Embedded Development & AI
June 2019 Meetup

By George Franklin

Our Sponsors

Limerick AI – (Application and Games) Software Development MeetUp

http://meetup.com/LimerickAiSD

Twitter @LimerickAiSD

Big Thanks To

- For Everyone Joining & Attending
- Also to our sponsors the Bank of Ireland for the WorkBench area, at the Limerick Branch.
- Jetbrains, Redgate and Manning publications for sponsoring the Limerick Ai meetup.
- Tonight we have 2 Jetbrains licenses to raffle and a book of choice to raffle from Manning Publications.

Agenda

- motivation for a discussion on Embedded Development and to see if other members are interested in this area.

- A look at up and coming courses in the West if Ireland and on-line courses covering specialist diploma/Masters NFQ Level 9 around the subjects of AI, Embedded Development, Mechatronics and Autonomous Driving/Advanced driver-assistance systems (ADAS).

- An overview of Embedded development boards, IDE's, tooling.

- Short Break and Raffle : Sponsored Raffle with 2 Jetbrains licenses and 1 book of choice of from Manning Publications.

- A look at some development board resources, Mainly Arm Cortex based processors and Boards. Including some FPGA boards

- I have decided to leave looking at the ROS (Robot operating System) and OpenCV – for image & video manipulation for another time). So that they may get the full attention that they deserve, in a talk.

The Motivation

A look at Courses in the West of Ireland

- Msc in Artificial Intelligence at the University of Limerick
- Duration 2 Years
- Deadline for applications is July 22nd 2019
- UL website link : https://www.ul.ie/cpe/node/2901
- Brochure https://www.ul.ie/cpe/sites/default/files/MSc%20AI%20Brochure%202019%20Final.pdf

ARTIFICIAL INTELLIGENCE - MSC

Qualification:	NFQ Level 9, Major award, 90 credits
Delivery:	Part-Time, Online, Blended Learning
Duration:	2 Years
Start Date:	September
Faculty:	Science & Engineering

Continued ..

Certificate in Artificial Intelligence	Masters in Artificial Intelligence	
	Year 1	Year 2
Year 1 14 weeks (Sep-Dec) • Introduction to Scientific Computing For AI • Introduction to Deep Learning and Frameworks	SEMESTER 1 15 Weeks (Jan-May) • Artificial Intelligence and Machine Learning • Data Analytics • Advanced Topic Seminars and Project Identification SEMESTER 2 6 Weeks (May-Jun) • Research Methods and Project Specification • Risk, Ethics, Governance and Artificial Intelligence	SEMESTER 3 15 Weeks (Sep-Dec) • Machine Vision • Machine Learning Applications SEMESTER 4 15 Weeks (Jan-May) • Deep Learning • Data Science • Theory and Practice of Advanced AI Ecosystems SEMESTER 5 6 Weeks (May-Jun) • Project and Dissertation – AI

Courses

- Msc in Artificial Intelligence at the Cork Institute of Technology
- Duration 2 years
- Application Closing Date is 16th Sept 2019
- UL website link : https://www.cit.ie/course/CRKARIN9
- Brochure
- https://www.cit.ie/contentFiles/files/computerscience/Artificial%20Intelligence%20(Full-Time).pdf

Artificial Intelligence (MSc)

Course Code	CR_KARIN_9
Field of Study	Computing,Computer Science
Type of Course:	Masters Degree
Type of Qualification:	MSc
Type of Study:	Full time,Online
Application Closing Date:	16th September 2019

Semester Schedules

Stage 1 / Semester 1

Mandatory

Module Code	Module Title
COMP9061	Practical Machine Learning
COMP9016	Knowledge Representation
COMP9058	Metaheuristic Optimisation
COMP9062	Big Data Processing
COMP9011	Research Practice & Ethics

Elective

Module Code	Module Title
COMP9063	Computer Simulation & Analysis
COMP9064	AI for Sustainability
COMP9065	Recommender Systems
COMP9066	Natural Language Processing
FREE6001	Free Choice Module

Stage 1 / Semester 2

Mandatory

Module Code	Module Title
COMP9067	Deep Learning
COMP9057	Decision Analytics
COMP9068	AI Research Project

Elective

Module Code	Module Title
COMP9069	Robotics & Autonomous Systems
COMP9070	Planning & Scheduling
COMP9071	Fraud and Anomaly Detection
FREE6001	Free Choice Module

Course Modules Details link
https://courses.cit.ie/index.cfm/page/course/courseId/2002

Example Module from Above

- Msc in Artificial Intelligence at the Cork Institute of Technology
- Module Code:COMP9069 : Title Robotics & Autonomous Systems

- Link to module : https://courses.cit.ie/index.cfm/page/module/moduleId/13423

- **Module Description**:
- The application of reinforcement learning to robotics and autonomous systems has the potential to transform many industries such as manufacturing, construction and logistics. Traditional robotics design requires highly controlled more-or-less stationary environments for correct operation, the integration of reinforcement learning into robotic systems is allowing robots to overcome this constraint and thus operate in unconstrained environments. Reinforcement learning in these systems results in robots that can modify behaviour with changing environmental conditions, continuously improve operation and adapt to system failures. This module will focus on the application of reinforcement learning to both articulated systems (e.g. robotic arms and walking robots) and autonomous systems (e.g. quad-copters and rovers).

Learning Outcomes	
On successful completion of this module the learner will be able to:	
LO1	Develop and simulate models for articulated and autonomous robotic systems.
LO2	Evaluate the applicability of reinforcement learning in robotics.
LO3	Adapt reinforcement learning algorithms to robotic motion control and autonomous applications.
LO4	Appraise the application of deep reinforcement learning to robotic systems.

Courses

- **Master of Engineering in Connected and Autonomous Vehicles at the Institute of Technology Sligo**
- Duration 2.5 years
- Application Closing Date is 10th Sept 2019
- https://www.itsligo.ie/courses/meng-connected-autonomous-vehicles/

Entry Requirements

Graduates with a Level 8 Honours Degree 2:1 or above in Electronic Engineering, Mechatronic Engineering, Mechanical Engineering, Computer Science or a related discipline are eligible to apply for this programme.

Programming knowledge (Ideally C, C++ or Python) and Level 8 Engineering Maths are pre-requisites to the course.

Applicants who do not meet these criteria but have the willingness to address them will be considered. Candidate interviews and entrance exams will be used to assess suitability for the programme.

In addition, international students, whose first language is not English, will be required to prove their English competency through previous examination results, recognized English language tests such as IELTS (6.5 or equivalent required) and through oral communication skills at interview.

Semester Modules

Semester 1

Title	Credits
Applied Linear Algebra	05
ADAS and Autonomous System Architecture	05
Environment Detection	05

Semester 2

Title	Credits
Multiple View Geometry in Computer Vision	05
Automotive System Safety & Cybersecurity	05
Applied Statistics and Probability	05

Semester 3

Title	Credits
Connected Vehicles	05
Machine Learning	05
Vehicle Dynamics and Control	05

Semester 4

Title	Credits
Modelling, Simulation and Test Methods for Advanced Driver Assistance Systems	05
Multi-Modal Sensor Systems	05
Research Methods	05

Semester 5

Title	Credits
MEng CAV Research Dissertation	30

Example Module from Above

- Master of Engineering in Connected and Autonomous Vehicles at the Institute of Technology Sligo
- Module Code:ENG09025 : Title ADAS and Autonomous System Architecture
- Link to module : https://amm.itsligo.ie/en/module/web/20/ENG09025/201700/999999
- **Description** ADAS and Autonomous System Architecture provides the learner with an appreciation for the bigger picture of the automotive industry. The student will gain an understanding of the multi-disciplinary nature of the industry, as well as knowledge of its supply chain. Different system architectures and design constraints are introduced.

Indicative Syllabus

- Overview of the automotive landscape and associated vocabulary (LO1)
- The supply chain, and associated roles and responsibilities (LO1)
- Difference between advanced driver assistance systems, automated driving and autonomous vehicles (LO1)
- Introduction to Automotive System architecture - understanding the traditional car, the modern version (ADAS), the emerging and the future (LO2)
- Current driving hardware architectures (LO2)
- Current driving software architectures (LO2)
- Characteristics of Vehicle networks, including bandwidth, latency, fault tolerance (LO3)
- CAN and FlexRay Overview (LO3)
- Networking trends, including V2 , V2X (LO3)
- Understanding the data flow and data flood associated with autonomous vehicles (LO3)
- Design constraints of Automotive as an embedded system, including cost, power availability, thermal dissipation, size and weight (LO4)
- Real-time systems when applied to automotive (LO4)
- RTOS vs. OS (LO4)
- Design considerations for real-time (LO4)
- Topical issues within automotive (LO5)

Courses

- **Post Graduate Certificate in Automotive Artificial Intelligence**
- Duration 1 years
- Application Closing Date is 30th August 2019
- https://www.itsligo.ie/courses/pg-certificate-automotive-ai/
-

Semester 1

Title	Credits
Applied Linear Algebra	05
ADAS and Autonomous System Architecture	05
Machine Learning	05

Semester 2

Title	Credits
Multiple View Geometry in Computer Vision	05
Modelling, Simulation and Test Methods for Advanced Driver Assistance Systems	05
Multi-Modal Sensor Systems	05

Example Module from Above

- Post Graduate Certificate in Automotive Artificial Intelligence
- Module Code: COMP09012 : Title Machine Learning
- Link to module : https://amm.itsligo.ie/en/module/web/20/COMP09012/201700/999999
- **Description:** This module introduces the topic of machine learning algorithms (algorithms that learn from data), with the first part of the module dedicated to the standard shallow forms of machine learning before moving on to Deep Learning and Convolutional Neural Networks for use in computer vision tasks, particularly recognition, classification and localisation….

Indicative Syllabus

Machine Learning: Logistic Regression, KNN, SVM, kernel SVM, Decision Trees.

Principle Component Analysis (PCA)

Model evaluation and Hyperparameter Tuning.

Deep Learning: Perceptron, multi-layer perceptron, representation learning.

Back propagation, SGD (variants), Cost/loss functions, regularisation.

Convolutional Neural Networks.

Dropout, Batch-normalisation.

Image classification, image recognition, localisation.

Transfer Learning.

Introduction to Reinforcement Learning.

Data Sets, ImageNet, Kaggle.

Data distributions, bias, variance, divergence (KL, JS, Wasserstein Distance).

Machine Learning Frameworks such as TensorFlow, Keras, PyTorch and Caffe.

Citizen rights under data protection, data storage.

Others Courses

- University of Limerick: **Embedded Systems Engineering (Specialist Diploma) (Blended Learning) (January 2020)**
- NFQ Level : 9 - ECTS Credits : 36
- Application Deadline 20/12/2019
- Start Date 29/01/2020
- End Date 16/12/2020
- **This course starts in January 2020, hence applications will not open until around October 2019.**
- **Link : https://springboardcourses.ie/details/7255**

Online web delivered Courses

- **Of course the Udacity nanodegree(s) are well known**
- https://eu.udacity.com/course/intro-to-self-driving-cars--nd113
- https://eu.udacity.com/course/self-driving-car-engineer-nanodegree--nd013
- https://eu.udacity.com/course/robotics-software-engineer--nd209
- https://www.udacity.com/school-of-autonomous-systems#

Embedded Development

- The definition of Embedded Development may have changed over the years.
- The general principle is that we relate ED to deploying application or system code to an embedded device, perhaps by way of uploading our code into a device's flash etc. Generally referral to as firmware or device driver firmware.

- So embedded devices could be contained in consumer devices like toasters, Microwaves ,fridges, Cars, TVs etc. Most people used define embedded devices as single purpose devices, however embedded devices today can be more complex and capable with multimedia offerings.

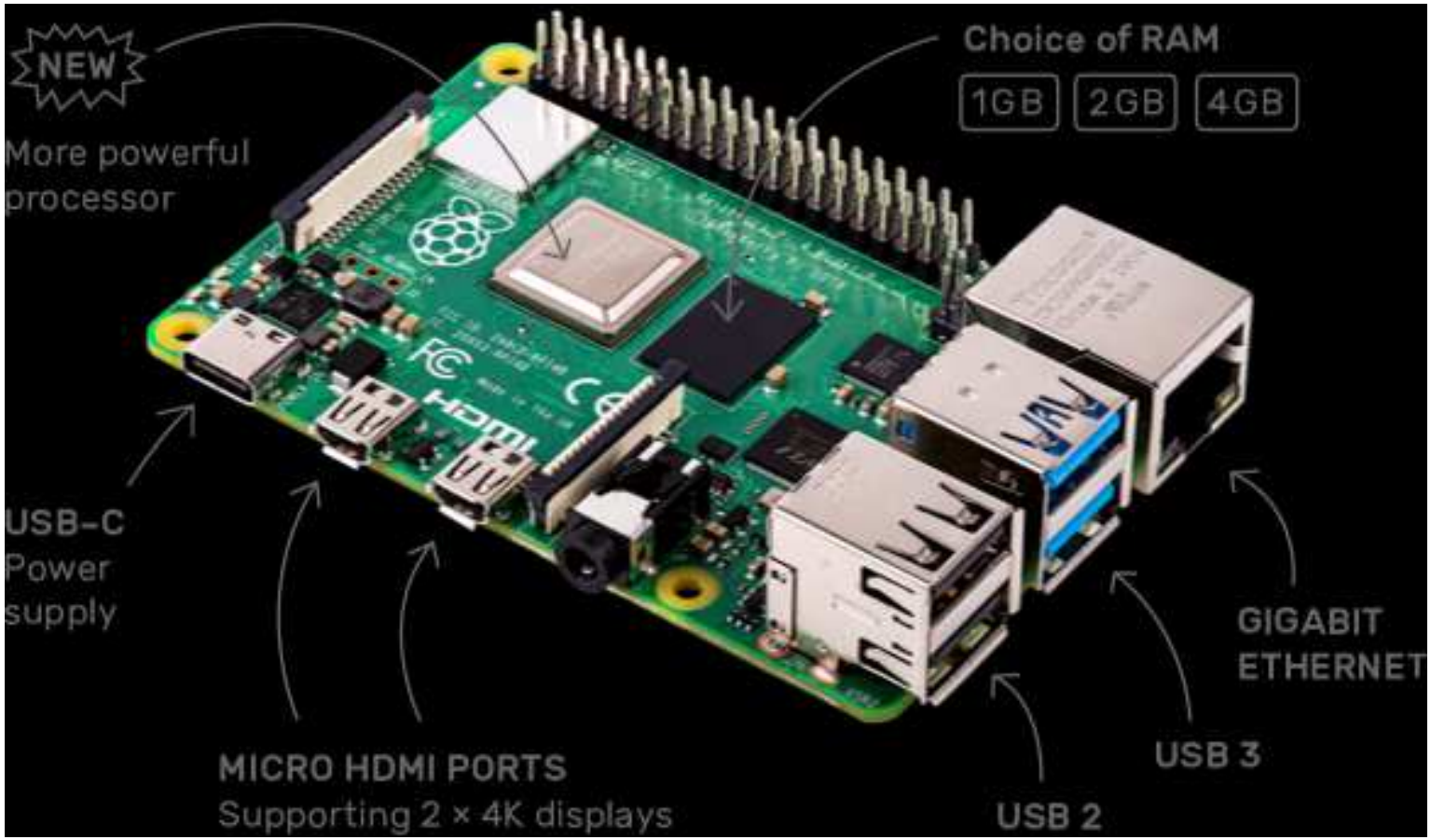
- Your still find Kiosks today that run full windows OS version like XP embedded which were binary identical to Windows XP Professional, but licensed only for embedded systems devices. The developer could also select which XP features to include in the OS deployment, so as to reduce the attack space or increase performance and reduce the OS memory overhead etc.

