

eZ430-Chronos Wireless Watch Development Tool: Teardown & Getting Started



www.ti.com/chronoswiki

eZ430-Chronos for wireless networking applications

Complete hardware,
software and
support community

Simplify & inspire development with
world's first customizable tool within
an intelligent sports watch

Unparalleled system
integration and
ultra-low power

Easily enable wireless connectivity,
longer battery life, improved
ergonomics

Low cost development
kit at \$49

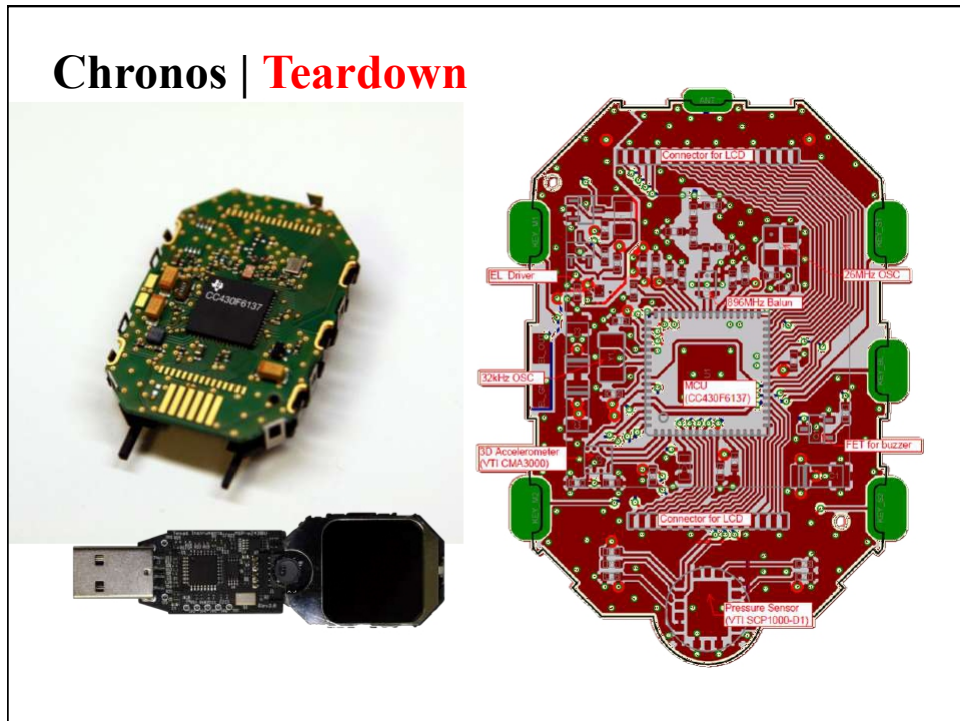
Increase accessibility and reduce
development cost



