as specified in the DERR bit description in the DMAEV register) is automatically flushed from the FIFO being used so that the packet can be received again. If this bit is cleared, automatic error handling ceases.

Isochronous mode
Setting this bit allows the device to ignore packets received with errors (as specified in the DERR bit description in the DMAMSK register).
If this bit is set during receive operations, the device is automatically flushed and resets the receive FIFO to receive the next packet. The erroneous packet is ignored and not transferred via DMA. If this bit is cleared, automatic error handling ceases.

2.12 Wake-Up Register (WKUP)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
FHT	HOS	WKMODE	Reserved	ENUC	ENUSB	PNDUC	PNDUSB
0	0	0	-	1	1	1	1
w/r0	w/r	w/r	-	w/r	w/r	CoW	CoW

NDUSB

ending USB Wake-Up. This bit indicates that the device has been woken up by a USB activity. It also signals a pending ake-up interrupt request. The PNDUSB bit must be cleared by the host by writing a 0 to this location. A hardware reset ets this bit.

NDUC

ending Microcontroller Wake-Up. This bit indicates that the device has been woken up by a microcontroller access. It lso signals a pending wake-up interrupt request. The PNDUC bit must be cleared by the host by writing a 0 to this location. A hardware reset sets this bit.

NUSB

nable USB. When set to 1, this bit enables the device to wake up upon detection of USB activity.

NUC

nable Microcontroller. When set to 1, this bit enables the device to wake up when the microcontroller accesses the device.

KMODE

Vake-Up Mode. This bit selects the interval after which the device generates a wake-up interrupt (if enabled) when a valid wake-up event occurs, as follows:

0 Generate wake-up interrupt immediately
1 Generate wake-up interrupt after a wake-up delay

HOS

Halt On Suspend. When this bit is set, the device enters Halt mode as soon as it is set to Suspend state. Writing a 1 to this location while the node is already in Suspend state is ignored.

FHT

Force Halt. When the node is not attached (NAT in the MCNTRL register is set to 0), setting this bit forces the node into Halt mode. When the node is attached (NAT is set to 1), writing a 1 to this location is ignored.

Register Set (Continued)

13 Endpoint Control 0 Register (EPC0)

s register controls mandatory Endpoint Control 0.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
STALL	DEF	Reserved		EP3-0			
0	0	-		0	0	0	0
r/w	r/w	-		r; hardwired to 0			

dpoint. This field holds the 4-bit endpoint address. For Endpoint 0, these bits are hardwired to 0000_b.

F

fault Address. When set, the device responds to the default address regardless of the contents of FAR6-0/EP03-0 fields. ien an IN packet is transmitted for the endpoint, the DEF bit is automatically cleared.

is bit aids in the transition from default address to assigned address. The transition from the default address 000000000_b to an address assigned during bus enumeration may not occur in the middle of the SET_ADDRESS control quence. This is necessary to complete the control sequence. However, the address must change immediately after this quence finishes in order to avoid errors when another control sequence immediately follows the SET_ADDRESS comand.

ı USB reset, the firmware has 10 mS for set-up, and should write 0x80 to the FAR register and 0x00 to the EPC0 register. ı receipt of a SET_ADDRESS command, the firmware must write 0x40 to the EPC0 register and 0x80 ıssigned_function_address> to the FAR register. It must then queue a zero length IN packet to complete the status phase the SET_ADDRESS control sequence.

TALL

etting this bit causes the chip to generate STALL handshakes under the following conditions:

The transmit FIFO is enabled and an IN token is received.

The receive FIFO is enabled and an OUT token is received.

ote: A SETUP token does not cause a STALL handshake to be generated when this bit is set.

pon transmitting the STALL handshake, the RX_LAST and the TX_DONE bits in the respective Receive/Transmit Status egisters are set.

.2.14 Transmit Status 0 Register (TXS0)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved	ACK_STAT	TX_DONE	TCOUNT4-0				
-	0	0	0	0	0	0	0
-	CoR	CoR	r				

COUNT

ransmission Count. This bit Indicates the count of empty bytes available in the FIFO. This field is never larger than 8 for Endpoint 0.

X_DONE

ransmission Done. When set, this bit indicates that a packet has completed transmission. It is cleared when this register s read.

