

# 8. UART

## 8.1 UART

### 8.1.1 UART

UART (Universal Asynchronous Receiver/Transmitter)

UART (Universal Synchronous/Asynchronous Receiver/Transmitter) 90 (UART serial COI)

### 8.1.2 UART

1) UART (TTL)

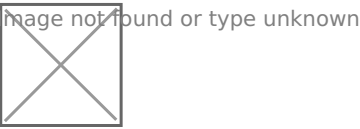
UARTx\_TXD (UARTx\_RXD)

UARTx\_RXD (UARTx\_TXD)

GND

UART (TTL) (TTL) 1 0 (TTL) 2V 2.4V 3.4V (TTL)

2 TTL RS232



RS232	"1"	-5V~-15 V	"0"	+5 V +15 V
-------	-----	-----------	-----	------------

TTL RS232 MAX232 SP:

3 TTL    USB



TTL	USB	D+ D-	NRZI	TTL USB	PL2303 CH340, CP2102
-----	-----	-------	------	---------	----------------------

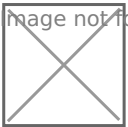
100ASK\_IMX6ULL TTL USB

### 8.1.3 UART

UART

9600,19200,115200

UART	1	+8	+1	+1
------	---	----	----	----



1 +8 +1 +1

ASCII 'A' ASCII 41                      0100 0001                      LSB                      MSB

1) :

1

2) :

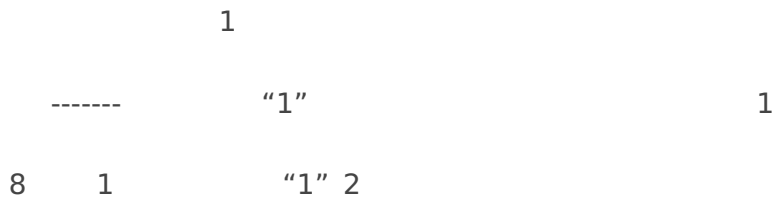
UART	UARTx_TXD	0
------	-----------	---

3) :

5 6 7 8                      8

```
'A' 8      0100 0001      bit 0   1   bit 1   0      bit 7   0
```

4)



5)



## 8.2 IMX6ULL UART

UART:Universal Asynchronous Receiver/Transmitter

### 8.2.1 IMX6ULL UART

IMX6ULL 8 UART 8 UART

- TIA/EIA-232-F 5Mbit/s
- IR Ir-DA 115.2Kbit/s
- 9 RS-485
- 1 2
- 
- 115.2Kbit/s
- 
- SRST\_B

Chapter 55 Universal Asynchronous Receiver/Transmitter(UART)

### 8.2.2 IMX6ULL UART

IMX6ULL 8 UART 17 IO

UART1

image not found or type unknown



9

8.2.2.1 UART1\_URXD

image not found or type unknown



8.2.2.2 UART1\_UTXD

image not found or type unknown



8.2.2.3 UART1\_UCR1

1

DMA

bit0

image not found or type unknown



8.2.2.4 UART1\_UCR2

2

RTS

5

UART

image not found or type unknown



8.2.2.5 UART1\_UCR3

3

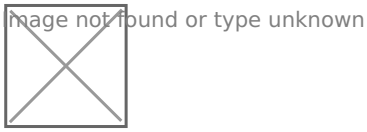
bit2

image not found or type unknown



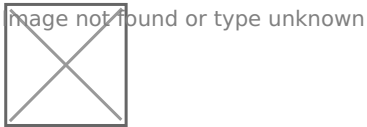
### 8.2.2.6 UART1\_UFCR

UART



### 8.2.2.7 UART1\_USR2

UART



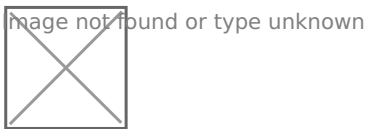
### 8.2.2.8 UART1\_UBIR UART1\_UBMR

UART

Chapter 55 Universal Asynchronous Receiver/Transmitter(I

## 8.3 IMX6ULL UART

### 8.3.1 UART



Chapter 55 Universal Asynchronous Receiver/Transmitter

## 8.3.2 UART1

uart.c

8.3.2.1 1 UART1

## Chapter 18: Clock Controller Module (CCM)

CCM CSCDR1

image not found or type unknown



CCM\_CSCDR1 [UART\_CLK\_SEL] CCM\_CSCDR1 [UART\_CLK\_PODF]

image not found or type unknown



CCM\_CSCDR1

CCM\_CSCDR1 [UART\_CLK\_SEL] 0 CCM\_CSCDR1 [UART\_CLK\_PODF] 0

pll3\_80m UART UART\_CLK\_PODF 1 UART 80MHz

image not found or type unknown



0

② UART CCM\_CCGR5

image not found or type unknown



CCM\_CCGR5 CCM\_CCGR5[CG12] 11

4-1.3 CCM GPIO 11

8.3.2.2 2 GPIO UART1

GPIO UART1 Chapter 32: IOMUX Controller (IOMUXC)

