

CMPE240  
Sept. 7

Sept. 7.

Note: 1<sup>st</sup> LPC1769 from 2022S  
Semester, Waiting List.  
CANVAS. Scott.

2<sup>nd</sup> LPC1768 pin-to-pin  
Compatible. (Mbed)

a. Step 1. MCUXpresso IDE  
1768 Binary Code.

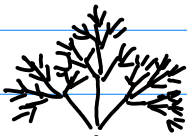
Step 2. "Firmware" Upload  
the binary file to the  
Flash. Need a prob

Step 3. Interactive Debugging.

3<sup>rd</sup> LPC1114 Digi-Key in Stock.  
LPC1114

GPP/SPI, FLASH (ON-chip)

1/8 of the size  
Comparing to LPC1768/9



Homework (0 pt)

1. Form A Team By Wednesday.  
4-Person
2. Select/Finalize your target  
platform. By the end of the week.

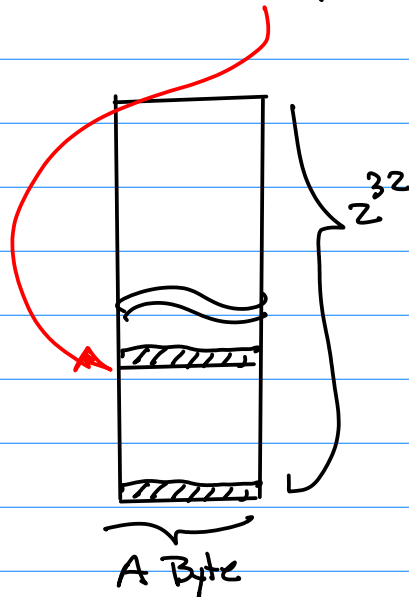
Example: Register File

} Special Purpose Registers  
General Purpose Register

GPx CON  
Prefix 3 Letters  
for Port "x", x=0,1,2,3



GPx CON its address is 32 Bits,  
it maps to the memory  
map.



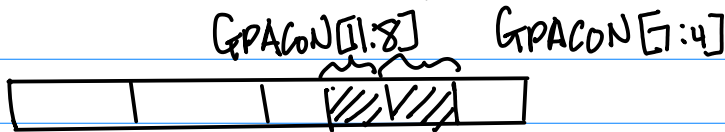
Note: The Task of Init & Config  
Can be realized by using HLL  
(High Level Language), C/C++,  
to deposit A Binary Pattern to that  
Memory Location (Addr. is a pointer)

QmpE240

Sept. 7

For Example for Samsung ARM-11.

Consider:  
Power Up Address + Power Traces.  
Booting.



GPxCON its address is 32 Bits,  
it maps to the memory  
map.

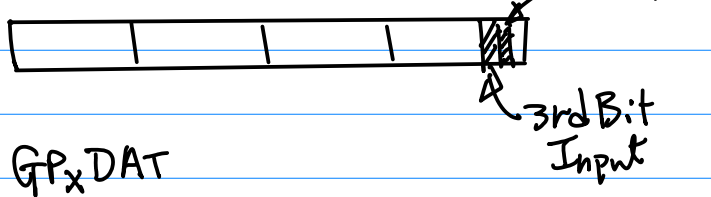
Design Requirements (Spec.)

1. 2nd Bit AS An Output
2. 3rd Bit AS An Input

To perform Init & Config.

GPACON[7:4] = 0001 = 0x1

GPACON[11:8] = 0000 = 0x0



GPACON[3:0] = 0x10

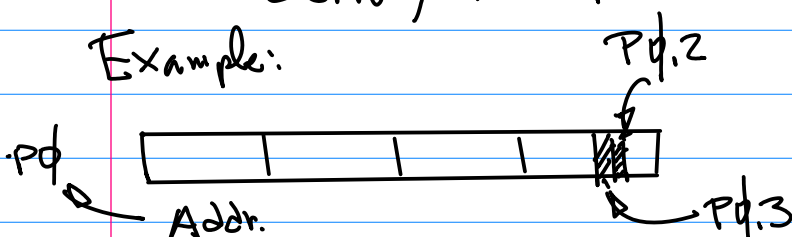
Sept. 12 (Monday)

1. Homework (2pts)
2. Special Purpose Register

Note: Target platform.

LC17ba, LC11C24

Example:



CMPE240  
Sept. 12 (Monday)

Homework Due A week from this Wed.  
Sample Code: On Github.

"Legs" Layout

Note: 1° Connectivity Table

CPU Pin	J2	Note
GPP/PD.2	1A-?	Output $V_{CC} = I \cdot R$
GPP/PD.3	1B-?	Input
GND	1A-?	GND.

Prototypic Board Option 1.

1769

CMOS

$V_{CC} = I \cdot R + V_{LED}$  ... (1)

$I \approx 10mA$

Materials:

- ① LPC 1769 OR 1724
- ② Resistors.  $250 \sim 1k \Omega$
- ③ LED.
- ④

A. Output { "1" ON  
"0" OFF

B. Input Testing { "1"  
"0"

SW

3.3V

$\approx 1.2V_{CC}$

$\sim GPP \dots \sim zip$

Sept. 14 (Wed).

Note: 1° Check Homework Assignment on CANVAS.

Two Options | Prototype Board  
| e-Bay, Board B.