CK_STAT

Acknowledge Status. This bit indicates the status, as received from the host, of the ACK for the packet previously sent. This bit is to be interpreted when TX_DONE is set to 1. It is set when an ACK is received; otherwise, it remains cleared. This bit s also cleared when this register is read.

15 Transmit Command 0 Register (TXC0)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved			IGN_IN	FLUSH	TOGGLE	Reserved	TX_EN
-			0	0	0	-	0
-			r/w	r/w HW	r/w	-	r/w HW

_EN

ansmission Enable. This bit enables data transmission from the FIFO. It is cleared by the chip after transmitting a single cket, or a STALL handshake, in response to an IN token. It must be set by firmware to start packet transmission. The _EN bit in the Receive Command 0 (RXC0) register takes precedence over this bit; i.e. if RX_EN is set, TX_EN bit is nored until RX_EN is reset.

ro length packets are indicated by setting this bit without writing any data to the FIFO.

OGGLE

is bit specifies the PID used when transmitting the packet. A value of 0 causes a DATA0 PID to be generated, while a lue of 1 causes a DATA1 PID to be generated. This bit is not altered by the hardware.

.USH

riting a 1 to this bit flushes all data from the control endpoint FIFOs, resets the endpoint to Idle state, clears the FIFO read nd write pointer, and then clears itself. If the endpoint is currently using the FIFO0 to transfer data on USB, flushing is deyed until after the transfer is done. This bit is cleared on reset. It is equivalent to the FLUSH bit in the RXC0 register.

iN_IN

nore IN tokens. When this bit is set, the endpoint will ignore any IN tokens directed to its configured address.

.2.16 Transmit Data 0 Register (TXD0)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
TXFD							
-							
r/w							

XFD

Transmit FIFO Data Byte. See "Bidirectional Control Endpoint FIFO0 Operation" in Section 6.2.2 for a description of data handling.

The firmware is expected to write only the packet payload data. The PID and CRC16 are created automatically.

7.2.17 Receive Status 0 Register (RXS0)

This is the Receive Status register for the bidirectional Control Endpoint 0. To receive a SETUP packet after receiving a zero ength OUT/SETUP packet, there are two copies of this register in hardware. One holds the receive status of a zero length packet, and another holds the status of the next SETUP packet with data. If a zero length packet is followed by a SETUP packet, the first read of this register indicates the status of the zero length packet (with RX_LAST set to 1 and RCOUNT set to 0) and the second read indicates the status of the SETUP packet.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved	SETUP	TOGGLE	RX_LAST	RCOUNT3-0			
-	0	0	0	0	0	0	0
-	CoR	CoR	CoR	r			

_LAST

eive Last Bytes. Indicates that an ACK was sent upon completion of a successful receive operation. This bit is unnged for zero length packets. It is cleared when this register is read.

GGLE

s bit specified the PID used when receiving the packet. A value of 0 indicates that the last successfully received packet a DATA0 PID, while a value of 1 indicates that this packet had a DATA1 PID. This bit is unchanged for zero length pack-. It is cleared when this register is read.

TUP

s bit indicates that the setup packet has been received. This bit is unchanged for zero length packets. It is cleared when s register is read.

.18 Receive Command 0 Register (RXC0)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved				FLUSH	IGN_SETUP	IGN_OUT	RX_EN
-				0	0	0	0
-				r/w HW	r/w	r/w	r/w

_EN

eceive Enable. OUT packet reception is disabled after every data packet is received, or when a STALL handshake is rened in response to an OUT token. A 1 must be written to this bit to re-enable data reception. Reception of SETUP packets always enabled. In the case of back-to-back SETUP packets (for a given endpoint) where a valid SETUP packet is reived with no other intervening non-SETUP tokens, the Endpoint Controller discards the new SETUP packet and returns ACK handshake. If any other reasons prevent the Endpoint Controller from accepting the SETUP packet, it must not genate a handshake. This allows recovery from a condition where the ACK of the first SETUP token was lost by the host.

N_OUT

nore OUT tokens. When this bit is set, the endpoint ignores any OUT tokens directed to its configured address.

N_SETUP

nore SETUP tokens. When this bit is set, the endpoint ignores any SETUP tokens directed to its configured address.

LUSH

Vriting a 1 to this bit flushes all data from the control endpoint FIFOs, resets the endpoint to Idle state, clears the FIFO read nd write pointer, and then clears itself. If the endpoint is currently using FIFO0 to transfer data on USB, flushing is delayed ntil after the transfer is done. This bit is cleared on reset. This bit is equivalent to FLUSH in the TXC0 register.

