

Perangkat Lunak Prediktor Pass Satelit

Wellem O. Sineri¹, Yohanes Y. Rumbewas², Janne AC. Wejai³, Kamirul⁴

Lembaga Penerbangan Antariksa Nasional

Article Info

Article history:

Received Jun 12th, 2021

Revised Aug 20th, 2021

Accepted Aug 31th, 2021

Keyword:

stasiun bumi
countdown timer
sdp4, sgp4

ABSTRACT

Stasiun bumi biak merupakan salah satu stasiun pengendali satelit LAPAN seperti LAPAN-A1 (TUBSAT), LAPAN-A2(ORARI), dan LAPAN-A3 (IPB). Setiap harinya, di Balai Kendali Satelit, Pengamatan Antariksa dan Atmosfer, dan Penginderaan Jauh, Biak rutin melaksanakan kegiatan akuisisi data satelit LAPAN. Untuk melaksanakan hal tersebut, para operator harus sudah berada di ruang kendali satelit sebelum satelit-satelit tersebut lewat guna melaksanakan beberapa persiapan terkait kendali satelit. Dalam hal ini, dapat terjadi kesalahan dalam melihat dan membaca jadwal terjadinya pass satelit dikarenakan selama ini para operator menggunakan satelit pass prediktor dari perangkat lunak yang sudah ada namun tidak dilengkapi dengan countdown timer ataupun alarm. Tujuan dari penelitian ini agar mampu merancang bangun perangkat lunak prediktor pass satelit yang mampu memprediksi waktu terjadinya pass atau kemunculan satelit di suatu ground station. Metode penelitian yang digunakan yaitu pendekatan SDP4 dan SGP4 merupakan pendekatan (algoritma prediksi) ini, dapat dihitung waktu-waktu terjadinya pass satelit di lokasi tertentu di bumi beserta lintasan (groundtrack) satelit tersebut. Berdasarkan permasalahan tersebut maka dilakukan rancang bangun perangkat lunak prediktor pass satelit yang dapat memprediksi dan menampilkan informasi terkait satelit yang akan lewat. Perangkat lunak ini akan dilengkapi dengan fitur countdown timer dan alarm agar operator mendapatkan peringatan dan dapat mempersiapkan kebutuhan sebelum terjadinya pass.

Copyright © 2021 STMIK Triguna Dharma.
All rights reserved.

Corresponding Author: *First Author

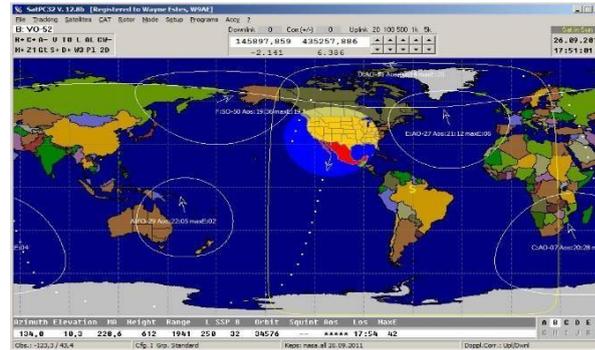
Nama : Wellem Obed Sineri

Instansi : Lembaga Penerbangan Antariksa Nasional (LAPAN)

