D.Roche, 10/19/2014

PROTOTYPES TO PRODUCTION

www.expataudio.com

www.ti.com

Curriculum

- Before you start!
- Specification to Selection
- Elegance, Power & Design
- Software and Legal Woes
- Intensive Prototyping
- Design for manufacturing
- …

Lesson #1 – Before you start

- **Before you start designing!**
  - Introduction
  - Where this lesson came from
    - Audio Accessory Ebook –SLYC134
- Your idea
- Competitive analysis
- Understanding your distribution channel
- Setting pricing – understanding *value*
- Market Requirements Document
  - Market Targets
  - Costs Targets
  - Physical Constraints
  - Feature Creep
- Hidden costs and surprises
- Our Example

Lesson #2 – Spec to Selection

- From Specification to Selection
- Build a rough physical sketch of your product.
- Build a high level block diagram.
  - Signal flow
  - Control flow
  - Power flow.
- Device Selection
  - Analog vs. Digital signal chain
  - Microcontrollers
  - Clocks
  - Power Efficiency
- Dealing with the unexpected
- Sourcing
  - What parts to design in
  - Where to buy the parts

Lesson #3 – Elegance, Power & Design

- Simple market expectations.
  - Efficiency
  - Size
  - Elegance
- Power Supplies.
  - Modules are your friend
  - Avoid Mains like the plague.
- Look for failure points in your product
- System Partitioning
  - Testing
  - Upgradeability
  - Form Factor
  - Multi-Sku Design
  - Hardware Debug Capability
  - In Circuit Programming
- Lessons learnt with the Soundbar.

Lesson #4 – Software & Legal Woes

- Software ☹
  - Processing
  - Control
  - User Interface
  - Back Doors
- Legal & Compliance Issues
  - Consult a lawyer! ☺ (more to come)

Lesson #5 – Intensive Prototyping

- Break your product. At least twice
  - Power Supply
  - Connectors
  - ESD shock
    - http://www.compliance-club.com/archive/old_archive/010619.htm
  - EMI “test”
    - http://www.eevblog.com/forum/reviews/near-field-probes/

Lesson #6 – Design for Manufacturing

- Assume you won’t hand assemble as much as possible.
  - $$$ and Error Prone
- Document for others to build
- How will you test good products?
  - Test & Programming Jigs etc.
- How will you fix production failures
  - What classes as a fail?
  - Keep and eye on yield. It costs $$$$$$$$

Lesson #1

Before you start!

Quick Intro’s Around the class

- Audio Product Definer @ Texas Instruments
- Run small Pro Audio module business @ Expat Audio
- 13 years in the Industry
  - Sales, Marketing, Definition
- Supported multiple customers from concept to production

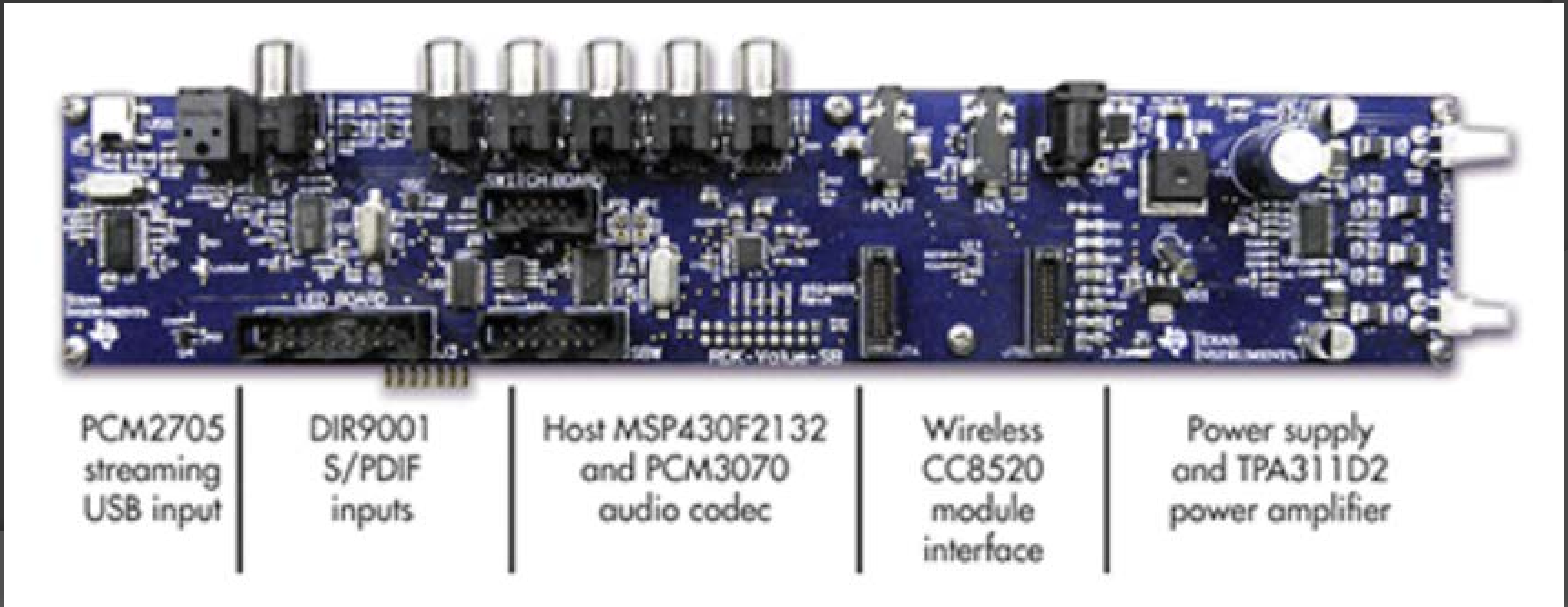
LG Life's Good SAMSUNG SONY make.believe Panasonic ideas for life VIZIO

Why it’s done

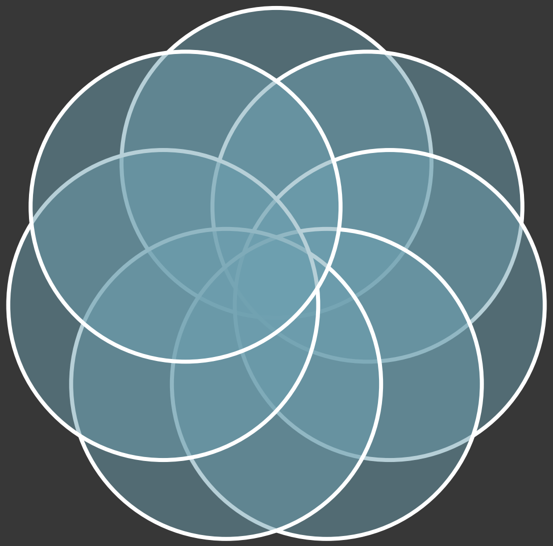
- Too many Kickstarters that get too excited and don’t think it through.
  - Great ideas, poorly executed.
- Bigger Companies have a process for product development. Helps keep discipline.
  - Smaller companies have the advantage of being faster than big, but need discipline to do the boring stuff too.
- Wanted to share the experience developing the Reference Soundbar Design.

Where can I find more details.

- http://www.ti.com/lit/ml/slyc134/slyc134.pdf


PCM2705 streaming USB input
DIR9001 S/PDIF inputs
Host MSP430F2132 and PCM3070 audio codec
Wireless CC8520 module interface
Power supply and TPA3111D2 power amplifier

In most cases, you are everyone!



Marketing

Industrial Design

Definition

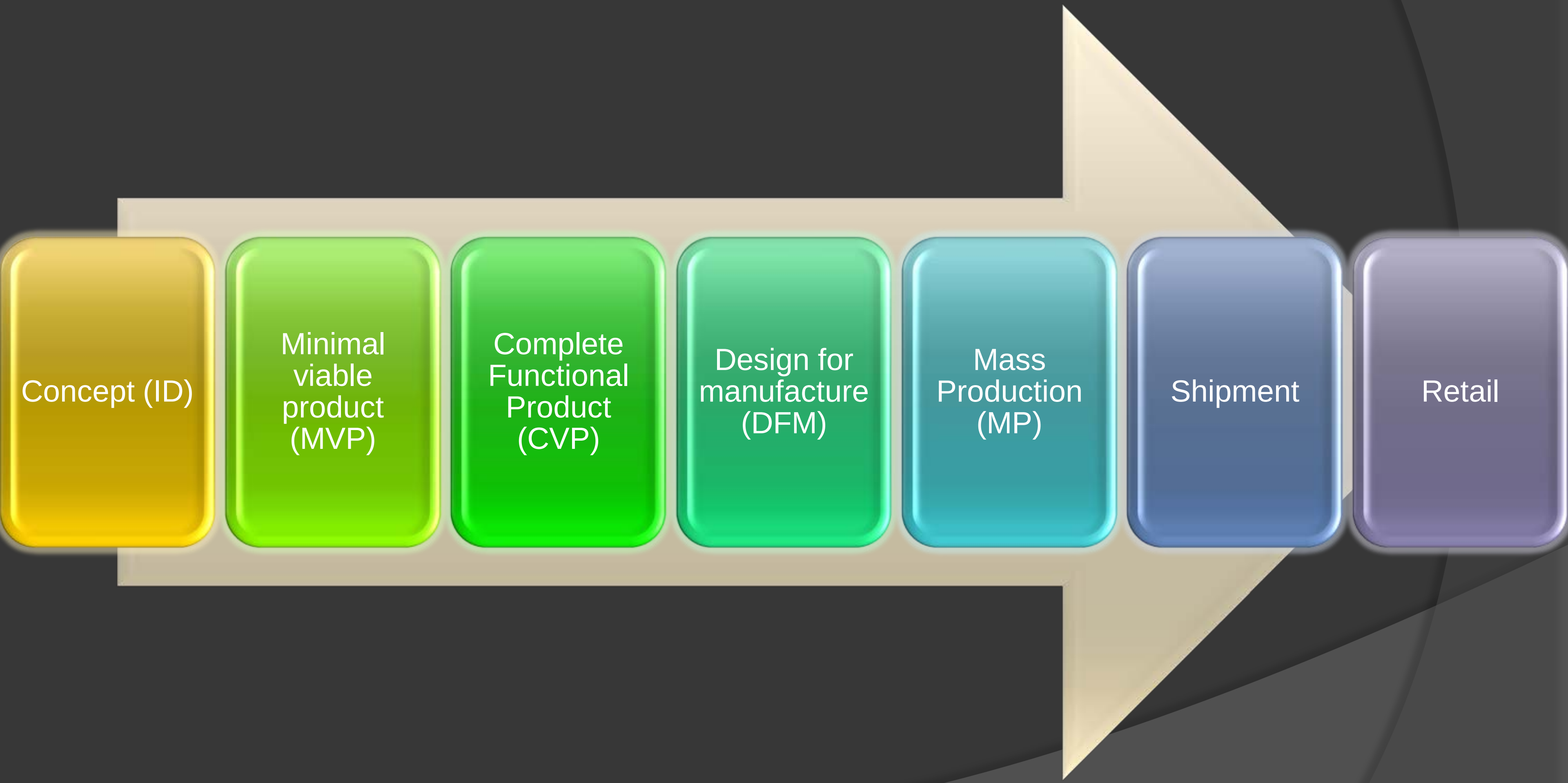
Business

Test Engineer

Electronic Design

Graphic design

Product Development Cycle



Concept (ID)

Minimal viable product (MVP)

Complete Functional Product (CVP)

Design for manufacture (DFM)

Mass Production (MP)

Shipment

Retail

Two starting documents

- Business Case
  - Product Justification
  - Project Boundary Conditions.
  - High Level Financials

  - GO / NO GO

- Market Requirements Document
  - Product Specification

Build your business case

- Validate the idea
- Where do you fit into the market
- What are your costs/time/risks?
  - HW cost
  - Production Costs
  - Distribution costs
  - Costs of doing business.

