MoDisp: IoT-enabled Modular Display System

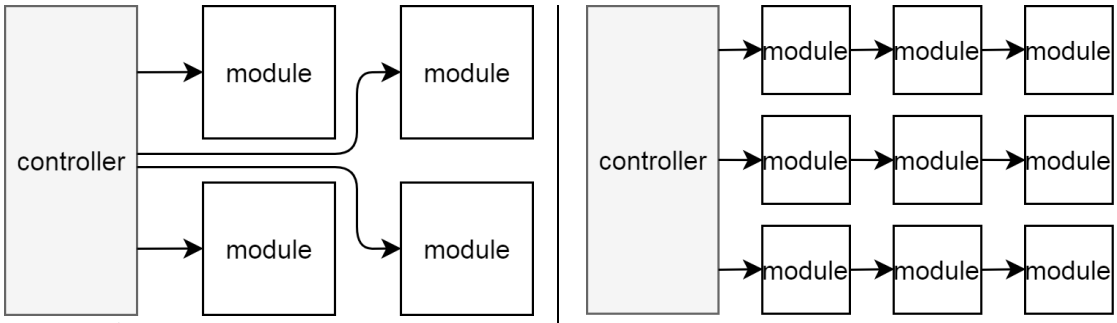
Advisor: Ching-Hu Lu (Department of Electrical Engineering, National Taiwan University of Science and Technology)
Members: YU, TZU-HSU (EE, NTUST)
CHAN, YEN-HSUN (EE, NTUST)
CHENG, HUNG-CHENG (EE, NTUST)

Abstract

MoDisp is a modular display system that outperforms typical large display systems in **scalability** and **reliability** by providing features like **hot-plug**, **configuration detection**, **error detection**, **auto-reroute**, etc.

Main concept

Typical large display systems can be classified into the following 2 topologies:

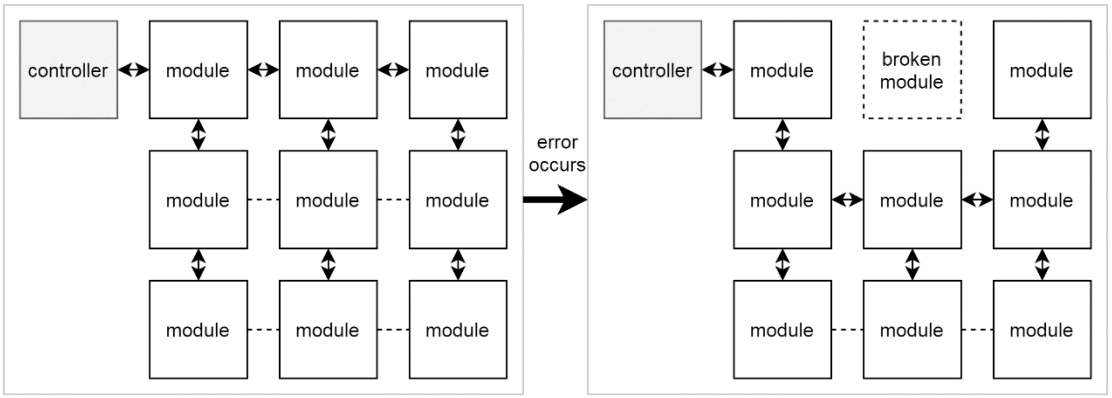


▲Typical large display system (star topology)

▲Typical large display system (star topology and daisy-chain)

These systems are **poor** in **scalability** (limited number of ports on the controller) and **reliability** (a single broken module in a daisy-chain can cause many to malfunction).

But what if each module has the ability to **route packets**? This is the main idea of MoDisp, which has **dynamic tree topology**.



▲auto-rerouting and dynamic tree topology

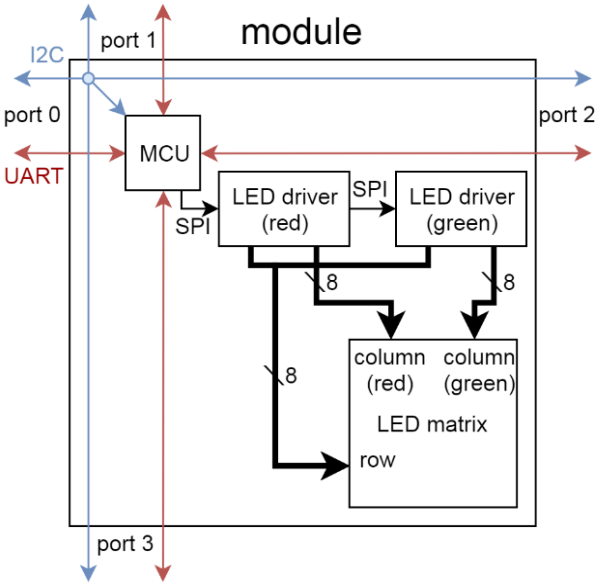
Compared with typical systems, MoDisp has **improved scalability** (the controller needs **only 1 port** and number of modules is only limited by single-port bandwidth) and reliability (**error detection** and **auto-reroute**). Moreover, MoDisp allows the display to be **free-form** (any connected configuration) and **non-uniform** (e.g. RGB LED matrices mixed with single-color ones), which is useful in some applications like screen walls of irregular shape and installation art etc.