3 GND GPIO UART1\_TXD UART1\_RXD

① UART1\_TX

IOMUXC\_SW\_MUX\_CTL\_PAD\_UART1\_TX\_DATA

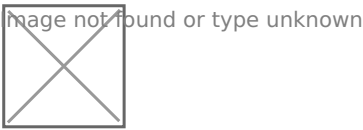
image not found or type unknown



IOMUXC\_SW\_MUX\_CTL\_PAD\_UART1\_TX\_DATA[MUX\_MODE]      0101      0    UART\_TX

②    UART1\_RX

IOMUXC\_SW\_MUX\_CTL\_PAD\_UART1\_RX\_DATA



IOMUXC\_SW\_MUX\_CTL\_PAD\_UART1\_RX\_DATA[MUX\_MODE]      0101      0    UART\_RX

```
IOMUXC_SW_MUX_CTL_PAD_UART1_TX_DATA = (volatile unsigned int *) (0x20E0084);
IOMUXC_SW_MUX_CTL_PAD_UART1_RX_DATA = (volatile unsigned int *) (0x20E0088);

*IOMUXC_SW_MUX_CTL_PAD_UART1_RX_DATA = 0;
*IOMUXC_SW_MUX_CTL_PAD_UART1_TX_DATA = 0;
```

③    UART1\_TX      IOMUXC\_SW\_PAD\_CTL\_PAD\_UART1\_TX\_DATA

4-1.3 IOMUXC      (Mode    )      0x10b0

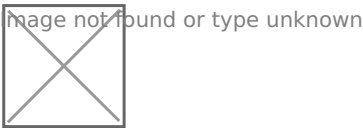
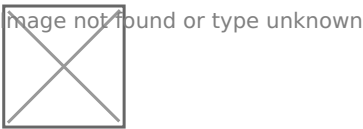
④    UART1\_RX      IOMUXC\_SW\_PAD\_CTL\_PAD\_UART1\_RX\_DATA

4-1.3 IOMUXC      (Mode    )      0x10b0

### 8.3.2.3    3    UART1

UART1

①    UART1\_UCR2 0x2020084



UART1

```
UART1->UCR2 |= (1<<14) | (1<<5) | (1<<2) | (1<<1);
```

[14] 1    RTS

[8] : 0: 0

[6] : 0: 1 0

[5] : 1: 8

[2] : 1:

[1] : 1:

② UART1\_UCR3 0x2020088

image not found or type unknown

image not found or type unknown

[RXDMUXSEL] 1

[2] : 1:IM6ULL UART MUXED 1

```
UART1->UCR3 |= (1<<2);
```

③ UART1\_UFCR 0x2020090

image not found or type unknown

image not found or type unknown

UART1\_UFCR[9-7] UART

101 5

```
UART1->UFCR = 5 << 7; /* Uart clk 80MHz */
```

④ UART1\_UBIR 0x20200A4 , UART1\_UBMR 0x20200A8

UART1\_UBIR UART1\_UBMR



image not found or type unknown

image not found or type unknown

IMX6ULL

image not found or type unknown

a. 115200      BaudRate = 115200

b.UART1      80Mhz Ref Freq = 80000000

c.IMX6ULL       $115200 = 80000000 / (16 * (UBMR + 1) / (UBIR + 1))$

d.      UBMR UBIR

e. UART1\_UBIR = 71    UART1\_UBMR = 3124

```
UART1->UBIR = 71;
UART1->UBMR = 3124;
```

## 8.3.2.4    4    UART1

UART1 UART1\_UCR1(0x2020080)

image not found or type unknown

image not found or type unknown

UART1\_UCR1[0] 1    UART 0    UART

```
Base->UCR1 |= (1 << 0); /*    */
```

## 8.3.3

### 8.3.3.1 1 UART1

UART1\_USR2      UART1      [TXDC]

image not found or type unknown



UART1\_USR2[3] : 0      1

uart.c

```
void PutChar(int c)
{
    while (!((UART1->USR2) & (1<<3))); /*
    UART1->UTXD = (unsigned char)c;
}
```

