UNIT
14

Shift Register(74HC595)

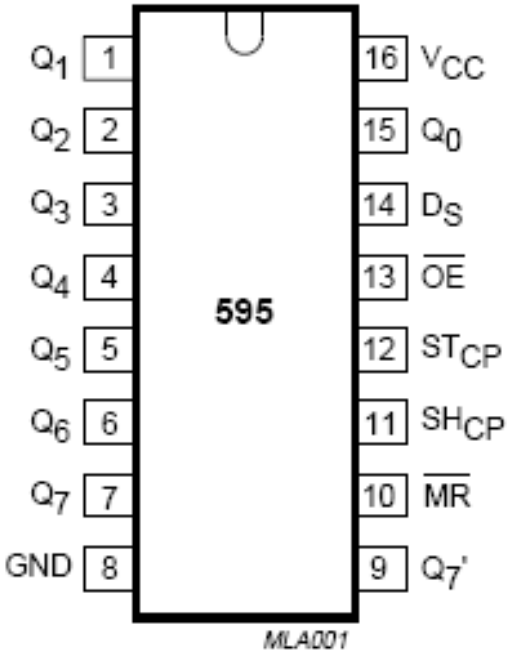
로봇SW 교육원
최상훈(shchoi82@gmail.com)

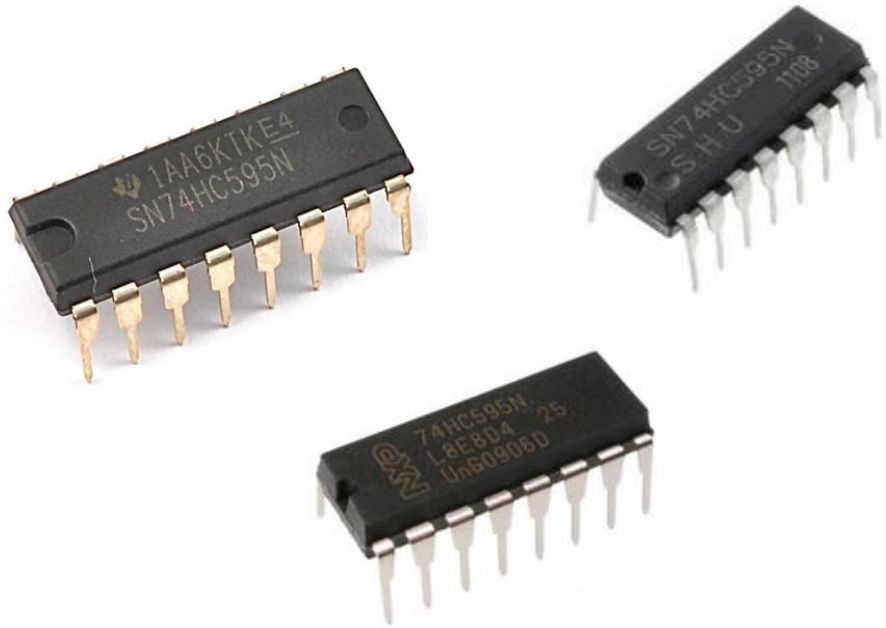
학습 목표

- 74HC595 Shift Register **제어**

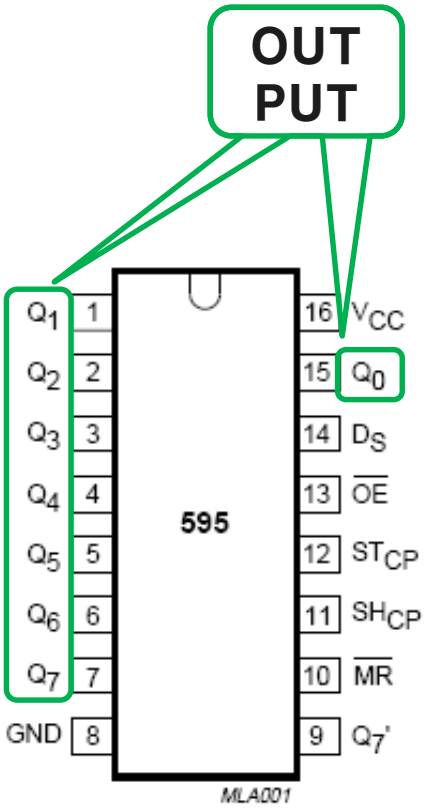
74HC595 Shift Register

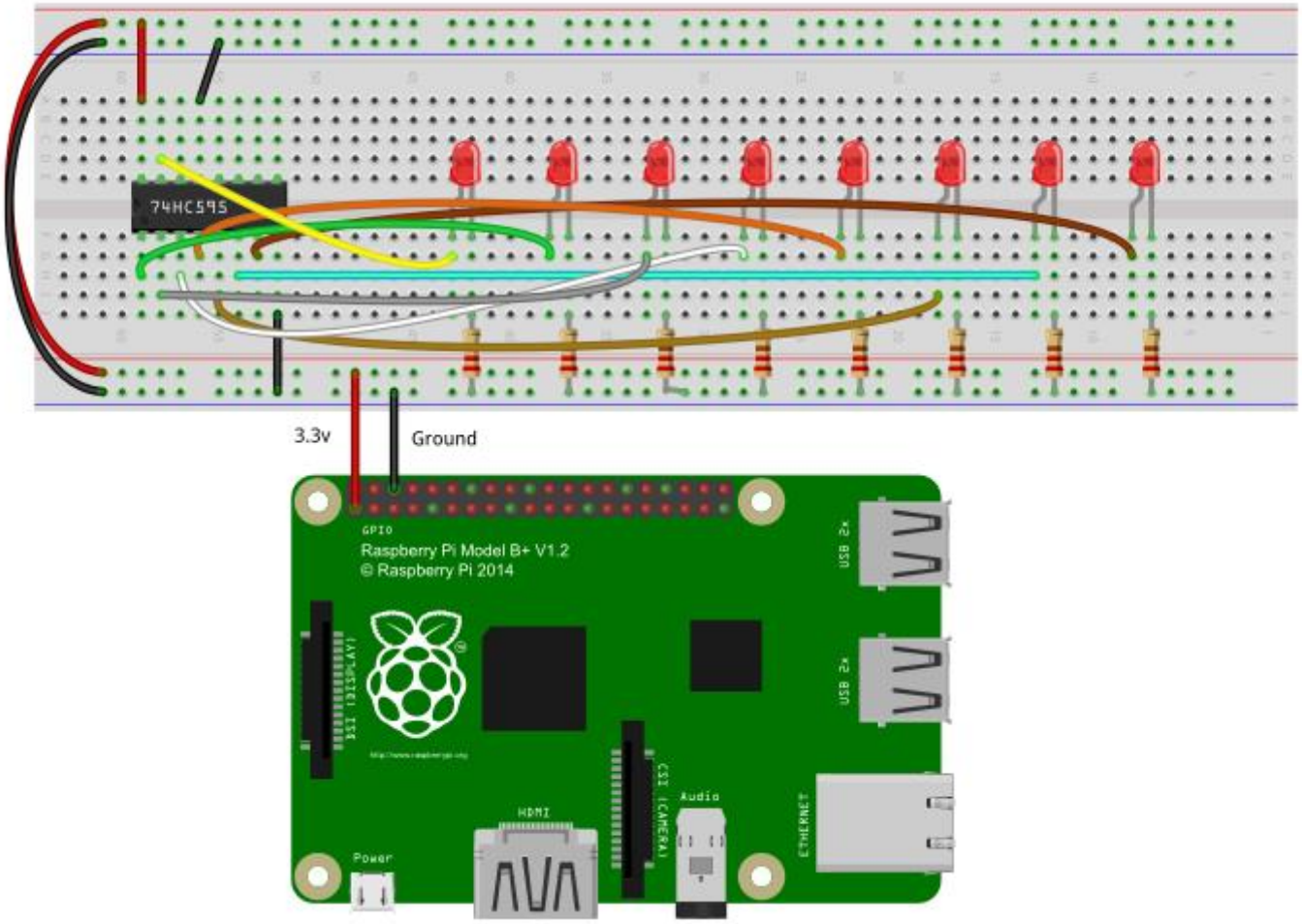
- **쉬프트 레지스터**
  - **플립플롭**(flip-flop) **또는 래치**(latch)
  - 1**비트 값을 저장할 수 있는 기본 회로**
  - **플립플롭 여러 개를 일렬로 연결**
  - **직렬-병렬변환**(SIPO : Serial In Parallel Out) : **직렬 입력으로 다중 출력**



