解　説　論　文

コンピュータ安全と機能安全

Computer Safety and Functional Safety

向殿政男 Masao MUKAIDONO

アブストラクト　コンピュータと安全との関係は，大きく分けて二つの側面がある．一つは，コンピュータを内部や外部の危険源から守り，コンピュータを正常に稼動させ続けることを目的とする“コンピュータのための安全”であり，二つ目は，他のシステムの安全を維持する機能をコンピュータで実現させようとする“コンピュータによる安全”である．筆者は前者をコンピュータ安全と呼んでいるが，後者の典型例として機能安全と呼ばれる新しい安全の分野がある．本稿では，コンピュータ安全と機能安全の一般論とともに最近の動向について解説する．

キーワード　コンピュータ安全，機能安全，IEC61508，フォールトトレランス，フェイルセイフ

1．まえがき

現代社会では，コンピュータの高度な機能を利用することで，多くの機械，製品，施設・設備等のシステムが設計され，運用・維持されている．コンピュータは社会の発展・進歩・維持に大きく貢献しており，今後とも変わることはないだろう．特に，列車，飛行機，原子炉，航空宇宙，医療機器等，人命に係るようなセーフティクリティカルシステムにもコンピュータは積極的に利用されている．しかし，時として，コンピュータの障害が原因で，不幸で悲惨な事故が発生しているのが現実である．コンピュータと安全とは，近年，切っても切れない関係になってきており，その関係はますます深く，かつ複雑になってきている．すなわち，社会の多くの物事が，大きくは，社会インフラ維持・管理から，小さくは，家電製品の制御まで，コンピュータの機能に依存してのみ存在し得る状態になってきていて，コンピュータの障害が社会や身の回りの安全を直撃する時代になってきている．

長い間，安全に直接かかわる部分には，コンピュータは使用すべきでないと主張する人々がいて，一般にもそのように考えられてきた．なぜならば，コンピュータを構成している半導体などの電子部品は，どのように壊れるか保証はないし，ソフトウェアにバクがないことを保証することはほとんど不可能だからである．安心してコンピュータに安全を任せるわけにはいかないという主張である．一方，コンピュータの高度な機能を安全の実現のために使わない手はないと考えるのも，また自然な発想である．現実には，上記のようなセーフティクリティカルシステムと呼ばれる人命に係るような高度なシステムでも，コンピュータが主要な役割を果たしており，コンピュータなしにはこれらのシステムが実現できないのも事実である．

コンピュータと安全の関係をどのように考えたらよいのであろうか．ここでは，この関係を二つの側面に分けて考えることにする．一つは，素朴に，コンピュータに障害が発生しないように正しく機能し続けられること，すなわち，部品の故障，ソフトウェアのバグ，人間のミス，自然災害，外部からの悪意のある侵入等々，内部・外部からの脅威に対して，いかにコンピュータを守るか，という側面である．これを“コンピュータの安全を守る”という意味からここでは，“コンピュータ安全”と呼ぶことにする．もう一つの側面は，他のシステムの安全を監視し，システムを安全に制御し，必要ならば安全のためにシステムを止めるという，いわゆる安全装置の役割をコンピュータに果たさせて，いかにコンピュータで安全を守るか，という側面である．これを“コンピュータを用いて安全の機能を実現する”という意味から，最近注目されている用語を用いて，“機能安全”と呼ぶことにする．

本稿では，コンピュータ安全と機能安全の考え方と技法を紹介し，その違いとお互いの関係を考察することにする．そのためには，まず，一般論として，安全性と信頼性の関係を明確にしておく必要がある．

2．信頼性と安全性

2.1　信頼性と安全性は異なった概念

“信頼性を上げれば安全性は上がる”というのは，多くの場合はそのとおりである．したがって，安全性の問題は信頼性の問題に帰着できると考えている人がいるが，実はこの考えは正しくない．信頼性と安全性はお互いに深い関係にあるが，本質的には異なった概念である．例えば，危ないときには止

向殿政男　正員：フェロー　明治大学理工学部情報科学科，理工学研究科新領域創造専攻安全学系
Masao MUKAIDONO, Fellow (Dept. of Computer Science, School of Science and Technology, Meiji University, Kawasaki-shi, 214-8571 Japan).
電子情報通信学会 基礎・境界ソサイエティ
Fundamentals Review Vol.4 No.2 pp.129-135 2010年10月

