

FPGAWorld 2014, DTU Lyngby, Denmark  
11 September 2014

# A Multi-Paradigm Approach to Teaching Students Embedded Systems Design using FPGAs and CPLDs



by

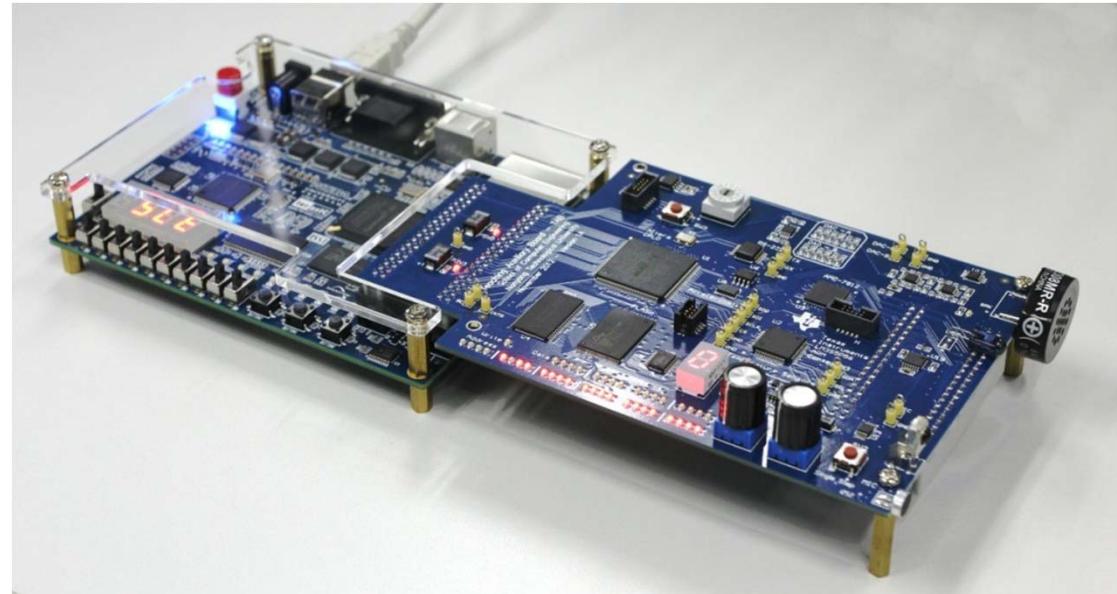
**David C. Dyer**

*School of Engineering  
University of Warwick, UK  
Previously at SCE*



**Yan Lin Aung**

*CHiPES Research Centre  
Nanyang Technological University  
Singapore*



# Agenda

- 1) From Novice to Expert : Knowledge and Skills
- 2) What we have built.
  - Inside the FPGA
  - Teaching Auxiliary Board - TAB
  - Inside the CPLD
- 3) Holistic learning opportunities. Examples
  - HDL and ISAs
  - Simulation of VIP via a soft-core ARM Cortex-M1
  - What is VIP like?
  - Soft-core VIP in VHDL, (and Verilog and LISA?)
- 4) Video
- 5) Demonstration(s)

