

ANALISIS DAN PERANCANGAN TOLERANSI , KESALAHAN PADA *ON BOARD DATA* *HANDLING* SATELIT LAPAN A2

Abdul Karirri, Gunawan S. Prabowo

Peneliti Pusat Teknologi Elektronika Dirgantara, LAPAN

Email: karlm.lapan@gmail.com

gunawan_prab@yahoo.com

ABSTRACT

Lapan A2 satellite is the second generation of the lapan micro satellite developed by the corporation between Lapan and Technische Universitat Berlin. The subsystem of Lapan A2 will be developed, particularly OBDH (On Board Data Handling) subsystem. OBDH perform two major functions. The first function is it receive, validates, decodes, and distributes commands to other spacecraft system. The second is it gathers, processes, and formats spacecraft housekeeping and data mission for downlink or used by on board computer.

The result of fault tolerance analysis that will be implemented in the OBDH design will be based on redundancy of hardware, software, and information. Hardware redundancy will be implemented using double processor and memory storage. Whereas software redundancy will be implemented by using single version check point and restart, and program boot strap method as operating system and program spare, Hamming code and mirroring data as redundancy information will be implemented for error correction and error detection. The selections have considered the power consumption and the structure size constraints of the satellite.

Keywords : *OBDH, Fault Tolerance, Satellite, Redundancy*

ABSTRAK

Satelit Lapan A2 adalah satelit mikro LAPAN generasi ke-2 yang dibangun melalui kerjasama antara LAPAN dan *Technische Universitat Berlin*. Dalam generasi ke-2 ini akan dilakukan pengembangan di dalam subsistem, khususnya subsistem OBDH (*On Board Data Handling*). OBDH mempunyai dua fungsi utama, pertama adalah menerima, memvalidasi, mendekode, dan mendistribusikan komando ke subsistem lain dalam sistem satelit, dan yang kedua mengumpulkan memproses data *housekeeping* serta data misi untuk dikirim ke stasiun bumi atau digunakan oleh *On Board Computer*.

Hasil analisa sistem toleransi kesalahan (*fault tolerance*) yang akan diimplementasikan dalam perancangan OBDH akan dilihat dari tiga sisi yaitu redundansi perangkat keras, redundansi perangkat lunak, dan redundansi informasi. Penggunaan redundansi perangkat keras dengan menggandakan prosesor dan *memory storage*, Sedangkan redundansi perangkat lunak menggunakan metode *single version check point and restart* dan program *boot strap* sebagai cadangan sistem operasi dan program. Metode *hamming code* untuk *error detection* dan *corection* serta metode *mirroring* data akan digunakan sebagai redundansi informasi. Pemilihan redundansi tersebut telah mempertimbangkan keterbatasan dari konsumsi daya dan ukuran yang dibatasi oleh sistem satelit secara keseluruhan.

Kata kunci: *OBDH, Toleransi kesalahan, Satelit, Redundansi*

1 PENDAHULUAN

Satelit LAPAN A2 adalah satelit mikro yang dibangun melalui kerjasama

antara Lapan dan *Technische Universitat Berlin*. Sale Ut M merupakan pengembangan dari satelit LAPAN Tubsat

dengan peningkatan pengembangan di beberapa subsistem. Hal tersebut merupakan titik awal perancangan satelit di Indonesia dan akan menjadi dasar dalam perancangan satelit-satelit berikutnya. Misi utama satelit mikro LAPAN A2 adalah *surveillance* yang merupakan pengembangan dari satelit LAPAN Tubsat. Sebagaimana misi LAPAN Tubsat, beberapa kejadian seperti kebakaran hutan, banjir, gempa, gunung berapi, kecelakaan kapal laut dan pesawat terbang, dan lain-lain dapat dimonitor.

Sistem operasi akan dikembangkan sistem pengambilan gambar secara otomatis dengan target (gambar yang akan diambil) yang diseting sebelumnya. Data masukan sikap satelit akan diambil melalui *star sensor* untuk mengendalikan sikap satelit menghadap ke bumi sehingga ketika posisi berada pada posisi target maka kamera satelit sudah siap menghadap ke bumi. Posisi target akan diambil dari data GPS [*Global Positioning Satellite*).

Untuk mendukung misi satelit maka perlu dilakukan aktivitas operasi satelit yaitu kegiatan rutin yang dilakukan oleh sebuah Fasilitas Pengendali Misi [*Mission Control Facility*]. Aktivitas ini dapat dibedakan dalam dua bagian yakni pelaksanaan misi dan pemantauan status kesehatan satelit. Pelaksanaan misi dilakukan melalui serangkaian pengendalian sikap, pengaktifan muatan satelit dan pengiriman data muatan tersebut ke stasiun bumi penerima. Status satelit dapat dipantau melalui pengambilan data telemetri baik yang *real time* maupun telemetri jangka panjang.