- What’s the potential return on investment (ROI)

Your Idea

- Always keep a “sea of ideas”
  - Some will be over the top, some will be flops, but keep trying!

- Always be brutal with yourself, and solicit brutal feedback.
  - How is it different?
  - Is that difference really valued in the market, or do I think it’s technically cool?
  - What’s the MVP?

- Balance the investment (time/$$$) with the potential return.
  - Back of a napkin – market size, unit volume, potential sales price, profit margin.

- How easy is it to rip off?
  - Is it worth ripping off?
  - If you make it open source, what differentiation/leadership can you maintain?

- Set your limits. What would kill the project? If it happens KILL IT!

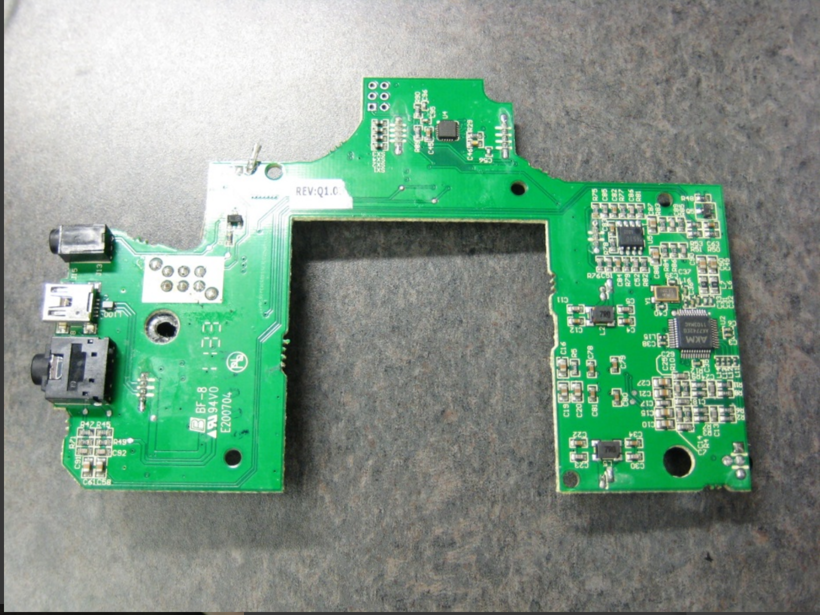
Competitive Analysis

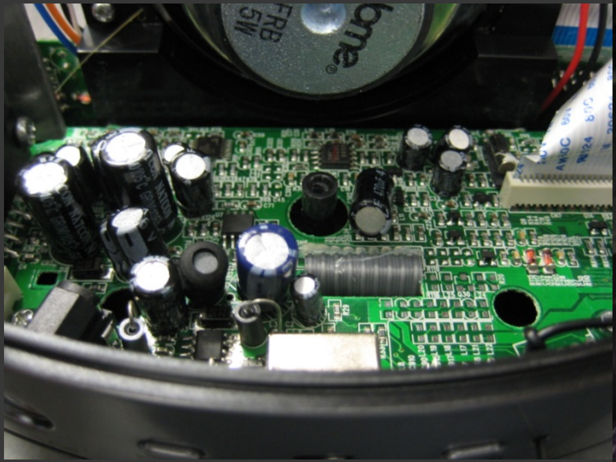
- What's available today?

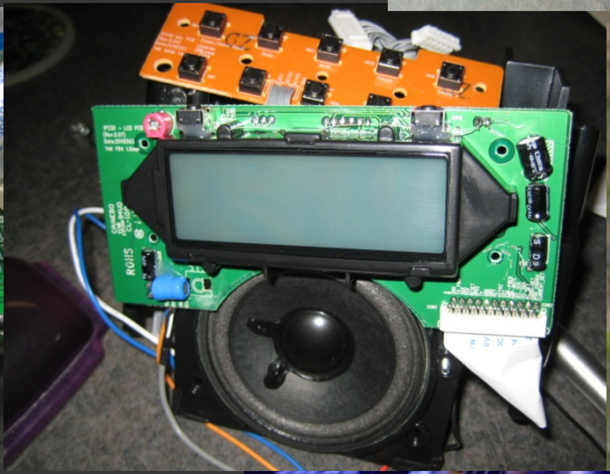
- Buy at least 3 different competitors solutions.
  - Teardown
  - Cost Analysis
  - Feature Analysis
  - Estimate Profit Margin

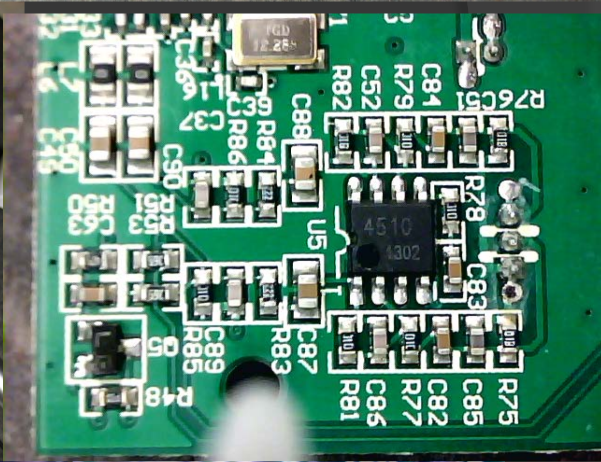
Teardowns

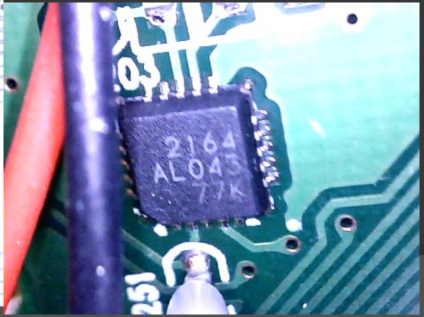
- What does their architecture look like?
- How much would it cost you to make?
- How much further could they cost reduce if they saw your product/

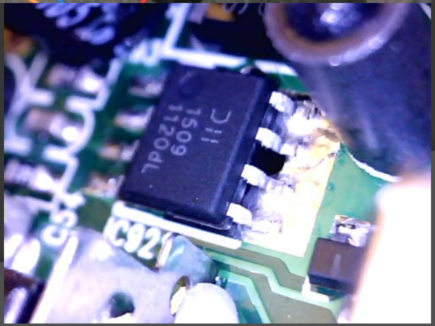


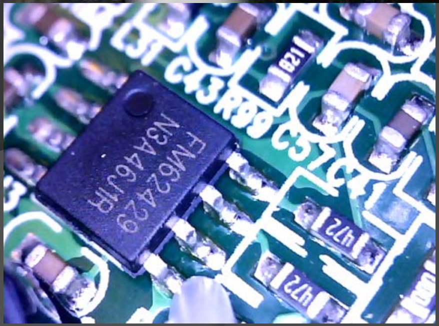












Getting your product to market

- What does your Distribution channel look like?
  - Adafruit? Seeed? Direct? Radioshack? ;)
  - What profit margins are expected?
  - How is your time best spent?

- How will you handle logistics?
  - Orders
  - Shipping
  - Returns
  - Customer Service

  - ALL TAKE TIME! ☹

Setting your cost structure

- Market Targets
- Costs Targets
- Physical Constraints
- Feature Creep
  - Any changes in featureset add time and cost. Analyze – can it be added in gen 2?
  - Better to get revenue quickly and develop next gen whilst being paid!

- PROFIT MARGIN = 1-(COB/Resale)

Setting your pricing

- Your time is not free.

- You’re in business… not doing this for “all your friends on kickstarter”

- ***The value of your product should be set by what the market will pay, not 2x your cost!***

Market Requirements Document

- Specification for the product.
- Starts as this->

- It should become
  - Market Overview (part of bus case)
  - Product Requirements
  - Product Features
  - Costs Targets
  - Physical Constraints
  - Feature Creep

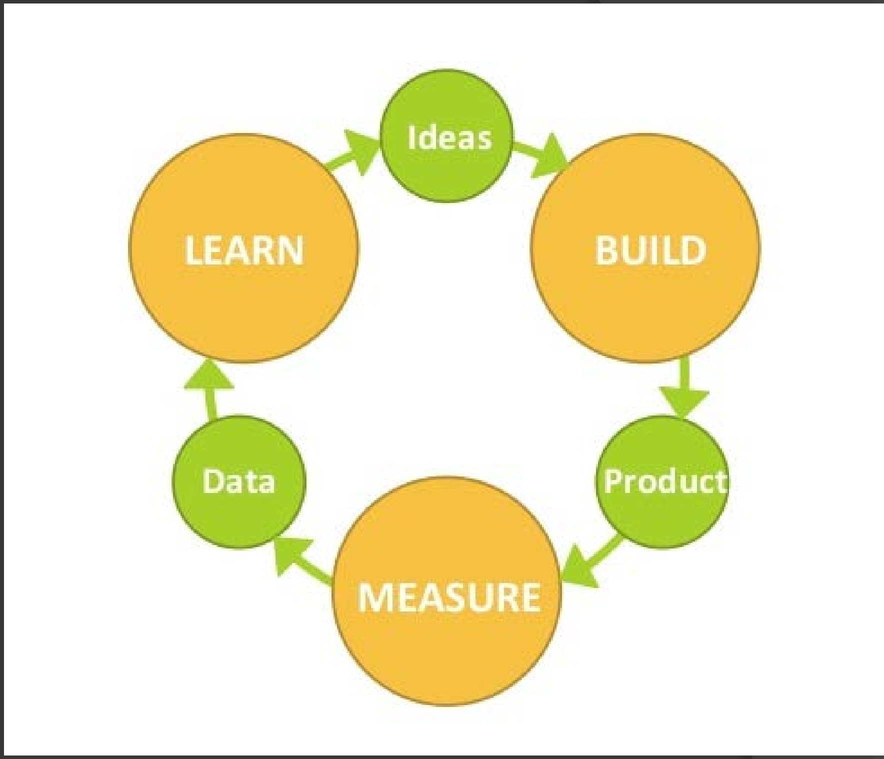
- Even if you are doing it all yourself, put some effort into this.

- Work towards the MVP

The MVP

Steve Blank: "You’re selling the vision and delivering the minimum feature set to visionaries, not everyone."

"*The minimum viable product is that version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort.*



http://www.slideshare.net/chenelaine/lean-startup-hardware-edition-20563840

Hidden Costs and Surprises

- Yield ← don't ignore this!
- Extra Development time.
- Additional Development tools/test equipment
- Certification CE/UL
- Multiple Prototypes to early adopters.

