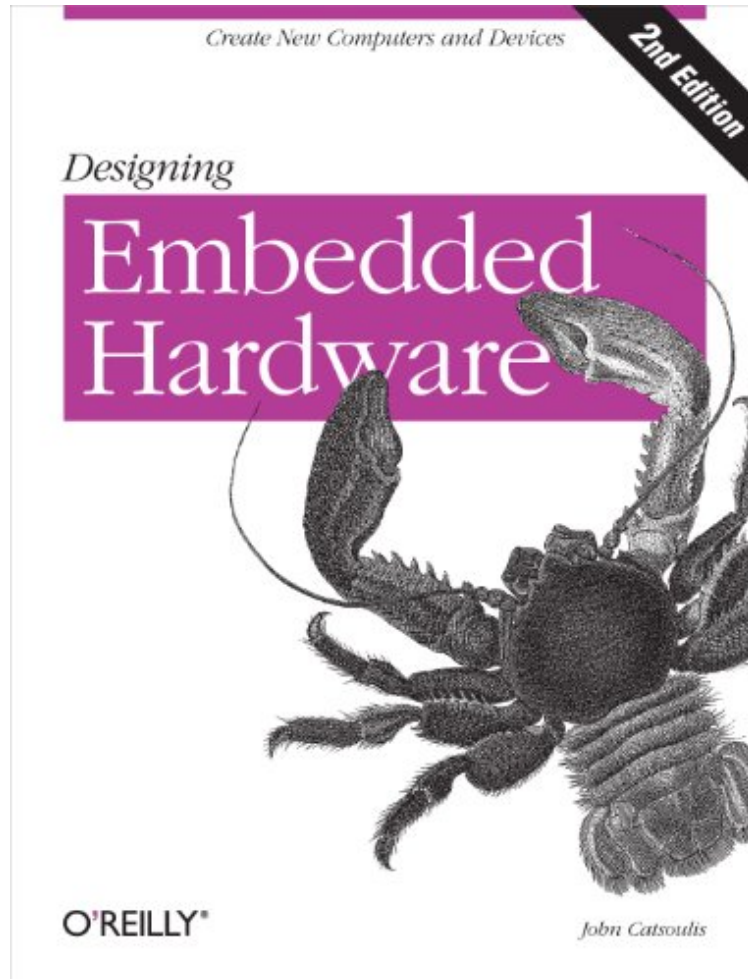


(Mobile book) Designing Embedded Hardware: Create New Computers and Devices

Designing Embedded Hardware: Create New Computers and Devices

Von John Catsoulis

DOC | *audiobook | ebooks | Download PDF | ePub



DOWNLOAD 

 READ ONLINE

Produktinformation -Verkaufsrank: #373973 in eBooksVerffentlicht am: 2005-05-16Erscheinungsdatum: 2009-06-30File Name: B0043D2E4U | File size: 73.Mb

Von John Catsoulis : Designing Embedded Hardware: Create New Computers and Devices before purchasing it in order to gage whether or not it would be worth my time, and all praised Designing Embedded Hardware: Create New Computers and Devices:

KundenrezensionenHilfreichste Kundenrezensionen2 von 2 Kunden fanden die folgende Rezension hilfreich. Spannt den Bogen von der Theorie in die PraxisVon R. B.Sehr praxisnahe Erklrung von Hardwaredesign / Platinendesign. Immer konkret am Beispiel mit vielen konkreten Schaltplnen. Wer nach der Uni das Gefhlt hat, alles und doch gar nichts zu wissen. Wer denkt, trotz 3 E-Technik, 3 Informatik und 2 Regelungstechnik Scheinen nichtmal einen elektronischen Wecker selber bauen zu knnen, der findet hier den Bogen von der Theorie in die Praxis.Kapitel: - Computerarchitektur (zzgl. zu eventuellem Vorwissen aus der Uni kommen hier interessante Zusatzdetails, z.B. Low-

