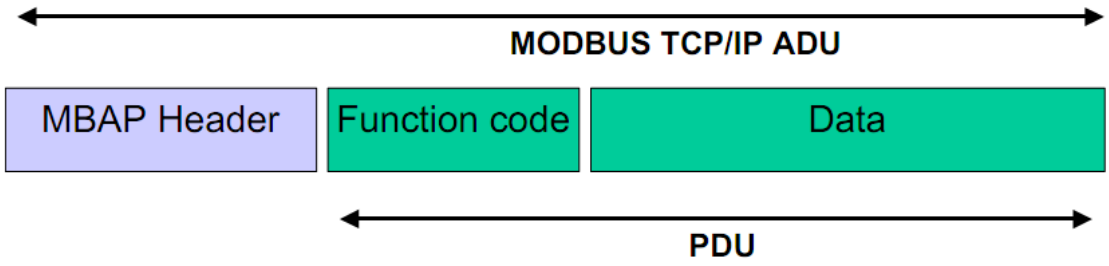
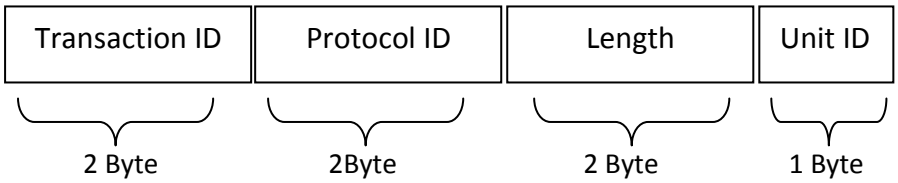
MODBUS TCP/IP PROTOCOL



- **MBAP (Modbus Application Protocol) Header terdapat 4 Bagian.**



a. Transaction ID : ID setiap kali transaksi, setiap transaksi request selalu menunjukkan ID yang berbeda. Sedangkan response menjawab sesuai ID transaksi yang dijawab.

b. Protocol ID : nilainya 0x00

c. Length : merupakan panjang data. Dalam hal ini Unit ID termasuk dalam hitungan. Length adalah panjang Unit ID (1byte) ditambah dengan function Code ditambah lagi dengan Data.

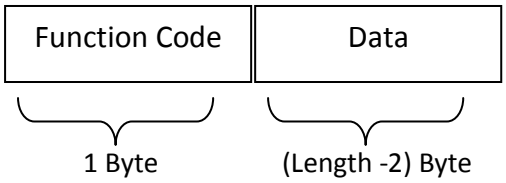
d. Unit ID : ID slave yang digunakan untuk komunikasi.

- PDU terdapat 3 macam PDU :

  1. Request PDU
  2. Response PDU
  3. Exception Response PDU

  Dari 3 macam tersebut memiliki perbedaan struktur masing.

- **PDU (Protocol Data Unit).**



a. Function Code : merupakan kode transaksi yang diinginkan, akses data analog, digital, register dll. Saat transaksi response maka master akan mengembalikan nilai Function Code yang sama, terkecuali apabila terdapat error. Maka server akan mengembalikan nilai error yg sudah di *throw*. Berikut adalah tabel daftar function code yang sudah public.

				Function Codes			
				code	*Sub code*	*(hex)*	*Section*
Data Access	**Bit access**	Physical Discrete Inputs	Read Discrete Inputs	**02**		**02**	6.2
		Internal Bits Or Physical coils	Read Coils	**01**		**01**	6.1
			Write Single Coil	**05**		**05**	6.5
			Write Multiple Coils	**15**		**0F**	6.11
	16 bits access	Physical Input Registers	Read Input Register	**04**		**04**	6.4
		Internal Registers Or Physical Output Registers	Read Holding Registers	**03**		**03**	6.3
			Write Single Register	**06**		**06**	6.6
			Write Multiple Registers	**16**		**10**	6.12
			Read/Write Multiple Registers	**23**		**17**	6.17
			Mask Write Register	**22**		**16**	6.16
			Read FIFO queue	**24**		**18**	6.18
	File record access		Read File record	**20**		**14**	6.14
			Write File record	**21**		**15**	6.15
	Diagnostics		Read Exception status	**07**		**07**	6.7
			Diagnostic	**08**	**00-18,20**	**08**	6.8
			Get Com event counter	**11**		**OB**	6.9
			Get Com Event Log	**12**		**0C**	6.10
			Report Slave ID	**17**		**11**	6.13
			Read device Identification	**43**	**14**	**2B**	6.21
Other			Encapsulated Interface Transport	**43**	**13,14**	**2B**	6.19

b. Data : saat request menyatakan starting Addres register yang ingin dibaca dan berapa banyak data yang dibaca. Sedangkan respons menyatakan nilai yang diinginkan dan panjang data response tersebut (bytes).

Berikut adalah contoh komunikasinya:

Request

Function code	1 Byte	**0x04**
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Input Registers	2 Bytes	0x0001 to 0x007D

Response

Function code	1 Byte	**0x04**
Byte count	1 Byte	2 x **N***
Input Registers	**N*** x 2 Bytes	

***N** = Quantity of Input Registers

Error

Error code	1 Byte	**0x84**
Exception code	1 Byte	01 or 02 or 03 or 04

Here is an example of a request to read input register 9:

Request		**Response**	
Field Name	*(Hex)*	*Field Name*	*(Hex)*
Function	**04**	Function	**04**
Starting Address Hi	**00**	Byte Count	**02**
Starting Address Lo	**08**	Input Reg. 9 Hi	**00**
Quantity of Input Reg. Hi	**00**	Input Reg. 9 Lo	**0A**
Quantity of Input Reg. Lo	**01**		

Contoh request dari JT :

	a	b	c	d	e	f	g
<Buffer	00 01	00 00	00 06	00	04	00 11	00 01>

a. Transaction ID
b. Protocol ID
c. Length
d. Unit ID
e. Function Code
f. Start Register
g. Amount Data that requested

Contoh response dari JT :

	a	b	c	d	e	f	g
<Buffer	00 01	00 00	00 05	00	04	02	12 34>

a. Transaction ID
b. Protocol ID
c. Length
d. Unit ID
e. Function Code
f. Length Data that transfered
g. Value Data that requested