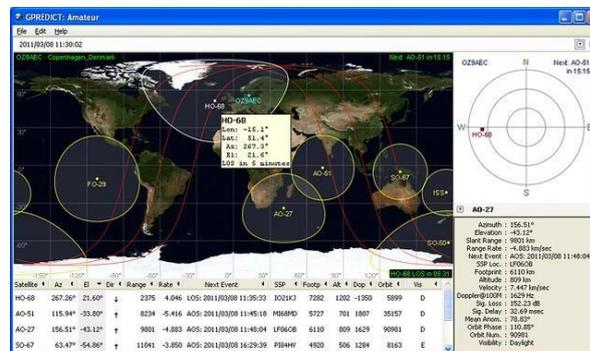
Email: obedsineri@gmail.com

1. PENDAHULUAN

Balai Kendali Satelit, Pengamatan Antariksa dan Atmosfer, dan Penginderaan Jauh, Biak merupakan salah satu balai yang bertugas melaksanakan kegiatan akuisisi data *house keeping* satelit LAPAN-A1, LAPAN-A2, dan LAPAN-A3. Untuk melaksanakan hal tersebut, para operator harus sudah berada di ruang kendali satelit sebelum satelit-satelit tersebut lewat guna melaksanakan beberapa persiapan terkait kendali satelit. Dalam hal ini, dapat terjadi kesalahan dalam melihat dan membaca jadwal terjadinya pass satelit atau keterlambatan dalam melakukan persiapan dikarenakan selama ini para operator menggunakan satelit pass prediktor dari perangkat lunak yang sudah ada namun tidak dilengkapi dengan *countdown timer* ataupun alarm sebagai pengingat. Contoh perangkat lunak tersebut adalah SatPC32 dan GPredictor yang hanya menampilkan waktu-waktu terjadinya pass



(a)



(b)

Gambar 1. (a) Perangkat lunak SatPC dan (b) GPredict, hanya menampilkan informasi pas satelit tanpa adanya *countdown timer* dan alarm.

Prediksi satelit pass dapat diimplemmentasikan melalui beberapa pendekatan beberapa diantaranya adalah pendekatan SDP4 dan SGP4. Melalui pendekatan (algoritma prediksi) ini, dapat dihitung waktu-waktu terjadinya pass satelit di lokasi tertentu di bumi beserta lintasan (*groundtrack*) satelit tersebut. Informasi terkait waktu lewat satelit dan fitur alarm pada perangkat lunak yang akan dirancang sangat dibutuhkan agar operator dapat tepat waktu dalam mempersiapkan kebutuhan dan pengecekan sebelum satelit lewat seperti kondisi antenna dan perangkat komunikasi ke satelit [1].

Oleh karenanya, akan dilakukan rancang bangun perangkat lunak *prediktor pass satelit* yang dapat memprediksi dan menampilkan informasi terkait satelit yang akan lewat. Perangkat lunak ini akan dilengkapi dengan fitur *countdown timer* dan alarm agar operator mendapatkan peringatan dan dapat mempersiapkan kebutuhan sebelum terjadinya pass. Berdasarkan latar belakang masalah yang telah diuraikan di atas, dapat diambil rumusan masalah sebagai berikut

- Bagaimana melakukan prediksi waktu terjadinya pass satelit pada stasiun bumi ?
- Bagaimana melakukan perhitungan *countdown timer* terhadap terjadinya satelit pass ?
- Bagaimana proses menyalakan alarm penanda persiapan pass akan terjadi dan ketika pass dimulai ?

Tujuan Penelitian

- Perangkat lunak *prediktor pass satelit* yang mampu memprediksi waktu terjadinya *pass* atau kemunculan satelit di suatu *ground station*. Perangkat lunak ini juga direncanakan mampu menampilkan *countdown timer* dan menyalakan *alarm* pada saat terjadinya pass satelit.
- Mempelajari sistem dan teori yang berkaitan dengan proses prediksi terjadinya pass satelit.
- Membandingkan kinerja perangkat lunak dengan metode yang sudah ada (terdapat) pada beberapa perangkat lunak lain.
- Meningkatkan kemampuan SDM yang ada dengan melakukan pengembangan sistem perangkat lunak *prediktor pass satelit*

2. METODE PENELITIAN

Adapun metoda ataupun tahapan yang direncanakan akan dilakukan untuk merancang bangun perangkat lunak *prediktor pass satelit* adalah sebagai berikut

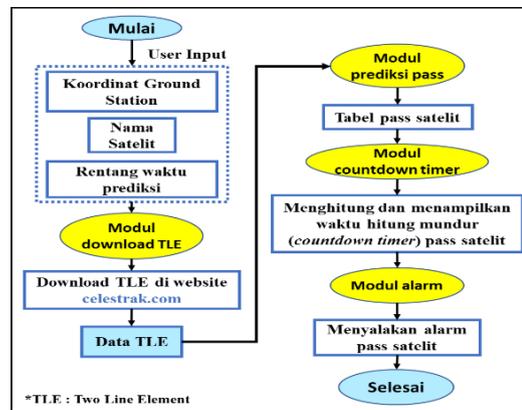
a. Menentukan kebutuhan dan spesifikasi perancangan.