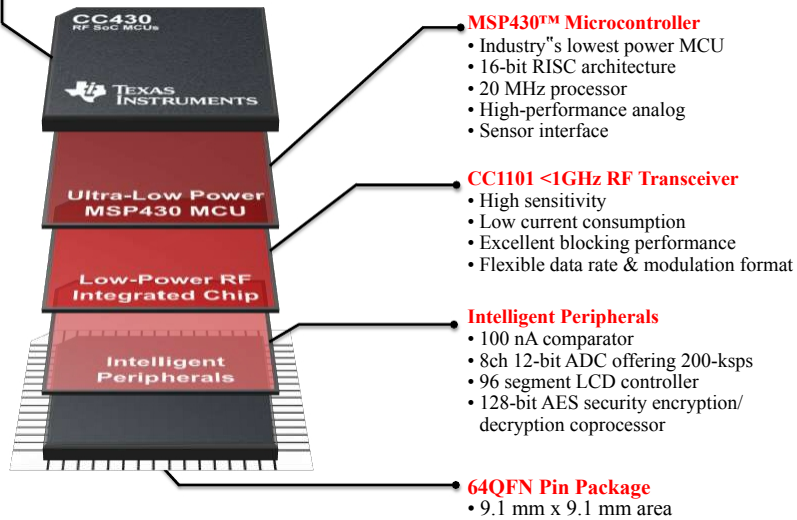
Chronos | **Advanced Integration**



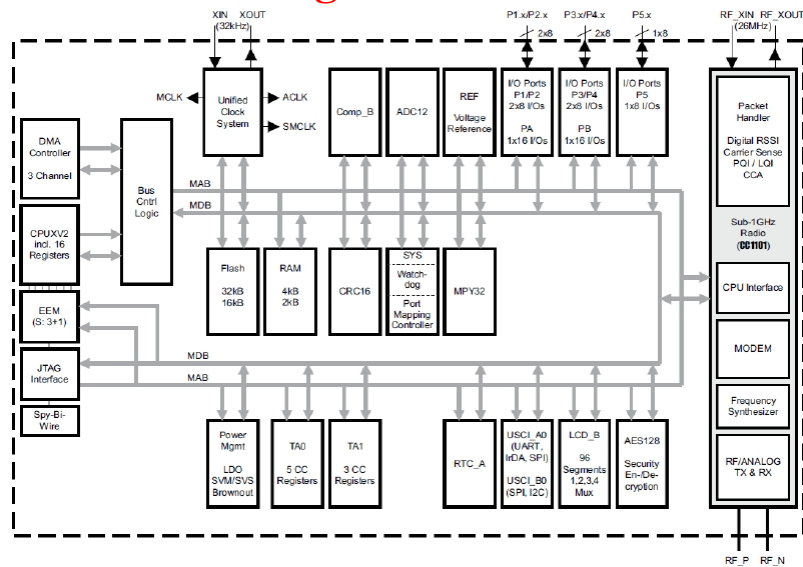
Chronos | **Teardown**



CC430 | RF + Ultra-Low Power MCU



CC430 Block Diagram



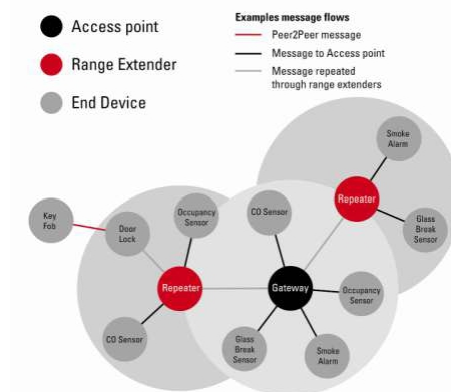
Chronos | The Software

- Free **development software**
 - Code Composer Studio
 - IAR Embedded Workbench
 - MSPGCC*
- Production-ready, **open-source** projects
- **RF stacks available**
 - SimpliciTI
 - BlueRobin
 - W-Mbus
 - 6LoWPAN
 - More coming...
- **User generated apps** and support on www.ti.com/chronoswiki



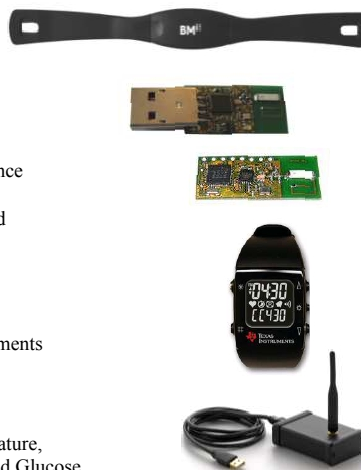
SimpliciTI

- TI proprietary **low-power RF** network protocol
- Low Cost:
 - < 8K FLASH
 - <1K RAM
- Flexible:
 - simple **star** w/ extendor
 - **p2p** communication
- Simple: Utilizes a very **basic** core API
- Low Power: Supports **sleeping** devices



BM Wireless' BlueRobin Key Facts

- **BlueRobin™** targets at
 - body area networks
 - long range monitoring systems
- **BlueRobin** provides
 - ultra-low power operation in TX and RX mode
 - multi-user support with patented collision avoidance
 - bi-directional and long range communication
 - remote data storage with automatic data download
 - built-in data encryption
- **BlueRobin** offers flexibility through
 - hardware independent implementation
 - small memory footprint and low resource requirements
 - support of all ISM bands (433MHz to 2.4GHz)
- **BlueRobin** key apps
 - Heart Rate, Speed, Distance, Steps, GPS, Temperature,
 - Altitude, Rotations, Weight, Blood Pressure, Blood Glucose