Dalam tulisan ini akan dibahas mengenai sistem toleransi kesalahan dalam komunikasi data di dalam OBDH satelit, sehingga operasi satelit dapat tetap berjalan walaupun terjadi kesalahan.

2 SISTEM TOLERANSI KESALAHAN

Sistem toleransi kesalahan adalah sistem yang tetap dapat melaksanakan fungsinya meskipun terjadi kesalahan

sampai pada tingkat tertentu pada sistem tersebut. Kesalahan tersebut dapat berupa *byzantine fault*, *timing fault*, *ommission fault* atau *crash fault*

Agar memiliki sifat toleransi kesalahan, suatu sistem harus mempunyai cadangan komponen yang berlebih. Komponen cadangan ini diperlukan pada waktu sistem mengalami kegagalan. Pada kondisi normal komponen cadangan ini tidak diperlukan. Jika terjadi kegagalan pada sistem, komponen cadangan ini baru akan dipakai.

Redundansi pada sistem toleransi kesalahan dapat berupa : redundansi perangkat keras, redundansi perangkat lunak, redundansi waktu, dan redundansi informasi. Redundansi perangkat keras menyediakan perangkat keras tambahan (umumnya dua atau tiga kali) pada sistem. Redundansi perangkat lunak menyediakan versi perangkat lunak yang berbeda sehingga ketika terjadi kesalahan versi yang lain akan digunakan. Redundansi waktu memanfaatkan kelonggaran jadwal pada masing-masing *task* sehingga *task* dapat dijalankan jika diperlukan dan masih memenuhi *critical deadline* masing-masing *task* tersebut. Sedangkan redundansi informasi menkodekan data sehingga jumlah kesalahan bit dapat dideteksi dan/atau dikoreksi.

2.1 Perangkat Keras Toleransi Kesalahan

Terdapat tiga tipe perangkat keras toleransi kesalahan yakni redundansi perangkat keras pasif atau statis, aktif atau dinamis dan *hybrid*. Redundansi perangkat keras pasif adalah menutup [*masking*] kesalahan dengan cara menyembunyikan kesalahan yang muncul. Dalam tipe ini tidak diperlukan kegiatan pendeteksian kesalahan namun memerlukan *fault containment* atau membatasi penyebaran kesalahan. Pendekatan redundansi perangkat keras aktif melibatkan kegiatan pendeteksian, penentuan lokasi dan membatasi penyebaran kesalahan, dan

kemudian melakukan perbaikan dengan konfigurasi ulang. Redundansi perangkat keras *hybrid* menggabungkan kedua pendekatan di atas. Bentuk-bentuk ketiga redundansi tersebut adalah sebagai berikut :

2.1.1 Redundansi perangkat keras pasif

- a. *Triple Modular Redundancy* (TMR) adalah metode dimana sebuah perangkat keras M digandakan tiga kali lipat dan bekerja secara paralel. Keluaran dari ketiga unit akan dipilih oleh *voter* dan memilih nilai mayoritasnya. Sistem ini dapat menutupi kesalahan pada satu unit M.
- b. *N-Modular Redundancy* (NMR) adalah metode dimana sebuah perangkat keras M digandakan N kali lipat dan bekerja secara paralel demikian pula halnya dengan *voter* V dimana N bernilai ganjil. Keluaran dari N unit dikirimkan ke N *voter* dan memilih nilai mayoritasnya.

2.1.2 Redundansi perangkat keras aktif

- a. *Duplication with Comparison* adalah metode dimana • dua modul identik melakukan proses komputasi yang sama secara paralel dan hasilnya dibandingkan. Skema ini hanya dapat mendeteksi kesalahan namun tidak memberikan toleransi.
- b. *Standby Sparing* adalah metode dimana modul utama dalam status operasional dan satu atau lebih modul bertindak sebagai cadangan atau dalam status *standby*. Blok deteksi *error* digunakan untuk menentukan kapan dan modul mana yang mengalami kesalahan. Modul yang mengalami kesalahan dibuang dari sistem dan digantikan dengan cadangan.
- c. *Pair-and-a-spare Technique* adalah kombinasi skema *stand by sparing* dan *duplication with comparison*. *Switch* akan meneruskan keluaran dari dua modul dan membandingkannya. Selama kedua keluaran tersebut sesuai dengan spesifikasi, modul cadangan tidak

digunakan. Namun bila ditemukan *error*, maka *switch* akan mencari modul yang mengalami kesalahan dan memilih modul penggantinya.