める，危険が予想されるときには実行しないというように，信頼性を下げてでも安全性を高めることがあり得ることを考えると，両者が異なった概念であることが分かる．安全性が確保された上で要求された本来の機能を果たすことに関する信頼性を問題にすることもできるが，このことが安全性と信頼性とが異なった概念であることを如実に表している．このことに関してもう少し詳しく見てみよう．

信頼性とは，与えられた条件下で，与えられた機能をいかに持続させるかを問題にしているのであって，機能を失った後の状態は問題にしていない．一方，安全性の方は，故障等で機能を失った後の状態も問題にしている．正しく機能していれば安全性が確保されている可能性が高いので，一般的に信頼性が高ければ安全性は高くなるというのは，冒頭のとおり，多くの場合正しい（想定外という与えられた条件外の状況が発生すれば，正しく機能していても安全が脅かされる可能性はある）．しかし，問題は故障等の不具合の発生時，正しい機能を失ったときである．故障したときには，危険になることも安全になることもある．安全性は，故障等の不具合が発生したときにも安全であることを目指している．安全が確認されないならば，初めから機能を止めて安全側に固定しておく，または不具合が発生したら安全側に固定して機能を止めてしまうというのは，安全性の重要な機能の一部である．このように信頼性と安全性とは，本来の機能を目指すのか，それとも安全を目指すのかの違いがある．前述のように，本来の機能と安全を確保するという機能の両方を含んだものを新しい機能と考えてその信頼性を問うという発想もあるが，基本的に本来の機能と安全の機能とは別の概念で，分けて考えるべきものである．別概念であるからこそ，安全の信頼度（安全が確保されている確率），安全機能の信頼性という発想が出てくる．

2.2　信頼性と安全性の手法

信頼性と安全性とが異なった概念であるから，実現する手法も両者では一般的に異なっている．信頼性を上げるには，故障しないような高信頼の部品を作る（フォールトアボイダンス）とか，たとえ故障しても他の部品で補って機能を正しく保つ（多重系，冗長系，フォールトトレランス）とか，チェックや監視等（時間冗長とか情報冗長）の技術に重点が置かれる．一方，安全性を上げるためには，初めから危険なところ（危険源）がないように設計するとともに，たとえ部品が故障しようとも，常に安全側になるように構造的に作るというフェイルセイフの考え方が本質的となる．このためには，安全側がどのような状態であるかが分かっていなければならない．列車は止まっていれば安全であり，飛行機は飛ばなければ安全である．一方，信号でいえば，青信号は危険側であり，赤信号は安全側である(注1)．

故障の発生は認めるが，それが常に安全側故障になるように構造的に作るのが，安全性設計の基本である．例えば，止まっていることが安全側ならば，故障したら止めてしまうというのが，安全の技術的な対応であり，安全が確認されるまで動かさないようにするというのが安全の組織的対応である．ところで，飛んでしまった飛行機には，安全側はあるのであろうか．もちろん，飛行中にエンジンが止まれば，落ちるだけであり（危険側故障），この場合には，いかに飛行するという機能を継続するかが課題となる．すなわち，安全側が存在しない状態では，正しい機能をいかに継続させるかという信頼性の話になる．この場合，主として信頼性で安全性を確保していることになる．このように，信頼性と安全性とは，お互いに深い関係にあり，両者は不可分の関係にある．

それでは，安全性，信頼性，どちらを優先すべきであろうか．人命を預かるような分野では，安全側が存在する場合にはまず安全を確保する構造を構築した上で，信頼性を高めるという順番が大事であり，この順番を間違えてはいけない．ただし，誤報の多い火災報知機は切られる運命にあるように，信頼性に裏打ちされていない安全性は，実効性に問題が生ずることを忘れてはならない．

3．コンピュータの安全を守る・・・コンピュータ安全(1)

コンピュータシステム自身が，内部，外部からの障害，脅威等に耐えて，本来の機能を正しく果たすことを，ここでは，コンピュータの安全を守るという意味でコンピュータ安全と呼ぶことにする．コンピュータ安全を考える場合，(0) 何のために，(1) 何を，(2) 何から，(3) 何を用いて，守るのかという安全の構造を明確にする必要がある(2)．(0) 何のために守るかに関しては，コンピュータが使用される応用分野によって異なるであろうが，究極は社会活動の健全な維持と人の安全・安心のためであり，ここで余り言及する必要はないだろう．(1) 何を守るかに関しては，この場合はコンピュータとその機能を守る，といえるが，具体的には大きくは三つに分けられる．一つ目はコンピュータのハードウェアを守る，二つ目はコンピュータのソフトウェアを守る，三つ目はコンピュータで取り扱っている情報の内容を守ることである．一般的には，前者の二つがコンピュータシステムの信頼性であり，後者が情報セキュリティの分野である．(2) 何から守るかについては，大きく分けると，自然災害，故障，人間のミス，人間の故意・悪意等であろう．(3) 何を用いて守るかについては，大きく分けると，①技術で守る，②人間の注意で守る，③組織・仕組みで守る，の三つに分類できる．従来の

