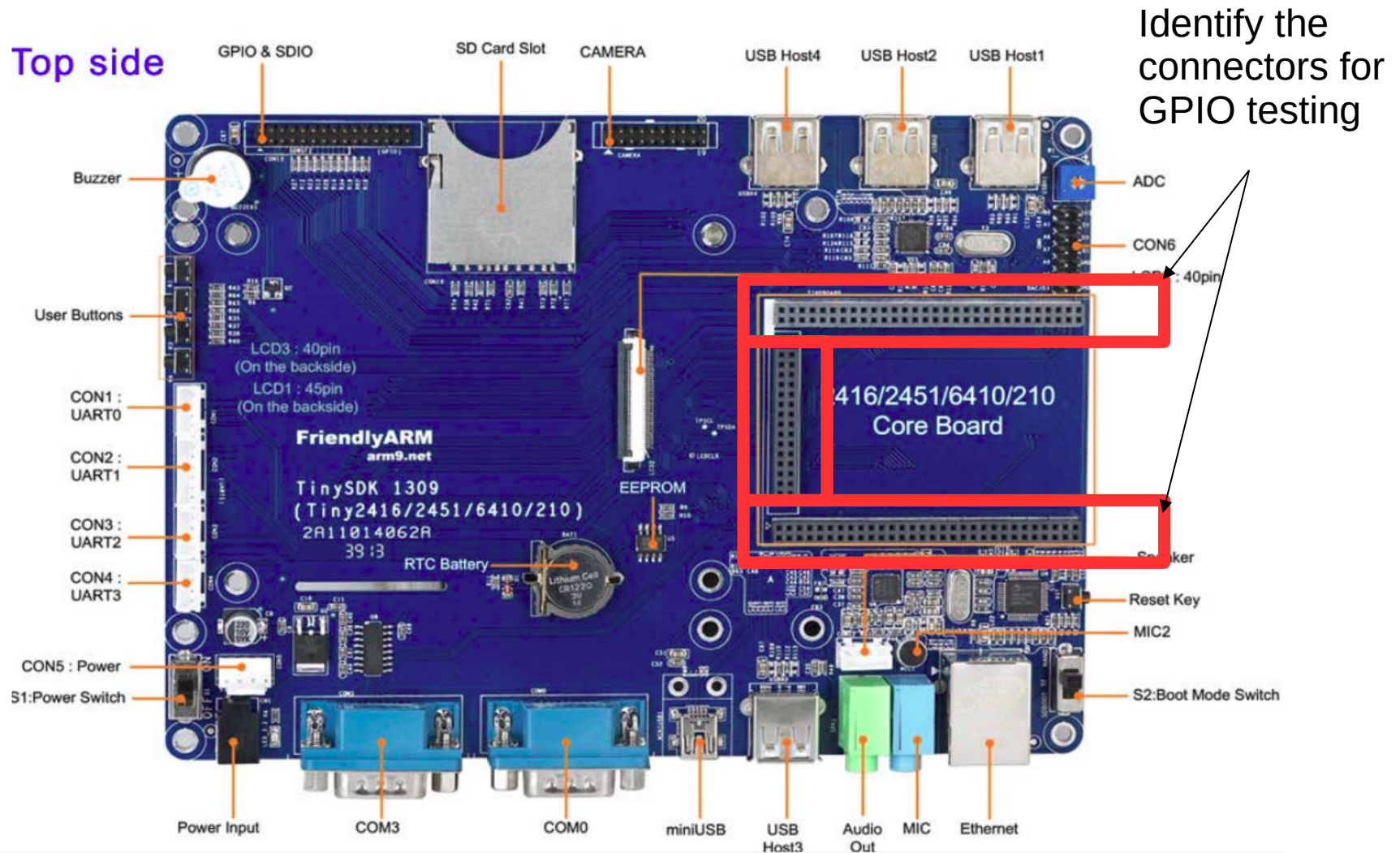


# Connector with CPU GPIO Pins



Note: Check your board revision number, you may have different type of board and different connectors, check the vendor document

