

目次

IEEE802.15.4g(920MHz無線)を設定する (IPv6, 6LoWPAN)	1
設定	1
/etc/network/interfaces ファイルへの設定	1
I/F の有効化	2
設定 (ID)	3
ドライバの組み込み	3
Line Discipline のアタッチ	3
IEEE802.15.4g の設定	5
6LoWPAN リンクの作成	7
インターフェース UP	7
疎通の確認 (Link Local)	9
IPv6 アドレスの割り当て	10
RA (Router Advertisement) の設定	11
資料	14
パケットキャプチャ	14

IEEE802.15.4g(920MHz無線)を設定する (IPv6, 6LoWPAN)

MA-E360/N には IEEE802.15.4g 準拠の無線モジュールを搭載しています。

Linux の wpan デバイスとして認識させ 6LoWPAN と組み合わせることで IPv6 で通信できるネットワーク I/F として利用することが可能です。



(TI 社サイトより)

下記情報は現開発時点の情報です。リリース時には変更される場合があります。

設定

ifupdown パッケージに、wpan/lowpan I/F 設定を追加しました。
/etc/network/interfaces ファイルに記述するだけで、

- ドライバの組み込み
- Line Discipline のアタッチ
- 無線チャネルの設定
- PAN ID の設定
- 6LoWPAN リンクの作成
- IPv6 アドレスの割り当て

を行います。

/etc/network/interfaces ファイルへの設定

inet6 の “static” もしくは “auto” メソッドが利用可能です。

interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback
```

```
auto eth0
#iface eth0 inet static
#   address 192.168.253.253
#   netmask 255.255.255.0
#   gateway 192.168.253.1
#   dns-nameservers 192.168.253.1
#post-up /sbin/ethtool -s eth0 autoneg off speed 100 duplex full

iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 192.168.254.254
    netmask 255.255.255.0
    #post-up /sbin/ethtool -s eth1 autoneg off speed 100 duplex full

iface lowpan0 inet6 static
    address 2001:dead:beef::1/64
    ttydev ttyS0
    ldnum 25
    panid 0x1234
    channel 33
    wpan 0
```

追加設定項目について

Ubuntu 標準の ifupdown に追加した設定項目は下記のとおりです。

項目	内容	Notes
ttydev	IEEE802.15.4g module 接続 tty device	ttyS0 固定
ldnum	IEEE802.15.4g module ドライバ Line Discipline No.	25 固定
panid	IEEE802.15.4g PAN ID	
channel	IEEE802.15.4g チャネル	33〜59, 奇数のみ
wpan	wpan I/F No.	0 固定

I/F の有効化

Ethernet(eth0) など他の I/F と同様に、ifup コマンドで有効化します。

```
user1@plum:~$ sudo ifup lowpan0
wpa_supplicant: /sbin/wpa_supplicant daemon failed to start
run-parts: /etc/network/if-pre-up.d/wpasupplicant exited with return code 1
Waiting for DAD... Done
```

```
user1@plum:~$ ifconfig wpan0
wpan0      Link encap:UNSPEC  HWaddr
00-80-6D-01-01-00-00-01-00-00-00-00-00-00-00-00-00
          UP BROADCAST RUNNING NOARP  MTU:127  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:300
          RX bytes:0 (0.0 B)  TX bytes:619 (619.0 B)

user1@plum:~$ ifconfig lowpan0
lowpan0    Link encap:UNSPEC  HWaddr
00-80-6D-01-01-00-00-01-00-00-00-00-00-00-00-00-00
          inet6 addr: fe80::280:6d01:100:1/64 Scope:Link
          inet6 addr: 2001:dead:beef::1/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1280  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

user1@plum:~$
```

設定 (旧)

ドライバの組み込み

serial2.ko というモジュールとなっていますので、それを組み込みます。

```
root@plum:/home/user1# modprobe serial2
```

```
root@plum:/home/user1# lsmod
Module                Size  Used by
serial2               7277   0
```

Line Discipline のアタッチ

IEEE802.15.4g のモジュールとの間の通信は UART となっていますので `ldattach` コマンドにより Line Discipline をアタッチします。

Line Discipline の番号は、Kernel 既存の番号の最後 “ 25 ” にしてあります。

```
root@plum:/home/user1# ldattach -s 115200 25 /dev/ttyS0
```

参考: Line Discipline 定義 (linux/include/uapi/linux/tty.h)

tty.h