Dalam proses pembuatan perangkat lunak prediktor pass satelit dibutuhkan adalah :

- a) Perangkat lunak Visual Studio yang sudah terinstal bahasa pemrograman c++
- b) Notepad ++
- c) Komputer Dekstop

b. Melakukan studi literatur.

Tahapan yang digunakan dalam memprediksi waktu terjadinya pass satelit. Pada metode ini dilakukan untuk merangkum apa saja proses dan tahapan yang harus dilakukan untuk memprediksi pass satelit beserta fitur tambahannya. Secara ringkas, tahapan atau proses yang akan diterapkan pada perangkat lunak Prediktor Pass Satelit ditampilkan pada Gambar 2 berikut :



Gambar 2. Diagram alir proses prediksi pass satelit

c. Pembuatan *graphic user interface* (GUI)

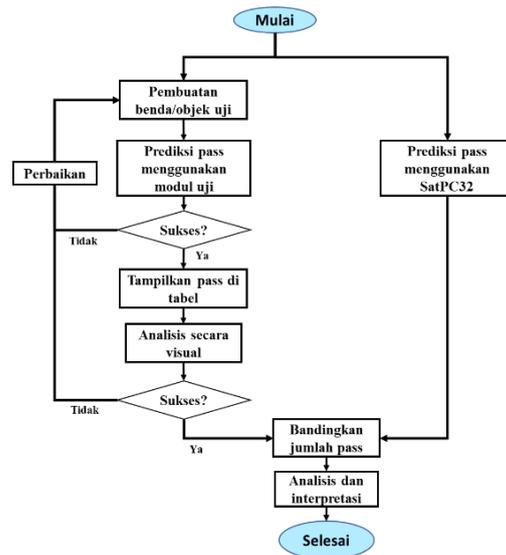
Adapun proses desain *GUI* secara keseluruhan perangkat lunak prediktor pass satelit di desain menggunakan 4 (empat) buah modul pendukung yang memiliki fungsi sebagai berikut :

Tabel 1. Modul utama perangkat lunak prediktor pass satelit

Nama modul	Fungsi / Kegunaan
Modul download TLE	Memperbarui <i>Two Line Element</i> (TLE) melalui jaringan internet.
Modul prediksi <i>pass</i>	Melakukan perhitungan untuk memprediksi waktu lewat satelit pada suatu ground station (<i>pass</i>)
Modul <i>countdown timer</i>	Melakukan penghitungan mundur terhadap terjadinya suatu <i>pass</i>
Modul alarm	Menyalakan alarm penanda persiapan <i>pass</i> akan terjadi dan ketika <i>pass</i> sedang dimulai.

d. Penetapan Konfigurasi Pengujian

Pengujian dilaksanakan mengikuti konfigurasi seperti yang ditampilkan pada diagram alir berikut :



Gambar 3. Konfigurasi/alur yang diterapkan pada proses pengujian modul prediksi pass untuk perangkat lunak prediksi pass satelit.

e. Pembuatan benda uji berupa modul prediksi pass khusus versi pengujian

Modul prediksi pass versi pengujian berbeda dengan modul aslinya dalam hal cara membaca data masukan untuk modul. Pada modul ini, alamat file TLE di komputer, nama satelit, koordinat ground station, serta rentang waktu prediksi dituliskan langsung pada badan program, sedangkan pada modul versi aslinya, masukan ini berupa sebuah variable yang disimpan dalam sebuah file konfigurasi. Tujuan dari pembuatan modul prediksi pass versi pengujian ini adalah untuk mempermudah proses pengujian agar tidak harus melibatkan modul-modul lain di luar modul ini sendiri, karena proses pengujian dilakukan secara spesifik pada modul tertentu (dalam hal ini modul prediksi pass). Berikut ditampilkan cuplikan *source code* modul prediksi pass versi pengujian [2].

