

AFFIDAVIT OF NATHANIEL E FRANK-WHITE

1. I am a Records Request Processor at the Internet Archive. I make this declaration of my own personal knowledge.
2. The Internet Archive is a website that provides access to a digital library of Internet sites and other cultural artifacts in digital form. Like a paper library, we provide free access to researchers, historians, scholars, and the general public. The Internet Archive has partnered with and receives support from various institutions, including the Library of Congress.
3. The Internet Archive has created a service known as the Wayback Machine. The Wayback Machine makes it possible to browse more than 450 billion pages stored in the Internet Archive's web archive. Visitors to the Wayback Machine can search archives by URL (i.e., a website address). If archived records for a URL are available, the visitor will be presented with a display of available dates. The visitor may select one of those dates, and begin browsing an archived version of the Web. Links on archived files in the Wayback Machine point to other archived files (whether HTML pages or other file types), if any are found for the URL indicated by a given link. For instance, the Wayback Machine is designed such that when a visitor clicks on a hyperlink on an archived page that points to another URL, the visitor will be served the archived file found for the hyperlink's URL with the closest available date to the initial file containing the hyperlink.
4. The archived data made viewable and browsable by the Wayback Machine is obtained by use of web archiving software that automatically stores copies of files available via the Internet, each file preserved as it existed at a particular point in time.
5. The Internet Archive assigns a URL on its site to the archived files in the format `http://web.archive.org/web/[Year in yyyy][Month in mm][Day in dd][Time code in hh:mm:ss]/[Archived URL]` aka an "extended URL". Thus, the extended URL `http://web.archive.org/web/19970126045828/http://www.archive.org/` would be the URL for the record of the Internet Archive home page HTML file (`http://www.archive.org/`) archived on January 26, 1997 at 4:58 a.m. and 28 seconds (1997/01/26 at 04:58:28). The date indicated by an extended URL applies to a preserved instance of a file for a given URL, but not necessarily to any other files linked therein. Thus, in the case of a page constituted by a primary HTML file and other separate files (e.g., files with images, audio, multimedia, design elements, or other embedded content) linked within that primary HTML file, the primary HTML file and the other files will each have their own respective extended URLs and may not have been archived on the same dates.
6. Attached hereto as Exhibit A are true and accurate copies of screenshots of the Internet Archive's records of the archived files for the URLs and the dates specified in the attached coversheet of each printout.



7. Attached hereto as Exhibit B are true and accurate copies of the Internet Archive's records of the archived files for the URLs and the dates specified in the following table.

Folder	URL
001	https://web.archive.org/web/20040417091747/http://www.netchip.com/library/spec/1080spec06.pdf
002	https://web.archive.org/web/20101230133512/http://www.fairchildsemi.com/ds/FS/FSUSB63.pdf

8. I declare under penalty of perjury that the foregoing is true and correct.

DATE: 08/31/2023

Nathaniel Frank-White

Nathaniel E Frank-White

Please see attached
All Purpose
Jurat form
for additional
Notary Events



EXHIBIT A





Browse Books by Subject

- > [Biological Sciences](#) (18)
- > [Computer and Information Science](#) (108)
- > [Earth Sciences](#) (15)
- > [Electrical and Electronic Engineering](#) (128)
- > **[Engineering](#) (86)**
 - > [Aerospace Engineering](#) (6)
 - > **[Automotive Industry](#) (5)**
 - > [Civil Engineering](#) (4)
 - > [Control Engineering](#) (11)
 - > [Energy Engineering](#) (14)
 - > [Engineering Management](#) (6)
 - > [Environmental Engineering](#) (17)
 - > [Industrial Engineering](#) (12)
 - > [Mechanical Engineering](#) (15)
 - > [Structural Engineering](#) (4)
 - > [Systems Engineering](#) (12)
 - > [Textile Engineering](#) (1)
- > [Materials Science](#) (27)
- > [Medicine](#) (17)
- > [Technology](#) (39)

[Engineering](#) > [Automotive Industry](#)

Automotive Industry



[Wind Tunnels](#)

Edited by Satoru Okamoto
ISBN 978-953-307-295-1
136 pages, February 2011

140 Downloads



[New Trends and Developments in Automotive Industry](#)

Edited by Marcello Chiaberge
ISBN 978-953-307-999-8
394 pages, January 2011

130 Downloads



[New Trends and Developments in Automotive System Engineering](#)

Edited by Marcello Chiaberge
ISBN 978-953-307-517-4
664 pages, January 2011

118 Downloads



[Urban Transport and Hybrid Vehicles](#)

Edited by Seref Soyulu
ISBN 978-953-307-100-8
192 pages, August 2010

620 Downloads



[Fuel Injection](#)

Edited by Daniela Slano
ISBN 978-953-307-116-9
254 pages, August 2010

685 Downloads

Publish with InTech

- ✓ High quality publications
- ✓ High impact and visibility
- ✓ Worldwide readership
- ✓ No copyright transfer
- ✓ Efficient publishing process
- ✓ Free access to research

[Publish Your Work](#)

Open Access



Freely Accessible Scientific Literature. Higher Impact. Global Reach.

[Find out more](#)

News

- [5th Anniversary of the First Book Published](#)
- [InTech Authors Recipients of IEEE Awards](#)
- [InTech Supports the Open Archives Initiative](#)
- [InTech Presents InTechOpen](#)

[About Us](#)

[About InTech](#)

[Open Access](#)

[About Open Access](#)

[For Authors](#)

[Publish with InTech](#)

[Collaborations](#)

[Become a Book Editor](#)

[Top](#)



[Our Story](#)
[Facts and Figures](#)
[Social Responsibility](#)
[Scientific Board](#)
[Blog](#)
[InTechOpen](#)
[Jobs at InTech](#)
[Contact](#)

[Open Access Resources](#)
[OA Research Funders](#)
[For Libraries](#)
[OAI-PMH](#)

[Benefits for Authors](#)
[Invited Authors](#)
[Start Your Own Book](#)
[Propose a Book Title](#)
[Participate in a Book Project](#)
[Contribute to a Journal](#)
[Start Your Own Journal](#)
[FAQs](#)

[Become a Reviewer](#)
[Share Free Knowledge](#)
[Buy Print Copies](#)



[Copyright Policy](#) | [Terms of Use](#) | [Privacy Policy](#)
InTech - Open Access Publisher © 2004 - 2011





Browse By Subject

- > Biological Sciences (18)
- > Computer and Information Science (108)
- > Earth Sciences (15)
- > Electrical and Electronic Engineering (128)
- > Engineering (86)
 - > Aerospace Engineering (6)
 - > Automotive Industry (5)
 - > Civil Engineering (4)
 - > Control Engineering (11)
 - > Energy Engineering (14)
 - > Engineering Management (6)
 - > Environmental Engineering (17)
 - > Industrial Engineering (12)
 - > Mechanical Engineering (15)
 - > Structural Engineering (4)
 - > Systems Engineering (12)
 - > Textile Engineering (1)
- > Materials Science (27)
- > Medicine (17)
- > Technology (39)

Share the Knowledge!



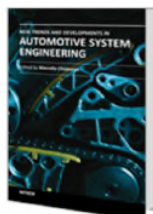
citeulike

404 Not Found

nginx/1.25.1

Engineering > Automotive Industry

New Trends and Developments in Automotive System Engineering



Edited by: Marcello Chiaberge

ISBN 978-953-307-517-4, Hard cover, 664 pages

Publisher: InTech

Publication date: January 2011

Subject: Automotive Industry

In the last few years the automobile design process is required to become more responsible and responsibly related to environmental needs. Basing the automotive design not only on the appearance, the visual appearance of the vehicle needs to be thought together and deeply integrated with the "power" developed by the engine. The purpose of this book is to try to present the new technologies development scenario, and not to give any indication about the direction that should be given to the research in this complex and multi-disciplinary challenging field.



Download
zipped PDF
55.25 Mb

Downloaded 119 times

Bookmark to: citeulike

Table of Contents

Experimental Test of a Diesel Engine Using Envo-Diesel as an Alternative Fuel

M.A.Kalam and H.H. Masjuki

Analytical Methods for Determining Automotive Fuel Composition

Jonas Gruber, Renata Lippl, Rosamaria W. C. Li and Adriano R. V. Benvenho

Automotive Fuel Consumption in Brazil: Applying Static and Dynamic Systems of Demand Equations

Mariana Iootty, Helder Pinto Jr. and Francisco Ebeling

Fatigue and Fracture Behavior of Forging Die Steels

Ryuichiro Ebara

Optimization of Injection Moulded Polymer Automotive Components

Ribeiro, C.J. and Viana, J.C.

High Mn TWIP Steels for Automotive Applications

B. C. De Cooman, Kwang-geun Chin and Jinkyung Kim

Powder Injection Moulding - An Alternative Processing Method for Automotive Items

Berenika Hausnerova

Thermomechanical and Isothermal Fatigue Behavior of Gray Cast Iron for Automotive Brake Discs

Omar Maluf, Jéferson Aparecido Moreto, Maurício Angeloni, Marco Antônio Colósio, José Carlos Santos, Waldek Wladimir Bose Filho and Dirceu Spinelli

Advanced Robotic Radiative Process Control for Automotive Coatings

Fan Zeng and Beshah Ayalew

DC/DC Step-Up Converters for Automotive Applications: a FPGA Based Approach

M. Chiaberge, G. Botto and M. De Giuseppe

The Thermo-Mechanical Behavior in Automotive Brake and Clutch Systems

Abdullah M. Al-Shabibi



[Dynamic Analysis of an Automotive Lower Suspension Arm using Experiment and Numerical Technique](#)

S. Abdullah, N.A. Kadhim, A.K. Ariffin and M. Hosseini

[Increased Cooling Power with Nucleate Boiling Flow in Automotive Engine Applications](#)

Helfried Steiner, Günter Brenn, Franz Ramstorfer and Bernd Breitschädel

[The 'Equivalent Cable Bundle Method': an Efficient Multiconductor Reduction Technique to Model Industrial Cable Networks](#)

Guillaume Andrieu, Xavier Bunlon, Lamine Koné, Jean-Philippe Parmantier, Bernard Démoulin and Alain Relneix

[Fatigue Characteristics of Automotive Jounce Bumper](#)

Aldy Ali, R.S. Sidhu and M.S.A. Samad

[On the Control of Automotive Traction PEM Fuel Cell Systems](#)

Ahmed Al-Durra, Stephen Yurkovich and Yann Guezennec

[An Adaptive Two-Stage Observer in the Control of a New Electromagnetic Valve Actuator for Camless Internal Combustion Engines](#)

Paolo Mercorelli

[Integrated Controller Design for Automotive Semi-Active Suspension Considering Vehicle Behavior with Steering Input](#)

Masaki Takahashi, Takashi Kumamaru and Kazuo Yoshida

[Design of an Embedded Controller for Some Applications of an Automotives](#)

Preeti Bajaj and Dinesh Padole

[Arbitration Schemes for Multiprocessor Shared Bus](#)

Preeti Bajaj and Dinesh Padole

[Towards Automotive Embedded Systems with Self-X Properties](#)

Gereon Weiss, Marc Zeller and Dirk Eilers

[4D Ground Plane Estimation Algorithm for Advanced Driver Assistance Systems](#)

Faisal Mufti, Robert Mahony and Jochen Heinzmann

[The Car Entertainment System](#)

Niels Koch

[Information and Communication Support for Automotive Testing and Validation](#)

Mathias Johanson

[Trends Towards Automotive Electronic Vision Systems for Mitigation of Accidents in Safety Critical Situations](#)

Ciarán Hughes, Ronan O'Malley, Diarmaid O'Cualain, Martin Glavin and Edward Jones

[Advancements in Automotive Antennas](#)

Brendan D. Pell, Edin Sulic, Wayne S. T. Rowe, Kamran Ghorbani and Sabu John

[Automotive VHDL-AMS Electro-Mechanics Simulations](#)

Mariagrazia Graziano and Massimo Ruo Roch

[Potential and Drawbacks of Raman \(Micro\)spectrometry for the Understanding of Iron and Steel Corrosion](#)

Philippe Colombari

[Computational Techniques for Automotive Antenna Simulations](#)

Falk Bogdanov, Roman Jobava, David Karkashadze, Paata Tsereteli, Anna Gheonjian, Ekaterina Yavolovskaya, Detlef Schleicher, Christoph Ullrich and Hicham Tazi

[Voltage Stability Analysis of Automotive Power Nets based on Modeling and Experimental Results](#)

Tom P. Kohler, Rainer Gehring, Joachim Froeschl, Dominik Buechert and Hans-Georg Herzog

[Urban and Extra Urban Vehicles: Re-Thinking the Vehicle Design](#)

Andrea Festini, Andrea Tonoli and Enrico Zenerino

[Analysis Approach of How University Automotive Competitions Help](#)



[About InTechOpen](#)

[About us](#)
[About InTech](#)
[Link to us](#)
[Contact](#)

[Open Access](#)

[About Open Access](#)
[How to reference](#)
[For Libraries](#)
[OAI-PMH](#)
[InTechOpen Copyright](#)
[Policy](#)

[Connect with us](#)

[Blog](#)
[Twitter](#)
[Facebook](#)
[YouTube](#)

© 2004 - 2011 InTech - Open Access Publisher



<https://web.archive.org/web/20070826063456/http://www.epn-online.com:80/topProduct/liste/cat/Digital+ICs/subCat/ASICs%2C+ASSPs+and+Gate+Arrays>





Get direct access via **EPNdirect** to Europe's most comprehensive database of electronic products & suppliers

Search Insert keyword(s) or reference [Advanced Search Criteria](#)

[Login / Register](#) [Media Information 2007](#)

[Home](#) | [Products](#) | [Suppliers](#) | [Events](#) | [Datasheet](#) | [Subscriptions](#)

You are in : > [HOME](#) > [TOP PRODUCTS](#) > [Group: Digital ICs](#) > [Sub-group: ASICs, ASSPs and Gate Arrays](#)



EPN Magazine
Issue August 2007
Sample issue

FREE subscribe now

TOP PRODUCTS

Product group : Digital ICs
Product Sub-group : ASICs, ASSPs and Gate Arrays

Display:



Low Bit-Rate RALCWI Vocoder IC
(17/08/2007)
The CMX618 is a flexible, half-duplex, highly integrated RALCWI (Robust Advanced Low Complexity Waveform Interpolation) vocoder IC suited to both wireless and wireline voice applications such as voice scramblers, alarm systems, military field applications and digital professional... [More](#)



Embedded Array
(01/06/2007)
Leveraging the company's expertise in structured-ASIC technology, ChipX has introduced a series of embedded-array products built using UMC's 0.13µm high-speed process. These arrays allow engineers to avoid opting for a fully customised standard-cell design, which are often expens... [More](#)

MCU Solution
(01/05/2007)
Innovasic has now announced the availability of a TCP/IP stack for its established fido1100 series of industrial microcontroller devices. The stack will allow users to configure one of the microcontroller's UICs (Universal I/O Controllers) to communicate with other semiconductor ... [More](#)



