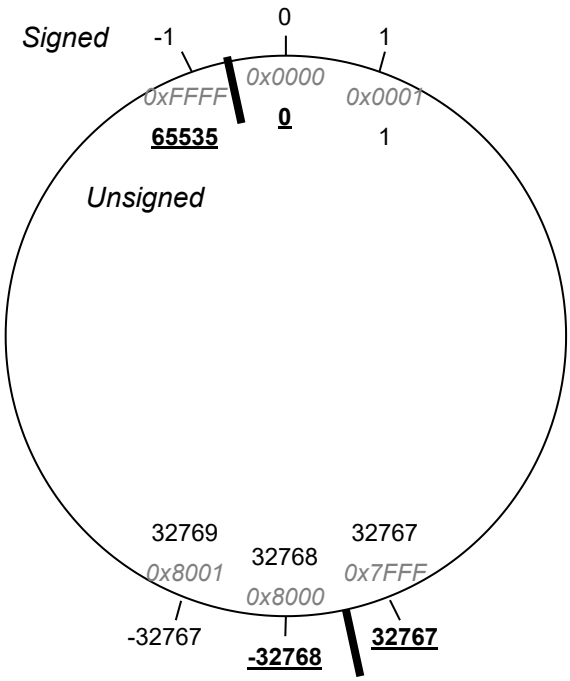
Objectives

- Simple math function for int16, int32



Absolute value of int16 and int32

```c
/*****************************************************************************
*/
/**
@brief      Absolute value of int16_t and int32_t

@param[in]  x   input value of integer

@return     absolute value of x

@note       Overflow is NOT detected
*****************************************************************************
***/

int16_t I16Abs(int16_t x);
int32_t I32Abs(int32_t x);
```

`int16_t I16Abs(int16_t x)`

- within range
  - [0, 32767] <- [0, 32767]
  - [32767, 0] <- [-32767, 0]

- overflow
  - Overflow <- [32768, more than)
- underflow
  - Underflow <- (less than, -32768]

Absolute value of int16 and int32 with saturation

```
/*****************************************************************************
*/
/**
@brief      Absolute value of int16_t and int32_t with saturation

@param[in]  x input value of int16_t or int32_t

@return     absolute value of x

@note       Overflow is detected and saturates it
            Valid range is [-32768 ~ 32767]
*/
int16_t I16AbsSat(int16_t x);
int32_t I32AbsSat(int32_t x);

```

`int16_t I16AbsSat(int16_t x)`

- within range
  - [0, 32767] <- [0, 32767]
  - [32767, 0] <- [-32767, 0]
- overflow
  - 32767 <- [32768, more than)
- underflow
  - 32767 <- (less than, -32768]

Add two int16(or int32) values without saturation

```
/*****************************************************************************
*/
/**
@brief      Add two values of int16_t and int32_t with-out saturation

@param[in]  x input value of int16_t or int32_t
@param[in]  y input value of int16_t or int32_t

@return     Add two value of x and y

@note       Overflow is NOT detected
*****************************************************************************
***/
```

```
12
13 int16_t I16Add(int16_t x, int16_t y);
14 int32_t I32Add(int32_t x, int32_t y);
```

`int16_t I16Add(int16_t x, int16_t y)`

- within range
  - [0, 32767] <- [0, 32767]
  - [32767, 0] <- [-32767, 0]
- overflow
  - Overflow <- [32768, more than)
- underflow
  - Underflow <- (less than, -32768]

Add two int16(or int32) values with saturation

```
1  /*****************************************************************************
   */
2  /**
3  @brief      Add two values of int16_t and int32_t with saturation
4
5  @param[in]  x input value of int16_t or int32_t
6  @param[in]  y input value of int16_t or int32_t
7
8  @return     Add two value of x and y
9
10 @note       Overflow is detected and saturates it
11 */
12 int16_t I16AddSat(int16_t x, int16_t y);
13 int32_t I32AddSat(int32_t x, int32_t y);
14
```

`int16_t I16AddSat(int16_t x, int16_t y)`

- within range
  - [0, 32767] <- [0, 32767]
  - [32767, 0] <- [-32767, 0]
- overflow
  - 32767 <- [32768, more than)
- underflow
  - 32767 <- (less than, -32768]