Topics today: IDE

- 1° GPP Software/Program
- 2° 2D Graphics Processing Engine Design.

Example: Set up the Expresso. 2

Key points:

- 1° Make sure Select Target Board LPC1769. (Ref. on github, 35 slides)

2° C/C++ Project Settings. →

"Semi-host"

3° Import LPC1769 patch.

1769 patch.zip

Note: 1° LPC1769 patch is already Config by NXP.

2° prob issue → fix:

Re connect OR Reboot.

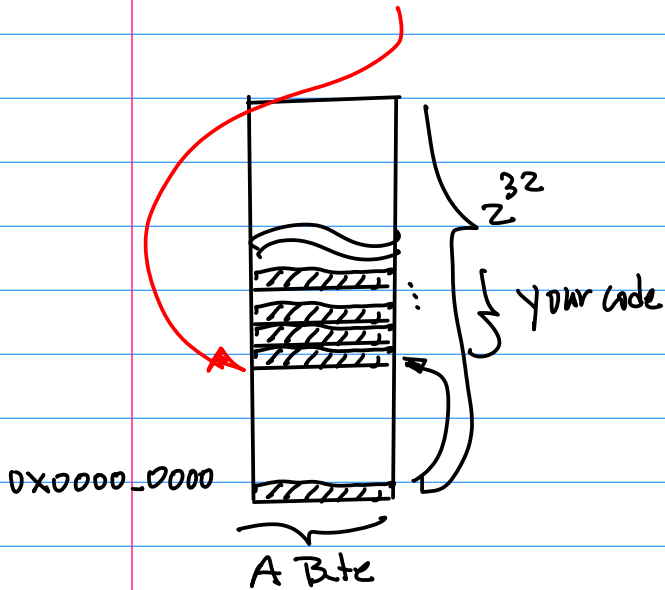
Import GPIO project to your MCU Expresso.

CMPE240

Sept. 14, 22

Run "Debug", Once prob is detected then  
Your Program (Binary)

Note: To Uniquely Define A Line,  
we can add a directional  
vector, Denoted as



Definition 1:

$$\vec{d}(x, y) \triangleq \vec{P}_{\text{end}}(x_{\text{end}}, y_{\text{end}}) - \vec{P}_i(x_i, y_i) \quad \dots (1)$$

Ending pt.                      Starting pt.

Definition 2: (Line Eqn. in Vector Form)

$$\vec{P}(x, y) = \vec{P}_i(x_i, y_i) + \lambda \vec{d}(x, y) \quad \dots (2)$$

$-\infty < \lambda < +\infty$

Consider G.E. Design.

Math. Formulation, for Vector Graphics.

Example.

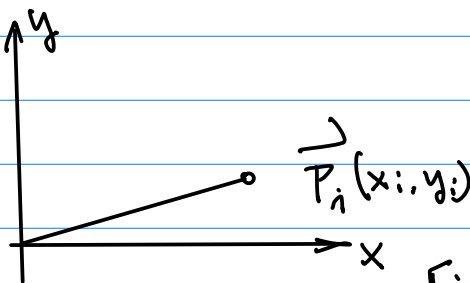


Fig. 1.

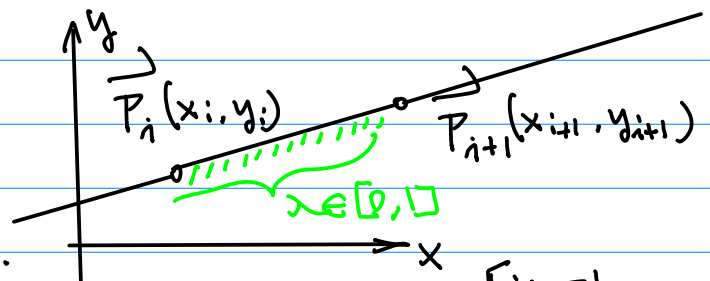


Fig. 2b.

A Point  $\rightarrow \vec{P}(x, y) \rightarrow \vec{P}_i(x_i, y_i)$   
Also, a line  
for  $i = 0, 1, 2, \dots$   
 $\downarrow$   
 $(x_i, y_i)$

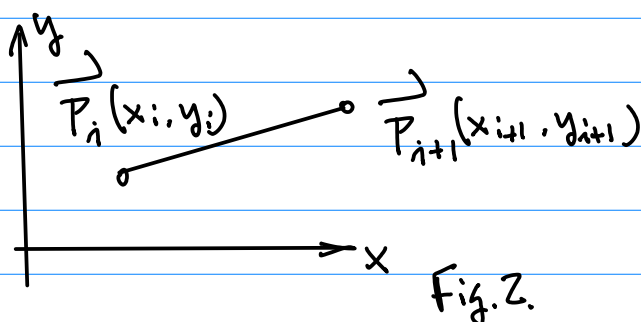


Fig. 2.

Observation 1:

When  $\lambda = 0$ , Eqn (2) gives the  
Starting pt.  $\vec{P}_i(x_i, y_i)$

$\lambda = 1$ ,  $\dots$  Ending point  
 $\vec{P}_{i+1}(x_{i+1}, y_{i+1})$

$0 < \lambda < 1$ ,  $\vec{P}(x, y)$  Any Arbitrary  
pt Between  $\vec{P}_i(x_i, y_i)$   
and  $\vec{P}_{i+1}(x_{i+1}, y_{i+1})$