Single-Chip UMTS Solution
(01/03/2007)
The QSC6240 chip for WEDGE (WCDMA and GSM/GPRS/EDGE) and the QSC6270 product for HEDGE (HSDPA/WCDMA and GSM/GPRS/EDGE) are the world's first single chips to combine a radio transceiver, a baseband modem and a multimedia processor together with power management functionality onto ... [More](#)

55nm Cell-Based IC
(01/03/2007)
The CB-55L is a cell-based integrated circuit built on NEC's 55nm UX7LS process technology. The ASIC demonstrates a 40% reduction in power consumption from the previous 90nm CB-90M. The CB-55L has more than twice the density of previous devices, fitting 925,000 gates per mm2. It ... [More](#)

Video-Processing IC
(01/03/2007)
The ABT2010 video-processing IC has been designed for format-conversion applications in HDTVs, DVD, HD-DVD, and Blu-Ray player/recorders, and AV receivers. Features of the video-processing chip include 10-bit precision video scaling that can independently scale an image horizontally... [More](#)



Mobile TV Chips
(01/03/2007)
The DIB908x-H has been designed to offer a cost-effective DVB-H-only solution for integration into mobile phones. The IC includes a demodulator, a second-generation multi-band RF tuner, an LNA and all filters while being only half as big its predecessor. A multi-mode, multi-band ... [More](#)

eASIC, ASIC Architect join hands to deliver PCI Express (PCIe) Endpoint Controller and DDR2 memory controller
(08/02/2007)
eASIC Corporation, a provider of Structured ASIC devices and Configurable logic IP, and ASICArchitect, a leading supplier of high speed IP cores, have unveiled the immediate availability of two new high-speed interfaces for eASIC's 90nm Nextreme Structured ASIC Family. PCI Express... [More](#)



Low-Cost Cell-Phone Platform
(01/02/2007)
Targeting the market's largest segment, the TrueENTRY X122 low-cost cellular-phone platform consists of silicon chips, software (including Agere's proven protocol stack and frameworks) and a development kit. It platform adheres to the General Packet Radio Service (GPRS) cellular s... [More](#)

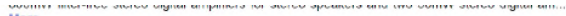


USB In Automotive: No Longer Just For PCs
(01/09/2006)
Media inside the automobile has followed in the footsteps of consumer electronics. From eight-track players to cassettes to CDs, it's clear that MP3 is next in line. Well over 150 million consumers have an iPod or MP3-equivalent, and FM transmitters and factory-installed iPod doc... [More](#)



Digital Audio Processor
(01/09/2006)
Aimed at mobile applications, the 5x5mm PSM711LQ is a high-performance sound processor combined with a digital power amplifier. It supports 1.4W in parallel mono speaker drive mode, two 0.7Wx1M filterless stereo digital amplifiers for stereo speakers and two 0.7Wx1M stereo digital am





Structured-ASIC Prototyping

(01/08/2006)

Gate Array In Tiny WCSP

(01/06/2006)

QDR-II Dual-Port Interfaces

(01/06/2006)

Digital-Audio-Codec Processor

(01/06/2006)

Embedded Flash Enables New Sensor-Interface Applications in Vehicles

(01/05/2006)

Embedded LCD Controller

(01/05/2006)

Multi-Gbit/s SerDes PHY

(01/04/2006)

Integrated SoC Solution

(01/04/2006)

Embeddable FPGA Cores

(01/04/2006)

Cached DSP Cores

(01/04/2006)

ASIC-Prototyping Platform

(01/04/2006)

Tools And IP Package

(01/04/2006)

Driving Medical-Device Innovation

(01/04/2006)

Optical Modulator Driver

(01/02/2006)

[More](#)

Page 1 2 3 4 to approx 4

Search in the
archives

Year

Month



Advanced Search
Criteria

[About EPN](#) | [Terms and conditions](#) | [Privacy policy](#) | [Contact us](#) | [Media information 2007](#)

Copyright RBI 2005. EPN is a trademark of Reed Business Information

Our group proposes on-line information covering the following topics

Business to Business

Business Directory

Social care - Director of social institutes

Electronics - Products - Design - Components

Corporate Publishing

Finance

Business to Consumer

Legal - Legal

Marketing - Communication - Medias

Management - Personal and professional development

HR - Training - Education

On-line invitations to tender - Data services

Health

[Home](#) | [Products](#) | [Suppliers by company / by product type](#) | [Events](#) | [Subscription to Datasheet / to Magazine](#)

EPN Online proposes a technical database about discrete semiconductors components, connectors, electromechanical components, sensors and transducers, passive electronic components and provides news about electronics design and actuality of electronics suppliers



<https://web.archive.org/web/20111112082401/http://www.epn-online.com:80/page/36489/usb-in-automotive-----no-longer-just-for-pcs.html>





Get direct access via **EPNdirect** to Europe's most comprehensive database of electronic products & suppliers

Search [Insert keyword(s) or reference] GO Advanced Search Criteria Login / Register Media information

Home | Products | Suppliers | Events | Datasheets | Magazine

You are in: > HOME > TOP PRODUCTS > USB In Automotive: No Longer Just For PCs

Sitemap

TOP PRODUCTS

Print | Digg This | Stashdot It | Add to Del.icio.us |

Product group : Digital ICs
Product Sub-group : ASICs, ASSPs and Gate Arrays

USB In Automotive: No Longer Just For PCs

Media inside the automobile...

Universal Serial Bus (USB) has long been the ubiquitous standard for PC peripheral connectivity, but recently it has started venturing into the automotive market. MP3-supported stereos, DVD players and advanced GPS navigation systems are all part of infotainment automotive electronics. Equipping our cars with these high-tech gadgets requires a sophisticated, yet simple, form of communication.

Reference: 16896

01/09/2006

Media inside the automobile has followed in the footsteps of consumer electronics. From eight-track players to cassettes to CDs, it's clear that MP3 is next in line. Well over 150 million consumers have an iPod or MP3-equivalent, and FM transmitters and factory-installed iPod docks are simply a band-aid until stereos incorporate direct connections. Infotainment also includes touch-screen navigation equipment, fold-down video screens and mobile-phone speaker systems: suddenly the car has become a PC on wheels. As more portable consumer electronics interface with infotainment units, a standard form of connectivity is needed that can work with nearly all devices. This same dilemma existed in the PC market over ten years ago, which led to the birth of USB. The automotive industry does not need to reinvent the wheel: nearly all consumer electronics are PC-dependent for some form of functionality (downloading MP3s, uploading pictures, etc.), and the connection most widely used is USB - so the automotive industry already has a universal standard to take advantage of.

Infotainment design using USB

To utilise USB inside the automobile, either the portable consumer device or the infotainment system must contain a USB host to control the flow of data. The USB host in these devices is not the typical host controller in your PC, but rather an optimised ASIC or embedded host IC targeted for portable consumer electronics. High-end infotainment systems (advanced GPS navigation, full-featured stereos) often contain an ASIC customised to perform some dedicated function. Since these ASICs are typically built from IP libraries, they often integrate USB host IP to support USB peripherals. Simpler systems (basic stereos, mobile phone docks) do not require as much processing power and may choose to use a very basic ASIC or an embedded microprocessor. In these cases, a USB host can be added using a discrete embedded host IC. Using such an embedded host allows the ASIC to slim down its feature set to save space and cost, while still supporting USB. A discrete embedded host can also allow quicker time-to-market. Whether a designer chooses to use an integrated or a discrete solution, all USB hosts require a host stack. The latter is essentially the brains to the host controller, enabling all USB functionality inside a consumer application. A typical host stack contains the application-programming interface, a real-time operating system and core USB drivers. When designing a USB host into an infotainment system, the application and usage model will define the necessary bandwidth requirements. For most automotive applications, full-speed USB (12 Mbps) is sufficient. For example, the average MP3 is compressed between 128 and 192Kbps, with the highest quality MP3 pushing 320Kbps, well below the 12Mbps full-speed USB can support. Even uncompressed WAV files stream from a CD transfer at a maximum data rate of 1.5Mbps, which again falls well into the range of full-speed USB. High-speed USB (480Mbps) should only be considered if the system has 1GByte or above of on-board storage or streams high-resolution video. Automotive infotainment systems must be prepared to operate in the most extreme temperatures. To ensure proper functionality of all automotive devices in all conditions, the Automotive Electronics Council (AEC) has defined a specification known as the AEC Q100, which calls for automotive components to operate in a temperature range from -40 to +150°C, with varying grades in between. Certain car manufacturers may require all components in an infotainment system to comply with a certain grade outlined in the AEC Q100 specifications.

The USB solution

With many specifications and requirements to consider, choosing the right USB solution can be a challenge. Cypress Semiconductor's EZ-Host (CY7C67300) embedded host solution is an AEC-qualified device with whole product support. Combined with the CY4665 reference design, Cypress thus provides a turn-key solution that allows quick time-to-market and easy customisation for all infotainment platforms. Complete firmware, schematics, host stacks and documentation put control in the designer's hands.

Cypress Semiconductor
3901 North First Street
PO Box 649079
95164-9079 San Jose - USA -California
tel: +1 408 943 2600
fax: +1 408 943 2796

Company website

RELATED ARTICLES FROM Cypress Semiconductor

■ Design Tool For PSoC (28/06/2007)

RELATED ARTICLES

- Two Non-Magnetic TVS Arrays (24/10/2011)
- Automotive-Qualified FPGAs (22/04/2010)
- Super-resolution ASSP supports LVDS interface (25/02/2010)
- CSSPs in HSDPA USB Modems (28/01/2010)
- Security modules feature multiple interfaces (24/12/2009)

back to top

Search in the archives

[Insert keyword(s) or reference]

Year

Month

GO Advanced Search Criteria

About EPN | Terms and Conditions | Privacy policy | Contact us | Media information | Distiblog









Get direct access via **EPNdirect** to Europe's most comprehensive database of electronic products & suppliers

Search Insert keyword(s) or reference GO Advanced Search Criteria Login / Register Media information

[Home](#) | [Products](#) | [Suppliers](#) | [Events](#) | [Datasheets](#) | [Magazine](#)

You are in: > [HOME](#) > [TOP PRODUCTS](#) > [USB In Automotive: No Longer Just For PCs](#)

[Sitemap](#)

TOP PRODUCTS

[Print](#) | [Digg This](#) | [Stashdot IT](#) | [Add to Del.icio.us](#) |

Product group : Digital ICs
Product Sub-group : ASICs, ASSPs and Gate Arrays

USB In Automotive: No Longer Just For PCs

Media inside the automobile...

Universal Serial Bus (USB) has long been the ubiquitous standard for PC peripheral connectivity, but recently it has started venturing into the automotive market. MP3-supported stereos, DVD players and advanced GPS navigation systems are all part of infotainment automotive electronics. Equipping our cars with these high-tech gadgets requires a sophisticated, yet simple, form of communication.

Reference: 18896

01/09/2006

Media inside the automobile has followed in the footsteps of consumer electronics. From eight-track players to cassettes to CDs, it's clear that MP3 is next in line. Well over 150 million consumers have an iPod or MP3-equivalent, and FM transmitters and factory-installed iPod docks are simply a band-aid until stereos incorporate direct connections. Infotainment also includes touch-screen navigation equipment, fold-down video screens and mobile-phone speaker systems: suddenly the car has become a PC on wheels. As more portable consumer electronics interface with infotainment units, a standard form of connectivity is needed that can work with nearly all devices. This same dilemma existed in the PC market over ten years ago, which led to the birth of USB. The automotive industry does not need to reinvent the wheel: nearly all consumer electronics are PC-dependent for some form of functionality (downloading MP3s, uploading pictures, etc.), and the connection most widely used is USB - so the automotive industry already has a universal standard to take advantage of.

Infotainment design using USB

To utilise USB inside the automobile, either the portable consumer device or the infotainment system must contain a USB host to control the flow of data. The USB host in these devices is not the typical host controller in your PC, but rather an optimised ASIC or embedded host IC targeted for portable consumer electronics. High-end infotainment systems (advanced GPS navigation, full-featured stereos) often contain an ASIC customised to perform some dedicated function. Since these ASICs are typically built from IP libraries, they often integrate USB host IP to support USB peripherals. Simpler systems (basic stereos, mobile phone docks) do not require as much processing power and may choose to use a very basic ASIC or an embedded microprocessor. In these cases, a USB host can be added using a discrete embedded host IC. Using such an embedded host allows the ASIC to slim down its feature set to save space and cost, while still supporting USB. A discrete embedded host can also allow quicker time-to-market. Whether a designer chooses to use an integrated or a discrete solution, all USB hosts require a host stack. The latter is essentially the brains to the host controller, enabling all USB functionality inside a consumer application. A typical host stack contains the application-programming interface, a real-time operating system and core USB drivers. When designing a USB host into an infotainment system, the application and usage model will define the necessary bandwidth requirements. For most automotive applications, full-speed USB (12 Mbps) is sufficient. For example, the average MP3 is compressed between 128 and 192Kbps, with the highest quality MP3 pushing 320Kbps, well below the 12Mbps full-speed USB can support. Even uncompressed WAV files stream from a CD transfer at a maximum data rate of 1.5Mbps, which again falls well into the range of full-speed USB. High-speed USB (480Mbps) should only be considered if the system has 1GByte or above of on-board storage or streams high-resolution video. Automotive infotainment systems must be prepared to operate in the most extreme temperatures. To ensure proper functionality of all automotive devices in all conditions, the Automotive Electronics Council (AEC) has defined a specification known as the AEC Q100, which calls for automotive components to operate in a temperature range from -40 to +150°C, with varying grades in between. Certain car manufacturers may require all components in an infotainment system to comply with a certain grade outlined in the AEC Q100 specifications.

The USB solution

With many specifications and requirements to consider, choosing the right USB solution can be a challenge. Cypress Semiconductor's EZ-Host (CY7C67300) embedded host solution is an AEC-qualified device with whole product support. Combined with the CY4645 reference design, Cypress thus provides a turn-key solution that allows quick time-to-market and easy customisation for all infotainment platforms. Complete firmware, schematics, host stacks and documentation put control in the designer's hands.

Cypress Semiconductor
3901 North First Street
PO Box 649079
95164-9079 San Jose - USA -California
tel: +1 408 943 2600
fax: +1 408 943 2796

[Company website](#)

RELATED ARTICLES FROM Cypress Semiconductor

■ [Design Tool For PSoC](#) (28/08/2007)

RELATED ARTICLES

- [Two Non-Magnetic TVS Arrays](#) (24/10/2011)
- [Automotive-Qualified FPGAs](#) (22/04/2010)
- [Super-resolution ASSP supports LVDS interface](#) (25/02/2010)
- [CSSPs in HSDPA USB Modems](#) (28/01/2010)
- [Security modules feature multiple interfaces](#) (24/12/2009)

[back to top](#) ▲

Search in the archives

Insert keyword(s) or reference

Year

Month

GO

Advanced Search Criteria

[About EPN](#) | [Terms and Conditions](#) | [Privacy policy](#) | [Contact us](#) | [Media information](#) | [Distiblog](#)

EPN Online is a registered trademark of EPN Online. All rights reserved. EPN Online is not responsible for the content of any external links.









