

Lập trình nhúng cơ bản

Tuần 2

----Phần thực hành---

Nội dung

- ▶ Cách tổ chức bộ nhớ trong LPC2378
- ▶ Cách thiết lập pin
- ▶ GPIO
- ▶ Debug chương trình

Cách tổ chức bộ nhớ trong LPC2378

- ▶ Bộ xử lý ARM có thể quản lý một không gian địa chỉ bộ nhớ 4 GB.
- ▶ Bảng sau đây thể hiện sự phân bố địa chỉ của vùng bộ nhớ trong các thiết bị ARM:

Địa chỉ vùng nhớ LPC23xx

| Address range | General use | Address range details and description | |
|-------------------------------|--------------------------------------|---|--|
| 0x0000 0000 to 0x3FF FFFF | On-Chip NV Memory and fast I/O | 0x0000 0000 - 0x0007 FFFF | Flash Memory (up to 512 kB) |
| 0x4000 0000 to 0x7FFF FFFF | On-Chip RAM | 0x4000 0000 - 0x4000 7FFF | RAM (up to 32 kB) |
| | | 0x7FD0 0000 - 0x7FD0 1FFF | USB RAM (8 kB) |
| | | 0x7FE0 0000 - 0x7FE0 3FFF | Ethernet RAM (16 kB) |
| 0x8000 0000 to 0xDFFF FFFF | Off-Chip Memory | Two static memory banks, 64 KB each: 0x8000 0000 - 0x8000 FFFF 0x8100 0000 - 0x8100 FFFF | Static memory bank 0, 64 KB Static memory bank 1, 64 KB |
| 0xE000 0000 to 0xFFFF FFFF | APB Peripherals | 0xE000 0000 - 0xE008 FFFF 0xE01F C000 - 0xE01F FFFF | 36 peripheral blocks, 16 kB each (some unused). System Control Block |
| 0xF000 0000 to 0xFFFF FFFF | AHB Peripherals | 0xFFE0 0000 - 0xFFE0 3FFF 0xFFE0 4000 - 0xFFE0 7FFF 0xFFE0 8000 - 0xFFE0 BFFF 0xFFE0 C000 - 0xFFE0 FFFF 0xFFFF F000 - 0xFFFF FFFF | Ethernet Controller General Purpose DMA Controller External Memory Controller (EMC) USB Controller Vectored Interrupt Controller (VIC) |