```
#ifndef _UAPI_LINUX_TTY_H
#define _UAPI_LINUX_TTY_H

/*
 * 'tty.h' defines some structures used by tty_io.c and some defines.
 */

#define NR_LDISCS      30

/* line disciplines */
#define N_TTY          0
#define N_SLIP         1
#define N_MOUSE        2
#define N_PPP          3
#define N_STRIP        4
#define N_AX25         5
#define N_X25          6    /* X.25 async */
#define N_6PACK         7
#define N_MASC          8    /* Reserved for Mobitex module
<kaz@cafe.net> */
#define N_R3964         9    /* Reserved for Simatic R3964 module */
#define N_PROFIBUS_FDL 10    /* Reserved for Profibus */
#define N_IRDA          11    /* Linux IrDa -
http://irda.sourceforge.net/ */
#define N_SMSBLOCK      12    /* SMS block mode - for talking to GSM data
*/

/* cards about SMS messages */
#define N_HDLC          13    /* synchronous HDLC */
#define N_SYNC_PPP      14    /* synchronous PPP */
#define N_HCI           15    /* Bluetooth HCI UART */
#define N_GIGASET_M101  16    /* Siemens Gigaset M101 serial DECT
adapter */
#define N_SLCAN         17    /* Serial / USB serial CAN Adaptors */
#define N_PPS           18    /* Pulse per Second */
#define N_V253          19    /* Codec control over voice modem */
#define N_CAIF           20    /* CAIF protocol for talking to modems */
#define N_GSM0710       21    /* GSM 0710 Mux */
#define N_TI_WL         22    /* for TI's WL BT, FM, GPS combo chips */
#define N_TRACESINK      23    /* Trace data routing for MIPI P1149.7 */
#define N_TRACEROUTER    24    /* Trace data routing for MIPI P1149.7 */
#define N_IEEE802154     25    /* Serial / USB serial IEEE802154.4 device
*/

#endif /* _UAPI_LINUX_TTY_H */
```

下記のように“wpan0”が生えてきます。

```
root@plum:/home/user1# ip link show wpan0
33: wpan0: <BROADCAST,N0ARP> mtu 127 qdisc noop state DOWN mode DEFAULT
group default qlen 300
    link/ieee802.15.4 c6:ba:b2:3c:45:e1:55:b4 brd ff:ff:ff:ff:ff:ff:ff:ff
```

現在EUI64 のアドレスはランダムで割り当てています。
リリース時には、モジュールに割り当てた EUI64 を ldattach 時に取得するようになります。

また、現状(Linux Kernel v4.0)での IEEE802.15.4 stack ではMTU が 127bytes 固定となっています。
IEEE802.15.4g(ECHONET Lite) では 255bytes を使うこととなりますのでKernel の変更を予定しています。

IEEE802.15.4g の設定

インターフェースを UP するまえに、無線関係の設定を行います。

チャネルの設定

“iwpan” コマンドによりwpan デバイスの状態を確認してみます。

```
root@plum:/home/user1# iwpan list
wpan_phy wpan-phy0
supported channels:
    page 9: 4,5,6,7,8,9,10,11,12,13,14,15,16,17
current_page: 0
current_channel: 0
cca_mode: 0
tx_power: 0
```

チャネルは、Page: 9 の 4-17 (ARIB STD-T108 ch33,34 から ch59,60) が利用できます。

IEEE802.15.4g (920MHz) チャネル割り当て (ARIB STD-T108 表3-12 より)

単位チャネル番号	中心周波数(MHz)	channel	単位チャネル番号	中心周波数(MHz)	channel
24,25	920.7	-	43,44	924.5	9
25,26	920.9	(0)	44,45	924.7	-
26,27	921.1	-	45,46	924.9	10
27,28	921.3	(1)	46,47	925.1	-
28,29	921.5	-	47,48	925.3	11
29,30	921.7	(2)	48,49	925.5	-
30,31	921.9	-	49,50	925.7	12

単位チャネル番号	中心周波数(MHz)	channel	単位チャネル番号	中心周波数(MHz)	channel
31,32	922.1	(3)	50,51	925.9	-
33,34	922.5	4	51,52	926.1	13
34,35	922.7	-	52,53	926.3	-
35,36	922.9	5	53,54	926.5	14
36,37	923.1	-	54,55	926.7	-
37,38	923.3	6	55,56	926.9	15
38,39	923.5	-	56,57	927.1	-
39,40	923.7	7	57,58	927.3	16
40,41	923.9	-	58,59	927.5	-
41,42	924.1	8	59,60	927.7	17
42,43	924.3	-	60,61	927.9	-

※ ECHONET Lite の推奨仕様([JJ-300.10 第2.1版\(PDF\)](#))では100kbps時 “**ARIB** 規定 33〜60 チャネルの(奇数 + 偶数)チャネル、” とされているため、上記チャネルの割り当てとしています。

ch33,34(中心周波数 922.5MHz, 帯域幅 400kHz) に設定してみます。