Home > Documentation > Technical Articles

USB In Automotive- Not Just For PCs Anymore

Last Updated: 11/13/2008

Universal Serial Bus (USB) has long been the ubiquitous standard for PC peripheral connectivity, but recently USB has started venturing into the automotive market. MP3-supported stereos, DVD players and advanced GPS navigation systems are all part of "infotainment" automotive electronics designed to provide "information" and "entertainment" to passengers. Equipping our cars with these high-tech gadgets requires a sophisticated, yet simple, form of communication - a role USB has been performing its whole life. To read more about this article, click the download link or go to [EPN](#).

Rate Technical Ar..

Be the first to rate

Related Pages

[Automotive EZ-Host™](#)
[USB Hi-Speed Peripherals](#)

Related Files

File Title	Language	File Size	Last Updated
Embedded Host in Automotive.pdf	English	19 KB	02/26/2009
USB In Automotive Not Just For PCs Anymore.pdf	Korean	889 KB	02/26/2009
Embedded Host in Automotive_v3_j.pdf	Japanese	247 KB	02/26/2009
USB e Automotive qualcosa di pi.pdf	Italian	629 KB	02/26/2009
USB jetzt auch im Auto_Dec. 06.pdf	German	159 KB	02/26/2009
L'USB dans l'automobile pas seulement pour les ordinateurs.pdf	French	49 KB	02/26/2009

Need help? Ask a question and find answers in the [Cypress Developer Community Forums](#).

Low/intermittent bandwidth users tip: Firefox and Chrome browsers will allow downloads to be resumed if your connection is lost during download.

Share

Tweet

Digg +

SHARE





To download a document, point at the hyperlink, right click on the mouse, and then choose "Save Target As" or "Save Link As". If you still can't download, please contact support@netchip.com.

- [NET1080](#): NET1080 Specification Rev 0.6
- [NET1031](#): ScanUSB - Single-Chip USB Scanner Controller Rev 1.0
- [NET2888](#): TurboUSB - USB Interface Controller Specification. Rev 3.0
- [NET2890](#): USB Programmable Interface Controller Specification Rev 2.0 Draft 9





[Overview](#)
[Key Features and Benefits](#)
[Standard Specifications](#)



TurboCONNECT Reference Design

Overview

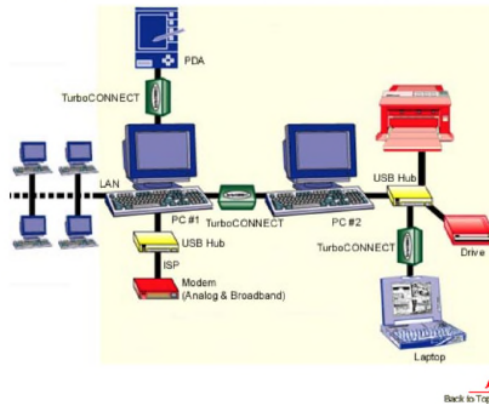
Complete Turnkey Reference Design with NetChip's NET1080 and device drivers.

TurboCONNECT is a turnkey Reference Design Kit that comes complete with documentation, bill of materials, schematics, and device drivers. OEMs can take it to manufacturing directly without spending any time in hardware and software development.

TurboCONNECT lets you offer a quick and easy solution for host to host communications at turbo speed equivalent to an Ethernet 10BT connection. Users can now network two or more Universal Serial Bus (USB) computers through standard Plug & Play technology without adding network cards.

With TurboCONNECT, two or more USB computers can share programs, files, peripheral equipment, and more. The product is perfect for small offices, home users, multi-player gaming, or business people who take their work home or on the road. They can work on their notebook PC, then connect it to a desktop PC via TurboCONNECT, and transfer files.

The standard TurboCONNECT includes two USB upstream connectors and allows the connected PCs to be separated by up to 10 meters for flexible placement.



Key Features and Benefits:

- Up to 12Mbps data transfer rate
- Full NDIS implementation
- Multiple network nodes supported
- Plug & Play compatible
- LED's for status and activity
- OEM software and hardware customization

Standard Specifications:

- Operating Environment: Windows98, Windows CE, and Windows 2000
- OCHI and UCHI compatible
- Ports: Two upstream, USB type B receptacle
- Compliance: FCC Class B, USB specification v1.0 and v1.1
- Power Supply: Powered from USB bus, no external supply required
- Power Management: Suspend current reduces to <500 µa
- Dimensions: 0.5"H x 1.0"W x 2.13"L

For more information on the TurboCONNECT Reference design, please [contact NetChip](#).

[Back to Top](#)



EXHIBIT B

Folder 001



Technology, Inc.

335 Pioneer Way

Mt View, California 94041

(650) 526-1490 Fax (650) 526-1494

e-mail: sales@netchip.com

Internet: www.netchip.com

NET1080 USB TurboCONNECT PRELIMINARY SPECIFICATION

For Revision 2 IC

Doc #: 605-1080-0206

Revision: 0.6

Date: 1/31/00

This document contains material that is confidential to NetChip. Reproduction without the express written consent of NetChip is prohibited. All reasonable attempts were made to ensure the contents of this document are accurate; however no liability, expressed or implied is guaranteed. NetChip reserves the right to modify this document, without notification, at any time.

Revision History

Revision	Issue Date	Comments
0.1	February 16, 1999	NET1080 Initial Draft Specification Proposal
0.2	April 22, 1999	New Updates
0.3	May 4, 1999	Register, Pin Changes
0.4	June 23, 1999	Pin Changes
0.5	November 8, 1999	Additional Pin Changes, Typo Fixes
0.6	January 31, 2000	Removing I2C EEPROM section

NET1080 USB TurboCONNECT

1	Highlights	5
1.1	Introduction	5
1.2	Features	5
1.2.1	General Features	5
1.2.2	USB Features.....	5
1.3	Overview.....	5
1.4	NET1080 Block Diagram.....	6
1.5	NET1080 System Block Diagram.....	6
	Changes from Rev 1 to Rev 2.....	6
	Changes from Rev 1 to Rev 2.....	7
2	Pin Connection Diagram.....	8
3	Pin Description	9
3.1	Table 3-1: Pin Types	9
3.2	Table 3-2: Pin Descriptions.....	9
4	Functional Description	12
4.1	Start-Up.....	12
4.1.1	Reading from Initialization Device	12
4.1.2	Enumeration and Configuration by the Host PC	12
4.2	Initialization Interface	12
4.2.1	MicroWire EEPROMs.....	12
4.3	USB Interface.....	14
4.3.1	Default Control Endpoint (Endpoint 00h).....	14
4.3.2	Data OUT Endpoint (Endpoint 01h).....	14
4.3.3	Data IN Endpoint (Endpoint 81h).....	14
4.3.4	Mailbox OUT Endpoint (Endpoint 02h).....	14
4.3.5	Mailbox IN Endpoint (Endpoint 82h).....	14
4.3.6	Status Endpoint (Endpoint 83h).....	14
4.4	Suspend Mode	14
4.5	Root Port Reset.....	15
4.6	NET1080 Power Configuration.....	15
5	Register Descriptions.....	16
	USB Register Description.....	16
5.1.1	USB Register Summary	16
5.1.2	(Address 0h; MFGID) Manufacturer ID	16
5.1.3	(Address 1h; PRODID) Product ID.....	16
5.1.4	(Address 2h; RELNUM) Release Number	17
5.1.5	(Address 3h; MAXPWR) Maximum Power.....	17
5.1.6	(Address 4h; USBCTL) USB Control Register.....	18
5.1.7	(Address 5h; INTPOLL) Interrupt Polling Interval Register.....	18
5.1.8	(Address 6h; STRINDEX1) String Index 1.....	19
5.1.9	(Address 7h; STRINDEX2) String Index 2.....	19
5.1.10	(Address 10h; TTL) Time to Live.....	19
5.1.11	(Address 11h; STATUS) Status.....	19
5.1.12	(Address 1Fh; CHIPREV) NET1080 Silicon Revision	20
6	USB Vendor-Specific Device Requests (Endpoint 0).....	21
6.1	Register Write	21
6.2	Register Read	21
6.3	EEPROM Write	22
6.4	EEPROM Read.....	22
6.5	Manufacturer Test Mode.....	23

6.5.1	Device Clear Feature (Timing Test Mode).....	23
6.5.2	Device Set Feature (Timing Test Mode).....	23
7	USB Standard Device Requests (Endpoint 0).....	24
7.1	Control Read Transfers	24
7.1.1	Get Device Status	24
7.1.2	Get Interface Status.....	24
7.1.3	Get Endpoint 00h, 01h, 81h, 02h, 82h, 83h, 84h Status.....	24
7.1.4	Get Device Descriptor (18 Bytes).....	24
7.1.5	Get Configuration Descriptor (53 bytes).....	25
7.1.6	Get String Descriptor 0.....	27
7.1.7	Get String Descriptor 1.....	27
7.1.8	Get String Descriptor 2.....	27
7.1.9	Get String Descriptor 3.....	27
7.1.10	Get Configuration	28
7.1.11	Get Interface	28
7.1.12	Set Address.....	28
7.2	Control Write Transfers	28
7.2.1	Set Configuration.....	28
7.2.2	Set Interface.....	28
7.2.3	Device Clear Feature	28
7.2.4	Device Set Feature	28
7.2.5	Endpoint Clear Feature	29
7.2.6	Endpoint Set Feature.....	29
8	Bulk and Interrupt Endpoints	30
8.1	Endpoint 01h 'OUT' Transactions (Data Out Pipe).....	30
8.2	Endpoint 81h 'IN' Transactions (Data In Pipe).....	30
8.3	Endpoint 02h 'OUT' Transactions (Mailbox Out Pipe).....	30
8.4	Endpoint 82h 'IN' Transactions (Mailbox In Pipe).....	30
8.5	Endpoint 83h 'IN' Transactions (Status Input Pipe).....	30
9	Electrical Specifications.....	31
9.1	Absolute Maximum Ratings	31
9.2	Recommended Operating Conditions	31
9.3	DC Specifications.....	31
9.3.1	DC Specifications	31
9.3.2	USB Port DC Specifications.....	32
9.3.3	USB Port AC Specifications.....	32
9.3.4	USB Port AC/DC Specification Notes	33
9.3.5	USB Port AC Waveforms	33
10	Mechanical Drawing	35

1 Highlights

1.1 Introduction

The NetChip USB TurboCONNECT provides a single chip solution for connecting two USB Hosts. No external microprocessor is needed. The Netchip USB TurboCONNECT offers a quick and easy solution for host-to-host communications at speed equivalent to an Ethernet 10BT connection. Users can now network two or more Universal Serial Bus (USB) computers through standard Plug & Play technology without adding network cards.

1.2 Features

1.2.1 General Features

- Single-Chip USB Host-to-Host Connection
- Low Power CMOS in 48-pin PQFP Package
- Supports 100 mA low power USB Specification
- Supports 500 μ A suspend current USB Specification
- 6 MHz crystal oscillator with internal PLL.
- MicroWire™ EEPROM interface
- Data transfer rate of 5 Mbps or greater
- 256 Byte FIFO per direction for data transfer
- Automatic retry of failed packets
- Low latency packet forwarding (minimum of 2.6 microseconds)

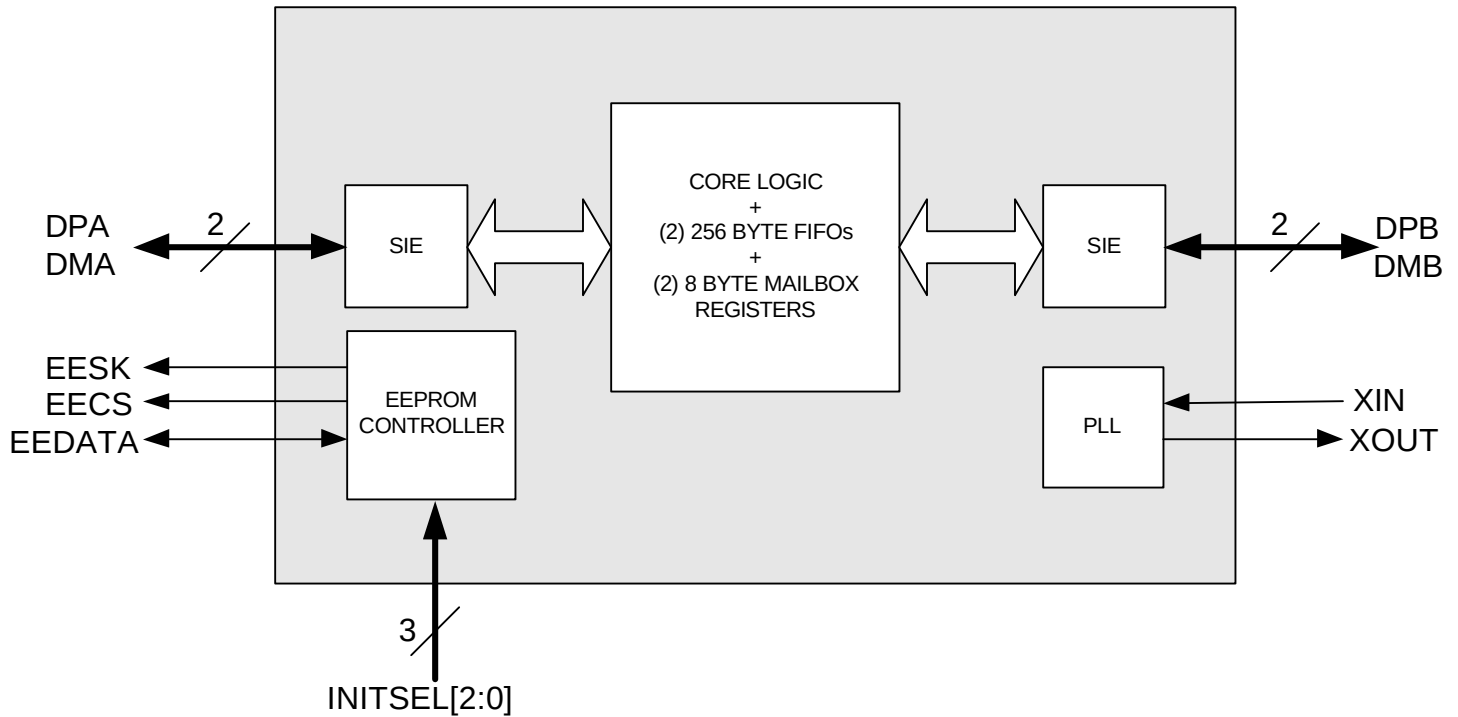
1.2.2 USB Features

- USB Specification Version 1.1 Compliant
- Supports Full-Speed USB Bandwidth of 12 Mbps
- Status Reporting through USB Interrupt Endpoints