- For this presentation We will restrict the discussion of embedded devices to small factor computer devices that run linux OS or MCU (MicroController Units).

- Next the technology change we have seen in the consumer market. Today everyone can be an embedded developer for 10 or 20 euro!!!!!

The RaspberryPi and Aurdino influence

- The Raspberry Pi came out in 2012 and is a small board computer which runs Operating system FreeBSD, Linux , NetBSD, OpenBSD, Plan 9, RISC OS, Windows 10 ARM64 and Windows 10 IoT Core.
- Recently the Latest one (Raspberry Pi 4) was released in June 2019 ad has upto 4gb of ram, with a 1.5 quad core processor.

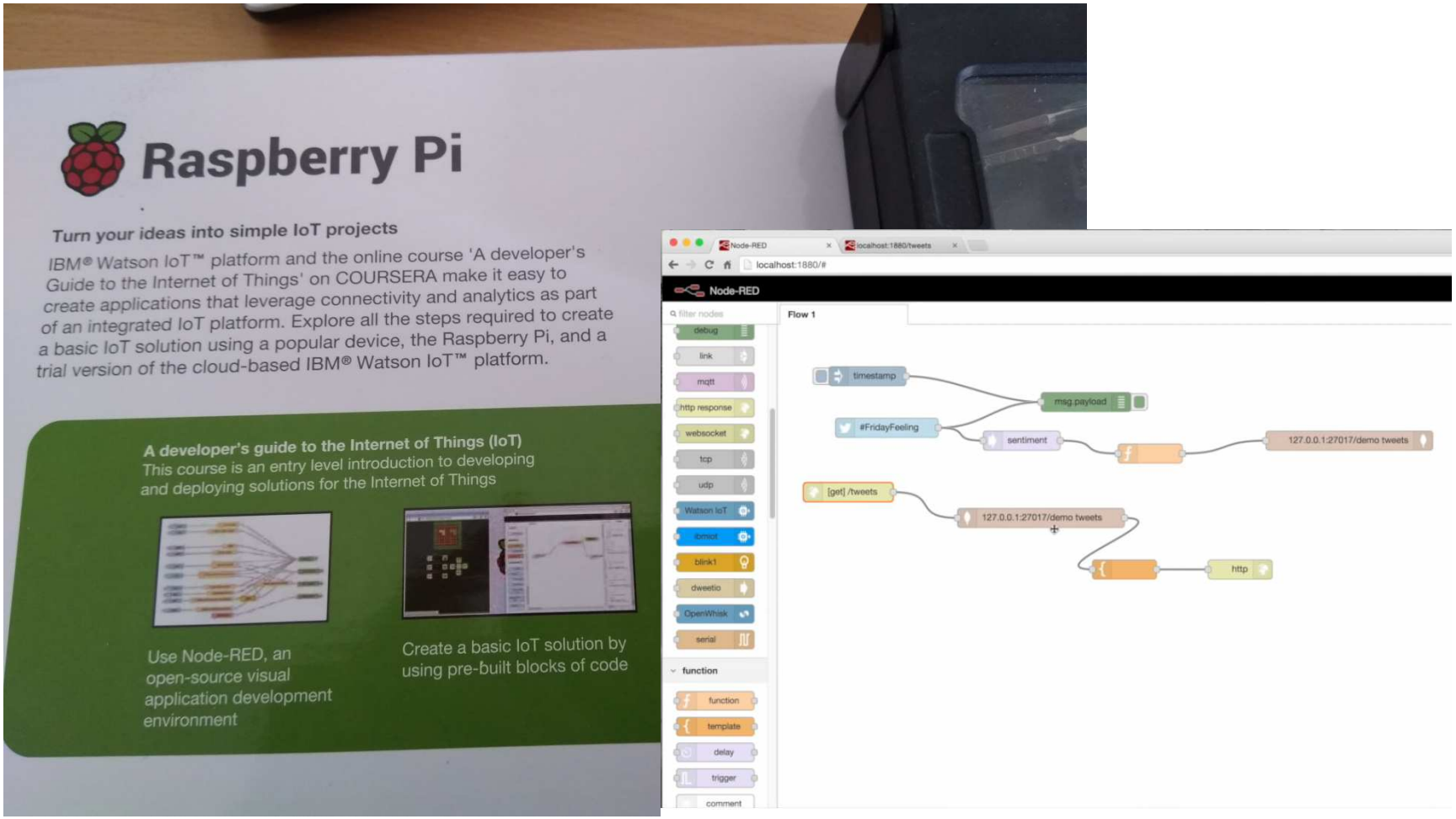


The RaspberryPi and Sense Hat

- Some older Raspberry Pi from left to right. Further right an a Raspberry pi 3b with a Sense hat shield

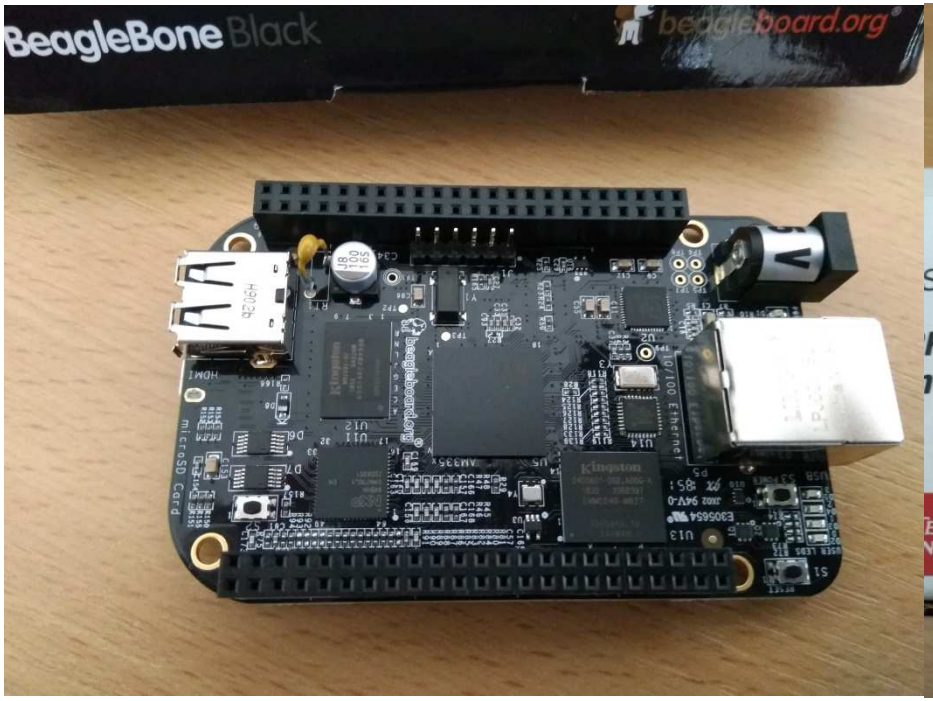
The RaspberryPi and Sense Hat

- https://nodered.org/ Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.



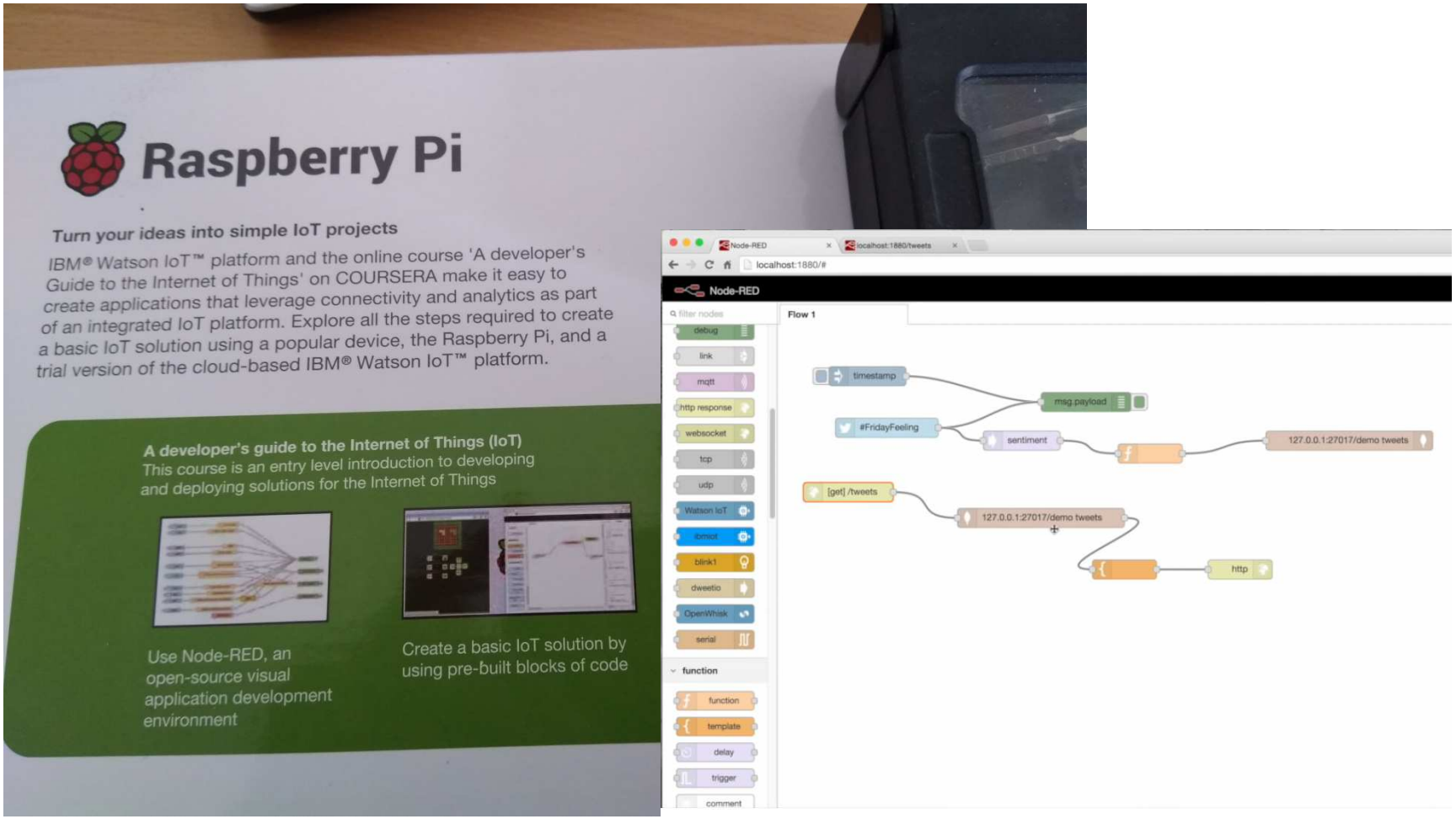
BeagleBone

- The Beagle Bone range of boards are also popular.
- Black – the most popular comes in wired Ethernet (AS SEEN below Left) or wireless form
- Blue - popular for robotics with support on board battery, wireless and lots of motor control functions
- Green - for low power applications
- White – the original one, but lacks some of the functionality seen in later boards – so less popular these days.
- Addon boards are called capes - Below right we see a BeagleBone Black with an Realtime PRU cape. This allows you to build circuits or access peripherals on the cape or prototype a circuit and have the 2 real-time PRU cores run predefined tasks that are separate from your A8 application cores running your operating system. The processor is an AM335x 1GHz ARM® Cortex-A8 with 2x Two Programmable Real-Time Units (PRUs) (https://beagleboard.org/black)



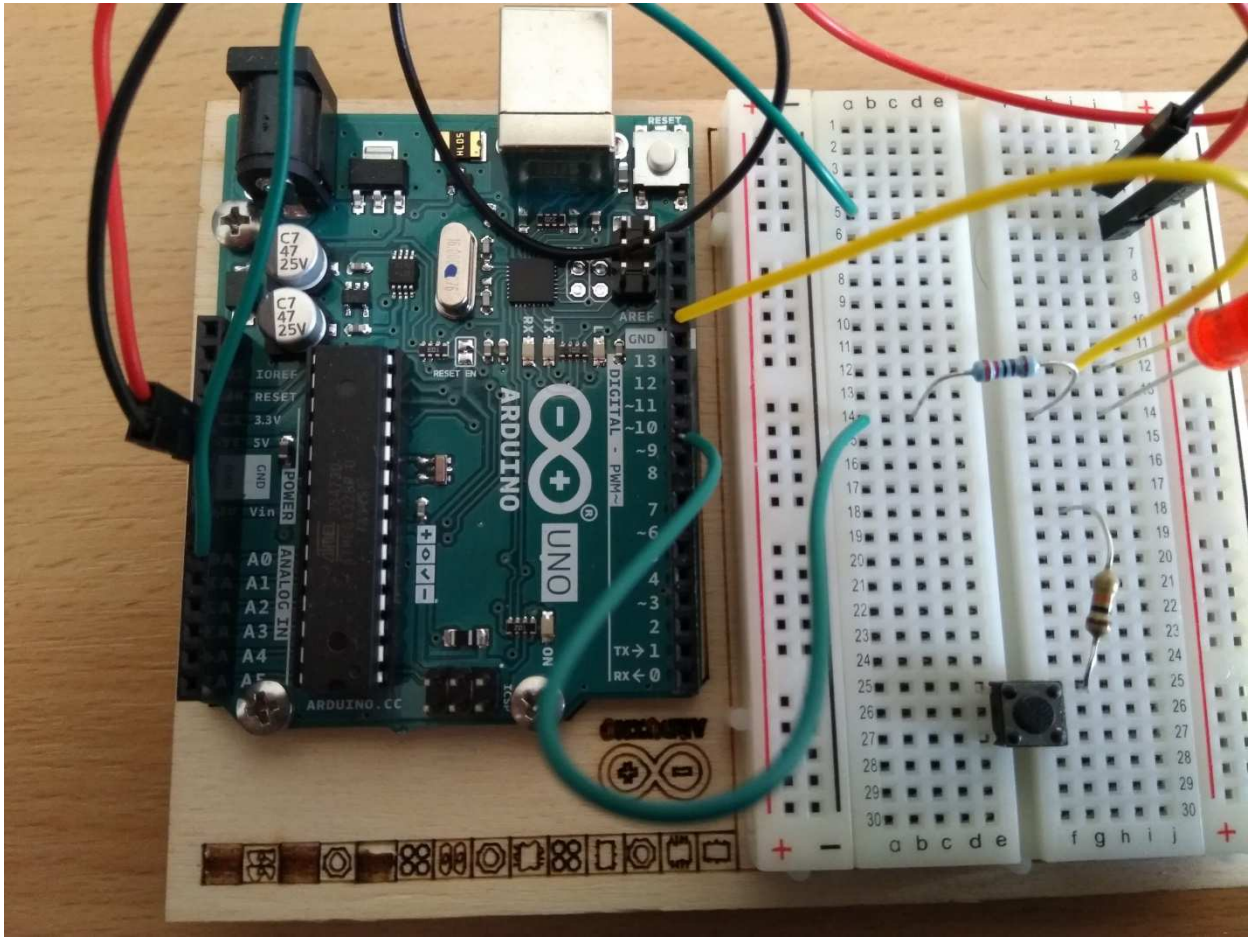
The RaspberryPi and Sense Hat

- https://nodered.org/ Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.