```
root@plum:/home/user1# iwpan wpan-phy0 set channel 9 4
```

再度確認してみるとpage: 9, channel: 4 に設定されたことが確認できます。

```
root@plum:/home/user1# iwpan list
wpan_phy wpan-phy0
supported channels:
  page 9: 4,5,6,7,8,9,10,11,12,13,14,15,16,17
current_page: 9 <----
current_channel: 4 <----
cca_mode: 0
tx_power: 0
```

PAN ID の設定

PAN ID を設定します。例として0x1234 に設定してみます。

```
root@plum:/home/user1# iwpan dev wpan0 set pan_id 0x1234
```

PAN ID が 0x1234 に設定できました。

```
root@plum:/home/user1# iwpan dev wpan0 info
```

```
Interface wpan0
  ifindex 9
  wpan_dev 0x1
  extended_addr 0x9065eb5ea23bce7e
  short_addr 0xffff
  pan_id 0x1234  <-----
  type node
  max_frame_retries -1
  min_be 3
  max_be 5
  max_csma_backoffs 4
  lbt 0
```

6LoWPAN リンクの作成

6LoWPAN は、wpan I/F を使用した仮想リンクですので“ip link” コマンドにより type “lowpan” として作成します。

```
root@plum:/home/user1# ip link add link wpan0 name lowpan0 type lowpan
```

ip link show で確認するとMTU 1280 bytes (IPv6 minimum MTU) の仮想リンク “lowpan0” が作成できていることがわかります。

```
root@plum:/home/user1# ip link show lowpan0
10: lowpan0: <BROADCAST,MULTICAST> mtu 1280 qdisc noop state DOWN mode
DEFAULT group default
    link/ieee802.15.4 90:65:eb:5e:a2:3b:ce:7e brd ff:ff:ff:ff:ff:ff:ff:ff
```

インターフェース UP

wpan0, lowpan0 デバイスを UP します。

```
root@plum:/home/user1# ip link set wpan0 up
root@plum:/home/user1# ip link set lowpan0 up
```

wpan0, lowpan0 両デバイスの状態が “UP” になったことを確認します。

```
root@plum:/home/user1# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

```
2: dummy0: <BROADCAST,NOARP> mtu 1500 qdisc noop state DOWN mode DEFAULT
group default
    link/ether 6e:38:24:14:91:c0 brd ff:ff:ff:ff:ff:ff
3: tunl0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN mode DEFAULT group
default
    link/ipip 0.0.0.0 brd 0.0.0.0
4: gre0@NONE: <NOARP> mtu 1476 qdisc noop state DOWN mode DEFAULT group
default
    link/gre 0.0.0.0 brd 0.0.0.0
5: gretap0@NONE: <BROADCAST,MULTICAST> mtu 1462 qdisc noop state DOWN mode
DEFAULT group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
6: ip_vti0@NONE: <NOARP> mtu 1364 qdisc noop state DOWN mode DEFAULT group
default
    link/ipip 0.0.0.0 brd 0.0.0.0
7: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP mode DEFAULT group default qlen 1000
    link/ether 00:80:6d:8d:30:08 brd ff:ff:ff:ff:ff:ff
8: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state
DOWN mode DEFAULT group default qlen 1000
    link/ether 00:80:6d:8d:30:09 brd ff:ff:ff:ff:ff:ff
9: wlan0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 127 qdisc pfifo_fast state
UNKNOWN mode DEFAULT group default qlen 300
    link/ieee802.15.4 90:65:eb:5e:a2:3b:ce:7e brd ff:ff:ff:ff:ff:ff:ff:ff
10: lowpan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1280 qdisc noqueue state
UNKNOWN mode DEFAULT group default
    link/ieee802.15.4 90:65:eb:5e:a2:3b:ce:7e brd ff:ff:ff:ff:ff:ff:ff:ff
```

ifconfig コマンドでも確認できます。こちらのほうが見やすいかもしれません。

```
root@plum:/home/user1# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:80:6d:8d:30:08
          inet addr:192.168.253.35  Bcast:192.168.253.255
Mask:255.255.255.0
          inet6 addr: fe80::280:6dff:fe8d:3008/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1219 errors:0 dropped:0 overruns:0 frame:0
          TX packets:802 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:109803 (109.8 KB)  TX bytes:122197 (122.1 KB)
          Interrupt:170

eth1      Link encap:Ethernet  HWaddr 00:80:6d:8d:30:09
          inet addr:192.168.254.254  Bcast:192.168.254.255
Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
```

```

RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:65536  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lowpan0  Link encap:UNSPEC  HWaddr 90-65-EB-5E-A2-3B-
CE-7E-00-00-00-00-00-00-00
        inet6 addr: fe80::9265:eb5e:a23b:ce7e/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1280  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wpan0    Link encap:UNSPEC  HWaddr 90-65-EB-5E-A2-3B-
CE-7E-00-00-00-00-00-00-00
        UP BROADCAST RUNNING NOARP  MTU:127  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:300
        RX bytes:0 (0.0 B)  TX bytes:276 (276.0 B)
```