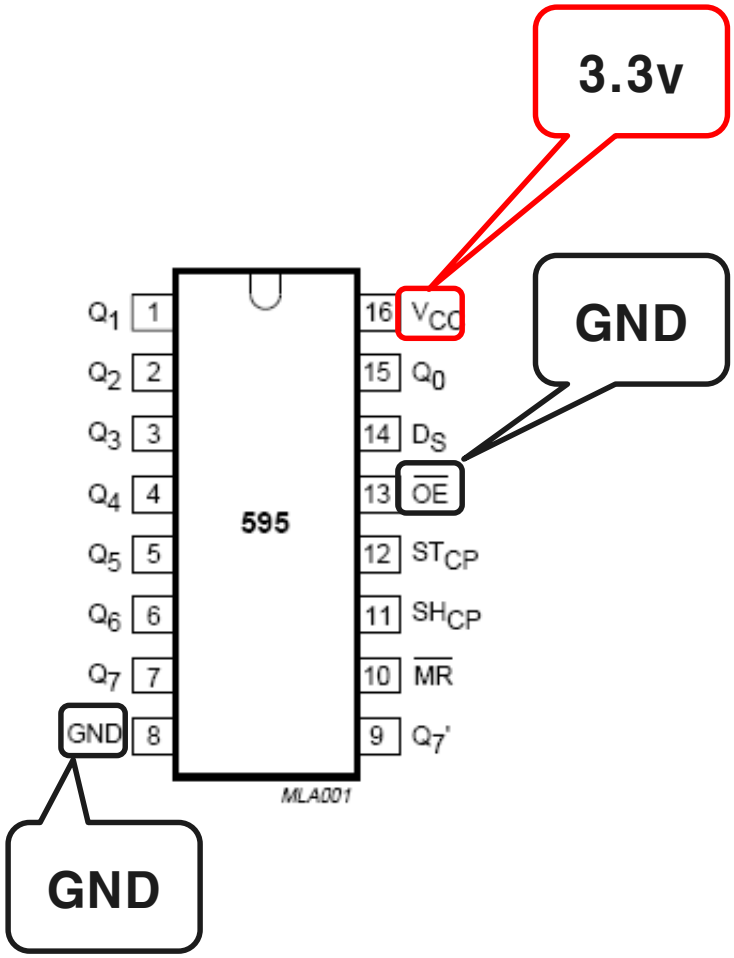


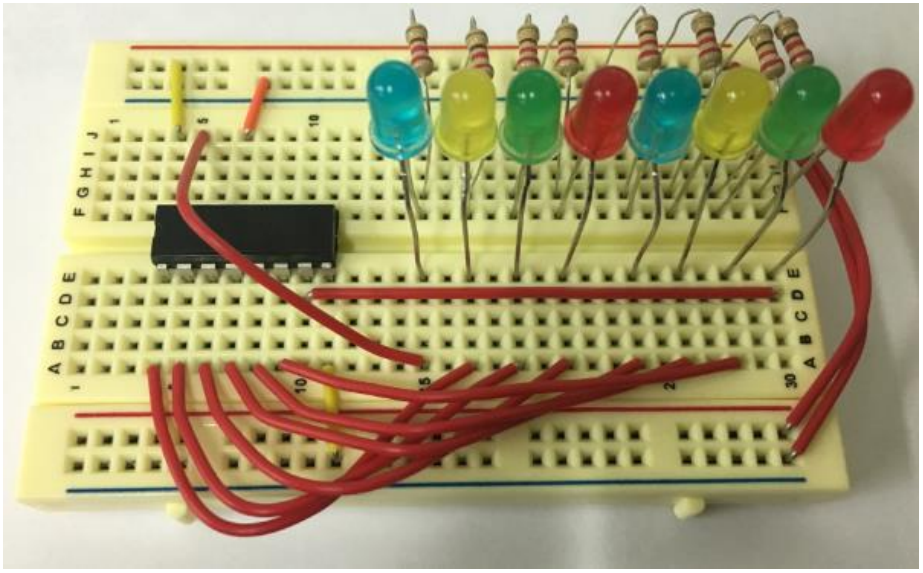
74HC595 Shift Register

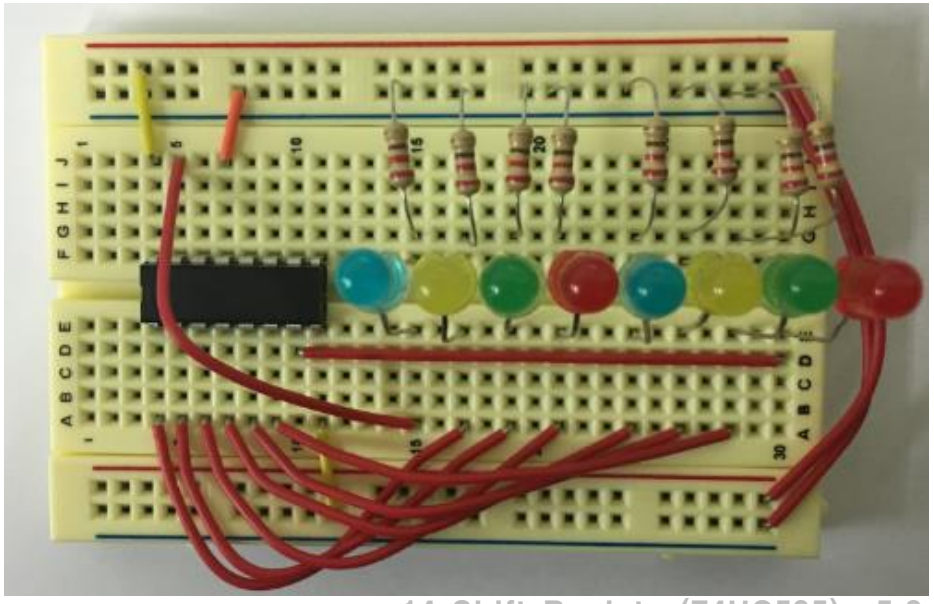




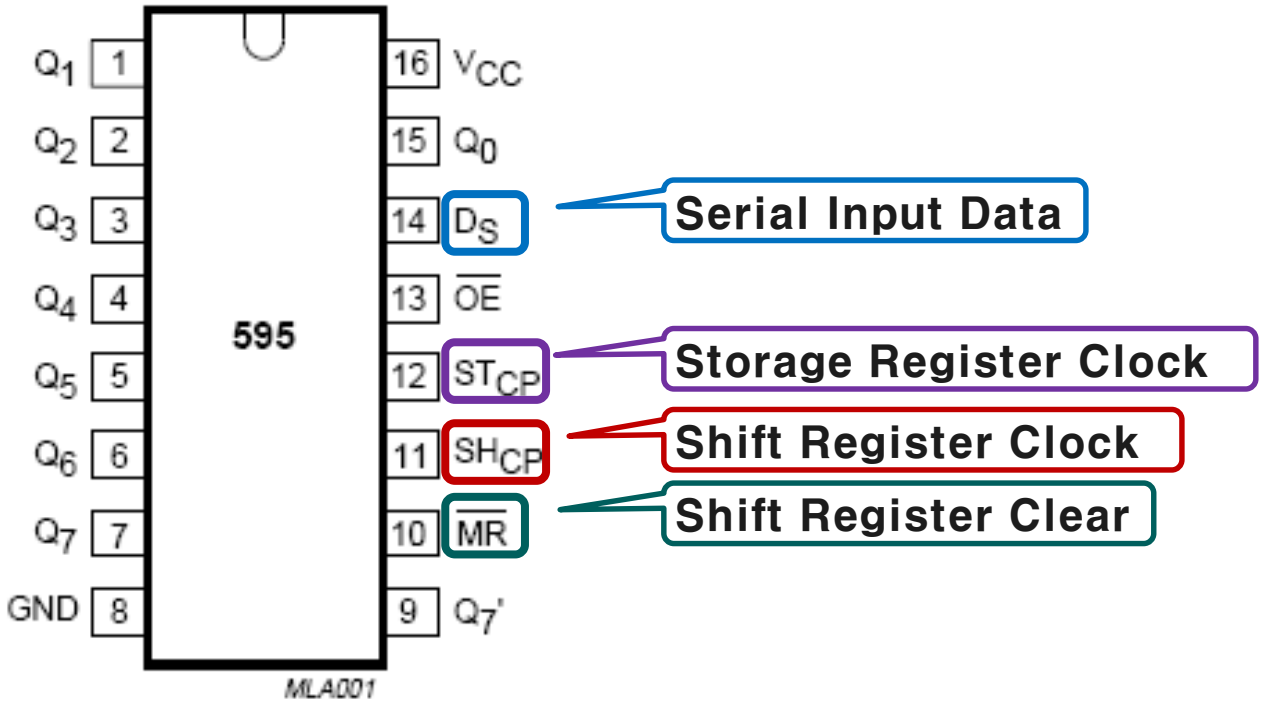
74HC595 Shift Register







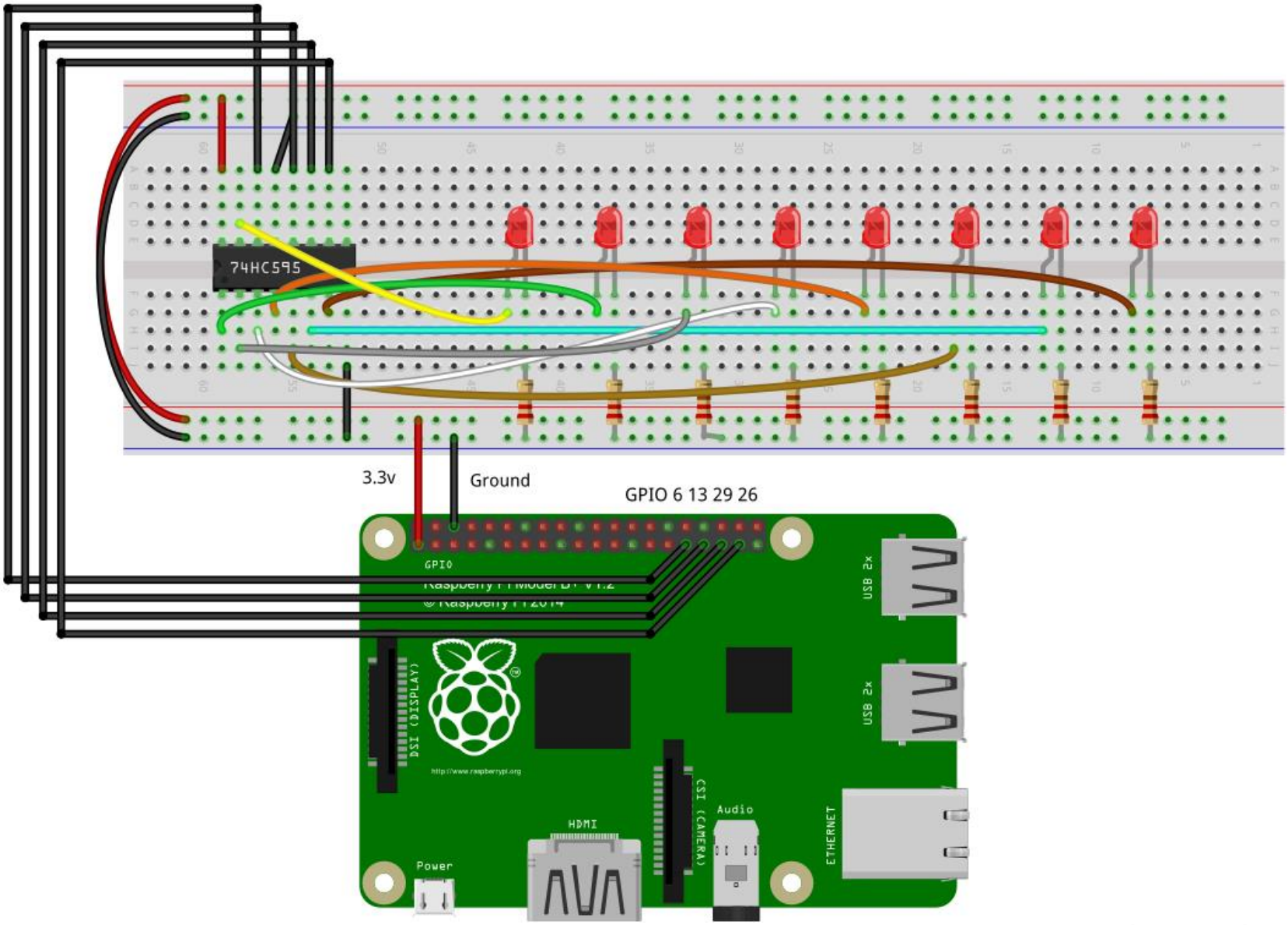
74HC595 Shift Register



L : low , H : high, ▲ : low → high

입 력				기 능
Data	STR_CLK	SHR_CLK	SHR_CLR	
			L	쉬프트 레지스터 초기화
L		▲	H	쉬프트 레지스터 shift, Q0에 low 입력
H		▲	H	쉬프트 레지스터 shift, Q0에 high 입력
	▲		H	쉬프트 레지스터 → 스토리지 레지스터

74HC595 Shift Register



gpio유틸(wiringPi)을 이용한 제어

- 핀모드 output으로 설정

```
$ gpio -g mode 6 output
$ gpio -g mode 13 output
$ gpio -g mode 19 output
$ gpio -g mode 26 output
```

Serial Input Data

Storage Register Clock

Shift Register Clock

Shift Register Clear

```
$ gpio -g write 26 1
```

- 데이터 설정

```
$ gpio -g write 6 1
```

또는

```
$ gpio -g write 6 0
```

- 스토리지 레지스터 클럭

```
$ gpio -g write 13 0
$ gpio -g write 13 1
```

