

Linux.conf.au 2011

Saving Money with Misterhouse, having a smart house, and other neat tricks

<http://marc.merlins.org/linux/talks/MisterhouseUseCase/>

Marc MERLIN

marc_soft@merlins.org



Quick Outline

- Misterhouse overview.
- Home automation wire solutions: X10, Insteon, Zwave.
- Other HA protocols: 1-wire, XPL.
- HAI RC-xx and Omnistat2 thermostats.
- Power monitoring: Brultech ECM-1240
- Misterhouse and perl programming.
- Putting it all together: running home made AC with misterhouse, on demand closet cooling, watering plants when need be.
- Teach your cat(s) to respect your authorita!
- Graphing with cacti.

Misterhouse Overview

- Misterhouse was written about 10 years ago to allow for complex home automation.
- Based on perl.
- Multiplatform (Linux, Mac, Windows).
- Works on small router like devices (MIPS, ARM, etc...).
- Went from few protocols to supporting almost all open protocols.
- Helpful user base on misterhouse-users@lists.sf.net

Powerline Home Automation Solutions

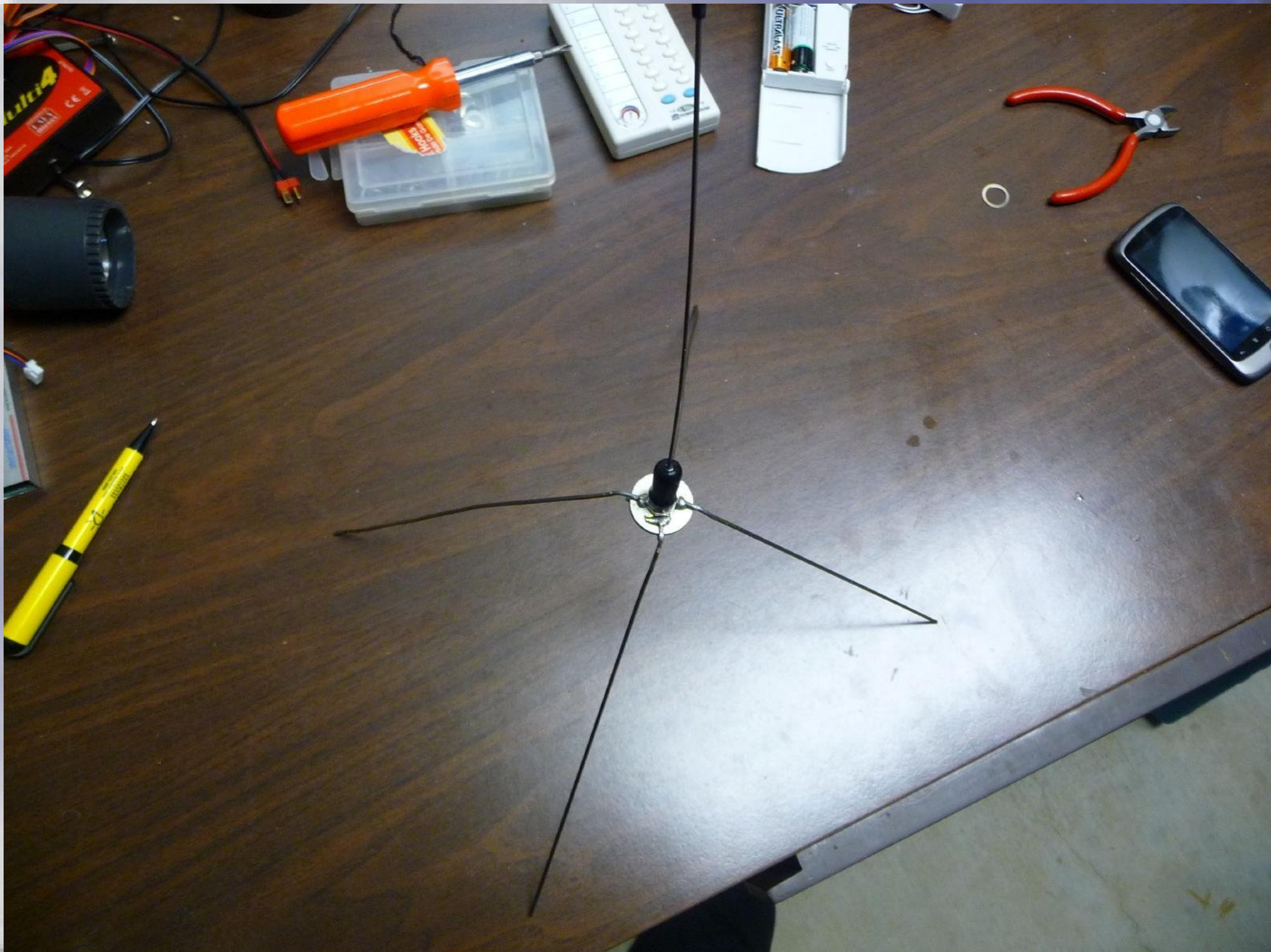
- Where it all started: X10
- Insteon is “X10 mostly done right”.
- Phase Coupling/Bridging (X10 bridge booster, or Insteon Wireless Bridges).
- Z-Wave “look ma, no wires”, but not plug and play.
- UPB, and others,
- Backup solution: Toggling plugs with 1-wire network or serial port.

X10

- Very simple, A1-A16 .. P1-P10 codes.
- Devices typically only send or receive codes (instead of devices do both).
- Basic X10 protocol does not allow state queries.
- X10 is broadcast, there may be N unknown listeners.
- X10 does not have acknowledgement or resends (RF can have some resends).
- Some vendors have extended X10 codes which allow for custom dim levels, or state queries.
- X10 can require a phase bridge and amplifier.
- Devices like CM11 allow reading and sending powerline codes.

X10RF

- X10RF is a separate protocol in the 310Mhz range (US), which needs to be gatewayed to the powerline.
- X10RF shares X10 codes, N devices may receive or send on a given code.
- Basic X10RF gateways only listen on one house code, making gatewaying of many devices cumbersome.
- Better X10RF receivers exist to receive codes directly on a PC via serial port (CM26a, W800, or rfxcom).
- Bypassing the power line can save 2 seconds on code reception.
- Depending on sender, code is sent 2 to 6 times to help with RF noise.







20800R332A
32 bit
RF Receiver

X10sec

- X10sec is also in the 310Mhz range, but a different protocol (4 bytes).
- CM26a won't decode it, but W800 and rfxcom will.
- Mostly used by DS10a security sensors for doors & windows.
- Works fine for monitoring your mailbox being opened :)
- Signal is sent 3 times and current state resent every hour just in case.
- Low battery bit is included in protocol.



mailboxes.com
145 288 4950 1996-2007
Model 145 288 4950 1996-2007

12

```

my %x10secstate;
my $x10sec_timers;
my $x10sec_inactivity;

my $open_timer = 10 * 60;
my $inactivity_timer = 3600*24*7;

foreach my $dev (["x10sec_garage1", "Main Garage Door"], ["x10sec_garage2", "Small
Garage Door"], ["x10sec_mailbox", "Mailbox"]) {
    my $obj = $dev->[0]; my $name = $dev->[1];
    $obj = &::get_object_by_name($obj);
    $x10secstate{$obj} = $obj->state if (not defined $x10secstate{$obj});

    if ($obj->state_now) {
        my $oldstate = $x10secstate{$obj};
        my ($sec,$min,$hour,$mday,$mon,$year,$wday,$yday,$isdst) = localtime();
        $year += 1900; $mon++;
        print_log "X10SEC: $name is ".$obj->state." (was $oldstate)";

        open(LOG, ">>/var/log/x10sec");
        printf LOG ("%0.4d/%0.2d/%0.2d %0.2d:%0.2d:%0.2d $name: ".$obj->state." (was
$oldstate)\n", $year,$mon,$mday,$hour,$min,$sec);
        close(LOG);

        if ($obj->state ne $oldstate) {
            if ($obj->state =~ /alert/) {
                system('date | Mail -s "'. $name.'" Just Opened" EMAIL');

                # little hack to trick the mh parser, or it'll try to define
                # $x10_sec_timers->{$device} as a timer and fail.
                my $newtimer = new Timer();
                $x10sec_timers->{$obj} = $newtimer;
                $x10sec_timers->{$obj}->set($open_timer);
            }
        }
    }
}