Confessions of kickstarter

- https://medium.com/@mikeestee/confessions-of-a-hardware-startup-b9ca02b01453
- http://www.psfk.com/2014/08/nomiku-wifi-sous-vide-kickstarter.html
- http://jkridner.s3.amazonaws.com/esc/BEAGLE_ESC_4.ppt
- http://makerfaire.com/makers/lessons-learned-from-two-kickstarter-hardware-projects/
- https://www.kickstarter.com/projects/173231925/lono-sprinkler-controller/posts?page=2
- http://www.gadgetfactory.net/2014/10/kickstarter-2014-pt-2/
- https://theblueprint.com/topics/lessons-learned/
- http://startupjuncture.com/2014/07/24/hardware-crowdfunding-relayr/
- http://www.fastcompany.com/3004024/why-your-kickstarter-project-late
- http://www.pcworld.com/article/2013400/6-kickstarter-nightmares-and-how-to-prevent-them.html
- http://www.reddit.com/r/hwstartups/comments/2jhdtr/another_set_of_observations_from_my_personal/
- http://techcrunch.com/2014/10/11/8-things-about-hardware-crowdfunding-we-learned-from-20-campaigns/
- http://www.reddit.com/r/hwstartups/comments/2h7d4h/what_are_problems_you_face_as_a_hw_startup/

In our next class:

- Spec to Selection of parts
- Building up your system block diagram
- Finding the right parts

Lesson #2

SPEC TO SELECTION

Lesson #2 – Spec to Selection

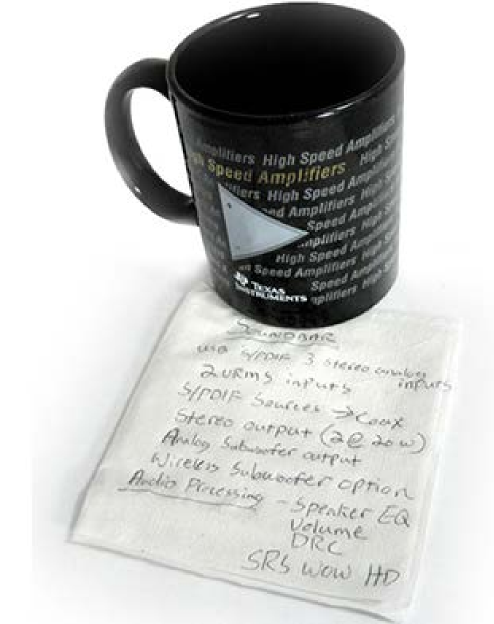
- From Specification to Selection
- Build a rough physical sketch of your product.
- Build a high level block diagram.
  - Signal flow
  - Control flow
  - Power flow.
- Device Selection
  - Analog vs. Digital signal chain
  - Microcontrollers
  - Clocks
  - Power Efficiency
- Dealing with the unexpected
- Sourcing
  - What parts to design in
  - Where to buy the parts

Getting to the spec…

- We started with a napkin →

- We built this out into 2 documents
  - MRD – Marketing Requirement Doc
  - Business Case

- After thorough self-criticism, we decided to forge ahead!

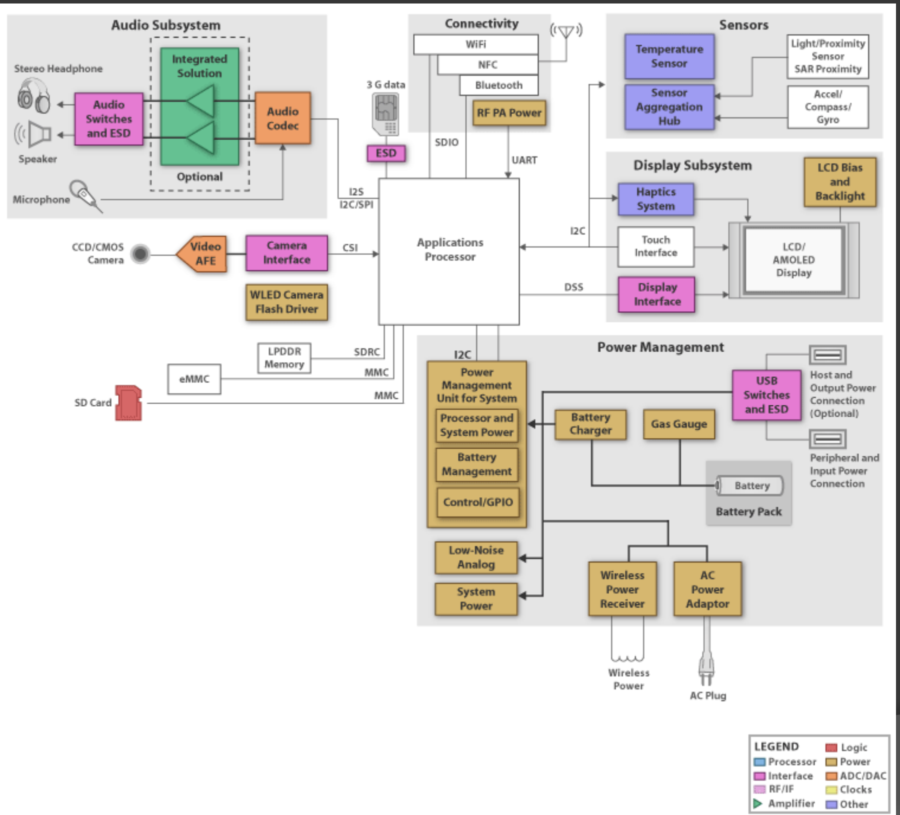


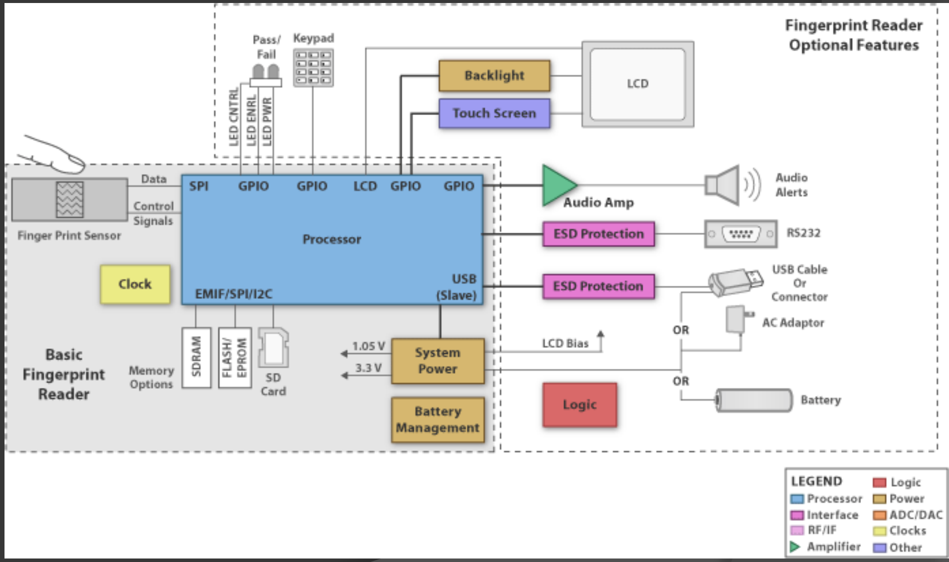
Sketches of the product

- Used to break down the system into multiple sections / pcb’s etc.
  - Connectors over here, buttons there, speakers there etc.
  - ? How to connect it all together
  - What does your power supply look like?
    - USB? Mains 110V? Universal?

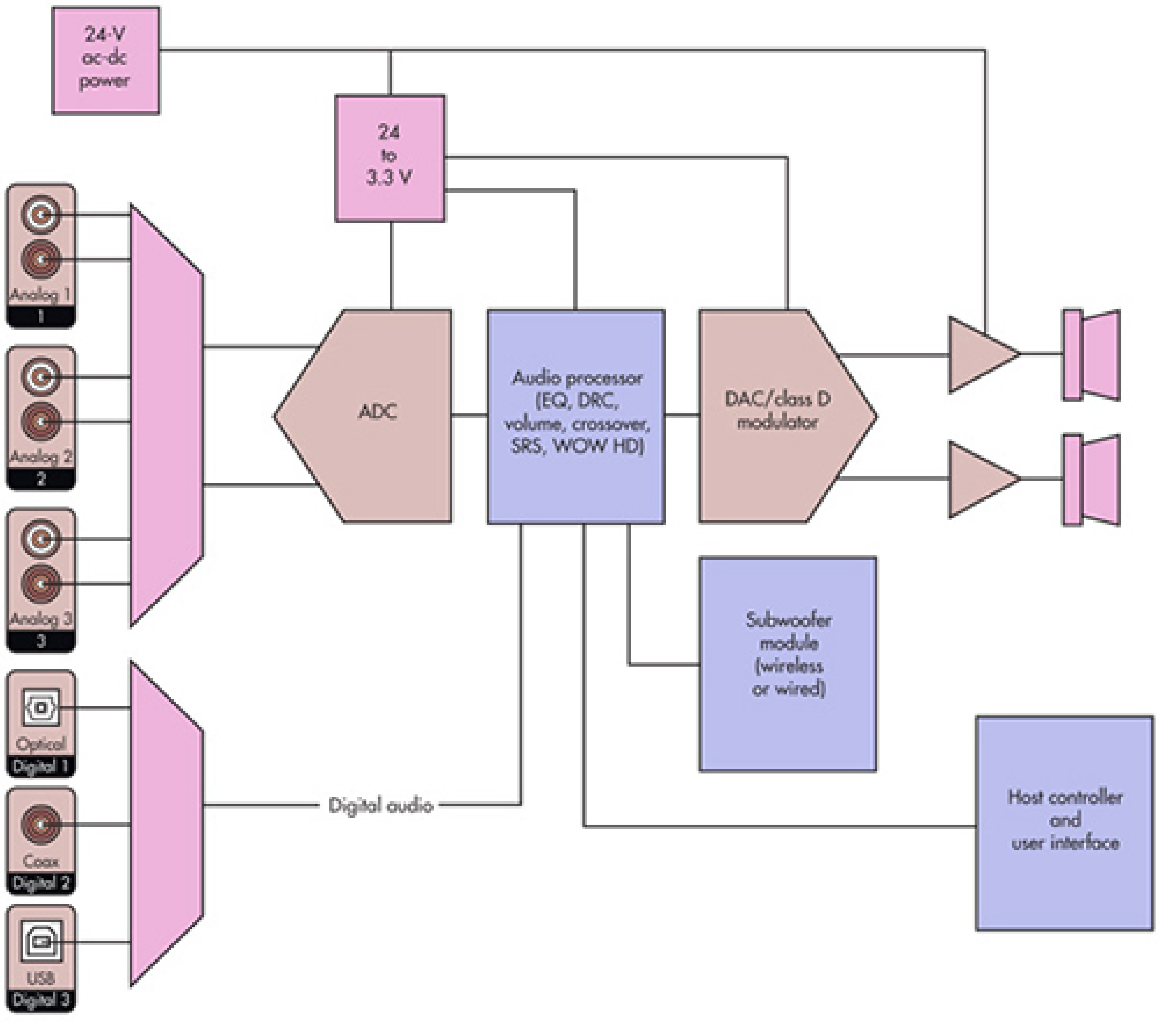
Build a high level block diagram

- Emphasis on
  - Signal Flow
  - Control Flow
  - Power Flow
- Look for suggestions, find similar block diagrams (e.g. Ti.com)