(注1)　青信号は安全を，赤信号は危険を意味すると教わっている身からすると，一見，矛盾しているように思えるかもしれない．なぜ，青信号が危険側かというと，青信号が出ていると列車は出発するし，車は止まらないで走り続ける．危険性の高い本来の任務を実行するから，青信号は危険側なのである．一方，赤信号ならば止まるので事故は起こり得ないから，安全側なのである．このことは，信号が間違えた場合を考えれば明らかになる．青信号が間違えて赤信号になった場合には，不便ではあるが事故にはつながらない．これは安全側の故障であり，赤信号は安全側になる．逆に，赤信号であるべきとに青信号に故障すると，事は重大である．大事故につながる可能性がある．この故障は危険側故障であり，青信号が危険側であることが分かる．

コンピュータの高信頼化研究は，コンピュータのハードウェアを故障から技術的に守ることに主眼が置かれていたが，コンピュータ安全の分野は非常に広く，現在では，ソフトウェアの信頼性と情報セキュリティが極めて重要な課題となってきている．

与えられた機能が正しく果たせなくなる根本的な原因をここでは，危険源（ハザード）と呼ぶことにすると，コンピュータ安全に関しては，ハードウェア，ソフトウェア，及び情報の分野の主な危険源をまとめると表1のようになる．

ここでは，技術者にとって最も関心の高い技術的対応，すなわち，ハードウェアに関しては故障から技術で守る話，ソフトウェアに関してはバグから技術で守る話，情報に関しては故意・悪意から技術で守る話について簡単に触れてみることにする[(1)]．

表1　コンピュータ安全における守るべき対象と危険源

何を守るか	何から守るか（主な危険源）
ハードウェア	自然災害（地震，水害，等） 構成部品の故障 人間のミス（オペレータミス，設計ミス等） 人間の故意・悪意（破壊，テロ等） 電源の停止
ソフトウェア	バグ 設計ミス 要求仕様設定ミス
情報	人間の故意・悪意（盗聴，漏えい，改ざん等） 人間の運用ミス（漏えい，紛失，消滅等） 構成部品の故障

3.1　ハードウェアの信頼性・・・故障からハードウェアを守る

部品の故障からコンピュータを守るための最も基本的なアプローチは，故障しにくいような高信頼度の部品を使用するか，必要ならば製造することである．これは初めからフォールト（障害，欠陥）の存在を回避することを目指すアプローチで，フォールトアボイダンス（fault avoidance）と呼ばれる．これは，物理的な現象解析から品質管理，生産管理等の技術が関係してくる．次に，二重系等の多重系を用いて，信頼性を上げるアプローチがある．一つや時には二つ以上の部品が故障しても，他の部品がこれに代わって機能を果たすという冗長系の考え方である．このアプローチはフォールトトレランス（fault tolerance）と呼ばれる．冗長系による高信頼性の実現方法は，部品レベルだけでなく，システムレベルも含めたいろいろなレベルで用いらる．例えば，3台同時に稼働させて多数決で出力を決める三重多数決システム，全く同じシステムを同時に2台稼働させて出力を照合させるなり，動いている方のシステムを利用するデュアルシステム，1台を稼働させていて，もう1台を予備系として待機，または重要度の低い作業をしながら待機させるデュープレックスシステムなど，冗長系にも多くの形態があり得る．

冗長系の場合，いつまでも放置しておくと，同時に故障して外に影響が出てきてしまうので，故障の存在や故障個所を見いだす故障検出や故障診断等のテストの技術が重要になる．更に，テストをし易いように設計するテスト容易化設計，内部で自分自身を常にチェックするセルフチェッキングという技術もある．これ以外にも，機能分散や負荷分散や危険分散等の分散システムの技術，システム再構成の技術等々，従来のコンピュータシステムの信頼性技術の多くが，この分野の技術として開発されている．