```

```

    my $newtimer2 = new Timer();
    $x10sec_inactivity->{$obj} = $newtimer2;
    $x10sec_inactivity->{$obj}->set($inactivity_timer);

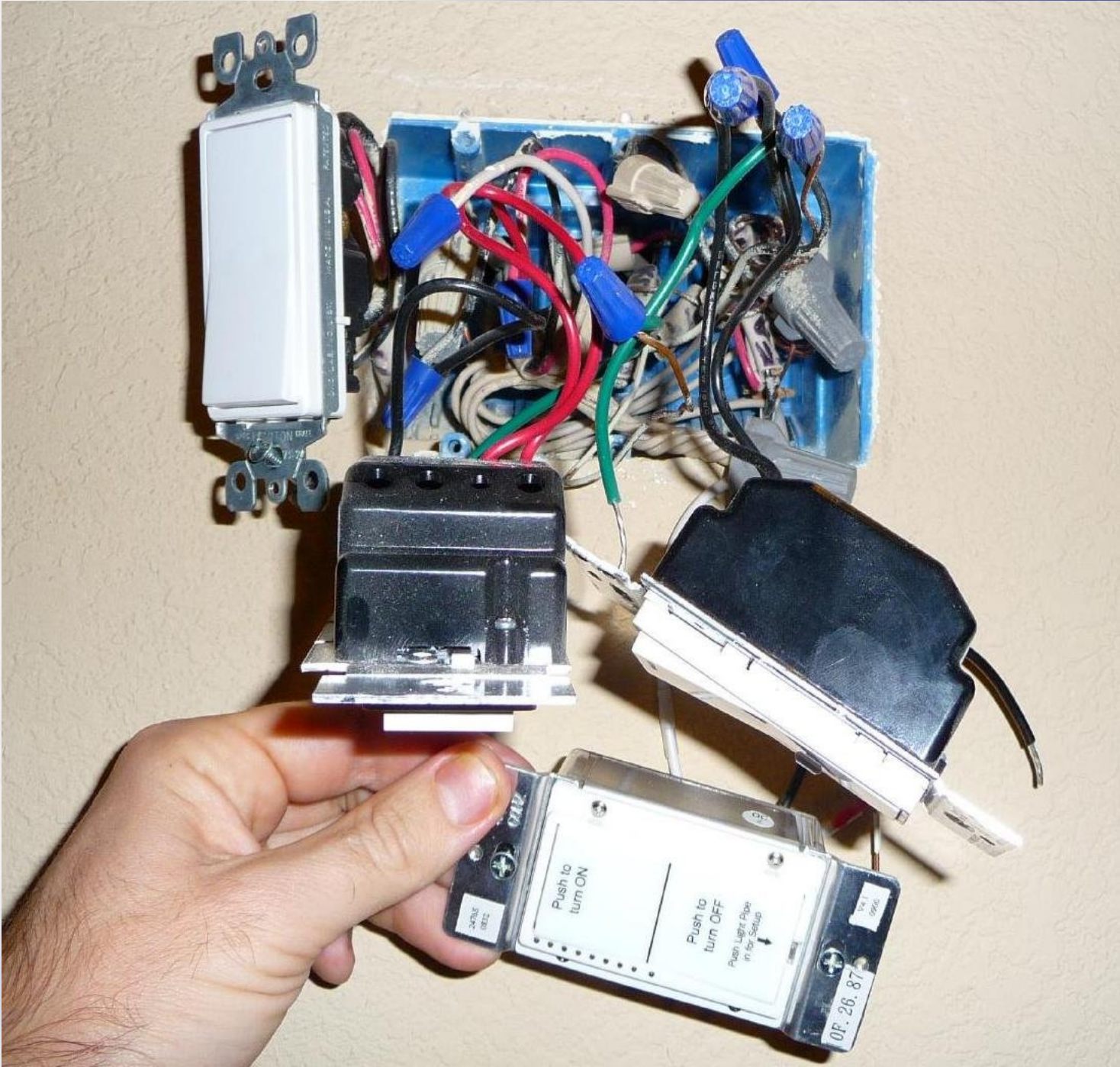
} elsif ($obj->state =~ /normal/) {
    my $timer = "(unset)";
    if ($x10sec_inactivity->{$obj}) {
        $timer = (($inactivity_timer/60) - $x10sec_inactivity->{$obj}-
>minutes_remaining);
    }
    system('date | Mail -s "'. $name.' Just Closed after '.$timer.'mn"
EMAIL');
    $x10sec_timers->{$obj}->unset if (defined $x10sec_timers->{$obj});
    $x10sec_inactivity->{$obj}->unset if (defined $x10sec_inactivity-
>{$obj});
} else {
    system('date | Mail -s "'. $name.' UNKNOWN STATE" EMAIL');
}
$x10secstate{$obj} = $obj->state;
}
}

if (defined $x10sec_timers->{$obj} and $x10sec_timers->{$obj}->expired) {
    system('date | Mail -s "'. $name.' open longer than '.sprintf("%.1f",
$open_timer/60).' mn" EMAIL');
}
if (defined $x10sec_inactivity->{$obj} and $x10sec_inactivity->{$obj}->expired)
{
    system('date | Mail -s "'. $name.' inactive too long ('.sprintf("%.1f",
$inactivity_timer/3600).' hours)" EMAIL');
}
}

```

Insteon

- Insteon is X10 mostly done right. It uses 5V DC signaling like X10 and can read/send X10 codes too.
- One 6 byte address per device.
- Commands are sent to one or more devices and acknowledged.
- Each device resends data received synchronously for better range.
- Data is resent 3 times to work around powerline noise.
- Phases are bridged with RF access points, and they can also be used to jump across a UPS or decoupled line.
- Requires a neutral wire to power the switch.
- No support in Australia – New Zealand :(



Interfacing Insteon with Misterhouse

- Loads of instructions I wrote are here:
<http://misterhouse.wikispaces.com/Insteon>
- Interfacing with a computer uses a PLM (power line modem), or part 2412/3 from smarthome (serial or USB).
- Commands aren't fire and forget, they require queuing, ack processing, and resends.
- You can't just easily send commands from cron, and that's where the misterhouse insteon code comes in play.
- Switch to switch links can be programmed in misterhouse and mass-sent to all your switches instead of painstakingly programmed locally.
- Misterhouse keeps track of switch status updates.