Vùng nhớ cho các thiết bị
ngoại vi, vd: Battery
RAM: 0xE008 4000

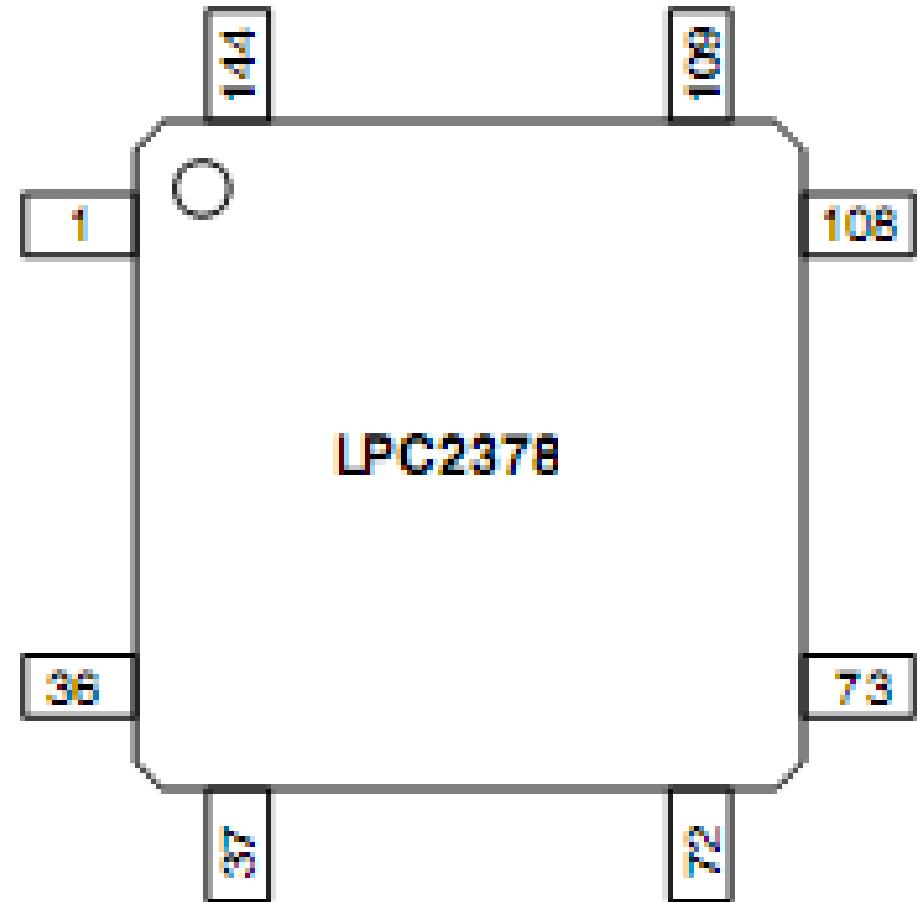
Vùng nhớ nếu có
gắn thêm RAM ngoài

Nội dung

- ▶ Cách tổ chức bộ nhớ trong LPC2378
- ▶ Cách thiết lập pin
- ▶ GPIO
- ▶ Debug chương trình

Cấu hình pin - LPC2378

- ▶ Chip LPC2378 có 144 chân. Với sơ đồ chân như hình vẽ.
- ▶ Mỗi chân tương ứng có những chức năng, nhiệm vụ khác nhau.



Ví dụ

| | | | |
|-----------------|------|-----|--|
| P0[0]/RD1/TXD/ | 66. | I/O | P0[0] — General purpose digital input/output pin. |
| | | I | RD1 — CAN1 receiver input. (LPC2378 only) |
| | | O | TXD3 — Transmitter output for UART3. |
| | | I/O | SDA1 — I ² C1 data input/output (this is not an open-drain pin). |
| P0[1]/TD1/RXD3/ | 67. | I/O | P0[1] — General purpose digital input/output pin. |
| SCL1 | | O | TD1 — CAN1 transmitter output. (LPC2378 only) |
| | | I | RXD3 — Receiver input for UART3. |
| | | I/O | SCL1 — I ² C1 clock input/output (this is not an open-drain pin). |
| P0[2]/TXD0 | 141. | I/O | P0[2] — General purpose digital input/output pin. |
| | | O | TXD0 — Transmitter output for UART0. |
| P0[3]/RXD0 | 142 | I/O | P0[3] — General purpose digital input/output pin. |
| | | I | RXD0 — Receiver input for UART0. |

PINSEL Register

- ▶ Thanh ghi PINSEL (PINSEL0 – PINSEL10) đảm trách việc kết nối giữa các pin và các thiết bị ngoại vi.
- ▶ Các thanh ghi cấu hình này điều khiển thiết lập những chức năng khác nhau, với mỗi 2bit tương ứng ta có 4 trường hợp như sau:

PINSEL Register (tt)

| Giá trị với mỗi 2bits của PINSEL | Chức năng |
|----------------------------------|---|
| 00 | Chức năng mặc định, thường là port GPIO |
| 01 | Chức năng 1 |
| 10 | Chức năng 2 |
| 11 | Chức năng 3 |

