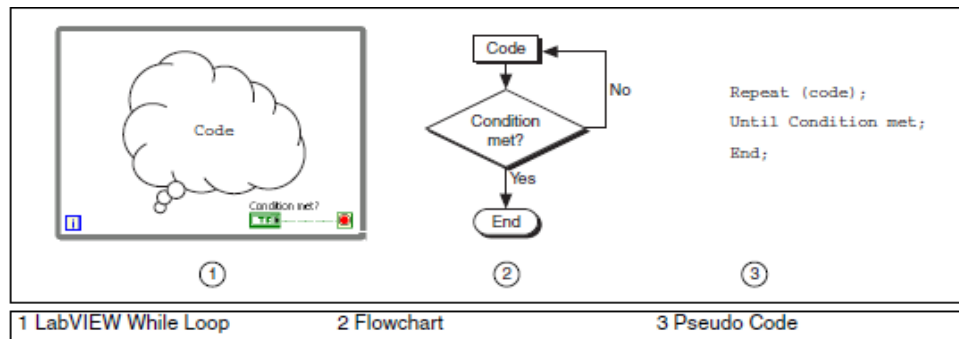


## 2.3 While Loops

**While Loop** adalah metode perulangan dimana ada kondisi yang harus dipenuhi supaya looping bisa berjalan terus. While Loop mengulangi eksekusi sub diagram didalamnya sampai terminal kondisi menerima nilai Boolean tertentu. Nilai Boolean tergantung dari sifat dari While Loop. Dengan cara klik kanan pada terminal kondisi dapat dilakukan pemilihan **Stop if True** atau **Continue if True**. While Loop akan selalu dieksekusi minimal satu kali. Jumlah pengulangan eksekusi dapat diambil dari terminal iterasi (i).

Dibawah ini merupakan ilustrasi dari While Loop pada LabVIEW, flowchart persamaan dari fungsi While Loop dan contoh kode program dari fungsi While Loop.



Gambar 2.6. While Loop



Eksekusi While loop dalam sub diagram ditentukan oleh kondisi terminal, masukan terminal, dan spesifik harga Boolean yang diterima. Kondisi terminal **Stop if True**, diperlihatkan seperti gambar disamping.



Ketika kondisi terminal **Stop if True**, While Loop akan mengeksekusi subdiagram tersebut sampai kondisi terminal yang diterimanya berharga TRUE. Anda dapat mengubah kondisi terminal tersebut menjadi **Continue if True**, dengan cara klik kanan pada terminal Stop if True lalu pilih **Continue if True** sehingga gambarnya berubah seperti yang diperlihatkan pada gambar disamping.

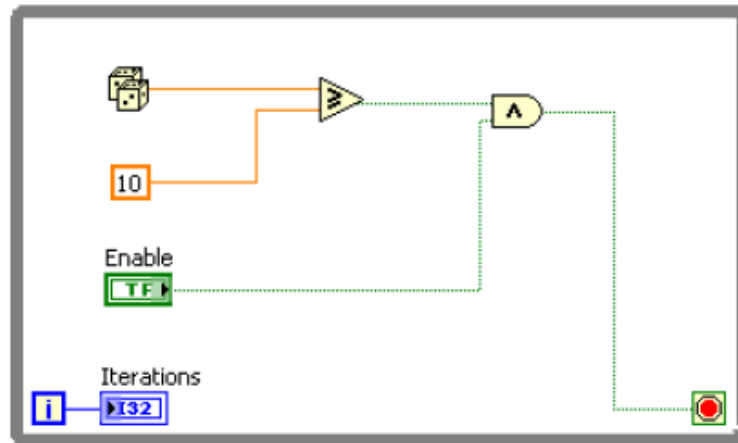
Ketika kondisi terminal **Continue if True**, While Loop akan mengeksekusi subdiagram tersebut sampai kondisi terminal yang diterimanya berharga FALSE. dan anda dapat menggunakan kedua operasi tersebut sesuai dengan operasi yang diinginkan.



Terminal iterasi (sebuah terminal keluaran), diperlihatkan seperti gambar disamping, berisi nomor dari iterasi-iterasi sampai selesai dieksekusi. Perhitungan iterasi selalu dimulai dari nol, sampai iterasi berikutnya kembali ke nol.

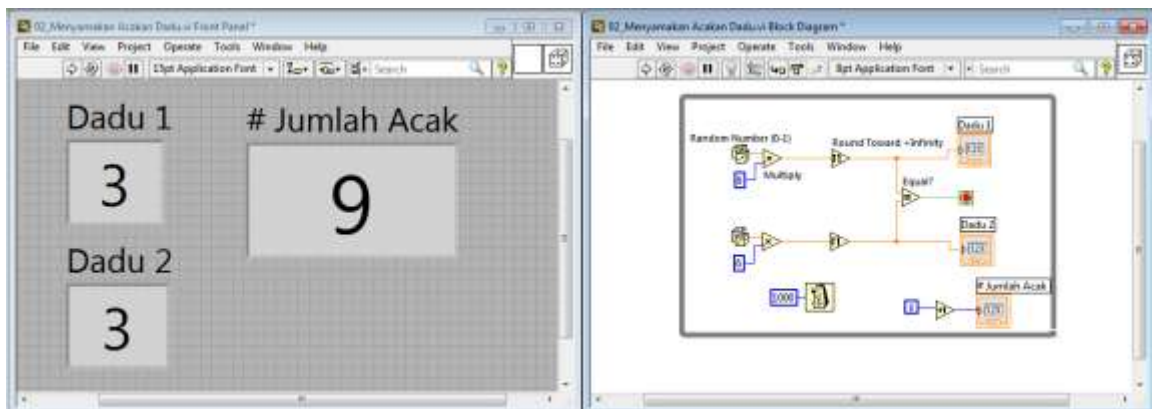
Dalam block diagram dibawah ini, While Loop akan dieksekusi sampai keluaran dari sub VI lebih besar sama dengan 10 dan kontrol **Enable**-nya adalah True.