Z-Wave

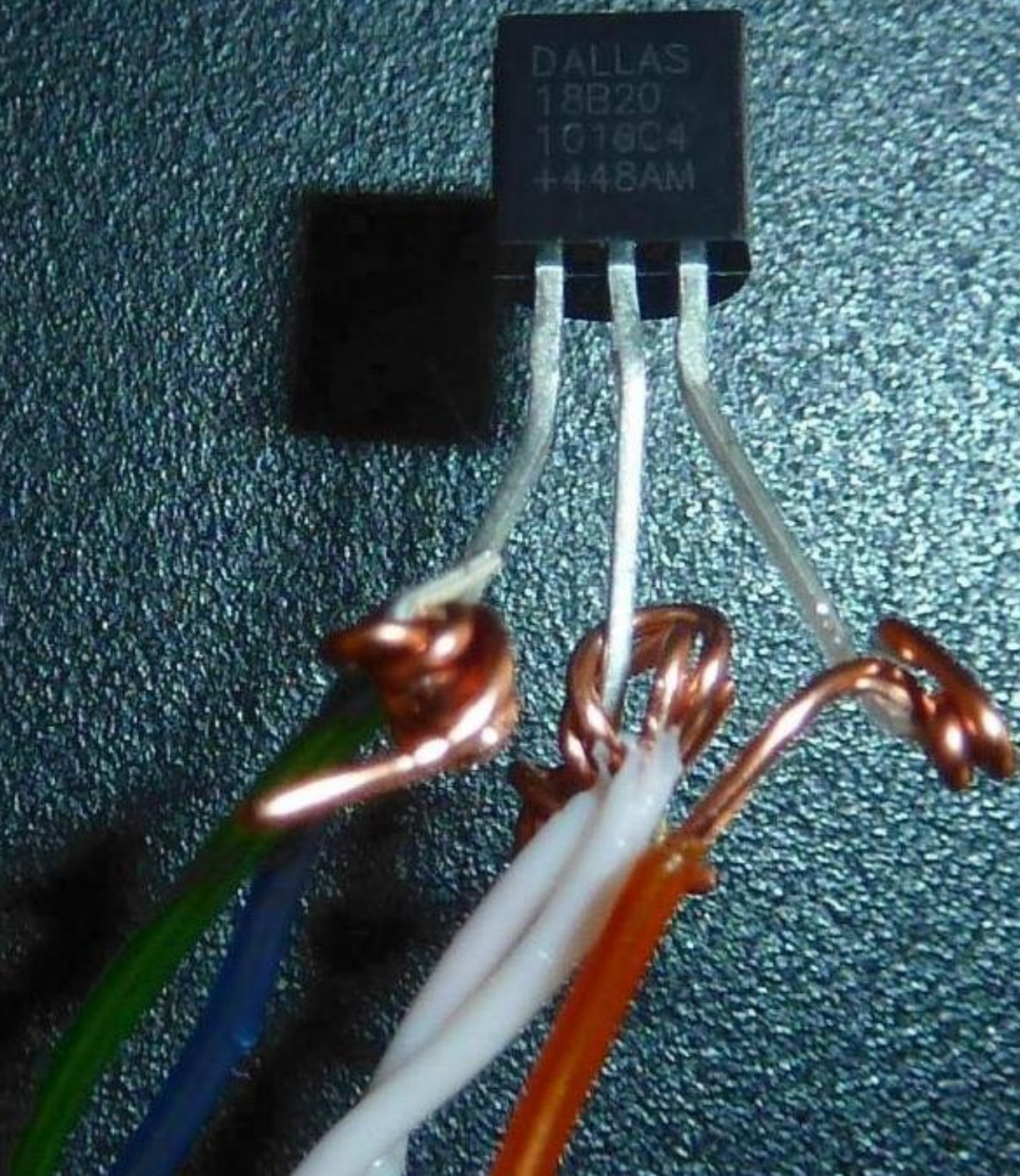
- Z-Wave has wireless communication only.
- Does not require a neutral wire.
- More expensive than Insteon.
- More companies support it, but not all the same way. A switch that reports toggles costs twice as much as one that doesn't.
- Complicated to setup, requires a master controller programmer and star/bridge programming.
- Supported by misterhouse, but little documentation.

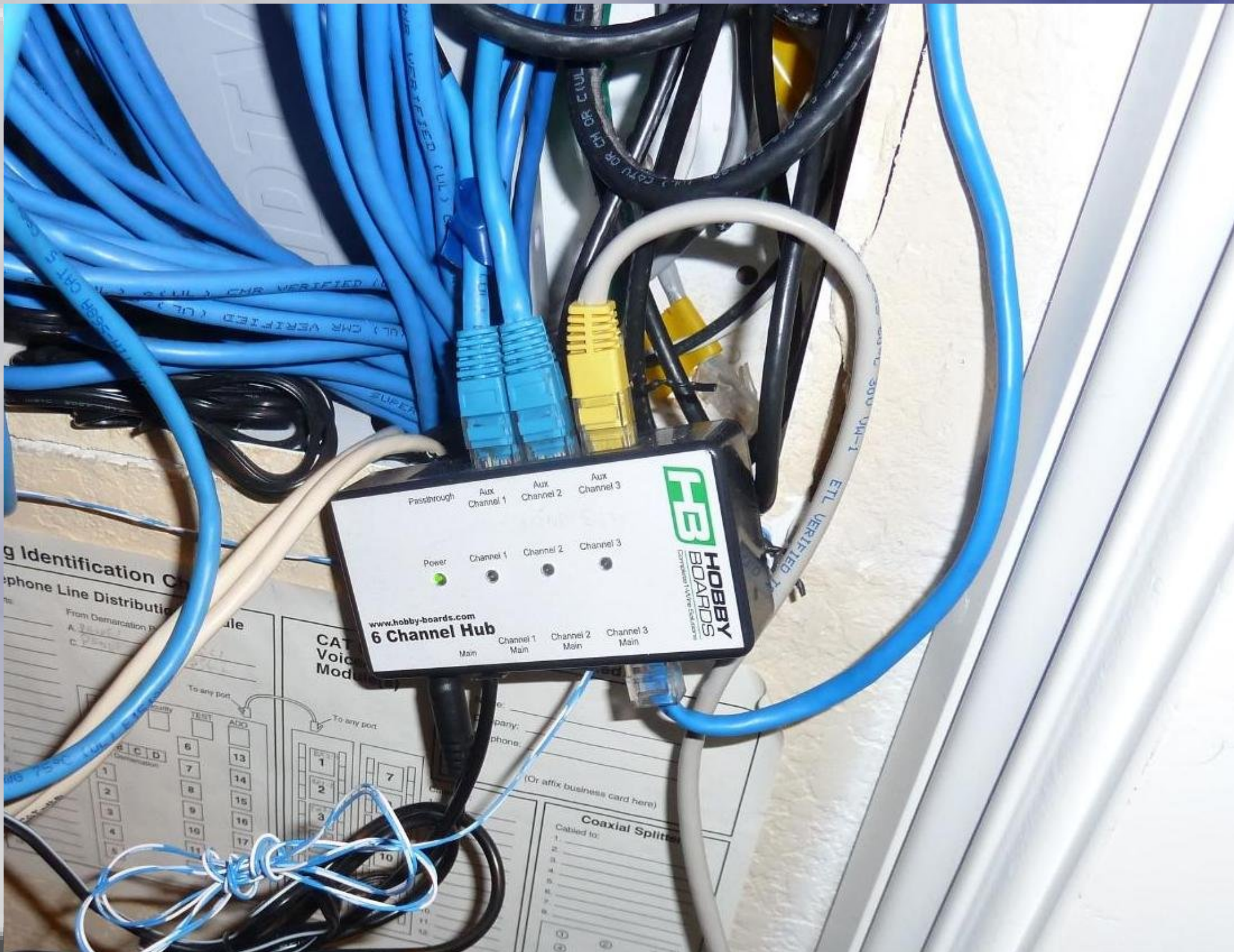
1-wire

- 1-wire was initially meant for embedded devices and cars: ground plane and one 1 signal wire.
- 1-wire is more reliable/faster with a 3rd wire for 5V
- Some 3rd party 1-wire devices also require 12-24V.
- Bus topology, not star, unless you use a hub or loop the signal wire back a Cat-5 pair.
- http://marc.merlins.org/perso/linuxha/post_2009-12-23_Temp
<http://marc.merlins.org/perso/linuxha/2009-12.html>
- 1000ft limit, more or less
- 1-wire interface is DS2490 (USB) or DS9097U (serial)
- They can be read with owfs or digitemp



DALLAS
18B20
1016C4
+448AM





HB HOBBY BOARDS
CENTRAL TALKING SOLUTIONS

Passthrough Aux Channel 1 Aux Channel 2 Aux Channel 3

Power Channel 1 Channel 2 Channel 3

www.hobby-boards.com
6 Channel Hub

Main Channel 1 Main Channel 2 Main Channel 3 Main

g Identification C
phone Line Distribio

CAT
Voice
Mod

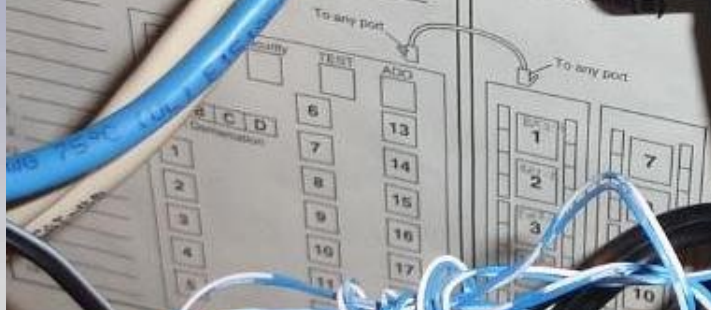
(Or affix business card here)

Coaxial Splitter

Cabled to:

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.
- 9.
- 10.