2.1.3 Redundansi perangkat keras *hybrid*

- a. *Self-purging Redundancy* adalah metode dimana semua modul M1 ... Mn berpartisipasi aktif dalam sistem. Setiap modul mempunyai kemampuan untuk melepaskan dirinya dari sistem bila mengalami kesalahan. Ini fitur yang sangat menarik karena *hot replacement* dapat dilakukan tanpa mengganggu proses sistem.
- b. *N-modular Redundancy with Spares* berbasiskan pada NMR dengan tambahan blok *disagreement detector* yang digunakan untuk membandingkan keluaran *voter* dengan keluaran dari masing-masing modul. Modul yang tidak sesuai diberi label *faulty* dan dikeluarkan dari sistem dan kemudian modul cadangan diaktifkan.
- c. *Triple-duplex Architecture* merupakan kombinasi antara TMR dan *duplication with comparison*. TMR dapat menutupi kesalahan tanpa menginterupsi proses sistem yang sedang berlangsung sedangkan *duplication with comparison* dapat mendeteksi kesalahan dan modul yang mengalami kesalahan tidak dilibatkan dalam proses *voting*. Dengan skema ini, dua modul yang mengalami kesalahan dapat ditoleransi.

2.2 Perangkat Lunak Toleransi Kesalahan

Teknik perangkat lunak toleransi kesalahan dibagi menjadi dua kelompok yakni *single-version* dan *multi-version*. Teknik *single-version* ditujukan untuk meningkatkan kemampuan toleransi kesalahan dari satu modul perangkat lunak sedangkan teknik *multi-version* menggunakan redundansi modul perangkat lunak yang dibangun mengikuti aturan-aturan *design diversity*.

3 KESALAHAN DALAM SISTEM SATELIT

3.1 Efek Radiasi dan Ionisasi

Besarnya radiasi untuk satelit yang berada pada orbit LEO adalah antara 10 sampai 20 krad, tetapi jumlahnya tergantung dari berapa lama satelit berada di luar sabuk *Van Allen*, yang berfungsi sebagai perisai bagi partikel energetik yang terperangkap. Penyerapan radiasi dapat menyebabkan kerusakan sementara atau selamanya pada material. Contoh, kebocoran arus bertambah dan logika bit berubah.

Untuk satelit dengan misi LEO [*Low Earth Orbit*] yang menggunakan waktunya di bawah sabuk *Van Allen* harus memiliki toleransi 100 krad. Untuk misi Polar toleransinya 100k sampai 1Mrad pertahun. Untuk orbit equatorial diharapkan memiliki toleransi beberapa krad pertahun. Indikasi adanya kerusakan pada komponen OBC antara lain, bertambahnya konsumsi daya komponen, dan ada laporan yang menyebutkan konsumsi daya bertambah 2x lipat pada saat komponen rusak. Sebagai tambahan peralatan yang rusak dapat beroperasi pada *dock* rendah.

Efek dari radiasi yang lain disebut *displacement damage*. *Displacement damage* terjadi di luar angkasa ketika proton dan elektron dengan kecepatan tinggi menabrak atom-atom semikonduktor, menyebabkan kerusakan struktur. Tubrukan elastis ini terjadi untuk kejadian *charge particles* pada energi 1 Mev atau kurang. Pada energi diatas 8 Mev, proton dapat menyebabkan tubrukan *inelastic nuclear* dengan energi yang cukup merusak struktur silikon. Beberapa konsekuensi akibat *displacement damage* antara lain, berkurangnya penguatan transistor bipolar, bertambahnya kebocoran arus, dan mengurangi gerak pembawa.

Pada tahun-tahun belakangan ini satelit mempunyai pengalaman *bit flips* dalam memori dan perangkat pemroses. Perubahan keadaan ini lebih disebabkan oleh radiasi partikel *single energetic*.

kemudian disebut *single event upset* Sinar *galactic* dan solar *cosmic* yang dibentuk dari *nuclei* ion-ion berat merupakan penyebab terjadinya *single event* Ada beberapa tipe *single event effects* seperti, *single event burnout*, *single hard error*, dan lain-lain.

SEU (*Single Event Upset*) merupakan *error* yang tidak merusak biasanya mempengaruhi sel-sel logika (contohnya: flip-flop) yang mana menyebabkan bit pada memori mengalami perubahan logika (0 ke 1 atau 1 ke 0). Karena SEU normalnya tidak menyebabkan perubahan permanen, maka sering disebut *soft errors*. Frekuensi SEU merupakan fungsi dari kepadatan pengepakan pada IC (*Integrated Circuitry*. VLSI (*Very Large Scale Integration*) lebih *prone* terhadap efek SEU daripada teknologi IC skala medium dan besar.