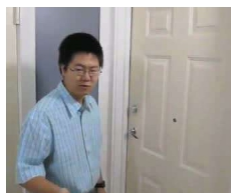


Chronos | Projects

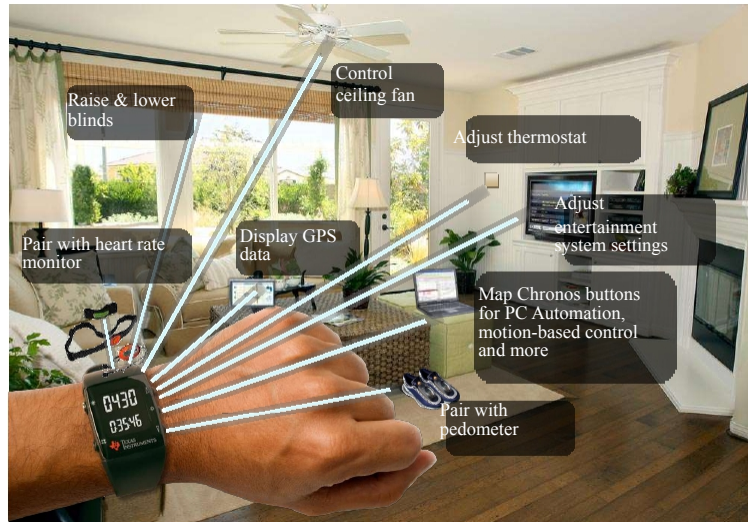
- Watch functions: time, date, alarm, stopwatch
- Fitness function: running speed, distance, heart rate, calories burned
- Sensor data logging w/ wireless PC download
- PowerPoint Control
- Media Remote
- Motion-based mouse/PC game control
- Wireless App Updating
- Wireless door lock
- Virtual Theremin
- Robotics control

More apps to come...

www.ti.com/chronoswiki



Endless possibilities | Chronos serves as a central hub for nearby wireless sensors

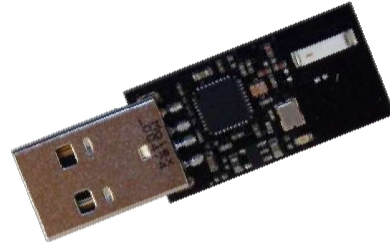


Chronos | Frequency Differences

- Available in 3 different frequencies
 - 433 MHz : Japan, India, WW
 - 868 MHz : Europe
 - 915 MHz : N. & S. America
- 868 & 915
 - Use identical watch hardware
 - Frequency is software selectable
 - 868 MHz RF Access Point has extra 0Ω R
- 433 MHz
 - Still in R&D
 - Available in June (estimated)
 - RF Range predicted to be *worse* than 868/915

Chronos | RF Access Point

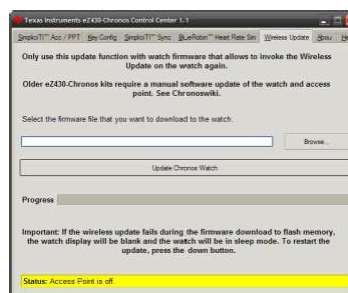
- Based on CC1111 w/ integrated USB + <1GHz RF
- “Fixed function” to communicate with Chronos & PC
- Can be manually reprogrammed with CC Debugger
- Supports wireless updating of Chronos firmware via RF BSL
 - RF BSL not included on first production batch



Header to JTAG signals has to be manually added

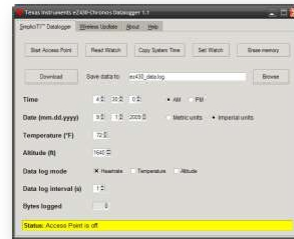
RF BSL | Wireless Updating

- Wireless update of watch firmware
- Small RF Stack resides in BSL memory
- Application must include function to invoke BSL
- Pros:
 - ~10x faster than downloading code via SBW
 - No need to open enclosure
- Cons:
 - No debug capability
 - Power hog
 - Not supported out-of-the-box on early units (before 4/2010)



Chronos | Data Logger

- Chronos can be used as a data logger for:
 - Heart rate
 - Temperature
 - Altitude
- User definable intervals
 - 1 to 255 seconds
- 8kB of Flash memory reserved
- The stored data can be transferred to a PC