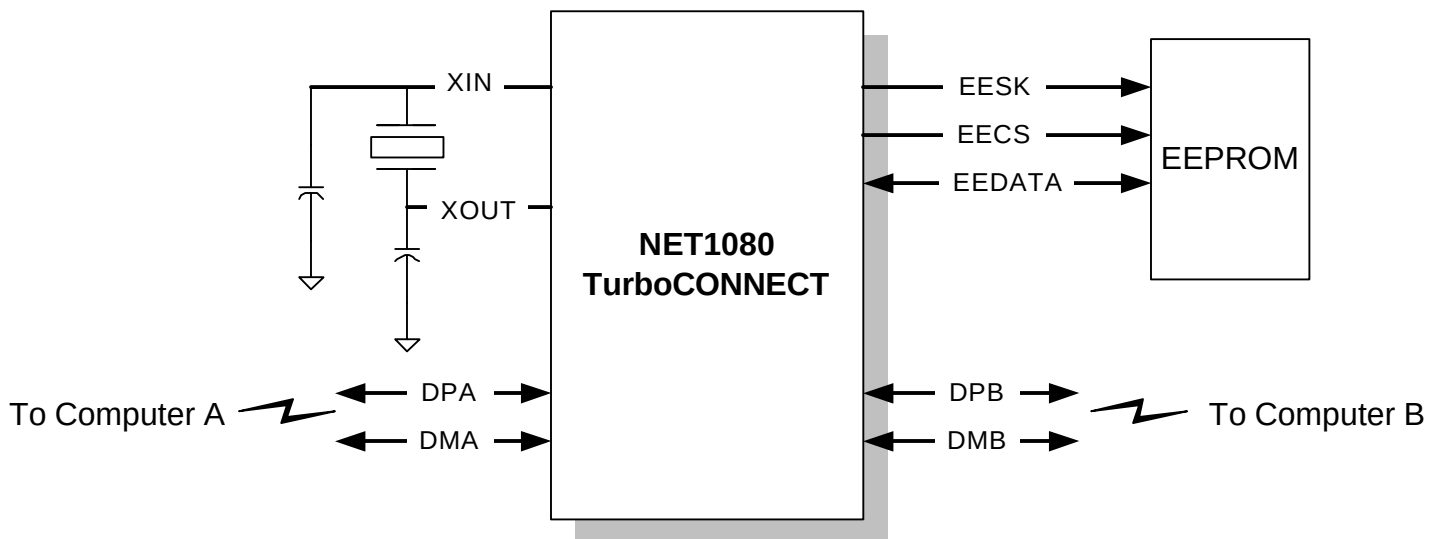
1.3 Overview

The NET1080 USB TurboCONNECT allows two Hosts to communicate with each other simultaneously. It consists of two SIEs to interface to two Hosts, bi-directional FIFOs, and an EEPROM interface. The EEPROM interface is provided to initialize the NET1080 with parameters necessary for enumeration of the device. Parameters loaded from the serial device include the Manufacturer ID, Product ID, Revision Number, Maximum Power fields, and USB control bits, as well as optional string descriptors. For maximum data throughput, the NET1080 provides a 256 byte FIFO per direction for data transfer. Other endpoints that are provided are a pair of CONTROL endpoints that support the USB standard requests and vendor specific requests, a pair of status endpoints that report the number of bytes in the next available packet. The NET1080 has the capability to enter low-power suspend as required by the USB specification. When enabled, it can perform a device remote wakeup on either port when data becomes available.

1.4 NET1080 Block Diagram



1.5 NET1080 System Block Diagram

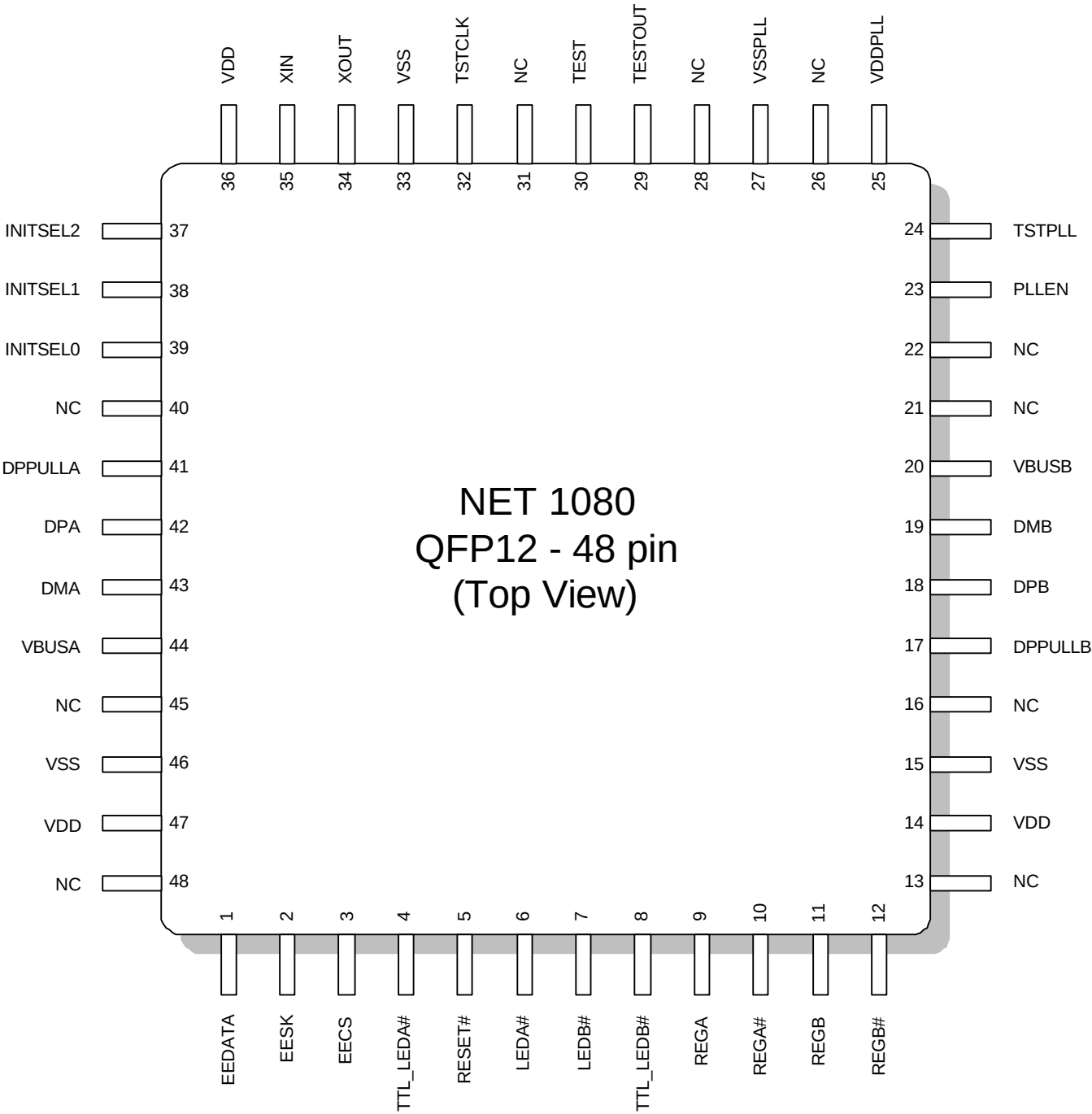


Changes from Rev 1 to Rev 2

The following changes were made to the NET1080 from Revision 1 to Revision 2:

1. Power Control: Revision 1 derives power only from the A port if both ports are powered. Revision 2 derives power from both ports unless one port is suspended, in which case it derives power from the other port.
2. OUT NAKs when other port is disconnected (erratum 630-0102-0103): Revision 1 NAKs OUT packets if the other port is disconnected causing packets to back up in the host driver. Revision 2 accepts (ACKs) packets if the other side is disconnected, but silently discards the packets.
3. Fixed Get Status failures on endpoints 1, 2, 82, and 83 (erratum 630-0102-0101).
4. Fixed Non-endpoint 0 traffic interfering with control protocol (erratum 630-0102-0102).
5. Changed LED algorithm to flash on for 255 milliseconds when CONFIGURED, and flash on for 32 milliseconds when data is transferred to the other port.
6. Added TTL_LED for each port. TTL_LED flashes on for 32 milliseconds whenever a packet on the port is discarded due to Time-To-Live expiration.

2 Pin Connection Diagram



3 Pin Description

3.1 Table 3-1: Pin Types

Pin Type	Description
I	Input
O	Output
I/O	Bi-Directional (with Tri-State)
S	Schmitt Trigger
TS	Tri-State
TP	Totem Pole
OD	Open Drain
PD1	50K Ω Internal Pull-Down Resistor
PU1	50K Ω Internal Pull-Up Resistor
PD2	100K Ω Internal Pull-Down Resistor
PU2	100K Ω Internal Pull-Up Resistor
#	Active Low

3.2 Table 3-2: Pin Descriptions

Signal Name	Pin	Type	Description
EECS (MicroWire)	3	O, TP, 2mA	EEPROM Chip Select
EESK (MicroWire)	2	O, TP, 2mA	EEPROM Clock
EEDATA (MicroWire)	1	I/O, TS, S, 2mA, PU2	EEPROM Read/Write
INITSEL0	39	I	Initialization Select Bit 0. Select source of register initialization data.
INITSEL1	38	I	Initialization Select Bit 1. Select source of register initialization data.
INITSEL2	37	I	Initialization Select Bit 2. Select source of register initialization data. Bit 2 1 0 Source - - ----- 0 0 0 MicroWire EEPROM (93CS46, 93CS06) 0 0 1 MicroWire EEPROM (93CS56, 93CS66) 0 1 0 Reserved 0 1 1 Reserved 1 0 0 Reserved 1 0 1 Reserved 1 1 0 Reserved 1 1 1 Reserved
RESET#	5	I, S, PU1	Reset. Connect to power-on reset.
DMA	43	I/O	USB Data Port A. DPA and DMA are differential data signals of the USB data port A. A 27 ohm series resistor between DMA and the USB connector is required to meet the USB impedance requirements.

DPA	42	I/O	USB Data Port A. DPA and DMA are differential data signals of the USB data port A. An external 1.5 K Ω resistor must be connected from DPA to DPPULLA. This pull-up resistor indicates to the Host or upstream hub that a full-speed device is connected to the USB. A 27 ohm series resistor between DPA and the USB connector is required to meet the USB impedance requirements.
DMB	19	I/O	USB Data Port B. DPB and DMB are differential data signals of the USB data port B. A 27 ohm series resistor between DMB and the USB connector is required to meet the USB impedance requirements.
DPB	18	I/O	USB Data Port B. DPB and DMB are differential data signals of the USB data port B. An external 1.5 K Ω resistor must be connected from DPB to DPPULLB. This pull-up resistor indicates to the Host or upstream hub that a full-speed device is connected to the USB. A 27 ohm series resistor between DPB and the USB connector is required to meet the USB impedance requirements.
TESTOUT	29	O, TS	Test Output. For manufacturing test purposes. Leave unconnected.
TEST	30	I, PD1	Test Mode. Set to 0 or leave unconnected.
XIN	35	I	6 MHz Oscillator Input. Connect to 6 MHz crystal.
XOUT	34	O	6 MHz Oscillator Output. Connect to 6 MHz. The oscillator stops when the USB Host suspends the device.
TSTCLK	32	I, PD2	PLL Test Clock. Set to 0 or leave unconnected if the internal PLL is enabled. If the internal PLL is disabled, input an external 48 MHz signal.
TSTPLL	24	I, PD1	PLL Test Enable. For manufacturing test purposes. Set to 0 or leave unconnected.
PLEN	23	I	PLL Enable. Set to 1 to enable the internal PLL. Set to 0 to disable the internal PLL. In this case, an external 48 MHz signal must be applied to the TSTCLK input.
VBUSA	44	I, PD2	USB Port A power (+5V).
VBUSB	20	I, PD2	USB Port A power (+5V).
DPPULLA	41	O, TS, 12 mA	DPA Pull-up. This active high output provides the VCC for the DPA pull-up resistor.
DPPULLB	17	O, TS, 12 mA	DPB Pull-up. This active high output provides the VCC for the DPB pull-up resistor.
LEDA#	6	O, TS, 12 mA, PU2	LED A Control. Active low output to control the A host side LED.
LEDB#	7	O, TS, 12 mA, PU2	LED B Control. Active low output to control the B host side LED.
TTL_LEDA#	4	O, TS, 12 mA, PU2	Time-To-Live LED A Control. Active low output to control the A host side TTL LED.
TTL_LEDB#	8	O, TS, 12 mA, PU2	Time-To-Live LED B Control. Active low output to control the B host side TTL LED.
REGA	9	O, OD, 6 mA	Regulator A Control. Active high open drain output to control the A host side regulator.
REGA#	10	O, TP, 6 mA	Regulator A Control. Active low output to control the A host side regulator.
REGB	11	O, OD, 6 mA	Regulator B Control. Active high open drain output to control the B host side regulator.
REGB#	12	O, TP, 6 mA	Regulator B Control. Active low output to control the B host side regulator.
VDDPLL	25	Pwr	PLL Supply Voltage. VDDPLL = 3.3V
VSSPLL	27	Gnd	PLL Supply Voltage. VSSPLL = GND

VDD	14,36,47	Pwr	Supply Voltage. VDD = 3.3 volts.
VSS	15,33,47	Gnd	Ground.
NC	13, 16, 21, 22, 26, 28, 31, 40, 45, 48	--	No Connect. Do not connect these pins.

4 Functional Description

4.1 Start-Up

Before the NET1080 is ready to transfer data between two USB Hosts, a sequence of operations must occur involving the NET1080, the USB driver, and the TurboCONNECT drivers. First, the NET1080 reads required USB parameters from the external serial memory device. Next, the USB driver must enumerate the NET1080 on both sides. Finally, the TurboCONNECT driver configures the NET1080 with the correct parameters.

4.1.1 Reading from Initialization Device

When the NET1080 first comes out of a reset applied at its RESET# input, it reads various parameters required by the USB Host from an external serial memory. The NET1080 generates all signals necessary to read the data from the serial memory. Until this data is read, the NET1080 will respond to Host requests for a configuration descriptor or device descriptor with a NAK. If no valid serial memory device is installed, then the default values are used.

4.1.2 Enumeration and Configuration by the Host PC

USB defines a set of descriptor requests that are part of the enumeration process. The NET1080 responds to these requests automatically. When enumeration is complete, control of the NET1080 is passed to the TurboCONNECT driver.

As part of the enumeration process, the USB Host issues a Get Device Descriptor request. The NET1080 responds with the descriptor shown in section 7.1.4. Some of the values, such as the Manufacturer ID and Product ID, may come from the optional serial EEPROM. This Manufacturer ID and Product ID allow the Host to associate the manufacturer-supplied TurboCONNECT driver with the NET1080. The TurboCONNECT driver then finishes initializing the NET1080.

4.2 Initialization Interface

The NET1080 initialization must be completed before the NET1080 will respond to the Device Descriptor requests from the Host. The table below lists the different values of the INITSEL bits that are used to select various EEPROMs.