SEL (*Single Event Latch up*) merupakan produk SEU yang menyebabkan kerusakan permanen, dan kadang-kadang disebut *hard error*. Partikel radiasi ionisasi menciptakan koneksi langsung VDD (*Voltage Drain*, voltase antara *drain* dan *common*) ke *ground* dalam transistor IC. Jika chip IC tidak dilindungi oleh beberapa pembatas arus, seperti resistor seri yang dihubungkan ke pin power, perangkat kemungkinan akan mengalami kerusakan permanen. Jika terjadi SEL, IC yang dilindungi oleh resistor pembatas arus biasanya diperlukan untuk mematikan power agar keadaan terkunci bisa dihapus.

3.2 Suhu

Suhu yang ekstrem di niang angkasa merupakan masalah lain untuk sistem satelit. Nilai temperatur yang sangat rendah atau terlalu tinggi dapat menyebabkan kerusakan komponen elektronika. Pada saat suhu naik, daya yang diperlukan berkurang 5% - 10% dan kehandalan berkurang sampai dengan 90%. Hal itulah mengapa suhu tidak digunakan untuk mengurangi konsumsi daya.

3.3 Gangguan-Gangguan Orbit

Hukum Kepler didasarkan pada asumsi bahwa distribusi massa benda utama *spherical simetrik* serta tidak memperhitungkan gaya-gaya non-gravitational maupun gravitasi benda ketiga. Akibatnya orbit satelit yang sebenarnya tidak akan tepat mengikuti hukum Kepler. Ada beberapa hal yang menyebabkan orbit satelit tidak tepat mengikuti hukum Kepler yaitu yang disebut dengan gangguan orbit:

- Gangguan orbit akibat bentuk bumi yang tidak bulat,
- *Atmospheric drag* (gaya hambat atmosfer),
- Gaya gravitasi bulan,
- Radiasi surya.

4 SISTEM SATELIT LAPAN A2

Satelit LAPAN A2 merupakan pengembangan satelit LAPAN Tubsat. Satelit ini akan diterbangkan dengan cara *piggy back*. Oleh karena itu dimensi strukturnya akan dibatasi hingga 60cm x 60cm x 80cm. LAPAN A2 kemungkinan akan terbang dalam SPPO [*Sun Synchronous Polar Orbit*].

Beberapa pengembangan dari sistem LAPAN Tubsat yaitu penambahan sistem GPS untuk mendapatkan data waktu satelit dan posisi orbit. Sistem ADCS [*Attitude Determination and Control System*] untuk mendapatkan stabilitas sikap selama akuisisi data observasi.

LAPAN A2 akan dilengkapi dengan 4 buah solar panel bertipe *triple junction*. Dua buah paket baterai Li-ion (Lithium-ion) dengan *charge control* akan digunakan sebagai penyimpan daya. Masing-masing paket terdiri dari 4 cells Li-ion dengan kapasitas 6.5 Ah. Maksimum tegangan yang dihasilkan adalah 16.8 V dengan tegangan bus unregulated akan dioperasikan pada tegangan 13V -16V. Masing-masing paket baterai Li-ion mempunyai dimensi 11 x 11 x 9 cm dengan massa tidak lebih dari 1.5 kg.

LAPAN A2 akan menggunakan sistem kontrol sikap yang sama dengan

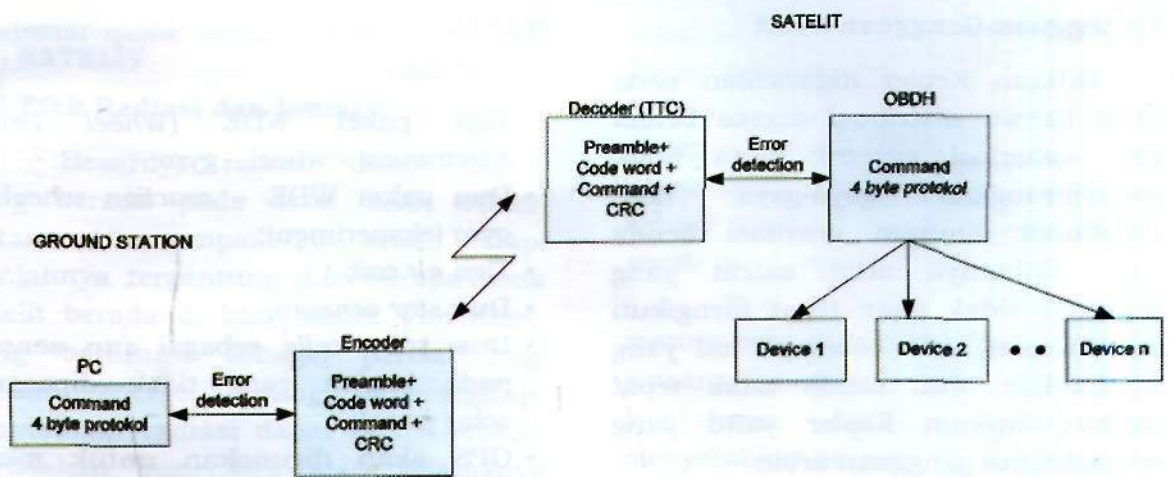
LAPAN Tubsat. Kontrol sikap tersebut terdiri dari:

