

ZSharpIR Library

Generated by Doxygen 1.8.14

Contents

1	SharpIR	1
2	Data Structure Index	4
2.1	Data Structures	4
3	File Index	4
3.1	File List	4
4	Data Structure Documentation	4
4.1	ZSharpIR Class Reference	4
4.1.1	Detailed Description	5
4.1.2	Constructor & Destructor Documentation	5
4.1.3	Member Function Documentation	5
4.1.4	Field Documentation	6
5	File Documentation	7
5.1	README.md File Reference	7
5.2	ZSharpIR.cpp File Reference	7
5.3	ZSharpIR.h File Reference	8
5.3.1	Macro Definition Documentation	8
Index		9

1 SharpIR

Arduino Infra Red Sharp Lib

Based on an original work of Dr. Marcal Casas-Cartagena .

1. Perform 25 reading of analog pin (Nb samples can be changed in .h)
2. Sort values
3. Convert median value to cm

Usage

- `#include <SharpIR.h>`
- `SharpIR sharp(ir_analog_pin, model);`
- `int dist = sharp.distance();`

Model :

- GP2Y0A02YK0F → "20150"
- GP2Y0A21YK → "1080"
- GP2Y0A710K0F → "100500"
- GP2YA41SK0F → "430"

Sharp IR Volt Centimeter conversion

GP2Y0A02YK0F

Model: "20150" [20cm to 150cm]

Volt	Distance
2,8	15
2,5	20
2	30
1,55	40
1,24	50
1,05	60
0,905	70
0,82	80
0,7	90
0,66	100
0,6	110
0,55	120
0,5	130
0,455	140
0,435	150

Using MS Excel, we can calculate function (For distance > 15cm) :

Distance = 60.374 X POW(Volt , -1.16)

GP2Y0A21YK

Model: "1080" [10cm to 80cm]

Volt	Distance
2,6	10
2,1	12
1,85	14
1,65	15
1,5	18
1,39	20
1,15	25
0,98	30
0,85	35
0,75	40
0,67	45
0,61	50
0,59	55
0,55	60
0,5	65
0,48	70
0,45	75
0,42	80

Using MS Excel, we can calculate function (For distance > 10cm) :

$$\text{Distance} = 29.988 \times \text{POW}(\text{Volt}, -1.173)$$

GP2D12_24

Model: "1081" [10cm to 80cm]

Volt	Distance
2.534	10
1.391	20
0.952	30
0.727	40
0.581	50
0.493	60
0.483	70

Based on <https://mycurvefit.com/> :

$$\text{Distance} = 24.65251 / (\text{Volt} - 0.1065759)$$

GP2Y0A710K0F

Model: "100500" [100cm to 500cm]

Based on the SHARP datasheet we can calculate the linear function: $y = 137500x + 1125$ which gives us: $1 / ((\text{Volt} - 1125) / 137500) = \text{distance_in_cm}$ (For distance > 100cm)

GP2YA41SK0F (<=> GP2D120)

Model: "430" [4cm to 30cm]

Based on the SHARP datasheet we can calculate the function (For distance > 3cm) :

Distance = 12.08 X POW(Volt , -1.058)

2 Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

[ZSharpIR](#) 4

3 File Index

3.1 File List

Here is a list of all files with brief descriptions:

[ZSharpIR.cpp](#) 7

[ZSharpIR.h](#) 8

4 Data Structure Documentation

4.1 ZSharpIR Class Reference

```
#include <ZSharpIR.h>
```

Collaboration diagram for ZSharpIR:

ZSharpIR
+ GP2Y0A41SK0F + GP2Y0A21YK0F + GP2D12_24 + GP2Y0A02YK0F + GP2Y0A710K0F
+ ZSharpIR() + distance() + setARefVoltage() + SetAnalogReadResolution()

Public Member Functions

- [ZSharpIR](#) (int irPin, const uint32_t _sensorType)
- int [distance](#) ()
- void [setARefVoltage](#) (int refV)
setARefVoltage: set the ADC reference voltage: (default value: 5000mV, set to 3300mV, typically 3.3 on Arduino boards)
- void [SetAnalogReadResolution](#) (int res)
SetAnalogReadResolution: set the ADC resolution : (default value: 10, set to 12, typically 10 on Arduino boards)

Static Public Attributes

- static const uint32_t [GP2Y0A41SK0F](#) = 430
- static const uint32_t [GP2Y0A21YK0F](#) = 1080
- static const uint32_t [GP2D12_24](#) = 1081
- static const uint32_t [GP2Y0A02YK0F](#) = 20150
- static const uint32_t [GP2Y0A710K0F](#) = 100500

4.1.1 Detailed Description

Definition at line 25 of file ZSharpIR.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 ZSharpIR()

```
ZSharpIR::ZSharpIR (  
    int irPin,  
    const uint32_t _sensorType )
```

Definition at line 30 of file ZSharpIR.cpp.