# Knowledge and Skill Matrix

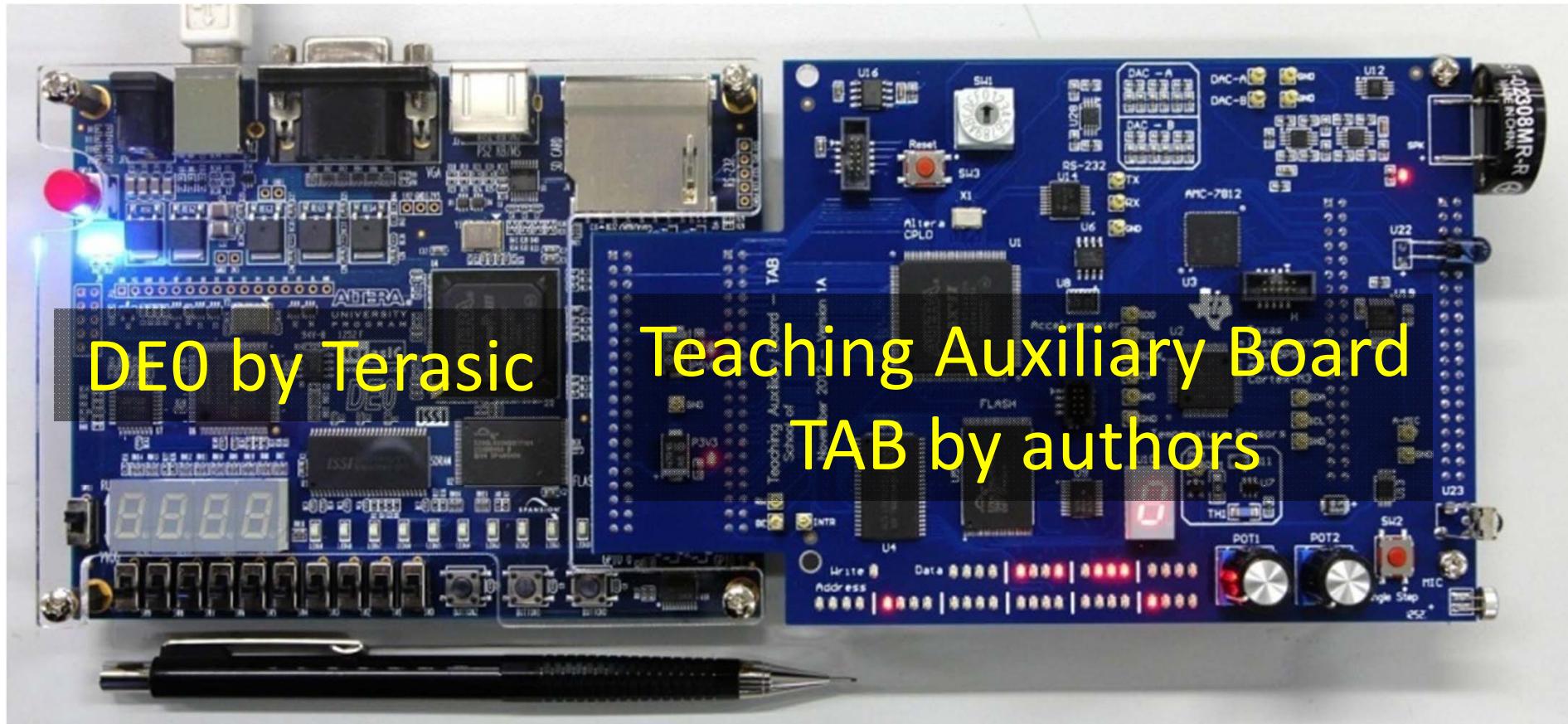


	System Design	VHDL or Verilog or ?	CPLD	FPGA	Processors & Programming	Product Verification
Design Team Leader in Safety-Critical Systems	✓✓	✓✓	✓ ✓	✓ ✓	✓✓✓✓✓	✓✓✓ ✓✓
Experienced designer	✓✓	✓✓	✓	✓	✓✓	✓✓
'Academic'	?	?	?	?	?	?
'Teacher'	?	✓	✓	✓	?	?
Designer	?	?	?	?	?	?
Post Doctorate	?	?	?	?	?	?
Graduate	?	?	?	?	?	?
Novice	✗	✗	✗	✗	✗	✗

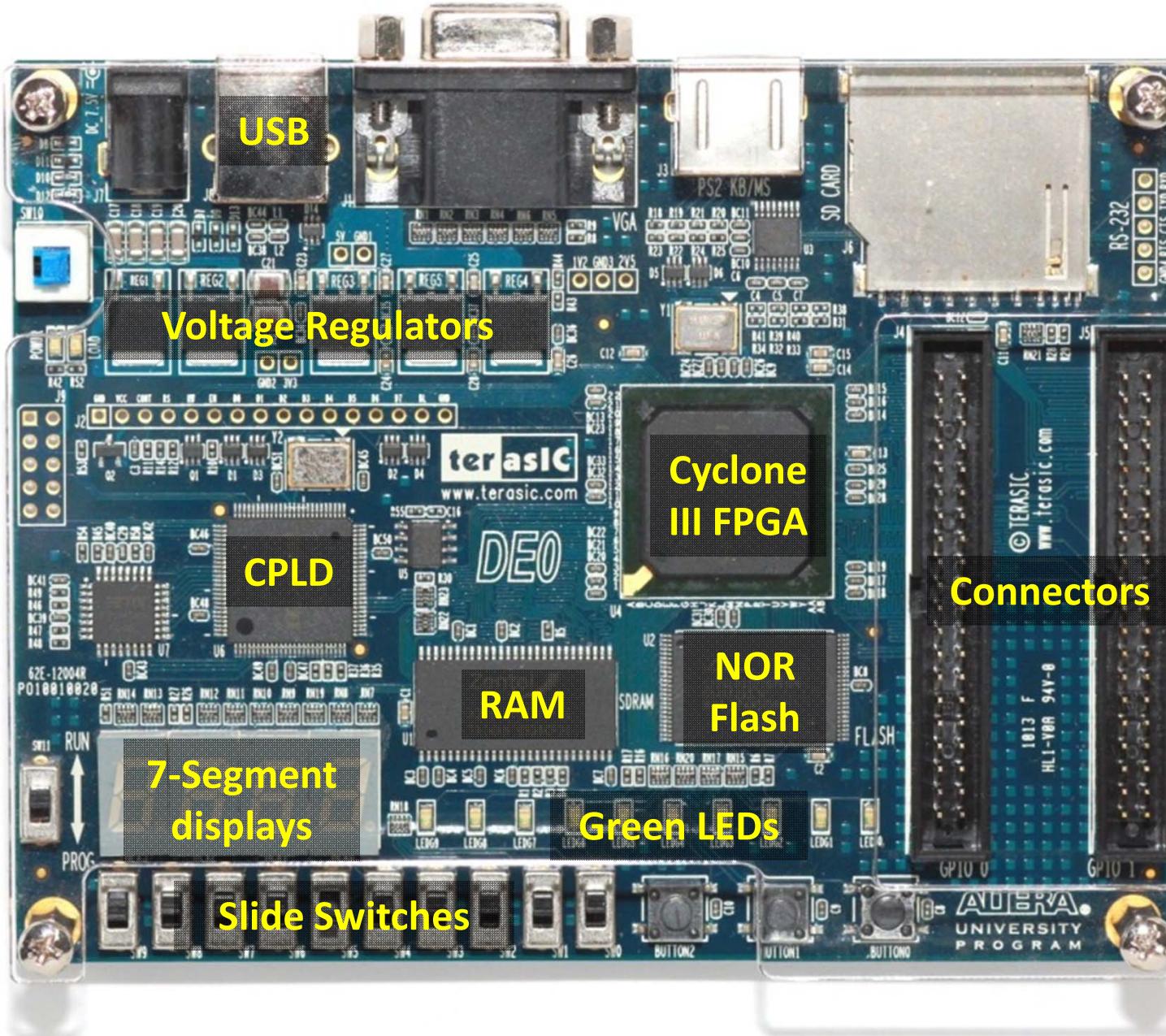
# What have we designed and built?