INITSEL	Source
0	MicroWire™ EEPROM (93CS06, 93CS46)
1	MicroWire™ EEPROM(93CS56, 93CS66)
2-7	Reserved

4.2.1 MicroWire EEPROMs

The NET1080 supports initialization and read/write operations to the following MicroWire EEPROMs:

- 93C06L – 16 16-bit words (256 bits), CMOS, Low Voltage (2.7V to 4.5V)
- 93C46L – 64 16-bit words (1 Kbits), CMOS, Low Voltage (2.7V to 4.5V)
- 93C56L – 128 16-bit words (2 Kbits), CMOS, Low Voltage (2.7V to 4.5V)
- 93C66L – 256 16-bit words (4 Kbits), CMOS, Low Voltage (2.7V to 4.5V)

Each of these devices has a 4-wire MicroWire™ serial interface. The NET1080 supports a 3-wire implementation of this interface in which the EEPROM DI pin is connected to the NET1080 EEDATA pin, and the EEPROM DO pin is connected to the EEPROM EEDI pin through a 33 ohm resistor.

After reset is inactive, the INITSEL bits are sampled to determine the source of the initialization information. If INITSEL[2:0] is 0 or 1, the NET1080 will determine if a non-blank MicroWire EEPROM is connected. If no EEPROM is detected, or the device is blank (first word is FFFF), then the default register values will be reported to the Host PC.

Otherwise, the following required fields from Table 3-1 below are loaded into the NET1080 from the EEPROM. These required fields are reported in the Device and Configuration descriptors during USB device enumeration. Information must be stored in the EEPROM as 16-bit words. For string descriptors, bits 7:0 of an EEPROM word are returned to the Host first, followed by bits 15:8.

Table 4-1: Required MicroWire Serial EEPROM Fields

Word Index	Contents	Default Value
0	Manufacturer ID. This field is reported to the Host PC in the USB device descriptor's "Manufacturer ID" field during enumeration.	0525h
1	Product ID. This field is reported to the Host PC in the USB device descriptor's "Product ID" field during enumeration.	1080h
2	Release Number. This field is reported to the Host PC in the USB device descriptor's "Release Number" field during enumeration.	0200h
3	Max Power. This field is reported to the Host PC in the USB configuration descriptor's "Max Power" field during enumeration. It is reported in units of 2mA. This field should reflect the maximum current drawn by the TurboCONNECT from the USB power pins.	0032h
4	USB Features. This field contains several control bits that are reported to the Host PC in the USB device descriptor and configuration descriptor. See USB Control Register for details.	0013h
5	Interrupt Polling Rate. This field determines the rate at which the interrupt endpoints are polled by the Host.	0001h
6 (bits 15:8)	Language ID String Start Index.	0000h
6 (bits 7:0)	Manufacturer String Start Index.	0000h
7 (bits 15:8)	Product String Start Index.	0000h
7 (bits 7:0)	Serial Number String Start Index.	0000h

In addition, if any "String Enable" fields are set in the **USB Control Register**, the fields in Table 4-2 must be defined.

Table 4-2: Additional MicroWire Serial EEPROM Fields for String Descriptors

Word Index	Contents
8-9h	Reserved. Write these words as a zero.
Ah-FFh	Available. This range of the serial EPROM is available for defining string descriptors.

If no string descriptors are used, the EEPROM image may be as short as 8 words (to provide the required fields to the Host PC). With string descriptors, the EEPROM can end at the end of the last string descriptor's definition. Since the string start index fields are only 8-bits, the beginning address of the last string in the EEPROM can't have a starting word address greater than 255.

The MicroWire EEPROMs can also be accessed from either Host using Vendor Specific commands to endpoint 0. Only one 16-bit word can be read or written for each Vendor Specific command. See sections 6.3 and 6.4.

For an EEPROM write, the Host issues a Vendor Specific Host to Device request to endpoint 0. The EEPROM word location is passed in the wIndex field of the Setup packet, and the 16-bit write data is passed in the wValue field of the Setup packet. Further attempts to access the EEPROM result in a NAK handshake until the write cycle has completed.

For an EEPROM read, the Host issues a Vendor Specific Device to Host request to endpoint 0. The EEPROM word location is passed in the wIndex field of the Setup packet. The associated IN tokens are NAKed until the data has been read from the EEPROM. A 2-byte packet containing the EEPROM read data is then returned to the host.

4.3 USB Interface

The interface from the NET1080 to USB allows two Hosts to pass data between them. Each USB port on the NET1080 has the following endpoints.

4.3.1 Default Control Endpoint (Endpoint 00h)

Endpoint 0 is a bi-directional USB control endpoint which processes 8-byte packets to/from the Host PC. Each packet is decoded and USB control read and write 'requests' are handled automatically. Endpoint 0 supports the required USB protocol read and write requests (e.g. descriptor reads), as well as vendor-specific extensions. The vendor-specific extensions provide the capability to read and write internal registers.

4.3.2 Data OUT Endpoint (Endpoint 01h)

Endpoint 01h is a USB bulk OUT endpoint that accepts packets up to 64 bytes in length. Data received from a Host on this endpoint is transferred to a 256 byte FIFO that is connected to the Data IN Endpoint on the opposite USB port. When data is transferred to the other port, the LED for this port is turned on for a minimum of 32 milliseconds. Up to three complete packets can be in the FIFO at any time

4.3.3 Data IN Endpoint (Endpoint 81h)

Endpoint 81h is a USB bulk IN endpoint that transmits packets up to 64 bytes in length. In response to an IN token from the Host, data is read from a 256 byte FIFO that is connected to the Data OUT Endpoint on the opposite USB port, and is returned to the Host.

4.3.4 Mailbox OUT Endpoint (Endpoint 02h)

Endpoint 02h is a USB bulk OUT endpoint that accepts packets up to 16 bytes in length. Data received from a Host on this endpoint is transferred to an 8 byte IN Mailbox Endpoint on the opposite USB port. If the IN Mailbox of the other port is already busy, this endpoint returns a NAK handshake.

4.3.5 Mailbox IN Endpoint (Endpoint 82h)

Endpoint 82h is a USB interrupt IN endpoint that transmits the 8 bytes of IN Mailbox data. If no data is available, this endpoint returns a NAK handshake.

4.3.6 Status Endpoint (Endpoint 83h)

Endpoint 83h is a USB interrupt IN endpoint that reports the current number of bytes available in the next packet on the Data IN endpoint. If there are no packets available, the endpoint returns a NAK handshake. If there is a zero-length packet available, this endpoint returns 0.

4.4 Suspend Mode

When there is a three-millisecond period of inactivity on the USB, the USB specification requires a device to enter into a low-power suspended state. While in this state the device may not draw more than 500 μ A of current from the USB connector's power pins. To facilitate this, the NET1080 automatically enters the suspend state after detecting the three millisecond period of inactivity. The NET1080 goes into suspend after detecting the suspend condition on both USB ports. If only one port is suspended, the NET1080 draws power from the other port.

In the suspended state, the NET1080's oscillator shuts down, and most output pins are tri-stated to conserve power (see section 3, **Pin Description**).

If the host has enabled Device Remote Wakeup, the NET1080 will wake up the suspended host when there is data available on its Data IN endpoint. The NET1080 leaves suspend mode when a Host initiated wake-up is detected.

4.5 Root Port Reset

If the SIE in the NET1080 detects a single-ended zero on the root port for greater than 2.5 microseconds, it is interpreted as a root port reset. The following resources are reset for that port:

- SIE
- USB state machines
- USB Address Register is set to 0

4.6 NET1080 Power Configuration

The USB specification defines both bus-powered and self-powered devices. A *bus-powered* device is a peripheral that derives all of its power from the upstream USB connector, while a *self-powered* device has an external power supply. The NET1080 is a low power bus-powered peripheral. It draws its power from either USB Port A or Port B, or both ports.

The USB specification specifies the following requirements for maximum current draw:

- A peripheral not configured by the Host may draw only 100 mA from the USB power pins.
- A configured low-power device may not draw more than 100 mA from the USB connector's power pins.
- In suspend mode, the peripheral may not draw more than 500 μ A from the USB connector's power pins

The NET1080 meets all of these specifications. It is designed to be used as a bus-powered device.

5 Register Descriptions

The NET1080 contains a 16 x 16-bit local register space that can be accessed by either Host over USB using Vendor Specific commands to Endpoint 0. When the NET1080 receives a RESET signal from the RESET# pin or from a USB RESET command, the registers are set to their default values.

Writes to reserved registers are ignored, and reads from reserved registers return a value of 0. For compatibility with future revisions, unused bits within a register should always be written with a zero.

USB Register Description

These registers determine some of the values reported in USB device and configuration descriptors returned to the Hosts. The values in the registers may be changed by the initialization interface before a descriptor is returned to the Host. See Section 4.2.

5.1.1 USB Register Summary

Address Range	Register	Register Description
0 h	MFGID	USB Manufacturer ID Field
1 h	PRODID	Product ID Field
2 h	RELNUM	Device release number
3 h	MAXPWR	Maximum USB power used
4 h	USBCTL	USB control
5 h	INTPOLL	Interrupt Polling rate
6 h	STRINDEX1	USB String Index 1 (Language ID, Manufacturer)
7 h	STRINDEX2	USB String Index 2 (Product, Serial Number)
8 h-F h	Reserved	
10 h	TTL	Time to Live
11 h	STATUS	Status Register
12 h-1E h	Reserved	
1F h	CHIPREV	Current Silicon Revision

5.1.2 (Address 0h; MFGID) Manufacturer ID

Bits	Description	Read	Write	Default Value
15:0	USB Manufacturer ID. This register determines the Manufacturer ID during a 'Get Device Descriptor' request (assigned by USB-IF).	Yes	Yes	0525h

5.1.3 (Address 1h; PRODID) Product ID

Bits	Description	Read	Write	Default Value
15:0	Product ID. This register determines the Product ID during a 'Get Device Descriptor' request.	Yes	Yes	1080h

5.1.4 (Address 2h; RELNUM) Release Number

Bits	Description	Read	Write	Default Value
15:0	Release Number. This register determines the device release number during a 'Get Device Descriptor' request.	Yes	Yes	REL NUM

Note: RELNUM is the silicon revision, encoded as a 4-digit BCD value. The value of RELNUM for the first release of the chip is 0100h. The least-significant two digits are incremented for mask changes, and the most-significant two digits increment for major revisions. This value can be changed by the EEPROM to implement an application release number.

5.1.5 (Address 3h; MAXPWR) Maximum Power

Bits	Description	Read	Write	Default Value
15:8	Reserved.	Yes	No	0
7:0	Maximum Power. This register determines the maximum USB power during a 'Get Configuration Descriptor' request. Power is reported in units of 2 mA.	Yes	Yes	32h

5.1.6 (Address 4h; USBCTL) USB Control Register

Bits	Description	Read	Write	Default Value
15:13	Reserved	Yes	No	0
12	Language ID String Enable. If set, a language string ID is programmed into the EEPROM. The location of the language string is specified by the “ Language String Start Index ” byte in the serial EEPROM. If clear, the response to a Language ID String read is determined by the String Default Enable bit.	Yes	Yes	0
11	Manufacturer String Enable. If set, a manufacturer’s string is programmed into the EEPROM. The location of the manufacturer’s string is specified by the “ Manufacturer String Start Index ” byte in the serial EEPROM. If clear, the response to a Manufacturer String read is determined by the String Default Enable bit.	Yes	Yes	0
10	Product String Enable. If set, a product string is programmed into the EEPROM. The location of the product string in the EEPROM is specified by the “ Product String Start Index ” byte in the serial EEPROM. If clear, the response to a Product String language ID string read is determined by the String Default Enable bit.	Yes	Yes	0
9	Serial Number String Enable. If set, a serial number string is programmed into the EEPROM. The location of the serial number string is specified by the “ Serial Number String Start Index ” byte in the serial EEPROM. If clear, the response to a language ID string read is determined by the String Default Enable bit.	Yes	Yes	0
8	String Default Enable. If set, default strings are returned to the Host if the corresponding String Enable (bits 12:9) is 0. If clear, a string is returned only if the corresponding String Enable (bits 12:9) is 1.	No	Yes	1
7:4	Reserved	Yes	No	0
3	Flush Other. This bit flushes the Data IN FIFO for other port. This bit is self-clearing. When this bit is programmed from the EEPROM, it refers to the B port.	0	Yes/ Flush	0
2	Flush This. This bit flushes the Data IN FIFO for this port. This bit is self-clearing. When this bit is programmed from the EEPROM, it refers to the A port.	0	Yes/ Flush	0
1	Disconnect Other Port. Writing a 1 to this bit causes the DP Pull-up resistor not to be driven high, thus simulating a disconnect on the other USB Port. The port will stay disconnected until this bit is cleared. When programmed from the EEPROM, this bit refers to the B port.	Yes	Yes	0
0	Disconnect This Port. Writing a 1 to this bit causes the DP Pull-up resistor not to be driven high, thus simulating a disconnect on this USB Port. This bit clear itself after a 500-millisecond delay, and the port will automatically re-connect. When programmed from the EEPROM, this bit refers to the A port, and it will not clear itself after 500 milliseconds.	0	Yes	0

5.1.7 (Address 5h; INTPOLL) Interrupt Polling Interval Register

Bits	Description	Read	Write	Default Value
15:8	Interrupt Polling Interval Register for Endpoint 83h. This register specifies the interrupt polling interval in milliseconds for the Status Endpoint 83h. It is returned as the last byte of the Status Endpoint descriptor when the Get Configuration Descriptor request is issued.	Yes	Yes	1

7:0	Interrupt Polling Interval Register for Endpoint 82h. This register specifies the interrupt polling interval in milliseconds for the Mailbox Endpoint 82h. It is returned as the last byte of the Mailbox Endpoint descriptor when the Get Configuration Descriptor request is issued.	Yes	Yes	64h
-----	---	-----	-----	-----

5.1.8 (Address 6h; STRINDEX1) String Index 1

Bits	Description	Read	Write	Default Value
15:8	Language ID String Index. This field determines the EEPROM starting word address of the Language ID string.	Yes	Yes	0
7:0	Manufacturer String Index. This field determines the EEPROM starting word address of the Manufacturer string.	Yes	Yes	0

5.1.9 (Address 7h; STRINDEX2) String Index 2

Bits	Description	Read	Write	Default Value
15:8	Product String Index. This field determines the EEPROM starting word address of the Product string.	Yes	Yes	0
7:0	Serial Number String Index. This field determines the EEPROM starting word address of the Serial Number string.	Yes	Yes	0

5.1.10 (Address 10h; TTL) Time to Live

Bits	Description	Read	Write	Default Value
15:8	Time to Live Other Port. This field determines how many milliseconds a packet will stay in the Data FIFO on the other port before it is discarded. If set to 0, packets will not be discarded.	Yes	Yes	0
7:0	Time to Live This Port. This field determines how many milliseconds a packet will stay in the Data FIFO on this port before it is discarded. If set to 0, packets will not be discarded.	Yes	Yes	0

5.1.11 (Address 11h; STATUS) Status

Bits	Description	Read	Write	Default Value
15	Port A. This bit is set if read from the A port.	Yes	No	0
14	Connected Other Port. This bit indicates that the other side has connected	Yes	No	0
13	Suspended Other Port. This bit indicates that the other side has suspended.	Yes	No	0
12	Mailbox Data Available Other Port. This bit is set if there is data available in the USB Mailbox for the other port.	Yes	No	0
11:10	Reserved.	Yes	No	0
9:8	Packets Available Other Port. This field indicates the number of packets available in the Data FIFO of the other port.	Yes	No	0
7	Reserved.	Yes	No	1
6	Connected This Port. This bit indicates that this side has connected	Yes	No	0
5	Suspended This Port. This bit indicates that this side has suspended.	Yes	No	0
4	Mailbox Data Available This Port. This bit is set if there is data available in the USB Mailbox for this port.	Yes	No	0

3:2	Reserved.	Yes	No	0
1:0	Packets Available This Port. This field indicates the number of packets available in the Data FIFO of this port.	Yes	No	0

5.1.12 (Address 1Fh; CHIPREV) NET1080 Silicon Revision

Bits	Description	Read	Write	Default Value
15:0	NET1080 Silicon Revision. This is the Silicon Revision Number for the NET1080	Yes	No	RELNUM

Note: RELNUM is the silicon revision, encoded as a 4-digit BCD value. The value of RELNUM for the first release of the chip is 0100h. The least-significant two digits are incremented for mask changes, and the most-significant two digits increment for major revisions. This value cannot be changed.