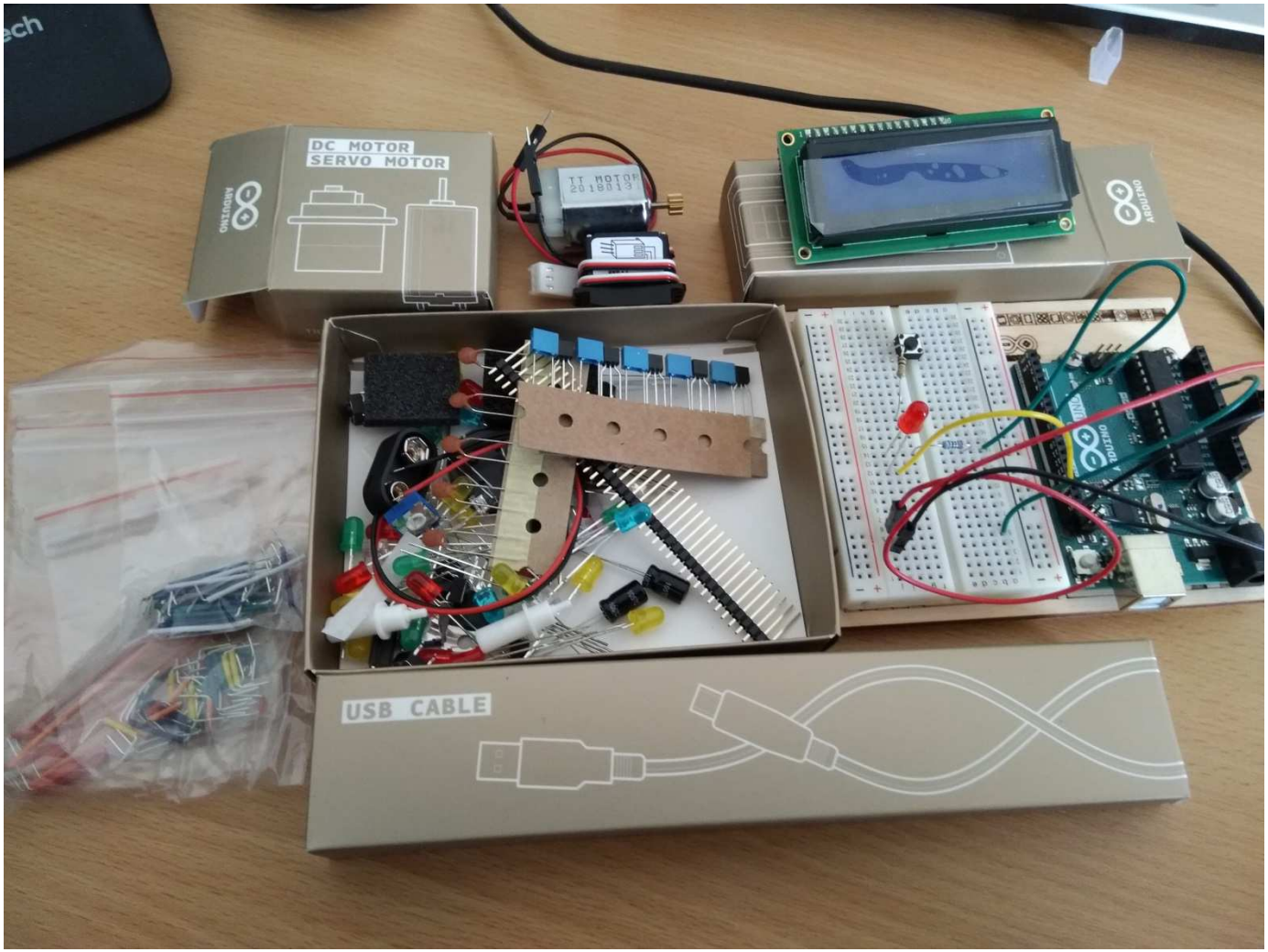
The Aurdino influence

- The Arduino UNO is the most popular and well known of the Arduino family and is a microcontroller based on the Microchip ATmega328P. The Arduino shield pin layout is now probably the most famous part of the design as shield boards can be used across lots other consumer products to add cross product support. The only thing that can be a problem is that the Arduino has 5v and other microprocessor boards may not be 5v pin tolerate. As in the case of STMicroelectronics as their I/O pins are generally 3.3v.



Arduino no Kit

- The Arduino can be also bought with some supporting parts and a book (not shown) to run through some exercises



The Aurdino Due

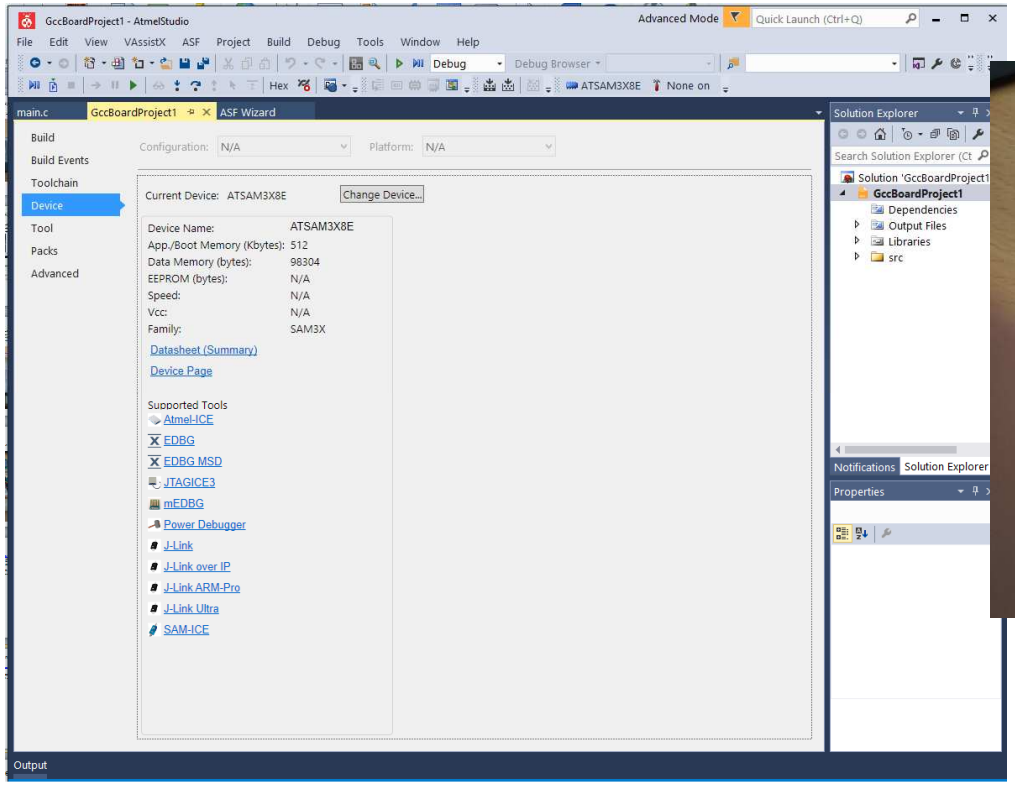
- The **Arduino Due** is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller. It has 54 digital input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), a 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button.
- **Warning: Unlike most Arduino boards, the Arduino Due board runs at 3.3V. The maximum voltage that the I/O pins can tolerate is 3.3V. Applying voltages higher than 3.3V to any I/O pin could damage the board.**
- https://store.arduino.cc/due

Development IDE fro the Arudino Uno & Due

- The Arduino(s) have supporting IDEs https://www.arduino.cc/en/Main/Software

- However a lot of developers quickly outgrow the constraints of the Arduino IDE and want a better development experience and debugging support for a number of MicroControllers and next consider using Amtel Studio

- Atmel Studio (currently at 7) is based on Visual Studio - https://www.microchip.com/mplab/avr-support/atmel-studio-7
- The Arduino Uno for example doesn't have an on-board Debugger. The Atmel Ice debugger is a external debugger of choice and is supported in Amtel Studio 7.



Debugging

- Debugging a Microcontroller for example needs either an on board debugger or an external debugger.
- ST LINK Debugger : Proper boards from e.g. STMicroElectronics all have an STLINK debugger.
- JTAG : JTAG is used by a lot of different boards. https://en.wikipedia.org/wiki/JTAG
- Segger JLINK https://www.segger.com/ is a well known producer of external debugging equipment
- Segger also have a non – commercial firmware upgrade if you wish to change a STLINK on board debugger to a JLINK (details on next page) https://www.segger.com/products/debug-probes/j-link/models/other-j-links/st-link-on-board/
- Debugging define 2 methods soft and hard. Soft means debug points are put in the software as instructions and Hard means the on-board or external debugging intercepts the board's operations using comparators, but you are limited to a number of Hard debug points. The number depends on the debugger – normally in the range of 4 to 8.
- Below some ST Link external debuggers – but now all STMicroElectronics ***evaluation*** boards (Discovery & Nucleo) have on board STLink debuggers

Converting an ST-LINK to an On-Board J-Link

Converting ST-LINK On-Board Into a J-Link

- Fully compatible with and same features as J-Link OB [More..]
- Wide range of IDEs are supported
- Virtual COM port (VCOM) support

SEGGER offers a firmware upgrading the ST-LINK on-board on the Nucleo and Discovery Boards.

This firmware makes the ST-LINK on-board compatible with J-Link OB, allowing users to take advantage of most J-Link features like the ultra fast flash download and debugging speed or the free-to-use GDBServer.

Getting Started with ST-LINK On-Board

In order to get started with ST-LINK on-board and upgrade it to a J-Link OB, just a few steps are necessary:

- Make sure that the ST-LINK USB drivers are installed: Download
- Make sure that the J-Link software package V5.12b or later is installed: Download
- Download the SEGGER STLinkReflash utility: Download
- Start the STLinkReflash utility
- Agree to the license terms
- Connect ST-LINK on-board to PC
- Select "Upgrade to J-Link"
- Wait for operation to complete

Invasion of the Arm processors !!

- Straight from the Wiki !!!!!
- ARM, previously Advanced RISC Machine, originally Acorn RISC Machine, is a family of reduced instruction set computing (RISC) architectures for computer processors, configured for various environments. Arm Holdings develops the architecture and licenses it to other companies, who design their own products that implement one of those architectures—including systems-on-chips (SoC) and systems-on-modules (SoM) that incorporate memory, interfaces, radios, etc.
- It also designs cores that implement this instruction set and licenses these designs to a number of companies that incorporate those core designs into their own products.

The Arm Cortex processor Range

- The **A range** is focused to towards application performance requirements. So we may see higher clock speed and increased processor cache sizes.

- The **R Range** is focused towards Real-Time requirements. This means the design of the Cache is **deterministic by design** and the design is concerned with consistory of operation and provides support for ECC (Error checking Memory)
  - The R range Features includes
  - Non overlapping memory regions
  - Tightly coupled memory
  - Increased exception handling in hardware
  - Hardware division instructions
  - Memory protection unit (MPU)
  - Deterministic interrupt handling as well as fast non-maskable interrupts
  - ECC on L1 cache and buses
  - Dual-core lockstep for CPU fault tolerance

- The **M Range** is used for general purpose small to medium performance purposed devices that range from 5 euro disposable devices to multimedia capable devices. The licensed M range is used in Microcontroller designs and so will match the need for a weather a floating Point unit is required, clock speed. The most common models we in Microcontrollers are the M7,M4,M3,M1,M0+, M0 (with M7 being the higher performing end of this list).

Application/Media Focused Range

- ARM have licensed Cortex Ax level processors to companies who have designed them to run the everyday devices we commonly i.e. Mobile phones, Laptops, Notebooks, device tablets such as the ipad, Android based. (common A models we see in consumer devices are the A8,A9,A15). Below of some the models from the latest high end of the Cortex-A range.

Cortex-A77

Third generation premium CPU based on DynamIQ technology.

- Leadership performance and efficiency for 5G mobile solutions
- Improved responsiveness for on device machine learning
- Built for next generation smartphones and laptops

Cortex-A76

Provides laptop-class performance with smartphone efficiency

- Designed for devices undertaking complex compute tasks
- Greater single threaded performance and improved energy efficiency
- Enables faster responsiveness and at-the-edge support for machine learning applications

Cortex-A76AE

The world's first autonomous-class processor with integrated safety

- Purpose-built for functional safety applications such as ADAS and autonomous vehicles
- First application processor with Split-Lock capability
- Safety capable to industry standards. ISO26262 ASIL D systematic

Real-Time Focused Range

- ARM have purposed the Cortex R range for safety critical requirements. The range includes R4, R5,R7, R8 and R52. (with R52 being the highest performance of the range). The R4 & R5 are commonly seen in affordable developer licensed boards.

Cortex-R5

Offers seamless, real-time performance with functional safety

- Dual core configurations enables twice the performance of previous Cortex-R processors
- Ideally suited for the development of safety-critical products
- Most commonly used in high-volume deeply embedded SoC applications

Cortex-R4

Smallest, real-time performance processor

- Offers excellent energy efficiency and cost effectiveness
- Prioritize reliability and error management with built-in error handling
- Ideal for embedded applications including automobiles and cameras

MicroController Focused Range

- ARM have purposed the Cortex M range for for MicroController embedded devices. The M is range is very large, the most common developers will see via developer evaluation boards are the M7, M4, M3, M0+, M0. The **Cortex-M1 is** Optimised for FPGA designs.