- 쉬프트 레지스터 클럭

```
$ gpio -g write 19 0
$ gpio -g write 19 1
```

- 쉬프트 레지스터 초기화

```
$ gpio -g write 26 0
$ gpio -g write 26 1
```

실습1-1


```
pi@robotcode ~ $ gpio -g mode 6 output
pi@robotcode ~ $ gpio -g mode 13 output
pi@robotcode ~ $ gpio -g mode 19 output
pi@robotcode ~ $ gpio -g mode 26 output
pi@robotcode ~ $ gpio -g write 26 1
pi@robotcode ~ $
pi@robotcode ~ $ gpio -g write 6 1
pi@robotcode ~ $ gpio -g write 19 0
pi@robotcode ~ $ gpio -g write 19 1
pi@robotcode ~ $ gpio -g write 13 0
pi@robotcode ~ $ gpio -g write 13 1
pi@robotcode ~ $
pi@robotcode ~ $ gpio -g write 6 0
pi@robotcode ~ $ gpio -g write 19 0
pi@robotcode ~ $ gpio -g write 19 1
pi@robotcode ~ $ gpio -g write 19 0
pi@robotcode ~ $ gpio -g write 19 1
pi@robotcode ~ $ gpio -g write 6 1
pi@robotcode ~ $ gpio -g write 19 0
pi@robotcode ~ $ gpio -g write 19 1
pi@robotcode ~ $ gpio -g write 13 0
pi@robotcode ~ $ gpio -g write 13 1
pi@robotcode ~ $
```