6 USB Vendor-Specific Device Requests (Endpoint 0)

Vendor specific requests to endpoint 0 are used to access the internal registers of the NET1080 and to perform read and write cycles to the EEPROM. For register and EEPROM accesses, the wIndex determine the address. A bRequest of 10h is used for register accesses, and a bRequest of 11h is used for EEPROM accesses.

6.1 Register Write

A vendor request to endpoint 0 is used to perform register writes. One 16-bit write can be performed for each transfer. The data is not written to the target register until it has been determined that the USB transfer was successful. The target register address is passed in bits 7:0 of the wIndex field of the Setup packet. The write data is passed in the wValue field of the Setup packet. If the wLength field is not 0, the endpoint will return a STALL handshake. The register is not actually written until the Status Phase of the vendor request has completed.

Setup Byte	Contents	Description
0	BmRequestType	Bit 7 = direction (0=Out) Bits 6:5 = type (2=vendor) Bits 4:0 = Recipient (0 = device)
1	Brequest	10h = Register access
2	wValue (LSB)	Write data (LSB)
3	wValue (MSB)	Write data (MSB)
4	wIndex (LSB)	Register Address
5	wIndex (MSB)	Reserved – set to 0
6	wLength (LSB)	Number of bytes in Data Phase – must be 0
7	wLength (MSB)	Reserved – set to 0

6.2 Register Read

A vendor request to endpoint 0 is used to perform register reads. One 16-bit read can be performed for each transfer. The target register address is passed in bits 7:0 of the wIndex field of the Setup packet. If the wLength field is not 2, the endpoint will return a stall acknowledge.

Setup Byte	Contents	Description
0	BmRequestType	Bit 7 = direction (1=IN) Bits 6:5 = type (2=vendor) Bits 4:0 = Recipient (0 = device)
1	Brequest	10h = Register access
2	wValue (LSB)	Reserved – set to 0
3	wValue (MSB)	Reserved – set to 0
4	wIndex (LSB)	Register Address
5	wIndex (MSB)	Reserved – set to 0
6	wLength (LSB)	Number of bytes to read, must be 2
7	wLength (MSB)	Reserved – set to 0

Following is a description of the contents of the IN packet associated with this control read transfer.

Byte Index	Contents
0	Register Read Data (LSB)
1	Register Read Data (MSB)

6.3 EEPROM Write

A vendor request to endpoint 0 is used to perform EEPROM writes. One 16-bit write can be performed for each transfer. The EEPROM word address is passed in bits 9:0 of the wIndex field of the Setup packet. The write data is passed in the wValue field of the Setup packet. If the wLength field is not 0, the endpoint returns a STALL handshake. Further accesses to endpoint 0 are NAK'ed in the Status Phase until the write operation has completed.

Setup Byte	Contents	Description
0	BmRequestType	Bit 7 = direction (0=Out) Bits 6:5 = type (2=vendor) Bits 4:0 = Recipient (0 = device)
1	Brequest	11h = EEPROM access
2	wValue (LSB)	Write data (LSB)
3	wValue (MSB)	Write data (MSB)
4	wIndex (LSB)	EEPROM word Address
5	wIndex (MSB)	EEPROM word Address
6	wLength (LSB)	Number of bytes in Data Phase – must be 0
7	wLength (MSB)	Reserved – set to 0

6.4 EEPROM Read

A vendor request to endpoint 0 is used to perform EEPROM reads. One 16-bit read can be performed for each transfer. The EEPROM word address is passed in bits 9:0 of the wIndex field of the Setup packet. If the wLength field is not 2, the endpoint will return a stall acknowledge.

Setup Byte	Contents	Description
0	BmRequestType	Bit 7 = direction (1=IN) Bits 6:5 = type (2=vendor) Bits 4:0 = Recipient (0 = device)
1	Brequest	11h = EEPROM access
2	wValue (LSB)	Reserved – set to 0
3	wValue (MSB)	Reserved – set to 0
4	wIndex (LSB)	EEPROM word Address
5	wIndex (MSB)	EEPROM word Address
6	wLength (LSB)	Number of bytes to read, must be 2
7	wLength (MSB)	Reserved – set to 0

Following is a description of the contents of the IN packet associated with this control read transfer. IN tokens are NAK'ed until two bytes have been read from the EEPROM.

Byte Index	Contents
0	EEPROM Read Data (LSB)
1	EEPROM Read Data (MSB)

6.5 Manufacturer Test Mode

A vendor specific request can be used to set or clear the Timing Test Mode bit. Timing Test Mode is used during silicon verification and should not be enabled during normal operation.

6.5.1 Device Clear Feature (Timing Test Mode)

Offset	Number of Bytes	Description	Default Value
--	0	Clear the selected device feature wValue = 80h --> Timing test mode (clears test bit) wIndex ignored wLength = 0	--

6.5.2 Device Set Feature (Timing Test Mode)

Offset	Number of Bytes	Description	Default Value
--	0	Set the selected device feature wValue = 80h --> Timing test mode (sets test bit) wIndex ignored wLength = 0	--

7 USB Standard Device Requests (Endpoint 0)

7.1 Control Read Transfers

7.1.1 Get Device Status

Offset	Number of Bytes	Description	Default Value
0	2	bits 15:2 = Reserved bit 1 = Device Remote Wakeup enabled bit 0 = 0	0000h

7.1.2 Get Interface Status

Offset	Number of Bytes	Description	Default Value
0	2	bits 15:0 = Reserved	0000h

7.1.3 Get Endpoint 00h, 01h, 81h, 02h, 82h, 83h, 84h Status

Offset	Number of Bytes	Description	Default Value
0	2	bits 15:1 = Reserved bit 0 = Endpoint is stalled	0000h

7.1.4 Get Device Descriptor (18 Bytes)

Offset	Number of Bytes	Description	Default Value
0	1	Length	12h
1	1	Type (device)	01h
2	2	USB Specification Release Number	0101h
4	1	Class Code	FFh (Vendor Specific)
5	1	Sub Class Code	00h
6	1	Protocol	00h
7	1	Maximum Endpoint 0 Packet Size	08h
8	2	USB Manufacturer ID	Value of MFGID register
10	2	Product ID	Value of PRODID register
12	2	Device Release Number	Value of RELNUM register
14	1	Index of string descriptor describing manufacturer	01h (if USBCTL Mfg. String Enable = 1) 00h (if USBCTL Mfg. String Enable = 0)
15	1	Index of string descriptor describing product	02h (if USBCTL Prod. String Enable = 1) 00h (if USBCTL Prod. String Enable = 0)
16	1	Index of string descriptor describing serial number	03h (if USBCTL Serial Number String Enable = 1) 00h (if USBCTL Serial Number String Enable = 0)

17	1	Number of configurations	01h
----	---	--------------------------	-----

7.1.5 Get Configuration Descriptor (53 bytes)

Note that all interface and endpoint descriptors are returned when this request is issued

Offset	Number of Bytes	Description	Default Value
Configuration Descriptor			
0	1	Length	09h
1	1	Type (configuration)	02h
2	2	Total length returned for this configuration	0035h
4	1	Number of Interfaces	01h
5	1	Number of this configuration	01h
6	1	Index of string descriptor describing this configuration	00h
7	1	Attributes bit 7 = Reserved – must be 1 bit 6 = Self-powered bit 5 = Remote Wakeup capability bits 4:0 = Reserved – set to 0	A0h
8	1	Maximum USB power required (in 2 mA units)	Value in MAXPWR register
Interface 0 Descriptor			
0	1	Size of this descriptor in bytes	09h
1	1	Type (interface)	04h
2	1	Number of this interface	00h
3	1	Alternate Interface	00h
4	1	Number of endpoints used by this interface (excluding endpoint 0)	05h
5	1	Class Code	FFh (Vendor Specific)
6	1	Sub Class Code	00h
7	1	Device Protocol	00h
8	1	Index of string descriptor describing this interface	00h
Endpoint 1 Descriptor			
0	1	Size of this descriptor	07h
1	1	Descriptor Type (endpoint)	05h
2	1	Endpoint Address bit 7 = direction (1 = IN, 0 = OUT) bits 6:4 = reserved bits 3:0 = endpoint number	01h
3	1	Endpoint Attributes Bits 7:2 = reserved Bits 1:0 00 = Control 01 = Isochronous 10 = Bulk 11 = Interrupt	02h
4	2	Maximum packet size of this endpoint	0040h
6	1	Interval for polling endpoint (not used)	00h

Endpoint 81h Descriptor			
0	1	Size of this descriptor	07h
1	1	Descriptor Type (endpoint)	05h
2	1	Endpoint Address bit 7 = direction (1 = IN, 0 = OUT) bits 6:4 = reserved bits 3:0 = endpoint number	81h
3	1	Endpoint Attributes bits 7:2 = reserved bits 1:0 00 = Control 01 = Isochronous 10 = Bulk 11 = Interrupt	02h
4	2	Maximum packet size of this endpoint	0040h
6	1	Interval for polling endpoint (in ms)	00h
Endpoint 2 Descriptor			
0	1	Size of this descriptor	07h
1	1	Descriptor Type (endpoint)	05h
2	1	Endpoint Address bit 7 = direction (1 = IN, 0 = OUT) bits 6:4 = reserved bits 3:0 = endpoint number	02h
3	1	Endpoint Attributes Bits 7:2 = reserved Bits 1:0 00 = Control 01 = Isochronous 10 = Bulk 11 = Interrupt	02h
4	2	Maximum packet size of this endpoint	0008h
6	1	Interval for polling endpoint (not used)	00h
Endpoint 82h Descriptor			
0	1	Size of this descriptor	07h
1	1	Descriptor Type (endpoint)	05h
2	1	Endpoint Address bit 7 = direction (1 = IN, 0 = OUT) bits 6:4 = reserved bits 3:0 = endpoint number	82h
3	1	Endpoint Attributes bits 7:2 = reserved bits 1:0 00 = Control 01 = Isochronous 10 = Bulk 11 = Interrupt	03h
4	2	Maximum packet size of this endpoint	0008h
6	1	Interval for polling endpoint (in ms)	Determined by bits 7:0 of INTPOLL register

Endpoint 83h Descriptor			
0	1	Size of this descriptor	07h
1	1	Descriptor Type (endpoint)	05h
2	1	Endpoint Address bit 7 = direction (1 = IN, 0 = OUT) bits 6:4 = reserved bits 3:0 = endpoint number	83h
3	1	Endpoint Attributes bits 7:2 = reserved bits 1:0 00 = Control 01 = Isochronous 10 = Bulk 11 = Interrupt	03h
4	2	Maximum packet size of this endpoint	0001h
6	1	Interval for polling endpoint (in ms)	Determined by bits 15:8 of INTPLL register

7.1.6 Get String Descriptor 0

Offset	Number of Bytes	Description	Default Value
0	4	Language ID (English = 09, U.S. = 04)	04h, 03h h0409

7.1.7 Get String Descriptor 1

Offset	Number of Bytes	Description	Default Value
0	38	Manufacturer Descriptor	26h, 03h "Netchip Technology"

7.1.8 Get String Descriptor 2

Offset	Number of Bytes	Description	Default Value
0	48	Product Descriptor	30h, 03h "NET1080 USB TurboCONNECT"

7.1.9 Get String Descriptor 3

Offset	Number of Bytes	Description	Default Value
0	10	Serial Number Descriptor	0Ah, 03h "1001"

7.1.10 Get Configuration

Offset	Number of Bytes	Description	Default Value
0	1	Returns current device configuration	00h

7.1.11 Get Interface

Offset	Number of Bytes	Description	Default Value
0	1	Returns current alternate setting for the specified interface	00h

7.1.12 Set Address

Offset	Number of Bytes	Description	Default Value
--	0	Sets USB address of device Value = device address, Index = 0	--

7.2 Control Write Transfers

7.2.1 Set Configuration

Offset	Number of Bytes	Description	Default Value
--	0	Sets the device configuration Value = Configuration value (0 or 1 supported),	--

7.2.2 Set Interface

Offset	Number of Bytes	Description	Default Value
--	0	Selects alternate setting for specified interface Value = Alternate setting, Index = specified interface	--

7.2.3 Device Clear Feature

Offset	Number of Bytes	Description	Default Value
--	0	Clear the selected device feature Value = feature selector FS = 1 --> Device Remote Wakeup (disable)	--

7.2.4 Device Set Feature

Offset	Number of Bytes	Description	Default Value
--	0	Set the selected device feature Value = feature selector FS = 1 --> Device Remote Wakeup (enable)	--

7.2.5 Endpoint Clear Feature

Offset	Number of Bytes	Description	Default Value
--	0	Clear the selected endpoint feature Value = feature selector, Index = endpoint number FS = 0 --> Endpoint stall (clears stall bit)	--

7.2.6 Endpoint Set Feature

Offset	Number of Bytes	Description	Default Value
--	0	Set the selected endpoint feature Value = feature selector, Index = endpoint number FS = 0 --> Endpoint stall (sets stall bit)	--

8 Bulk and Interrupt Endpoints

8.1 Endpoint 01h 'OUT' Transactions (Data Out Pipe)

Offset	Number of Bytes	Description	Default Value
--	64	Host writes data to the FIFO connected to the Data IN Pipe of the opposite USB port.	--

8.2 Endpoint 81h 'IN' Transactions (Data In Pipe)

Offset	Number of Bytes	Description	Default Value
--	64	Host reads data from the FIFO connected to the Data IN Pipe of this USB port.	--

8.3 Endpoint 02h 'OUT' Transactions (Mailbox Out Pipe)

Offset	Number of Bytes	Description	Default Value
--	8	Host writes data to the FIFO connected to the Mailbox IN Pipe of the opposite USB port. If the IN Mailbox is already busy, this endpoint returns a NAK handshake.	--

8.4 Endpoint 82h 'IN' Transactions (Mailbox In Pipe)

Offset	Number of Bytes	Description	Default Value
--	8	Host reads data from the Mailbox IN Pipe of this USB port using an interrupt IN transaction. If no bytes are available in the Mailbox, this endpoint returns a NAK handshake..	--

8.5 Endpoint 83h 'IN' Transactions (Status Input Pipe)

Offset	Number of Bytes	Description	Default Value
--	1	Host reads this endpoint using an interrupt IN transaction. The value returned is the number of bytes available in the Data IN endpoint. If no data is available, this endpoint returns a NAK handshake. A value of 0 indicates a zero-length packet is available.	--

9 Electrical Specifications

9.1 Absolute Maximum Ratings

Conditions that exceed the Absolute Maximum limits may destroy the device.