Cortex-M7

Highest performance Cortex-M processor

- Simplifies signal processing to bring high-performance DSP to the masses
- Built in floating point processing reduces power consumption
- Supports innovative MCUs for more intensive automation tasks

Cortex-M4

Control and performance for mixed signal devices

- Integrated digital signal processing (DSP) simplifies system design
- Essential microcontroller features make it ideal for industrial applications
- Most widely deployed Cortex-M processor with broad ecosystem

Cortex-M3

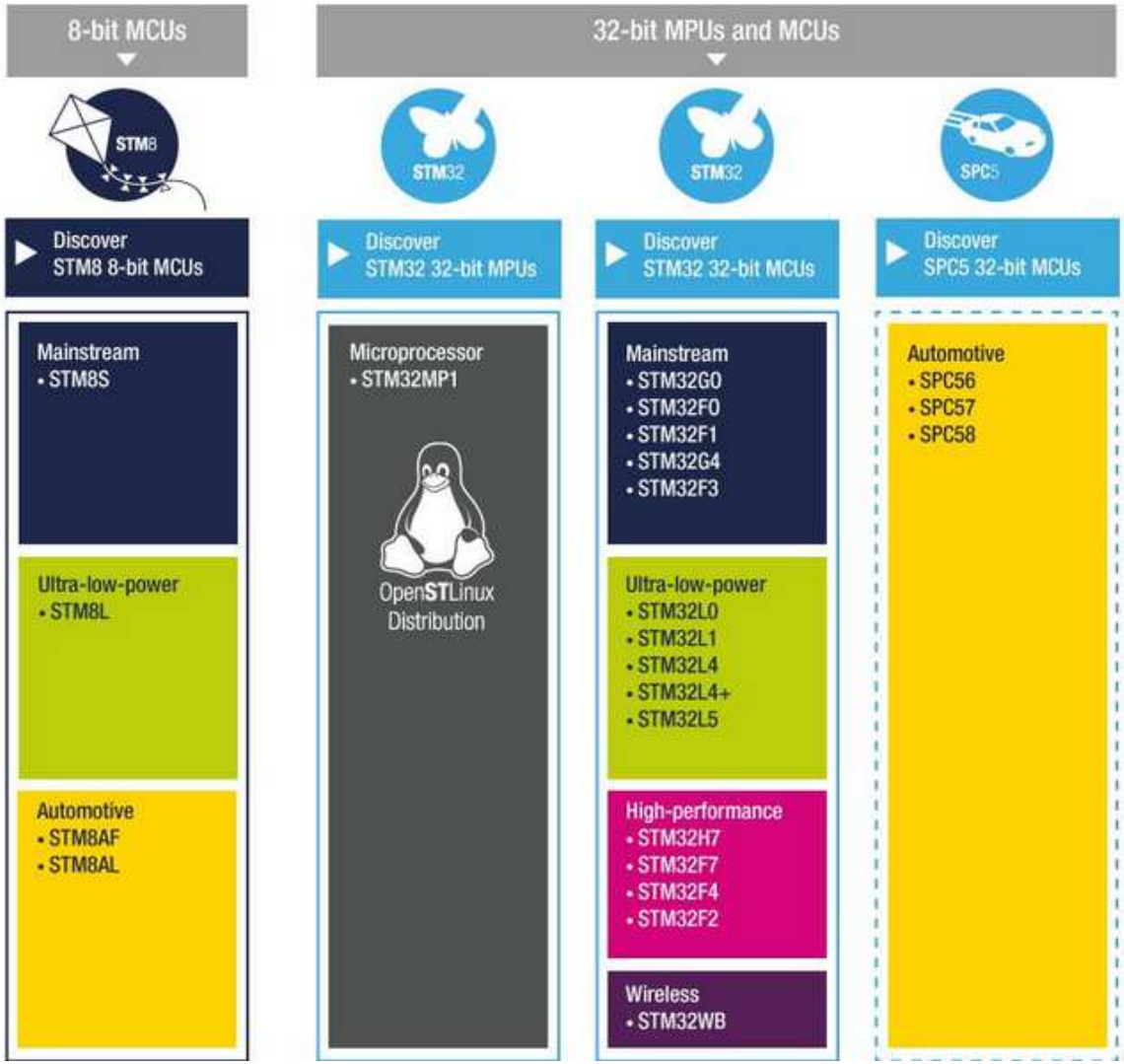
Exceptional 32-bit performance with low power consumption

- Available for a no license fee through DesignStart
- Commonly used across many smart home devices
- Deployed in billions of devices across a broad set of embedded applications

Example Arm Cortex-M based MicroController Board

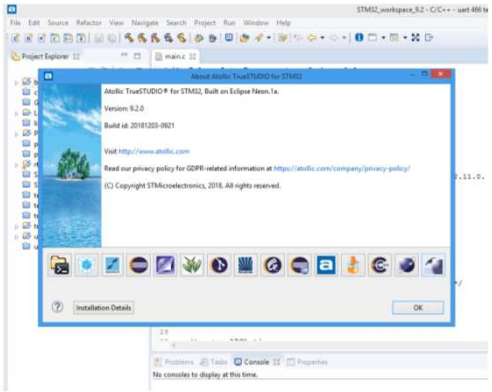
- STmicroelectronics STM32L476RGT6 64 pins, Arm Cortex M4 at 80mhz - 1 Mbyte of ram , 128KB of flash.

The STmicroelectronics range of Microprocessors

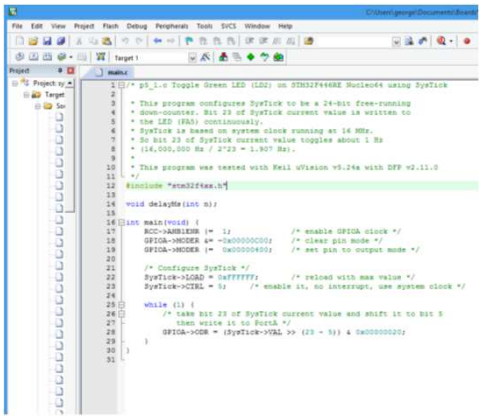


Common Arm Development IDEs

- Atollic TrueSTUDIO (free to use) https://atollic.com/truestudio/
- TrueStudio is eclipse based C/C++ IDE for STM32 projects

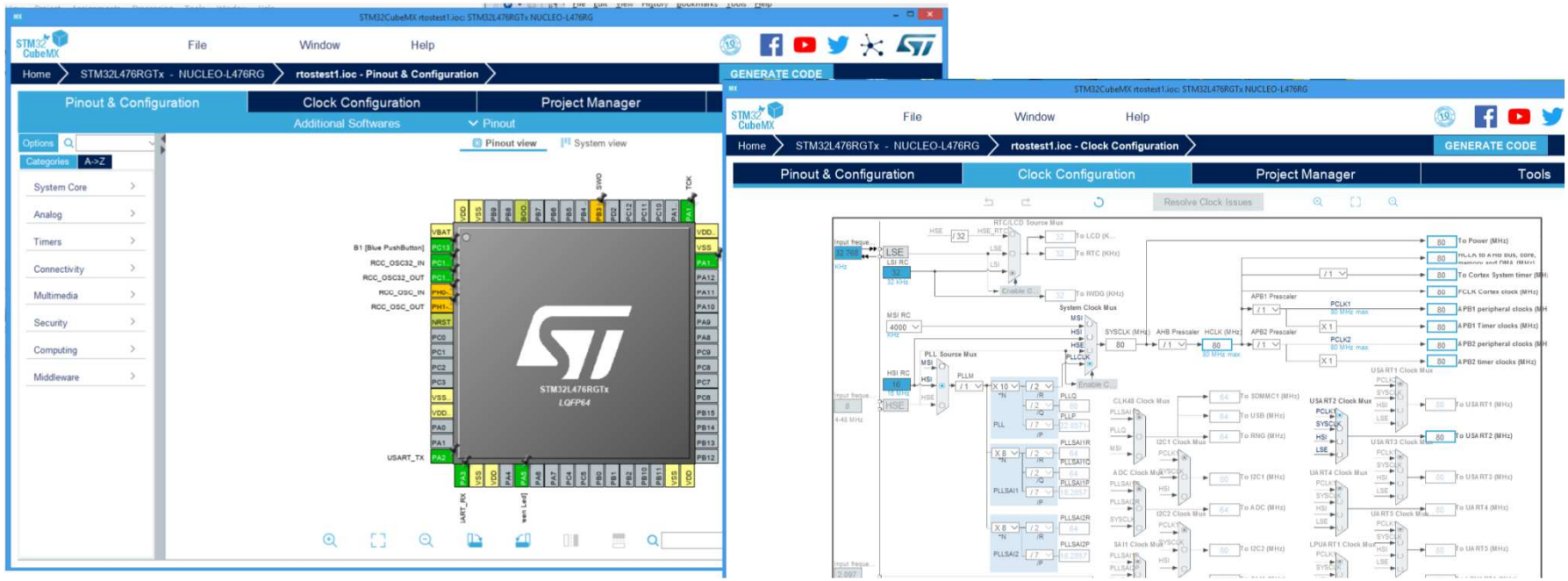


- Keil (code limited to 32KB – but can used with unrestricted size limits with an Cortex-m0, with the license on their website). http://www.keil.com/
- Pros - Keil has great debugging support
- Cons - Keil has a lack of editor formatting/refactoring support and requires the support of external programs to resolve these problems.



ST32CubeMx

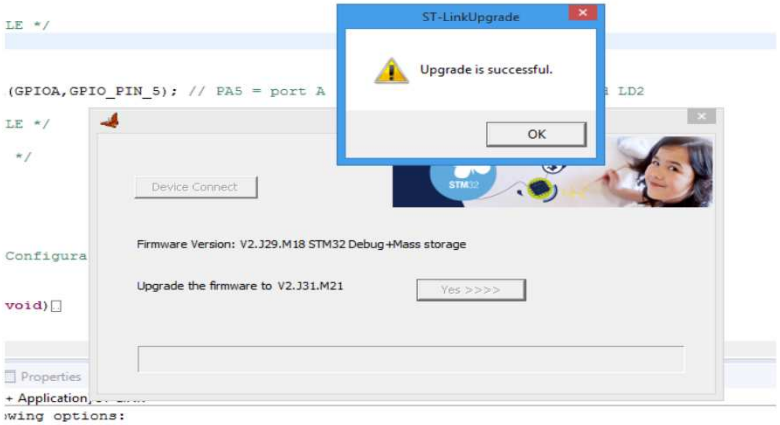
- ST32CubeMx is a supporting tool from STMicroElectronics is a board support package (BSP) configuration tool to let you include/exclude board hardware support in your application.
- It allows you to configure and initialize the hardware by a HAL layer (e.g. STM32Cube) that sits on top of a CMSIS (**Cortex Microcontroller Software Interface Standard)** layer.
- The website describes as : STM32CubeMX is a graphical tool that allows a very easy configuration of STM32 microcontrollers and microprocessors, as well as the generation of the corresponding initialization C code for the Arm® Cortex®-M core or a partial Linux® Device Tree for Arm® Cortex®-A core), through a step-by-step process.



Upgrading STLink

- The first time you try to debug and upload your elf file to your STmicroelectronics board you may have a pop windows saying that the STLINK firmware on your attached board need upgrading. If using True Studio this will prompt as below to upgrade the firmware. You click on Device Connect to choose your device and press yes to upgrade.
- There are also older stand alone tools to upgrade your STLINk board https://www.st.com/en/development-tools/stsw-link004.html

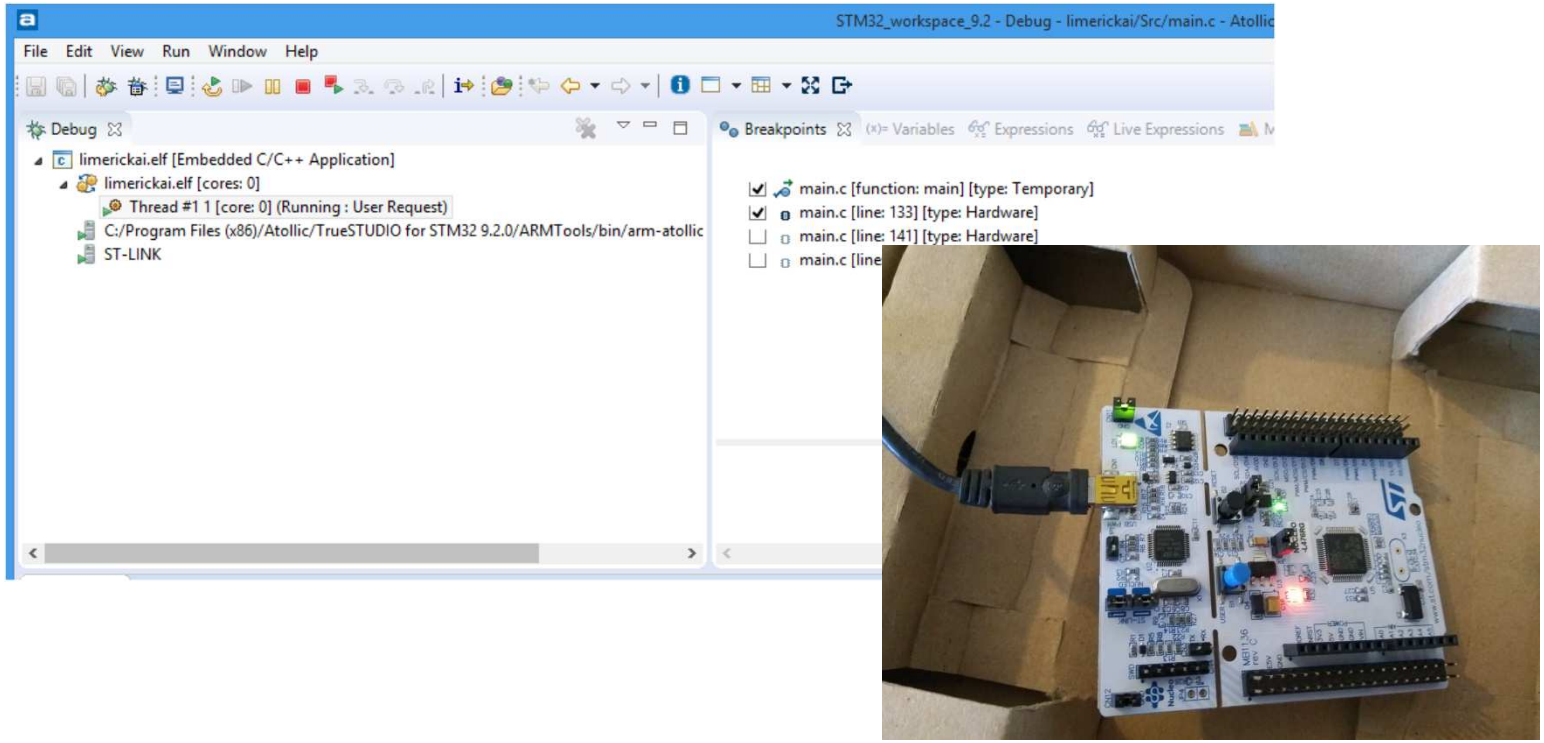
```
<terminated> limerickai.elf [Embedded C/C++ Application] ST-LINK


STMicroelectronics ST-LINK GDB server. Version 5.1.0
Copyright (c) 2018, STMicroelectronics. All rights reserved.

Starting server with the following options:
        Persistent Mode            : Disabled
        Logging Level              : 1
        Listen Port Number         : 61234
        Status Refresh Delay       : 15s
        Verbose Mode               : Disabled
        SWD Debug                  : Enabled


Error in initializing ST-LINK device.
Reason: ST-LINK firmware upgrade required. Please upgrade the ST-LINK firmware using the upgrade tool.
```



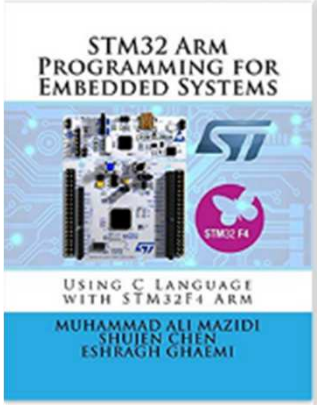
Demo

Build Demo

- We will use STM32CubeMx to generate a a project for TrueStudio and add the code to flash an on board LED on an NUCLEO–L476RG
- Purpose to explain where to put the code and how to reedit the project in STM32CubeMx



STM32 Recommended Books



- STM32 Arm Programming for Embedded Systems (Volume 6) Paperback (not available in electronic form) Authors Muhammad Ali Mazidi , Shujen Chen , Eshragh Ghaemi
- This book takes a very low level technical approach by programming registers directly (not using an HAL layer) and I recommend purchasing a NUCLEO-F446RE board as all the examples are written for this board.