疎通の確認 (Link Local)

別の MA-E360/N を立ち上げ、無線のチャンネルと PAN ID を同一に設定しておきます。
lowpan0 I/F の Link Local アドレスを確認し、そのアドレスに対して PING6 を送信してみます。

```

root@plum:/home/user1# ping6 fe80::e7bf:e85d:44b7:fd9a%lowpan0
PING fe80::e7bf:e85d:44b7:fd9a%lowpan0(fe80::e7bf:e85d:44b7:fd9a) 56 data
bytes
64 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=1 ttl=64 time=102 ms
64 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=2 ttl=64 time=58.1 ms
64 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=3 ttl=64 time=58.1 ms
64 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=4 ttl=64 time=58.4 ms
64 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=5 ttl=64 time=58.0 ms
^Croot@plum:/home/user1#
```

応答が返ってきました。

6LoWPAN により IEEE802.15.4 のフレーム長より長いフレームの送受信が可能か、PING6 のパケットサイズを長くしてみます。

```
root@plum:/home/user1# ping6 fe80::e7bf:e85d:44b7:fd9a%lowpan0 -s 1500 -i 5 -c 5
PING fe80::e7bf:e85d:44b7:fd9a%lowpan0(fe80::e7bf:e85d:44b7:fd9a) 1500 data bytes
1508 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=1 ttl=64 time=850 ms
1508 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=2 ttl=64 time=805 ms
1508 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=3 ttl=64 time=805 ms
1508 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=4 ttl=64 time=803 ms
1508 bytes from fe80::e7bf:e85d:44b7:fd9a: icmp_seq=5 ttl=64 time=804 ms

--- fe80::e7bf:e85d:44b7:fd9a%lowpan0 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 20016ms
rtt min/avg/max/mdev = 803.255/813.828/850.654/18.453 ms
```

1500 バイトに設定してもきちんと通信できることが確認できました。

IPv6 アドレスの割り当て

Ubuntu の標準設定ファイル¹⁾で設定します。
例ですので「2001:dead:beef::1」にしてみます。

interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

auto eth0
#iface eth0 inet static
#   address 192.168.253.253
#   netmask 255.255.255.0
#   gateway 192.168.253.1
#   dns-nameservers 192.168.253.1
#post-up /sbin/ethtool -s eth0 autoneg off speed 100 duplex full

iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 192.168.254.254
    netmask 255.255.255.0
    #post-up /sbin/ethtool -s eth1 autoneg off speed 100 duplex full
```

```
iface wlan0 inet static
    address 10.254.0.1
    netmask 255.255.255.0

iface lowpan0 inet6 static
    address 2001:dead:beef::1
    netmask 64
```

ifup コマンドでアドレスを設定することができます。

```
root@plum:/home/user1# ifup lowpan0
Waiting for DAD... Done
root@plum:/home/user1# ifconfig lowpan0
lowpan0    Link encap:UNSPEC  HWaddr 90-65-EB-5E-A2-3B-
CE-7E-00-00-00-00-00-00-00-00
    inet6 addr: fe80::9265:eb5e:a23b:ce7e/64 Scope:Link
    inet6 addr: 2001:dead:beef::1/64 Scope:Global
    UP BROADCAST RUNNING MULTICAST  MTU:1280  Metric:1
    RX packets:0 errors:0 dropped:0 overruns:0 frame:0
    TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:0
    RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@plum:/home/user1#
```

RA (Router Advertisement) の設定

以上で 920MHz 無線を “IPv6” として扱うことができるようになりましたので、通常の IPv6 と同じように使用できます。

まずは `radvd` を立ち上げ、配下の MA-E360/N の IPv6 アドレスの自動設定を行ってみます。

sysctl 設定の変更

`radvd` を導入する前に、IPv6 の転送を有効にしておく必要があります。
`/etc/sysctl.conf` に下記エントリを追加します。

```
net.ipv6.conf.all.forwarding = 1
```

すぐ有効化するために、コマンドラインで実行しておきます。

```
root@plum:/home/user1# sysctl net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1
root@plum:/home/user1#
```

radvd の導入

radvd パッケージを apt-get により導入します。