실습1-2


```
pi@robotcode ~ $ gpio -g write 26 0
pi@robotcode ~ $ gpio -g write 26 1
pi@robotcode ~ $ gpio -g write 13 0
pi@robotcode ~ $ gpio -g write 13 1
pi@robotcode ~ $
```

실습2-1

파일명 : 74hc595_ex1.c

```
#include <stdio.h>
#include <wiringPi.h>

#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

int main(void)
{
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        return 1;
    }

    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);

    digitalWrite(SHR_CLEAR, 1);

    // serial data
    digitalWrite(SDATA, 1);
```

실습2-1

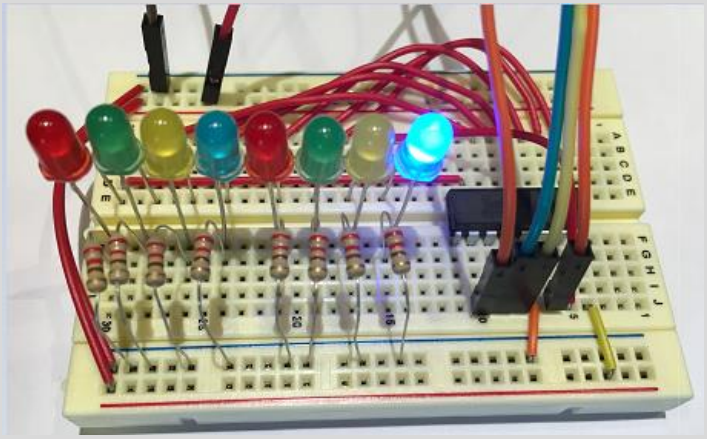
파일명 : 74hc595_ex1.c

```
 // shift
    digitalWrite(SHR_CLK, 0);
    digitalWrite(SHR_CLK, 1);

    // letch
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);

    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi 74hc595_ex1.c -o 74hc595_ex1
pi@robotcode ~ $ sudo ./74hc595_ex1
pi@robotcode ~ $ sudo ./74hc595_ex1
pi@robotcode ~ $
```



실습3-1

파일명 : 74hc595_ex2.c

```
#include <stdio.h>
#include <wiringPi.h>

#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

int
main(void)
{
    int i;
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        return 1;
    }

    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);

    digitalWrite(SHR_CLEAR, 1);

    digitalWrite(SDATA, 1);      // serial data
```

실습3-1

파일명 : 74hc595_ex2.c

```
    for(i = 0 ; i < 8 ; i++){
        digitalWrite(SHR_CLK, 0);
        digitalWrite(SHR_CLK, 1);    // shift
        digitalWrite(STR_CLK, 0);
        digitalWrite(STR_CLK, 1);    // latch
        delay(1000);
    }

    return 1;
}
```

```
pi@robotcode ~ $ gpio -g write 26 0
pi@robotcode ~ $ gpio -g write 26 1
pi@robotcode ~ $ gpio -g write 13 0
pi@robotcode ~ $ gpio -g write 13 1
pi@robotcode ~ $
pi@robotcode ~ $ gcc -Wall -W -lwiringPi 74hc595_ex2.c -o 74hc595_ex2
pi@robotcode ~ $ sudo ./74hc595_ex2
pi@robotcode ~ $
```

https://youtu.be/cDeNeKAATeM

실습4-1

파일명 : 74hc595_ex3.c

```
#include <stdio.h>
#include <wiringPi.h>

#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

int
main(void)
{
    int i;
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        return 1;
    }

    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);
    digitalWrite(SHR_CLEAR, 0);
    digitalWrite(SHR_CLEAR, 1);
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
```

실습4-2

파일명 : 74hc595_ex3.c

```
    for(i = 0 ; i < 32 ; i++){
        digitalWrite(SDATA, i%2);    // serial data
        digitalWrite(SHR_CLK, 0);
        digitalWrite(SHR_CLK, 1);    // shift
        digitalWrite(STR_CLK, 0);
        digitalWrite(STR_CLK, 1);    // latch
        delay(400);
    }

    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi 74hc595_ex3.c -o 74hc595_ex3
pi@robotcode ~ $ sudo ./74hc595_ex3
pi@robotcode ~ $
pi@robotcode ~ $ gpio -g write 26 0
pi@robotcode ~ $ gpio -g write 26 1
pi@robotcode ~ $ gpio -g write 13 0
pi@robotcode ~ $ gpio -g write 13 1
```

https://youtu.be/OvMgZbxR1go

실습5-1

파일명 : 74hc595_ex4.c

```
#include <stdio.h>
#include <wiringPi.h>

#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

void allclear(void)
{
    digitalWrite(SHR_CLEAR, 0);
    digitalWrite(SHR_CLEAR, 1);
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
}

void set(int index)
{
    int i;
    digitalWrite(SDATA, 1);
    digitalWrite(SHR_CLK, 0);
    digitalWrite(SHR_CLK, 1);
    digitalWrite(SDATA, 0);
    for(i = 0 ; i < index ; i++){
        digitalWrite(SHR_CLK, 0);
        digitalWrite(SHR_CLK, 1);    // shift
    }
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
}
```

실습5-2

파일명 : 74hc595_ex4.c

```
int
main(void)
{
    int i;
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        return 1;
    }

    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);

    for(i = 0 ; i < 100 ; i++){
        clear();
        set(i%8);
        delay(100);
    }

    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi 74hc595_ex4.c -o 74hc595_ex4
pi@robotcode ~ $ sudo ./74hc595_ex4
pi@robotcode ~ $
```

https://youtu.be/Nh_mDUAh-9Y

stdint.h의 자료형

- C99에서 도입된 자료형
- 크기와 sign을 정확히 표시
- 헤더 파일 stdint.h

Specific integral type limits

Specifier	Signing	Bits	Bytes	Minimum Value	Maximum Value
`int8_t`	Signed	8	1	-2^7 which equals −128	$2^7 - 1$ which is equal to 127
`uint8_t`	Unsigned	8	1	0	$2^8 - 1$ which equals 255
`int16_t`	Signed	16	2	-2^{15} which equals −32,768	$2^{15} - 1$ which equals 32,767
`uint16_t`	Unsigned	16	2	0	$2^{16} - 1$ which equals 65,535
`int32_t`	Signed	32	4	-2^{31} which equals −2,147,483,648	$2^{31} - 1$ which equals 2,147,483,647
`uint32_t`	Unsigned	32	4	0	$2^{32} - 1$ which equals 4,294,967,295
`int64_t`	Signed	64	8	-2^{63} which equals −9,223,372,036,854,775,808	$2^{63} - 1$ which equals 9,223,372,036,854,775,807
`uint64_t`	Unsigned	64	8	0	$2^{64} - 1$ which equals 18,446,744,073,709,551,615

실습6-1

파일명 : 74hc595_ex5.c

```
#include <stdio.h>
#include <wiringPi.h>
#include <stdint.h>
#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

void allclear(void)
{
    digitalWrite(SHR_CLEAR, 0);
    digitalWrite(SHR_CLEAR, 1);
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
}

// unsinged 8bit int
void set8(uint8_t value)
{
    int i;

    for(i = 0 ; i < 8 ; i++){
        int mask = 0b1 << i;
        if((value & mask) == 0)
            digitalWrite(SDATA, 0);
        else
            digitalWrite(SDATA, 1);
        digitalWrite(SHR_CLK, 0); //
        digitalWrite(SHR_CLK, 1); //
    }
```