- Tiga paket WDE (*Wheel Drive Electronics*) - wheel - gyro;
- Dua paket WDE - *reaction wheel* - gyro (eksperimen);
- Tiga *air coil*;
- Dua *star sensors*;
- Dua *solar cells* sebagai sun sensor pada bagian yang tidak dipasang solar panel;
- GPS akan digunakan untuk mendapatkan waktu yang tepat [*exact time*] dan posisi untuk mengambil gambar.

Protokol komunikasi yang dirancang adalah sebuah komando yang dikirim via radio terdiri dari 2 *preamble* dan *general command*. *Preamble* pertama adalah 8 byte untuk sinkronisasi dengan modem. *Preamble* berikutnya adalah 4 byte yang merupakan *code word* untuk sinkronisasi dan bagian terakhir adalah *device common*.

Tabel4-1: LIST KOMPONEN SATELIT LAPAN A2

Subsistem	Komponen
Payload	- 1000 mm Video Camera
	- 50 mm Video Camera
	- matrix CCD imager
Power	- Lithium Ion Batteries
	- Solar Panels
	- Solar Cells as Sun Sensor
ACS	- Power Control Units
	- Gyros
	- Reaction Wheels
	- Star Sensors
	- GPS
Data Handling	- Coils
	- OBDH (primer)
Communication	- OBDH (experimen)
	- TTC Transceiver
	- UHF Antennas
	- Eksperimental TTC+UHF antenna
	- Pemancar S-band
	- Antena S-band Antennas



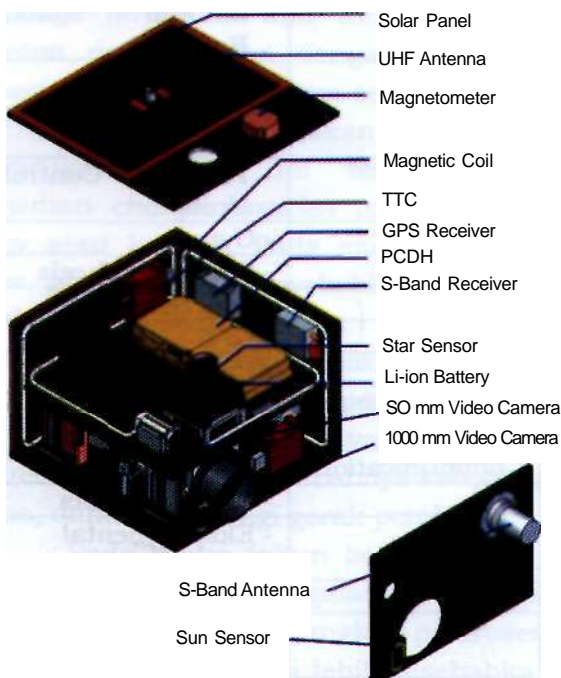
Gambar 4-1: Proses pengiriman komando

Setelah protokol sampai TTC maka TTC akan melanjutkan komando tersebut ke OBDH dengan membuang sebuah *header* dan menyisakan 4 byte protokol untuk dilanjutkan ke OBDH. *Error Detection* yang digunakan dalam hal ini adalah memantulkan (*echo*) atau mengembalikan tiap byte dari 4 byte. Jika hasil *echo* tidak sama maka komando dianggap mengalami gangguan. *Interfacing* dengan setiap subsistem juga menggunakan protokol 4 byte. Semua komando akan sampai ke subsistem OBDH harus dalam protokol 4 byte. Seandainya perlu ditambahkan kode lain maka kode tersebut harus dimasukan ke dalam protokol 4 byte. Protokol 4 byte bekerja dalam sebuah basis *handshake*.

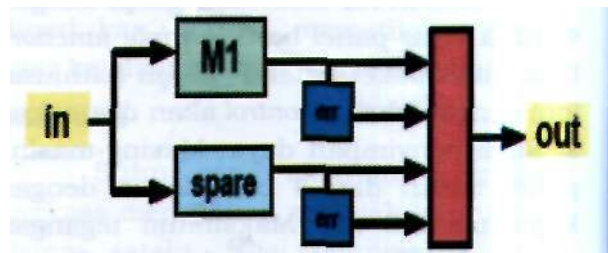
5 ANALISA DAN IMPLEMENTASI TOLERANSI KESALAHAN

Toleransi kesalahan adalah sistem yang menggunakan dan mengelola redundansi dengan tepat. Agar sistem tetap bekerja dengan benar meskipun dideteksi adanya kesalahan pada beberapa bagian sistem, satelit harus memiliki cadangan yang harus segera bekerja seketika.