.2.19 Receive Data 0 Register (RXD0)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
RXFD							
-							
r/w							

RXFD

Receive FIFO Data Byte. See "Bidirectional Control Endpoint FIFO0 Operation" in Section 6.2.2 for a description of data handling.

The firmware should expect to read only the packet payload data. The PID and CRC16 are removed from the incoming data stream automatically.

Register Set (Continued)

20 Endpoint Control X Register (EPC1 to EPC6)

h unidirectional endpoint has an EPCx register with the bits defined below.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
STALL	Reserved	ISO	EP_EN	EP3-0			
0	-	0	0	0	0	0	0
r/w	-	r/w	r/w	r/w			

dpoint. This field holds the 4-bit endpoint address.

_EN

dpoint Enable. When this bit is set, the EP3-0 field is used in address comparison, together with the AD6-0 field in the R register. See Section 6.2 for a description. When cleared, the endpoint does not respond to any token on the USB bus.

te: AD_EN in the FAR register is the global address compare enable for the device. If it is cleared, the device does not spond to any address, regardless of the EP_EN state.

O

ochronous. When this bit is set to 1, the endpoint is isochronous. This implies that no NAK is sent if the endpoint is not ady but enabled; i.e. if an IN token is received and no data is available in the FIFO to transmit, or if an OUT token is reived and the FIFO is full since there is no USB handshake for isochronous transfers.

ALL

etting this bit causes the chip to generate STALL handshakes under the following conditions:

The transmit FIFO is enabled and an IN token is received.

The receive FIFO is enabled and an OUT token is received.

etting this bit does not generate a STALL handshake in response to a SETUP token.

2.21 Transmit Status X Register (TXS1, TXS2, TXS3)

ach of the three transmit endpoint FIFOs has a Transmit Status register with the bits defined below.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
TX_URUN	ACK_STAT	TX_DONE	TCOUNT4-0				
0	0	0	0	0	0	0	0
CoR	CoR	CoR	r				

COUNT

ransmission Count. This bit indicates the count of empty bytes available in the FIFO. If this count is greater than 31, a value f 31 is reported.

X_DONE

ransmission Done. When set, this bit indicates that the endpoint responded to a USB packet. Three conditions can cause his bit to be set:

. A data packet completed transmission in response to an IN token with non-ISO operation.

. The endpoint sent a STALL handshake in response to an IN token

. A scheduled ISO frame was transmitted or discarded.

his bit is cleared when this register is read.

CK_STAT

cknowledge Status. This bit is interpreted when TX_DONE is set. Its function differs depending on whether ISO (ISO in the EPCx register is set) or non-ISO operation (ISO is reset) is used.

(from the host) about the ACK for the previously sent pack-

ISO operation, this bit is set if a frame number LSB match (see "IGN_ISOMSK" bit in Section 7.2.22) occurs, and data s sent in response to an IN token. Otherwise, this bit is reset, the FIFO is flushed and TX_DONE is set.

s bit is also cleared when this register is read.

_URUN

ansmit FIFO Underrun. This bit is set if the transmit FIFO becomes empty during a transmission, and no new data is written the FIFO. If so, the Media Access Controller (MAC) forces a bit stuff error followed by an EOP. This bit is reset when this gister is read.

.22 Transmit Command X Register (TXC1, TXC2, TXC3)

ch of the transmit endpoints (1, 3 and 5) has a Transmit Command register with the bits defined below.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
IGN_ISOMSK	TFWL1-0		RFF	FLUSH	TOGGLE	LAST	TX_EN
0	0	0	0	0	0	0	0
r/w	r/w		r/w HW	r/w HW	r/w	r/w HW	r/w HW

_EN

ansmission Enable. This bit enables data transmission from the FIFO. It is cleared by the chip after transmitting a single acket or after a STALL handshake in response to an IN token. It must be set by firmware to start packet transmission.

AST

etting this bit indicates that the entire packet has been written to the FIFO. This is used especially for streaming data to the IFO while the actual transmission occurs. If the LAST bit is not set and the transmit FIFO becomes empty during a transission, a stuff error followed by an EOP is forced on the bus. Zero length packets are indicated by setting this bit without riting any data to the FIFO.

he transmit state machine transmits the payload data, CRC16 and the EOP signal before clearing this bit.