Ví dụ

| PINSEL0 | Pin name | Function when 00 | Function when 01 | Function when 10 | Function when 11 | Reset value |
|---------|----------|------------------|------------------|------------------|------------------|-------------|
| 1:0 | P0.0 | GPIO Port 0.0 | RD1 <u>1</u> | TXD3 | SDA1 | 00 |
| 3:2 | P0.1 | GPIO Port 0.1 | TD1 <u>1</u> | RXD3 | SCL1 | 00 |
| 5:4 | P0.2 | GPIO Port 0.2 | TXD0 | Reserved | Reserved | 00 |
| 7:6 | P0.3 | GPIO Port 0.3 | RXD0 | Reserved | Reserved | 00 |
| 9:8 | P0.4 | GPIO Port 0.4 | I2SRX_CLK | RD2 <u>1</u> | CAP2.0 | 00 |
| 11:10 | P0.5 | GPIO Port 0.5 | I2SRX_WS | TD2 <u>1</u> | CAP2.1 | 00 |
| 13:12 | P0.6 | GPIO Port 0.6 | I2SRX_SDA | SSEL1 | MAT2.0 | 00 |
| 15:14 | P0.7 | GPIO Port 0.7 | I2STX_CLK | SCK1 | MAT2.1 | 00 |
| 17:16 | P0.8 | GPIO Port 0.8 | I2STX_WS | MISO1 | MAT2.2 | 00 |
| 19:18 | P0.9 | GPIO Port 0.9 | I2STX_SDA | MOSI1 | MAT2.3 | 00 |
| 21:20 | P0.10 | GPIO Port 0.10 | TXD2 | SDA2 | MAT3.0 | 00 |
| 23:22 | P0.11 | GPIO Port 0.11 | RXD2 | SCL2 | MAT3.1 | 00 |

Địa chỉ thanh ghi PINSEL

- ▶ Mỗi thanh ghi PINSEL có 4 bytes

| Name | Description | Access | Reset Value ^[1] | Address |
|---------|---------------------------------|--------|----------------------------|-------------|
| PINSEL0 | Pin function select register 0. | R/W | 0x0000 0000 | 0xE002 C000 |
| PINSEL1 | Pin function select register 1. | R/W | 0x0000 0000 | 0xE002 C004 |
| PINSEL2 | Pin function select register 2. | R/W | 0x0000 0000 | 0xE002 C008 |
| PINSEL3 | Pin function select register 3. | R/W | 0x0000 0000 | 0xE002 C00C |
| PINSEL4 | Pin function select register 4. | R/W | 0x0000 0000 | 0xE002 C010 |
| PINSEL5 | Pin function select register 5. | R/W | 0x0000 0000 | 0xE002 C014 |
| PINSEL6 | Pin function select register 6. | R/W | 0x0000 0000 | 0xE002 C018 |
| PINSEL7 | Pin function select register 7. | R/W | 0x0000 0000 | 0xE002 C01C |

Nội dung

- ▶ Cách tổ chức bộ nhớ trong LPC2378
- ▶ Cách thiết lập pin
- ▶ **GPIO**
- ▶ Debug chương trình

GPIO

- ▶ General Purpose Input Output
- ▶ Là chân input / output
- ▶ LPC2378 có 104 chân có chức năng GPIO
- ▶ Được thiết lập qua 5 ports 32-bit

| Pin Name | Type |
|-----------|--------|
| P0.[31:0] | Input/ |
| P1.[31:0] | Output |
| P2.[31:0] | |
| P3.[31:0] | |
| P4.[31:0] | |

GPIO (tt)

- ▶ Việc điều khiển trực tiếp trong thanh ghi GPIO chỉ có hiệu quả khi chức năng GPIO được chọn tại các PINSEL
- ▶ Ví dụ:

| PINSEL0 | Pin name | Function when 00 | Function when 01 | Function when 10 | Function when 11 | Reset value |
|---------|----------|------------------|------------------|------------------|------------------|-------------|
| 1:0 | P0.0 | GPIO Port 0.0 | RD1[1] | TXD3 | SDA1 | 00 |
| 3:2 | P0.1 | GPIO Port 0.1 | TD1[1] | RXD3 | SCL1 | 00 |

- ▶ Muốn chọn Port 0.0 và 0.1 là GPIO thì PINSEL0= 0x00; (giá trị mặc định của PINSEL)

GPIO (tt)

- ▶ Low GPIO được điều khiển bởi các thanh ghi: IOPIN, IOSET, IODIR, IOCLR. (chỉ cho Port 0 và Port 1)
- ▶ Fast GPIO được điều khiển bởi các thanh ghi: FIOPIN, FIOSET, FIODIR, FIOCLR , FIOMASK,.