- A platform that allows staff and students to explore a wider variety of 'experiments' than can be achieved using off-the-shelf commercial products alone.



# DE0 unit from Terasic

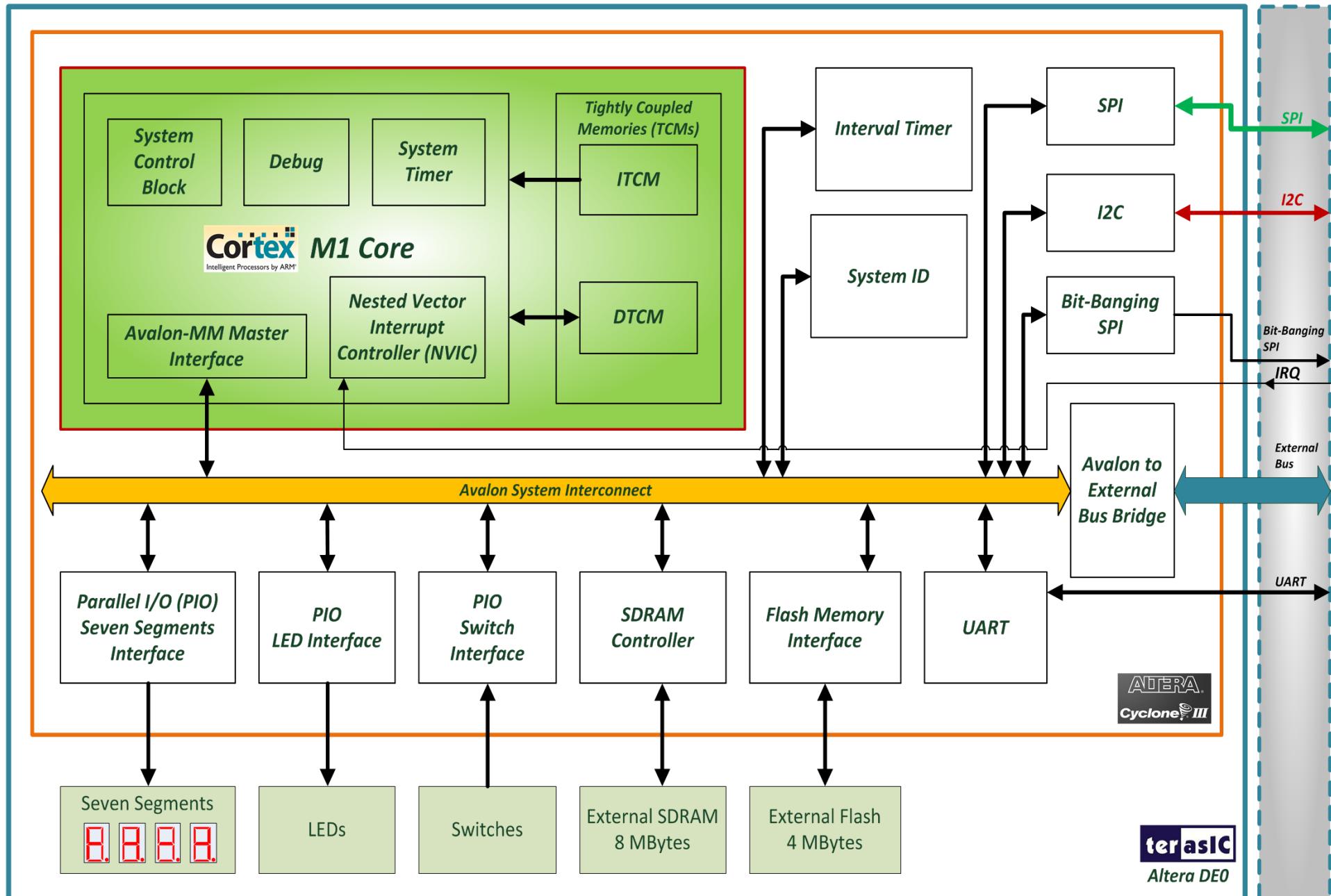


# FPGA Content



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

THE UNIVERSITY OF  
WARWICK

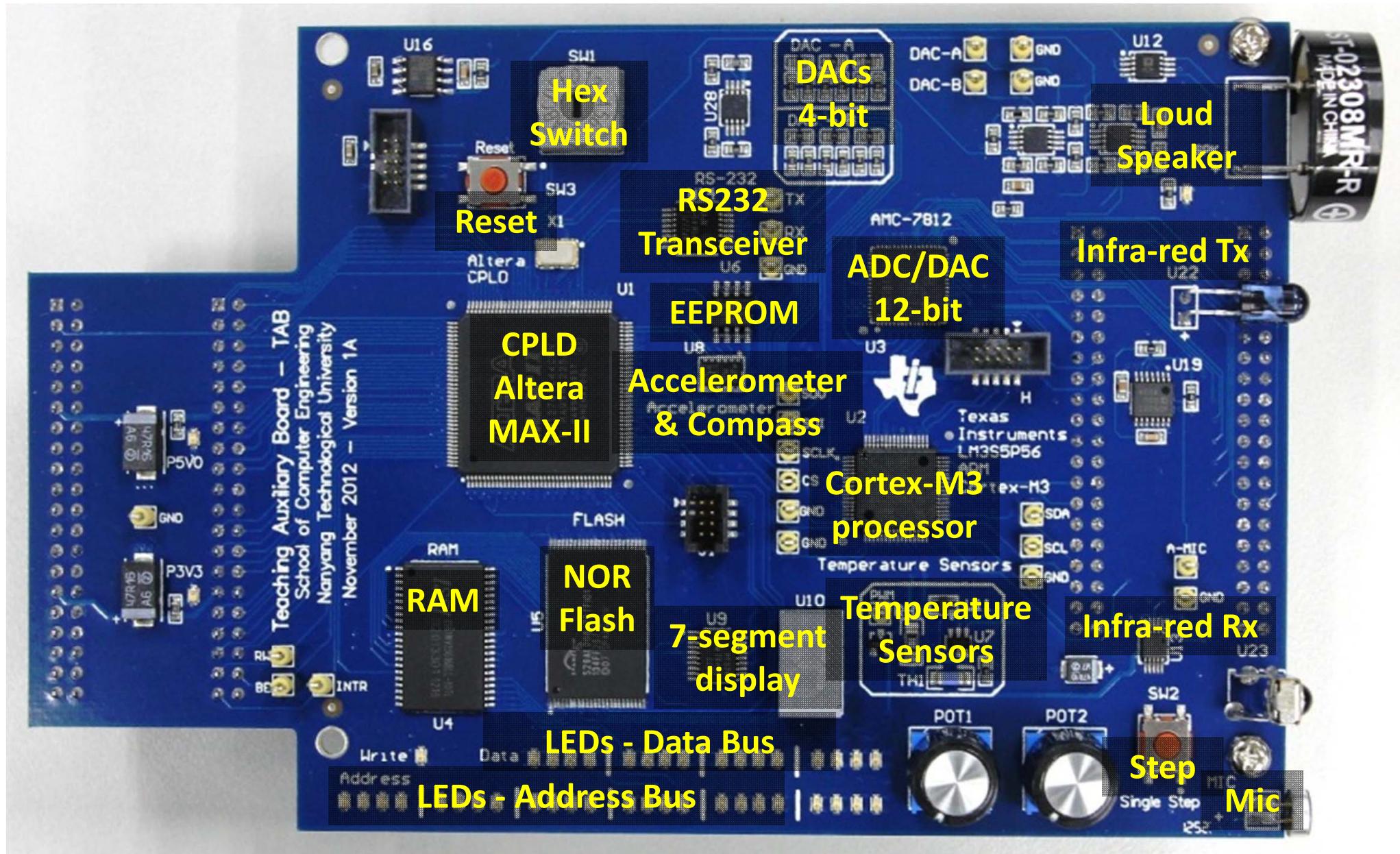


# TAB physical realisation



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

THE UNIVERSITY OF  
WARWICK

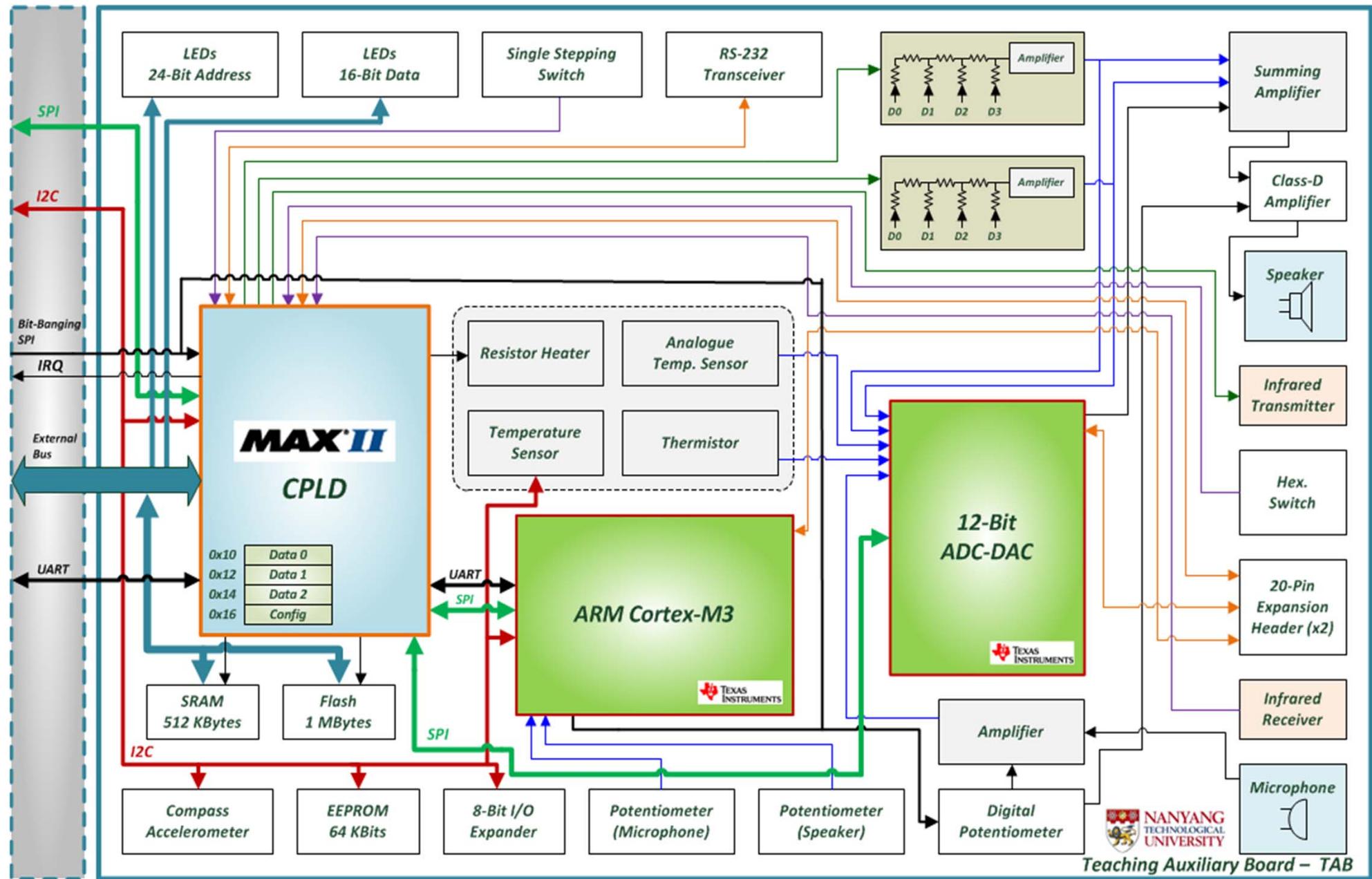


# Teaching Auxiliary Board (TAB) Block Diagram



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

THE UNIVERSITY OF  
WARWICK

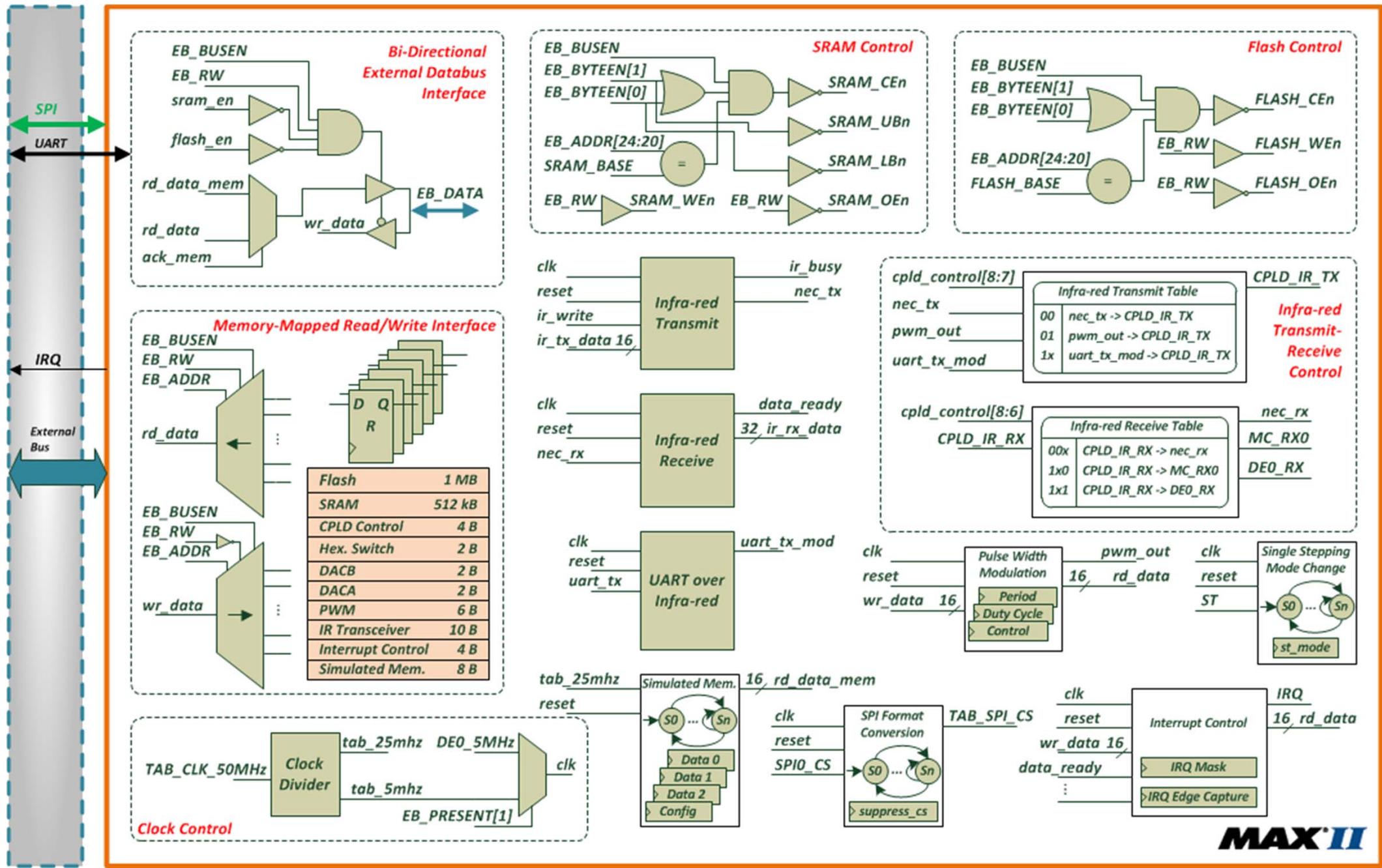