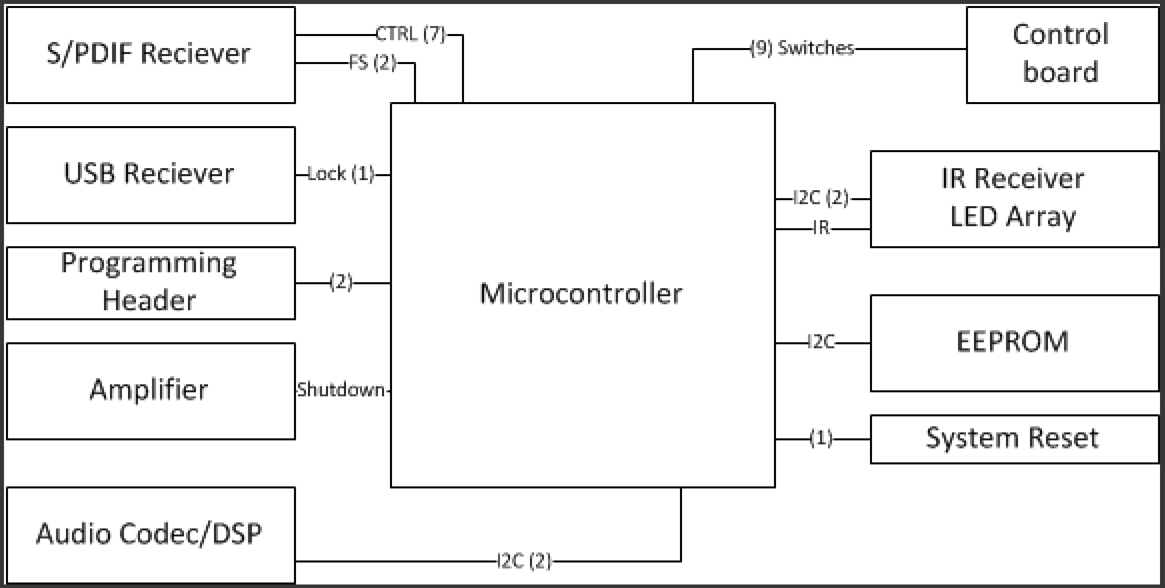




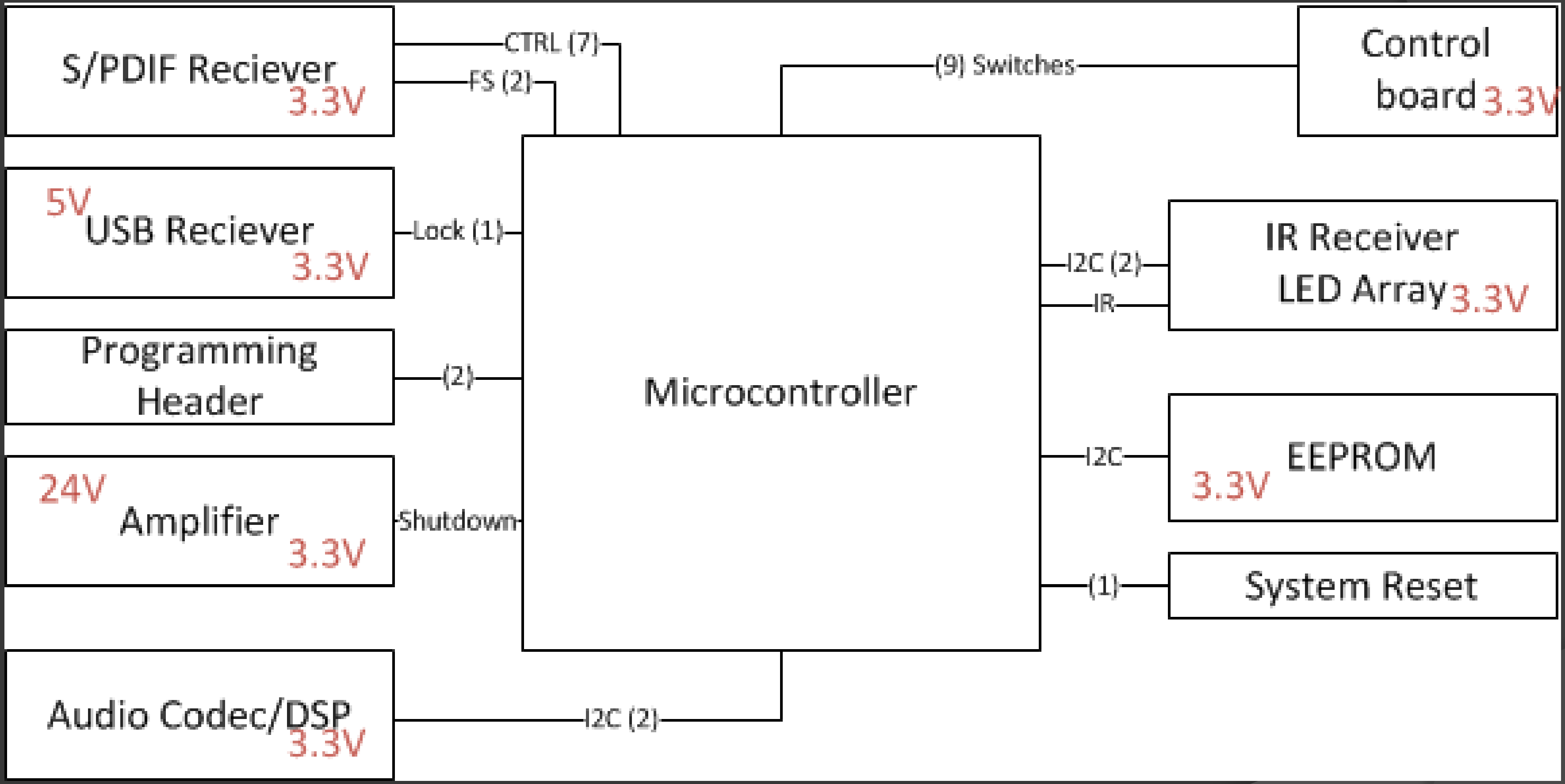
High Level Block Diagram



Example Control Block Diagram



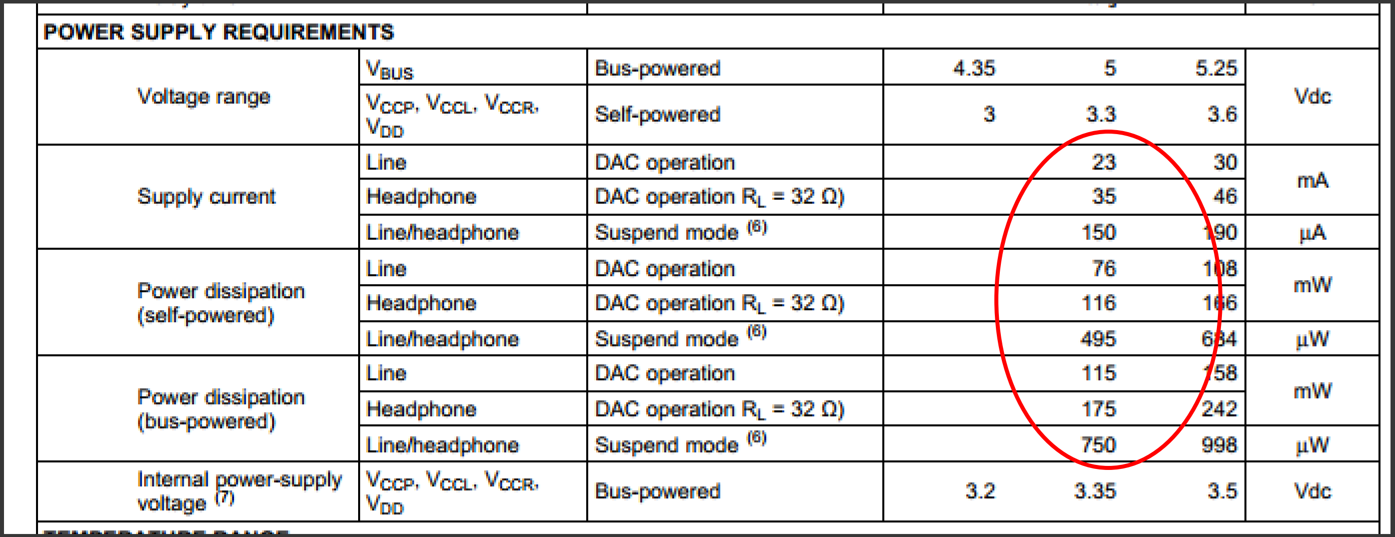
Power Supply Requirements



Next Step – start to build up current requirements for each device, and sum the current.

DON’T FORGET LED CURRENT!

Get the current consumption from datasheets

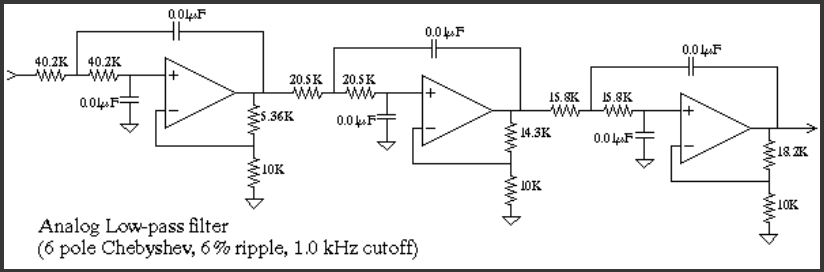


POWER SUPPLY REQUIREMENTS						
Voltage range	V_{BUS}	Bus-powered	4.35	5	5.25	Vdc
	V_{CCP}, V_{CCL}, V_{CCR}, V_{DD}	Self-powered	3	3.3	3.6	
Supply current	Line	DAC operation		23	30	mA
	Headphone	DAC operation R_L = 32 Ω)		35	46	
	Line/headphone	Suspend mode [6]		150	190	µA
Power dissipation (self-powered)	Line	DAC operation		76	108	mW
	Headphone	DAC operation R_L = 32 Ω)		116	166	
	Line/headphone	Suspend mode [6]		495	684	µW
Power dissipation (bus-powered)	Line	DAC operation		115	158	mW
	Headphone	DAC operation R_L = 32 Ω)		175	242	
	Line/headphone	Suspend mode [6]		750	998	µW
Internal power-supply voltage [7]	V_{CCP}, V_{CCL}, V_{CCR}, V_{DD}	Bus-powered	3.2	3.35	3.5	Vdc

- Where possible, assume worst case.

Analog Vs. Digital Processing

- ANALOG
  - For simple, inflexible systems
  - Maintains Signal to Noise Ratio well
  - Mostly commodity products = cost optimized.
  - May be easier to debug
  - May need higher PSU voltages



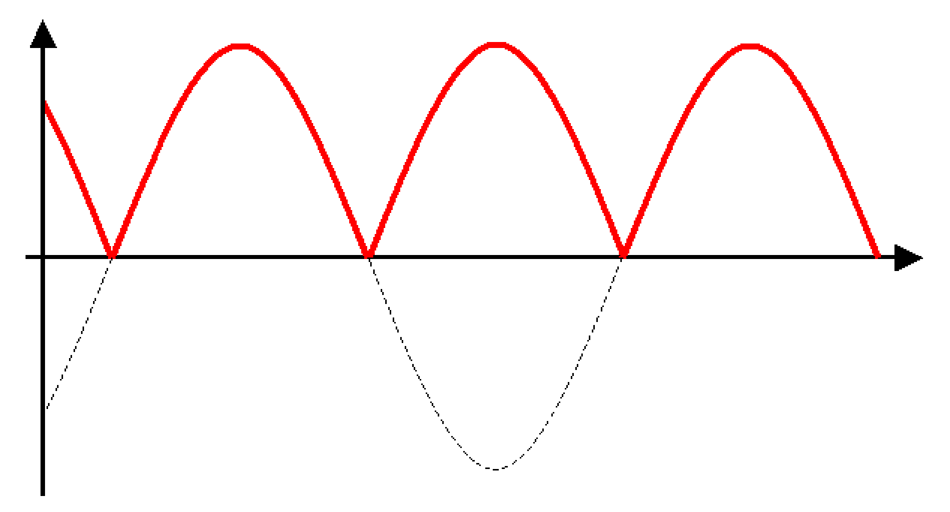
Analog Low-pass filter
(6 pole Chebyshev, 6% ripple, 1.0 kHz cutoff)