실습6-2

파일명 : 74hc595_ex5.c

```
 // letch
    digitalWrite(STR_CLK, 0); //
    digitalWrite(STR_CLK, 1); //
}

int
main(void)
{
    int i;
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        return 1;
    }
    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);
    allclear();
    for(i = 0 ; i < 10 ; i++){
        set8(0b10101010);          delay(100);
        set8(0b01010101);          delay(100);
    }
    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi 74hc595_ex5.c -o 74hc595_ex5
pi@robotcode ~ $ sudo ./74hc595_ex5
pi@robotcode ~ $
```

https://youtu.be/NI8-LFA_noI

실습7-1

파일명 : 74hc595_ex6.c

```
#include <stdio.h>
#include <stdlib.h>
#include <wiringPi.h>
#include <stdint.h>
#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

void allclear(void);

void init(void)
{
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        exit(1);
    }
    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);
    allclear();
}
void allclear(void)
{
    digitalWrite(SHR_CLEAR, 0);
    digitalWrite(SHR_CLEAR, 1);
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
}
```

실습7-2

파일명 : 74hc595_ex6.c

```
void set8(uint8_t value) // unsinged 8bit int
{
    int i;
    for(i = 0 ; i < 8 ; i++){
        int mask = 0b1 << i;
        if((value & mask) == 0)
            digitalWrite(SDATA, 0);
        else
            digitalWrite(SDATA, 1);
        digitalWrite(SHR_CLK, 0); //
        digitalWrite(SHR_CLK, 1); //
    }

    // letch
    digitalWrite(STR_CLK, 0); //
    digitalWrite(STR_CLK, 1); //
}

int main(void)
{
    int i;
    uint8_t arr[] = {   0b10000000,
                        0b01000000,
                        0b00100000,
                        0b00010000,
                        0b00001000,
                        0b00000100,
                        0b00000010,
                        0b00000001};
```

실습7-3

파일명 : 74hc595_ex6.c

```
    init();

    for(i = 0 ; i < sizeof(arr) ; i++){
        set8(arr[i]);
        delay(300);
    }
    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi ./74hc595_ex6.c -o ./74hc595_ex6
./74hc595_ex6.c: In function ‘main’:
./74hc595_ex6.c:67:16: warning: comparison between signed and unsigned
integer expressions [-Wsign-compare]
pi@robotcode ~ $ sudo ./74hc595_ex6
pi@robotcode ~ $
```

