

# $\mu$ ARM, Interrupt e Device

Marco Melletti  
melletti.marco@gmail.com

March 31, 2015

# Device

- ▶ Dischi
- ▶ Nastri / USB
- ▶ Interfacce di rete
- ▶ Stampanti
- ▶ Terminali

8 x [  ,  ,  ,  ,  ]

# Dove sono i Device

base address:  $(0x40 + dev\_type * 256 + dev\_num * 32) \dots$

arch.h: DEV\_REG\_ADDR(line, dev) !

Device Register Generali:

Field n.	Address	Field Name
0	$(base) + 0x0$	<b>STATUS</b>
1	$(base) + 0x4$	<b>COMMAND</b>
2	$(base) + 0x8$	<b>DATA0</b>
3	$(base) + 0xC$	<b>DATA1</b>

Device Register Terminali:

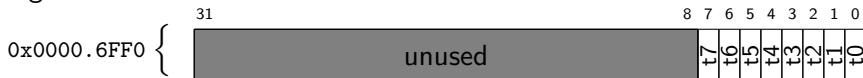
Field n.	Address	Field Name
0	$(base) + 0x0$	<b>RECV_STATUS</b>
1	$(base) + 0x4$	<b>RECV_COMMAND</b>
2	$(base) + 0x8$	<b>TRANSM_STATUS</b>
3	$(base) + 0xC$	<b>TRANSM_COMMAND</b>

# Interrupts

Interrupt vector:

0x6FE0		Disks
0x6FE4		Tapes
0x6FE8		Network
0x6FEC		Printers
0x6FF0		Terminals

e.g. Terminals:



se ( $t_5 == 1$ )  $\Rightarrow$  interrupt pendente sulla linea del terminale 5

## Altre aree interessanti

Address	Function
0x00000020	Installed Devices Vector
0x000002DC	Time of Day (Hi)
0x000002E0	Time of Day (Low)
0x000002E4	Interval Timer
0x00007000	Exception States Vector

Exception States Vector:

(uARMconst.h)

0x7000	INTERRUPT_OLD	INTERRUPT_NEW
0x70B0	TLB_OLD	TLB_NEW
0x7160	PGMTRAP_OLD	PGMTRAP_NEW
0x7210	SYSBP_OLD	SYSBP_NEW

# Lavorare con Interrupt e Device

Proviamo a modificare la funzione `tprint()` in versione "polling" per farle utilizzare gli interrupt...

# Riferimenti

Riferimento principale:

- ▶  $\mu$ ARM Informal Specifications  
(<http://mellotanica.github.io/uARM/uarmdoc.pdf>)

Per i dettagli sui device:

- ▶  $\mu$ MPS Principles of Operation (<http://www.cs.unibo.it/~renzo/so/princOfOperations.pdf>)

Funzioni e strutture di supporto:

- ▶ `arch.h`, `uARMconst.h`, `uARMtypes.h`, `libuarm.h`

# Domande?

Contattatemi tranquillamente via mailing list (SO) o per email:  
`melletti.marco@gmail.com`