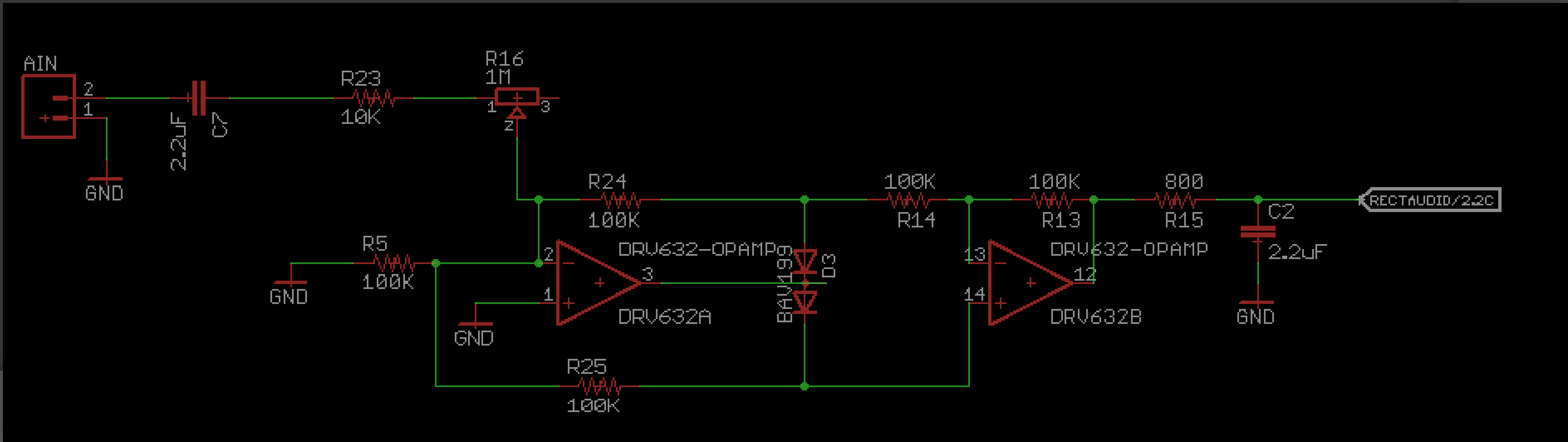
- DIGITAL
  - More complex
  - Confined to data converter resolution
  - Tends to rely on specific parts once designed in. Harder to commoditize.
  - Requires programming skills.

```
100 'LOW-PASS WINDOWED-SINC FILTER
110 'This program filters 5000 samples with a 101 point windowed-sinc
120 'filter, resulting in 4900 samples of filtered data.
130 '
140 '                  'INITIALIZE AND DEFINE THE ARRAYS USED
150 DIM X[4999]          'X[ ] holds the input signal
160 DIM Y[4999]          'Y[ ] holds the output signal
170 DIM H[100]           'H[ ] holds the filter kernel
180 '
190 PI = 3.14159265
200 FC = 0.1             'The cutoff frequency (0.1 of the sampling rate)
210 M% = 100             'The filter kernel length
```

Example where Analog is better…

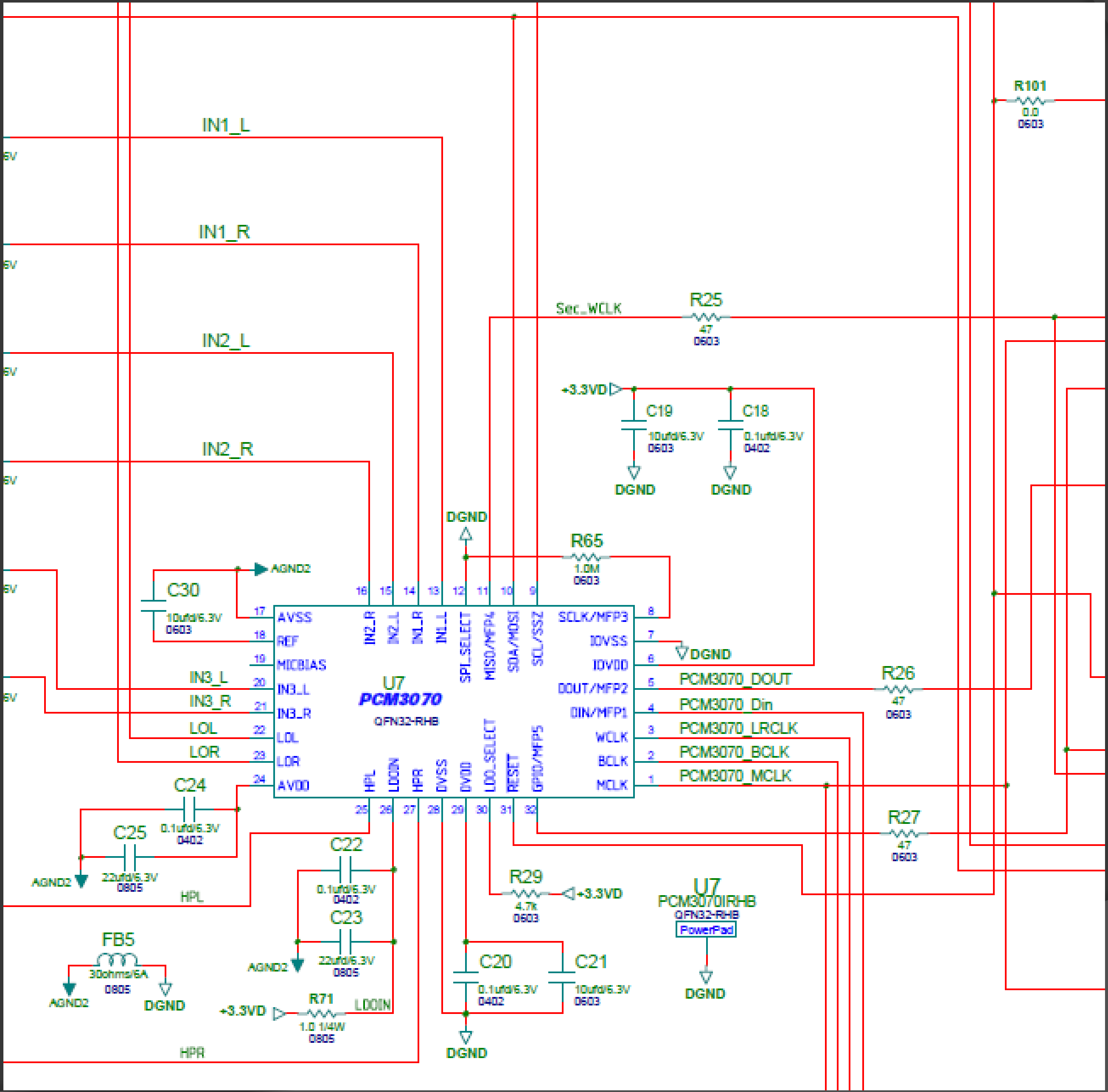
- Full Wave Rectifier circuit
- Do-able in software, but lose 6dB of SNR
- Able to use DRV632





Soundbar Audio Proc Selection

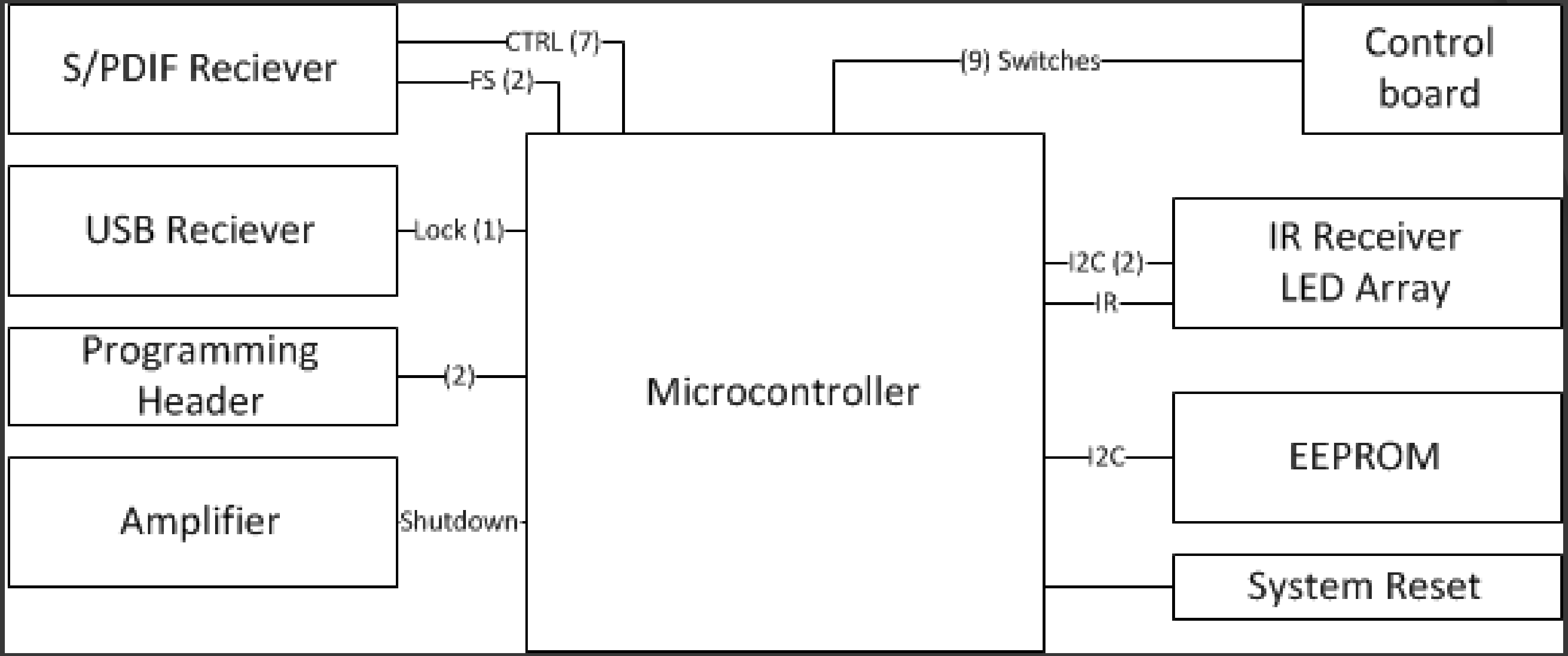
- PCM3070 Audio Codec with miniDSP
  - Allowed Analog and Digital I/O
  - Could switch Analog inputs
  - Enough processing power on chip to hand 2.0 and 2.1 systems.
  - Integrated branded algorithms (SRS WOW HD)
- Downside – need a microcontroller to boot it.


IN1_L
IN1_R
IN2_L
IN2_R
6V
6V
6V
6V
6V
6V
R101
0.0
0603
Sec_WCLK
R25
47
0603
+3.3VD
C19
10ufd/6.3V
0603
C18
0.1ufd/6.3V
0402
DGND
DGND
DGND
R65
1.0M
0603
AGND2
C30
10ufd/6.3V
0603
16
15
14
13
12
11
10
9
17
AVSS
18
REF
19
MICBIAS
20
IN3_L
21
IN3_R
22
LOL
23
LOR
24
AVDD
IN2_R
IN2_L
IN1_R
IN1_L
SPI_SELECT
MISO/MFP4
SDA/MOSI
SCL/SSZ
SCLK/MFP3
8
IOVSS
7
DGND
IOVDD
6
DOUT/MFP2
5
DIN/MFP1
4
WCLK
3
BCLK
2
MCLK
1
U7
PCM3070
QFN32-RHB
HPL
LDOIN
HPR
DVSS
DVDD
LDO_SELECT
RESET
GPIO/MFP5
25
26
27
28
29
30
31
32
IN3_L
IN3_R
LOL
LOR
PCM3070_DOUT
PCM3070_Din
PCM3070_LRCLK
PCM3070_BCLK
PCM3070_MCLK
R26
47
0603
R27
47
0603
C24
0.1ufd/6.3V
0402
C25
22ufd/6.3V
0805
AGND2
HPL
C22
0.1ufd/6.3V
0402
C23
22ufd/6.3V
0805
AGND2
R29
4.7k
0603
+3.3VD
U7
PCM3070IRHB
QFN32-RHB
PowerPad
DGND
FB5
30ohms/6A
0805
AGND2
DGND
+3.3VD
R71
1.0 1/4W
0805
LDOIN
C20
0.1ufd/6.3V
0402
C21
10ufd/6.3V
0603
DGND
HPR