- ①
- ②







Misterhouse 1-wire interfacing.

- At least 4 ways to do it: xAP, xPL, Owfs.pm, and cacti_owfs script.
- Avoid all solutions where read(1) can block.

```
Reading owfs value saved to log file (non blocking):
```

```
if (new_minute) {
    $var = `cacti_owfs --value Family_Room`;
    chomp($var);

    if ($var eq "U") {
        $Weather{TempIndoor} = undef;
        print_log "MYLOG7: Weather: could not set family room temperature(undef)";
    } elsif ($var =~ /\d+/) {
        $var = int($var);
        $Weather{TempIndoor} = $var;
        #print_log "MYLOG7: Weather: set family room temperature to $var";
    } else { die "Could not get family room temp"; } } }
```

```
Or using xPL support in mh when running owfs and xpl-owfs to broadcast values:
XPL_SENSOR, bnz-owfs.*:28.998D4D020000, computer_closet_temp, , temp
```

```
my $compcloset_temp = $computer_closet_temp->state();
if ($compcloset_temp > 85) {
    $compcloset_fan->set(ON);
    print_log "MYLOGC: Computer closet: turning fan on (temp $compcloset_temp)";
}
```


Wireless: Oregon Scientific WMR200

- When running wires is impractical, wireless weather solutions exist. WMR200 is the cheapest.
- Not the best in class, but wind speed, wind direction, outside humidity and temp, rainfall, indoor humidity/temp/baro plus console for \$200.
- WMR200 console can be connected via serial port
- Rfxcom 433.92Mhz receiver or 433.92/310Mhz dual band for X10 too.
- Console is limited, but extra senders are cheap (baro/temp/humidity for \$40) and many can be received on a PC.
- Be careful, many sensors can be almost in sync and drift into periods of overlap.



Green
ELECTRONICS
E200N SERIES
WEATHER STATION
KEYWORD: CABLE FREE

63.5	50	00
63.5	47	00
0.00	0.00	0.00
13.120	35.3	





XPL

- XPL is a network protocol to broadcast sensor data
- Can be used to broadcast owfs/1-wire data (xpl-owfs)
- W800 can be read directly in mh or via xpl-w800
- Rfxcom (required for WMR 200 data packets) only works with xpl-rfxcom or Iranger's rfx-xpl.
- <http://misterhouse.wikispaces.com/X10RFX10SecRFXCOM>

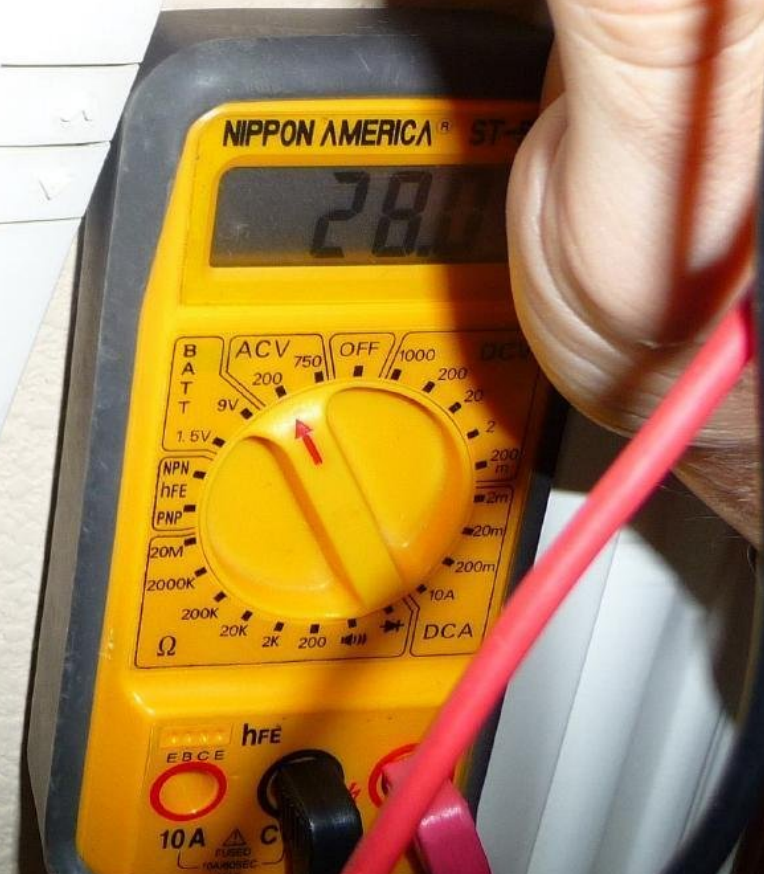
```
STR918,WGR918 addr: 232, chan: 0 wind: 33 ° NNE, speed 1.20 m/s / 2 kts average 1.00 m/s / 1 kts battery level 100%
RGR126,RGR682,RGR918 addr: 147, chan: 0 rain: total 1157 mm, 0 mm/hr, count 9
STR918,WGR918 addr: 232, chan: 0 wind: 48 ° NE, speed 1.60 m/s / 3 kts average 1.00 m/s / 1 kts battery level 100%
THGR918 addr: 225, chan: 1 temp: 23.70 °C / 74.66 °F hum: 43 % comfort battery level 80%
STR918,WGR918 addr: 232, chan: 0 wind: 49 ° NE, speed 1.60 m/s / 3 kts average 1.00 m/s / 1 kts battery level 100%
BTHR918N,BTHR968 addr: 230, chan: 0 temp: 26.50 °C / 79.70 °F hum: 45 % normal baro: 1014 Hg / 29.94 inHg, forecast: partly cloudy
STR918,WGR918 addr: 232, chan: 0 wind: 42 ° NE, speed 2.20 m/s / 4 kts average 1.00 m/s / 1 kts battery level 100%
THGR918 addr: 225, chan: 1 temp: 23.70 °C / 74.66 °F hum: 43 % comfort battery level 80%
STR918,WGR918 addr: 232, chan: 0 wind: 357 ° N, speed 1.40 m/s / 2 kts average 1.00 m/s / 1 kts battery level 100%
RGR126,RGR682,RGR918 addr: 147, chan: 0 rain: total 1157 mm, 0 mm/hr, count 9
```

HAI Thermostats (RC-80, Omnistat2)