Input didefinisikan langsung didalam program dalam bentuk *static input*

```
String^ Satname = "LAPAN-A2";
String^ TLEPath = "C:\Users\User\Desktop\resource.txt";

double Latitude = -1.17560; // Biak Ground Station
double Longitude = 136.1010;
double Altitude = 0.121217;

System::DateTime start_date = MyStringToDateTime("07-10-2019 00:00:00");
System::DateTime end_date = MyStringToDateTime("07-10-2019 23:59:59");

Coordinate geo(Latitude, Longitude, Altitude);
Tle tle(satname, TLEPath);
SGP4 sgp4(tle);

std::list<struct PassDetails> pass_list;
pass_list = GeneratePassList(geo, sgp4, start_date, end_date, 180);

int npass = sizeof(pass_list);

if (pass_list.begin() == pass_list.end())
{
}
else
{
    std::list<struct PassDetails>::const_iterator i = pass_list.begin();
    do
    {
        //satname
        System::String^ SatName = msclr::interop::marshal_as<System::String^>(satname);
```

Gambar 4. Cuplikan *source code* modul prediksi pass versi pengujian.

f. Sistem perolehan dan pengolahan data

- a) Data yang akan diperoleh dalam pengujian ini data berupa hasil prediksi pass yang terdiri atas :
- Nama Satelit

- Elevasi maksimum
 - Waktu mulai pass /AOS/Acquisition Of Signal
 - Waktu berakhir pass /LOS/Los Of Signal
- b) Agar jumlah pass yang dihasilkan oleh modul yang dirancang dan yang dihasilkan oleh SatPC32 dapat dibandingkan, maka dilakukan pengolahan pada data uji yaitu dengan menghitung jumlah pass-nya masing-masing dalam rentang waktu masing-masing 24 jam.

3. ANALISA DAN HASIL

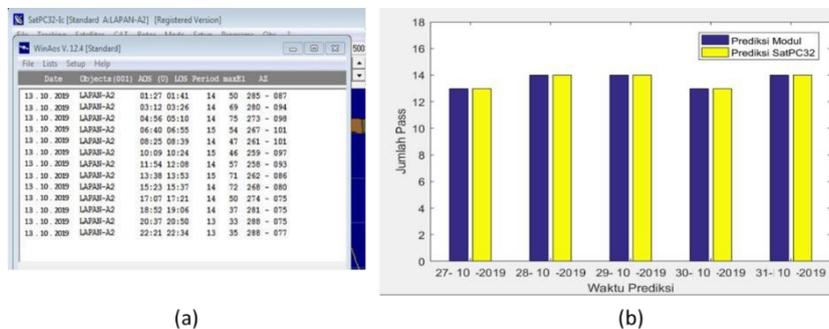
3.1. Modul Prediksi Pass

Berikut adalah hasil prediksi pass yang telah dihasilkan modul dan berhasil ditampilkan pada tabel. Dari hasil ini, modul menampilkan hasil prediksi. Selain itu, modul juga telah mampu menampilkan data sesuai kolomnya seperti pada Gambar 4. Hasil ini menggunakan satelit LAPAN-A2 sebagai data masukan, stasiun Bumi Biak sebagai ground station, dan merupakan hasil prediksi untuk 13 Oktober 2019.

SATELLITE	MAX EL.	AOS	LOS	DUR.	CODI
LAPAN-A2	51.52	01:27:34	01:41:34	14	BI
LAPAN-A2	75.5	03:11:55	03:26:06	14.18	BI
LAPAN-A2	75.01	04:56:19	05:10:34	14.25	BI
LAPAN-A2	54.88	06:40:48	06:54:59	14.18	BI
LAPAN-A2	46.29	08:25:19	08:39:27	14.13	BI
LAPAN-A2	46.45	10:09:49	10:23:59	14.17	BI
LAPAN-A2	55.35	11:54:16	12:08:33	14.28	BI
LAPAN-A2	75.6	13:38:40	13:53:04	14.4	BI
LAPAN-A2	75.31	15:23:08	15:37:29	14.35	BI
LAPAN-A2	51.68	17:07:41	17:21:49	14.13	BI
LAPAN-A2	39.02	18:52:19	19:06:10	13.85	BI
LAPAN-A2	34.32	20:36:56	20:50:36	13.67	BI
LAPAN-A2	35.82	22:21:27	22:35:08	13.68	BI

Gambar 5. Modul telah dapat memprediksi dan menampilkan hasil prediksi pass satelit LAPAN-A2 (13-10-2019) pada tabel yang disediakan.