3.2　ソフトウェアの信頼性・・・バグをいかに減らすか

何百万，何千万のプログラミングステップからなるソフトウェアを，バグが存在しないように開発することは，ほとんど，不可能なように思える．また，出来上がったソフトウェアシステムのバグを検出することも，その検査すべき場合の数の可能性を考えると，ほとんど不可能であるといわざるを得ない．したがって，現時点では，大規模なソフトウェアにバグが存在しないと断言できる人はだれもいない状態にある．強いていえば，今のところ正常でありバグは見いだされていません，としかいいようがないのが現状である．それでも，常にバグゼロを目指して信頼性の高いソフトウェアの開発に取り組まざるを得ない．その方法には，上記のように，主に，バグを発生させないようなソフトウェアの開発方法，出来上がったソフトの正しさをチェックする方法の二つが主流である．それ以外にも枯れた実績のあるソフトウェアを使う，ハードウェアの冗長系と同じ発想で同一要求仕様に対して異なったグループと異なった言語で開発をしたソフトウェアを比較して，一致したときだけ正しいとするNバージョン方式，等々，幾つかの方法が用いられている．表2に代表的なソフトウェア信頼性の手法を挙げておく．

表2　ソフトウェアの信頼性手法

Nバージョンプログラミング
ソフトウェアダイバーシチ
ソフトウェアバグ曲線
人工知能による欠陥の修正
形式的方法
時相論理
形式的証明
解析可能なプログラム（ex.:構造化の概念）
構造化プログラミング手法
言語サブセットの使用
型が強化されたプログラミング言語
適切なプログラミング言語
使用によって証明された翻訳プログラム
認証されたツール
信頼/立証されたモジュール
ウォークスルー/設計レビュー

3.3 コンピュータ内の情報の信頼性・・・脅威から情報を守る

部品等の故障や雑音などで，コンピュータ内の情報が変化して正しくなくなってしまう内的な問題と，外部から故意，悪意等により，コンピュータ内の情報の盗聴，漏えい，改ざんの問題の両面がある（情報の漏えい等は，作業者のミスでも起こり得る）．

コンピュータ内で情報が正しくなくなってしまうことを守る技術としては，パリティビット符号や誤り検出・訂正符号のような符号理論に基づく情報の冗長性技術を用いて検出，訂正を行うのが一般的である．過渡的な誤りに関しては，もう一度同じことを行うリトライによる時間冗長技術が，また，データベースなどでは，ファイルを二重化するなどの空間冗長技術が主に用いられている．

一方，人間の故意・悪意に基づく情報の盗聴，漏えい，改ざん等の問題は，情報セキュリティ問題として，これまで多くの技術が開発されて，組織や規定が策定されて，人間の倫理の問題として，広く取り組まれているが，ここではこれ以上，深入りしないことにする．

4. コンピュータで安全を守る・・・・機能安全

4.1 安全を守るコンピュータの役割[(3)]

システムに組み込まれたコンピュータの主な役割は，そのシステムが実現すべき高度な機能を果たすことである．しかし，人命が係るようなセーフティクリティカルなシステムでは，そのシステムの安全を守るという機能も果たさなくてはならない．両者の機能を同時に果たすのは，一般的に困難であるとともに，要求される信頼度のレベルが異なるはずである．第一，前述したように目的が異なる．そこで，本来のシステムの機能を担当するコンピュータと安全を担当するコンピュータとを分けるという発想が必要となる．すなわち，一般に，安全をつかさどる部門を安全関連系，そうでない部分を非安全関連系といい，両者を分ける必要がある．例えば，産業用ロボットでいえば，高速に，高度に，効率的に稼働させるという本来のロボットの機能をつかさどるのが非安全関連系で，人間が近づいたら止める（停止の安全），人間がいないときしか動かない（隔離の安全）という安全の機能を実現する安全装置の部分が，安全関連系に相当する．両者にコンピュータが応用される可能性がある．我々の目的は，後者の安全関連系におけるコンピュータの応用にある．