- Mastering STM32 - https://www.carminenoviello.com/mastering-stm32/
- Mastering STM32 takes a higher level approach with more discussion around STM32CUbeMX, troubleshooting and good coverage of getting going with peripherals and understanding Input capture/PWM, I2C,SPI. Etc.

- Between these books you will have enough knowledge to tackle most projects. “Mastering STM32” is the best place to start and once you have some confidence and are used to uploading your programs to your board then I would suggest to take a deeper dive into the low level direct register programming approach taken by the “STM32 Arm Programming for Embedded Systems” book.

What’s next when you have learned the basics

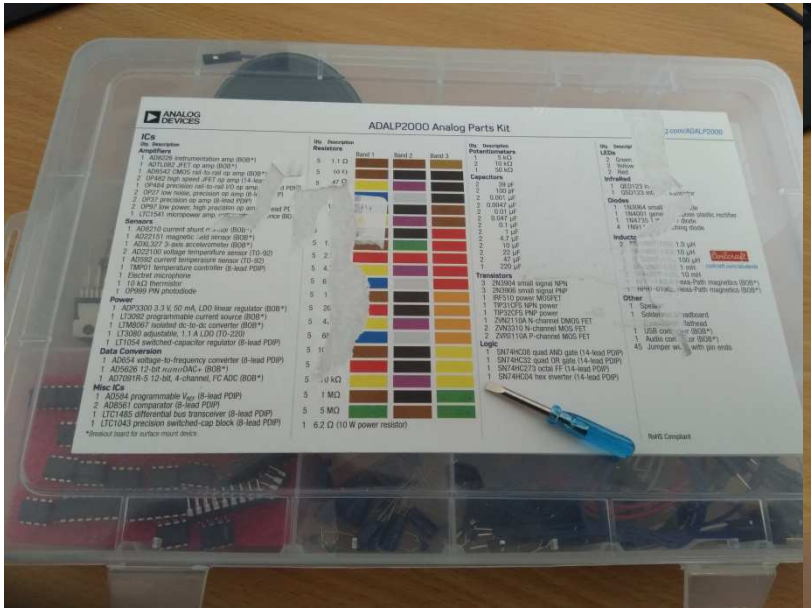
- Learn to program all the peripherals attached via I2C and SPI – perhaps buy some external components like a GPS module (left) or an Accelerometer (right) and talk to them over I2C.

IOT perhaps?

- Buy some inexpensive shields and stack them to collect environmental data, NFC data and upload it to Azure IoT Hub for example.

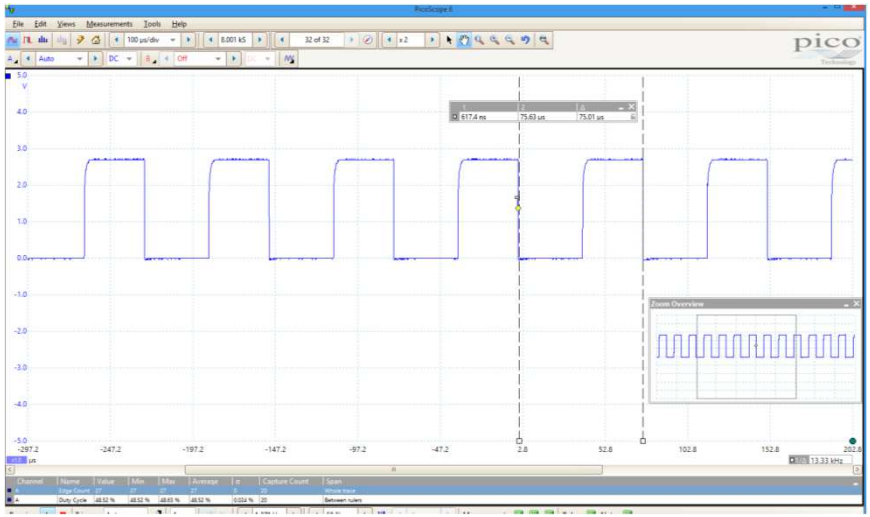
Perhaps buy some components

- There are many parts kits around you have analog pins on the Stm32,so perhaps build an analog device and capture readings.
- Analog Devices do a well known analog parts kit (part No ADALP2000Analog). (That Screwdriver is actually tiny encase your were wondering !!!!!)



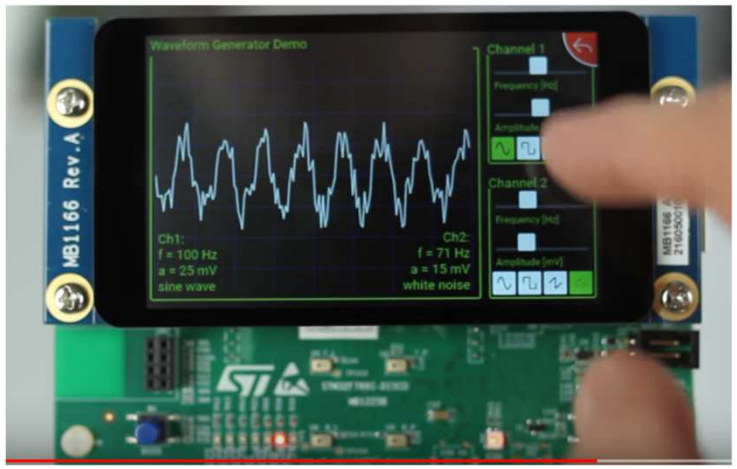
Now your more or less fully committed to Embedded development – next step !!!

- Design some circuits and build them.
- But your need a way to measure and see rise time of your signals and also troubleshoot or debug input/outputs. You can't always flash an LED or output a debug message via a UART – since you need to take account of latency.
- Circuits behaviour differently under higher clock speeds, you need to examine, measure and test your designs.
- Your need an oscilloscope : I recommend a Pico scope. https://www.picotech.com/oscilloscope/2000/picoscope-2000-overview



Or Perhaps you want to build an full embedded application with LCD display to control your house environment.

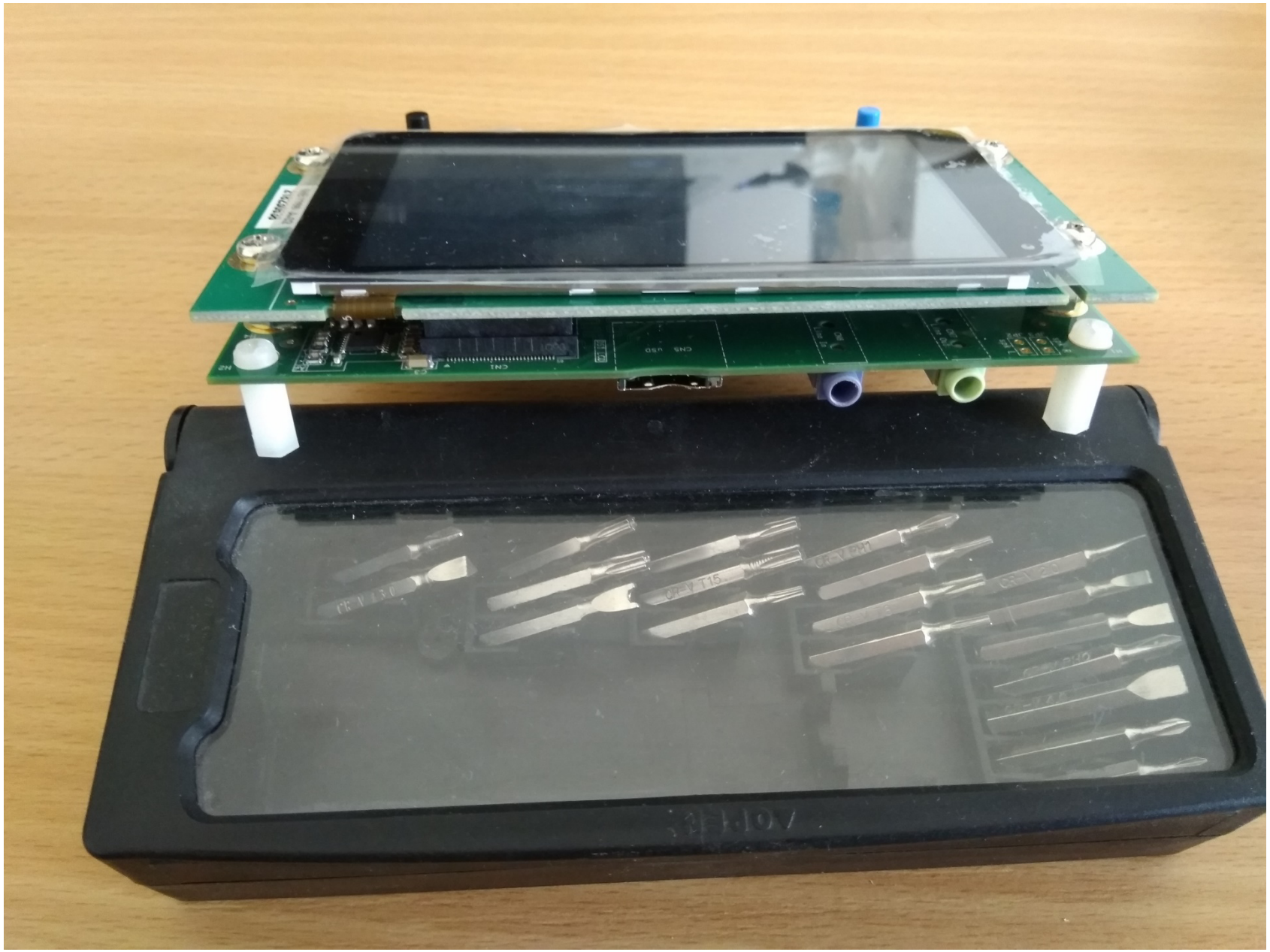
- https://www.st.com/en/evaluation-tools/32f769idiscovery.html
- STM32F769 – 4inch TFT Lcd screen
- Ethernet
- 128mb of ram
- 512Mbit flash
- Cortex M7 running at 216 Mhz with double precision FPU.
- 2x usb
- 4 audio jacks
- Wifi
- And more ….
- https://www.youtube.com/watch?v=k7-n1F2faKc



Another Example

- STmicroelectronics STM32L476RGT6 64 pins, Arm Cortex M4 at 80mhz - 1 Mbyte of ram , 128KB of flash.

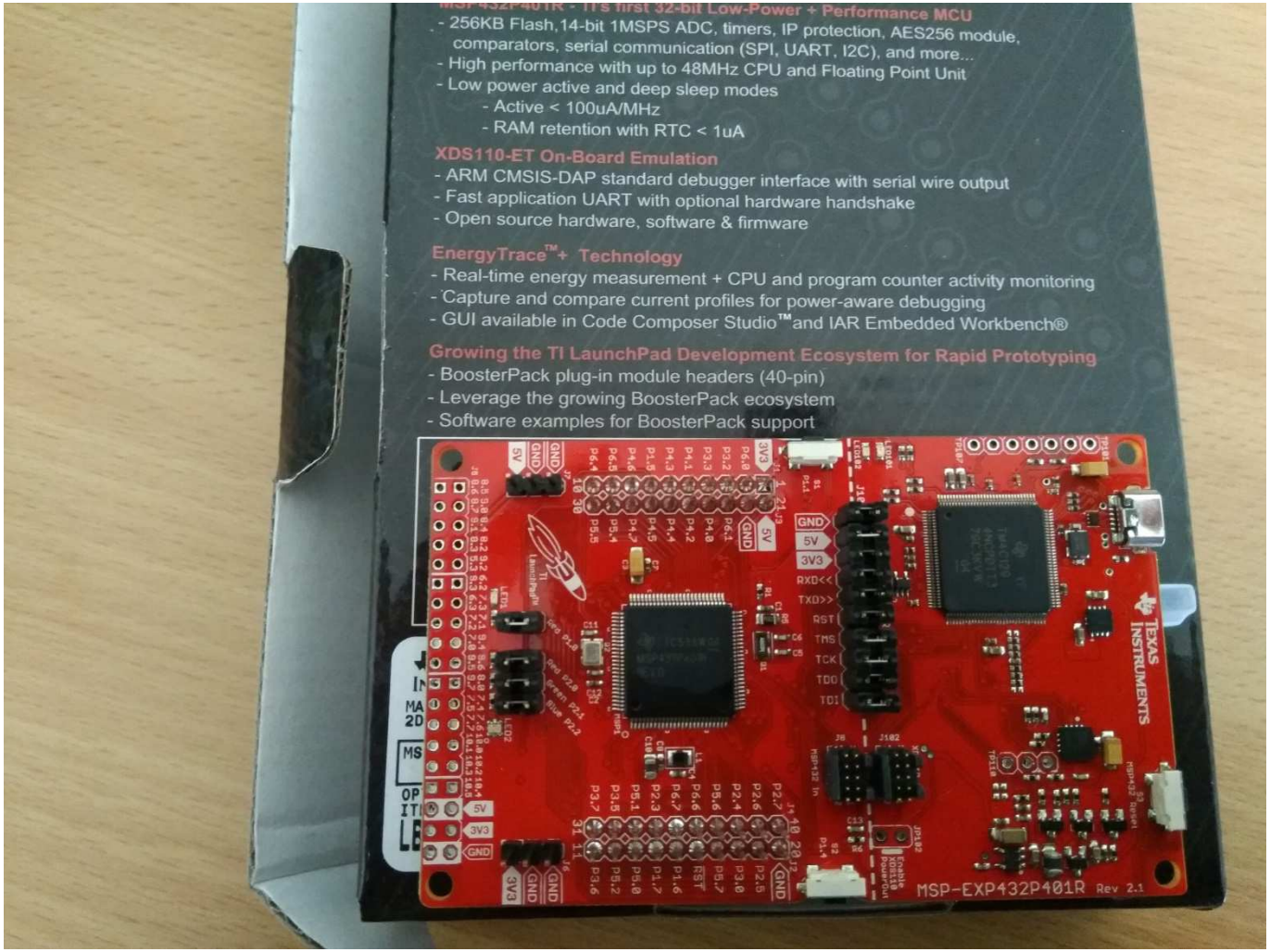
MB1166 Rev.A



Texas Instruments Boards (Launchpad Range)

Texas Instruments Boards (Launchpad Range)

MSP432P401R – which includes a 48MHz ARM® Cortex®-M4F, 80uA/MHz active power and 660nA RTC operation, SAR Precision ADC with 16-bit performance and AES256 accelerator.