Hasil perbandingan jumlah prediksi untuk 5 hari (27-31 Oktober 2019) menggunakan modul yang dirancang dan perangkat lunak SatPC32 ditampilkan pada Gambar 5 berikut.



Gambar 6. (a) Jumlah pass yang dihasilkan SatPC32 pada 13 Oktober 2019 dan (b) Perbandingan jumlah pass yang dihasilkan modul dan SatPC32 pada rentang 5 hari (dipilih rentang tanggal secara acak).

Berdasarkan analisis perbandingan menggunakan jumlah pass pada Gambar 5 diatas, modul yang dirancang menghasilkan jumlah pass yang sama dengan jumlah pass yang dihasilkan oleh perangkat lunak SatPC32.

Dalam melakukan analisis perbandingan jumlah pass, error rata-rata (dalam %) dapat di hitung dengan persamaan :

$$E = \frac{\sum_{i=1}^5 \left(\frac{|N_{modul}^i - N_{satpc}^i|}{N_{satpc}^i} \times 100\% \right)}{5} \quad (1)$$

dengan $i = 1, 2, \dots, 5$ yang merupakan urutan tanggal (nomor urut hari), N_{satpc} dan N_{modul} masing-masing adalah jumlah prediksi pada hari ke- i oleh SatPC32 dan modul yang diuji. Dengan menggunakan persamaan (1) di atas dan menggunakan data yang telah ditampilkan pada Gambar 5(b), maka error jumlah pass yang dihasilkan oleh modul terhadap SatPC32 secara rata-rata adalah 0% [3], [4].

3.2. Modul Countdown Timer

Membuat benda uji berupa modul countdown timer versi khusus untuk keperluan pengujian. Modul ini berbeda dengan modul aslinya dalam hal cara membaca data masukan untuk modul. Karena data masukan pada modul ini berupa sekumpulan pass hasil prediksi, maka kumpulan pass tersebut di-generate langsung di badan program. Sedangkan pada modul aslinya, kumpulan pass diambil pada tabel. Hal ini dilakukan agar pengujian difokuskan pada kemampuan modul dalam menghitung dan menampilkan hasil *counting* tanpa melibatkan modul dan properti lain (tabel pass). Sehingga, proses analisa dapat difokuskan pada kemampuan modul dalam menghasilkan dan menampilkan hasil *counting*. Selain itu hasil counting juga akan divalidasi dengan sistem *countdown timer online*. Online Clock Counter, "Tick Counter : Create your own countdown to any date for free", (<https://www.tickcounter.com/>) [5].

Input menggunakan kumpulan pass yang di-generate langsung di badan program, bukan dari tabel

```
String Date = "31 - 10 - 2019";
array INPUT = GeneratePassList("LAPAN-A2", "BIAK", Date);
if (NextAOSRowID != 999999){
    DateTime CurrDate = DateTime.Now(true);
    int CurrSec = (CurrDate.Hour() * 3600) + (CurrDate.Minute() * 60) + (CurrDate.Second());
    int AOS = INPUT[1];
    int dtime = AOS - CurrSec;
    NextCDT->Text = SecondToDurationText(dtime);
}
else{
    NextName->Text = "No Visibility";
    NextCDT->Text = "-----";
    NextMaxEl->Text = "-----";
    NextDuration->Text = "-----";
    NextAOS->Text = "-----";
    NextLOS->Text = "-----";
}
```

Gambar 7. Cuplikan source code modul countdown timer versi pengujian.

