Best Awarded 2011
Embedded Technology exhibition
at Yokohama Pacifico

High Resilience system、FUJIMI

Non stop microcomputer system

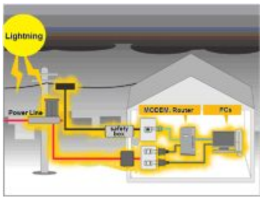
Recover transparently from microcontroller malfunctions

The micro-computer/controller (MCU) is common everywhere: in homes, factories, automobiles and so on. It is an unseen necessity of our modern life. However, the MCH has an inherent problem. There are malfunctions in which the MCU is not damaged, but does not operate as intended. This occurs often.

[Examples from everyday life]

You have probably experienced these problems:

- A modem could not connect to the Internet, or a router was out of operation
  - Can be caused by lightning-induced surges and ESD (Electrostatic Discharge).
- A smart phone would not respond or it became very hot.
  - Can be caused by ESD.
- A lift (elevator) did not stop at the desired floor, or the door of a lift opened while it was moving.
  - Can be caused by electrical noise from the electric motor.
- An air conditioner did not cool.
  - Can be caused by power line noise.
- A control in your automobile malfunctioned.
  - Can be caused by radiation from a police radar or the electric field from nearby lightning.



Surge from a lightning nearby.

The Lift bite a boy with his bike by a malfunction

In a factory or an outdoor facility:

- Measurement equipment gave readings out of range or stopped measuring
  - Can be caused by nearby lightning, or ESD intrusion.
- A robot behaved strangely or stopped operating.
  - Can be caused by electric arcs from a nearby welder.
- An automated production line made defective products, or halted.
  - Can be caused by surge voltage on the power line or ESD from a belt conveyer.
- A thermostat ignored its set temperature.
  - Can be caused by nearby lightning, or surges on the power line.

Sparks at a welder

ESD Arc from your finger

A lightning hit a wind sensor

The causes given above are among the most common; these problems can have other causes as well. The examples above, both from daily life and industry, can usually be solved if you can power the device off and on once; then the malfunction will be cleared and functioning returned to normal.

These types of problems occur in almost all microcomputers/controllers (MCU), and we use terms such as “freeze”, “disruption”, “hang” and so on. However, these are not real failures of the MCU since it was not permanently damaged.　They are temporary abnormalities in operation of a microcomputer. (Of course occasionally, in rare cases, there are real permanent hardware failures).

[Causes of the problem]

One of the root causes of the above problems is an electrical signal (noise) beyond the limited operating voltage range of the MCU.
Once such high voltages enter a MCU, it cannot operate as expected since it is fabricated from a semiconductor, which has many very stringent operating limitations. What are these electrical noise signals?　The most common one is ESD (Electro Static Discharge).
There are many other noise sources, such as nearby lightning, power transients from motors and solenoids, radio frequency interference (RFI), and electromagnetic pulse (EMP) disturbances.

Electrostatic charges are easy to generate when the air is very dry. In the winter, when you put your sweater on, you can see small sparks and hear the crackle of static. Even these small sparks can have a voltage of several thousand volts (KV). Your body can hold an electrostatic charge of over 10KV.
When this high voltage ESD reaches a MCU, a MCU designed to operate at only a few volts, normally 3V to 5V, it is to be expected that the MCU halts or operates abnormally. The good thing is, usually the energy content of such ESD is quite small and cannot permanently damage the MCU itself. When you pick your smartphone up from the desk, if your body is holding a large electrostatic charge (from walking across a carpet, perhaps), there can be a small electric arc from your finger to the smartphone, and the smartphone may malfunction, with a frozen screen or no response to the keys. In the worst case, the smartphone may become very hot.

An internet modem and router are connected by copper wires to their power sources and Ethernet connections. These power lines and Ethernet signal lines can act as antennas for electromagnetic pulses caused by lightning. If this noise enters the equipment, it can easily cause a freeze.

[Current status]

This MCU problem is well known to embedded-system design engineers, who employ various methods of circuit protection. However, these cannot completely avoid the problem. Various systems and equipment still incur errors or halt. This is caused by the very high voltages and unpredictable timing of external events such as surges from lightning or ESD, against necessarily limited protection capabilities. There is no perfect protection.

[Internal cause]

In a micro-computer, the principal active part is the CPU (Central Processing Unit), analogous to the human brain. This CPU reads instructions from the program and executes the desired operations. The CPU is affected by noise which can alter the order of instruction execution. This is the most common cause of system malfunction. There is only one way to recover from this situation: a Reset, which initializes the part just as when power is first turned on. Once this Reset signal is given, the CPU returns to normal operation and the system malfunction disappears.

[A solution to the problem]

Here, we introduce you one solution to these system malfunctions, called FUJIMI. In a FUJIMI system, the CPU is reset perhaps one hundred times per second. These CPU resets allow the system to return to normal from otherwise inescapable malfunctions caused by electrical noise, such as ESD, in a time so brief that it is transparent to the system’s user.