4.1.3 Member Function Documentation

4.1.3.1 distance()

```
int ZSharpIR::distance ( )
```

Definition at line 58 of file ZSharpIR.cpp.

References [GP2D12_24](#), and [NB_SAMPLE](#).

4.1.3.2 SetAnalogReadResolution()

```
void ZSharpIR::SetAnalogReadResolution (
    int res )
```

SetAnalogReadResolution:set the ADC resolution : (default value: 10, set to 12, typically 10 on Arduino boards)

Definition at line 144 of file ZSharpIR.cpp.

4.1.3.3 setARefVoltage()

```
void ZSharpIR::setARefVoltage (
    int refV )
```

setARefVoltage:set the ADC reference voltage: (default value: 5000mV, set to 3300mV, typically 3.3 on Arduino boards)

Definition at line 136 of file ZSharpIR.cpp.

4.1.4 Field Documentation

4.1.4.1 GP2D12_24

```
const uint32_t ZSharpIR::GP2D12_24 = 1081 [static]
```

Definition at line 34 of file ZSharpIR.h.

Referenced by distance().

4.1.4.2 GP2Y0A02YK0F

```
const uint32_t ZSharpIR::GP2Y0A02YK0F = 20150 [static]
```

Definition at line 35 of file ZSharpIR.h.

4.1.4.3 GP2Y0A21YK0F

```
const uint32_t ZSharpIR::GP2Y0A21YK0F = 1080 [static]
```

Definition at line 33 of file ZSharpIR.h.

4.1.4.4 GP2Y0A41SK0F

```
const uint32_t ZSharpIR::GP2Y0A41SK0F = 430 [static]
```

Definition at line 32 of file ZSharpIR.h.

4.1.4.5 GP2Y0A710K0F

```
const uint32_t ZSharpIR::GP2Y0A710K0F = 100500 [static]
```

Definition at line 36 of file ZSharpIR.h.

The documentation for this class was generated from the following files:

- [ZSharpIR.h](#)
- [ZSharpIR.cpp](#)

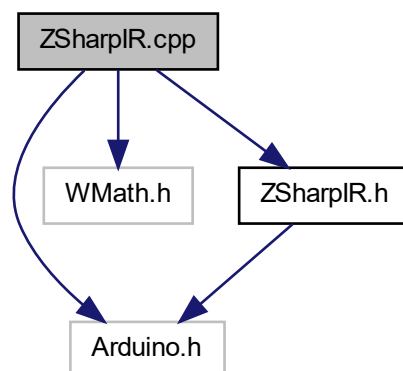
5 File Documentation

5.1 README.md File Reference

5.2 ZSharpIR.cpp File Reference

```
#include "Arduino.h"  
#include "WMath.h"  
#include "ZSharpIR.h"
```

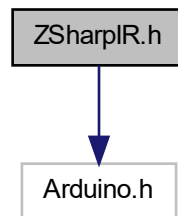
Include dependency graph for ZSharpIR.cpp:



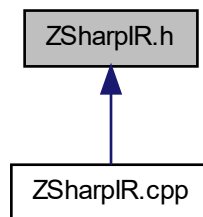
5.3 ZSharpIR.h File Reference

```
#include "Arduino.h"
```

Include dependency graph for ZSharpIR.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class [ZSharpIR](#)

Macros

- `#define` [NB_SAMPLE](#) 10

5.3.1 Macro Definition Documentation

5.3.1.1 NB_SAMPLE

```
#define NB_SAMPLE 10
```

Definition at line 17 of file ZSharpIR.h.

Referenced by `ZSharpIR::distance()`.

Index

distance

[ZSharpIR](#), [5](#)

GP2D12_24

[ZSharpIR](#), [6](#)

GP2Y0A02YK0F

[ZSharpIR](#), [6](#)

GP2Y0A21YK0F

[ZSharpIR](#), [6](#)

GP2Y0A41SK0F

[ZSharpIR](#), [6](#)

GP2Y0A710K0F

[ZSharpIR](#), [7](#)

NB_SAMPLE

[ZSharpIR.h](#), [8](#)

README.md, [7](#)

setARefVoltage

[ZSharpIR](#), [6](#)

SetAnalogReadResolution

[ZSharpIR](#), [5](#)

[ZSharpIR.cpp](#), [7](#)

[ZSharpIR.h](#), [8](#)

[NB_SAMPLE](#), [8](#)

[ZSharpIR](#), [4](#)

distance, [5](#)

GP2D12_24, [6](#)

GP2Y0A02YK0F, [6](#)

GP2Y0A21YK0F, [6](#)

GP2Y0A41SK0F, [6](#)

GP2Y0A710K0F, [7](#)

setARefVoltage, [6](#)

SetAnalogReadResolution, [5](#)

[ZSharpIR](#), [5](#)