The 1-wire (w1) subsystem

The 1-wire bus is a simple master-slave bus that communicates via a single signal wire (plus ground, so two wires).

Devices communicate on the bus by pulling the signal to ground via an open drain output and by sampling the logic level of the signal line.

The w1 subsystem provides the framework for managing w1 masters and communication with slaves.

All w1 slave devices must be connected to a w1 bus master device.

Example w1 master devices:
    DS9490 usb device
    W1-over-GPIO
    DS2482 (i2c to w1 bridge)
    Emulated devices, such as a RS232 converter, parallel port adapter, etc

What does the w1 subsystem do?

When a w1 master driver registers with the w1 subsystem, the following occurs:

 - sysfs entries for that w1 master are created
 - the w1 bus is periodically searched for new slave devices

When a device is found on the bus, w1 core checks if driver for its family is loaded. If so, the family driver is attached to the slave.
If there is no driver for the family, default one is assigned, which allows to perform
almost any kind of operations. Each logical operation is a transaction in nature, which can contain several (two or one) low-level operations.
Let's see how one can read EEPROM context:
1. one must write control buffer, i.e. buffer containing command byte and two byte address. At this step bus is reset and appropriate device is selected using either W1_SKIP_ROM or W1_MATCH_ROM command.
Then provided control buffer is being written to the wire.
2. reading. This will issue reading eeprom response.

It is possible that between 1. and 2. w1 master thread will reset bus for searching
and slave device will be even removed, but in this case 0xff will be read, since no device was selected.

W1 device families

Slave devices are handled by a driver written for a family of w1 devices.

A family driver populates a struct w1_family_ops (see w1_family.h) and registers with the w1 subsystem.

Current family drivers:
w1_therm - (ds18?20 thermal sensor family driver)
    provides temperature reading function which is bound to ->rbin() method

of the above w1_family_ops structure.

w1_smem - driver for simple 64bit memory cell provides ID reading method.

You can call above methods by reading appropriate sysfs files.

What does a w1 master driver need to implement?

The driver for w1 bus master must provide at minimum two functions.

Emulated devices must provide the ability to set the output signal level
(write_bit) and sample the signal level (read_bit).

Devices that support the 1-wire natively must provide the ability to write and
sample a bit (touch_bit) and reset the bus (reset_bus).

Most hardware provides higher-level functions that offload w1 handling.
See struct w1_bus_master definition in w1.h for details.

w1 master sysfs interface

```
<xx-xxxxxxxxxxxx> - a directory for a found device. The format is family-serial
bus               - (standard) symlink to the w1 bus
driver            - (standard) symlink to the w1 driver
w1_master_add     - Manually register a slave device
w1_master_attempts - the number of times a search was attempted
w1_master_max_slave_count
                  - the maximum slaves that may be attached to a master
w1_master_name    - the name of the device (w1_bus_masterX)
w1_master_pullup  - 5V strong pullup 0 enabled, 1 disabled
w1_master_remove  - Manually remove a slave device
w1_master_search  - the number of searches left to do, -1=continual (default)
w1_master_slave_count
                  - the number of slaves found
w1_master_slaves  - the names of the slaves, one per line
w1_master_timeout - the delay in seconds between searches
```

If you have a w1 bus that never changes (you don't add or remove devices),
you can set the module parameter search_count to a small positive number
for an initially small number of bus searches. Alternatively it could be
set to zero, then manually add the slave device serial numbers by
w1_master_add device file. The w1_master_add and w1_master_remove files
generally only make sense when searching is disabled, as a search will
redetect manually removed devices that are present and timeout manually
added devices that aren't on the bus.

w1 slave sysfs interface

```
bus               - (standard) symlink to the w1 bus
driver            - (standard) symlink to the w1 driver
name              - the device name, usually the same as the directory name
w1_slave          - (optional) a binary file whose meaning depends on the
```

family driver

rw - (optional) created for slave devices which do not have appropriate family driver. Allows to read/write binary data.