If you are a computer system designer, you may worry about the time taken by these Reset operations. There is no need to worry. The CPU core is reset and no other part is affected. This means, in a MCU which has been disrupted, usually data in the SRAM or peripherals are not affected. Furthermore, the overhead of these Reset operations is less than 3 percent of total CPU execution time, so it can be ignored.

[FUJIMI-enhanced MCUs]

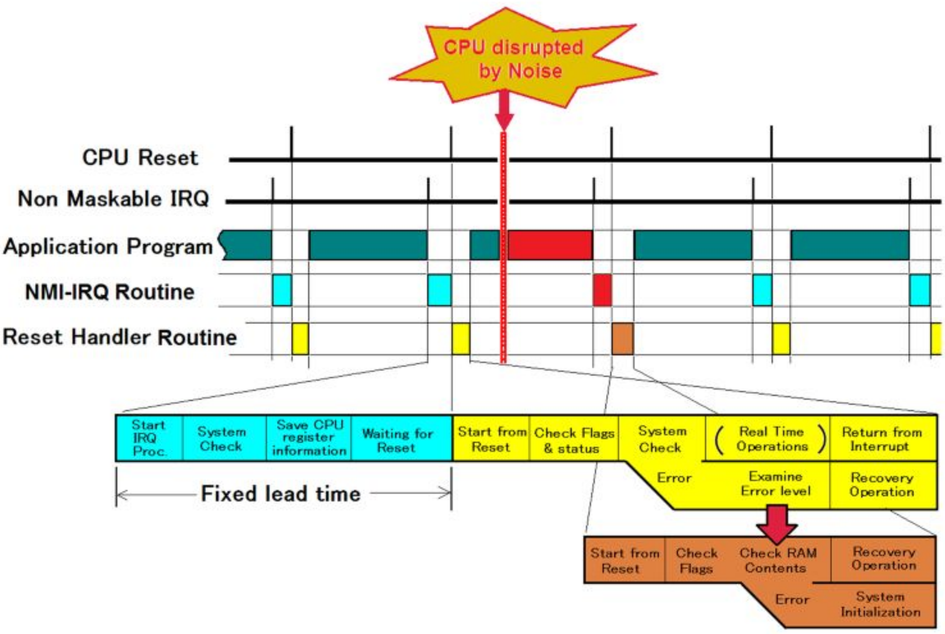
To build such a high resilience system, the MCU must be equipped with special features including a patented timer circuit.
Below are MCUs currently available with FUJIMI; please contact us for updates or any questions.

- Spansion Innovates Ltd. FM3 Series, FM4 series and FM0+ series (Selected versions)
- Dragonchip Ltd. Hong Kong. DC6688 Series (8051 core) (Selected versions)
- Freescale Semiconductor Ltd. LS1020A (Cortex A7 Dual core)

[Evolution of FUJIMI technology]

In many MCU applications, a process will be started by a periodic interrupt from a timer, called a “Real Time Interrupt” (RTI). Instead of this interrupt, we use a periodic CPU core Reset, preceded by a unique preparatory interrupt, which allows the CPU to be returned to a known safe condition within a predictable time period; if the CPU is disrupted causing a system malfunction, the CPU and system will be restored to normal operation within at most this period.

However, if we simply give the CPU a Reset signal directly, as with, for example, a watchdog timer, the sequence of program operation may make it difficult to return to normal operation. So, we originated the idea for a more comprehensive failure recovery method, which we have patented. In brief, we hide the reset routine within an interrupt routine, as shown below. This is a user-transparent Reset system which we call “FUJIMI”.



[Explanation of FUJIMI]

At first, the dedicated timer shall generate the interrupt, high level one (normally, non-maskable interrupt, NMI). At this NMI routine, the software shall do the self-check whether the system is normal or not. If it is normal, save the CPU register information to be used at the Return from Interrupt and also save the important parameters of the system. Then, waiting for the Reset.

Then, the Reset routine shall be started. At the first, the software shall check the flags and saved data whether the saving operation was done or not. If the flag and data are okay, execute optional application routine and then, simulate the instruction, Return from interrupt. By this, the software flow can be kept.

If there shall be error find at the NMI nor Reset routine, depending on the failed condition and system needs, the programmer can set how to recover the system, a fail-safe, total initialize, force to continue and so on. Since this is software, there are large flexibility and thus, this is called as “High Resilience” system.

Once this FUJIMI system can be set correctly, the system can operate forever as far as the power and clock can be given.

In this mean, this is a type of the fault-tolerant system.

Try the FUJIMI system, and make non-stop micro-computer/controller system.

LETech Co., Ltd.
Arca East Bldg. 16F, 3-2-1. Kinshi, Sumida-ku, Tokyo, 130-0013 Japan
Tel: +81-(0)3-5819-5300 Fax: +81-(0)3-5819-5277
URL: http://www.letech.jpn.com/mcu/index_mcu.html
Email: support@letech.jpn.com