

Model Extraction

for Legacy C Programs with SCCharts

C | A | U

Kiel University

Faculty of Engineering

Department of Computer Science

C Program Domain

Arbitrary C Program

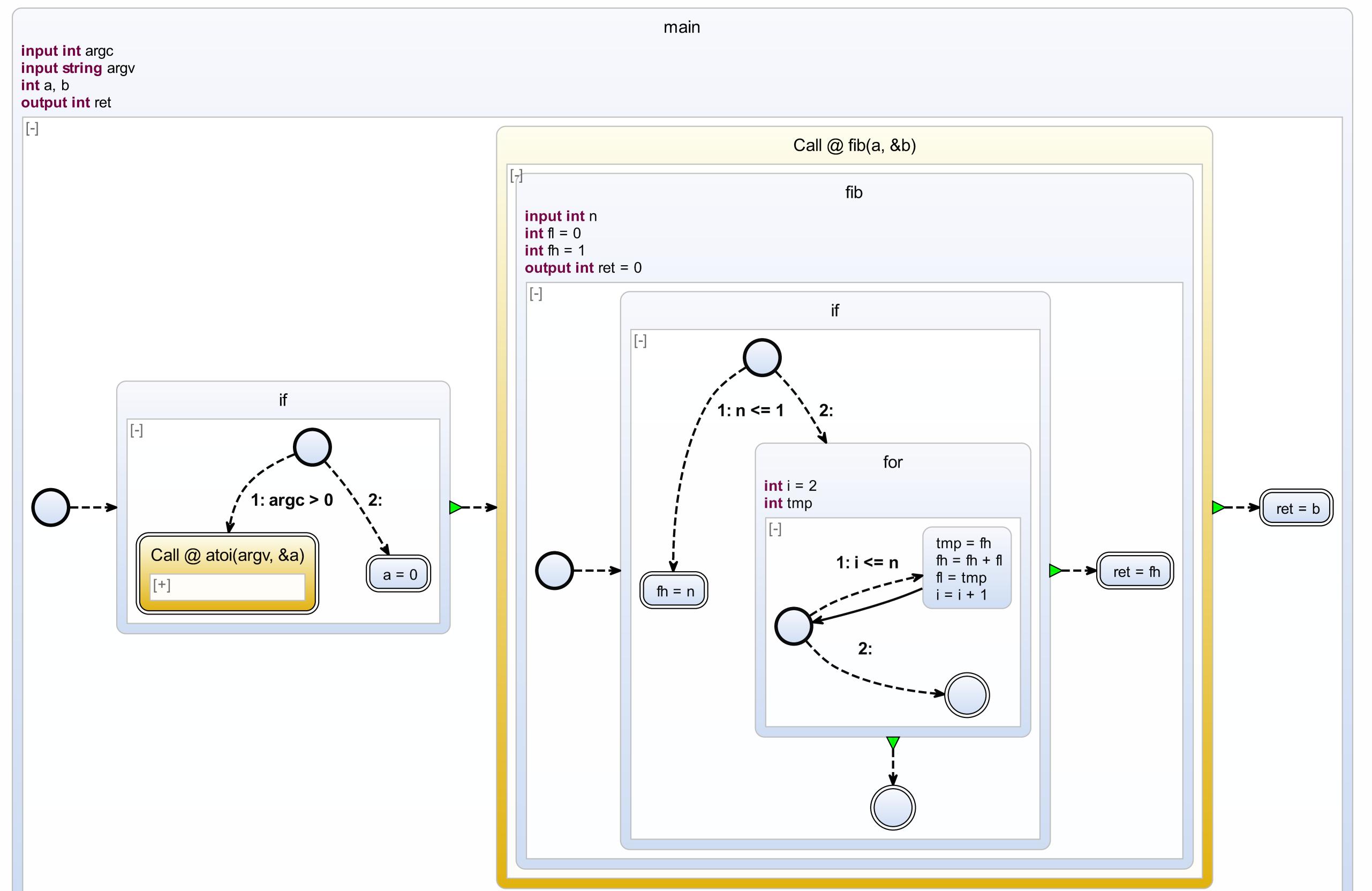
```
int main(int argc, char** argv) {  
    int a, b;  
    if (argc>0) {  
        a = atoi(argv[0]);  
    } else {  
        a = 0;  
    }  
    b = fib(a);  
    return b;  
}  
  
int fib(int n) {  
    int f1 = 0, fh = 1;  
    if (n<=1) {  
        fh = n;  
    } else {  
        for(int i=2; i<=n; i++) {  
            int tmp = fh;  
            fh += f1;  
            f1 = tmp;  
        }  
    }  
    return fh;  
}
```

Since KIELER SCCharts is based on Eclipse¹, we utilize the C/C++ Developers Toolkit² to parse arbitrary C programs and create an AST. The AST is then transformed into an SCCharts Model according to the mapping presented in [3].

¹<https://eclipse.org> ²<https://eclipse.org/cdt>

Model Domain

Derived SCChart



The SCCharts language was especially designed for safety-critical systems and introduced in [1].

C Program

SCCharts Model

Generated C Code

```
typedef struct {  
    char _G0;  
    TickData0 _call0;  
    TickData1 _call1;  
    ...  
} TickData;  
  
void reset(TickData *data) {  
    data->_G0 = 0;  
    data->_G0 = 1;  
    reset0(&data->_call0);  
    reset1(&data->_call1);  
}  
  
void tick(TickData *data) {  
    tickLogic(data);  
    data->_G0 = 0;  
    data->pg4 = data->g4;  
}  
  
In the last step, the C  
code is generated.  
Other target languages  
are also possible  
(e.g., Java, VHDL).  
}  
  
void tickLogic(TickData *data) {  
    data->g0 = data->_G0;  
    data->_cg0 = data->argc > 0;  
    data->g1 = data->g0 && data->_cg0;  
    if (data->g1) {  
        data->_call0.arg = data->argv;  
        tick0(&data->_call0);  
        data->a = data->_call0.ret;  
    }  
    data->g5 = data->pg4;  
    data->_cg5 = data->b != 0;  
    data->g6 = data->g0 && !data->_cg0;  
    if (data->g6) {  
        data->a = 0;  
    }  
    data->g2 = data->g5 && !data->_cg5 || data->g6 || data->g1;  
    if (data->g2) {  
        data->_call1.n = data->a;  
        tick1(&data->_call1);  
        data->b = data->_call1.ret;  
    }  
    data->_cg2 = data->b != 0;  
    data->g4 = data->g2 && !data->_cg2;  
    data->g3 = data->g5 && data->_cg5 || data->g2 && data->_cg2;  
    if (data->g3) {  
        data->ret = data->b;  
    }  
}
```

Code Generator

