

LAN 1 DHCP
 LAN 2 10.0.0.1

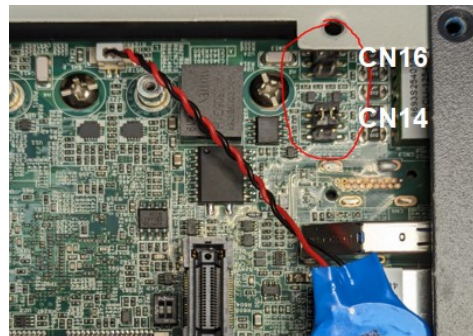
COM1 - /dev/ttyAP0
 COM2 - /dev/ttyAP1

ESU-150 kommer som standard uten uten passord på root bruker.
 Kjør **passwd** for å sette nytt passord.

Serieporter

Tilkobling:

COM		
PIN	RS-232	RS-485
1	TX	D-
2	RX	D+



Serieporter satt til RS-232 (CN16 øverst og CN14 nederst)

Serieporter RS-232/485 jumpers

CN14	COM1 RS-232		COM1 RS-485	
CN16	COM2 RS-232		COM2 RS-485	

ECU-150 kommer som standard satt opp som RS-485

Debug port (USB micro)

RS-232 115200- 8N1

Teste Serieporter med «advcomtest»

Sende data på COM1

advcomtest -s /dev/ttyAP0 9600 'test'

Lese data på COM2

advcomtest -r /dev/ttyAP1 9600

Merk: advcomtest, har problemer med tilkobling til annen hardware. Bruk minicom til annet testing.

Teste Serieporter med Minicom

RS-485 opp mot ADAM-4017 (ASCII) Start Minicom med «minicom -s», Ctr + a -> o for å åpne meny

```
Welcome to minicom 2.7.1
OPTIlqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
Compx A -   Serial Device       : /dev/ttyAP0
Portx B -   Lockfile Location   : /var/lock
  x C -     Callin Program      :
Presx D -   Callout Program     :
  x E -     Bps/Par/Bits        : 9600 8N1
  x F -     Hardware Flow Control : Yes
  x G -     Software Flow Control : No
  x
  x   Change which setting? █
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
  x Screen and keyboard       x
  x Save setup as 9600         x
  x Save setup as..           x
  x Exit                       x
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
```

```
root@ecu150-a539e8: ~
Welcome to minicom 2.7.1
OPTIONS: I18n
Compiled on Dec 23 2019, 02:06:26.
Port /dev/ttyAP0, 14:35:20

Press CTRL-A Z for help on special keys

$01M
█014017P
```

Python eksempelkode

Using the pymodbus library to read coil 0-1, then setting coil 1 to True, and then do another reading

```
#Using the pymodbus library to read coil 0-1, then setting coil 1 to True, and then do another reading

from pymodbus.client import ModbusSerialClient
from pymodbus.transaction import ModbusRtuFramer

# Create a Modbus RTU client
client = ModbusSerialClient(method='rtu',
    port='/dev/ttyAP0',
    baudrate=9600,
    stopbits=1,
    bytesize=8,
    parity='N',
    framer=ModbusRtuFramer)

# Connect to the Modbus RTU slave
client.connect()

# Define the slave address
slave_address = 1

# read current coil states
r = client.read_coils(0,4,1)
print(r.bits)

# write to coil 1 the value 1
client.write_coil(1,True,1)

# reread the coil states
r = client.read_coils(0,4,1)
print(r.bits)
```