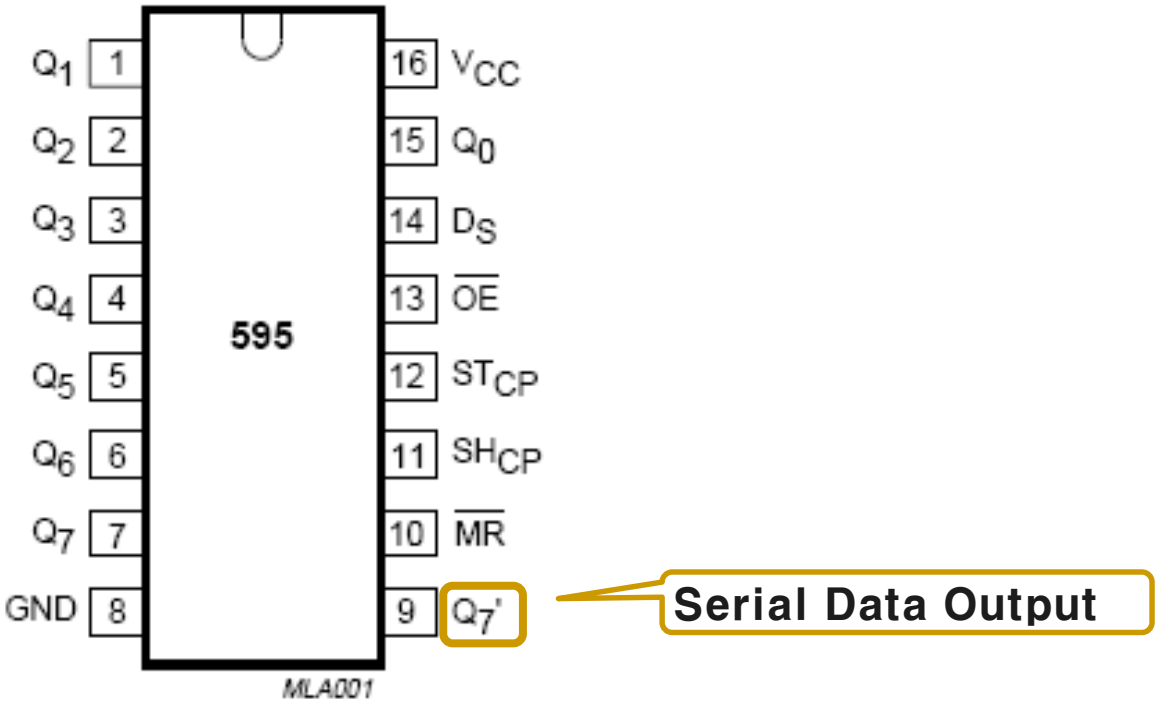
https://youtu.be/RqWMDruAHRE

미션

- https://youtu.be/n4VjnhAGMQ8

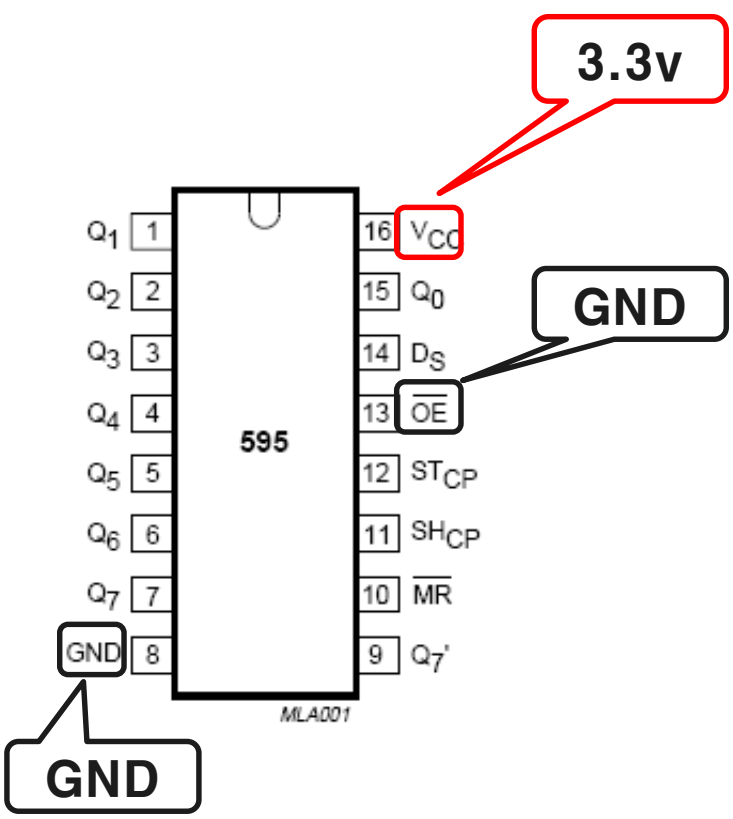
Dual 74HC595

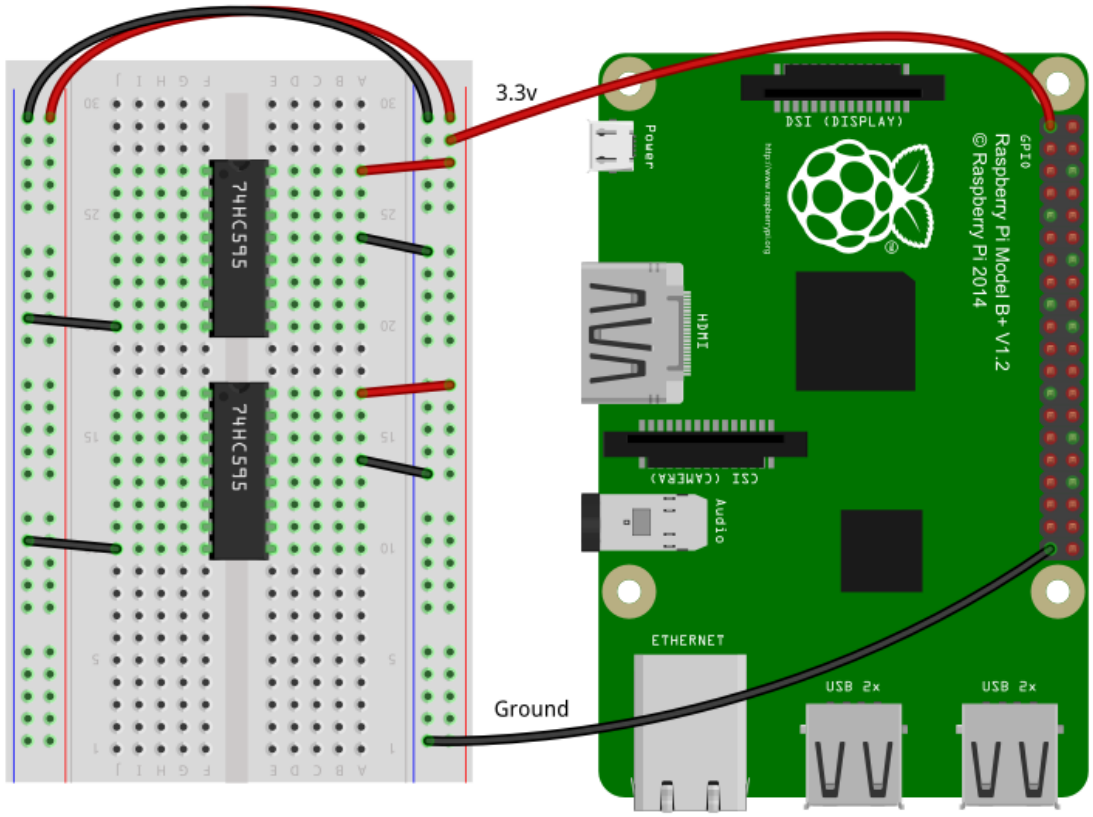
- 2**개의** 74HC595 **연결**
- 16Bit Shift Register
- Serial Data Output : Q7**의 데이터 쉬프트**