システムに組み込まれたコンピュータでシステムの安全を実現するためには，二つの階層に分けて考える必要がありそうである．一つの層は，そのコンピュータが組み込まれているシステム自身が目的としている安全である．現実にそのシステムが何を守るかはシステムの本来の目的に依存していて，安全関連系のコンピュータは，システムの本来の安全を守る一部の役割を担っていて，安全関連系のコンピュータには，安全を守る機能に必要な要求仕様が与えられるだけで，システム本体が何を守るかには直接関係しないと考えるのが自然だろう．したがって，二つ目の階層は，安全関連系のコンピュータ自身が果たすべき安全の役割であり，それはシステムが本来の目的を安全に実行するために，システム設計の段階でコンピュータに要求された安全の機能を正しく果たすことである．その中には，安全関連系が対象とする非安全関連系の中でコンピュータ等で制御されているサブシステム（被制御系:EUC-Equipments Under Control-）を監視したり，安全に制御したり，また，システムに故障等が発生した場合や危険が予想される場合には，安全側に固定する指令を出す機能が含まれる．すなわち，安全関連系のコンピュータに要求される機能は，(1) 何を守るのかに関しては，

(a) 与えられた安全要求機能がコンピュータに正しく組み込まれていること
(b) 与えられた安全機能がコンピュータにより正しく実行されること

の2点に集約できるだろう（要求仕様に漏れがある場合には，・・実際にはここに多くの問題があるのだが，・・安全関連系のコンピュータの責任範囲ではない）．上の二つの機能は，安全という用語を除けば，通常の機能を果たすコンピュータと全く同じ役割である．当然ではあるが，安全関連系自身にも故障等が発生することを考慮しなければならない．すなわち，(2) 何から守るかについては，**3.** のコンピュータの信頼性の項で述べたのと同じで，安全関連系コンピュータでも，設計・仕様ミス，ハードの故障，ソフトのバグ，人間の故意・悪意等から安全を守らなければならない．しかし，ハードウェアに故障は付き物であり，ソフトウェアの完全性は期し難いので，うまく守れない場合もあり得る．そのとき，システムの状態を，または安全関連系の出力を安全側に固定できるのであろうか．安全関連系のコンピュータに要求される機能として，上記の (a)，(b) に加えて，

(c) 故障等が発生したときに安全側に固定すること

が問われることになる．しかし，コンピュータは指令を出すことはできても実際にシステムを安全側に固定することもできないし，自分自身の故障に対して出力を安全側の固定できる保証はない．コンピュータで安全を守ることに関しては，この問題が常に存在することを忘れてはならない．

さて，(3) 何を用いて守るかについては，ソフトウェアに基づいて，それぞれの危険源に対応することになるが，そのためには，故障とソフトウェアのバグの本質的な性質の違いを明確にしておく必要がある．

4.2 ソフトウェアのバグは故障か

これまで，故障という言葉を何げなく一般的に使ってきたが，故障という日本語は，故障するという動詞にも，故障しているという状態にも使用されるので，概念的にはかなり広くて，時にはあいまいである．ここでは，まず，ハードウェアを例にして，故障する（動詞），フォールトがある（状態），誤る（機能）の三つの関係を明らかにしておこう．ハードウェアでは，劣化，摩耗などの物理現象等が原因で，当初は正しい機能を果たしたものが，途中で正しい機能を失う（故障する:Failure）ことによって，正しい機能を遂行できない状態（フォールト，障害，不具合:Fault）になり，その機能を利用しようとすると正しい機能と不一致（誤り，エラー :Error）が生ずる．原因，結果の連鎖は，一般には，上記のような関係に整理してよいと考えられる．このとき，ソフトウェアには，ハードウェアの部品が劣化して故障するような意味での故障は確かに存在しない．途中から故障するのではなく，バグは最初からフォールトとして潜在していて，そこを利用するときに初めて誤りが生ずる．上の連鎖であえてフォールトの原因を探せば，そのソフトウェアを製作した人の頭脳の一時的な故障かもしれない（ミスといった方がよいだろう）．この点からは，各種の設計ミスも文書の間違いも，ソフトウェアのバグと同じ部類である．

ハードウェアのような故障をランダム故障と呼び，確率や統計学で理論的に取り扱われている．ソフトウェアのバグや設計ミスのような故障は，システマティック故障と呼び（決定論的故障とも，系統的故障ともいう），両者は必然的に分けられなければならない．これらの故障からシステムを守る手法は，両者で異なる．ランダム故障に対しては，ハードウェアの信頼性で述べた確率論に基づく技術的な手法が主として用いられる．システマティック故障には，確率論的な手法は余り役に立たず，ソフトウェアが出来上がった時点や開発の途中経過で，バグがないかどうかをチェックするか，または，いかにバグの少ないソフトウェアを最初から構築するかという製作のプロセスを重視した技法が採用される．