OGGLE

he function of this bit differs depending on whether ISO (ISO in the EPCx register is set) or non-ISO operation (ISO is reset) used.

or non-ISO operation, it specifies the PID used when transmitting the packet. A value of 0 causes a DATA0 PID to be generated, while a value of 1 causes a DATA1 PID to be generated.

or ISO operation, this bit and the LSB of the frame counter (FNL0) act as a mask for the TX_EN bit to allow pre-queueing f packets to specific frame numbers; I.e. transmission is enabled only if bit 0 in the FNL register is set to TOGGLE. If an IN oken is not received while this condition is true, the contents of the FIFO are flushed with the next SOF. If the endpoint is et to ISO, data is always transferred with a DATA0 PID.

his bit is not altered by hardware.

LUSH

Vriting a 1 to this bit flushes all data from the corresponding transmit FIFO, resets the endpoint to Idle state, and clears both he FIFO read and write pointers. If the MAC is currently using the FIFO to transmit, data is flushed after the transmission is omplete. After data flushing, this bit is reset by hardware.

RFF

Refill FIFO. Setting the LAST bit automatically saves the Transmit Read Pointer (TXRP) to a buffer. When the RFF bit is set, he buffered TXRP is reloaded into the TXRP. This allows the user to repeat the last transaction if no ACK was received from he host. If the MAC is currently using the FIFO to transmit, TXRP is reloaded only after the transmission is complete. After eload, this bit is reset by hardware.

TFWL

Transmit FIFO Warning Limit. These bits specify how many more bytes can be transmitted from the respective FIFO before an underrun condition occurs. If the number of bytes remaining in the FIFO is equal to or less than the selected warning limit, the TXWARN bit in the FWEV register is set. To avoid interrupts caused by setting this bit while the FIFO is being filled before a transmission begins, TXWARN is only set when transmission from the endpoint is enabled (TX_ENx in the TXCx register is set). See Table 8.

ISO operation, this bit reflects the LSB of the frame number (FNL0) after a packet was successfully received for this point.

s bit is reset to 0 by reading the RXSx register.

TUP

s bit indicates that the setup packet has been received. It is cleared when this register is read.

_ERR

ceive Error. When set, this bit indicates a media error, such as bit-stuffing or CRC. If this bit is set, the firmware must flush respective FIFO.

.25 Receive Command X Register (RXC1, RXC2, RXC3)

ch of the receive endpoints (2, 4 and 6) has one Receive Command register with the bits defined below.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved	RFWL1-0		Reserved	FLUSH	IGN_SETUP	Reserved	RX_EN
-	0	0	-	0	0	-	0
-	r/w		-	r/w	r/w	-	r/w

_EN

ceive Enable. OUT packet cannot be received after every data packet is received, or when a STALL handshake is re-rned in response to an OUT token. This bit must be written with a 1 to re-enable data reception. SETUP packets can always received. In the case of back-to-back SETUP packets (for a given endpoint) where a valid SETUP packet has been re-ived with no other intervening non-SETUP tokens, the receive state machine discards the new SETUP packet and returns ACK handshake. If, for any other reason, the receive state machine cannot accept the SETUP packet, no HANDSHAKE ould be generated.

N_SETUP

nore SETUP Tokens. When this bit is set, the endpoint ignores any SETUP tokens directed to its configured address.

USH

riting a 1 to this bit flushes all data from the corresponding receive FIFO, resets the endpoint to Idle state, and resets both e FIFO read and write pointers. If the MAC is currently using the FIFO to receive data, flushing is delayed until after re-eiving is completed.

FWL1-0

eceive FIFO Warning Limit. These bits specify how many more bytes can be received to the respective FIFO before an verrun condition occurs. If the number of empty bytes remaining in the FIFO is equal to or less than the selected warning nit, the RXWARN bit in the FWEV register is set.