Sistem toleransi kesalahan akan diimplementasikan dalam subsistem OBDH satelit. Dalam hal ini OBDH akan dirancang dengan menggunakan dua buah prosesor, dimana prosesor kedua akan memonitor prosesor pertama. Seandainya prosesor pertama mengalami *failure* sehingga Power Control Unit meng-*o*^*j*kan power ke prosesor, maka prosesor kedua akan mengambil alih fungsi dari prosesor pertama.



Gambar 4-2: Satelit LAPAN A2

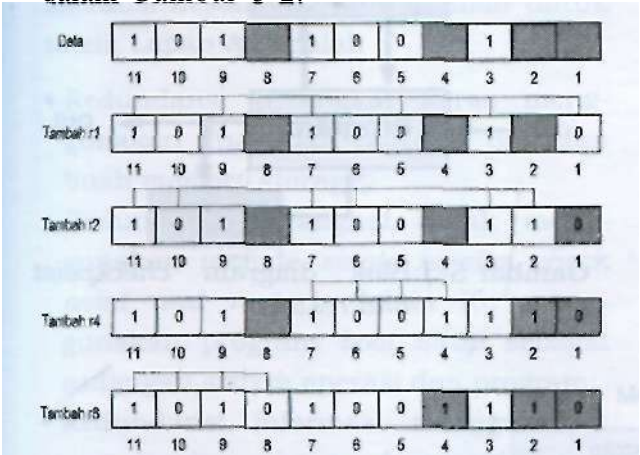


Gambar 5-1: Penggandaan prosesor

Pemilihan ini dikarenakan keterbatasan sistem perangkat keras yang dapat ditumpangkan di dalam sistem satelit ditinjau dari sisi ukuran dan konsumsi *power* yang akan berlipat sesuai dengan jumlah cadangan yang

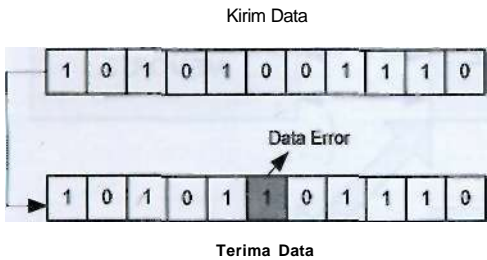
digunakan. Sehingga pemilihan sistem Triple Modular Redundancy (TMR) dan N-modular redundancy (NMR) tidak memungkinkan.

Kesalahan yang terjadi pada memory RAM, dalam hal RAM tidak diredundansi secara perangkat keras. Untuk itu akan digunakan metode hamming code untuk mengoreksi kesalahan satu bit (single flip). Contoh data 7 bit 1011001 akan dikodekan dalam kode word menjadi seperti tampak dalam Gambar 5-2.



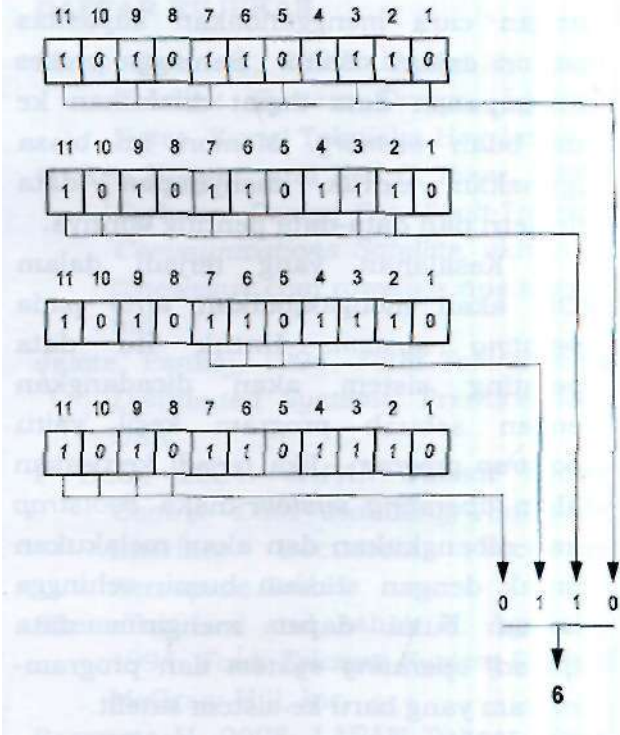
Gambar 5-2: Mengubah data menjadi kode word hamming

Sehingga data yang akan dikirim dalam kode word adalah 10101001110. Kode word tersebut akan diimplementasikan pada setiap byte kode komando. Sehingga setiap byte akan mendapat tambahan 4 bit data. Untuk menguji hasil dengan cara menyisipkan error dengan contoh sebagai berikut :



Gambar 5-3: Ilustrasi terjadinya error data

Dalam Gambar 5-3 tampak bit ke-6 mengalami error yaitu dengan berubahnya nilai 0 menjadi 1. Dengan metode hamming code terjadinya error dapat terdeteksi seperti tampak dalam analisa Gambar 5-4.