# CON1 Connector with GPIO Pins

Table 1. CON1 Pin Assignment

CON1.1	VDD_IO(3.3V)
CON1.3	GPE1

CON1.2	GND
CON1.4	GPE2

Reference: Vendor Document,  
or my class handout, see  
[github.com/hualili](https://github.com/hualili)

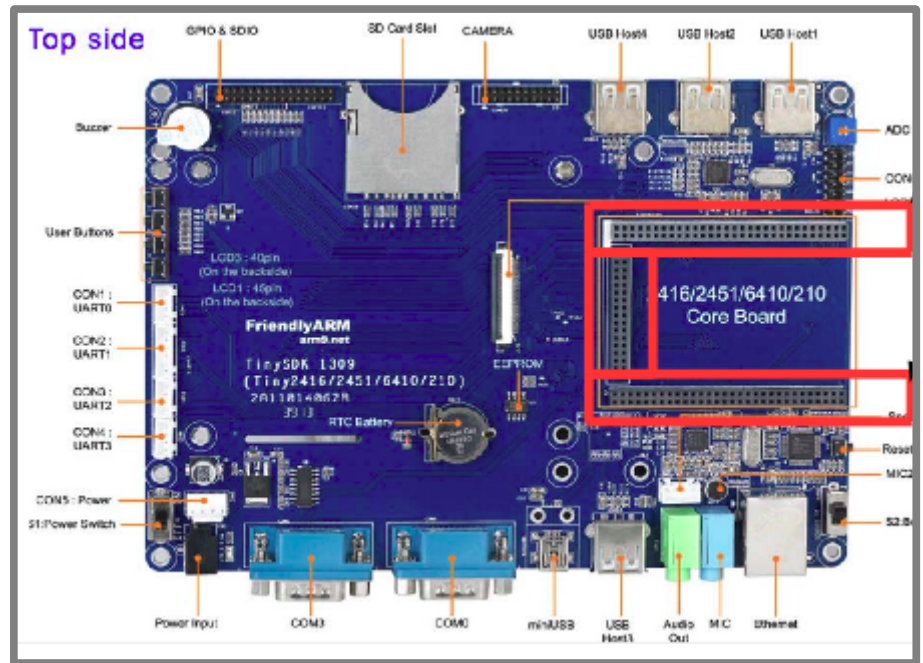
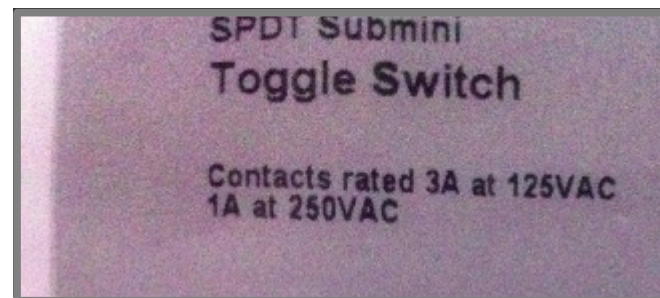