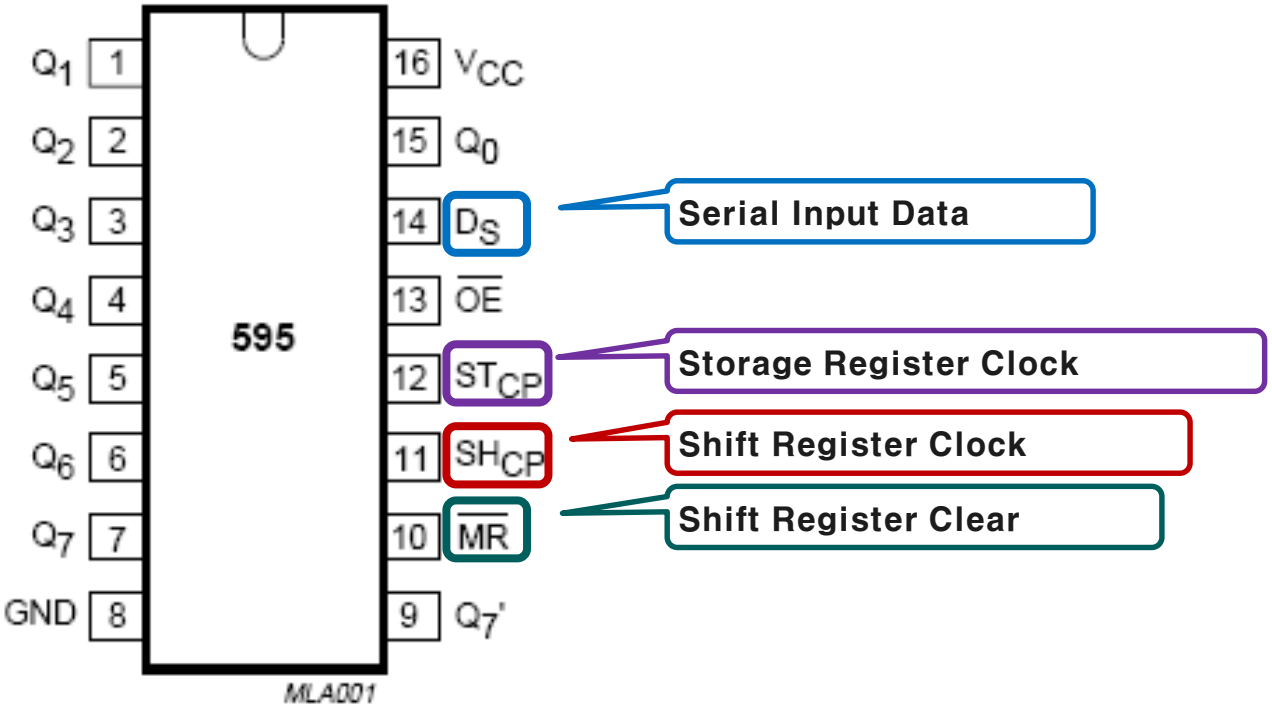
실습8-1: Dual 74HC595

- 2개의 74HC595 준비
- Vcc 에 3.3v 연결
- Output Enable, GND에 Ground연결

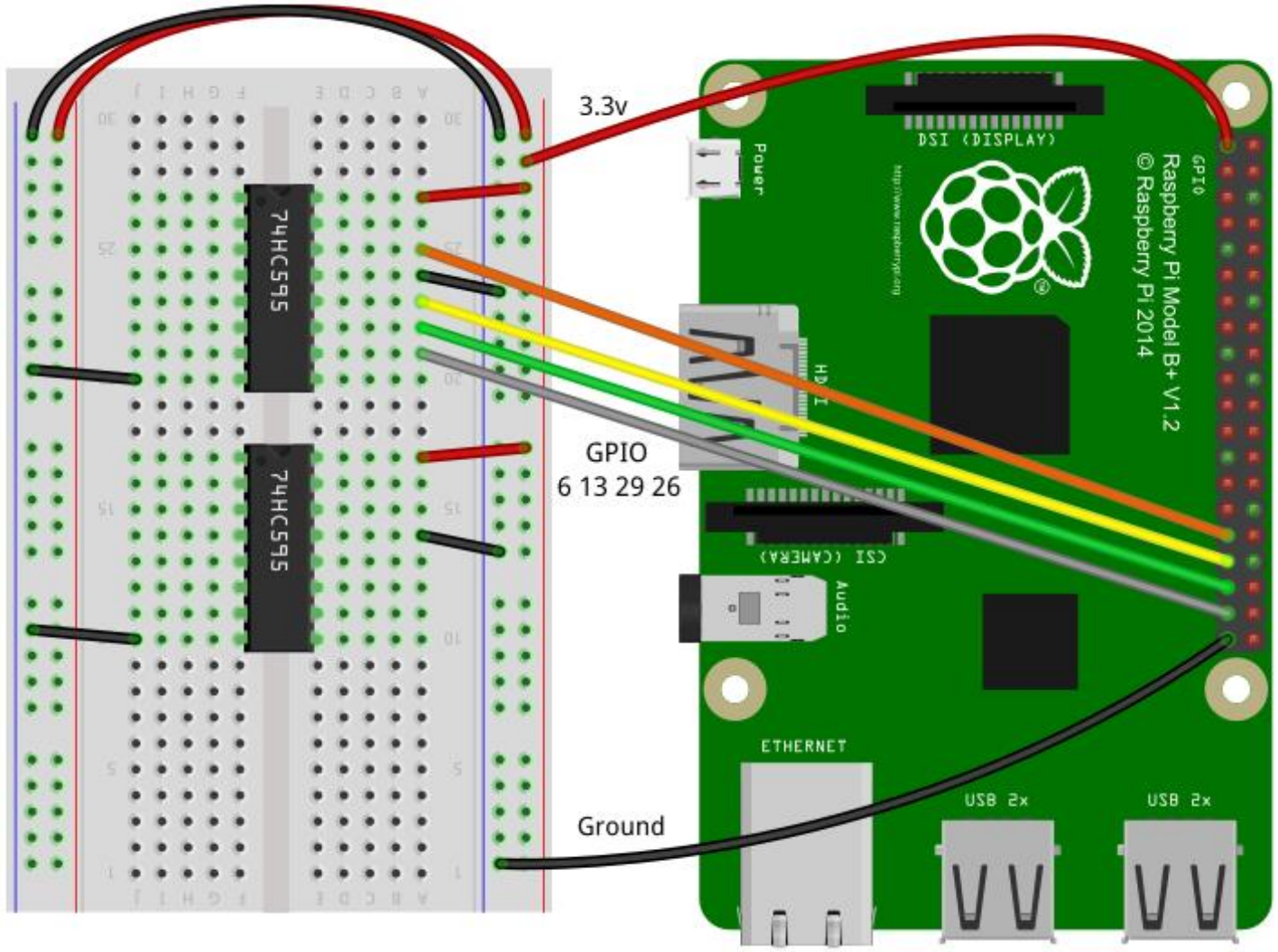




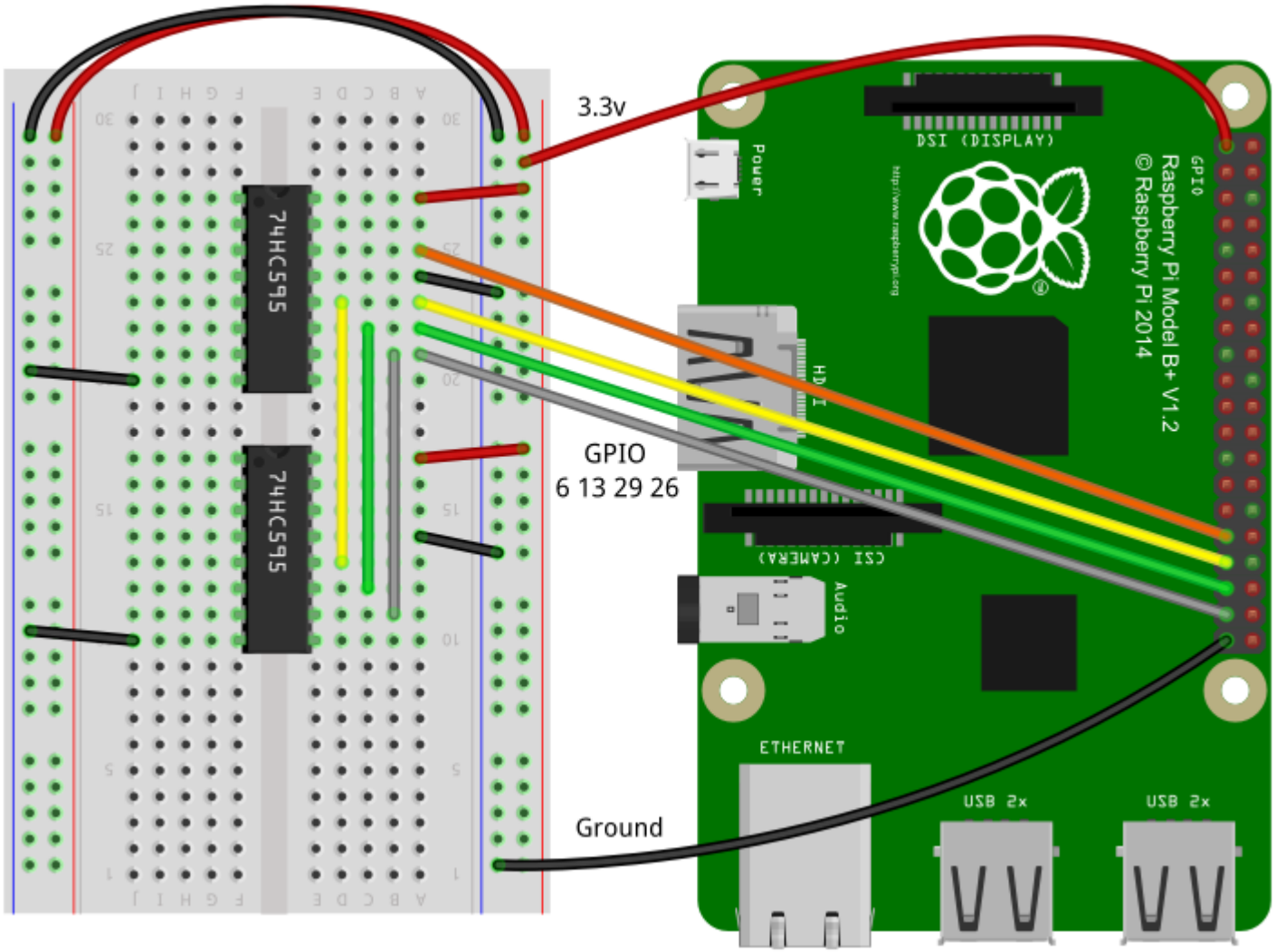
실습8-2: Dual 74HC595



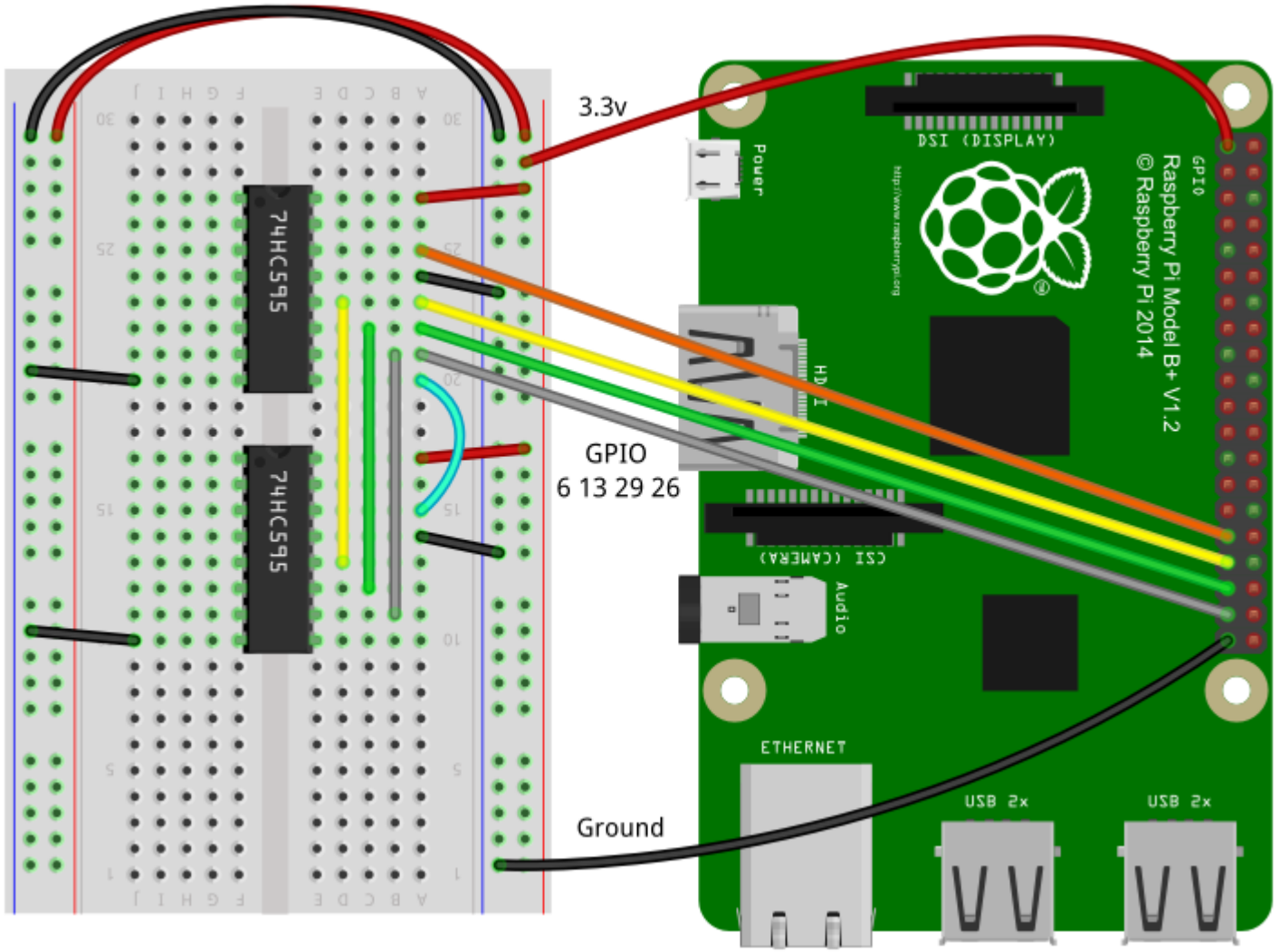
실습8-3: Dual 74HC595



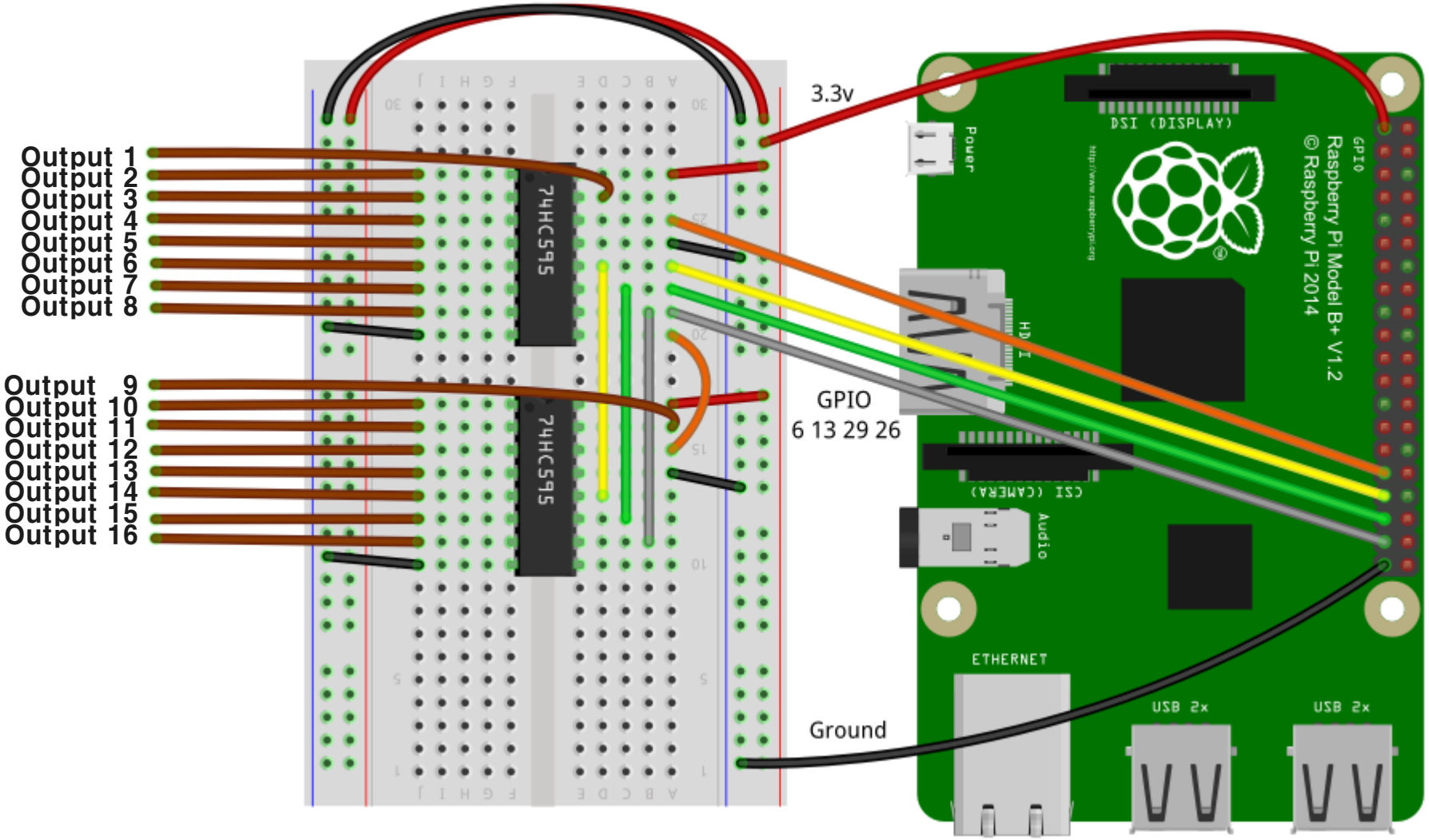
실습8-4: Dual 74HC595



실습8-5: Dual 74HC595



실습8-6: Dual 74HC595



실습8-7: Dual 74HC595

파일명 : dual_74hc595.c

```
#include <stdio.h>
#include <stdlib.h>
#include <wiringPi.h>
#include <stdint.h>
#define SDATA        6    // Serial Input Data
#define STR_CLK      13   // Storage Register Clock(LATCH)
#define SHR_CLK      19   // Shift Register Clock
#define SHR_CLEAR    26   // Shift Register Clear

void allclear(void);

void init(void)
{
    if(wiringPiSetupGpio() == -1){ // wiringPi
        fprintf(stderr, "wiringPiSetupGpio() erorr\n");
        exit(1);
    }
    pinMode(SDATA, OUTPUT);
    pinMode(STR_CLK, OUTPUT);
    pinMode(SHR_CLK, OUTPUT);
    pinMode(SHR_CLEAR, OUTPUT);
    allclear();
}
void allclear(void)
{
    digitalWrite(SHR_CLEAR, 0);
    digitalWrite(SHR_CLEAR, 1);
    digitalWrite(STR_CLK, 0);
    digitalWrite(STR_CLK, 1);    // latch
}
```

실습8-8: Dual 74HC595

파일명 : dual_74hc595.c

```
void set16(uint16_t value)
{
    int i;

    for(i = 0 ; i < 16 ; i++){