Sedangkan untuk proses validasi, digunakan *countdown timer* pembandingan yang tersedia di www.tickcounter.com, sehingga untuk proses validasi ini sistem penginderaan yang digunakan adalah *countdown timer* online tersebut, maka data yang akan diperoleh dalam pengujian ini data berupa hasil prediksi pass yang terdiri atas :

- a. *Counting down timer* (data hasil hitung mundur satelit yang akan segera lewat). Data ini didapatkan masing melalui modul yang dirancang dan dari countdown timer online untuk bahan pembandingan.
- b. Properti satelit yang akan segera lewat :
 - Nama Satelit
 - Elevasi maksimum
 - Waktu mulai pass /AOS/Acquisition Of Signal
 - Waktu berakhir pass /LOS/Los Of Signal
 - Durasi pass (dalam menit)
- c. Dari hasil analisis modul yang dirancang, telah mampu menampilkan hasil *counting down* beserta properti satelit terkait pada panel yang telah disediakan

(Nomor hari dalam tahun, 3 November 2019) (waktu)

307 03:51:19

Station: Satellite:

Next: CDT:

AOS: Max: LOS: Dur:

SATELLITE	MAX EL.	AOS	LOS	DUR.	CODE
LAPAN-A2	44.19	00:05:51	00:19:44	13.88	BI
LAPAN-A2	62.5	01:50:12	02:04:18	14.1	BI
LAPAN-A2	88.72	03:34:34	03:48:47	14.22	BI
LAPAN-A2	62.64	05:10:00	05:24:12	14.7	BI
LAPAN-A2	49.09	07:03:31	07:17:39	14.13	BI
LAPAN-A2	45.29	08:48:02	09:02:09	14.12	BI
LAPAN-A2	49.83	10:32:30	10:46:43	14.22	BI
LAPAN-A2	64.33	12:16:54	12:31:16	14.37	BI
LAPAN-A2	89.5	14:01:20	14:15:44	14.4	BI
LAPAN-A2	61.66	15:45:50	16:00:07	14.28	BI
LAPAN-A2	43.99	17:30:26	17:44:27	14.02	BI
LAPAN-A2	35.84	19:15:04	19:28:50	13.77	BI

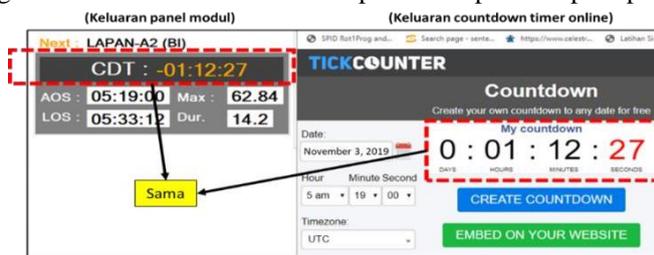
Gambar 8. Modul mampu menampilkan hasil *counting down* satelit yang akan segera lewat.

- d. Pengolahan data dalam pengujian ini dilakukan pada hasil *counting* yang masing-masing dihasilkan oleh modul yang dirancang dan dari *countdown timer online* www.tickcounter.com. Tujuan dari proses ini untuk mengukur seberapa besar error yang dihasilkan oleh modul jika dibandingkan dengan sistem *stopwatch online*. Persamaan yang digunakan adalah

$$E = \frac{|T_{modul} - T_{online}|}{T_{online}} \times 100\% \tag{2}$$

Dengan T_{modul} dan T_{online} masing-masing adalah hasil *counting* yang masing-masing dihasilkan oleh modul yang dirancang dan *countdown timer online* www.tickcounter.com.

- e. Dalam melakukan analisis perbandingan hasil *counting* yang dihasilkan oleh modul dan *countdown timer online*. Dari hasil analisis, terlihat bahwa modul mampu menghasilkan hitungan yang sama dengan yang dihasilkan oleh *conter online* seperti ditampilkan seperti pada Gambar berikut



Gambar 9. Salah satu contoh yang menunjukkan bahwa modul mampu menampilkan hasil hitungan yang sama dengan yang dihasilkan oleh *conter online*