Table 2. GPIO I/O testing

CPU	CON1	Description
GPE1	1.3	output
GPE2	1.4	input



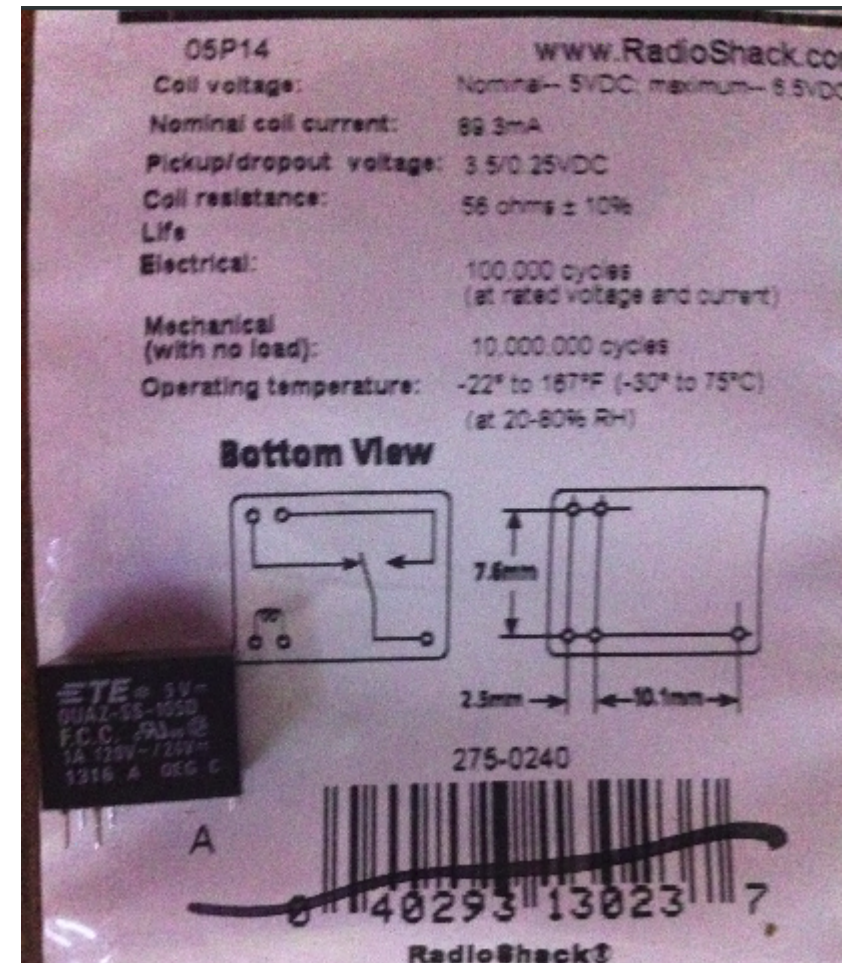
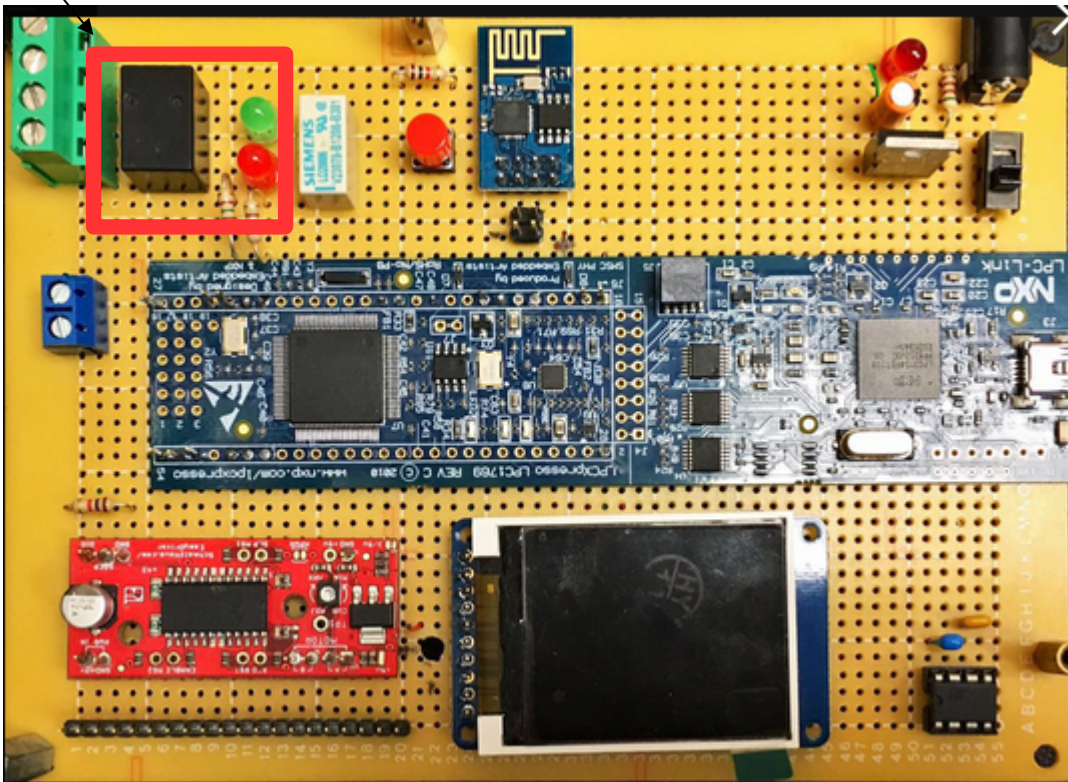
1. SPDT switch for GPIO input testing;
2. Build circuit for input testing and output testing. Output testing will have to allow GPP port to drive LED On/Off.