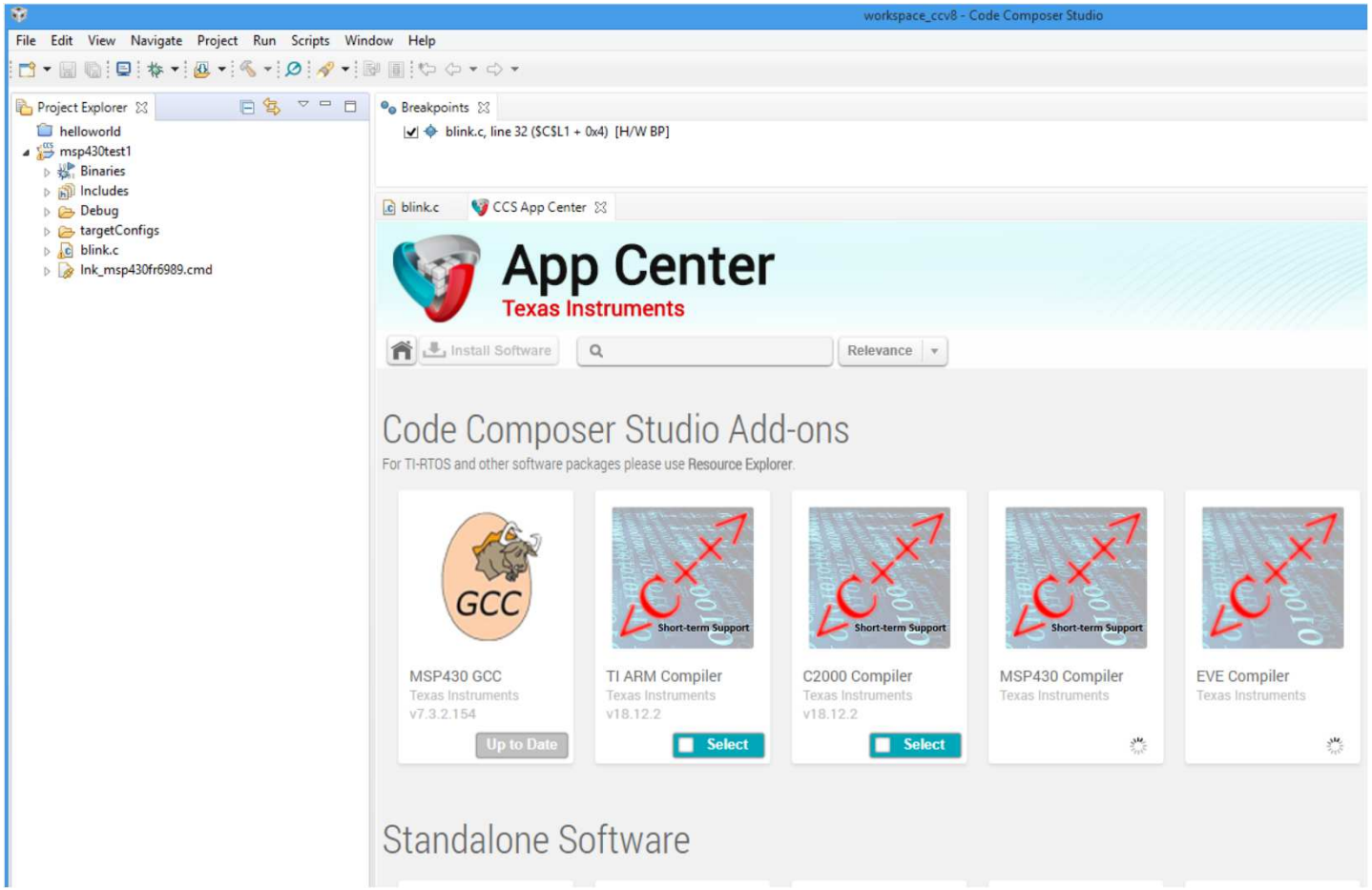


Texas Instruments Boards (Launchpad Range)

Hercules 100 pin RM42L432 R4 Cortex MCU - ecc memory for IEC61508 related critical industrial and medical applications

Code Composer latest version 9

Texas Instrument's Code Composer IDE support their range of product boards and is free to use.
(your see it's free down the bottom of the page) http://www.ti.com/tool/CCSTUDIO



Ti SimpleLink

A lot of Ti Launchpad's support SimpleLink
http://www.ti.com/wireless-connectivity/simplelink-solutions/overview/overview.html

SimpleLink™ MCU platform

One environment. Unlimited potential.

Our new SimpleLink™ microcontroller platform sets a new standard for developers with the broac
and wireless Arm® MCUs (System-on-Chip) in a single software development environment, which
hardware, software and tool options for your IoT and automotive applications. Invest once in the
development kit and use throughout your entire portfolio.

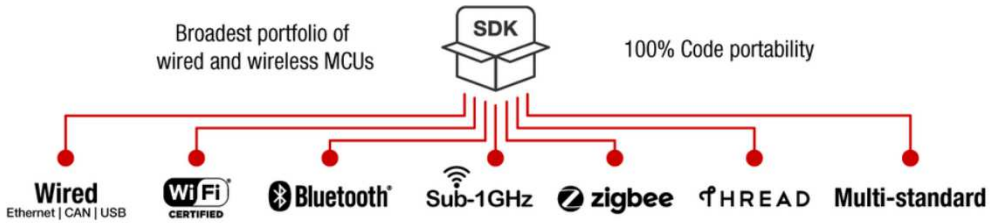
Explore SimpleLink | Innovate- BAW technology | Security | Low power | More on SimpleLink

100% application code portability

What is code portability?

The SimpleLink Platform was designed from the ground up to provide developers with flexibility to scale their product offering by adding new features to existing products or creating complementary solutions while preserving their software investment.

- 100 percent code compatibility across SimpleLink MCU portfolio
- TI Drivers offers standardized set of functional APIs for integrated peripherals
- Integrated TI-RTOS, a robust, intelligent kernel for complete, out-of-the-box development
- POSIX-compatible APIs offer flexible OS/kernels support
- Encryption-enabled security features
- IoT stacks and plugins to add functionality to your design



FreeRTOS

- From the FreeRTOS website : https://www.freertos.org/
- **We see some extracts on the website that state :**
- **FreeRTOS has a tick-less mode to directly support low power applications.**
- **FreeRTOS is designed to be simple and easy to use: Only 3 source files that are common to all RTOS ports, and one microcontroller specific source file are required, and its API is designed to be simple and intuitive.**
- **The RL78 port can create 13 tasks, 2 queues and 4 software timers in under 4K bytes of RAM!**

- **Licensing**
- Both the FreeRTOS kernel and Amazon FreeRTOS are provided under the following MIT open source license. The table below contains answers to common license questions.

- Other options for Licensing when a company requires commercial legal assurances
- **OpenRTOS** is a commercially licensed version of the FreeRTOS kernel that includes indemnification and dedicated support. FreeRTOS and OpenRTOS share the same code base. OpenRTOS is provided under license from AWS by WITTENSTEIN high integrity systems - an AWS strategic partner.

- Or need to be meet ISO safely assurances regulation standards.
- **SAFERTOS** is a derivative version of the FreeRTOS kernel that has been analysed, documented and tested to meet the stringent requirements of industrial (IEC 61508 SIL 3), medical (IEC 62304 and FDA 510(K)) automotive (ISO 26262) and other international safety standards. SafeRTOS includes independently audited safety lifecycle documentation artifacts. SafeRTOS is provided by WITTENSTEIN high integrity systems - an AWS strategic partner.

FreeRTOS

Kernel License Comparison

	FreeRTOS Kernel MIT License	OpenRTOS Commercial License
Is it free?	Yes	No
Can I use it in a commercial application?	Yes	Yes
Is it royalty free?	Yes	Yes
Is a warranty provided?	No	**Yes**
Can I receive professional technical support on a commercial basis?	Yes	Yes
Is legal protection provided?	No	**Yes**, IP infringement protection is provided
Do I have to open source my application code that makes use of the FreeRTOS services?	No	No
Do I have to open source my changes to the RTOS kernel?	No	No
Do I have to document that my product uses FreeRTOS?	No	No
Do I have to offer to provide the FreeRTOS code to users of my application?	No	No

» Table https://www.freertos.org/a00114.html#commercial

FreeRTOS Documentation

- https://www.freertos.org/Documentation/RTOS_book.htmlhttps://www.freertos.org/

- Mastering the FreeRTOS Real Time Kernel - a Hands On Tutorial Guide

https://www.freertos.org/Documentation/161204_Mastering_the_FreeRTOS_Real_Time_Kernel-A_Hands-On_Tutorial_Guide.pdf

- FreeRTOS V10.0.0 Reference Manual
- https://www.freertos.org/Documentation/FreeRTOS_Reference_Manual_V10.0.0.pdf

- Book companion source code
- https://sourceforge.net/projects/freertos/files/FreeRTOS/V9.0.0/source-code-for-book-examples.zip/download

RTOS Example to explain

- https://www.freertos.org/a00102.html

A typical main() function will have the following structure:

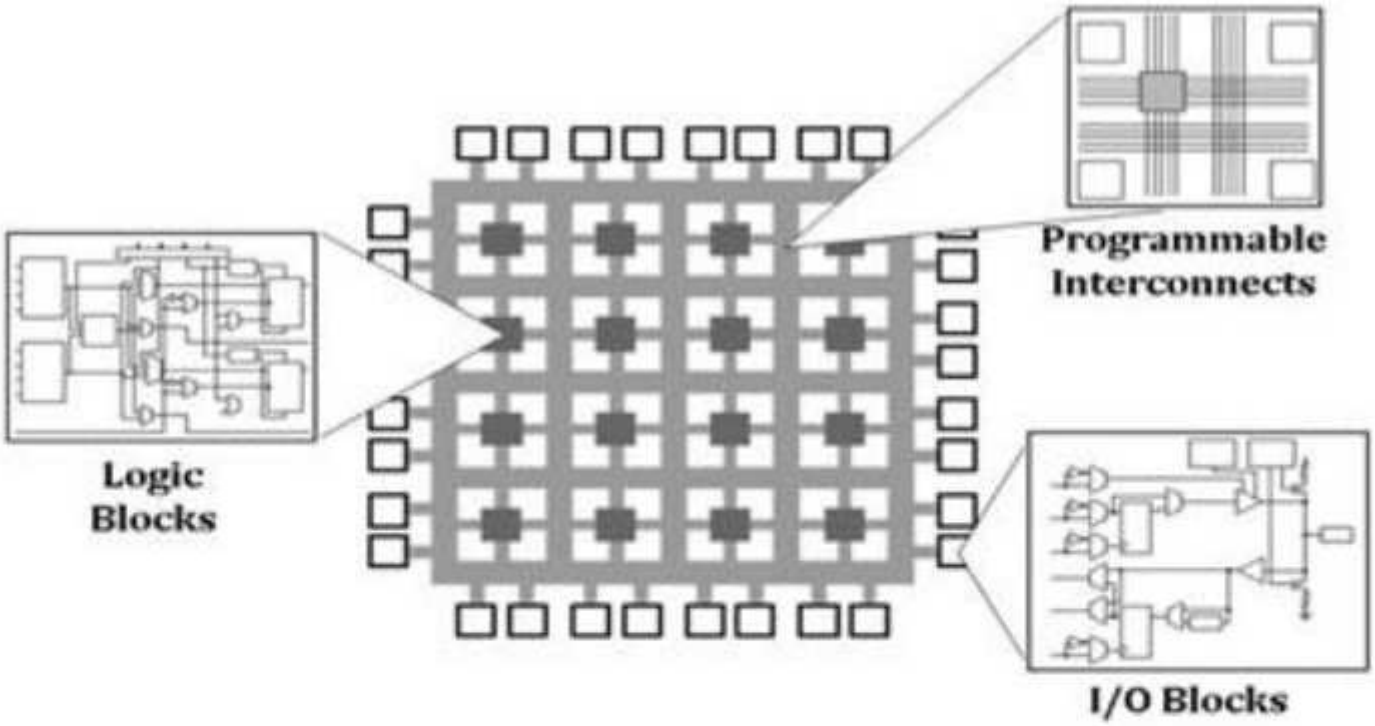
```
int main( void )
{
   /* Setup the microcontroller hardware for the demo. */
   prvSetupHardware();

   /* Create the common demo application tasks, for example: */
   vCreateFlashTasks();
   vCreatePollQTasks();
   vCreateComTestTasks();
   Etc.

   /* Create any tasks defined within main.c itself, or otherwise specific to the
   demo being built. */
   xTaskCreate( vCheckTask, "check", STACK_SIZE, NULL, TASK_PRIORITY, NULL );
   Etc.

   /* Start the RTOS scheduler, this function should not return as it causes the
   execution context to change from main() to one of the created tasks. */
   vTaskStartScheduler();

   /* Should never get here! */
   return 0;
}
```

I wanted to cover ROS(the Robot Operating System) and OpenCV – but decided to leave them to later date. Perhaps focused for a future more in-depth **Autonomous Robotics** meetup night.

ROS and Opencv is an important part of most Robotics or Autonomous Driving courses and so is certainly worth covering.