```
root@plum:/home/user1# apt-get update
■■■ snip ■■■
root@plum:/home/user1# apt-get install radvd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  radvd
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 0 B/59.4 kB of archives.
After this operation, 162 kB of additional disk space will be used.
Selecting previously unselected package radvd.
(Reading database ... 20592 files and directories currently installed.)
Preparing to unpack .../radvd_1%3a1.9.1-1.1ubuntu2_armhf.deb ...
Unpacking radvd (1:1.9.1-1.1ubuntu2) ...
Processing triggers for ureadahead (0.100.0-16) ...
Setting up radvd (1:1.9.1-1.1ubuntu2) ...
Starting radvd:
* /etc/radvd.conf does not exist or is empty.
* See /usr/share/doc/radvd/README.Debian
* radvd will *not* be started.
localepurge: Disk space freed in /usr/share/locale: 0 KiB
localepurge: Disk space freed in /usr/share/man: 0 KiB

Total disk space freed by localepurge: 0 KiB

root@plum:/home/user1#
```

設定 (/etc/radvd.conf)

radvd の設定ファイルを作成します。

[radvd.conf](#)

```
interface lowpan0
{
```

```
AdvSendAdvert on;  
prefix 2001:dead:beef::/64 { };  
};
```

起動

radvd を起動させます。

```
root@plum:/home/user1# service radvd start  
Starting radvd: radvd.  
root@plum:/home/user1#
```

```
root@plum:/home/user1# ps ax|grep radvd  
3497 pts/1    S        0:00 /usr/sbin/radvd -u radvd -p  
/var/run/radvd/radvd.pid  
3499 ?        S        0:00 /usr/sbin/radvd -u radvd -p  
/var/run/radvd/radvd.pid  
3501 pts/1    S+       0:00 grep --color=auto radvd  
root@plum:/home/user1#
```

動作の確認

きちんと Router Advertisement が機能しているか、もう片方側の MA-E360/N の情報を見えます。

```
root@plum:/home/user1# ifconfig lowpan0  
lowpan0  Link encap:UNSPEC  HWaddr 9A-37-05-E0-5E-7A-  
C3-5B-00-00-00-00-00-00-00-00  
        inet6 addr: 2001:dead:beef:0:650e:f3b:b1fb:832b/64 Scope:Global  
<----  
        inet6 addr: fe80::9837:5e0:5e7a:c35b/64 Scope:Link  
        inet6 addr: 2001:dead:beef:0:9837:5e0:5e7a:c35b/64 Scope:Global  
<----  
        UP BROADCAST RUNNING MULTICAST  MTU:1280  Metric:1  
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
        collisions:0 txqueuelen:0  
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)  
  
root@plum:/home/user1#
```

このように「2001:dead:beef:0/64」のアドレスが自動で振られていることが確認できました。

radvd を動作させている MA-E360/N へ、グローバルアドレスで PING6 が通るか確認してみます。

```
root@plum:/home/user1# ping6 2001:dead:beef::1 -c 5
PING 2001:dead:beef::1(2001:dead:beef::1) 56 data bytes
64 bytes from 2001:dead:beef::1: icmp_seq=1 ttl=64 time=74.7 ms
64 bytes from 2001:dead:beef::1: icmp_seq=2 ttl=64 time=74.5 ms
64 bytes from 2001:dead:beef::1: icmp_seq=3 ttl=64 time=74.6 ms
64 bytes from 2001:dead:beef::1: icmp_seq=4 ttl=64 time=74.5 ms
64 bytes from 2001:dead:beef::1: icmp_seq=5 ttl=64 time=74.3 ms

--- 2001:dead:beef::1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 74.382/74.578/74.783/0.368 ms
root@plum:/home/user1#
```

Link Local でないアドレスでも通信ができることが確認できました。

資料

パケットキャプチャ

参考のためwpan0 I/F でキャプチャしたデータを置いておきます。
Router Advertisement や Neighbor Solicitation, Neighbor Advertisement がでていること、
ICMP のパケットが 6LoWPAN により分割されて送信されていることが確認できます。

File	Stat	Note
6lowpan_capture.pcap	2015/05/15 13:30 21.7 KB	6LoWPAN packet capture

1)

/etc/network/interfaces

From:

<https://ma-tech.centurysys.jp/> - MA-X/MA-S/MA-E/IP-K Developers' WiKi

Permanent link:

https://ma-tech.centurysys.jp/doku.php?id=mae3xx_ope:setup_lowpan:start

Last update: **2015/09/17 21:26**