Symbol	Parameter	Conditions	Min	Max	Unit
V _{DD}	Supply Voltage	With Respect to Ground	-0.3	4.0	V
V _{DDPLL}	PLL Supply Voltage	With Respect to Ground	-0.3	4.0	V
V _I	DC input voltage	With Respect to Ground	-0.3	V _{DD} +0.5	V
I _{OUT}	DC Output Current, per pin		-30	30	mA
T _{STG}	Storage Temperature	No bias	-65	150	°C
T _{AMB}	Ambient temperature	Under bias	-40	85	°C
V _{ESD}	ESD Rating	R = 1.5K, C = 100pF		2	KV

9.2 Recommended Operating Conditions

Conditions that exceed the Operating limits may cause the device to function incorrectly.

Symbol	Parameter	Conditions	Min	Max	Unit
V _{DD}	Supply Voltage		3.0	3.6	V
V _{DDPLL}	PLL Supply Voltage		3.0	3.6	V
V _I	Input Voltage		0	V _{DD}	V
T _A	Operating Temperature		0	70	°C
t _R	Input rise time		1	10	ns/V
t _F	Input fall time		1	10	ns/V
F _{OSC}	Oscillator or Crystal Frequency		5.985	6.015	MHz

9.3 DC Specifications

9.3.1 DC Specifications

Operating Conditions: V_{DD}: 3.3V ± 5%, T_A = 0°C to 70°C

All typical values are at V_{DD} = 3.3V and T_A = 25°C

Operating Conditions: Notes 1, 2.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I _{DD}	V _{DD} Supply Current	V _{DD} = 3.3V		30	45	mA
I _{DDs}	V _{DD} Supply Current (Suspend)	Device suspended		5	75	μA
V _{IH}	High Level Input Voltage		2.0			V
V _{IL}	Low Level Input Voltage				0.8	V
I _{IL}	Input Leakage Current	Any pin	-1		1	μA
I _{OZ}	Off State Leakage Current		-1		1	μA
C _{I/O}	Input or Output Pin Capacitance	Pin to GND			10	pF

9.3.2 USB Port DC Specifications

Operating Conditions: V_{DD} : 3.3V $\pm$ 5%, T_A = 0°C to 70°C

All typical values are at V_{DD} = 3.3V and T_A = 25°C

Operating Conditions: Notes 1,2.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{DI}	Differential Input Sensitivity	$ (D+) - (D-) $	0.2			V
V_{CM}	Differential Common Mode Range	Includes VDI range	0.8		2.5	V
V_{SE}	Single Ended Receiver Threshold		0.8		2.0	V
V_{OH}	Static Output High	R_L of 15 K Ω to GND	2.8		3.6	V
V_{OL}	Static Output Low	R_L of 1.5 K Ω to 3.6V			0.3	V
I_{LO}	Hi-Z State Data Line Leakage	$0V < V_{IN} < 3.3V$	-10		+10	μA
C_{IO}	I/O Capacitance	Pin to GND			20	pF

9.3.3 USB Port AC Specifications

Operating Conditions: V_{DD} : 3.3V $\pm$ 5%, T_A = 0°C to 70°C

All typical values are at V_{DD} = 3.3V and T_A = 25°C

Operating Conditions: Notes 1,2,3.

Symbol	Parameter	Conditions	Waveform	Min	Typ	Max	Unit
T_R	Rise & Fall Times	$C_L = 50$ pF	Figure 9-1	4		20	ns
T_F		Notes 4,5		4		20	
T_{RFM}	Rise/Fall time matching	(T_R / T_F)	Figure 9-1	90		110	%
V_{CRS}	Output Signal Crossover Voltage			1.3		2.0	V
Z_{DRV}	Driver Output Resistance	Steady State Drive		10		15	Ω
T_{DRATE}	Data Rate			11.97	12	12.03	Mbs
T_{DDJ1}	Source Differential Driver Jitter to Next Transition	Notes 6,7.	Figure 9-2	-3.5	0	3.5	ns
T_{DDJ2}	Source Differential Driver Jitter for Paired Transitions	Notes 6,7	Figure 9-2	-4.0	0	4.0	ns
T_{DEOP}	Differential to EOP Transition Skew	Note 7	Figure 9-3	-2	0	5	ns
T_{EOPT}	Source EOP Width	Note 7	Figure 9-3	160	167	175	ns
T_{JR1}	Receiver Data Jitter Tolerance to Next Transition	Note 7	Figure 9-4	-18.5	0	18.5	ns
T_{JR2}	Receiver Data Jitter Tolerance for Paired Transitions	Note 7	Figure 9-4	-9	0	9	ns

T_{EOPR1}	EOP Width at Receiver; Must reject as EOP	Note 7	Figure 9-3	40			ns
T_{EOPR2}	EOP Width at Receiver; Must accept as EOP	Note 7	Figure 9-3	80			ns

9.3.4 USB Port AC/DC Specification Notes

1. All voltages measured from the local ground potential, unless otherwise specified.
2. All timings use a capacitive load (C_L) to ground of 50 pF, unless otherwise specified.
3. Full Speed timings have a 1.5 k Ω pull-up to 3.3 V on the D+ data line.
4. Measured from 10% to 90% of the data signal.
5. The rising and falling edges should be smoothly transitioning (monotonic).
6. Timing difference between the differential data signals.
7. Measured at crossover point of differential data signals.
8. The maximum load specification is the maximum effective capacitive load allowed that meets the target hub V_{BUS} droop of 330 mV.

9.3.5 USB Port AC Waveforms

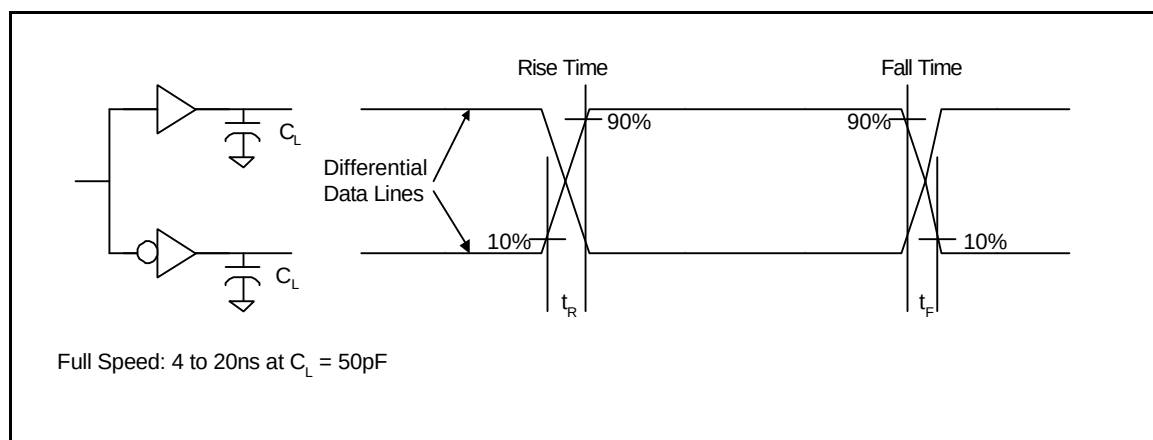


Figure 9-1. Data Signal Rise and Fall Time

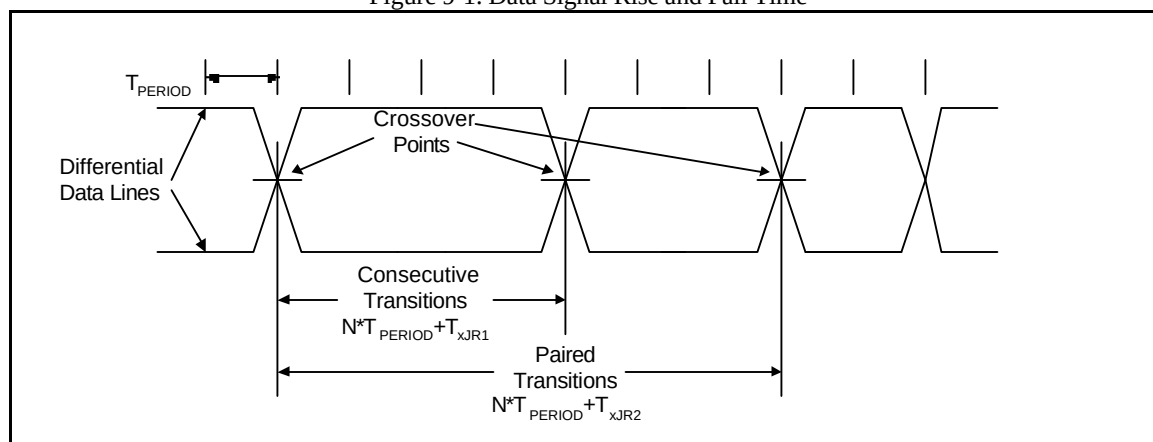


Figure 9-2. Differential Data Jitter

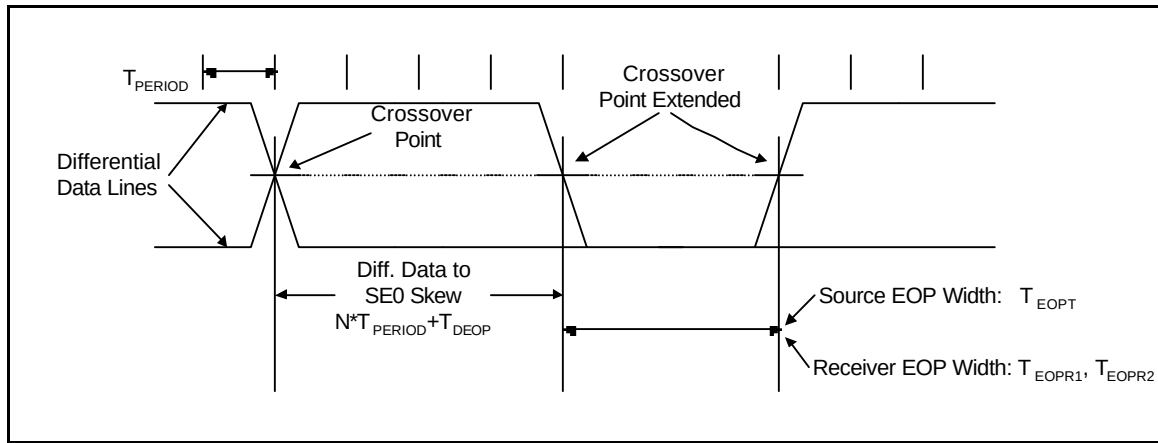


Figure 9-3. Differential to EOP Transition Skew and EOP Width

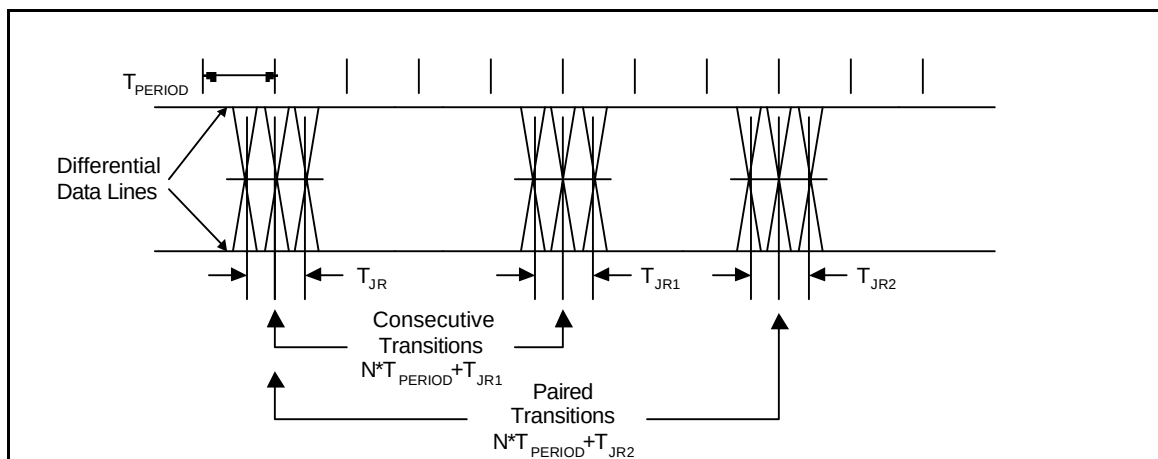
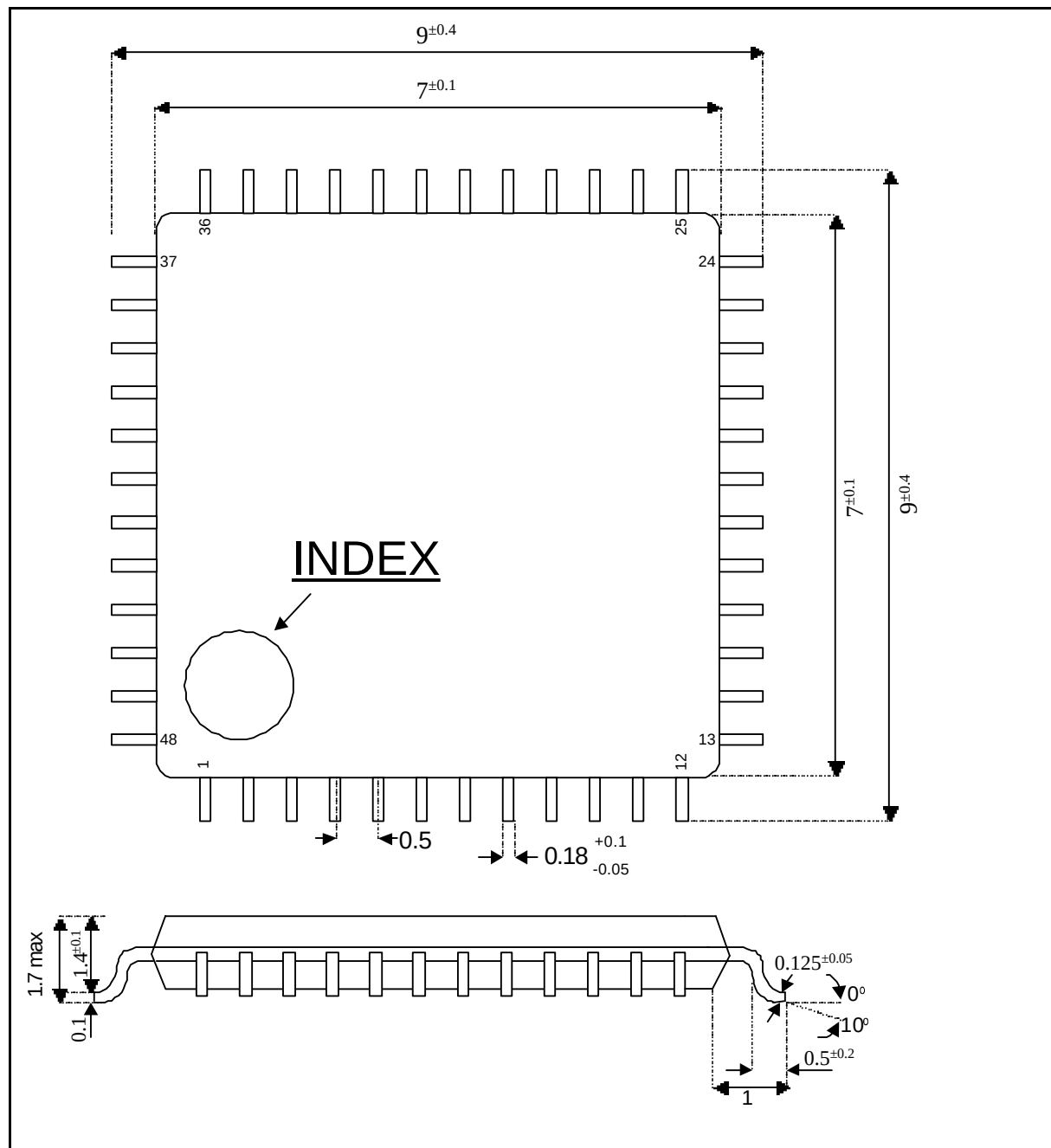


Figure 9-4. Receiver Jitter Tolerance

10 Mechanical Drawing



All dimensions in millimeters.