FPGAs

FPGAs what are they?



http://www.xilinx.com/images/fpga-block-structure.gif

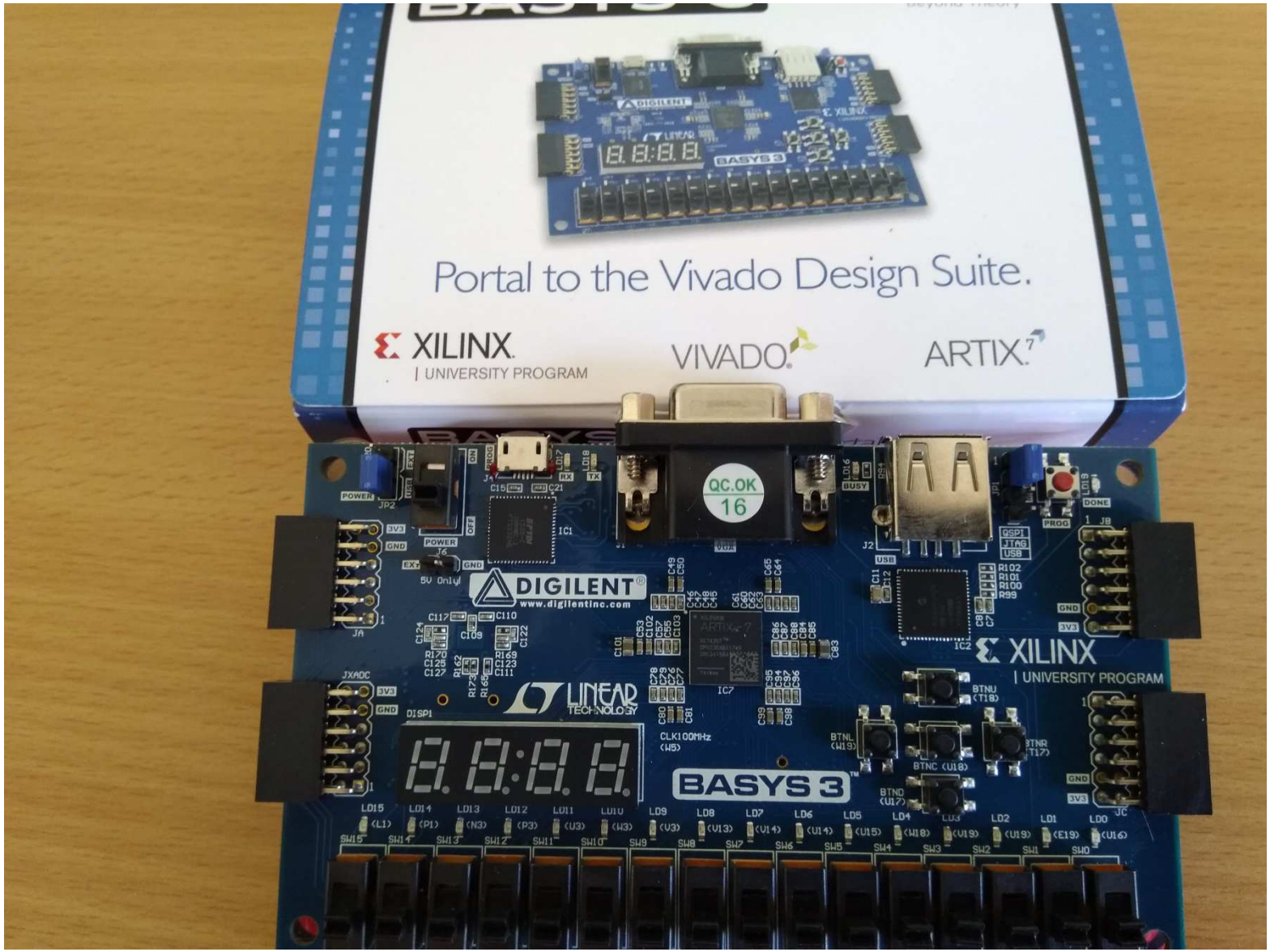
FPGAs How do they work?

- Every FPGA chip is made up of a finite number of predefined resources with programmable interconnects to implement a reconfigurable digital circuit and I/O blocks to allow the circuit to access the outside world.

- FPGA resource specifications often include the number of configurable logic blocks, number of fixed function logic blocks such as multipliers, and size of memory resources like embedded block RAM. Of the many FPGA specifications, these are typically the most important when selecting and comparing FPGAs for a particular application.

- The configurable logic blocks (CLBs) are the basic logic unit of an FPGA. Sometimes referred to as slices or logic cells, CLBs are made up of two basic components: flip-flops and lookup tables (LUTs). Various FPGA families differ in the way flip-flops and LUTs are packaged together, so it is important to understand flip-flops and LUTs.
- https://www.ni.com/en-us/innovations/white-papers/08/fpga-fundamentals.html

Basys3


Portal to the Vivado Design Suite.
XILINX
UNIVERSITY PROGRAM
VIVADO
ARTIX
DIGILENT
www.digilentinc.com
LINEAR TECHNOLOGY
BASYS 3
QC.OK
16

BASYS 3

DIGILENT
Beyond Theory

- **Versatile prototyping board for both student projects & classroom concepts**
- **Ideal starting point for anyone looking to learn the latest Xilinx architecture**
- Features the Xilinx Artix-7™ FPGA (XC7A35T-1CPG236C)
  - 33,280 logic cells in 5200 slices
  - 1,800 Kbits of fast block RAM
  - Internal clock speeds exceeding 450MHz
- Serial FLASH
- USB-UART bridge
- Four 12-pin Pmod™ connectors
- 12-bit VGA output
- Digilent USB-JTAG port for FPGA programming & communication
- **Designed for use exclusively with Vivado® Design Suite**

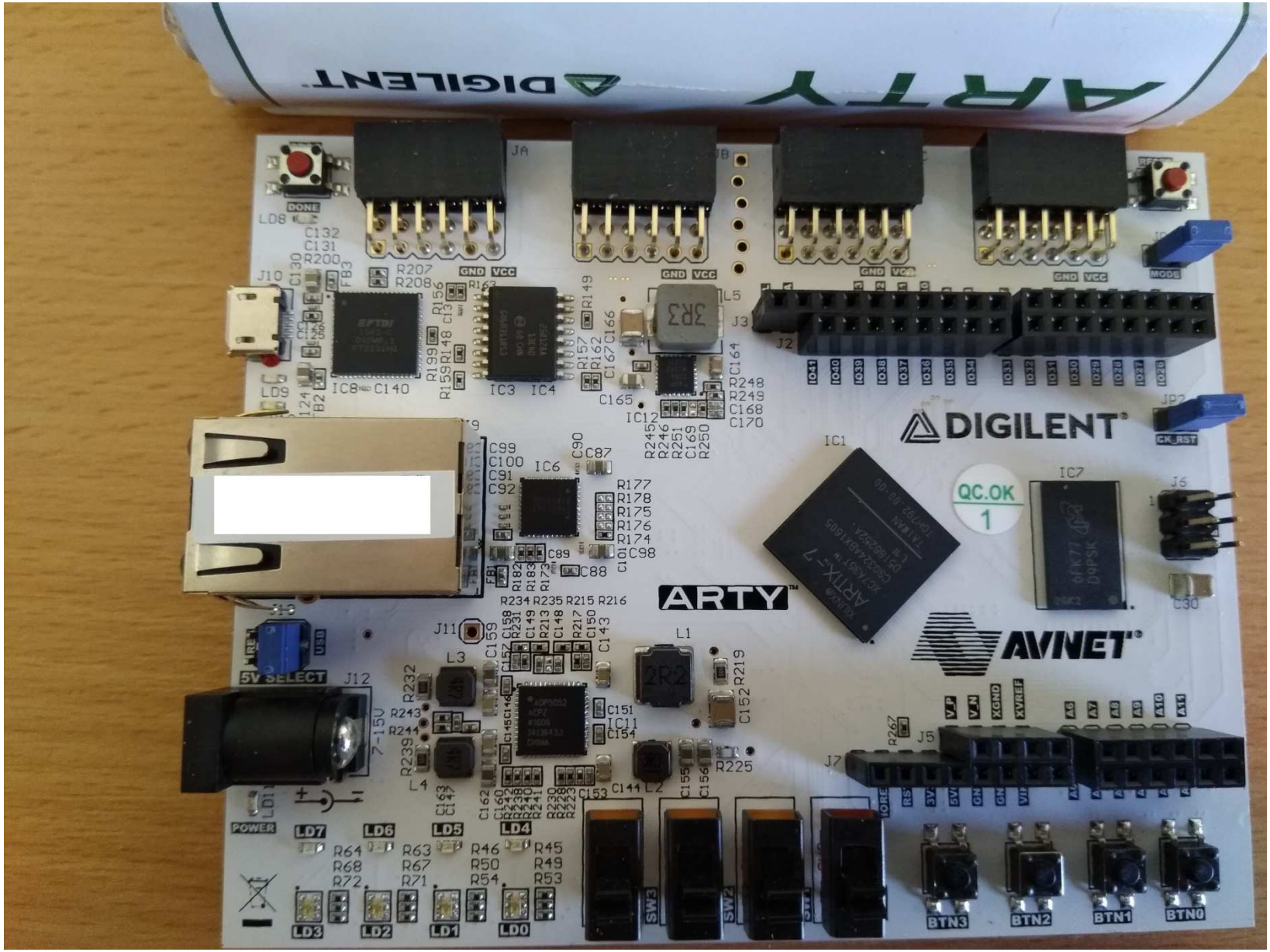
Designed in Cooperation With:

XILINX® LINEAR® TECHNOLOGY

Designed for Use With:

VIVADO®

Artix-7 35T Arty FPGA Evaluation Kit



Art7 35T Comes with a 1 year Vivado Design Suite License

rtix-7™ 35T Evaluation Kit and Tools Voucher

ncludes Vivado Design Suite, Design Edition
The Artix-7™ Kit "ARTY" includes one year of support and updates for the Xilinx Vivado® Design Suite, Design Edition License. This license is node ocked, and device locked to the Artix-7™ 35T device.

Vivado Design Suite – *Productivity. Multiplied.*
Vivado delivers a SoC-strength, IP-centric and system-centric, next generation development environment that has been built from the ground up to address the productivity bottlenecks in system-level integration and mplementation. Vivado is a generation ahead in overall productivity, ease-of-use, and system-level integration capabilities.

VIVADO

✓ Accelerating Implementation
✓ Accelerating Integration
✓ Accelerating Verification

IMPORTANT VOUCHER REDEMPTION INFORMATION

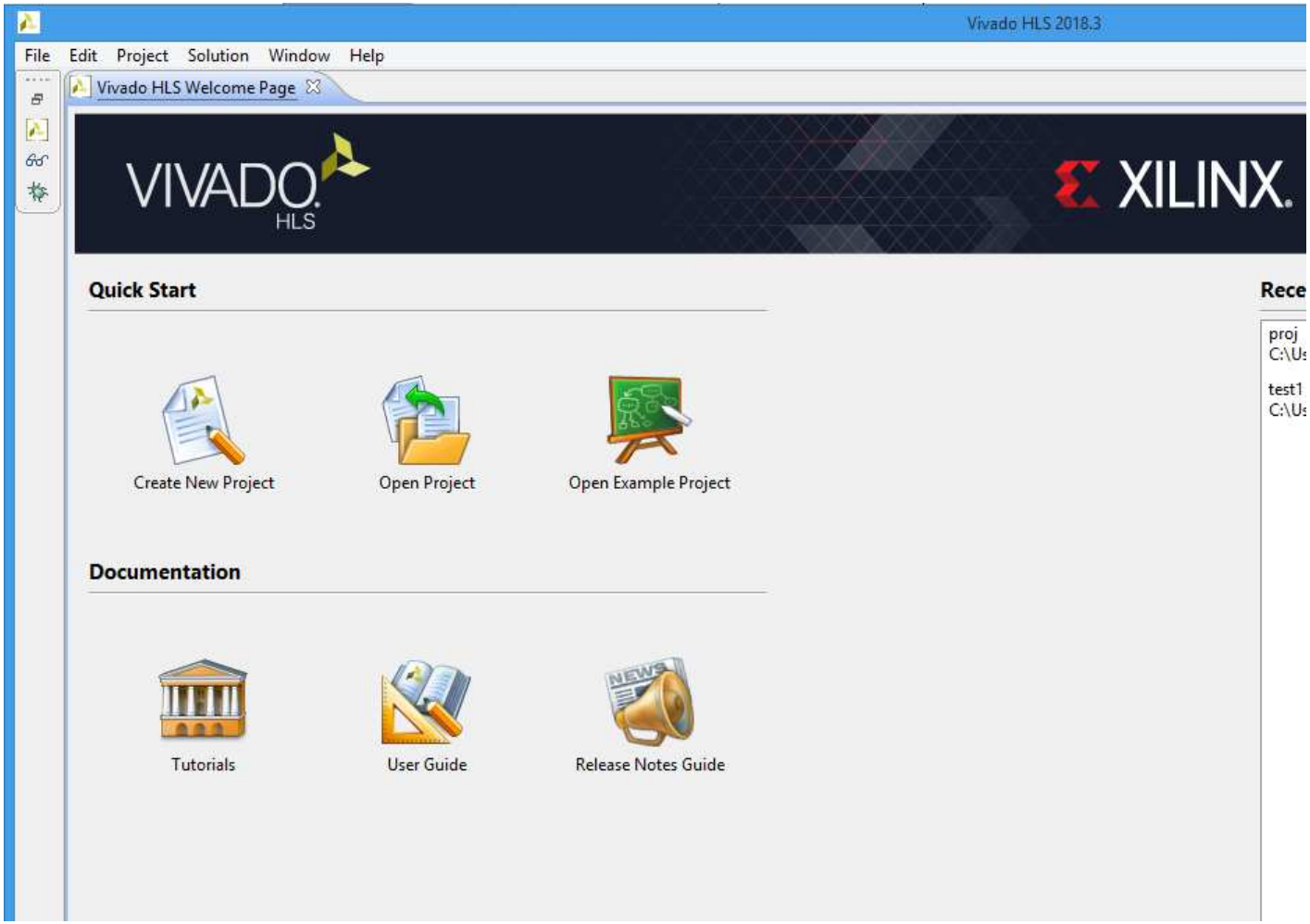
This voucher is proof that you are entitled to a Xilinx Software Design Tool Produ

In Order to Obtain Your License Please Log Into: http://www.Xilinx.com/getlicens
and enter the code listed below. A license will be added to your account features.