4.3 機能安全とは

安全を実現しようとする機能を安全機能（Safety function）という．安全装置のような安全関連系が実現しようとしている機能が安全機能の例である．例えば，非制御装置の安全状態を達成したり，維持したりする機能であり，時には，危険状態になったときに脱出したり，危険状態に入ることを阻止したりする機能をいう．これに対して，機能安全（Functional safety）とは，安全関連系の正しい機能で確保される被制御装置の安全性であり，安全性を実現する多くの手法の一つである．いい換えれば，安全関連系が正しく働いている時に果たすことで実現される安全性である．逆にいえば，安全関連系が故障等により正しい機能を果たせなくなったときに失われた安全機能が，機能安全により実現されていた安全性である．正しい機能が果たせなくなったときには，安全機能は失われることを前提としている．その確率をいかに少なくするかが機能安全の課題となる．

機能安全というのは，本質的安全や構造安全という考え方に対比して提唱され出した概念である．本質的安全とは，設計段階の初めから危険源がないように設計したり，危害が起きたときのことを考えて，エネルギーを小さくするなどして危害の大きさを小さくしたり，人間が近づかなくて済むように危険状態が発生する確率を小さくしたりすることで，設計の段階から安全を実現する考え方である．構造安全は，フェイルセイフのように，安全装置が故障したときのことも考えて，故障したときには，常に安全側になるように構造的に構築する考え方である．これに対して，機能安全は，簡単にいえば，安全装置の機能を用いて安全を維持し，安全装置が正しく機能する確率を高くすることでシステムが安全である確率を高くしようとする考え方である．また，機能安全に対して従来の本質的安全，構造的安全は，それぞれ，確率論的安全に対して確定論的安全と呼ばれており，前者は，確率的に安全を確保し，後者は確定的に安全を確保しようとする考え方である．機能安全が出てきた背景には，コンピュータを用いて安全関連系を構成して，安全機能を実現しようという動機からである．コンピュータ等に安心して安全を任せられない，という冒頭の意見は，伝統的に安全を構造的に構築してきた後者からの批判であり，安全の確保にコンピュータの高機能を利用しない手はないという主張が前者の立場である．現実には，両者は深くかかわり合っていて，今後の複雑で高機能なシステムの安全確保には，両者の考え方を融合する必要がある．絶対安全が存在しない以上，構造安全や本質的安全に基づいてどのくらいの確率で安全性が確保されるかを評価せざるを得ないし，第一，どんな場合にも，本質的安全や構造的安全が実現できる保証はない．複雑なシステムになればなるほど困難になってくる．一方，機能安全では，安全関連系が正しく機能しなくなったときには，被制御系が危険側故障になる場合と安全側故障になる場合とがあり，いかに危険側故障を小さくするかが重要な問題となる．ここには，構造安全の考え方を取り入れないわけにはいかない．今後のセーフティクリティカルシステムの構築では，両者の考え方の融合が必須になってくることは間違いない．

ここで紹介した機能安全では，安全機能を実現する安全関連系に，コンピュータや電気・電子機器等を用いて実現することを頭に置いている．このための国際安全規格がIEC61508（電気・電子・プログラマブル電子安全関連系の機能安全）である．

4.4 機能安全規格IEC 61508について

我々は，安全を確保する分野に安易にコンピュータを導入して，これまで大きな事故を経験している．いかにコンピュータを安全分野に導入すべきかという議論を制御システムやプラント制御の技術者等が集まり，IEC（国際電気標準会議）で

議論して出てきたのが機能安全の考え方であり，電気・電子・プログラマブル電子安全関連系の機能安全[4]という国際安全規格である（我が国では，IEC化途中の段階のものを翻訳して，JIS C 0508として発行している[5]）．全体で7部構成の膨大な規格であり（表3），かつ難解であるために，ここで詳しく解説している余裕はないが，基本的な考え方だけでも簡単に紹介してみよう（詳細は，幾つかの解説書が出ているので参照されたい[6]，[7]）．

まず，

(1) 安全は，システムのライフサイクルを通じて，設計から廃棄まですべてを包括的に考えて実施されなければならないこと，
(2) 安全関連部と非安全関連部に分割し，システムの安全は各種の外部のリスク低減方策と安全関連系とによって実現されること，
(3) 安全関連系も電気・電子・プログラマブル電子安全関連系とそれ以外の安全関連部に分かれるが，この規格では，電気・電子・プログラマブル電子安全関連系のみを取り扱っていること