# CPLD content and interfaces



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

THE UNIVERSITY OF  
WARWICK





- Our platform, described in more detail in Section 5 of the full paper, incorporates an FPGA and CPLD together with associated components and programming environments. They are intended to support classes from years 1 to 3 (or 4) of BSc degree programs, such as Computer Engineering, Electronics or Information Engineering. The concepts provide opportunities for improved holistic understanding of principles that may be developed further for deployment in commercial products. The expected topics/classes include
  - **Introduction to Instruction Set Architecture (ISA) use and design (or design then use).**



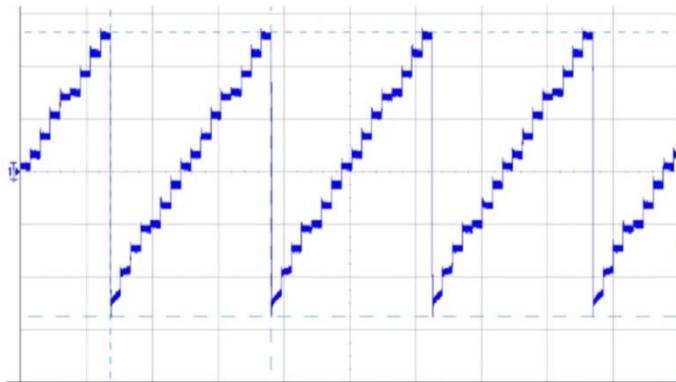