A Single Pole Double Throw (SPDT) switch is a switch that only has a single input and can connect to and switch between 2 outputs.



# GPIO (GPP) Output with SSR

SSR: Solid State Relay

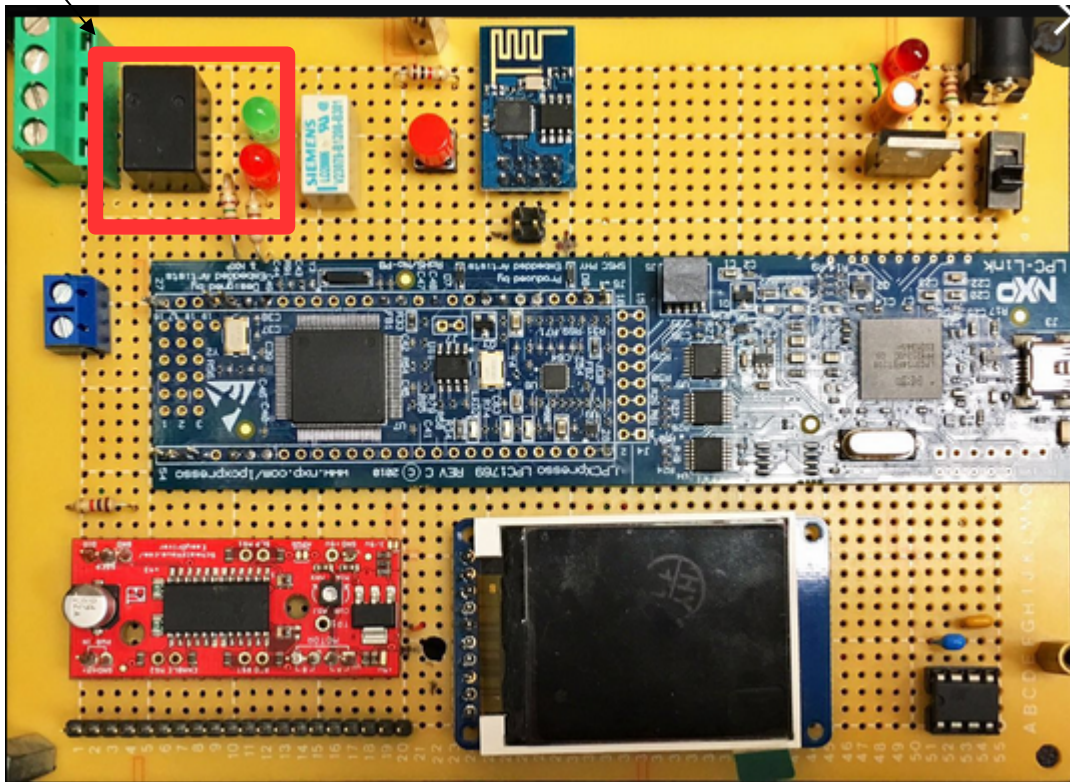


SSR: Solid State Relay

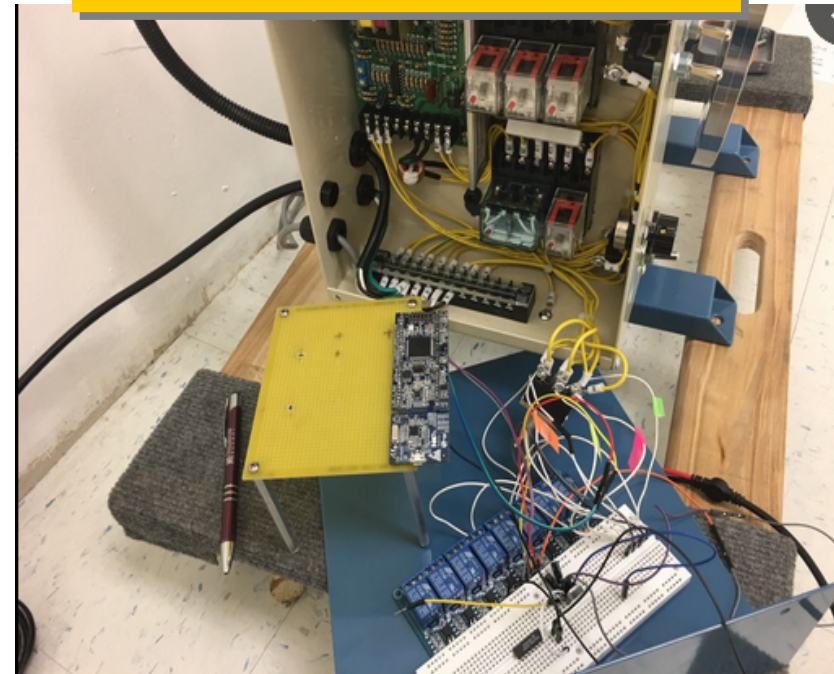


# GPIO (GPP) Output SSR Applications

SSR: Solid State Relay



SMT-POS Positioner



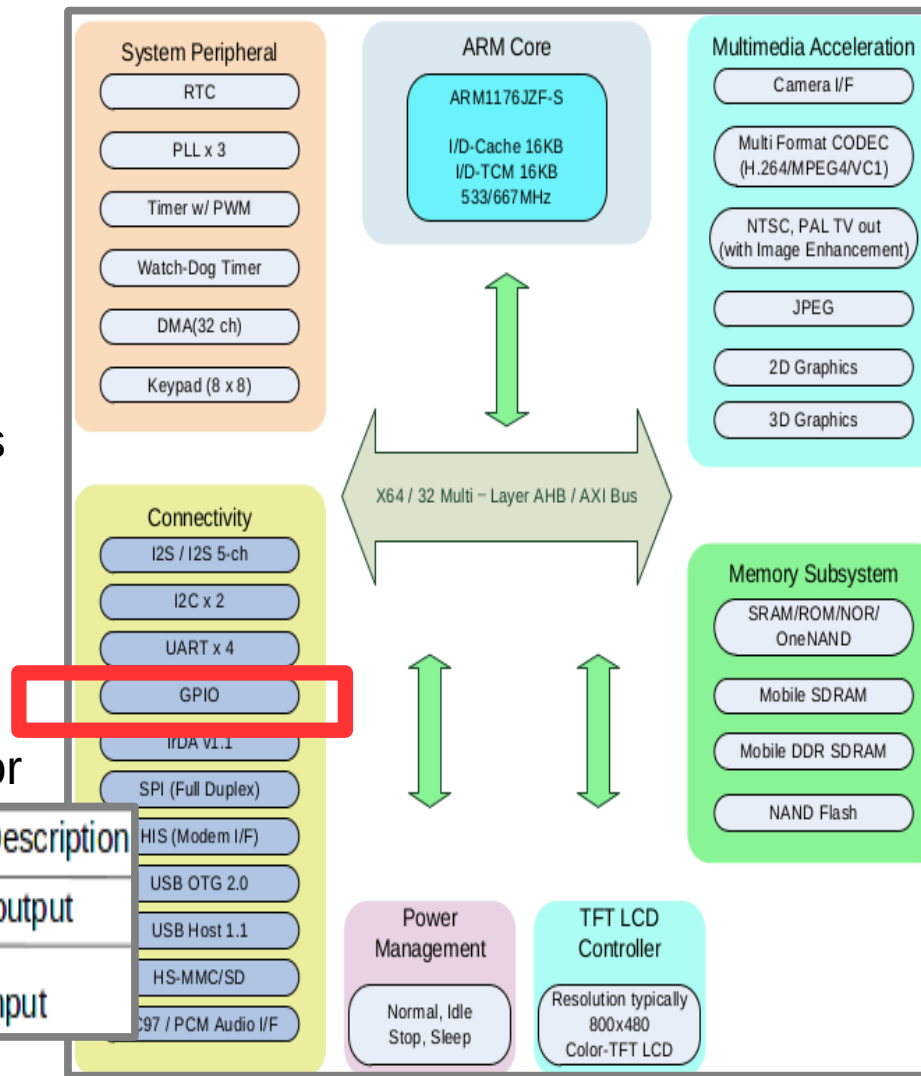
# ARM GPIO (GPP) Controller and SPRs

1. Definition of SPRs:  
Special purpose registers are those to perform init and config functions.