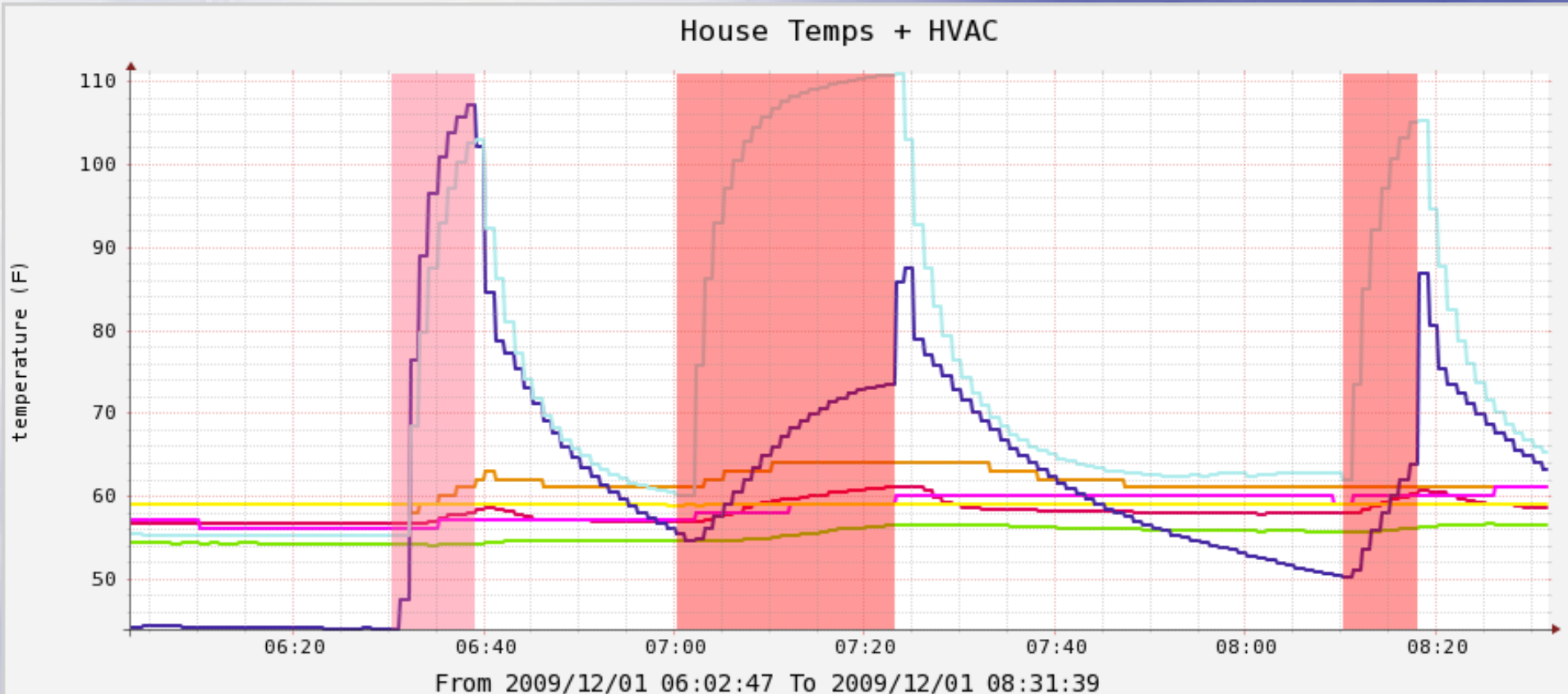
- HAI is a great company with hardware that's simple to communicate to, and published specs and protocols.
- Compatible with any 24VAC furnace/HVAC system.
- Serial port support, including serial port sharing.
- Old RC-80 stats are around \$50 on ebay, a bargain.
- New Omnistats2 are fancier, prettier, and compatible with the misterhouse support code too.
- Full heating/cooling schedule can be remotely programmed from a web page.
- Heating/cooling/fan status and setpoints can be read and sent
- Reading allows for controlling booster fans from mh.

6R91
UL C UL
LISTED
ENERGY MANAGEMENT
EQUIPMENT
FILE NUMBER E183082
13A00
CLASS 2, 24 Vac

1-B
2-C
3-D
4-PV
5-C
6-Y
7-W
8-B



HAI Thermostats (RC-80, Omnistat2)



Family Room	Current:	58.55	Average:	58.10	Minimum:	56.64	Maximum:	61.14
Living Room	Current:	56.52	Average:	55.30	Minimum:	54.05	Maximum:	56.64
Master Bedroom	Current:	61.00	Average:	61.26	Minimum:	58.00	Maximum:	64.00
Corridor	Current:	61.00	Average:	58.45	Minimum:	56.00	Maximum:	61.00
Hall / Wine Closet	Current:	58.89	Average:	58.90	Minimum:	58.77	Maximum:	59.00
MBR SubZone Vent	Current:	63.16	Average:	61.62	Minimum:	43.92	Maximum:	107.04
HVAC Vent	Current:	65.19	Average:	73.74	Minimum:	55.17	Maximum:	110.86
Main HVAC Heat								
Main HVAC Cool								
MBR HVAC Heat								
MBR HVAC Cool								

Interfacing HAI stats and Misterhouse

```
# update data once a minute, per omnistat offset seconds.
if ($New_Second and $Second eq $omnioffset[$omnistat]) {
    # we make the extended group1 call that also retrieves the stat's output status
    my ($cool_sp, $heat_sp, $mode, $fan, $hold, $temp, $output) = $omnistat[$omnistat]-
>read_group1("true");
    my $stat_type = $omnistat[$omnistat]->get_stat_type;

    # This mashes $hold and $mode together from registers cached in the group1 call and
    outputs a combined string
    $mode = $omnistat[$omnistat]->get_mode;

    Omnistat::omnistat_log("".$omniname[$omnistat]." Omnistat $stat_type: Indoor temp
    is $temp, HVAC Command: $output, heat to $heat_sp, cool to $cool_sp, mode: $mode");

    # only store the temperature from the first stat (which we'll assume is master)
    $Weather{TempIndoor} = $temp if ($omnistat == $omnolist[0]);
}

if ($state = $v_omnistat_fan[$omnistat]->said) {
    $omnistat[$omnistat]->fan($state);
}

if ($state = $v_omnistat_cool_sp[$omnistat]->said) {
    $omnistat[$omnistat]->cool_setpoint($state);
    speak "Air conditioning set to $state degrees for $omniname[$omnistat] Omnistat";
    Omnistat::omnistat_log("$omniname[$omnistat] Omnistat: Air conditioning set to
    $state degrees", 2);
}
```


Power Monitoring: Brultech ECM1240

- Uses an AC to AC converter to pick up phase offset between current and voltage (power factor).
- 120 and 240V compatible.
- Can sample 7 currents, 2 of which directional.
- Lots of CTs (current transducers) to choose from.
- Serial, ethernet, or Zigbee (wireless) interface.
- Cheap (starts around \$200).
- Needs windows to configure, but platform independent python code to query the data (up to once a second).



us E325096
Britech Research Inc

MICRO-4
us E325096
Britech Research Inc

AD-9123
ISSUE NO.
HACR TYPE
TPO-CARR
SMD
LISTED
C.B.
A/CU
10 KA
120/240 V
TYPE HOM

AB-4710
ISSUE NO.
HACR TYPE
TPO-CARR
LISTED
C.B.
A/CU
10 KA
120/240 V
TYPE HOM

10 KA
Non-Common Trip
Poles extencors.
disparo no comun
pôles extencurs
dechenement
non-common

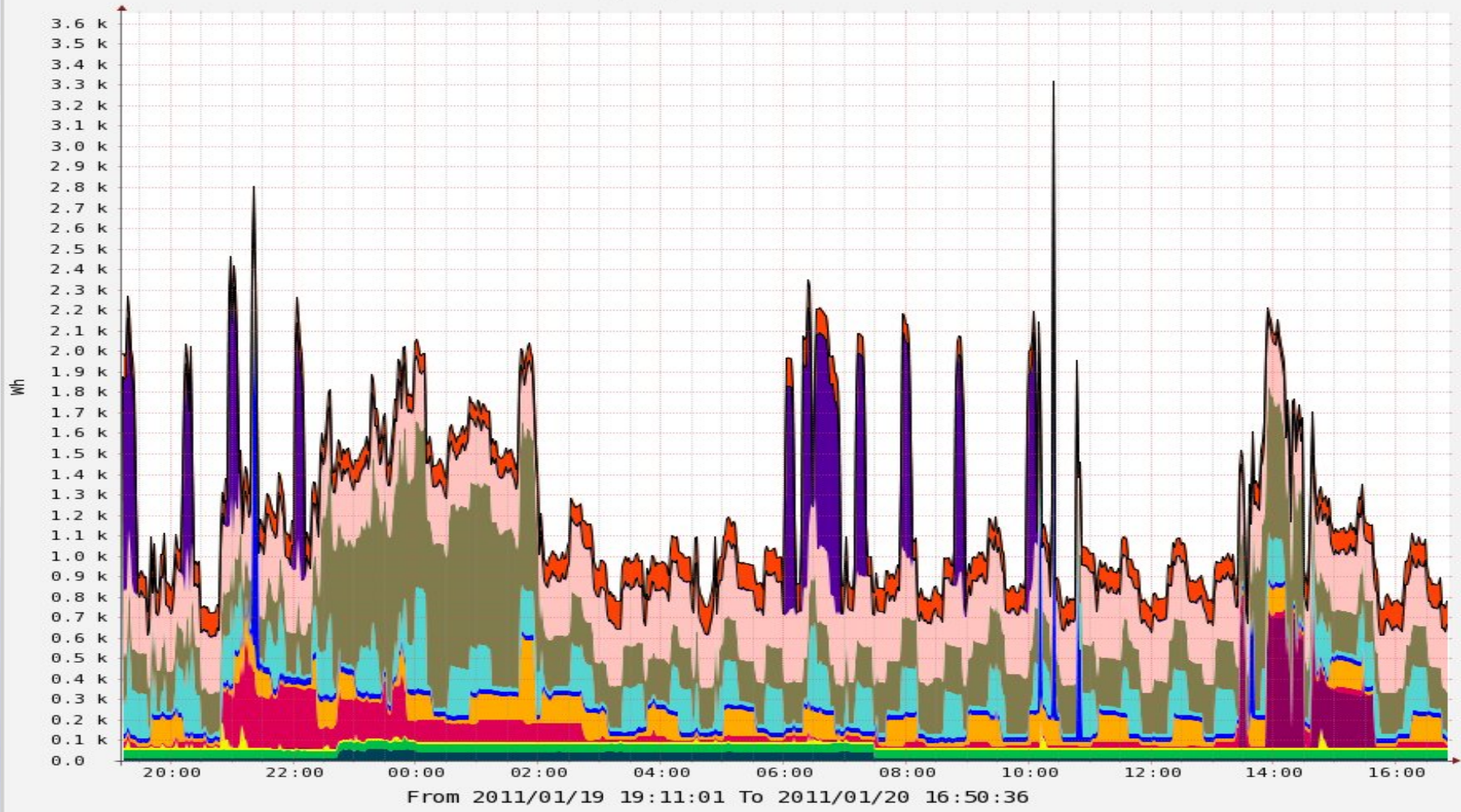
8162
TYPE/TPO HOMT
(1) 2P & (2) 1P
LISTED C.B.
AD-4348
ISSUE NO.
HACR TYPE
TPO-CARR
120/240 V - CTL
488-300004501