GPIO (tt)

| Thanh ghi | Đặc tả | Địa chỉ |
|-----------|---|--|
| IOPIN | Trạng thái hiện tại của chân GPIO | IO0PIN - 0xE002 8000 IO1PIN - 0xE002 8010 |
| IODIR | Thanh ghi điều khiển các port. ng PIN Giá trị = 1 : output | IO0DIR - 0xE002 8008 IO1DIR - 0xE002 8018 |
| IOSET | Thanh ghi điều khiển trạng thái output, liên kết với thanh ghi IOCLR Giá trị = 1 : mức cao | IO0SET - 0xE002 8004 IO1SET - 0xE002 8014 |
| IOCLR | Thanh ghi điều khiển trạng thái output, liên kết với thanh ghi IOSET Giá trị = 1 : mức thấp | IO0CLR - 0xE002 800C IO1CLR - 0xE002 801C |

GPIO (tt)

| Thanh ghi | Đặc tả | Địa chỉ |
|-----------|--|------------------------|
| FIOPIN | Trạng thái hiện tại của chân GPIO | FIOxPIN - 0x3FFF C0-- |
| FIODIR | Thanh ghi điều khiển các port. ng PIN Giá trị = 1 : output | FIOxDIR - 0x3FFF C0-- |
| FIOSET | Thanh ghi điều khiển trạng thái output, liên kết với thanh ghi FIOCLR Giá trị = 1 : mức cao | FIOxSET - 0x3FFF C0-- |
| FIOCLR | Thanh ghi điều khiển trạng thái output, liên kết với thanh ghi FIOSET Giá trị = 1 : mức thấp | FIOxCLR - 0x3FFF C0-- |
| FIOMASK | 0: các thanh gi trên có hiệu lực 1: các thanh gi trên không có hiệu lực | FIOxMASK - 0x3FFF C0-- |

GPIO (tt) – Ví dụ

0000 0000 0100 0000

- ▶ Cấu hình port 0.6 là output
 - IODIR0 = 0x00000040; //bit thứ 6 được bật lên 1
- ▶ Cấu hình port 0.6 ở mức tín hiệu cao
 - IOSET0 = 0x00000040;
- ▶ Cấu hình port 0.6 ở mức tín hiệu thấp
 - IOCLR0 = 0x00000040;

Ví dụ – config LEDs on MCB2300

- ▶ Các đèn LED trên board được thể hiện qua Port 2.0 đến Port 2.7

```
int main (void) {
```

```
    LED_Init();
```

```
    ....
```

```
void LED_Init(void) {
```

```
    PINSEL10 = 0;      /* Disable ETM interface, enable LEDs */
    FIO2DIR = 0x0000000F; /* P2.0..7 defined as Outputs
    FIO2MASK = 0x00000000;
```

```
}
```

Tắt/mở LEDs

```
void LED_On (unsigned int num) {  
    FIO2SET = (1 << num);  
}
```