2. 32 bit each with unique address.
3. In IDE (software, e.g., linux-arm-gcc in this case), \*.h file with something looks like the following  
#define SPR 0x2000\_0000  
map the CPU architecture to the arm gcc compiler

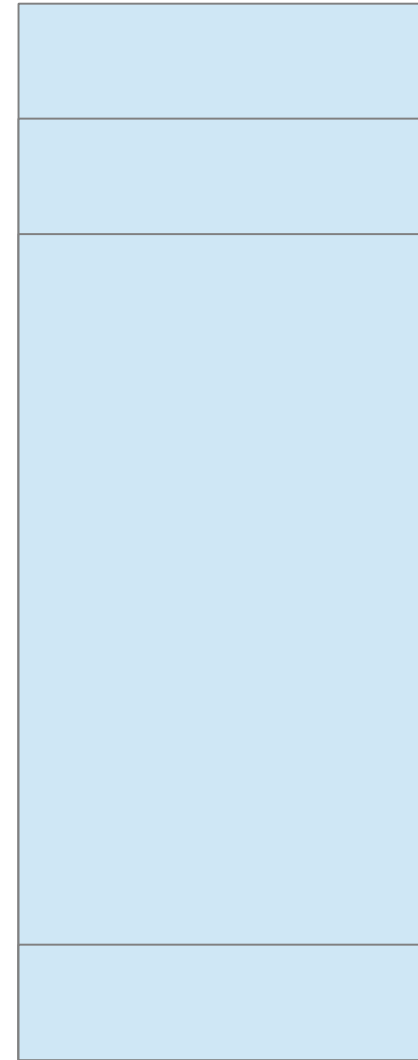
GPA port
GPB port
GPC port
GPD port
GPE port
GPF port

To connector

CPU	CON1	Description
GPE1	1.3	output
GPE2	1.4	input



Memory Map



Ref: pp. 306 datasheet

CPU datasheet Chapter1  
Ref: pp. 61 datasheet

Pwr up address  
0x0000\_0000

# SPR: GPECON Description

Reference: CPU Datasheet, 10.4 REGISTER DESCRIPTION

GPECON  
(Compiler used name:  
GPECON)

GPECON	0x7F008080	Configuration Register
GPEDAT	0x7F008084	Data Register
GPEPUD	0x7F008088	Pull up/down Register

Example: GPE Init &  
Config,

Set GPE1 output, "1"  
GPE2 input, "0"

So the binary  
pattern is 0x10

GPE1	[7:4]	0000 = Input 0010 = PCM EXTCLK[1] 0100 = AC97 RESETn 0110 = Reserved	0001 = Output 0011 = I2S CDCLK[1] 0101 = Reserved 0111 = Reserved
GPE2	[11:8]	0000 = Input 0010 = PCM FSYNC[1] 0100 = AC97 SYNC 0110 = Reserved	0001 = Output 0011 = I2S LRCLK[1] 0101 = Reserved 0111 = Reserved