ECM1240 Logs

- Samples Voltage and 7 Watt.second counters
- The 2 main channels count amps in both directions (solar panels).
- Multiple CTs can be linked to one channel (house lights).

```
2010/05/28 20:00:21: Volts:                120.00V
2010/05/28 20:00:21: Ch1 Watts:            -2.104KWh ( 1995W) < PG&E
2010/05/28 20:00:21: Ch1 Positive Watts:   78.890KWh ( 1995W)
2010/05/28 20:00:21: Ch1 Negative Watts:  80.995KWh (  0W)
2010/05/28 20:00:21: Ch2 Watts:           -140.897KWh ( -14W) < PV
2010/05/28 20:00:21: Ch2 Positive Watts:   0.012KWh (  0W)
2010/05/28 20:00:21: Ch2 Negative Watts: 140.909KWh ( 14W)
2010/05/28 20:00:21: Aux1 Watts:          40.387KWh ( 306W) < Computer Closet
2010/05/28 20:00:21: Aux2 Watts:          31.272KWh ( 200W) < MythTV/AV System
2010/05/28 20:00:21: Aux3 Watts:          4.456KWh (  51W) < Computer Office/BR4
2010/05/28 20:00:21: Aux4 Watts:          0.000KWh (  0W) < AC
2010/05/28 20:00:21: Aux5 Watts:          8.665KWh (  20W) < Kitchen Fridge
```

Aggregate House Use (ECM)



■ LVR Plugs/Kitchen Vent	Current:	12.9 Wh	Average:	25.2 Wh	Min:	10.4 Wh	Max:	59.1 Wh
■ Computer Office/BR4	Current:	39.0 Wh	Average:	39.5 Wh	Min:	38.0 Wh	Max:	63.3 Wh
■ Garage Plugs1	Current:	11.0 Wh	Average:	13.3 Wh	Min:	10.7 Wh	Max:	134.9 Wh
■ Washer/Dishwasher	Current:	0.7 Wh	Average:	32.8 Wh	Min:	0.0 Wh	Max:	675.7 Wh
■ All Lights	Current:	9.0 Wh	Average:	60.4 Wh	Min:	7.8 Wh	Max:	515.6 Wh
■ Kitchen Fridge	Current:	17.0 Wh	Average:	74.2 Wh	Min:	15.8 Wh	Max:	412.6 Wh
■ Kitchen Plugs/Microwave	Current:	22.2 Wh	Average:	38.8 Wh	Min:	20.6 Wh	Max:	2514.4 Wh
■ Garage Plugs2/Fridge/Yard Lights	Current:	20.1 Wh	Average:	119.4 Wh	Min:	16.2 Wh	Max:	500.1 Wh
■ MythTV/AV System/FMR Plugs	Current:	199.4 Wh	Average:	325.9 Wh	Min:	195.7 Wh	Max:	851.7 Wh
■ Computer Closet	Current:	326.1 Wh	Average:	307.1 Wh	Min:	244.8 Wh	Max:	380.8 Wh
■ Furnace and Fans	Current:	8.0 Wh	Average:	105.6 Wh	Min:	7.7 Wh	Max:	1108.3 Wh
■ AC	Current:	0.0 Wh	Average:	0.0 Wh	Min:	0.0 Wh	Max:	0.0 Wh
■ Unmonitored House Use	Current:	116.4 Wh	Average:	103.2 Wh	Min:	-11.7 Wh	Max:	141.5 Wh
■ ALL ITEMS	Current:	0.0 Wh	Average:	1139.2 Wh	Min:	0.0 Wh	Max:	3264.6 Wh
■ Total House Use	Current:	781.8 Wh	Average:	1245.4 Wh	Min:	724.6 Wh	Max:	3318.0 Wh

Ok, great, so why all this?

- Ok, we're geeks, we interface things just because we can :)
- But it's better if you have a plan of what you're going to do with the data:
 - Graph it and look at trends, find spikes/problems
 - Correlate electricity use with certain controlled loads
 - Control sprinklers when your grass humidity sensor says you should, and you haven't detected rain.
 - Control fans based on room temperature
 - Correlate computer closet temp and electricity use with raid checks or kernel rebuilds
 - Control lights on based on motion sensors, time of day, and counting overrides from local switches.