- Simplicity, or complexity, of ISA implementation via Hardware Description Languages (HDLs) with FPGA targets. Leading to Advanced ISA design and implementation with focus on low-power.
- Hardware peripheral interfacing and dedicated peripheral implementations in HDLs for FPGAs
- Real-time programming in assembler and ‘C’.
- Use of ‘Intellectual Property’ and integration with proprietary busses and architectures.
- Schedulers, Kernels and Operating systems. Particularly, uC/OS III and ThreadX
- Compiler design and program language translation.

## Two Examples



- 4-bit DACs –VIP 12-bit ISA simulated by M1 soft-core

- 4-bit ideal and non-ideal DACs
- Allows measurements and analysis related to linearity



```
000 02C FF0 MOV R2,#0xFF0
002 800      MOVS R0,#0
003 811      MOVS R1,#1
004 060 000 MOV [R2+0x000],R0
006 061 001 MOV [R2+0x001],R1
008 900      INC R0 ;ramp
009 971      PRSG R1 ;random
00A BF9      JMP -7
```

- Infra-red communications using Cortex-M1 16/32 bit ISA
  - 38 kHz modulated infra-red transmitter and receiver
  - Same technology and protocols as used in TV remote control units
  - Protocol manager is within CPLD

- Paradigm means
  - “an example that serves as a pattern or model for something, especially one that forms the basis of a methodology or theory” *Encarta Dictionary*.
- VIP is a 12-bit processor especially designed for teaching multiple principles in Instruction Set Architectures.
- Its ISA is formed from easy-to-read 4-bit hex fields.

