

Sinyal adalah suatu besaran fisis yang berubah terhadap waktu, ruang, ataupun dapat berubah terhadap variabel bebas lainnya. Ponsel harus memiliki sebuah sinyal ketika melakukan komunikasi.

Waktu atau masa menurut Kamus Besar Bahasa Indonesia adalah seluruh rangkaian saat ketika proses, perbuatan, atau keadaan berada atau berlangsung. Dalam hal ini, skala waktu merupakan interval antara dua buah keadaan/kejadian, atau bisa merupakan lama berlangsungnya suatu kejadian

Pada penelitian ini menggunakan waktu, sebagai eksperimen dan memanfaatkan data jam menit dan detik yang menjadi 3 variabel fokus untuk penentuan p dan q .

2.5 Kotlin dan Aliran Kontrol

Kotlin adalah sebuah bahasa pemrograman dengan pengetikan statis yang berjalan pada Mesin Virtual Java ataupun menggunakan kompiler LLVM yang dapat pula dikompilasikan kedalam bentuk kode sumber JavaScript. Pengembang utamanya berasal dari tim developer dari JetBrains yang bermarkas di Rusia (*FAQ - Kotlin Programming Language*, n.d.).

Pada penelitian ini operasi dan pengujian menggunakan bahasa *kotlin* yang dikompilasi atau dijalankan oleh *mobile android*. Beberapa aliran kontrolnya terdapat *if*, *when*, *for* dan *while*. Semua aliran digunakan menangani aritmatika dan konsep yang dibuat sesuai kebutuhan.

2.6 *Exception Handling*

Memeriksa semua kemungkinan kesalahan atau yang disebut *Exception Handling* untuk setiap metode karena ini dapat membuat kode tidak dapat dipahami jika setiap pemanggilan metode memeriksa semua kemungkinan kesalahan sebelum menjalankan pernyataan berikutnya. Hal ini menciptakan kebingungan antara kebenaran (memeriksa semua kemungkinan kesalahan) dan kejelasan (tidak mengacaukan aliran kode dasar dengan banyak pemeriksaan kesalahan). Jadi, untuk mengatasinya, diperlukan jalan tengah dan persyaratan ini dipenuhi oleh PENGECUALIAN.

Java menyediakan cara yang efisien untuk menangani kondisi tak terduga yang terjadi selama eksekusi program. Ketika ada kesalahan atau *bug* dalam program maka program akan segera berhenti setelah kesalahan ditemukan. Ini membuat program dalam keadaan tidak konsisten. Untuk menghindari situasi seperti itu, *Java* menggunakan *Exception Handling* yang merupakan keuntungan besar sehingga setiap kali ada kondisi luar biasa, program akan menanganinya dengan cara dan melanjutkan eksekusi program tanpa meninggalkannya dengan cara yang tidak konsisten. *java.lang*. Kelas *Throw* mampu menangani seluruh konsep pengecualian dan kesalahan. Dalam makalah ini kami fokus pada keuntungan dari penanganan *Exception* dan implementasinya dalam program untuk membuatnya kuat. *Exception* menyediakan cara yang bersih untuk memeriksa *error* tanpa kode yang berantakan. Pengecualian yang ditentukan pengguna untuk membuat program lebih dapat diandalkan. Penanganan pengecualian adalah fitur