等が大枠の前提である．

ここでプログラマブル電子安全関連系とは，コンピュータによる安全機能の実現を意味している．ハードウェアとしての電気や電子装置は，構造的にどのように壊れるか明確でないので，構造安全のように故障した後のことは深く考慮しないで，いかに機能を正しく発揮し続けるかという確率の観点に立った信頼性の立場から，冗長性に基づくフォールトトレランス技術の考え方を採用している．これがランダム故障に対する対策である．

一方，コンピュータのソフトウェアについては完全性は保証できないので，いかにバグの少ないソフトウェアを開発するかという観点から推薦すべき開発手法を数多く挙げるとともに，開発のプロセスをV字モデルに従って細かく指定する手法を採用している．また，出来上がったソフトウェアのテストに関しても，すべての可能性をチェックすることは不可能なので，どのくらいの範囲をテストしたか，またはできるかというカバレージの概念を採用している．更に，開発プロセスで信頼性を確保するために，製品そのものだけでなく，開発やテストのプロセスを評価する組織や人間のコンピテンシー（能力）と独立性を重視している．これがシステマティック故障に対する対策である．

機能安全で最も特徴的なのは，安全度水準（SIL: Safety Integrity Level）という考え方を採用していることである．電気・電子・プログラマブル電子安全関連系が正しく機能しないとは，ハードウェアとしてランダム故障が冗長性をかいくぐって出てきたか，ソフトウェアや仕様書に関するシステマティック故障がチェックできなかったことにより表に出てきたものである．しかし，これらの故障が必ずしも被制御系を危険にさらすわけではない．すなわち，故障にも危険側故障と安全側故障とがある．問題にすべきは，危険側故障であり，これをいかに低く抑えるかである．危険側故障率をどのくらい小さくするかで，SIL（安全度水準）が提案され，これは4段階に分かれている（表4）．まず，被制御系からの安全機能の要求が，非常停止のように必要なときにだけ機能すべき低頻度作動要求モードと，プレスの安全装置のように頻繁に作動要求が生じる高頻度作動要求や監視装置のように連続で運用するモードの二つに分け，四つのレベルのSILに対しての危険側故障率の値を定めている．SILは，故障率を規定しているが，安全は一般にリスクで評価されるはずである．すなわち，危害のひどさも考慮に入れなければならない．危害のひどさは，電気・電子・プログラマブル電子安全関連系が組み込まれているシステムがどのような安全を実現しているかにより異なる．そのシステムの故障により，どのくらいの被害が出るかにより，安全関連系のSILが決められる．すなわち，危害が大きくなると予想されるシステムの安全関連系は高いSILを割り当て，危害が小さいと予想されるシステムの安全関連系には低いSILを割り当てる，という使い方をする．このとき，安全関連系のハードウェアはこのSILの値から冗長性の構造が決まる．ソフトウェアについては，信頼性を数値で出すことができないので，そのSILのレベルに合ったソフトウェアの開発手法やプログラミング技法が指定され，必要なカバレージが指定され，かつ評価のために人間や組織の独立性が指定される，といった方法でこの規格は利用される．

機能安全規格IEC 61508は，機械安全の規格の全体からなる規格の階層化の中ではグループ規格であるB規格に位置付けられており，その下に多くの個別規格が制定される形になっている．すなわち，この規格はアンブレラ規格であり，現在，続々とこの規格の考え方に従い，個別の機械や装置の機能安全規格が検討されつつある（表5）．

表3　IEC61508 電気・電子・プログラマブル電子安全関連系の機能安全の構成

第1部 一般要求事項
第2部 電気・電子・プログラマブル電子安全関連系に対する要求事項
第3部 ソフトウェア要求事項
第4部 用語の定義及び略語
第5部 安全度水準決定方法の事例
第6部 第2部及び第3部の適用指針
第7部 技術及び手法の概観

表4　SIL：安全度水準

SIL	低頻度作動要求モード運用（注1）	高頻度作動要求又は連続モード運用（注2）
4	10^{-5}以上10^{-4}未満	10^{-9}以上10^{-8}未満
3	10^{-4}以上10^{-3}未満	10^{-8}以上10^{-7}未満
2	10^{-3}以上10^{-2}未満	10^{-7}以上10^{-6}未満
1	10^{-2}以上10^{-1}未満	10^{-6}以上10^{-5}未満

注1　作動要求当たりの設計上の機能失敗平均確率
注2　単位時間当たりの危険側故障確率[1/時間]