11	10	9	8	7	6	5	4	3	2	1	0
0-7	Dual operand				d				s		
8	Short Move				d				n		
9-A	Unary or Control				Operation				s/d or n		
B-F	JMP				2's complement -128 to +127 relative displacement						

# What does a VIP program look like?

- E.g to compute the first 16 Fibonacci numbers.
  - $F(0)=0; F(1)=1; F(n)=F(n-2)+F(n-1)$  - Where are these used?
  - 0, 1, 1, 2, 3, 5, 8, 13, ...

Adr	Opcodes	Mnemonic	
000	01C 100	MOV R1, #0x100	R1 = Address of $F(0)$
002	850	MOVS [R1], #0	$F(0)=0$
003	901	INC R1	R1 = Address of $F(1)$
004	851	MOVS [R1], #1	$F(1)=1$
005	88E	MOVS AR, #0xE	Loop counter AR = 14
006	901	INC R1	R1 = Address of $F(n)$
007	021	MOV R2, R1	Copy of current address
008	006 FFE	MOV R0, [R2+0xFFE]	$R0 = F(n-2)$
00A	406 FFF	ADD R0, [R2+0xFFFF]	$R0 = R0 + F(n-1)$
00C	050	MOV [R1], R0	$F(n) = F(n-2) + F(n-1)$
00D	A88	JDAR -8	Decrement AR and if non-zero jump back 8

E.g. 0 means MOVe, 4 means ADD

# What does a VIP program look like?

- E.g to compute the first 16 Fibonacci numbers.
  - $F(0)=0; F(1)=1; F(n)=F(n-2)+F(n-1)$
  - 0, 1, 1, 2, 3, 5, 8, 13,
- But using a different technique and with Accumulator-based paradigm

Adr	Opcodes	Mnemonic	
030	00C	LDA #0x100	A = Address of F(0)
032	810	LDB #0	B = F(0) = 0
033	821	LDX #1	X = F(1) = 1
034	041	STB [A]	Store F(n)
035	012	LDB X	?
036	424	ADDX [A]	?
037	900	INCA	Next address
038	70C	CMPA #0x110	Limit?
03A	DF9	JNZ -7	Repeat

- Binary to BCD conversion using Decimal ADD op-code

Convert binary number in R0 to 4-digit BCD number in R3:R2  
R3 is most significant part. From 000:000 to 004:095

```
A01      PSH   R1          /* save R1           */
830      MOVS  R3,#0       /* initialize high */
820      MOVS  R2,#0       /* and low parts   */
81C      MOVS  R1,#12      /* 12 shifts       */
930      ROL   R0
082      MOV   AR,R2      /* Decimal double  */
9A2      DADD  R2          /* with carry in   */
028      MOV   R2,AR      /* and carry out   */
083      MOV   AR,R3      /* repeated for   */
9A3      DADD  R3          /* for high part  */
038      MOV   R3,AR
911      DEC   R1          /* loop ?          */
DF7      JNE   R1
A11      POP   R1          /* recover R1      */
A50      RET
```

Implementation of DADD via HDL and 'C' are educational exercises



2<sup>nd</sup> year using TAB with ARM Cortex-M1 in 'C' and assembler



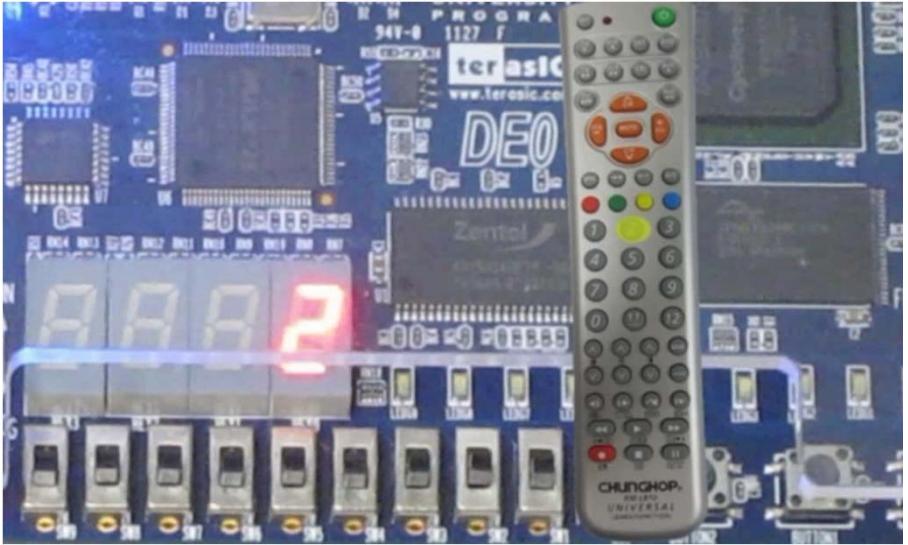
2<sup>nd</sup> year using TAB with ARM Cortex-M1 in 'C' and assembler