3.3. Modul Alarm

Modul alarm khusus versi pengujian berbeda dengan modul aslinya dikarenakan adanya penambahan source code untuk mendeteksi waktu eksekusi. Dalam hal ini, yang akan dideteksi adalah waktu eksekusi *pre-pass* (alarm persiapan) dan *current pass* (alarm tepat saat satelit mulai lewat). Waktu eksekusi yang dideteksi akan ditampilkan dalam suatu *MessageBox* sehingga hasilnya dapat dibandingkan dengan waktu yang seharusnya. Dengan membandingkan informasi waktu yang ditampilkan dengan waktu terjadinya pass, maka dapat dihitung selisihnya sebagai indikator apakah alarm telah menyala sesuai dengan waktu yang seharusnya.

```

// Deteksi PrePass
int AOSsec = std::stoi(AOSCell.substr(0, 2)) * 3600 + \
std::stoi(AOSCell.substr(3, 2)) * 60 + std::stoi(AOSCell.substr(6, 2));
if ((CurrTODSec < prepassdur) & (CurrTODSec > 0)) {
    dataGridView->Rows[1]->DefaultCellStyle->ForeColor = Color::Black;

    CurrStatus->Text = "PRE PASS ";
    CurrStatus->ForeColor = PrePassColor;
    CurrTODOT->ForeColor = PrePassColor;
    CurrPanel->BackColor = PrePassColor;

    dataGridView->Rows[1]->DefaultCellStyle->BackColor = PrePassColor;
    if ((CurrTODSec >= (prepassdur - AlarmDelayRange)) & \
        (DataAlarm == 0) & (CurrSec == (StartAlarmSec)) & \
        (CurrSec != (StopAlarmSec))) {
        PrePassAlarmOn();

        int PreDateTst = (PreDate.Hour() * 3600) + (PreDate.Minute() * 60) + (PreDate.Second());
        MessageBox->Show("PrePass Time = " + SecondToDurationText(PreDateTst) + \
            " ***** Alarm Executed on " + SecondToDurationText(AOSSec));
    }
}
    
```

Gambar 10. Cuplikan *source code* modul *alarm pre-pass* versi pengujian.

```

// Deteksi Current Pass
if ((CurrSec == AOSsec) & (CurrSec < LOSsec)) {
    CurrStatus->Text = "REAL TIME ";
    CurrStatus->ForeColor = CurrPassColor;
    CurrTODOT->ForeColor = CurrPassColor;
    CurrPanel->BackColor = CurrPassColor;
    dataGridView->Rows[1]->DefaultCellStyle->BackColor = CurrPassColor;

    // Deteksi Current AOS
    int AOSsec = std::stoi(AOSCell.substr(0, 2)) * 3600 + \
std::stoi(AOSCell.substr(3, 2)) * 60 + std::stoi(AOSCell.substr(6, 2));

    if (CurrSec == AOSsec) {
        if ((CurrSec < AlarmDelayRange) & (DataAlarm == 0) & \
            (CurrSec == (StartAlarmSec)) & (CurrSec < (StopAlarmSec))) {
            CurrPassAlarmOn();

            int CurrDateTst = (CurrDate.Hour() * 3600) + (CurrDate.Minute() * 60) + (CurrDate.Second());
            MessageBox->Show("Pass Time = " + SecondToDurationText(CurrDateTst) + \
                " ***** Alarm Executed on " + SecondToDurationText(AOSsec));
        }
    }
}
    
```

Gambar 11. Cuplikan *source code* modul *alarm current-pass* versi pengujian.

Adapun sistem perolehan dan pengolahan data yang dilakukan dalam proses pengujian ini dijabarkan sebagai berikut

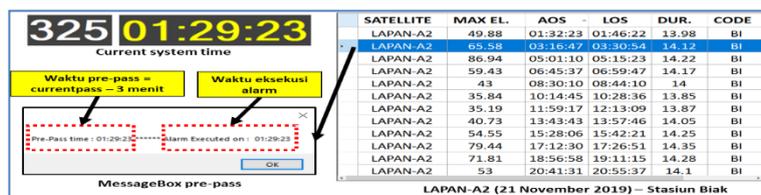
- Data yang akan diperoleh dalam pengujian ini data berupa hasil pengamatan apakah *MessageBox* muncul saat alarm menyala. Dalam pengujian ini *MessageBox* ditampilkan sebagai pengganti bunyi alarm (karena hasil akuisisi bunyi alarm tidak memungkinkan untuk didokumentasikan dan dilampirkan sebagai bahan pelaporan). Oleh karena itu, kemunculan *MessageBox* ekuivalen dengan tereksekusinya alarm.
- Pengolahan data dalam pengujian ini dilakukan pada data waktu pre-pass/current-pass dan data eksekusi alarm. Tujuan dari proses ini untuk mengukur apakah alarm menyala tepat pada waktunya. Idealnya, waktu pre-pass ataupun current-pass akan sama dengan waktu eksekusinya masing-masing. Sehingga, jika ada perbedaan diantara keduanya berarti terdapat error. Dalam pengujian ini, selisih waktu pre-pass dan waktu eksekusinya ($E_{pre-pass}$) serta selisih waktu current-pass dan waktu eksekusinya ($E_{curr-pass}$) secara rata-rata dihitung masing-masing menggunakan Persamaan (3) dan (4)