Gambar 5-4: Hasil mendeteksi posisi bit yang mengalami error

Dengan diketahuinya posisi error yang terjadi, maka dengan metode hamming nilai tersebut akan dikembalikan ke nilai aslinya sehingga data menjadi valid. Setelah data terkoreksi maka kode word tersebut akan diencodekan kembali pada nilai aslinya.

Kode hamming dapat mendeteksi dua bit flip tetapi tidak dapat mengoreksinya. Jika terjadi tiga atau lebih flip maka error tersebut mungkin tidak dapat terdeteksi. Untuk mengatasi hal tersebut maka akan dimanfaatkan sistem bank dalam memory. Setiap bank dalam memory akan dibagi menjadi 2, dimana setiap bank merupakan duplikasi satu sama lain. Seandainya terjadi error dalam satu bank maka bank lain yang merupakan duplikasinya akan digunakan.

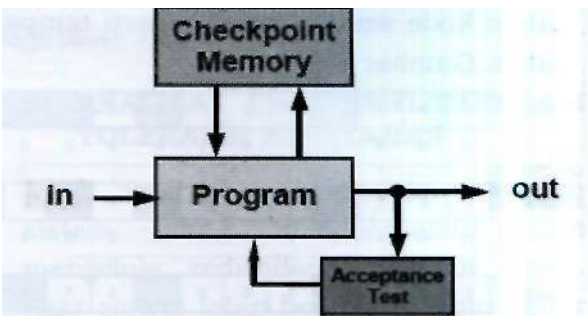
Kesalahan yang terjadi dalam flash memory diatasi dengan redundansi perangkat keras. Dikarenakan flash memiliki kapasistas yang lebih besar sehingga tidak diperlukan sistem mirror. Hal tersebut dapat diganti

dengan cara menggandakan kapasitas memori dalam sistem. Sehingga proses penyimpanan data dapat dilakukan ke dua buah *memory*. Memori ini biasa digunakan untuk menyimpan data telemetri dan data-data penting lainnya.

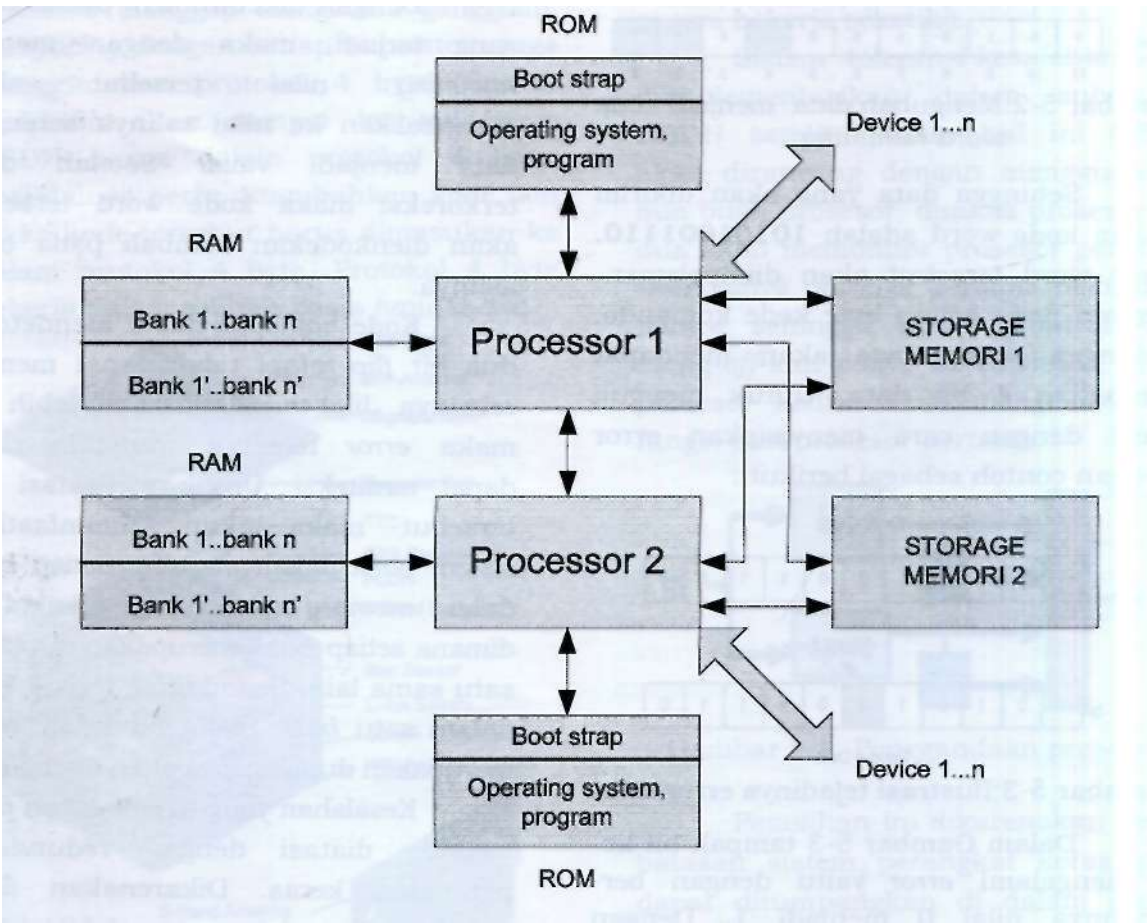
Kesalahan yang terjadi dalam ROM akan mengakibatkan *error* pada *operating system*. Untuk itu data *operating sistem* akan dicadangkan dengan sebuah program kecil yaitu *bootstrap* program. Jika terjadi kerusakan dalam *operating system* maka *bootstrap* akan dibangkitkan dan akan melakukan kontak dengan stasiun bumi, sehingga stasiun bumi dapat mengirim data (*upload*) *operating system* dan program-program yang baru ke sistem satelit.