# Stills images from video - TAB\_ver2.mp4



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

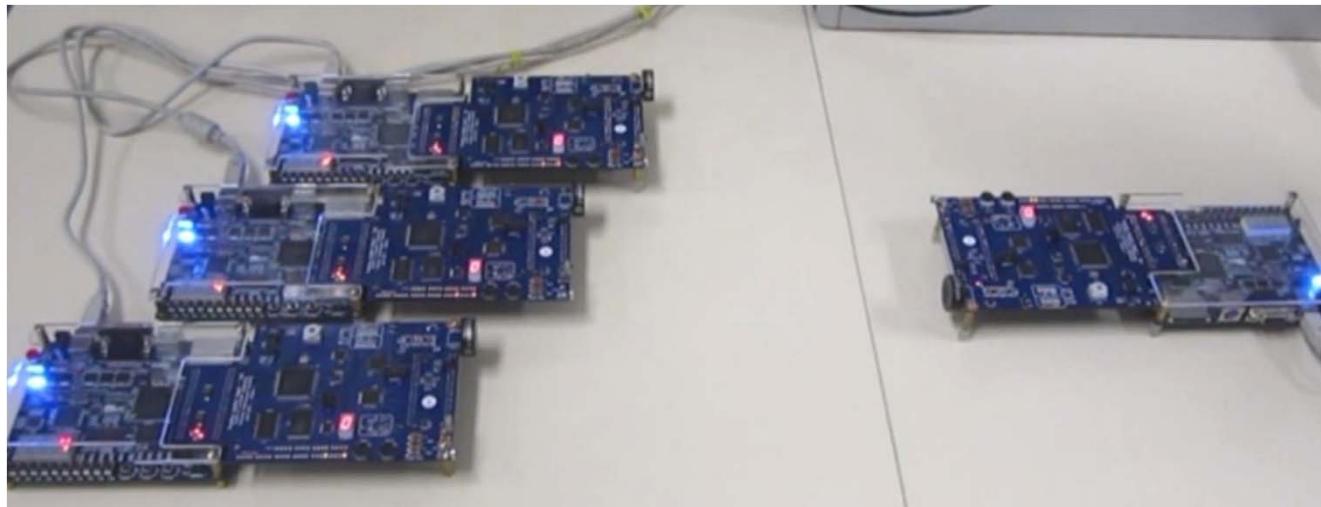
THE UNIVERSITY OF  
WARWICK



Receiving infrared message



Measuring generated waveforms



TAB-to-TAB communication

# Demonstration using µVision4



NANYANG  
TECHNOLOGICAL  
UNIVERSITY

THE UNIVERSITY OF  
WARWICK

The screenshot shows the µVision4 IDE interface. The Registers window on the left displays the core registers (R0-R13, R14-LR, R15-PC, xPSR) and banked memory. The Disassembly window in the center shows the assembly code for the main function. The Browser window on the right lists symbols and their definitions. The bottom status bar indicates the connection is Altera Blaster Cortex Debugge t1: 0.0000000 sec.

```
215: int main (void){  
216:  
217:     int i, j;  
218: //int k;  
219:     char selection;  
220:  
221: //uint8_t num1 = 0, num2 = 9, num3 = 0;  
222: //uint16_t data[3];  
223:  
224:     vector a, m;  
225:     uint32_t slide_sw_stat;  
226}
```

µVision4 Integrated Development  
Environment responding to buttons 1, 2  
and 3 on infra red remote control

TAB Experiments  
r - Record to SRAM  
s - Playback from SRAM  
x - Ramp DAC from external RAM  
z - Read DE0 BUTTON2 Interrupt Count  
Type any character to send. Press 0 to quit.  
IR got 0x758aff00.  
IR got 0x7689ff00.  
IR got 0x7788ff00.

- The Collatz Conjecture states that for any positive integer, n, if it is even then divide by 2, otherwise multiply by 3 and add 1. Eventually n will equal 1.

Typical core 'C' code is

```
void collatzf ( unsigned int n ){  
    while (n > 1) {  
        if ((n & 1)==1) n = n * 3 + 1;  
        else n = n >> 1; }  
    return; }
```

- Determine the initial value of n that produces the maximum number of loops using
  - 12-bit VIP, 32-bit ARMv4, ARM Thumb2 all in assembler
  - 'C' for Cortex-M1 compiled by Keil and IAR Systems
  - Verilog and VHDL targeted to FPGAs and CPLDs

- Would you like to know more?
- Our contact details are

- Dr. David C. Dyer

CPhys MInstP CEng FIET CITP MBCS

School of Engineering, University of Warwick

Coventry, Warwickshire, CV4 7AL

Tel. +44 24 7653015 and e-mail [d.c.dyer@warwick.ac.uk](mailto:d.c.dyer@warwick.ac.uk)

- Mr. Yan Lin Aung

Project Officer, CHiPES Research Centre

Nanyang Technological University

Singapore

Tel. +65 6790 6644 and e-mail [layan@ntu.edu.sg](mailto:layan@ntu.edu.sg)