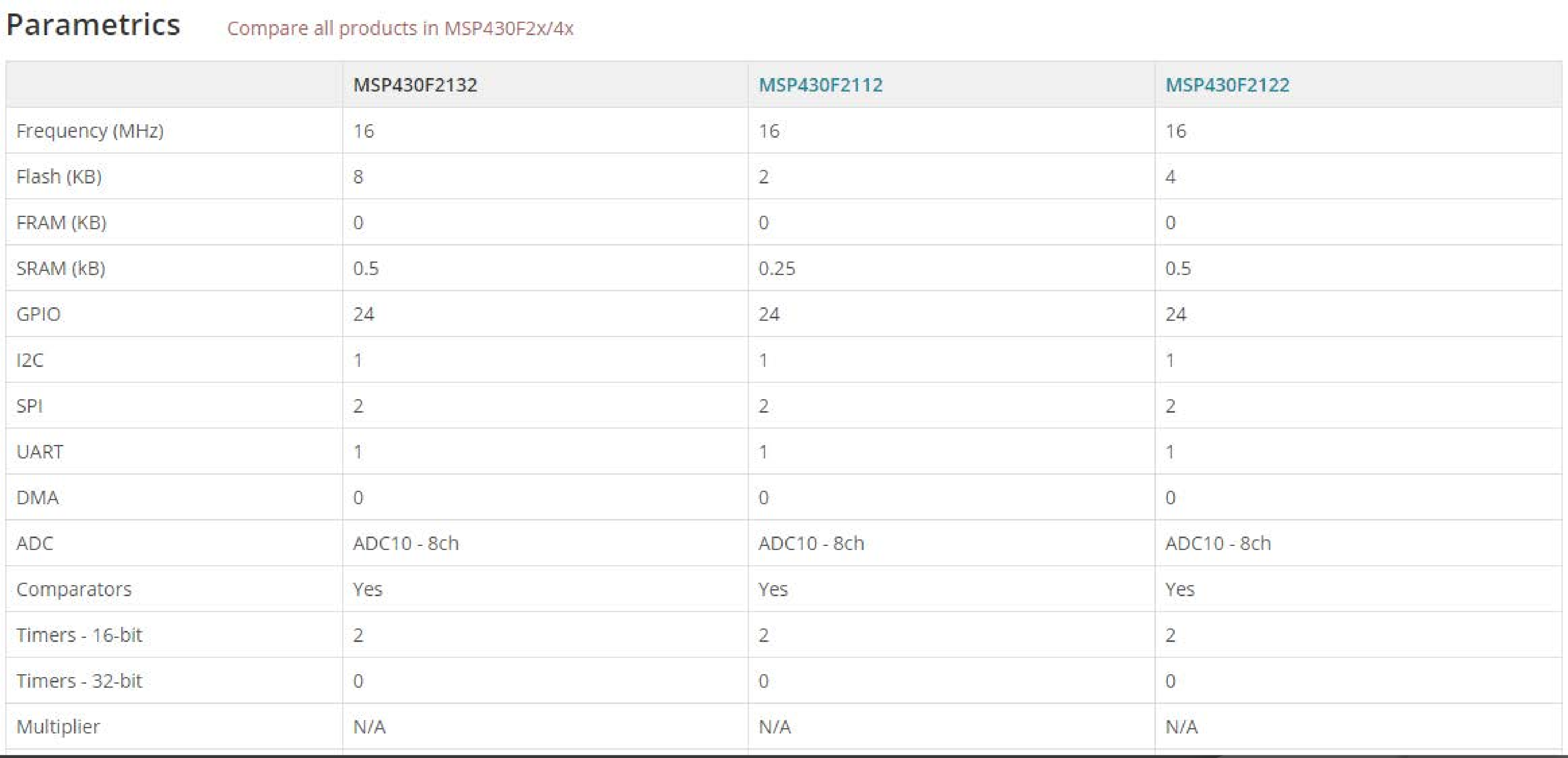
Microcontrollers

- Choose something with a good toolchain
  - Arduino/Energia awesome for getting the MVP out the door!
- Look for devices with more memory in P2P packages.
  - E.g. you think you need 1K RAM, but what if you guessed incorrectly?
- Count the number of I/O you need carefully
  - But allow yourself the capability to use an shift register if you must.

Counting I/O requirements



Looking for bigger P2P products



Parametrics

Compare all products in MSP430F2x/4x

	MSP430F2132	MSP430F2112	MSP430F2122
Frequency (MHz)	16	16	16
Flash (KB)	8	2	4
FRAM (KB)	0	0	0
SRAM (kB)	0.5	0.25	0.5
GPIO	24	24	24
I2C	1	1	1
SPI	2	2	2
UART	1	1	1
DMA	0	0	0
ADC	ADC10 - 8ch	ADC10 - 8ch	ADC10 - 8ch
Comparators	Yes	Yes	Yes
Timers - 16-bit	2	2	2
Timers - 32-bit	0	0	0
Multiplier	N/A	N/A	N/A

Clocks

- Most uC and DSP’s have an integrated oscillator
  - Some “self generate” using a DCO
  - Some require an external XTAL

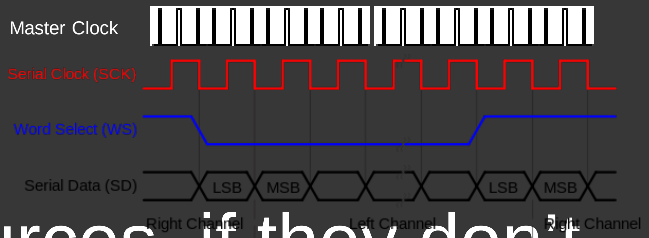
- Most converters above 10bits or so require XTAL grade oscillators for lower distortion or a recovered clock.
  - (original source = XTAL)

Clocking with Microcontrollers

- Most uC’s don’t have time critical functions. ±5% timing variation is fine.
  - E.g. running I2C or UART’s.
- USB has relatively strict requirements.
  - Most systems require a 6MHz / 12MHz XTAL.
- Better to design the PCB with an XTAL, then remove later as a cost optimization.
- Never run the audio system from a digitally generated/controlled clock.
  - uC don’t care about Jitter.

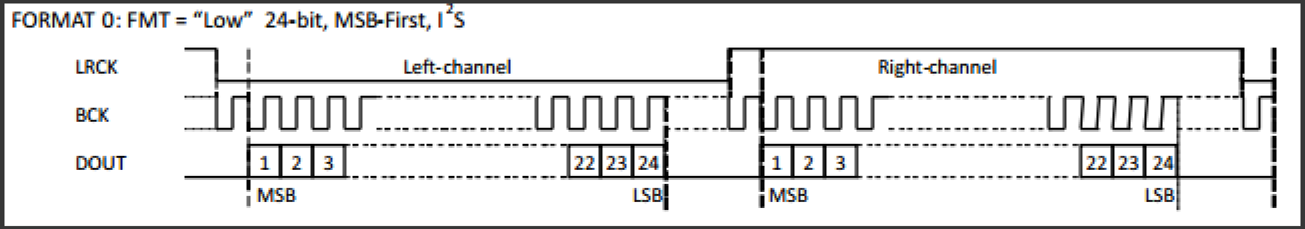
Audio Clocks

- I2S needs 4 wires



- Don’t try and mix sources, if they don’t come from the same clock source.
  - Even if they are both “48kHz”
- Audio systems care about Jitter.
- AD converters should always be master where possible.
- Get ready for some interesting clock routing systems
- Make sure you mute on clock errors.

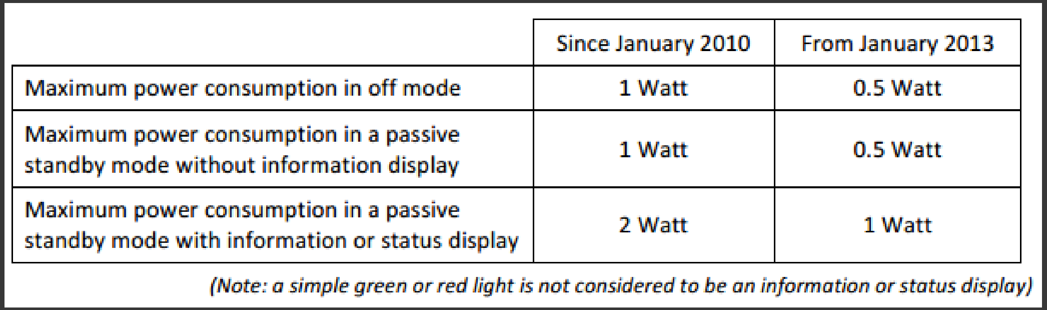
Quick note on I2S



- Stereo protocol by default.
- 2 clocks used for bit-clock and left/right clock.
- Master Clock required to drive converters
- Master/Slave relationships are important.
  - Digital Audio Receivers are typically masters
  - DSP’s and DAC’s are typically slaves
  - Try to run ADC’s as masters where possible.
- Running as master reduces jitter.

European Ecodesign Directive

- Systems must consume less than 500mW in standby



	Since January 2010	From January 2013
Maximum power consumption in off mode	1 Watt	0.5 Watt
Maximum power consumption in a passive standby mode without information display	1 Watt	0.5 Watt
Maximum power consumption in a passive standby mode with information or status display	2 Watt	1 Watt

(Note: a simple green or red light is not considered to be an information or status display)

Ecodesign
Your Future
How Ecodesign can help the environment by making products smarter

- Systems needs to intelligently standby after 15mins
  Of non usage (no signal)

- Real value in being able to wake up after auto-standby

- Your power supply (even a Usb Cube) is considered part of number.
  - El-Cheapo USB power cubes can idle with 300mW.

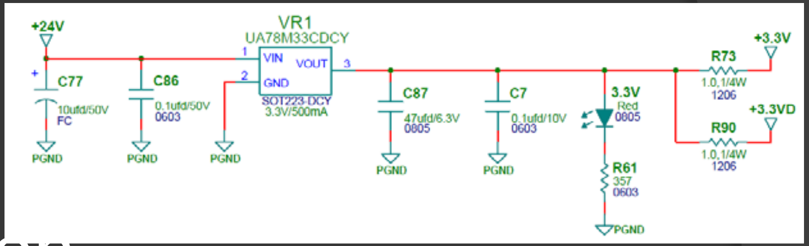
- Be Smart – Consider how you’ll auto powerup/down at the beginning of your design.

- Even if your battery powered, thought into this process will benefit battery life.

- Get out of the bad habbit of writing code that loops while waiting for inputs.
  - Put the device to “sleep” and wait for an interrupt.

Where did we go wrong with the soundbar?

- Used a linear regulator to drop 24V to 3.3V

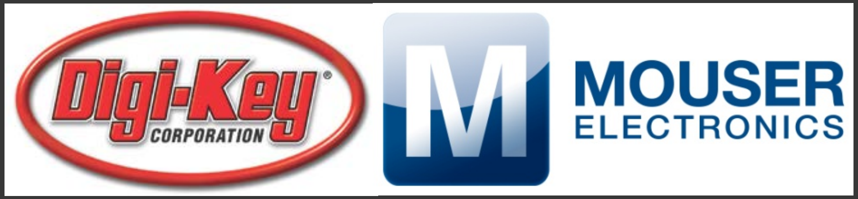


- Regulator power dissipation
  - PI = PO + PD + PQ

- Even with 1 LED on:
  - (3.3*0.01) + (21.7*0.01) + (24 * 0.005) = 0.37W
  - More in next class on that.

Sourcing

- If your going to build 100 per year, do yourself a favor.



- While they will feel “expensive” they are Easy and Reliable and have good tools.
  - E.g. BOM Managers etc.