Ví dụ: n=2
FIO2SET = (1 <<2)
Tương đương :FIO2SET = 0x04

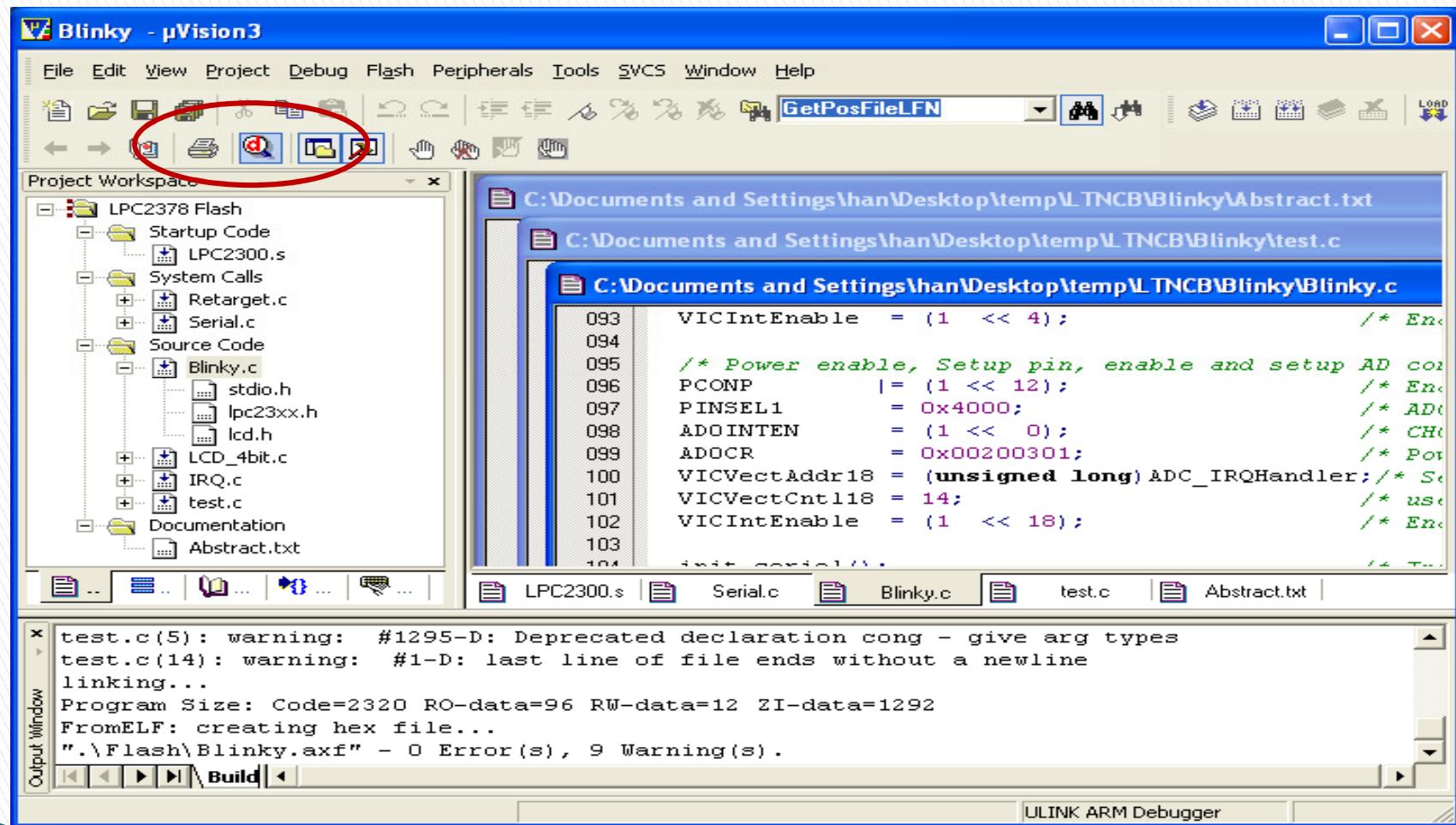
```
void LED_Off (unsigned int num) {  
    FIO2CLR = (1 << num);  
}
```

Nội dung

- ▶ Cách tổ chức bộ nhớ trong LPC2378
- ▶ Cách thiết lập pin
- ▶ GPIO
- ▶ Debug chương trình