Details

Each **module** has 4 **UART**[a] connection with its neighbors, these connections are used to transfer **control packets**.

Besides that, an **I²C bus** (controller as master)[b] is used to deliver **display contents**, **synchronization signal**, and **reset command** from the controller.

The microcontroller unit (MCU) is chosen to be ATMEGA328P, which is affordable and easy to get, and so are the LED drivers MAX7221[c].



▲inside the module

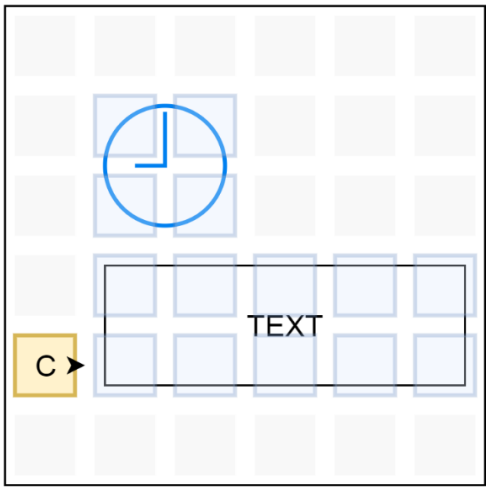
[a] UART with Inverted logic. This enables the MCU to differentiate between idle ports and unconnected ports.

[b] The I²C bus is unidirectional (controller -> modules) and not for control packets, for the reason that multiple modules may request to transfer control packets on loss of connection at the same time and thus complicated arbitration will be needed.

[c] There are too many malfunctioning Chinese clones on the market, so we decided to use an alternative driver AS1107 for replacement, which is fully compatible with MAX7221.

The **controller** only has 1 port, and it also serves as a **web server** for **user interface (UI)**. There are 2 main functions in the UI:

1. **Organize function of the display**
   By **drag-and-dropping** and resizing widgets (much like the way we organize **smartphone home screens**), users can decide the function of each area.
2. **Show currently connected modules**
   A transparent layer will be placed above the "home screen", which is not editable but shows the position of each module relative to the controller.



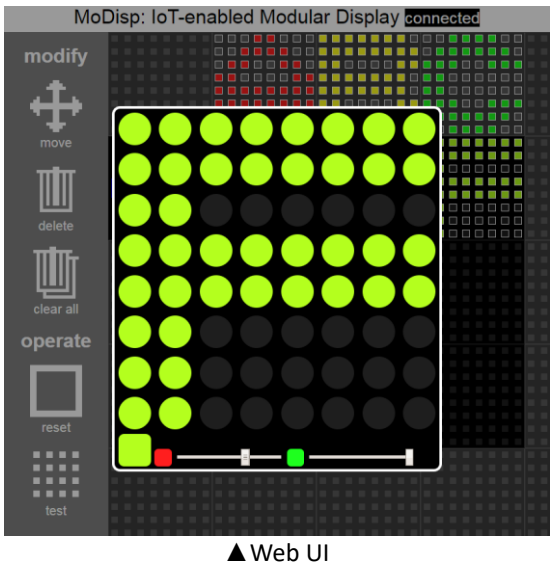
▲web UI

We chose C.H.I.P.[d], an open hardware computer running Linux, to act as the controller.

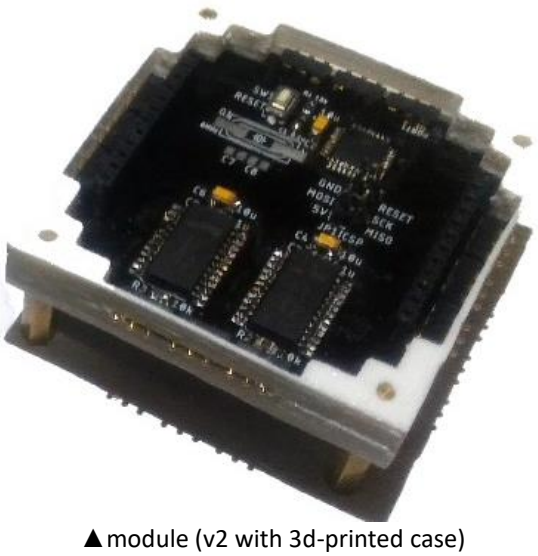
The **algorithm of auto-rerouting** can be found in the works of P. Bauwens and J. Doutreloigne [1] [2].