### 8.3.3.2 2 main

main.c

```
#include "uart.h"

int main()
{
    unsigned char cTestData = 'A'; /*
    Uart_Init();

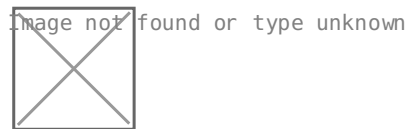
    while(1)
    {
        PutChar(cTestData);
    }

    return 0;
}
```

\*\*      \*\*      Git      NoosProgramProject/(8\_UART      /001\_uart\_txd\_char)

### 8.3.3.3 3 4-1.4

### 8.3.3.4 4 3-1.4



‘A’

## 8.3.4

### 8.3.4.1 1 UART1

UART1

UART1\_USR2

[ROR]



UART1\_USR2[0] : 0 1

uart.c

```
unsigned char GetChar(void) [ ]
{
    [ ]
    [ ]while (!(UART1->USR2 & (1<<0))); /* */
    [ ]return (unsigned char)UART1->URXD;
}
```

### 8.3.4.2 2 main

main

main.c

```
#include "uart.h"

int main()
{
    [ ]unsigned char cTestData ; /* */
    [ ]Uart_Init() [ ] ;

    [ ]while(1)
```

```

{
    cTestData = GetChar() ; // */
    PutChar( cTestData)    ; // */
}

return 0;
}

```

#### 4.4.4 3.4

Image not found or type unknown



100ask.6ull

\*\* \*\* Git NoosProgramProject/(8\_UART /002\_uart\_txd\_char)

## 8.3.5

main

main.c

```

while(1)
{
    cTestData = GetChar() ; // */

    if ( cTestData == ' \r' ) // */ \n\r */
    {
        PutChar(' \n' );
    }

    if ( cTestData == ' \n' )
    {
        PutChar(' \r' );
    }

    PutChar( cTestData)    ; // */ */
}

```

'\r' '\n'

## 8.3.6

### 8.3.6.1 1

\*\* \*\*

my\_printf.c

```
void PutStr(const char *s) {}  
{  
    while (*s)  
    {  
        PutChar(*s);  
        s++;  
    }  
}
```

### 8.3.6.2 2 main

main

main.c

```
PutStr("Hello, world! \n\r"); /* */
```

### 8.3.6.3 3 4-1.4

### 8.3.6.4 4 3-1.4

Hello world

```
Hello, world!
```

## 8.4 printf

## 8.4.1 1.3

① uart.c raise

uart.c

```
119 int raise(int signum) /* raise      */
120 {
121     return 0;
122 }
```

② Makefile

my\_printf.c                      libgcc.a              libgcc.a              Makefile

        "-L"                  "-l"                  -lgcc          "libgcc.a"

Makefile

a. \$(CC) -nostdlib -g -c -o my\_printf.o my\_printf.c

b. \$(LD) -T imx6ull.lds -g start.o uart.o main.o my\_printf.o -o my\_printf.elf

-lgcc -L<libgcc.a    >

\$(LD) -T imx6ull.lds -g start.o uart.o main.o my\_printf.o -o my\_printf.elf **gcc**

**-L/home/book/100ask\_imx6ull-sdk/ToolChain/gcc-linaro-6.2.1\*-2016.11-x86\_64\_arm-linux-gnueabi/lib/gcc/arm-linux-gnueabi/6.2.1**

## 8.4.2

format          ...

vc6.0          stdarg.h

```
typedef char * va_list;

#define _INTSIZEOF(n)   ( (sizeof(n) + sizeof(int) - 1) & ~(sizeof(int) - 1) )
#define va_start(ap, v) ( ap = (va_list)&v + _INTSIZEOF(v) )
#define va_arg(ap, t)   ( *(t *)((ap += _INTSIZEOF(t)) - _INTSIZEOF(t)) )
#define va_end(ap)      ( ap = (va_list)0 )
```

1 \_INTSIZEOF(n)

2 va\_start(ap,v) ap

3 va\_arg(ap,t)

4 va\_end(ap) NULL

printf

int printf(const char \*format, ...) my\_printf

my\_printf.c

```
int printf(const char *fmt, ...)
{
    va_list ap;

    va_start(ap, fmt);
    my_vprintf(fmt, ap);
    va_end(ap);
    return 0;
}
```