Power Architektur) - Assembler (am besten berspringen und C/C++ nehmen) - Progrsprache Forth (am besten berspringen und C/C++ nehmen) - Elektronik 1 mal 1 (Erst etwas Grundwissen (nur zur Auffrischung geeignet), dann folgt: Schematics verstehen, worauf bei Datenblttern achten, auf den Stromfluss achten (auch wenn Logik in U ist, spielt I eine Rolle), Tristate-Eingnge, PullUpResistor, wie vermeide ich Schden etc.) - Power Sources (wie bekomme ich Batteriespannung/WallJack Power an meine Bauteile. UND ZWAR MIT WELCHEN CHIPS KONKRET, WIE GENAU, WIE SIEHT DIE SCHALTUNG AUS !) - Building Hardware (eigene Prototypplatinen lten, prof. Platinen mit Autorouter erstellen, selber routen (worauf achten), Breadboards, Packaging, alles (etwas knapp) vom Ltkolben bis zum PCB-Druck, was man fr eigene Platinen braucht ...) - SPI (nicht nur Protokoll aus Adlerperspektive, sondern ganz genau, welche Chips, was wie verbinden etc.) - I2C (auch wieder konkrete Chips, z.B. konkret ein I2C Display anschlieen als Schaltung) - Serial Ports (auch wieder konkret mit Schaltplan) - IRDA - USB - Netzwerke (CAN/Ethernet: Mit Focus auf den konkreten Schaltplnen. Danach wei man ganz genau, wie man (mit den gegebenen Beispielchips) Ethernet auf seine Testplatine bringt. Wow !) - Analogtechnik (Nicht die langweilige "E-Technik-Algebra" zur Messtechnik. Hier folgen konkrete Schaltungen, konkrete Chips, genaue Bestckung (am Beispiel), wie bekomme ich das U-Signal in meinen uC ...) - PIC Mikroprozessor (auch hier wieder die Konkrete Beschaltung am Beispiel als Schematic. Z.B. Motorsteuerung, wie sieht so eine Schaltung in der Praxis am Beispiel aus, incl. dazu passender Spannungsversorgung etc.) - AVR Mikroprozessoren (auch wieder die genaue Beschaltung als Beispiel. Was muss alles dran. Ein uC braucht einen Pulstakt, wo kommt der genau her (Quarz wie genau dranhngen). Wie genau Adressbus, Datenbus, RAM-Chip verbinden etc.) - 68HC11 Mikroprozessoren - MAXC Mikroprozessoren - 68000 Mikroprozessoren (Etwas knapp. Wenn noch jemand den "Commodore Amiga" kennt, der verwendete einen derartigen Prozessor. Hier steht, wie man einen derart leistungsfhigen 32-Bit-Prozessor auf eigenen Platinen nutzen kann) - DSP (Etwas knapp. Wie man ein Board nur von einem DSP (als Hauptprozessor) steuern lassen knnte, wenn man intensive digitale Berechnungen hat. Wie immer Focus auf Beschaltung, was verbinde ich wie wo genau ...)Es wird auch auf Packaging eingegangen (etwas zu wenig / es gibt Chips in verschiedenen Bauformen). Gold wert sind die Tips am Rande. Z.B. dass man immer einen Entkopplungskondensator an den VCC-Eingang hngen sollte.Der Leser wird imho gut in die Lage versetzt, kleine Computer von A bis Z zu entwerfen (und zwar weit komplexerer Systeme, als nur die eingangs erwhte eine Digitaluhr). Auf groe Embedded Systeme mit Embedded Betriebssystemen geht der Author nicht ein. Behandelt werden eher sehr kleine Computer. Fr die ersten Schritte im Bereich Hardwareentwicklung finde ich das genau richtig. Auch ich sage: Kaufen !Ein Studium, Techniker oder E-Tech-Ausbildung ist fr die Lektre imho sehr hilfreich. Elektrotechnik Grundlagen sollten bereits vorhanden sein. Die sind leider viel langweiliger, als dieses wirklich sehr interessante Buch, muss man aber durch, wenn noch nicht vorhanden: Z.B. "Basiswissen Gleich- und Wechselstromtechnik" von Vieweg. Mit Softwareentwicklung sollte man auch schon erste Erfahrung haben, denn das "erste eigene board" mchte ja etwas zu tun bekommen :)Liest sich sehr flssig und ist dank der vielen Bilder und des hohen Tempos immer spannend.0 von 0 Kunden fanden die folgende Rezension hilfreich. Angenehmes BuchVon GreedenceDas Buch liest sich locker und kommt unaufgeregt daher. Es beginnt nahezu bei 0, erklrt die einzelnen Bauteile und wie man die Selben verbindet (PCB und co). Schade nur dass der Transistor ausgelassen wurde.Bin jetzt mit dem ersten 1/3 durch und ab sofort geht es um Microcontroller, Clock und Co. Ich freue mich darauf (Ein Stern abzug wegen fehlenden Transistor)0 von 0 Kunden fanden die folgende Rezension hilfreich. Schnes Buch mit konkreten HinweisenVon XA450Wer sich mit Mikrocontrollern und Konsorten beschftigen will und neu ist, sollte sich dieses Buch anschauen. Hier werden viele wichtige und grundlegende Dinge diesbezüglich mit Schaltplan und Nennung von konkreten Chips beschrieben hat.

KurzbeschreibungEmbedded computer systems literally surround us: they're in our cell phones, PDAs, cars, TVs, refrigerators, heating systems, and more. In fact, embedded systems are one of the most rapidly growing segments of the computer industry today.Along with the growing list of devices for which embedded computer systems are appropriate, interest is growing among programmers, hobbyists, and engineers of all types in how to design and build devices of their own. Furthermore, the knowledge offered by this book into the fundamentals of these computer systems can benefit anyone who has to evaluate and apply the systems.The second edition of Designing Embedded Hardware has been updated to include information on the latest generation of processors and microcontrollers, including the new MAXQ processor. If you're new to this and don't know what a MAXQ is, don't worry--the book spells out the basics of embedded design for beginners while providing material useful for advanced systems designers.Designing Embedded Hardware steers a course between those books dedicated to writing code for particular microprocessors, and those that stress the philosophy of embedded system design without providing any practical information. Having designed 40 embedded computer systems of his own, author John Catsoulis brings a wealth of real-world experience to show readers how to design and create entirely new embedded devices and computerized gadgets, as well as how to customize and extend off-the-shelf systems.Loaded with real examples, this book also provides a roadmap to the pitfalls and traps to avoid. Designing Embedded Hardware includes:The theory and practice

of embedded systems
Understanding schematics and data sheets
Powering an embedded system
Producing and debugging an embedded system
Processors such as the PIC, Atmel AVR, and Motorola 68000-series
Digital Signal Processing (DSP) architectures
Protocols (SPI and I2C) used to add peripherals
RS-232C, RS-422, infrared communication, and USB
CAN and Ethernet networking
Pulse Width Monitoring and motor control
If you want to build your own embedded system, or tweak an existing one, this invaluable book gives you the understanding and practical skills you need.