Redundansi perangkat lunak yang akan diimplementasikan adalah *single version check point and restart*.

Dikarenakan keterbatasan sistem memori maka metode inilah yang paling tepat untuk dipilih. Metode ini menggunakan "*recovery blocks*" dan "*acceptance tests*". Semua *state* dicatat dan seandainya terdapat *acceptance test* yang gagal dengan segera sistem kembali kepada *state* sebelum terjadi *error*. Alternatif lain adalah keluar dari *state error* menuju *state error handling run time* untuk dieksekusi.



Gambar 5-5: Blok diagram *checkpoint and restart*



Gambar 5-6: Blok diagram implementasi toleransi kesalahan

Dari analisa di atas maka dapat didesain sebuah blok diagram sistem toleransi kesalahan OBDH secara keseluruhan seperti tampak dalam Gambar 5~6.

6 KESIMPULAN

Untuk membuat sistem satelit bertambah *reliable* maka perlu dirancang sistem toleransi kesalahan di dalam OBDH yang merupakan otak dalam sistem satelit. Implementasi toleransi kesalahan yang optimal untuk satelit Lapan A2 adalah :

- Redundansi perangkat keras menggunakan dua buah prosesor dan dua buah *memory storage*;
- Redundansi perangkat lunak menggunakan metode *single version check point and restart*. Selain itu menggunakan program *boot strap* sebagai cadangan sistem operasi dan program;
- Redundansi informasi menggunakan metode *hamming code* untuk *error detection* dan *co'rection* serta metode *mirroring* data.

Pemilihan redundansi tersebut telah mempertimbangkan keterbatasan dari konsumsi power dan ukuran yang dibatasi oleh sistem satelit secara keseluruhan.

DAFTAR RUJUKAN

- Elena Dubrova, 2007. *Design of Fault Tolerant System Design*, Lecture Notes, Kungl Tekniska Hogskolan.
- G. Ken Hunter and Neil C. Rowe, 1999. *Software Design For Fault-Tolerant Communications Satellite*, gkhunter@newsguy.com, rowe@cs.nps.navy.mil.
- Jalote, Pankaj, 1994. *Fault Tolerance in Distributed Systems*, Prentice-Hall, USA.
- P Butz, 2004. *LAPAN Tubsat Power Control Data Handling*, Software Interface Document, Vectronic Aerospace GmbH.
- Shem-Tov Levi and Ashok K. Agrwala, 1994. *Fault Tolerant System Design* McGraw-Hill, Inc.
- Soewarto H, 2008. *LAPAN Tubsat, First Indonesian Microsatellite*, LAPAN.
- Tim LAPAN A2, 2008. *Preliminary Design of Lapan-A2*, LAPAN.
- Ugur YENIER, 2002. *Fault Tolerant Computing In Space Environment And Software Implemented Hardware Fault Tolerance Techniques*, Bosphorus University, Istanbul.
- Wertz, James R. and Larson, Wiley J., 1999. *Space Mission Analysis and Design*, Kluwer Academic Publishers, USA.
- www.cs.fiu.edu/ndounegt/cop3402/hamming.html