## 8.4.3 my\_vprintf(fmt, ap)

int vprintf(const char \*format, va\_list ap) my\_vprintf

my\_printf.c

```
static int my_vprintf(const char *fmt, va_list ap)
{
    char lead=' ';
    int maxwidth=0;

    for(; *fmt != '\0'; fmt++)
    {
        if (*fmt != '%') {
            putchar(*fmt);
            continue;
        }
    }
```

```

    lead=' ';
    maxwidth=0;

    //format : %08d, %8d, %d, %u, %x, %f, %c, %s
    fmt++;
    if(*fmt == '0'){
        lead = '0';
        fmt++;
    }

    while(*fmt >= '0' && *fmt <= '9'){
        maxwidth *=10;
        maxwidth += (*fmt - '0');
        fmt++;
    }

    switch (*fmt) {
    case 'd': out_num(va_arg(ap, int), 10, lead, maxwidth); break;
        case 'o': out_num(va_arg(ap, unsigned int), 8, lead, maxwidth); break;
    case 'u': out_num(va_arg(ap, unsigned int), 10, lead, maxwidth); break;
    case 'x': out_num(va_arg(ap, unsigned int), 16, lead, maxwidth); break;
    case 'c': outc(va_arg(ap, int )); break;
        case 's': outs(va_arg(ap, char *)); break;

    default:
        outc(*fmt);
        break;
    }

    return 0;
}

```

## 8.4.4 out\_c outs out\_num

void PutChar(int c) out\_c outs out\_num outc %c

my\_printf.c



```
static int outc(int c)
{
    _PutChar(c);
    _return 0;
}
```

outs            %s

my\_printf.c

```
static int outs (const char *s)
{
    while (*s != '\0')
        _PutChar(*s++);
    _return 0;
}
```

out\_num            %d %o %u %x

my\_printf.c

```
static int out_num(long n, int base, char lead, int maxwidth)
{
    unsigned long m=0;
    char buf[ MAX_NUMBER_BYTES], *s = buf + sizeof(buf);
    int count=0, i=0;

    *--s = '\0';

    if (n < 0){
        m = -n;
    }
    else{
        m = n;
    }

    do{
        *--s = hex_tab[m%base];
        count++;
    }
```

```

}while ((m /= base) != 0);

if( maxwidth && count < maxwidth){
    for (i=maxwidth - count; i; i--){
        *--s = lead;
    }

    if (n < 0)
        *--s = '-';

    return outs(s);
}

```

## 8.4.5 my\_printf\_test

my\_printf.c

```

int my_printf_test(void)
{
    printf("This is www.100ask.org   my_printf test\n\r") ;
    printf("test char           =%C,%c\n\r", 'A','a') ;
    printf("test decimal number =%d\n\r",    123456) ;
    printf("test decimal number =%d\n\r",    -123456) ;
    printf("test hex      number =0x%x\n\r",  0x55aa55aa) ;
    printf("test string      =%s\n\r",      "www.100ask.org") ;
    printf("num=%08d\n\r",    12345);
    printf("num=%8d\n\r",     12345);
    printf("num=0x%08x\n\r",  0x12345);
    printf("num=0x%8x\n\r",   0x12345);
    printf("num=0x%02x\n\r",  0x1);
    printf("num=0x%2x\n\r",   0x1);

    printf("num=%05d\n\r",  0x1);
    printf("num=%5d\n\r",   0x1);

    return 0;
}

```

## 8.4.6 main

main      8.3.5      my\_printf\_test

main.c

```
#include "my_printf.h"
#include "uart.h"
int main()
{
    Uart_Init();
    my_printf_test();
    return 0;
}
```

Git    NoosProgramProject/(8\_UART    /005\_myprintf\_test)

8.4.6.1    1      4-1.4

8.4.6.2    2      3-1.4

```
This is www.100ask.org    my_printf test
test char                =A, a
test decimal number =123456
test decimal number =-123456
test hex        number =0x55aa55aa
test string               =www.100ask.org
num=00012345
num=    12345
num=0x00012345
num=0x    12345
num=0x01
num=0x 1
num=00001
num=    1
```

\*\*    \*\*

Git    NoosProgramProject/(8\_UART    /005\_printf\_test

Revision #1

Created 3 March 2022 02:41:02 by

Updated 3 March 2022 02:41:16 by