©NetChip Technology, Inc., 1999

335 Pioneer Way, Mountain View, California 94041

TEL (650) 526-1490 FAX (650) 526-1494

<http://www.netchip.com>

Revision 0.5, November 8, 1999

EXHIBIT B

Folder 002

FSUSB63 — 3:1 High-Speed USB 2.0 Switch / Multiplexer

Features

Switch Type	3:1 USB Switch
USB	USB 2.0 High-Speed & Full-Speed Compliant
Break-Before-Make Time	126µs
R _{ON}	6Ω Typical
C _{ON}	6pF Typical
Bandwidth	830MHz
V _{CC}	2.7 to 4.4V
V _{CNTRL}	0 to V _{CC}
Operating Temperature	-40°C to 85°C
I _{CCSLP}	<1µA
I _{CCACT}	7.5µA Typical
Package	12- Lead UMLP 1.80 x 1.80 x 0.55mm, 0.40mm pitch
Top Mark	KG
Ordering Information	FSUSB63UMX

Applications

- Cell Phone, Digital Camera, Notebook
- LCD Monitor, TV, and Set-Top Box
- Netbook, Mobile Internet Device (MID)

Description

The FSUSB63 is a bi-directional, low-power, High-Speed (HS) USB 2.0 3:1 Multiplexer (MUX). It is optimized for switching among three high-speed (480Mbps) sources or any combination of high-speed and full-speed (12Mbps) USB sources, such as an application processor, to one USB 2.0 connector.

The FSUSB63 has a break-before-make time to force re-enumeration by the host when switching between different HS USB 2.0 controllers and thus requires minimal software changes.

The FSUSB63 is compliant with the requirements of USB 2.0 and features extremely low on capacitance (C_{ON}). The wide bandwidth exceeds the requirement to pass the third harmonic, resulting in signals with minimum edge and phase distortion. Superior channel-to-channel crosstalk also minimizes interference.

Related Resources

- For samples and questions, please contact:
Analog.Switch@fairchildsemi.com.
- FSUSB63 Demonstration Board

Typical Application

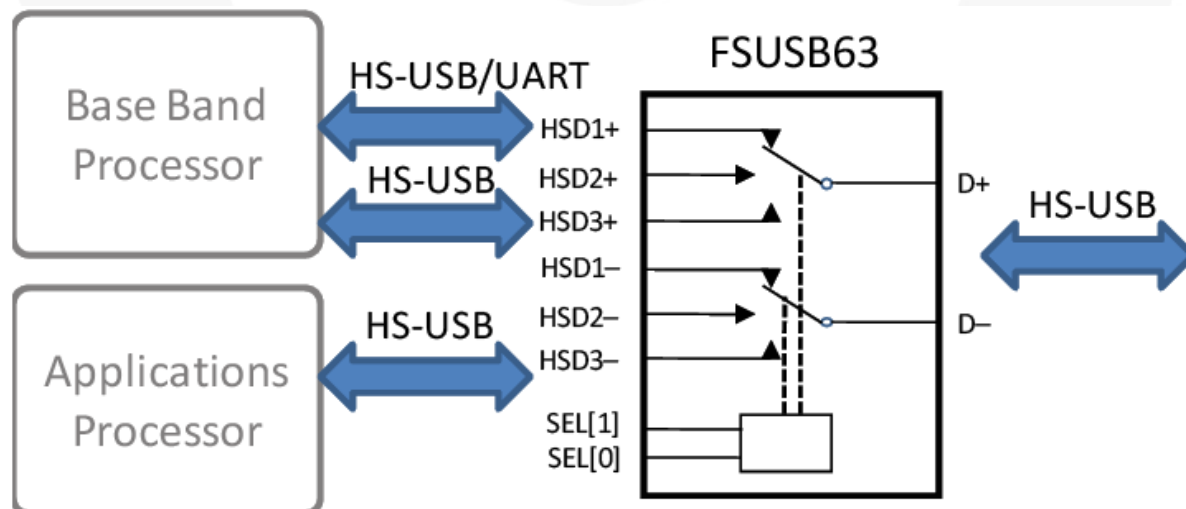
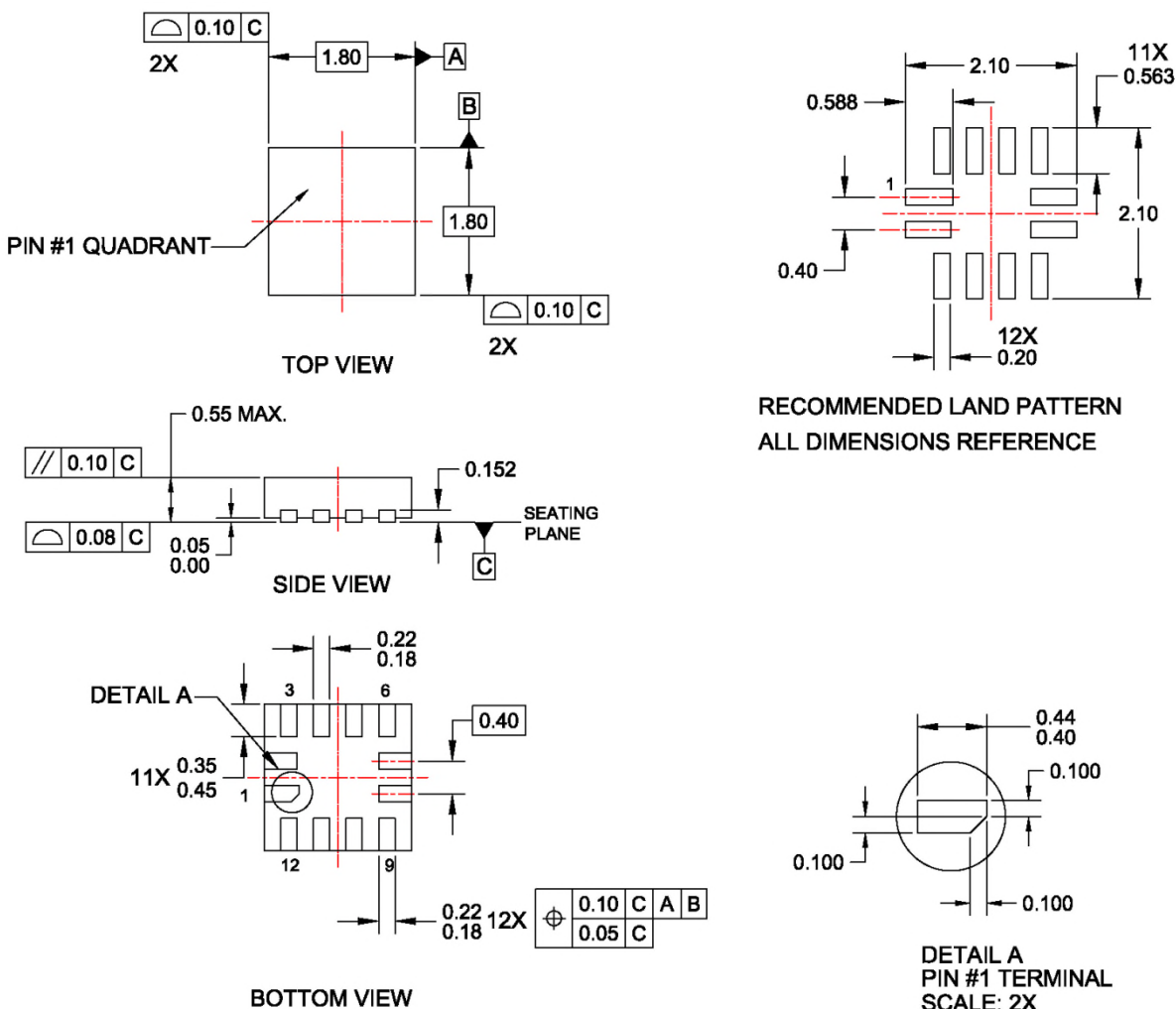


Figure 1. Analog Symbol

Physical Dimensions



NOTES:

- A. DIMENSIONS ARE IN MILLIMETERS.
- B. DIMENSIONS AND TOLERANCES PER ASME Y14.5M, 1994
- C. LANDPATTERN IS A MINIMAL TOE DESIGN
- D. DRAWING FILENAME: UMLP12Arev3

Figure 17. 12-Lead, Ultrathin Molded Leadless Package (UMLP)

Ordering Information

Part Number	Top Mark	Operating Temperature Range	Package
FSUSB63UMX	KG	-40 to +85°C	12-Lead, Quad, Ultrathin Molded Leadless Package (UMLP), 1.8mm x 1.8mm x 0.55mm, 0.4mm pitch

Package drawings are provided as a service to customers considering Fairchild components. Drawings may change in any manner without notice. Please note the revision and/or date on the drawing and contact a Fairchild Semiconductor representative to verify or obtain the most recent revision. Package specifications do not expand the terms of Fairchild's worldwide terms and conditions, specifically the warranty therein, which covers Fairchild products.

Always visit Fairchild Semiconductor's online packaging area for the most recent package drawings:
<http://www.fairchildsemi.com/packaging/>



TRADEMARKS

The following includes registered and unregistered trademarks and service marks, owned by Fairchild Semiconductor and/or its global subsidiaries, and is not intended to be an exhaustive list of all such trademarks.

AccuPower™
Auto-SPM™
Build it Now™
CorePLUS™
CorePOWER™
CROSSVOL™
CTL™
Current Transfer Logic™
DEUXPEED®
Dual Cool™
EcoSPARK®
EfficientMax™
ESBC™
F[®]
Fairchild®
Fairchild Semiconductor®
FACT Quiet Series™
FACT®
FAST®
FastvCore™
FETBench™
FlashWriter®
FPS™

F-PFS™
FRFET®
Global Power Resource™
Green FPS™
Green FPS™ e-Series™
Gmax™
GTO™
IntelliMAX™
ISOPANAR™
MegaBuck™
MICROCOUPLER™
MicroFET™
MicroPak™
MicroPak2™
MillerDrive™
MotionMax™
Motion-SPM™
OptoHit™
OPTOLOGIC®
OPTOPLANAR®
PDP SPM™

Power-SPM™
PowerTrench®
PowerXS™
Programmable Active Droop™
QFET®
QST™
Quiet Series™
RapidConfigure™
Saving our world, 1mW/W/kW at a time™
SignalWise™
SmartMax™
SMART START™
SPM®
STEALTH™
SuperFET®
SuperSOT™.3
SuperSOT™.6
SuperSOT™.8
SupreMOS®
SyncFET™
Sync-Lock™

SYSTEM GENERAL®
The Power Franchise®
the power franchise
TinyBoost™
TinyBuck™
TinyCalc™
TinyLogic®
TINYOPTO™
TinyPower™
TinyPWM™
TinyWire™
TriFault Detect™
TRUECURRENT™*
µSerDes™
SerDes®
UHC®
Ultra FRFET™
UniFET™
VCX™
VisualMax™
XS™

* Trademarks of System General Corporation, used under license by Fairchild Semiconductor.

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION, OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS. THESE SPECIFICATIONS DO NOT EXPAND THE TERMS OF FAIRCHILD'S WORLDWIDE TERMS AND CONDITIONS, SPECIFICALLY THE WARRANTY THEREIN, WHICH COVERS THESE PRODUCTS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury of the user.
2. A critical component in any component of a life support, device, or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

ANTI-COUNTERFEITING POLICY

Fairchild Semiconductor Corporation's Anti-Counterfeiting Policy. Fairchild's Anti-Counterfeiting Policy is also stated on our external website, www.fairchildsemi.com, under Sales Support.

Counterfeiting of semiconductor parts is a growing problem in the industry. All manufacturers of semiconductor products are experiencing counterfeiting of their parts. Customers who inadvertently purchase counterfeit parts experience many problems such as loss of brand reputation, substandard performance, failed applications, and increased cost of production and manufacturing delays. Fairchild is taking strong measures to protect ourselves and our customers from the proliferation of counterfeit parts. Fairchild strongly encourages customers to purchase Fairchild parts either directly from Fairchild or from Authorized Fairchild Distributors who are listed by country on our web page cited above. Products customers buy either from Fairchild directly or from Authorized Fairchild Distributors are genuine parts, have full traceability, meet Fairchild's quality standards for handling and storage and provide access to Fairchild's full range of up-to-date technical and product information. Fairchild and our Authorized Distributors will stand behind all warranties and will appropriately address any warranty issues that may arise. Fairchild will not provide any warranty coverage or other assistance for parts bought from Unauthorized Sources. Fairchild is committed to combat this global problem and encourage our customers to do their part in stopping this practice by buying direct or from authorized distributors.

PRODUCT STATUS DEFINITIONS

Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative / In Design	Datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	Datasheet contains preliminary data; supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice to improve design.
No Identification Needed	Full Production	Datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice to improve the design.
Obsolete	Not In Production	Datasheet contains specifications on a product that is discontinued by Fairchild Semiconductor. The datasheet is for reference information only.

Rev. I50

JURAT ATTACHMENT

A notary public or other officer completing this certificate verifies only the identity of the individual who signed the document to which this certificate is attached, and not the truthfulness, accuracy, or validity of that document.

STATE OF Texas }

COUNTY OF Tarrant }

The foregoing instrument was subscribed and sworn before me this date of
08/31/2023, by Nathaniel Frank-White

This notarial act was an online notarization.



(Notary Seal)

Notary's Signature *Brittany Rene Copeland*

Registration No.: 133645929

Commission Expiration Date: March 15, 2026

