



WIKIPEDIA
The Free Encyclopedia

- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia

- Interaction
 - Help
 - About Wikipedia
 - Community portal
 - Recent changes
 - Contact Wikipedia

- Toolbox
- Print/export

- Languages
 - Català
 - Deutsch
 - Español
 - فارسی
 - 한국어
 - Italiano
 - 日本語
 - Polski
 - Português
 - Русский

Article [Talk](#)

Read [Edit](#)

Search

Redundancy (engineering)

From Wikipedia, the free encyclopedia

In **engineering**, **redundancy** is the duplication of critical **components** or functions of a system with the intention of increasing reliability of the **system**, usually in the case of a backup or **fail-safe**.

In many **safety-critical systems**, such as **fly-by-wire** and **hydraulic** systems in **aircraft**, some parts of the control system may be triplicated,^[1] which is formally termed **triple modular redundancy** (TMR). An error in one component may then be out-voted by the other two. In a triply redundant system, the system has three sub components, all three of which must fail before the system fails. Since each one rarely fails, and the sub components are expected to fail independently, the probability of all three failing is calculated to be extremely small; often outweighed by other risk factors, e.g., human error. Redundancy may also be known by the terms "**majority voting systems**"^[2] or "**voting logic**".^[3]

Contents [hide]
1 Forms of redundancy
2 Function of redundancy
3 Voting Logic
4 Calculating the probability of system failure
5 See also
6 References
7 External links



Redundant power supply

Forms of redundancy

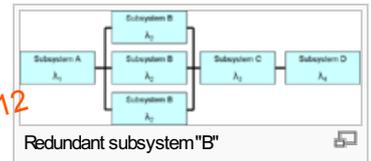
[\[edit\]](#)

There are four major forms of redundancy, these are:

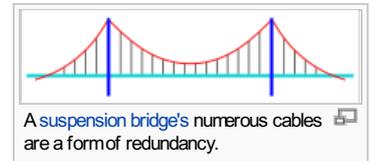
- Hardware redundancy, such as **DMR** and **TMR**
- Information redundancy, such as **Error detection and correction** methods
- Time redundancy, including transient fault detection methods such as **Alternate Logic**
- Software redundancy such as **N-version programming**

A modified form of software redundancy, applied to hardware may be:

- Distinct functional redundancy, such as both mechanical and hydraulic braking in a car. Applied in the case of software, code written independently and distinctly different but producing the same results for the same inputs.



Redundant subsystem "B"



A suspension bridge's numerous cables are a form of redundancy.

Function of redundancy

[\[edit\]](#)

The two functions of redundancy are passive redundancy and active redundancy. Both functions prevent performance decline from exceeding specification limits without human intervention using extra capacity.

Passive redundancy uses excess capacity to reduce the impact of component failures. One common form of passive redundancy is the extra strength of cabling and struts used in bridges. This extra strength allows some structural components to fail without bridge collapse. The extra strength used in the design is called the margin of safety.

Eyes and ears provide working examples of passive redundancy. Vision loss in one eye does not cause blindness but depth perception is impaired. Hearing loss in one ear does not cause deafness but directionality is impaired. Performance decline is commonly associated with passive redundancy when a limited number of failures occur.

Active redundancy eliminates performance decline by monitoring performance of individual device, and this monitoring is used in voting logic. The voting logic is linked to switching that automatically reconfigures components. Error detection and correction and the Global Positioning System (GPS) are two examples of active redundancy.

Electrical power distribution provides an example of active redundancy. Several power lines connect each generation facility with customers. Each power line include monitors that detect overload. Each power line also includes circuit breakers. The combination of power lines provides excess capacity. Circuit breakers disconnect a power line when monitors detect an overload. Power is redistributed across the remaining lines.

Voting Logic

[\[edit\]](#)

Voting logic uses performance monitoring to determine how to reconfigure individual components so that operation continues without violating specification limitations of the overall system. Voting logic often involve computers, but systems composed of items other than computers may be reconfigured using voting logic. Circuit breakers are an example of a form of non-computer voting logic.

Electrical power systems use **power scheduling** to reconfigure active redundancy. Computing systems adjust the production output of each generating facility when other generating facilities are suddenly lost. This prevents blackout conditions during major events like earthquake.

The simplest voting logic in computing systems involves two components: primary and alternate. They both run similar software, but the output from the alternate remains inactive during normal operation. The primary monitors itself and periodically sends an activity message to the alternate as long as everything is OK. All outputs from the primary stop, including the activity message, when the primary detects a fault. The alternate activates its output and takes over from the primary after a brief delay when the activity message ceases. Errors in

Gonzalez v. Arizona, No. 08-17415 archived on April 23, 2012

voting logic can cause both to have all outputs active at the same time, can cause both to have all outputs inactive at the same time, or outputs can flutter on and off.

A more reliable form of voting logic involves an odd number of 3 devices or more. All perform identical functions and the outputs are compared by the voting logic. The voting logic establishes a majority when there is a disagreement, and the majority will act to deactivate the output from other device(s) that disagree. A single fault will not interrupt normal operation. This technique is used with avionics systems, such as those responsible for operation of the [space shuttle](#).....

Calculating the probability of system failure

[edit]

Each duplicate component added to the system decreases the probability of system failure according to the formula:

$$p = \prod_{i=1}^n p_i$$

where:

- *n* - number of components
- *p_i* - probability of component *i* failing
- *p* - the probability of all components failing (system failure)

This formula assumes independence of failure events. That means that the probability of a component B failing given that a component A has already failed is the same as that of B failing when A has not failed. There are situations where this is unreasonable, such as using two power supplies connected to the same socket, whereby if one socket failed, the other would too.

It also assumes that at only one component is needed to keep the system running. If *m* components are needed for the system to survive, out of *n*, the probability of failure is^[*citation needed*]

$$P = 1 - ((1 - p)^{(n-m)} C_n^m)$$

Assuming all components have equal probability, *p*, of failure

This model is probably unrealistic in that it assumes that components are not replaced in time when they fail.

See also

[edit]

- Degeneracy
- Common mode failure
- Data redundancy
- Double switching
- Fault-tolerant design
- Radiation hardening
- Factor of safety
- Reliability engineering
- Reliability theory of aging and longevity
- Safety engineering
- Self-healing ring
- MTBF

References

[edit]

- [↑] Redundancy Management Technique for Space Shuttle Computers (PDF), IBM Research
- [↑] Majority voting systems
- [↑] Designing Integrated Circuits to Withstand Space Radiation
- [↑] Using powerline as a redundant communication channel

External links

[edit]

- Secure Propulsion using Advanced Redundant Control

Rate this page View page ratings

What's this?

 Trustworthy	 Objective	 Complete	 Well-written
★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★	★ ★ ★ ★ ★

I am highly knowledgeable about this topic (optional)

Categories: [Engineering concepts](#) | [Reliability engineering](#) | [Safety](#) | [Fault tolerance](#) | [Fault-tolerant computer systems](#)

This page was last modified on 18 April 2012 at 07:46.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. See [Terms of use](#) for details. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.

[Contact us](#)

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Mobile view](#)