$$E_{pre-pass} = \frac{\sum_{i=1}^N (|T_{pre-pass}^i - T_{execute}^i|)}{N} \quad (3)$$

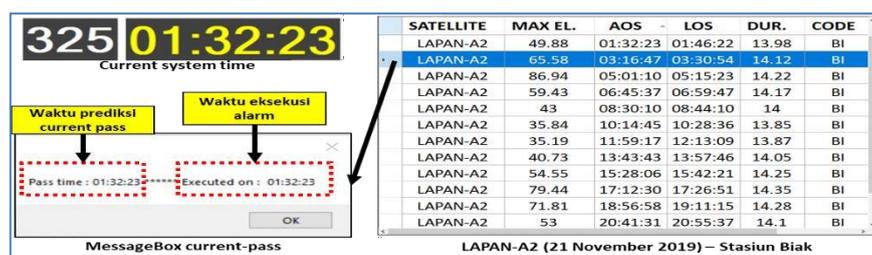
$$E_{curr-pass} = \frac{\sum_{i=1}^N (|T_{curr-pass}^i - T_{execute}^i|)}{N} \quad (4)$$

Dengan $T_{pre-pass}^i$, $T_{curr-pass}^i$, dan $T_{execute}^i$ masing-masing adalah waktu prediksi prediksi pre-pass, waktu prediksi current-pass, dan waktu eksekusi alarm. Sedangkan N adalah total jumlah pass yang diujicobakan. Dalam pengujian ini, pass yang akan diujicobakan adalah keseluruhan pass satelit LAPAN-A2 pada tanggal 21 November 2019 yang mana berjumlah 14 pass, sehingga $N = 14$.

- Hasil yang ditampilkan oleh modul alarm, mampu menampilkan *MessageBox* ketika alarm pre-pass dan alarm current pass mulai menyala. Hal ini mengindikasikan bahwa alarm telah dapat menyala pada saat terjadinya pre-pass dan current pass. Pada pengujian ini, alarm pre-pass dinyalakan 3 menit sebelum alarm current-pass menyala.



Gambar 12. Modul mampu menampilkan *MessageBox* saat pre-pass yang dapat dijadikan indikasi bahwa alarm juga telah menyala pada saat tersebut.



Gambar 13. Modul mampu menampilkan *MessageBox* saat current-pass yang dapat dijadikan indikasi bahwa alarm juga telah menyala pada saat tersebut.

4. KESIMPULAN

Berdasarkan penjelasan diatas, perangkat lunak prediktor pass satelit sangat dibutuhkan agar operator stasiun bumi dapat tepat waktu dalam mempersiapkan kebutuhan dan pengecekan sebelum satelit lewat seperti kondisi antena dan perangkat komunikasi ke satelit. Perangkat lunak ini dilengkapi dengan fitur countdown timer dan alarm agar operator mendapatkan peringatan dan dapat mempersiapkan kebutuhan sebelum terjadinya pass.

UCAPAN TERIMA KASIH

Berisikan ucapan terima kasih kepada orang-orang yang mendukung penyelesaian artikel ilmiah ini.

REFERENSI

- [1] <http://gpredict.oz9aec.net/>-diakses, 19 maret 2019
- [2] B. Stroustrup, *Programming Principles and Practice Using C++*. 2014.
- [3] D. A. Vallado and P. Crawford, "SGP4 orbit determination," 2008, doi: 10.2514/6.2008-6770.
- [4] B. Eckel, *Thinking In C++*, vol. 2, Standar. 2017.
- [5] Online Clock Counter , "Tick Counter : Create your own countdown to any date for free", (<https://www.tickcounter.com/>)-diakses 30 oktober - 4 november 2019