表5　IEC61508機能安全規格は，アンブレラ規格

・ISO10216	ロボット関連
・ISO13849-1	制御システム関連
・ISO26262	自動車関連
・IEC60601	医療機器関連
・IEC61311-6	PLC関連
・IEC61511	プロセス計装関連
・IEC61513	原子力発電関連
・IEC62304	医療機器関連
・IEC62424	鉄道信号関連

5．あとがき

コンピュータ安全と機能安全というように，ここではコンピュータと安全との関係を二つの側面に分けて考察してきたが，御賢察のように両者はお互いに深く関係し合っている．前者は，コンピュータにより実現されている機能の信頼性を対象にしており，後者はコンピュータによる安全性という機能の実現を対象にしている．したがって，機能安全では，安全性を確保する機能の信頼性の問題を重視せざるを得ず，機能安全はコンピュータ安全をその一部として当然含むことになる．

前述のように，信頼性と安全性は基本的には異なる概念であるが，現実には，安全性を実現するためには，安全性技術と信頼性技術との両者が用いられる必要がある．しかし，ここで紹介したように，コンピュータを用いて安全性を確保する場合は，信頼性技術を中心にせざるを得ない．その理由は，どのような状態が安全かは，コンピュータが組み込まれたシステムによって異なり，コンピュータ自身では定義できないこと，もう一つの理由が，前述したように，コンピュータの故障には，電子機器，ソフトウェアの性質から，現状の技術では非対称故障が実現できずに，フェイルセイフのように構造的に，物理的に，そして確定論的な安全性を実現することができないからである．したがって，通常はコンピュータの信頼性機能を使って安全性を実現しているシステムでも，危険側故障が発生した場合の最後の最後は，物理的な装置を使って停止させるなどして安全を確保させる場合が多い．例えば，エレベータでいえば，通常はコンピュータを使って安全が確保されているが，落下等の急激なスピードが発生した場合には，くさびを使って物理的に止める装置が常備されている．普段は表に出ないが，最後は安全側に止める構造的な安全技術が存在していることが裏打ちされた上で，コンピュータによる確率論的な安全性の威力が発揮される場合が多いことを知っておくべきである．

これまで，情報関係やソフトウェア関係の技術者は，安全に関して深く考える必要がなかったかもしれないが，これからのコンピュータシステムの利用や設計に係る人間は，それは技術者や管理者，企業のトップに係らず，コンピュータと安全性の関係を深く理解しておく必要がある．出来合いのソフトウェアやコンピュータシステムを組み合わせて，安易に安全にかかわるシステムを構築するような発想を決して持ってはならない．特に，人命に係るようなシステムの組込みソフトに従事する技術者は，安全設計についてを十分に理解して，システム構築をしなければならない[(7)]．なぜならば，制御対象のハードウェアの特性と人間が望む安全の内容とレベルを知らない限り，真の安全を実現することはできないが，組込みソフト開発の技術者は，コンピュータ側からするとこれらと最も近いところに位置するからである．

文　献

(1) 向殿政男, “情報とコンピュータ”, 安全の百科事典, 田中昌三(編), pp.81-87, 丸善 東京, 2002.
(2) 向殿政男, 北野　大, 菊池雅史, 小松原明哲, 山本俊哉, 松原健司, 安全学入門-安全の確立から安心へ, 研成社, 東京, 2009.
(3) 向殿政男, “ー総論ー安全と技術と社会,” 信学誌 vol.88, no.5, pp.310-315, May 2005.
(4) IEC 61508, “Functional safety of electrical/electronic/programmable electronic safety related systems,” 2000.
(5) JIS C 0508, “電気・電子・プログラマブル電子安全関連系の機能安全,” 2000.
(6) 井上洋一, 川池　襄, 平尾裕司, 蓬原弘一, 制御システムの安全, 向殿政男(監修), 日本規格協会, 2007.
(7) (社)組込みシステム技術協会, 組込み系技術者のための安全設計入門, 電波新聞社, 2010.

（SSS研究会提案，平成22年7月21日受付）

向殿政男（正員：フェロー）

昭40明大・工・電気卒．昭45同大学院博士課程了．同年明大・工・電気・専任講師，昭53同電子通信・教授，平元同大学・理工・情報・教授，平14同理工学部長．その間，フェイルセイフ理論，ファジー理論，機械安全，多値論理の研究に従事．日本ファジィ学会会長，日本信頼性学会会長．日本知能情報ファジィ学会フェロー，国際ファジィシステム学会(IFSA)フェロー．著書「ファジィ論理」，「よくわかるリスクアセスメント」，「ニューロとファジィ」など．