Computer Closet Temperature Control

```
my $last_compcloset_temp = 200;

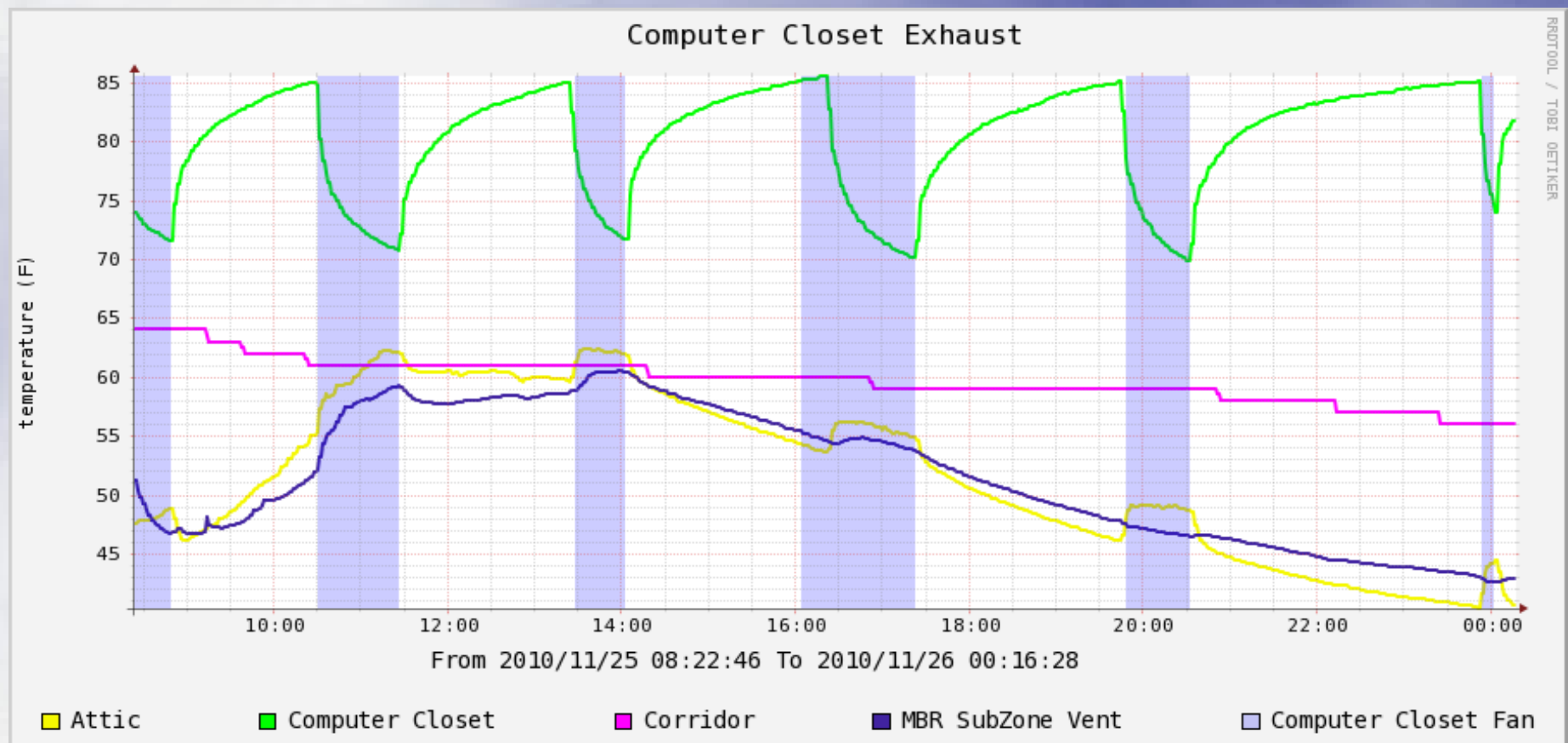
if (new_second 122)
{
    my $compcloset_temp = $computer_closet_temp->state();
    my $tempdiff = $last_compcloset_temp - $compcloset_temp;
    my $state = $compcloset_fan->state;

    if ($compcloset_temp < 75) {
        $compcloset_fan->set(OFF);
        print_log "Computer closet: turning fan off (temp $compcloset_temp)";
    }

    if ($compcloset_temp > 85) {
        $compcloset_fan->set(ON);
        print_log "MYLOGC: Computer closet: turning fan on (temp $compcloset_temp)";
    } elsif ($compcloset_temp < 80 and $tempdiff < 0.01) {
        $compcloset_fan->set(OFF);
        print_log "MYLOGC: Computer closet: turning fan off because it's not cooling anymore (temp $compcloset_temp and tempdiff is $tempdiff)" if($state eq ON);
    } else {
        print_log "MYLOGC: got computer closet temp $compcloset_temp, last was $last_compcloset_temp and diff is $tempdiff. Fan state is $state";
    }
    $last_compcloset_temp = $compcloset_temp;
}
```

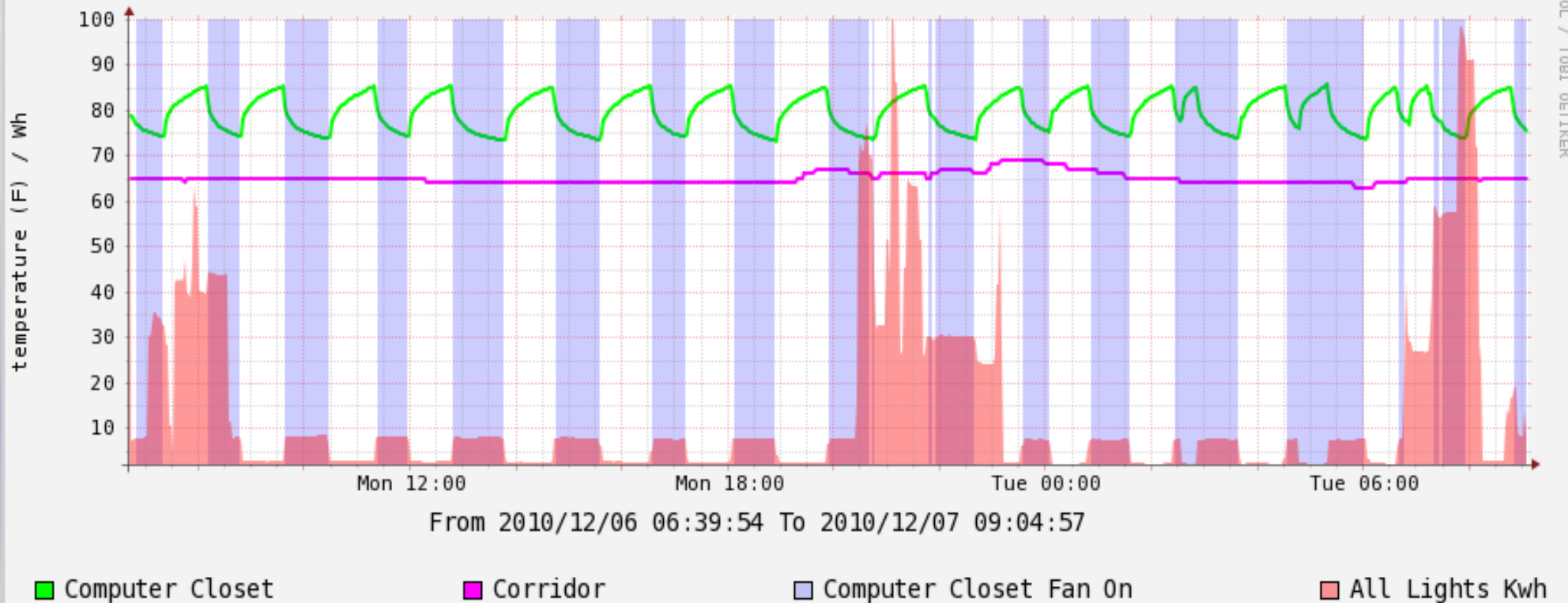
Computer Closet Temperature Control

- Why cool all the time as opposed to only when needed?
- In winter, house may be cool enough for passive cooling.
- Active cooling can be done in waves, like a fridge.



Composite graphs

Computer Closet Fan vs Power Used on All Lights Breaker



Cooling with Outside Air

- Automating opening windows and doors to get cool air inside at night.
- Without opening your doors and windows for security reasons.
- Only if inside – outside $> 5^{\circ}\text{C}$.
- Only if tomorrow is going to be warmer than X.
- Only if AC isn't already running.
- Oh, and how about managing 2 zones and resulting air pressure issues?



