

## Reduce된 코드를 가지고 translation (IntraCfg → Flang) 테스트

### <결과>

buf[idx] 와 같은 expression은 Flang에서 Array로 번역되는데,  
할당문 왼쪽에 올 때랑 오른쪽에 올 때의 번역이 일치하지 않았음.

나머지는 번역 잘 됨.

### <수정한 것>

buf[idx]는 일반 변수와 구별되도록 잘 수정 끝.

### <세부 결과>

아래에 첨부

T1:

```
draw_slider() {  
    int v1len;  
    char wb[41];  
    v1len = 1;  
    airac_observe(wb, v1len - 1);  
    v1len++;  
}
```

<<IntraCfg>>

```
skip  
alloc(__cil_tmp3,[1*41])  
set(wb,(char *)__cil_tmp3)  
set(v1len,1)  
call(@airac_observe(StartOf(wb), v1len-1))  
skip  
set(v1len,v1len+1)  
return 0  
skip
```

<<Flang>>

```
Alooc Id Bexp Const Const  
Assign Id Nothing  
Assign Id Const  
Assign Id Bexp Lval Id Const
```

---

T2:

```
int draw_slider_slen;  
draw_slider() {  
    int v1len, v2len;  
    char wb[41];  
    draw_slider_slen = v2len = draw_slider_slen + 1;  
    v1len = v2len;  
    airac_observe(wb, v1len);  
}
```

<<IntraCfg>>

```
skip  
alloc(__cil_tmp4,[1*41])  
set(wb,(char *)__cil_tmp4)  
set(v2len,@draw_slider_slen+1)  
set(@draw_slider_slen,v2len)  
set(v1len,v2len)  
call(@airac_observe(StartOf(wb), v1len))  
skip  
return 0  
skip
```

<<Flang>>

```
Alooc Id Bexp Const Const  
Assign Id Nothing  
Assign Id Bexp Lval Id Const  
Assign Id Lval Id  
Assign Id Lval Id
```

---

T3:

```
int dev_vol[25];
    ctrl_sliders() {
        int dev_nr = 0;
        airac_observe(dev_vol, dev_nr);
        dev_nr = 1 % 0;
    }
```

```
<<IntraCfg>>
skip
set(dev_nr,0)
call(@airac_observe(StartOf(@dev_vol), dev_nr))
skip
set(dev_nr,1%0)
return 0
skip
```

```
<<Flang>>
Assign Id Const
Assign Id Bexp Const Const
```

-----

T5:

```
int dev_count, init_sliders_dev_nr, init_sliders_dev_nrs;
    int dev_vol[25];
    int dev_ind[5];
    init_sliders() {
        dev_ind[init_sliders_dev_nr] = dev_count++;
        dev_ind[init_sliders_dev_nr] = -1;
        int dev_nr_tmp = 0;
        dev_nr_tmp = dev_ind[init_sliders_dev_nrs];
        if (dev_nr_tmp)
            goto __Cont;
        airac_observe(dev_vol, dev_nr_tmp);
        __Cont:
    ;
}
```

<<IntraCfg>>

```
skip
set(tmp,@dev_count)
set(@dev_count,@dev_count+1)
set(@dev_ind[@init_sliders_dev_nr],tmp)
set(@dev_ind[@init_sliders_dev_nr],-1)
set(dev_nr_tmp,0)
set(dev_nr_tmp,@dev_ind[@init_sliders_dev_nrs])
skip
assume(!dev_nr_tmp)
call(@airac_observe(StartOf(@dev_vol), dev_nr_tmp))
skip
skip
return 0
skip
skip
set(tmp,@dev_count)
set(@dev_count,@dev_count+1)
set(@dev_ind[@init_sliders_dev_nr],tmp)
set(@dev_ind[@init_sliders_dev_nr],-1)
set(dev_nr_tmp,0)
set(dev_nr_tmp,@dev_ind[@init_sliders_dev_nrs])
skip
assume(dev_nr_tmp)
skip
return 0
skip
```

<<Flang>>

```
Assign Id Lval Id
Assign Id Bexp Lval Id Const
Assign Array Lval Id
Assign Array Const
Assign Id Const
Assign Id Lval Array
Cond Uexp Lval Id
```

```
Assign Id Lval Id
Assign Id Bexp Lval Id Const
Assign Array Lval Id
Assign Array Const
Assign Id Const
Assign Id Lval Array
Cond Lval Id
```

-----

T6:

```
int dev_count, init_sliders_dev_nr, ctrl_sliders_dev_nrs;
    int dev_vol[25];
    int dev_ind[5];
    init_sliders() {
        dev_ind[init_sliders_dev_nr] = dev_count++;
        dev_ind[init_sliders_dev_nr] = -1;
    }
```

```
ctrl_sliders() {
    int dev_nr_tmp = 0;
    dev_nr_tmp = dev_ind[ctrl_sliders_dev_nrs];
    if (dev_nr_tmp)
        goto __Cont;
    airac_observe(dev_vol, dev_nr_tmp);
    __Cont:
    ;
}
```

```
<<IntraCfg>>
skip
set(dev_nr_tmp,0)
set(dev_nr_tmp,@dev_ind[@ctrl_sliders_dev_nrs])
skip
assume(!dev_nr_tmp)
call(@airac_observe(StartOf(@dev_vol), dev_nr_tmp))
skip
skip
return 0
skip
```

```
skip
set(dev_nr_tmp,0)
set(dev_nr_tmp,@dev_ind[@ctrl_sliders_dev_nrs])
skip
assume(dev_nr_tmp)
skip
return 0
skip
```

```
<<Flang>>
Assign Id Const
Assign Id Lval Array
Cond Uexp Lval Id
```

```
Assign Id Const
Assign Id Lval Array
Cond Lval Id
```

-----