- Don’t design in anything *too* esoteric & hard to source

Don’t be afraid to use modules

- Modules/Boards like the Raspberry Pi / Beaglebone are awesome platforms to get your MVP out the door.

- In many cases, sourcing the same parts will cost you more in BOM$ and in assembly.
  - Small volume BGA packages get $$$$

- Example:
- Analog Audio Compressor with Wifi Interface

Sourcing in China

- Find them on Alibaba and Taobao
- Get ready to install Skype (just for text chat) or QQ.
- Get ready to spend hours on chat at night
- Most don’t take paypal – you need to wire money ☹
- **Most Alibaba and Taobao agents don’t know anything about parts.**
- Some non-commodity IC’s can be the same price as digikey/mouser in sub 1K quantities.
- Search general terms “Linear Regulator”, sort by price, check the manufacturer and then check the real datasheet.
  - Commodity devices will be easier to source, ask for local manufacturer where possible. (e.g. plenty of sources for LM317 regulators)
- Prototype with known US parts, then do a version with the local manufactured parts.

Watch out for copies

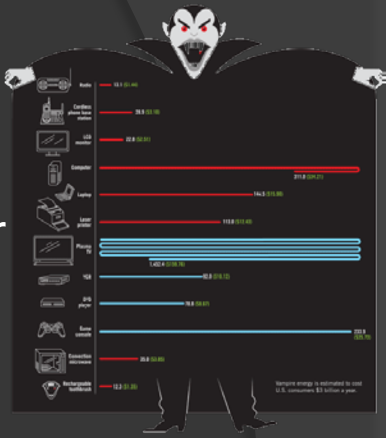
- Order and pay for the shipping on samples
- Ask for photographs of the devices, check lot codes
- If there’s a local manufacturer, take their parts, qualify them. E.g. Panjit Semiconductor

Lesson #3 – Elegance, Power & Design

- Simple market expectations.
  - Efficiency
  - Size
  - Elegance
- Power Supplies.
  - Modules are your friend
  - Avoid Mains like the plague.
- Look for failure points in your product
- System Partitioning
  - Testing
  - Upgradeability
  - Form Factor
  - Multi-Sku Design
  - Hardware Debug Capability
  - In Circuit Programming
- Lessons learnt with the Soundbar.

Market Expectations for modern product

- Get Efficient
  - Customers are asking questions about vampire power
  - How many hours of battery life?
  - European power efficiency laws / Energystar



- Industrial Design
  - “Smart” – size, shape and functionality
  - The days of sacrificing size for performance/functionality are dead. Big Hifi / TV long gone.

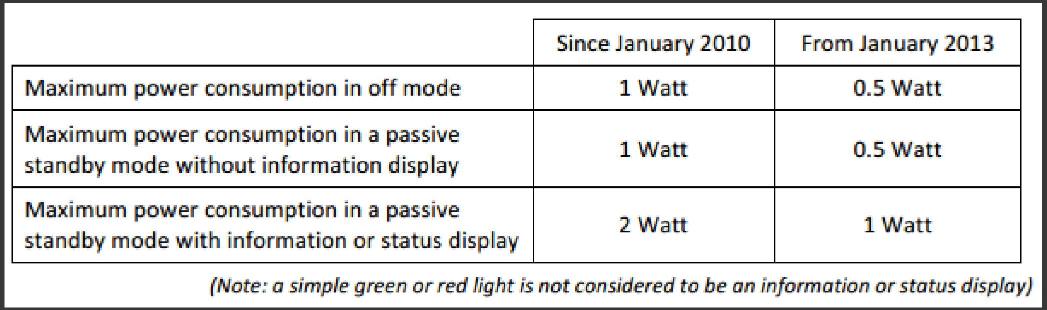
- Elegance
  - The difference between “that’s cool” and “wow!”
  - Is worth many more dollars than technical specification in retail

What’s Vampire Power?

- Power Consumption in Standby
- Can be watts of power if regulators are used, with LED’s etc.
- How can you get around it?
  - Separate standby power?
  - Backup batteries?
  - Clock Slowdown – wakeup and check every few seconds?

European Ecodesign Directive

- Systems must consume less than 500mW in standby



	Since January 2010	From January 2013
Maximum power consumption in off mode	1 Watt	0.5 Watt
Maximum power consumption in a passive standby mode without information display	1 Watt	0.5 Watt
Maximum power consumption in a passive standby mode with information or status display	2 Watt	1 Watt

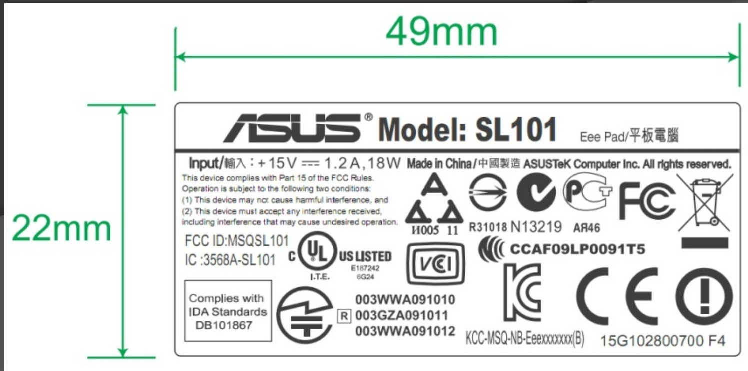
(Note: a simple green or red light is not considered to be an information or status display)

Ecodesign
Your Future
How Ecodesign can help the environment by making products smarter

- Systems needs to intelligently standby after 15mins
  Of non usage (no signal)
- Real value in being able to wake up after auto-standby
- Your power supply (even a Usb Cube) is considered part of this number.
  - El-Cheapo USB power cubes can idle with 300mW.
- Be Smart – Consider how you'll auto powerup/down at the beginning of your design.
- Even if your battery powered, thought into this process will benefit battery life.
- Get out of the bad habbit of writing code that loops while waiting for inputs.
  - Put the device to "sleep" and wait for an interrupt.

Power Supplies

- Retail = Certifications
- Accepting mains, or anything above 30V requires certification in US & Europe.
- **<u>Simple advice: </u>** where possible, use an external wall wart
- Wall wart suppliers will do the Mains to <30V for you and can be locally sourced by customers with the right power connectors etc.
- DC connectors are smaller than mains!
- USB Type-C now contains PD options.



ASUS® Model: SL101 Eee Pad/平板電腦

Input/輸入：+15V ⎓ 1.2A,18W Made in China/中國製造 ASUSTeK Computer Inc. All rights reserved.

This device complies with Part 15 of the FCC Rules.
Operation is subject to the following two conditions:
(1) This device may not cause harmful interference, and
(2) This device must accept any interference received,
including interference that may cause undesired operation.

FCC ID:MSQSL101
IC :3568A-SL101

И005 11 R31018 N13219 AЯ46

c UL US LISTED E187242 6G24 I.T.E.

CCAF09LP0091T5

Complies with
IDA Standards
DB101867

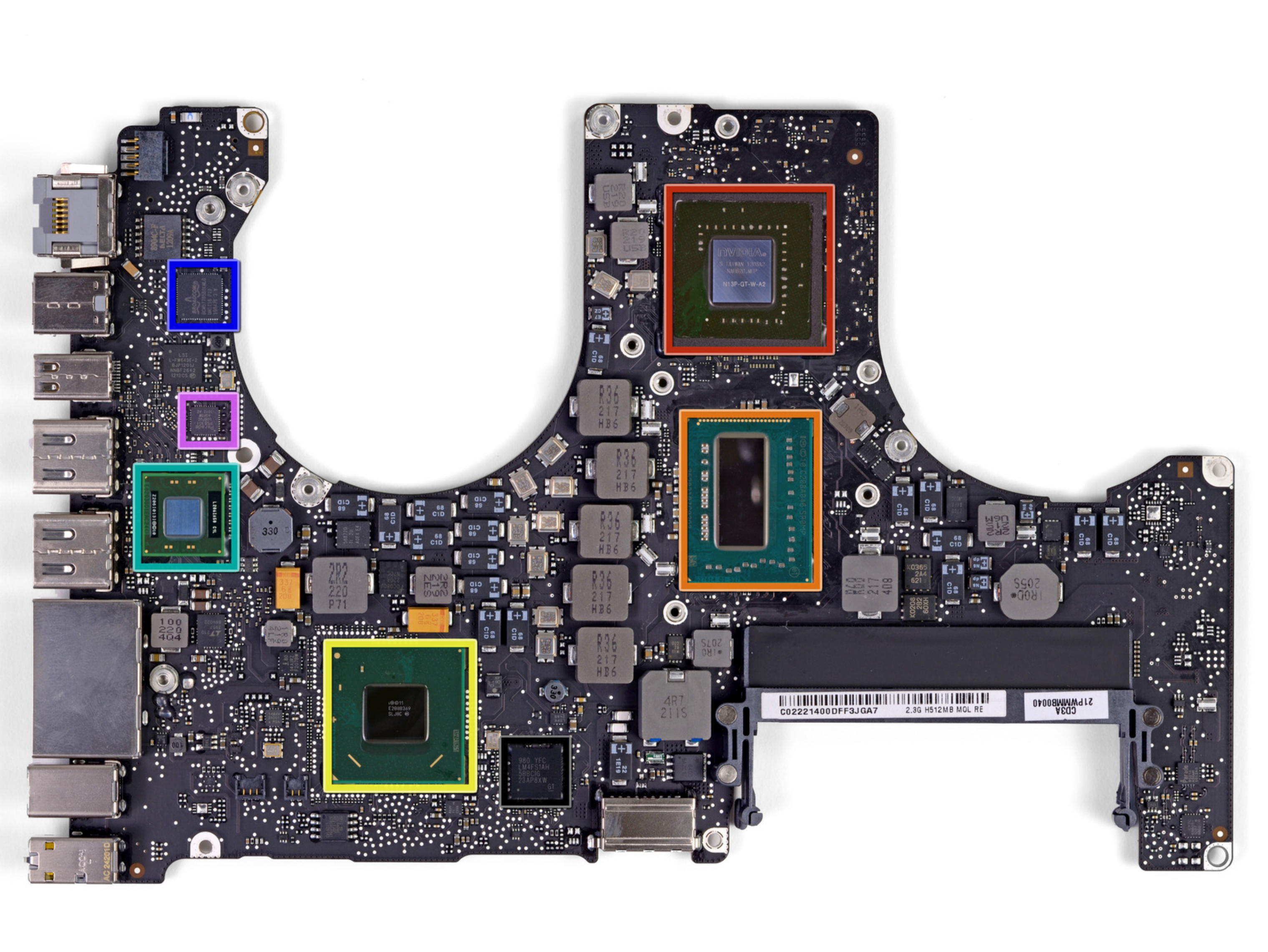
R 003WWA091010
003GZA091011
003WWA091012

KCC-MSQ-NB-Eeexxxxxxx(B)

15G102800700 F4

Industrial Design

- Get ready to be creative with your Industrial Design
- Breaking down circuit to modules allows ID to change easily.
- Get ready to stack boards.
- Component Z height becomes critical


NVIDIA
N13P-GT-W-A2
R36 217 HB6
2R2 220 P71
330
100 220 404
4R7 211S
C02221400DFF3JGA7
2.3G H512MB MOL RE
21PWMMB0040
CD3A

What is design elegance?