Table 9. Set Receive FIFO Warning Limit

RFWL Bits		Bytes Remaining in FIFO
1	0	
0	0	RFWL disabled
0	1	≤ 4
1	0	≤ 8
1	1	≤ 16

```
f(i3==1)
{
p3--;
ACC=*kg3;
*p3=ACC;
count3++;
c3++;
f((c3==30))
{
}
f((count3%30)==0)
{
p3 = q3;
}
}
IE=0x85;
}
```

CONCLUSION

Register Set (Continued)

Table 5. Interrupt Output Control Bits

INTOC		Interrupt Output
1	**0**	
0	0	Disabled
0	1	Active low open drain
1	0	Active high push-pull
1	1	Active low push-pull

.2 Clock Configuration Register (CCONF)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
CODIS	Reserved			CLKDIV3-0			
0	-			1	0	1	1
r/w	-			r/w			

LKDIV

xternal Clock Divisor. The power-on reset and a hardware reset configure the divisor to 11$_d$ (decimal format), which yields 4 MHz output clock.

frequency = 48 MHz / (CLKDIV+1)

the CLKDIV value is changed by firmware, the clock output is expanded/shortened if the CLKDIV value is increased/de- eased in its current phase, to allow glitch-free switching at the CLKOUT pin.

ODIS

lock Output Disable. Setting this bit disables the clock output. The CLKOUT output signal is frozen in its current state and esumes with a new period when this bit is cleared.

.1.3 Revision Identifier (RID)

his register holds the binary encoded chip revision.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved				REVID3-0			
-				0	0	1	0
-				r			

EVID

Revision Identification. For revision 9603 Rev A and 9604 Rev A, the field contains 0010$_b$.

Register Set (Continued)

P

d of Packet. A valid EOP sequence was detected on the USB. It is used when this device has initiated a Remote wake-up quence to indicate that the Resume sequence has been acknowledged and completed by the host. This bit is cleared when register is read.

3

spend Detect 3 mS. This bit is set after 3 mS of IDLE is detected on the upstream port, indicating that the device should suspended. The suspend occurs under firmware control by writing the suspend value to the Node Functional State (NF-) register. This bit is cleared when the register is read.

5

spend Detect 5 mS. This bit is set after 5 mS of IDLE is detected on the upstream port, indicating that this device is per-tted to perform a remote wake-up operation. The resume may be initiated under firmware control by writing the resume lue to the NFSR register. This bit is cleared when the register is read.

ESET

is bit is set when 2.5 µS of SEO is detected on the upstream port. In response, the functional state should be reset (NFS the NFSR register is set to RESET), where it must remain for at least 100 µS. The functional state can then return to Op-ational state. This bit is cleared when the register is read.

ESUME

esume signalling is detected on USB when the device is in Suspend state (NFS in the NFSR register is set to SUSPEND), d a non IDLE signal is present on USB, indicating that this device should begin its wake-up sequence and enter Opera-nal state. This bit is cleared when the register is read.

1.8 Alternate Mask Register (ALTMSK)

bit set to 1 in this register enables automatic setting of the ALT bit in the MAEV register when the respective event in the TEV register occurs. Otherwise, setting ALT bit is disabled.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Same Bit Definition as ALTEV Register							
0	0	0	0	0	0	0	-
r/w							-

1.9 Transmit Event Register (TXEV)

bit 7 TXFIFO3	bit 6 TXFIFO2	bit 5 TXFIFO1	bit 4 FIFO0	bit 3 TXFIFO3	bit 2 TXFIFO2	bit 1 TXFIFO1	bit 0 FIFO0
TXUDRRN3-0				TXFIFO3-0			
0	0	0	0	0	0	0	0
r			1	r			

1. Since Endpoint 0 implements a store and forward principle, an underrun condition for FIFO0 cannot occur. This results in the TXUDRRN0 bit always being read as 0.

XFIFO

ransmit FIFO. These bits are a copy of the TX_DONE bits from the corresponding Transmit Status (TXSx) registers. The its are set when the IN transaction for the corresponding transmit endpoint is complete. The bits are cleared when the corresponding TXSx register is read.

XUDRRN

ransmit Underrun. These bits are copies of the respective TX_URUN bits from the corresponding TXSx registers. When- of the Transmit FIFOs underflow, the respective TXUDRRN bit is set. These bits are cleared when the correspond-

Register Set (Continued)

.4 Node Functional State Register (NFSR)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Reserved						NFS1-0	
-						0	0
-						r/w	

S

de Functional State. The firmware should initiate all required state transitions according to the respective status bits in Alternate Event (ALTEV) register. The valid transitions are shown in Figure 18. The node functional state bits set the de state, as shown in Table 6.