Debug

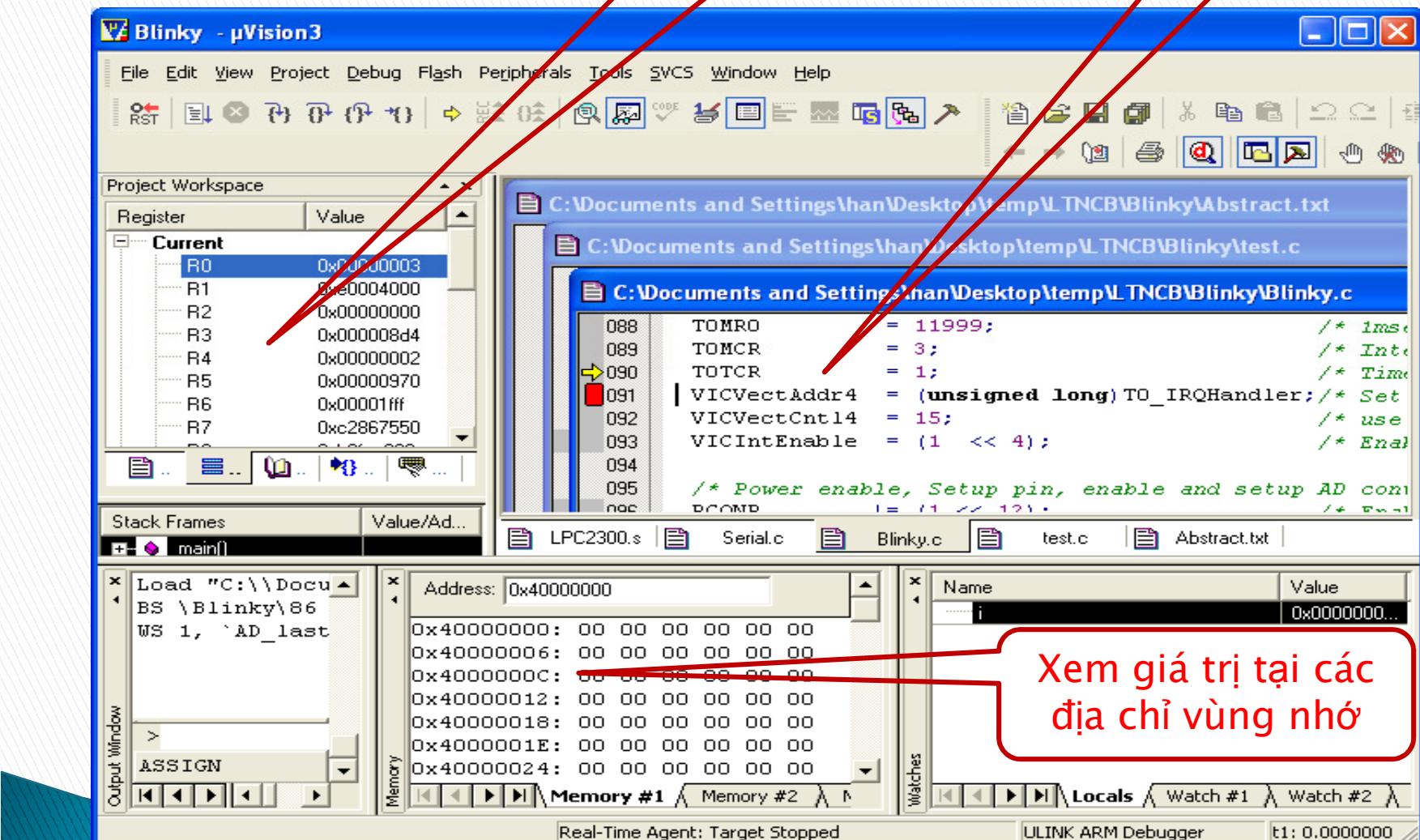
Ctr – F5



Debug (tt)

Giá trị các thanh ghi

Vùng Debug



Bài tập

- ▶ Sử dụng chương trình Blinky: biên dịch và download xuống board
- ▶ Cho thể hiện dòng chữ lên màn hình LCD
- ▶ Bật sáng các đèn LEDs trên board với hiệu ứng theo yêu cầu của GVTH (*)
- ▶ (*): *Bài nộp sau buổi thực hành*

Chúc các bạn làm bài tốt