- Externally – Industrial Design
  - The difference between “that’s cool” and “wow!”
  - Is worth many more dollars than technical specification in retail

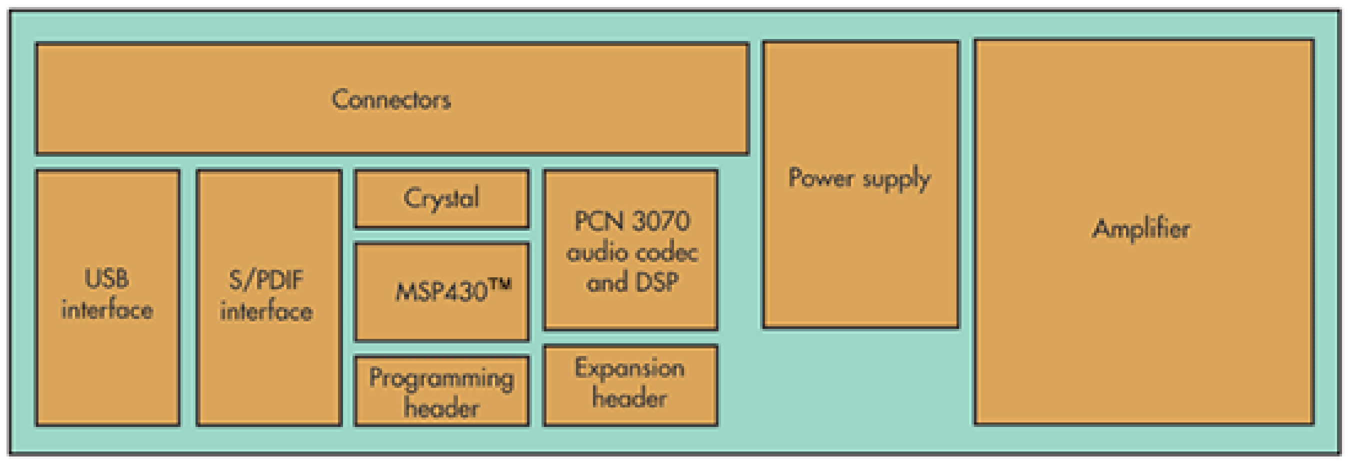
- Internally
  - A simple block diagram
  - High Performance
  - Low Power (even on mains)
  - Easy to debug
  - Easy to manufacture
  - Easy to repair

Look for failure points in your products

- What happens when one of the devices is powered down on the I2C bus?
- How susceptible is your design to ESD?
- Thermal distribution Spacing
- Overvoltage conditions on inputs
- How do you protect the design, and your customers from killing each other?

System Partitioning

- Reduce system to copy/paste sections
- Sections can be easily upgraded and changed if:
  - Define an interface that is flexible.
- Modular process allows changes in industrial design
- Easier to test each section and debug
- Upgrade & Quick Changes



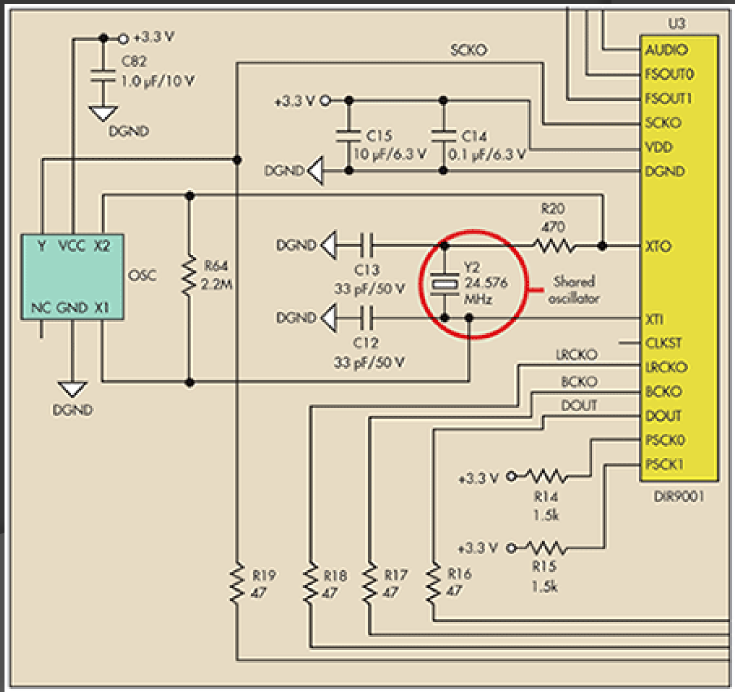
System Partitioning Ideas

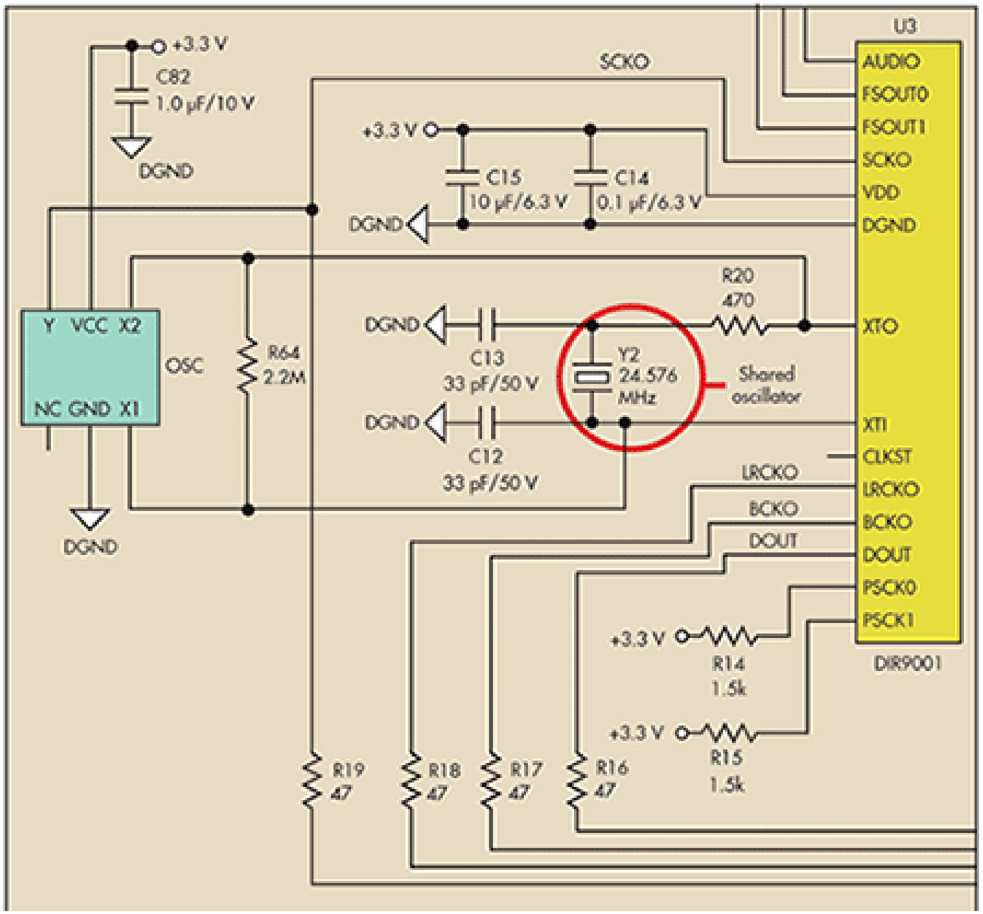
- How to test individual sections?
- Connect with 0 ohm between blocks
- Debug points (test points) all ‘round.
- I2C disconnects/isolations
- Allow microcontroller to be bypassed
- Allow Power supply to be bypassed
- Most devices have eval kits.
  - How to bluewire those kits?

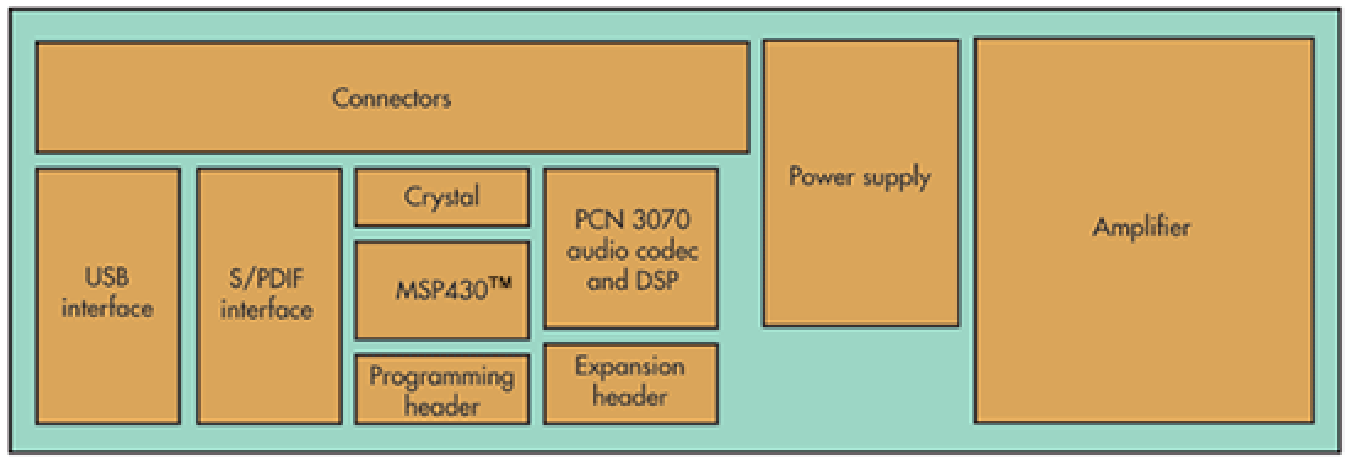
Multi-Sku Designs

- Product Spins Maximize R&D vs. Return
- Design options into the PCB
  - Pass certifications on the 1st design
- Stuffing options are your friend.
  - **ALWAYS have them in mind**

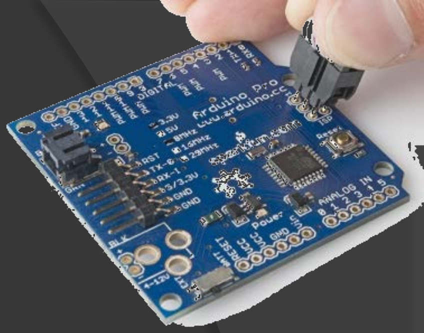
Stuffing options allow the same design and PCB layout to be used for an analog-only SKU versus a mixed analog and digital input SKU. In this case, an analog-only SKU does not require the DIR9001. By using the stuffing option, we remove the DIR9001, but reuse the footprint for the external crystal that the DIR9001 to generate the clocks for the analog SKU (along with the crystal oscillator).




U3
AUDIO
FSOUT0
FSOUT1
SCKO
VDD
DGND
XTO
XTI
CLKST
LRCKO
BCKO
DOUT
PSCK0
PSCK1
DIR9001
+3.3 V
C82
1.0 µF/10 V
DGND
SCKO
+3.3 V
C15
10 µF/6.3 V
C14
0.1 µF/6.3 V
DGND
R20
470
Y VCC X2
OSC
NC GND X1
DGND
R64
2.2M
DGND
C13
33 pF/50 V
Y2
24.576
MHz
Shared
oscillator
DGND
C12
33 pF/50 V
LRCKO
BCKO
DOUT
+3.3 V
R14
1.5k
+3.3 V
R15
1.5k
R19
47
R18
47
R17
47
R16
47


Connectors
Power supply
Amplifier
USB
interface
S/PDIF
interface
Crystal
MSP430™
Programming
header
PCN 3070
audio codec
and DSP
Expansion
header

In Circuit Programming & Factory Debug



- Plan for ICP and Factory Debug during schematic capture
- Firmware gets programmed at the end of the production flow.
  - Eeproms, Microcontrollers etc.
  - Programming over UART, USB, I2C/SPI
- Don’t add connectors for single use.
  - Pogo Pins etc are your friend.
- Factory Debug will save you $$$ in yield

Lessons Learnt

- Make sure you program the EEPROM before you program the microcontroller flash.

- Make sure that you use different I^2C addresses for the EEPROM on the programmer, versus the EEPROM on the soundbar board.