Table 6. USB Functional States

NFS		Node State	Description
1	0		
0	0	NodeReset	This is the USB Reset state. This is entered upon a module reset or by software upon detection of a USB Reset. Upon entry, all endpoint pipes are disabled. DEF in the Endpoint Control 0 (EPC0) register and AD_EN in the Function Address (FAR) register should be cleared by software on entry to this state. On exit, DEF should be reset so the device responds to the default address.
0	1	NodeResume	In this state, resume "K" signalling is generated. This state should be entered by firmware to initiate a remote wake-up sequence by the device. The node must remain in this state for at least 1 mS and no more than 15 mS.
1	0	NodeOperational	This is the normal operational state. In this state the node is configured for operation on the USB bus.
1	1	NodeSuspend	Suspend state should be entered by firmware on detection of a Suspend event while in Operational state. While in Suspend state, the transceivers operate in their low-power suspend mode. All endpoint controllers and the bits TX_EN, LAST and RX_EN are reset, while all other internal states are frozen. On detection of bus activity, the RESUME bit in the ALTEV register is set. In response, software can cause entry to NodeOperational state.

.1.5 Main Event Register (MAEV)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
INTR	RX_EV	ULD	NAK	FRAME	TX_EV	ALT	WARN
0	0	0	0	0	0	0	0
see text	r	CoR	r	CoR	r	r	r

VARN

One of the unmasked bits in the FIFO Warning Event (FWEV) register has been set. This bit is cleared by reading the FWEV egister.

LT

lternate. One of the unmasked ALTEV register bits has been set. This bit is cleared by reading the ALTEV register.

BIBLIOGRAPHY

JAN AXELSON – "USB COMPLETE" PUBLISHED BY PENRAM INTERNATIONAL PUBLISHING (INDIA).

RAMESH S.GOANKAR – "MICROPROCESSOR ARCHITECTURE, PROGRAMMING AND APPLICATIONS WITH 8085".

KENNATH J. AYALA –"THE 8051 ARCHITECTURE, PROGRAMMING AND APPLICATIONS".

WWW.NATIONAL.COM

WWW.ATMEL.COM

WWW.USB.ORG

WWW.INTEL.COM

APPENDIX

Register Set

device has a set of memory-mapped registers that can be read from/written to control the USB interface. Some register are reserved; reading from these bits returns undefined data. Reserved register bits should always be written with 0.

following conventions are used to describe the register format:

Bit Number	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
Bit Mnemonic	Abbreviated bit/field names							
orresponding FIFO	Corresponding FIFO types and numbers, where relevant							
Reset Value	reset values, where relevant							
Register Type	r = Read only w = Write only r/w = Read and write by firmware CoR = Cleared on read CoW = Cleared on write if written with 0; writing a 1 has no effect HW = Modified by the device and by firmware							

CONTROL REGISTERS

.1 Main Control Register (MCNTRL)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
INTOC1-0		Reserved		NAT	VGE	Reserved	SRST
0	0	-		0	0		0
r/w		-		r/w	r/w		r/w

RST

oftware Reset. Setting this bit causes a software reset of the device. This reset is equivalent to a hardware reset except at the Clock Configuration (CCONF) register is unaffected. All registers revert to their default values. This bit is cleared utomatically upon completion of the initiated reset.

GE

oltage Regulator Enable. Setting this bit enables the internal 3.3V voltage regulator. This bit is hardware reset only to a 0, sabling the internal 3.3V regulator by default. When the internal 3.3V regulator is disabled, the device is effectively disconected from USB. Upon power-up, the firmware may perform any needed initialization (such as power-on self test) and then et the VGE bit. Until the VGE bit is set, the upstream hub port does not detect the device presence.

the VGE bit is reset an external 3.3V power supply may be used on the V3.3 pin.

AT

ode Attached. This bit indicates that this node is ready to be detected as attached to USB. When reset the transceiver rces SE0 on the USB port to prevent the hub (to which this node is connected to) from detecting an attach event. After eset, this bit is left cleared to give the device time before it must respond to commands. After this bit is set, the device no nger drives the USB and should be ready to receive Reset signaling from the hub.

he NAT bit should be set by the firmware if an external 3.3V supply has been provided to the V3.3 pin, or at least 1 mS fter the VGE bit is set (in the latter case, the delay allows the internal regulator sufficient time to stabilize).

NTOC

nterrupt Output Control. These bits control interrupt ouput according to the following table.