Current state

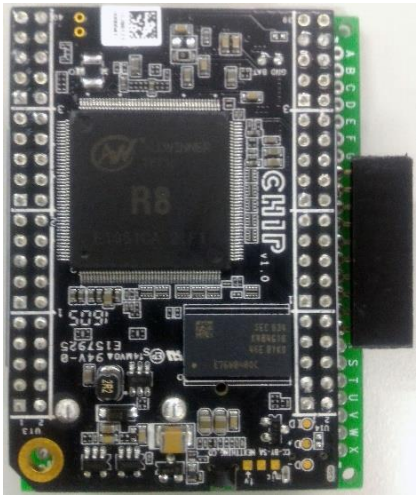
The prototype has won 2 awards, and is still under development.
A demonstration video is available at https://youtu.be/CasiWIhfjWg



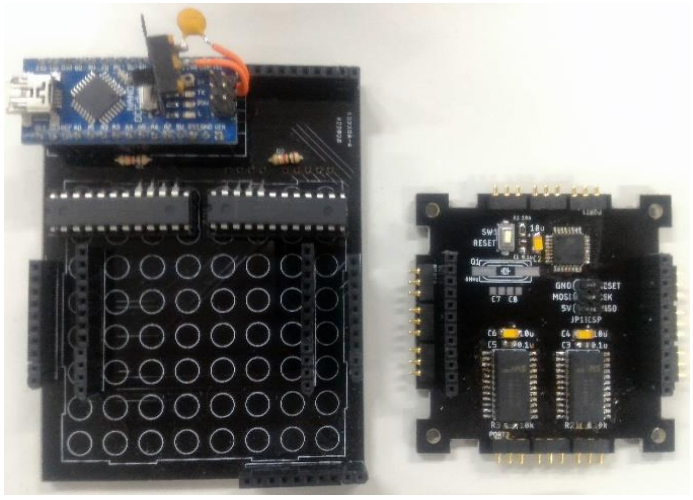
▲Web UI


▲module (v2 with 3d-printed case)

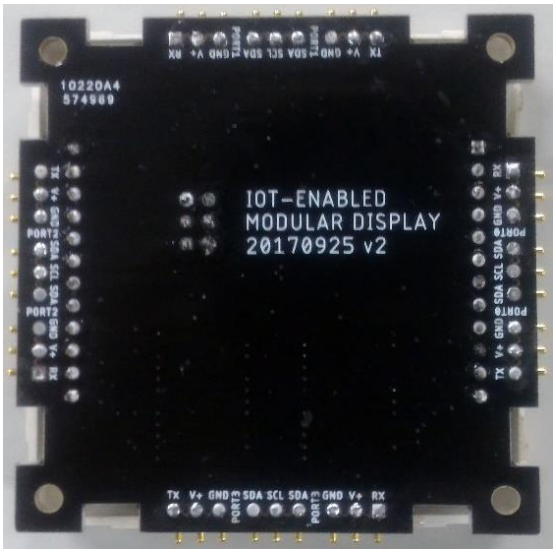
[d] https://getchip.com/



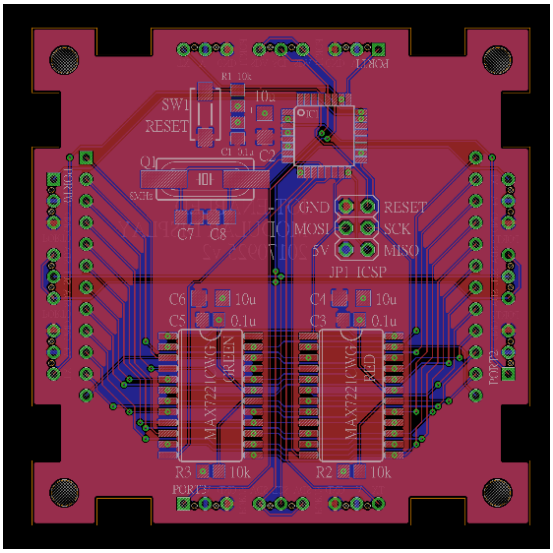
▲controller (bottom)



▲module (v1 v.s. v2)



▲module (v2 bottom)



▲module (v2 layout)

Division of work

Hardware:	YU, TZU-HSU
Firmware(controller):	YU, TZU-HSU
Firmware(module):	CHAN, YEN-HSUN
User interface:	YU, TZU-HSU; CHENG, HUNG-CHENG

References

[1] P. Bauwens and J. Doutreloigne, "FrIIDoM: An intelligent driver for automatic configuration in modular display systems," International Journal of Circuit Theory and Applications, vol. 42, no. 12, pp. 1246-1263, 2014.

[2] P. Bauwens and J. Doutreloigne, "Drivers for free-form modular displays," Journal of the Society for Information Display, vol. 18, no. 3, pp. 235-239, 2010.