        int mask = 0b1 << i;
        if((value & mask) == 0){
            digitalWrite(SDATA, 0);
        }
        else{
            digitalWrite(SDATA, 1);
        }
        digitalWrite(SHR_CLK, 0); //
        digitalWrite(SHR_CLK, 1); //
    }
    // letch
    digitalWrite(STR_CLK, 0); //
    digitalWrite(STR_CLK, 1); //
}

int
main(void)
{
    int i;
    uint16_t arr[] = {  0b1000000000000000,
                        0b0100000000000000,
                        0b0010000000000000,
```

실습8-9: Dual 74HC595

파일명 : dual_74hc595.c

```
                        0b0001000000000000,
                        0b0000100000000000,
                        0b0000010000000000,
                        0b0000001000000000,
                        0b0000000100000000,
                        0b0000000010000000,
                        0b0000000001000000,
                        0b0000000000100000,
                        0b0000000000010000,
                        0b0000000000001000,
                        0b0000000000000100,
                        0b0000000000000010,
                        0b0000000000000001};
    init();

    for(i = 0 ; i < 16 ; i++){
        set16(arr[i]);
        delay(100);
    }
    return 1;
}
```

```
pi@robotcode ~ $ gcc -Wall -W -lwiringPi dual_74hc595.c -o dual_74hc595
pi@robotcode ~ $ sudo ./dual_74hc595
pi@robotcode ~ $
```