Fungsi And akan selalu kembali True jika kedua masukannya berharga True. Jika sebaliknya akan kembali False.



Gambar 2.7. Loop Kemungkinan Tidak Terbatas.

#### Contoh 1. Menyamakan Acakan 2 buah Dadu

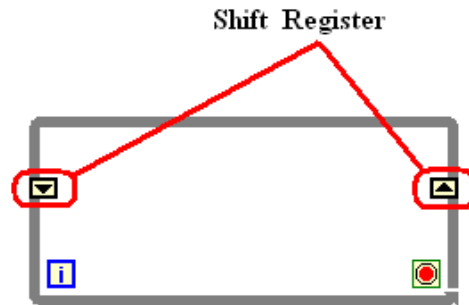


**Shift Register** (digunakan pada While Loop dan For Loop) untuk mentransfer suatu data yang digunakan atau dihitung pada iterasi sebelumnya. Anda dapat membuat Shift Register dengan meng-klik kanan pada suatu looping baik While Loop maupun For Loop.

**Shift Register** terdiri dari sepasang terminal yang berhadapan pada sisi vertical dari looping. Data yang berakhir pada setiap iterasi akan memasuki terminal di sebelah kanan sementara data iterasi sebelumnya akan muncul dari sebelah kiri.

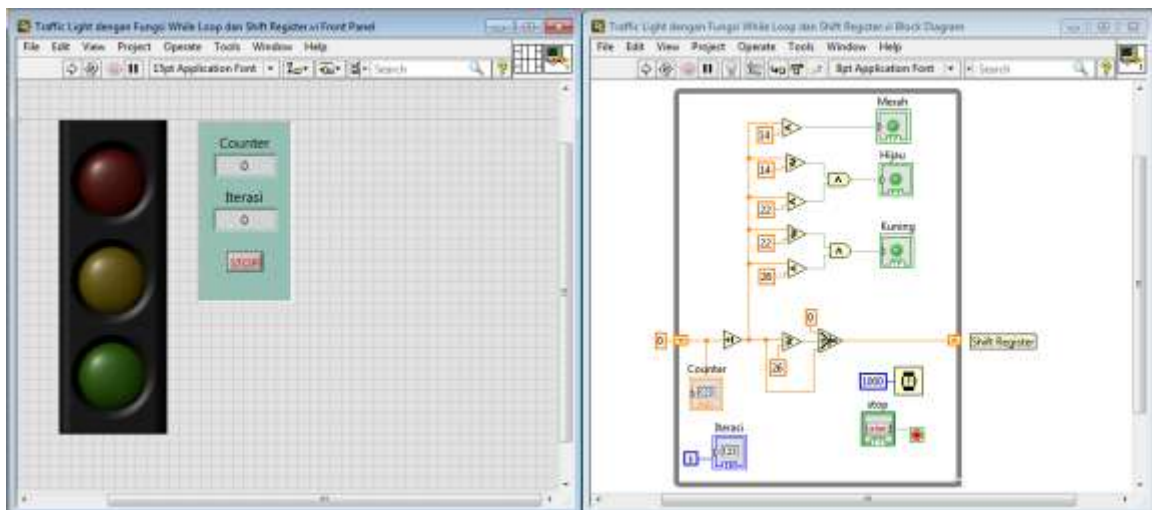
**Shift Register** dapat menampung berbagai macam tipe data baik numeric, boolean maupun string, tetapi hanya dapat dipakai 1 tipe data di setiap looping.

**Shift Register** juga dapat digunakan untuk mengingat nilai dari beberapa iterasi sebelumnya. Hal ini dapat dilakukan dengan menambah terminal di sebelah kiri. Klik kanan pada terminal kemudian pilih **Add Element**.



Gambar 2.11. Shift Register pada While Loop

## Contoh 2. Traffic Light Menggunakan Fungsi While Loop



## Tugas 3:

1. Buatlah program untuk mengontrol enam buah LED melalui *data line* agar berfungsi seperti lampu *traffic light*. Dengan aturan sebagai berikut :

MERAH		HIJAU		KUNING	MERAH
HIJAU	KUNING	MERAH			
8 detik	4 detik	1	8 detik	4 detik	1

**Keterangan :** Tentu saja Anda tidak harus menggunakan lampu LED berwarna merah, hijau dan kuning. Anda cukup menganalogikan menggunakan posisi lampu lalu lintas standar saja. [Lampu teratas adalah lampu merah, dst]