PC/Chronos Communication

- RF Access Point open
Virtual COM Port over USB
 - bps 115200
 - Data bits 8
 - Parity None
 - Stop Bits 1
 - Flow Control None
- Control Center automatically
 - Opens COM port
 - Transfers data
 - Decodes packets
 - Displays info
- API available for manual data transfer
- Scripts available for:
 - Python
 - Processing
 - Ruby
 - .net
- All community developed, supported

Chronos | Extra Hardware

- Heart Rate Monitors
 - BM-CS5 (800m)
 - BM-CS5SR (10m): 49€
- Bike Sensors
 - Speed and distance (according to BMi Q4 2010)
- Compatible RF Development Boards
 - [AMB8423](#)
 - [EM430F6137RF900](#)
 - [CC1101EMK433](#)
 - [CC1111EMK868-915](#)



Getting Started: What you need

- **Hardware:** www.ti.com/chronoswiki
 1. eZ430-Chronos
 2. Computer: Windows / Linux
- **Software:**
 1. eZ430-Chronos Software Package: [Windows \(SLAC341\)](#), [Linux \(SLAC388\)](#)
 2. IDE: [IAR](#) or [CCS](#)
 3. Firmware Update tool (adds rfBSL)
- **Documentation**
 1. [eZ430-Chronos User Guide \(SLAU292\)](#): Technical details and instructions for Chronos
 2. [CC430 User Guide](#): Technical user manual for CC430
 3. [CC430F613x Datasheet](#): Electrical specs for CC430F613x

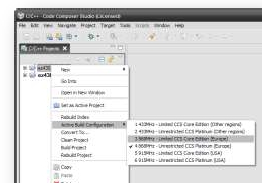
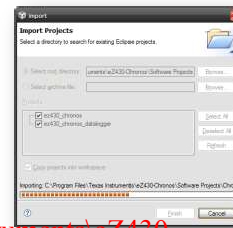


Chronos Software Package

- All files copied to **C:\Program Files\Texas Instruments\ez430-Chronos**
- What's included:
 - Control Center
 - Control Center GUI binary
 - Data logger GUI binary
 - GUI source code
 - Documentation
 - Chronos User Guide
 - Schematics, Layout (Gerbers) & BOM
 - RF Access Point
 - ez430 Debug Interface
 - Watch
- RF Access Point Driver
- Binary images (Recovery)
 - Sports watch/Chronos
 - Datalogger app
 - rfBSL
 - RF Access Point
- Software projects
 - Sports watch (CCS/IAR)
 - Datalogger (CCS/IAR)
 - RF Access Point (IAR)

Working with the Chronos Project (CCS)

- Open CCS
 - Pick any workspace
 - Close welcome screen
- Import Project
 - Project > Import existing...
 - Browse to **C:\Program Files\Texas Instruments\ez430-Chronos\Software Projects**
 - Select „Copy Projects into workspace”
- Select „Active Project”
 - Right Click on project name
- Select „Active Build”
 - Select correct frequency & IDE version (core vs. full)
- Debug



Adding RFBSL to RF Access Point

- Hardware You'll need
 - RF Access Point
 - CC Debugger
- Add connector to RF AP



- Run “**Smart RF Flash Programmer**”
- Load RF Access Point Recovery image
- Erase, Program, & Verify



Adding RF BSL to Chronos Watch

- RF BSL includes a small RF protocol stack with error recovery
- Resides within reserved BSL memory
 - Accessing via IAR/CCS is *tricky*
- Easiest update method
 - Use [Firmware Update Tool](#)
 - Select correct script for watch frequency
 - Automatically updates BSL + User Application
- User App needs to be modified to invoke BSL from menus



This week's lab:

1. get Chronos running with the default software. Demonstrate the accelerometer data on your PC.
2. Wirelessly load the data logger. Log some altitude data with the watch, while you travel twice from the basement to the 6th floor and back down, first trip on the elevator and second via stairs.
3. Plot the data and show how to identify the activity, stairs or elevator.
4. Compile and run the "hello world" project. Then, modify it to display something different, like temperature. To do that, you will probably want to take a good look at the default watch software source code.