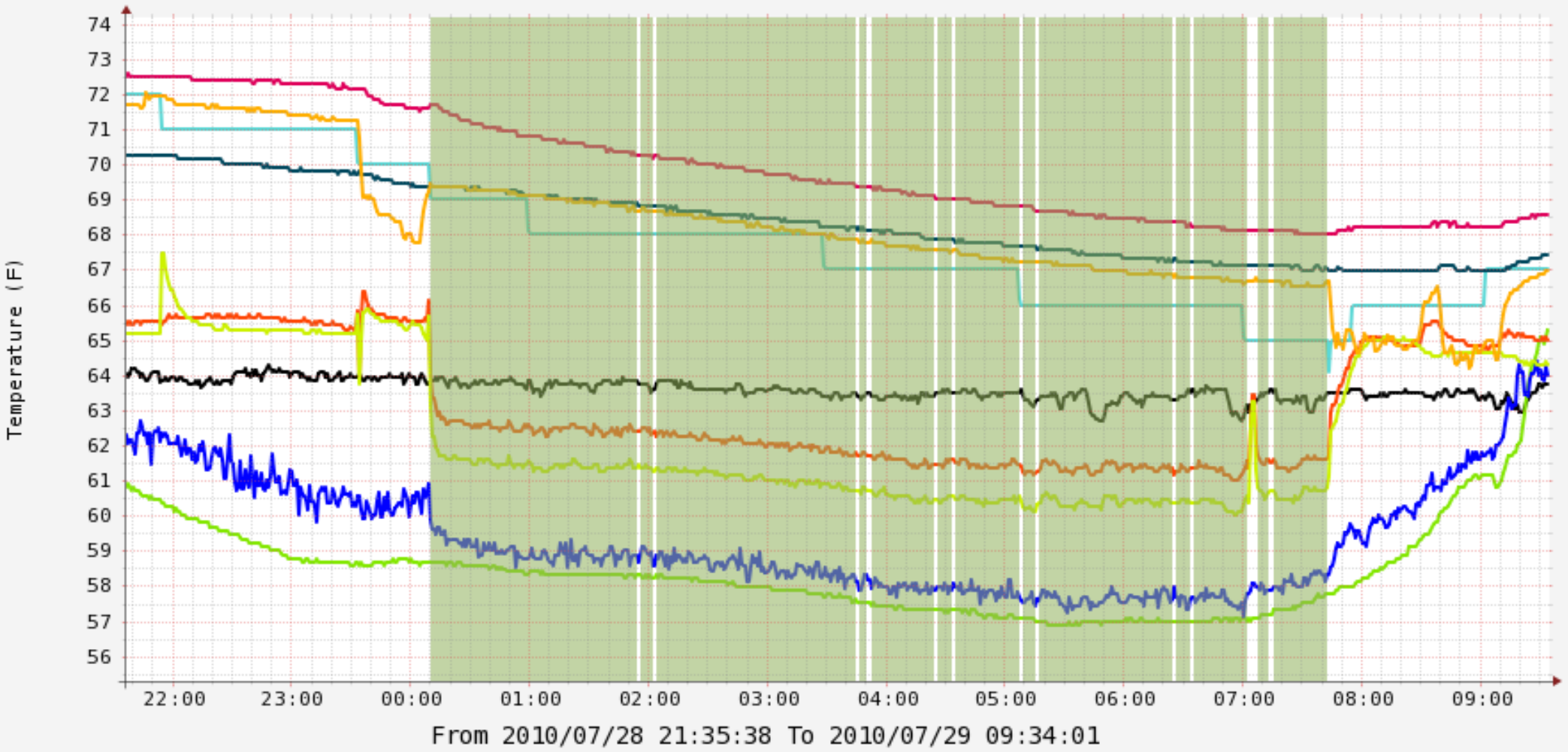
Cooling Code

A bit long to paste here, so find it on its web page:

<http://marc.merlins.org/perso/linuxha/2010-08.html>

<http://marc.merlins.org/perso/linuxha/2010-08.html#Booster-Fans-and-Heating-Cooling-with-Outside-Air-with-Misterhouse>

Cooling with Outdoors Air



■ Crawlspace	Current: 63.62	Average: 63.59	Minimum: 62.71	Maximum: 64.28
■ Outdoors by Roof	Current: 65.61	Average: 58.41	Minimum: 56.86	Maximum: 65.61
■ Outdoors by Crawlspace	Current: 64.27	Average: 59.32	Minimum: 57.13	Maximum: 64.38
■ Main HVAC Vent	Current: 70.36	Average: 66.44	Minimum: 62.66	Maximum: 73.97
■ FMR Vent	Current: 64.96	Average: 63.12	Minimum: 61.02	Maximum: 66.37
■ MBR Vent	Current: 64.39	Average: 62.40	Minimum: 60.02	Maximum: 67.46
■ Master Bedroom	Current: 67.00	Average: 67.77	Minimum: 64.07	Maximum: 72.00
■ Living Room	Current: 67.44	Average: 68.33	Minimum: 66.99	Maximum: 70.25
■ Dining Room	Current: 68.56	Average: 69.83	Minimum: 68.00	Maximum: 72.60
■ Family Room	Current: 66.99	Average: 68.10	Minimum: 64.20	Maximum: 72.02
■ FMR Booster Fan On				
■ MBR Booster Fan On				
■ Dampers Opened to Outside Air				

Teach the stupid cat some manners

- Sneaky cat knows he can't go on the kitchen counter, but he also knows he won't get caught by us at night.
- Motion sensor to detect movement at night and turn the lights on.
- Use webcam to monitor the vulnerable kitchen counter
- Use motion(1) to detect movement, but not light changes
- Tell motion(1) to run a shell script when appropriate movement is detected and trigger misterhouse code.
- Misterhouse code turns blender on and loud sound
- Profit!

<http://marc.merlins.org/perso/linuxha/2009-03.html#Insteon-Setup-and-my-own-Blender-Defender>



MS1B
DCS 900



Teach the stupid cat some manners!



AUTHORITAH

It's what sets apart a lard-butt in a State-issued costume from his
lard-butt brethren



2008-08-08
04:10:27-08

Smart designs

- Supplement, not replace thermostat and light switches (kind of nice to be able to turn lights on when your controlling computer is down).
- Have a “wife override” :) (it will break when you're not home).
- More generally, allow local control and programming that can be taken over by the computer when it works.
- Is your fish or your cat going to die if something fails when you're not home?
- Why not use arduino?
- Think about maybe selling the place one day: how much will still work without your master controller?

Misterhouse programming tips

- Code gets integrated in main misterhouse event loop.
- Any hang of more than 1sec is bad. It can drop event/alarm driven data like Insteon ACKs. Do not use sleep.
- my \$var=1 is tricky, it gets taken out of the loop for my, and =1 does not get re-applied each time around. Prefer my \$var; \$var=1 on the next line.
- Foocmd() #noloop
- #noloop=start
- cmd1; cmd2
- #noloop=stop
- print_log("Foo without newline") goes to mh/data/logs/print.log
- print_log("logs/foo.log mymsg") goes to mh/data/logs/foo.log

Example blend of MH calls and syntax:

```
my $mhsend = '/var/local/src/misterhouse/mh/bin/mhsend';
my $alarm = '/var/local/scr/alarm';

print_log("MYLOGCAM: Sunset") if (time_now("$Time_Sunset"));
print_log("MYLOGCAM: Sunrise") if (time_now("$Time_Sunrise"));

if (new_second) {
    my $backups = 60*24*7; # 10080
    if ($Second eq 0){
        system("cd $CAM1; cp $img '$img.sav'; savelog -c $backups -l -q -d $img; mv
'$img.sav' $img");
    }

    if (($garage1_neon_kpl->state eq ON) or ($garage2_neon_kpl->state eq ON) or
($garage_incandescent->state eq ON) or (time_greater_than("$Time_Sunrise+0:30")
and time_less_than("$Time_Sunset-0:45"))) {
        #print("MYLOGCAM: Taking picture with light present\n");
        system("bash -c 'cd $CAM1; $alarm 3 wget --user=foo --password=bar -O
$imgdl $cam1_url; mv $imgdl $img' &>/dev/null &") if ($Second % 5 eq 0);
    }
    else # garage is dark
    {
        # Once every 5 minutes.
        if ($Minute % 5 eq 0 and $Second eq 50){
            print_log("MYLOGCAM: Taking picture after turning light on");
            $garage_incandescent->set(ON);
            system("bash -c 'cd $CAM1; sleep 5; $alarm 3 wget --user=foo
--password=bar -O $imgdl $cam1_url; mv $imgdl $img; $mhsend --run garage
incandescent off' &>/dev/null &");
        }
    }
}
}
```

References

- <http://misterhouse.wikispaces.com/>
- <http://misterhouse.wikispaces.com/Insteon>
- <http://misterhouse.wikispaces.com/Z-Wave>
- <http://misterhouse.wikispaces.com/X10RFX10SecRFXCOM>
- http://marc.merlins.org/perso/linuxha/post_2010-06-25_Rfxcom_-W800_-
- <http://misterhouse.wikispaces.com/xAP+and+xPL+-+Getting+Started>
- http://marc.merlins.org/perso/linuxha/2010-08.html#Temperature_-moistu
- http://marc.merlins.org/perso/linuxha/post_2009-12-23_Temperature-mon
- http://marc.merlins.org/perso/linuxha/post_2009-03-16_Insteon-Setup-and
- <http://marc.merlins.org/linux/cacti/>

Thanks

- Too many to name all, but a few:
- Bruce Winter wrote the original code and maintained until 2005.
- Many like Gregg Liming (insteon and many others), David Norwood, Jim Duda, and many others from before my time playing with misterhouse.
- The rest of the misterhouse community for patches, and community help.

Questions?

- This talk and documents:

<http://marc.merlins.org/linux/talks/MisterhouseUseCase/>