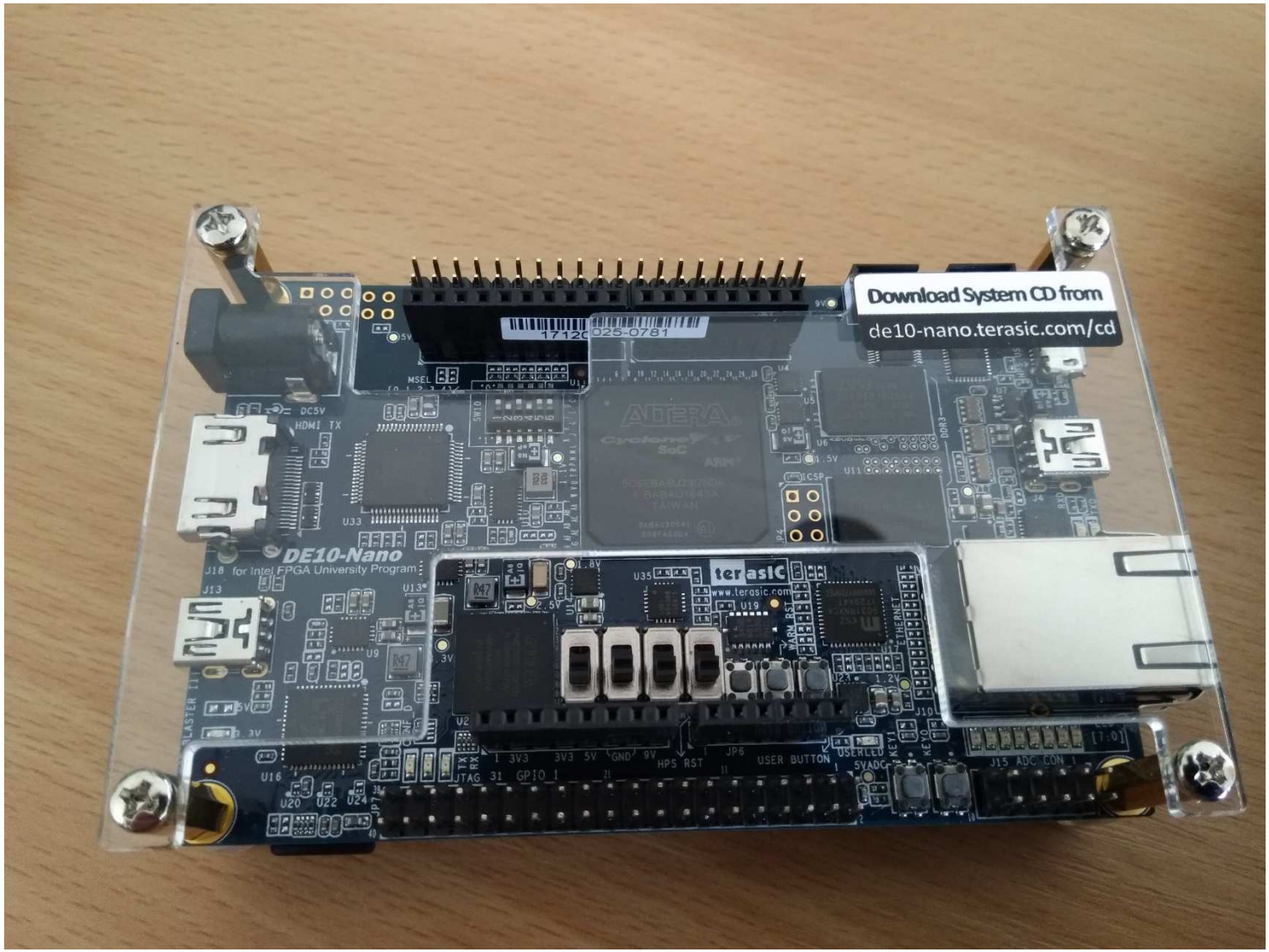
NOTE: This voucher code may only be used once and must be redeemed within one year.

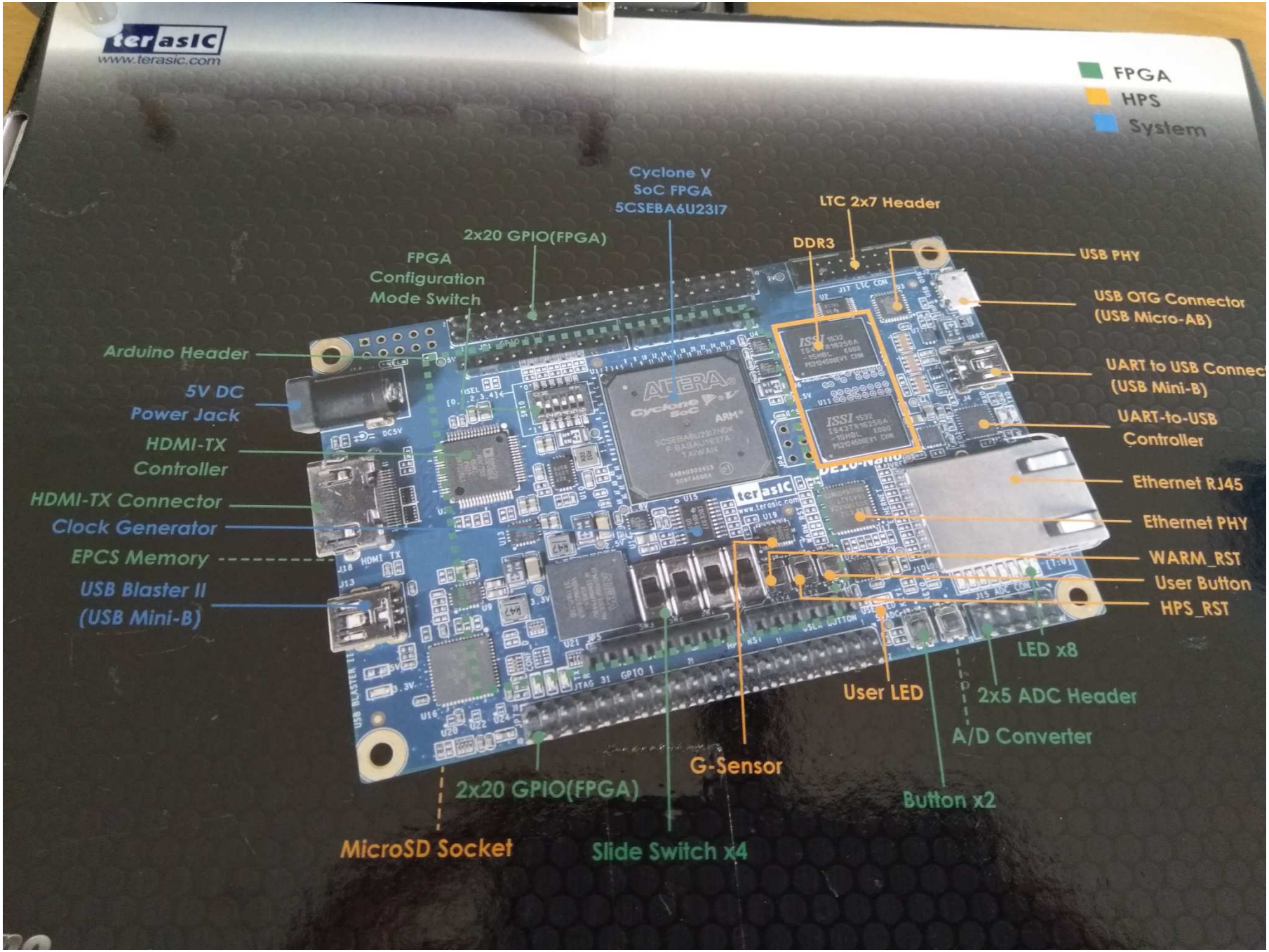
Vivado Design Suite

https://www.xilinx.com/support/documentation-navigation/design-hubs/dh0012-vivado-high-level-synthesis-hub.html

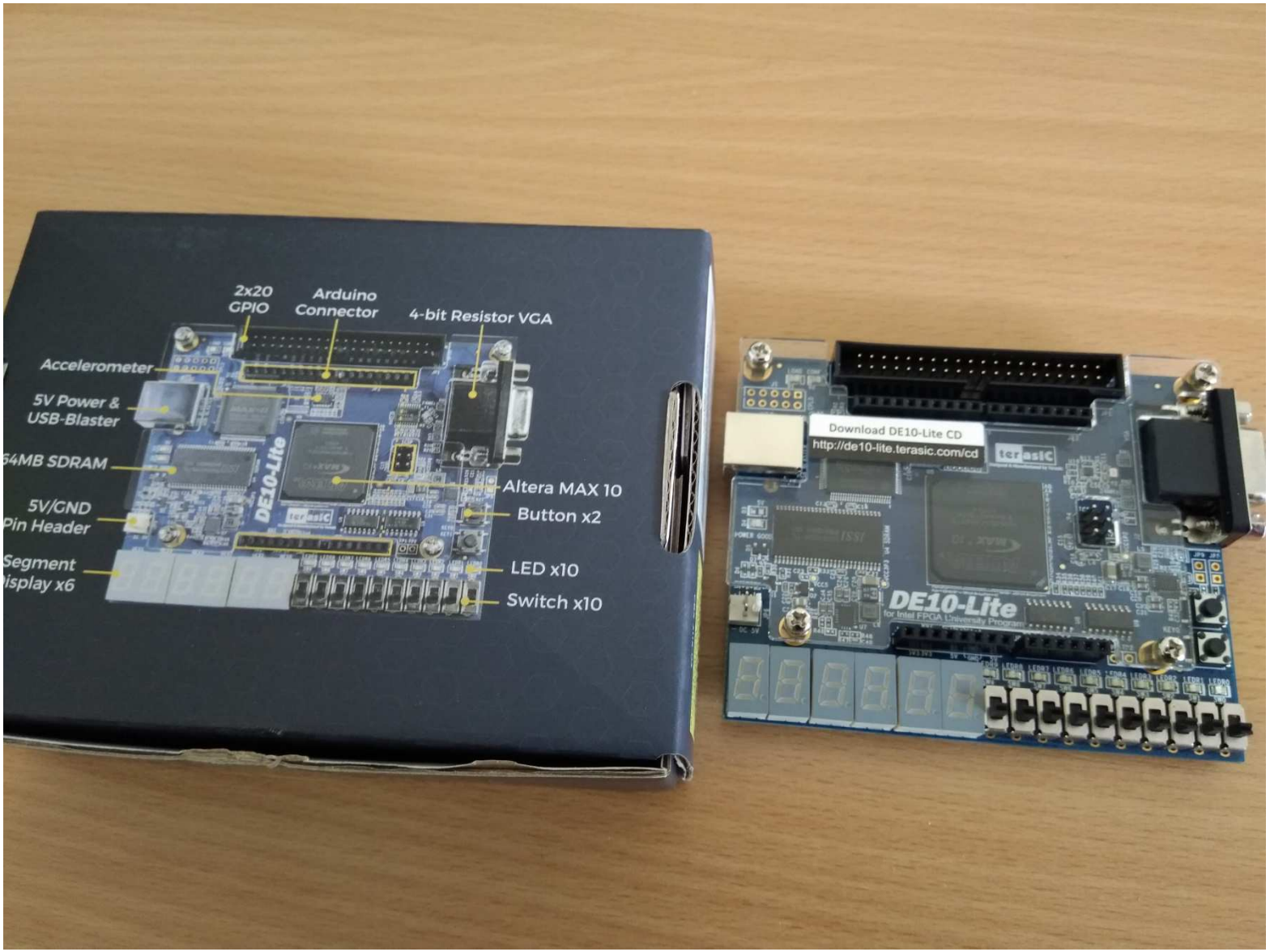


DE10 Nano


Download System CD from
de10-nano.terasic.com/cd
DE10-Nano
for Intel FPGA University Program
terasIC
www.terasic.com

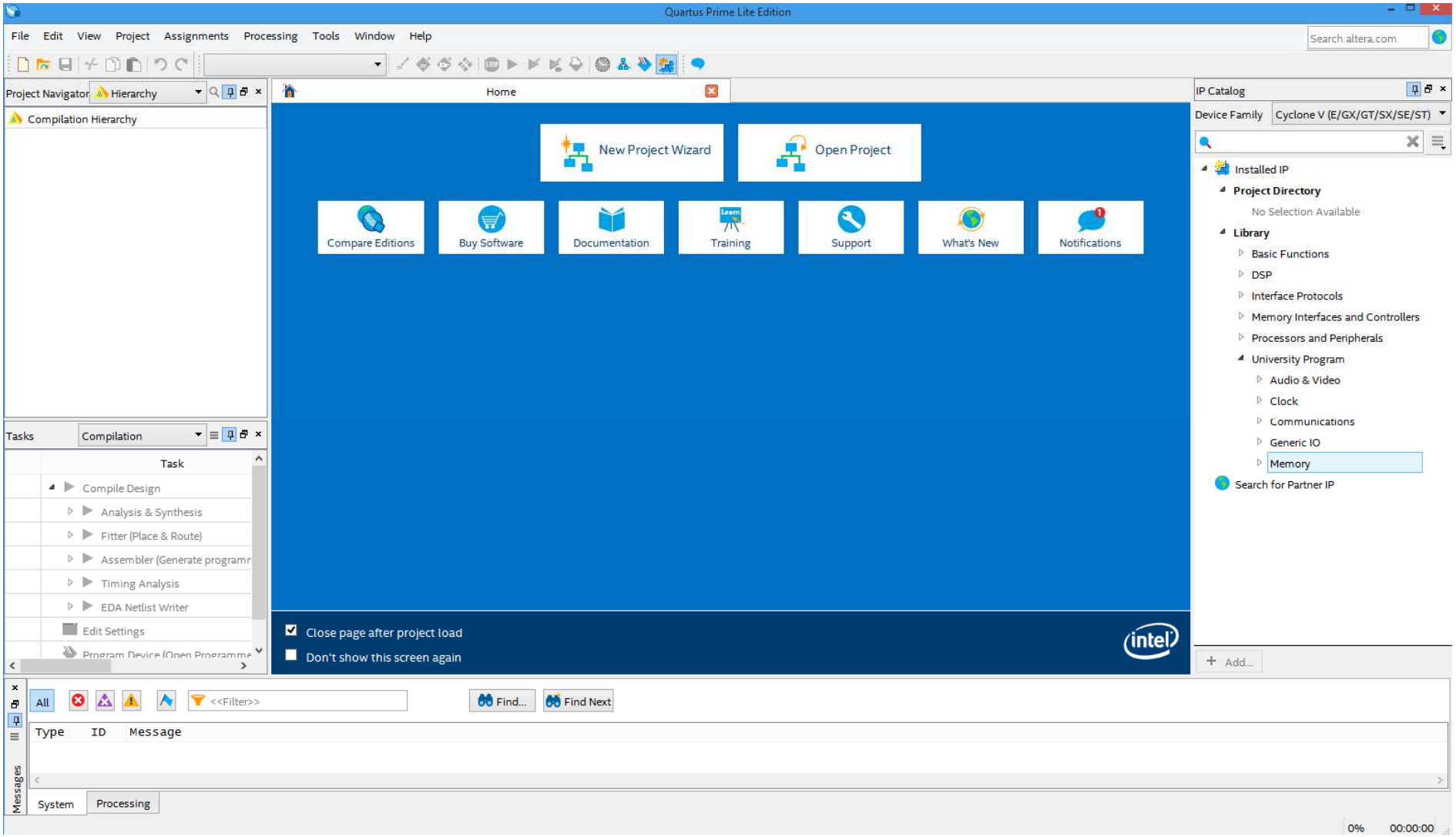

terasic
www.terasic.com
FPGA
HPS
System
Cyclone V
SoC FPGA
5CSEBA6U23I7
LTC 2x7 Header
2x20 GPIO(FPGA)
DDR3
USB PHY
FPGA
Configuration
Mode Switch
USB OTG Connector
(USB Micro-AB)
Arduino Header
UART to USB Connect
(USB Mini-B)
5V DC
Power Jack
UART-to-USB
Controller
HDMI-TX
Controller
Ethernet RJ45
HDMI-TX Connector
Ethernet PHY
Clock Generator
WARM_RST
EPCS Memory
User Button
USB Blaster II
(USB Mini-B)
HPS_RST
LED x8
User LED
2x5 ADC Header
A/D Converter
G-Sensor
2x20 GPIO(FPGA)
Button x2
MicroSD Socket
Slide Switch x4

DE10 Nano lite



Quartus Rime Verison 18.1 Lite Edition

http://fpgasoftware.intel.com/?edition=lite



That’s all folks